EM Algorithms for Weighted-Data Clustering with Application to Audio-Visual Scene Analysis

Israel Dejene Gebru, Xavier Alameda-Pineda, Florence Forbes, Radu Horaud

EM Algorithms for Weighted-Data Clustering with Application to Audio-Visual Scene Analysis

Israel D. Gebru, Xavier Alameda-Pineda, Florence Forbes and Radu Horaud

Abstract—**Data clustering has received a lot of attention and numerous methods, algorithms and software packages are available. Among these techniques, parametric finite-mixture models play a central role due to their interesting mathematical properties and to the existence of maximum-likelihood estimators based on expectation-maximization (EM). In this paper we propose a new mixture model that associates a weight with each observed point. We introduce the weighted-data Gaussian mixture and we derive two EM algorithms. The first one considers a fixed weight for each observation. The second one treats each weight as a random variable following a gamma distribution. We propose a model selection method based on a minimum message length criterion, provide a weight initialization strategy, and validate the proposed algorithms by comparing them with several state of the art parametric and non-parametric clustering techniques. We also demonstrate the effectiveness and robustness of the proposed clustering technique in the presence of heterogeneous data, namely audio-visual scene analysis.**

Index Terms—**finite mixtures, expectation-maximization, weighted-data clustering, robust clustering, outlier detection, model selection, minimum message length, audio-visual fusion, speaker localization.**

I. INTRODUCTION

Finding significant groups in a set of data points is a central problem in many fields. Consequently, clustering has received a lot of attention, and many methods, algorithms and software packages are available today. Among these techniques, parametric finite mixture models play a paramount role, due to their interesting mathematical properties as well as to the existence of maximum likelihood estimators based on expectation-maximization (EM) algorithms. While the finite Gaussian mixture (GMM) [1] is the model of choice, it is extremely sensitive to the presence of outliers. Alternative robust models have been proposed in the statistical literature, such as mixtures of t-distributions [2] and their numerous variants, e.g. [3], [4], [5], [6], [7], [8]. In contrast to the Gaussian case, no closed-form solution exists for the t-distribution and tractability is maintained via the use of EM and a Gaussian scale mixture representation, $\mathcal{T}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \alpha) = \int_0^\infty \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}/w) \ \mathcal{G}(w, \alpha/2, \alpha/2) dw$, where $\boldsymbol{x}$ is an observed vector, $\mathcal{N}$ is the multivariate Gaussian distribution with

mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}/w$, and $\mathcal{G}$ is the gamma distribution of a univariate positive variable w parameterized by α. In the case of mixtures of t-distributions, with mixing coefficients π_k, $\sum_{k=1}^K \pi_k \mathcal{T}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \alpha_k)$, a latent variable w can also be introduced. Its distribution is a mixture of K gamma distributions that accounts for the component-dependent α_k [2]. Clustering is then usually performed associating a positive variable w_i, distributed as w, with **each** observed point $\boldsymbol{x}_i$. The distributions of both w_i and $\boldsymbol{x}_i$ do not depend on i. The observed data are drawn from i.i.d. variables, distributed according to the t-mixture, or one of its variants [2], [3], [4], [5], [6], [7], [8].

In this paper we propose a finite mixture model in which variable w_i is used as a weight to account for the reliability of the observed $\boldsymbol{x}_i$ and this independently on its assigned cluster. The distribution of w_i is not a gamma mixture anymore but has to depend on i to allow each data point to be potentially treated differently. In contrast to mixtures of t-distributions, it follows that the observed data are independent *but not* identically distributed. We introduce the weighted-data Gaussian mixture model (WD-GMM). We distinguish two cases, namely (i) the weights are known a priori and hence they are fixed, and (ii) the weights are modeled as variables and hence they are iteratively updated, given initial estimates. We show that in the case of fixed weights, the GMM parameters can be estimated via an extension of the standard EM which will be referred to as the *fixed weighted-data* EM algorithm (FWD-EM). Then we consider the more general case of weights that are treated as random variables. We model these variables with gamma distributions (one distribution for each variable) and we formally derive a closed-form EM algorithm which will be referred to as the *weighted-data* EM algorithm (WD-EM). While the M-step of the latter is similar to the M-step of FWD-EM, the E-step is considerably different as both the posterior probabilities (responsibilities) and the parameters of the posterior gamma distributions (the weights) are updated (E-Z-step and E-W-step). The responsibilities are computed using the Pearson type VII distribution (the reader is referred to [5] for a recent discussion regarding this distribution), also called the Arellano-Valle and Bolfarine generalized t-distribution [9], and the parameters of the posterior gamma distributions are computed from the prior gamma parameters and from the Mahalanobis distance between the data and the mixture means. Note that the weights play a different role than the responsibilities. Unlike the responsibilities, which are probabilities, the weights are random variables that can take arbitrary positive values. Their posterior means can be used as an absolute measure of the relevance of the data.

Typically, an outlying data point which is far from any cluster center will have a small weight while it may still be assigned with a significant responsibility value to the closest cluster. Responsibilities indicate which cluster center is the closest but not if any of them is close at all.

The idea of weighted-data clustering has already been proposed in the framework of non-parametric clustering methods such as K-means and spectral clustering, e.g. [10], [11], [12], [13]. These methods generally propose to incorporate prior information in the clustering process in order to prohibit atypical data (outliers) to contaminate the clusters. The idea of modeling data weights as random variables and to estimate them via EM was proposed in [14] in the particular framework of Markovian brain image segmentation. In [14] it is shown that specific expert knowledge is not needed and that the data-weight distribution guide the model towards a satisfactory segmentation. A variational EM is proposed in [14] as their formulation has no closed form. In this paper we build on the idea that, instead of relying on prior information about atypical data, e.g. [10], [11], [12], [13], we devise a novel EM algorithm that updates the weight distributions. The proposed method belongs to the *robust clustering* category of mixture models because observed data that are far away from the cluster centers have little influence on the estimation of the means and covariances.

An important feature of mixture based clustering methods is to perform model selection on the premise that the number of components K in the mixture corresponds to the number of clusters in the data. Traditionally, model selection is performed by obtaining a set of candidate models for a range of values of K (assuming that the true value is in this range). The number of components is selected by minimizing a model selection criteria, such as the Bayesian inference criterion (BIC), minimum message length (MML), Akaike's information criteria (AIC) to cite just a few [1], [15]. The disadvantage of these methods is twofold. Firstly, a whole set of candidates has to be obtained and problems associated with running EM many times may emerge. Secondly, they provide a number of components that optimally approximate the density and not the true number of clusters present in the data. More recently, there seems to be a consensus among mixture model practitioners that a well-founded and computationally efficient model selection strategy is to start with a large number of components and to merge them [16]. [15] proposes a practical algorithm that starts with a very large number of components (thus making the algorithm robust to initialization), iteratively annihilates components, redistributes the observations to the other components, and terminates based on the MML criterion. [17] starts with an overestimated number of components using BIC, and then merges them hierarchically according to an entropy criterion. More recently [18] proposes a similar method that merges components based on measuring their pair-wise overlap.

Another trend in handling the issue of finding the proper number of components is to consider Bayesian non-parametric mixture models. This allows the implementation of mixture models with an infinite number of components via the use of Dirichlet process mixture models. In [19], [20] an infinite Gaussian mixture (IGMM) is presented with a computationally intensive Markov Chain Monte Carlo implementation. At first glance, IGMM may appear similar to FWD-EM. However, these two algorithms are quite different. While IGMM is fully Bayesian the proposed FWD-EM is not, in the sense that no priors are assumed on the parameters, typically the means and covariance matrices. IGMM implies Student predictive distributions while FWD-EM involves only Gaussian distributions.

More recently, more flexibility in the cluster shapes has been allowed by considering infinite mixture of infinite Gaussian mixtures (I^2GMM) [21]. The flexibility is however limited to a cluster composed of sub-clusters of identical shapes and orientations, which may alter the performance of this approach. Altogether, IGMM and I^2GMM are not designed to handle outliers, as illustrated in Section VIII, Figs. 2-f and 2-g. Infinite Student mixture models have also been considered [22], but inference requires a variational Bayes approximation which generates additional computational complexity.

Bayesian non-parametrics, although promising techniques, require a fully Bayesian setting. The latter, however, induces additional complexity for handling priors and hyper-priors, especially in a multi-variate context. In contrast, our latent variable approach allows exact inference. With respect to model selection, we therefore propose to extend the method of [15] to weighted-data mixtures. We formally derive an MML criterion for the weighted-data mixture model and we plug this criterion into an efficient algorithm which, starting with a large number of components, simultaneously estimates the model parameters, the posterior probabilities of the weights and the optimal number of components.

We also propose to apply the proposed weighted-data robust clustering method to the problem of fusing auditory and visual information. This problem arises when the task is, e.g. to detect a person that is both seen and heard, such as an active speaker. Single-modality signals – vision-only or audio-only – are often either weak or ambiguous, and it may be useful to combine information from different sensors, e.g. cameras and microphones. There are several difficulties associated with audio-visual fusion from a data clustering perspective: the two sensorial modalities (i) live in different spaces, (ii) are contaminated by different types of noise with different distributions, (iii) have different spatiotemporal distributions, and (iv) are perturbed by different physical phenomena, e.g. acoustic reverberations, lighting conditions, etc. For example, a speaker may face the camera while he/she is silent and may emit speech while he/she turns his/her face away from the camera. Speech signals have sparse spectro-temporal structure and they are mixed with other sound sources, such as music or background noise. Speaker faces may be totally or partially occluded, in which case face detection and localization is extremely unreliable. We show that the proposed method is well suited to find audio-visual clusters and to discriminate between speaking and silent people.

The remainder of this paper is organized as follows. Section II outlines the weighted-data mixture model; Section III sketches the FWD-EM algorithm. Weights modeled with ran-

dom variables are introduced in Section IV and the WD-EM is described in detail in Section V. Section VI details how to deal with an unknown number of clusters and Section VII addresses the issue of algorithm initialization. In Section VIII the proposed algorithms are tested and compared with several other parametric and non-parametric clustering methods. Section IX addresses clustering of audio-visual data. Section X concludes the paper. Additional results and videos are available online.[1]

II. GAUSSIAN MIXTURE WITH WEIGHTED DATA

In this Section, we present the intuition and the formal definition of the proposed weighted-data model. Let $\boldsymbol{x} \in \mathbb{R}^d$ be a random vector following a multivariate Gaussian distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$, namely $p(\boldsymbol{x}|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, with the notation $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$. Let $w > 0$ be a weight indicating the relevance of the observation $\boldsymbol{x}$. Intuitively, higher the weight w, stronger the impact of $\boldsymbol{x}$. The weight can therefore be incorporated into the model by "*observing $\boldsymbol{x}$ w times*". In terms of the likelihood function, this is equivalent to raise $p(\boldsymbol{x}; \boldsymbol{\theta})$ to the power w, i.e. $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^w$. However, the latter is not a probability distribution since it does not integrate to one. It is straightforward to notice that $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})^w \propto \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}/w)$. Therefore, w plays the role of the precision and is different for each datum $\boldsymbol{x}$. Subsequently, we write:

$$\hat{p}(\boldsymbol{x}; \boldsymbol{\theta}, w) = \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{\mu}, \frac{1}{w}\boldsymbol{\Sigma}\right), \quad (1)$$

from which we derive a mixture model with K components:

$$\tilde{p}(\boldsymbol{x}; \boldsymbol{\Theta}, w) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{\mu}_k, \frac{1}{w}\boldsymbol{\Sigma}_k\right), \quad (2)$$

where $\boldsymbol{\Theta} = \{\pi_1, \dots, \pi_K, \boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K\}$ are the mixture parameters, $\pi_1, \dots, \pi_K$ are the mixture coefficients satisfying $\pi_k \geq 0$ and $\sum_{k=1}^{K} \pi_k = 1$, $\boldsymbol{\theta}_k = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}$ are the parameters of the k-th component and K is the number of components. We will refer to the model in (2) as the *weighted-data Gaussian mixture model* (WD-GMM). Let $\boldsymbol{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$ be the observed data and $\boldsymbol{W} = \{w_1, \dots, w_n\}$ be the weights associated with $\boldsymbol{X}$. We assume each $\boldsymbol{x}_i$ is independently drawn from (2) with $w = w_i$. The observed-data log-likelihood is:

$$\ln \tilde{p}(\boldsymbol{X}; \boldsymbol{\Theta}, \boldsymbol{W}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \frac{1}{w_i}\boldsymbol{\Sigma}_k\right) \right). \quad (3)$$

It is well known that direct maximization of the log-likelihood function is problematic in case of mixtures and that the expected complete-data log-likelihood must be considered instead. Hence, we introduce a set of n hidden (assignment) variables $\boldsymbol{Z} = \{z_1, \dots, z_n\}$ associated with the observed variables $\boldsymbol{X}$ and such that $z_i = k$, $k \in \{1, \dots, K\}$ if and only if $\boldsymbol{x}_i$ is generated by the k-th component of the mixture. In the following we first consider a fixed (given) number of mixture components K, we then extend the model to an unknown K, thus estimating the number of components from the data.

III. EM WITH FIXED WEIGHTS

The simplest case is when the weight values are provided at algorithm initialization, either using some prior knowledge or estimated from the observations (e.g. Section VII), and are then kept fixed while alternating between the expectation and maximization steps. In this case, the expected complete-data log-likelihood is:

$$\mathcal{Q}_c(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}) = \mathrm{E}_{P(\boldsymbol{Z}|\boldsymbol{X}; \boldsymbol{W}, \boldsymbol{\Theta}^{(r)})}[\ln P(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{W}, \boldsymbol{\Theta})], \quad (4)$$

where $\mathrm{E}_P[\cdot]$ denotes the expectation with respect to the distribution P. The $(r+1)$-th EM iteration consists of two steps namely, the evaluation of the posterior distribution given the current model parameters $\boldsymbol{\Theta}^{(r)}$ and the weights $\boldsymbol{W}$ (E-step), and the maximization of (4) with respect to $\boldsymbol{\Theta}$ (M-step):

$$\boldsymbol{\Theta}^{(r+1)} = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}_c(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}). \quad (5)$$

It is straightforward to show that this yields the following FWD-EM algorithm:

A. The E-Step

The posteriors $\eta_{ik}^{(r+1)} = p(z_i = k | \boldsymbol{x}_i; w_i, \boldsymbol{\Theta}^{(r)})$ are updated with:

$$\eta_{ik}^{(r+1)} = \frac{\pi_k^{(r)} \hat{p}(\boldsymbol{x}_i; \boldsymbol{\theta}_k^{(r)}, w_i)}{\tilde{p}(\boldsymbol{x}_i; \boldsymbol{\Theta}^{(r)}, w_i)}, \quad (6)$$

where $\hat{p}$ and $\tilde{p}$ are defined in (1) and (2).

B. The M-Step

Expanding (4) we get:

$$\mathcal{Q}_c(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \eta_{ik}^{(r+1)} \ln \pi_k \mathcal{N}\left(\boldsymbol{x}_i; \boldsymbol{\mu}_k; \frac{1}{w_i}\boldsymbol{\Sigma}_k\right)$$
$$\stackrel{\Theta}{=} \sum_{i=1}^{n} \sum_{k=1}^{K} \eta_{ik}^{(r+1)} \left(\ln \pi_k - \ln |\boldsymbol{\Sigma}_k|^{1/2} \right.$$
$$\left. - \frac{w_i}{2}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k) \right), \quad (7)$$

where $\stackrel{\Theta}{=}$ denotes equality up to a constant that does not depend on $\boldsymbol{\Theta}$. By canceling out the derivatives with respect to the model parameters, we obtain the following update formulae for the mixture proportions, means, and covariances matrices:

$$\pi_k^{(r+1)} = \frac{1}{n} \sum_{i=1}^{n} \eta_{ik}^{(r+1)}, \quad (8)$$

$$\boldsymbol{\mu}_k^{(r+1)} = \frac{\sum_{i=1}^{n} w_i \eta_{ik}^{(r+1)} \boldsymbol{x}_i}{\sum_{i=1}^{n} w_i \eta_{ik}^{(r+1)}}, \quad (9)$$

$$\boldsymbol{\Sigma}_k^{(r+1)} = \frac{\sum_{i=1}^{n} w_i \eta_{ik}^{(r+1)} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r+1)}\right) \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r+1)}\right)^\top}{\sum_{i=1}^{n} \eta_{ik}^{(r+1)}}. \quad (10)$$

IV. MODELING THE WEIGHTS

As we already remarked, the weights play the role of precisions. The notable difference between standard finite mixture models and the proposed model is that there is a different weight w_i, hence a different precision, associated with *each* observation $\boldsymbol{x}_i$. Within a Bayesian formalism, the weights $\boldsymbol{W}$ may be treated as random variables, rather than being fixed in advance, as in the previous case. Since (1) is a Gaussian, a convenient choice for the prior on w, $p(w)$ is the conjugate prior of the precision with known mean, i.e. a gamma distribution. This ensures that the weight posteriors are gamma distributions as well. Summarizing we have:

$$P(w;\boldsymbol{\phi}) = \mathcal{G}(w;\alpha,\beta) = \Gamma(\alpha)^{-1}\beta^\alpha w^{\alpha-1}e^{-\beta w}, \quad (11)$$

where $\mathcal{G}(w;\alpha,\beta)$ is the gamma distribution, $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}\mathrm{d}t$ is the gamma function, and $\boldsymbol{\phi} = \{\alpha,\beta\}$ are the parameters of the prior distribution of w. The mean and variance of the random variable w are given by:

$$\mathrm{E}[w] = \alpha/\beta, \quad (12)$$
$$\mathrm{var}[w] = \alpha/\beta^2. \quad (13)$$

V. EM WITH RANDOM WEIGHTS

In this section we derive the WD-EM algorithm associated to a model in which the weights are treated as random variables following (11). The gamma distribution of each w_i is assumed to be parameterized by $\boldsymbol{\phi}_i = \{\alpha_i,\beta_i\}$. Within this framework, the expectation of the complete-data log-likelihood is computed over both the assignment and weight variables:

$$\mathcal{Q}_\mathrm{R}\left(\boldsymbol{\Theta},\boldsymbol{\Theta}^{(r)}\right) = \mathrm{E}_{P(\boldsymbol{Z},\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}^{(r)},\boldsymbol{\Phi})}[\ln P(\boldsymbol{Z},\boldsymbol{W},\boldsymbol{X};\boldsymbol{\Theta},\boldsymbol{\Phi})], \quad (14)$$

where we used the notation $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_1,\ldots,\boldsymbol{\phi}_n\}$. We notice that the posterior distribution factorizes on i:

$$P(\boldsymbol{Z},\boldsymbol{W}|\boldsymbol{X};\boldsymbol{\Theta}^{(r)},\boldsymbol{\Phi}) = \prod_{i=1}^n P(z_i,w_i|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$$

and each one of these factors can be be decomposed as:

$$P(z_i,w_i|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i) = P(w_i|z_i,\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)P(z_i|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i), \quad (15)$$

where the two quantities on the right-hand side of this equation have closed-form expressions. The computation of each one of these two expressions leads to two sequential steps, the E-W-step and the E-Z-step, of the expectation step of the proposed algorithm.

A. The E-Z Step

The marginal posterior distribution of z_i is obtained by integrating (15) over w_i. As previously, we denote the responsibilities with $\eta_{ik}^{(r+1)} = P(z_i = k|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$. The integration computes:

$$\eta_{ik}^{(r+1)} = \int P\left(z_i = k,w_i|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i\right)\mathrm{d}w_i$$
$$\overset{k}{\propto} \int \pi_k^{(r)} P\left(\boldsymbol{x}_i|z_i = k,w_i;\boldsymbol{\Theta}^{(r)}\right) P(w_i;\boldsymbol{\phi}_i)\,\mathrm{d}w_i$$
$$= \int \pi_k^{(r)} \hat{p}\left(\boldsymbol{x}_i;\boldsymbol{\theta}_k^{(r)},w_i\right) \mathcal{G}(w_i;\alpha_i,\beta_i)\,\mathrm{d}w_i$$
$$\propto \pi_k^{(r)} \mathcal{P}(\boldsymbol{x}_i;\boldsymbol{\mu}_k^{(r)},\boldsymbol{\Sigma}_k^{(r)},\alpha_i,\beta_i), \quad (16)$$

where $\mathcal{P}(\boldsymbol{x}_i;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k,\alpha_i,\beta_i)$ denotes the Pearson type VII probability distribution function, which can be seen as a generalization of the t-distribution:

$$\mathcal{P}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma},\alpha,\beta) =$$
$$\frac{\Gamma(\alpha + d/2)}{|\boldsymbol{\Sigma}|^{1/2}\,\Gamma(\alpha)\,(2\pi\beta)^{d/2}}\left(1 + \frac{\|\boldsymbol{x}-\boldsymbol{\mu}\|_{\boldsymbol{\Sigma}}^2}{2\beta}\right)^{-(\alpha+\frac{d}{2})} \quad (17)$$

B. The E-W Step

The posterior distribution of w_i, namely $p(w_i|z_i = k,\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$ is a gamma distribution, because it is the conjugate prior of the precision of the Gaussian distribution. Therefore, we only need to compute the parameters of the posterior gamma distribution:

$$P(w_i|z_i = k,\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$$
$$\overset{w_i}{\propto} P(\boldsymbol{x}_i|z_i = k,w_i;\boldsymbol{\Theta}^{(r)})P(w_i;\boldsymbol{\phi}_i)$$
$$= \mathcal{N}(\boldsymbol{x}_i;\boldsymbol{\mu}_k^{(r)},\boldsymbol{\Sigma}_k^{(r)}/w_i)\,\mathcal{G}(w_i;\alpha_i,\beta_i)$$
$$= \mathcal{G}(w_i;a_i^{(r+1)},b_{ik}^{(r+1)}), \quad (18)$$

where the parameters of the posterior gamma distribution are evaluated with:

$$a_i^{(r+1)} = \alpha_i + \frac{d}{2}, \quad (19)$$
$$b_{ik}^{(r+1)} = \beta_i + \frac{1}{2}\left\|\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r)}\right\|_{\boldsymbol{\Sigma}_k^{(r)}}^2 \quad (20)$$

The conditional mean of w_i, namely $\overline{w}_{ik}^{(r+1)}$, can then be evaluated with:

$$\overline{w}_{ik}^{(r+1)} = \mathrm{E}_{P(w_i|z_i=k,\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)}[w_i] = \frac{a_i^{(r+1)}}{b_{ik}^{(r+1)}}. \quad (21)$$

While estimating the weights themselves is not needed by the algorithm, it is useful to evaluate them in order to fully characterize the observations and to discriminate between inliers and outliers. First notice that the marginal posterior distribution of w_i is a mixture of gamma distributions:

$$p(w_i|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$$
$$= \sum_{k=1}^K p(w_i|z_i = k,\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)p(z_i = k|\boldsymbol{x}_i;\boldsymbol{\Theta}^{(r)},\boldsymbol{\phi}_i)$$
$$= \sum_{k=1}^K \mathcal{G}(w_i;a_i^{(r+1)},b_{ik}^{(r+1)})\,\eta_{ik}^{(r+1)}, \quad (22)$$

and therefore the posterior mean of w_i is evaluated with:

$$\overline{w}_i^{(r+1)} = \mathrm{E}[w_i|\boldsymbol{x}_i; \boldsymbol{\Theta}^{(r)}, \boldsymbol{\phi}_i] = \sum_{k=i}^{K} \eta_{ik}^{(r+1)} \overline{w}_{ik}^{(r+1)}. \quad (23)$$

By inspection of (19), (20), and (21) it is easily seen that the value of $\overline{w}_i$ decreases as the distance between the cluster centers and observation $\boldsymbol{x}_i$ increases. Importantly, the evaluation of $\overline{w}_i$ enables outlier detection. Indeed, an outlier is expected to be far from all the clusters, and therefore all $\overline{w}_{ik}$ will be small, leading to a small value of $\overline{w}_i$. It is worth noticing that this is not possible using only the responsibilities η_{ik}, since they are normalized by definition, and therefore their value is not an absolute measure of the datum's relevance, but only a relative measure of it.

C. The Maximization Step

This step maximizes the expected complete-data log-likelihood over the mixture parameters. By expanding (14), we have:

$$\mathcal{Q}_{\mathrm{R}}\big(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}\big)$$

$$\overset{\boldsymbol{\Theta}}{=} \sum_{i=1}^{n} \sum_{k=1}^{K} \int_{w_i} \eta_{ik}^{(r+1)} \ln \pi_k \mathcal{N}\left(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \frac{1}{w_i}\boldsymbol{\Sigma}_k\right)$$
$$\times \, p(w_i|\boldsymbol{x}_i, z_i = k, \boldsymbol{\Theta}^{(r)}, \boldsymbol{\phi}_i)\mathrm{d}w_i$$
$$= \sum_{i=1}^{n} \sum_{k=1}^{K} \eta_{ik}^{(r+1)} \Big(\ln \pi_k - \ln |\boldsymbol{\Sigma}_k|^{1/2}$$
$$- \frac{\overline{w}_{ik}^{(r+1)}}{2}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)\Big). \quad (24)$$

The parameter updates are obtained from canceling out the derivatives of the expected complete-data log-likelihood (24). As with standard Gaussian mixtures, all the updates are closed-form expressions:

$$\pi_k^{(r+1)} = \frac{1}{n}\sum_{i=1}^{n}\eta_{ik}^{(r+1)}, \quad (25)$$

$$\boldsymbol{\mu}_k^{(r+1)} = \frac{\displaystyle\sum_{i=1}^{n}\overline{w}_{ik}^{(r+1)}\eta_{ik}^{(r+1)}\boldsymbol{x}_i}{\displaystyle\sum_{i=1}^{n}\overline{w}_{ik}^{(r+1)}\eta_{ik}^{(r+1)}}, \quad (26)$$

$$\boldsymbol{\Sigma}_k^{(r+1)} = \frac{\displaystyle\sum_{i=1}^{n}\eta_{ik}^{(r+1)}\overline{w}_{ik}^{(r+1)}\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r+1)}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r+1)}\right)^\top}{\displaystyle\sum_{i=1}^{n}\eta_{ik}^{(r+1)}}. \quad (27)$$

It is worth noticing that the M-step of the WD-EM algorithm is very similar to the M-step of the FWD-EM algorithm (section III). Indeed, the above iterative formulas, (25), (26), (27) are identical to the formulas (8), (9), (10), except that the fixed weights w_i are here replaced with the posterior means of the random weights, $\overline{w}_{ik}^{(r+1)}$.

VI. ESTIMATING THE NUMBER OF COMPONENTS

So far it has been assumed that the number of mixture components K is provided in advance. This assumption is unrealistic for most real-world applications. In this Section we propose to extend the method and algorithm proposed in [15] to the weighted-data clustering model. An interesting feature of this model selection method is that it does not require parameter estimation for many different values of K, as it would be the case with the Bayesian information criterion (BIC) [23]. Instead, the algorithm starts with a large number of components and iteratively deletes components as they become irrelevant. Starting with a large number of components has the additional advantage of making the algorithm robust to initialization. Formally, the parameter estimation problem is cast into a transmission encoding problem and the criterion is to minimize the expected length of the message to be transmitted:

$$\mathrm{length}(\boldsymbol{X}, \boldsymbol{\Theta}) = \mathrm{length}(\boldsymbol{\Theta}) + \mathrm{length}(\boldsymbol{X}|\boldsymbol{\Theta}). \quad (28)$$

In this context, the observations and the parameters have to be quantized to finite precision before the transmission. This quantization sets a trade off between the two terms of the previous equation. Indeed, when truncating to high precision, $\mathrm{length}(\boldsymbol{\Theta})$ may be long, but $\mathrm{length}(\boldsymbol{X}|\boldsymbol{\Theta})$ will be short, since the parameters fit well the data. Conversely, if the quantization is coarse, $\mathrm{length}(\boldsymbol{\Theta})$ may be short, but $\mathrm{length}(\boldsymbol{X}|\boldsymbol{\Theta})$ will be long. The optimal quantization step can be found by means of the Taylor approximation [15]. In that case, the optimization problem corresponding to the *minimum message length* (MML) criterion, is:

$$\boldsymbol{\Theta}_{\mathrm{MML}} = \underset{\boldsymbol{\Theta}}{\mathrm{argmin}} \Big\{ -\log \mathrm{P}(\boldsymbol{\Theta}) - \log \mathrm{P}(\boldsymbol{X}|\boldsymbol{\Theta}, \boldsymbol{\Phi})$$
$$+ \frac{1}{2}\log|\mathbf{I}(\boldsymbol{\Theta})| + \frac{\mathcal{D}(\boldsymbol{\Theta})}{2}\Big(1 + \log\frac{1}{12}\Big)\Big\}, \quad (29)$$

where $\mathbf{I}(\boldsymbol{\Theta}) = -\mathrm{E}\{D_{\boldsymbol{\Theta}}^2 \log \mathrm{P}(\boldsymbol{X}|\boldsymbol{\Theta})\}$ is the *expected* Fisher information matrix (FIM) and $\mathcal{D}(\boldsymbol{\Theta})$ denotes the dimensionality of the model, namely the dimension of the parameter vector $\boldsymbol{\Theta}$. Since the minimization (29) does not depend on the weight parameters, $\boldsymbol{\Phi}$ will be omitted for simplicity.

In our particular case, as in the general case of mixtures, the Fisher information matrix cannot be obtained analytically. Indeed, the direct optimization of the log-likelihood does not lead to closed-form solutions. Nevertheless, it was noticed that the *complete* FIM upper bounds the FIM [15], and that the expected complete-data log-likelihood lower bounds the log-likelihood. This allows us to write the following equivalent optimization problem:

$$\boldsymbol{\Theta}_{\mathrm{MML}} = \underset{\boldsymbol{\Theta}}{\mathrm{argmin}} \Big\{ -\log \mathrm{P}(\boldsymbol{\Theta}) - \log \mathcal{Q}_{\mathrm{R}}\big(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}\big)$$
$$+ \frac{1}{2}\log|\mathbf{I}_c(\boldsymbol{\Theta})| + \frac{\mathcal{D}(\boldsymbol{\Theta})}{2}\Big(1 + \log\frac{1}{12}\Big)\Big\}, \quad (30)$$

where $\mathbf{I}_c$ denotes the expected complete-FIM and $\mathcal{Q}_{\mathrm{R}}$ is evaluated with (24).

As already mentioned, because there is a different weight w_i for each observation i, the observed data are not identically

distributed and our model cannot be considered a classical mixture model. For this reason, the algorithm proposed in [15] cannot be applied directly to our model. Indeed, in the proposed WD-GMM setting, the complete-FIM is:

$$\mathbf{I}_c(\boldsymbol{\Theta}) = \mathrm{diag}\left(\pi_1 \sum_{i=1}^n \mathbf{I}_i(\boldsymbol{\theta}_1), \dots, \pi_K \sum_{i=1}^n \mathbf{I}_i(\boldsymbol{\theta}_K), n\mathbf{M}\right) \quad (31)$$

where $\mathbf{I}_i(\boldsymbol{\theta}_k) = -\mathrm{E}\{D_{\boldsymbol{\theta}_k}^2 \log \mathcal{P}(\boldsymbol{x}_i|\boldsymbol{\theta}_k, \alpha_i, \beta_i)\}$ is the Fisher information matrix for the i-th observation with respect to the parameter vector $\boldsymbol{\theta}_k$ (mean and the covariance) of the k-th component, $\mathcal{P}$ is defined in (17), and $\mathbf{M}$ is the Fisher information matrix of the multinomial distribution, namely the diagonal matrix $\mathrm{diag}(\pi_1^{-1}, \dots, \pi_K^{-1})$. We can evaluate $|\mathbf{I}_c(\boldsymbol{\Theta})|$ from (31):

$$|\mathbf{I}_c(\boldsymbol{\Theta})| = n^{K(M+1)}|\mathbf{M}|\prod_{k=1}^K \pi_k^M \left|\frac{1}{n}\sum_{i=1}^n \mathbf{I}_i(\boldsymbol{\theta}_k)\right|, \quad (32)$$

where M denotes the number of free parameters of each component. For example, $M = 2d$ when using diagonal covariance matrices or $M = d(d+3)/2$ when using full covariance matrices.

Importantly, one of the main advantages of the methodology proposed in [15] is that one has complete freedom to choose a prior distribution on the parameters, $\mathrm{P}(\boldsymbol{\Theta})$. In our case, inspired by (32), we select the following prior distributions for the parameters:

$$\mathrm{P}(\boldsymbol{\theta}_k) \propto \left|\frac{1}{n}\sum_{i=1}^n \mathbf{I}_i(\boldsymbol{\theta}_k)\right|^{-\frac{1}{2}}, \quad (33)$$

$$\mathrm{P}(\pi_1, \dots, \pi_K) \propto |\mathbf{M}|^{-\frac{1}{2}}. \quad (34)$$

By substitution of (32)–(34) into (30) we obtain the following optimization problem:

$$\boldsymbol{\Theta}_{\mathrm{MML}} = \underset{\boldsymbol{\Theta}}{\mathrm{argmin}}\Big\{\frac{M}{2}\sum_{k=1}^K \log \pi_k - \log \mathcal{Q}_{\mathrm{R}}\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}\right)$$
$$+ \frac{K(M+1)}{2}\left(1 + \log\frac{n}{12}\right)\Big\}, \quad (35)$$

where we used $\mathcal{D}(\boldsymbol{\Theta}) = K(M+1)$.

One may notice that (35) does not make sense (diverges) if any of the π_k's is allowed to be null. However, in the current length coding framework, there is no point in transmitting the parameters of an empty component. Therefore, we only focus on the non-empty components, namely those components for which $\pi_k > 0$. Let $\mathcal{K}^+$ denote the index set of non-empty components and let $K^+ = |\mathcal{K}^+|$ be its cardinality. We can rewrite (35) as:

$$\boldsymbol{\Theta}_{\mathrm{MML}} = \underset{\boldsymbol{\Theta}}{\mathrm{argmin}}\Big\{\frac{M}{2}\sum_{k\in\mathcal{K}^+} \log \pi_k - \log \mathcal{Q}_{\mathrm{R}}\left(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(r)}\right)$$
$$+ \frac{K^+(M+1)}{2}\left(1 + \log\frac{n}{12}\right)\Big\}. \quad (36)$$

The above minimization problem can be solved by modifying the EM algorithm described in Section V (notice that there is an equivalent derivation for the fixed-weigth EM

algorithm described in Section III). Indeed, we remark that the minimization (36) is equivalent to using a symmetric improper Dirichlet prior for the proportions with exponent $-M/2$. Moreover, since the optimization function for the parameters of the Gaussian components is the same (equivalently, we used a flat prior for the mean vector and covariance matrix), their estimation formulas (26) and (27) still hold. Therefore, we only need to modify the estimation of the mixture proportions, namely:

$$\pi_k = \frac{\max\left\{0, \sum_{i=1}^n \eta_{ik} - \frac{M}{2}\right\}}{\sum_{k'=1}^K \max\left\{0, \sum_{i=1}^n \eta_{ik'} - \frac{M}{2}\right\}}. \quad (37)$$

The $\max$ operator in (37) verifies whether the k-th component is supported by the data. When one of the components becomes too weak, i.e. the required minimum support $M/2$ cannot be obtained from the data, this component is annihilated. In other words, its parameters will not be estimated, since there is no need in transmitting them. One has to be careful in this context, since starting with a large value of K may lead to several empty components. In order to avoid this singular situation, we adopt the component-wise EM procedure (CEM) [24], as proposed in [15] as well. Intuitively, we run both E and M steps for one component, before moving to the next component. More precisely, after running the E-Z and E-W steps for the component k, its parameters are updated if $k \in \mathcal{K}^+$, otherwise the component is annihilated if $k \notin \mathcal{K}^+$. The rationale behind this procedure is that, when a component is annihilated its probability mass is immediately redistributed among the remaining components. Summarizing, CEM updates the components one by one, whereas the classical EM simultaneously updates all the components.

The proposed algorithm is outlined in Algorithm 1. In practice, an upper and a lower number of components, K_{high} and K_{low}, are provided. Each iteration r of the algorithm consists of component-wise E and M steps. If needed, some of the components are annihilated, and the parameters are updated accordingly, until the relative length difference is below a threshold, $|\Delta\mathbf{LEN}_{\mathrm{MML}}^{(r)}| < \varepsilon$. In that case, if the message length, i.e. (36) is lower than the current optimum, the parameters, weights, and length are saved in $\boldsymbol{\Theta}_{\mathrm{min}}$, W_{min} and $\mathbf{LEN}_{\mathrm{min}}$ respectively. In order to explore the full range of K, the less populated component is artificially annihilated, and CEM is run again. The complexity of Algorithm 1 is similar to the complexity of the algorithm in [15], with the exception of the E-W step. However, the most computationally intensive part of this step (matrix inversion and matrix-vector multiplications in (20)) is already achieved in the E-Z step.

VII. ALGORITHM INITIALIZATION

The EM algorithms proposed in Section III, Section V, and Section VI require proper initialization of both the weights (one for each observation and either a fixed value w_i or parameters α_i, β_i) and of the model parameters. The K-means algorithm is used for an initial clustering, from which values for the model parameters are computed. In this section we concentrate onto the issue of weight initialization. An

---

Algorithm 1: WD-EM with model selection based on the MML criterion.

input : $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^n, K_{\text{low}}, K_{\text{high}}, \boldsymbol{\Theta}^{(0)} = \{\pi_k^{(0)}, \boldsymbol{\mu}_k^{(0)}, \boldsymbol{\Sigma}_k^{(0)}\}_{K=1}^{K_{\text{high}}}, \boldsymbol{\Phi}^{(0)} = \{\alpha_i^{(0)}, \beta_i^{(0)}\}_{i=1}^n$

output: The minimum length mixture model: $\boldsymbol{\Theta}_{\min}$ and the final data weights: $\boldsymbol{W}_{\min}$

Set: $r = 0, \mathcal{K}^+ = \{k\}_{k=1}^{K_{\text{high}}}, \textbf{LEN}_{\min} = +\infty$

while $|\mathcal{K}^+| \geq K_{\text{low}}$ **do**

  repeat

    for $k = 1$ **to** K_{high} **do**

      E-Z step using (16):
$$\eta_{ik}^{(r+1)} = \frac{\pi_k^{(r)} \mathcal{P}\left(\boldsymbol{x}_i; \boldsymbol{\mu}_k^{(r)}, \boldsymbol{\Sigma}_k^{(r)}, \alpha_i^{(r)}, \beta_{ik}^{(r)}\right)}{\sum_{l=1}^{K_{\text{high}}} \pi_l^{(r)} \mathcal{P}\left(\boldsymbol{x}_i; \boldsymbol{\mu}_l^{(r)}, \boldsymbol{\Sigma}_l^{(r)}, \alpha_i^{(r)}, \beta_{il}^{(r)}\right)}$$

      E-W step using (19)–(20):
$$\alpha_i^{(r+1)} = \alpha_i^{(0)} + \frac{d}{2}$$
$$\beta_{ik}^{(r+1)} = \beta_i^{(0)} + \frac{1}{2} \left\| \boldsymbol{x}_i - \boldsymbol{\mu}_k^{(r)} \right\|_{\boldsymbol{\Sigma}_k^{(r)}}^2$$
$$\overline{w}_{ik} = \frac{\alpha_i^{(r+1)}}{\beta_{ik}^{(r+1)}}$$

      M-step
$$\pi_k^{(r+1)} = \frac{\max\left\{0, \sum_{i=1}^n \eta_{ik}^{(r+1)} - \frac{M}{2}\right\}}{\sum_{l=1}^{K_{\text{high}}} \max\left\{0, \sum_{i=1}^n \eta_{il}^{(r+1)} - \frac{M}{2}\right\}}$$

      if $\pi_k^{(r+1)} > 0$ **then**

        Evaluate $\boldsymbol{\theta}_k^{(r+1)}$: mean $\boldsymbol{\mu}_k^{(r+1)}$ using (26) and covariance $\boldsymbol{\Sigma}_k^{(r+1)}$ using (27).

      else

        $K^+ = K^+ - 1$

      end

    end

    $\boldsymbol{\Theta}^{(r+1)} = \left\{\pi_k^{(r+1)}, \boldsymbol{\theta}_k^{(r+1)}\right\}_{k=1}^{K_{\text{high}}}$

    Compute optimal length $\textbf{LEN}_{MML}^{(r+1)}$ with (36).

    $r \leftarrow r + 1$

  until $|\Delta\textbf{LEN}_{MML}^{(r)}| < \varepsilon$

  if $\textbf{LEN}_{MML}^{(r)} < \textbf{LEN}_{\min}$ **then**

    $\textbf{LEN}_{\min} = \textbf{LEN}_{MML}^{(r)}$

    $\boldsymbol{\Theta}_{\min} = \boldsymbol{\Theta}^{(r)}$

    $\boldsymbol{W}_{\min} = \{\overline{w}_i\}_{i=1}^n$ with $\overline{w}_i = \sum_{k=1}^{K_{\text{high}}} \eta_{ik} \overline{w}_{ik}$

  end

  $k^* = \text{argmin}_{k' \in \mathcal{K}^+}\left(\pi_{k'}^{(r)}\right), \quad \mathcal{K}^+ = \mathcal{K}^+ / k^*$

end

---

interesting feature of our method is that the only constraint on the weights is that they must be positive. Initial w_i values may depend on expert or prior knowledge and may be experiment- or goal-dependent. This model flexibility allows the incorporation of such prior knowledge. In the absence of any prior information/knowledge, we propose a data-driven initialization scheme and make the assumption that densely sampled regions are more important that sparsely sampled ones. We note that a similar strategy could be used if one wants to reduce the importance of dense data and to give more importance to small groups of data or to sparse data.

We adopt a well known data similarity measure based on the Gaussian kernel, and it follows that the weight w_i associated with the data point i is evaluated with:

$$w_i = \sum_{j \in \mathcal{S}_i^q} \exp\left(-\frac{d^2(\boldsymbol{x}_i, \boldsymbol{x}_j)}{\sigma}\right), \tag{38}$$

where $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the Euclidean distance, $\mathcal{S}_i^q$ denotes the set containing the q nearest neighbors of $\boldsymbol{x}_i$, and σ is a positive scalar. In all the experiments we used $q = 20$ for the simulated datasets and $q = 50$ for the real datasets. In both cases, we used $\sigma = 100$. In the case of the FWD-EM algorithm, the weights w_i thus initialized remain unchanged. However, in the case of the WD-EM algorithm, the weights are modeled as latent random variables drawn from a gamma distribution, hence one needs to set initial values for the parameters of this distribution, namely α_i and β_i in (11). Using (12) and (13) one can choose to initialize these parameters such as $\alpha_i = w_i^2$ and $\beta_i = w_i$, such that the mean and variance of the prior distribution are w_i and 1 respectively.

VIII. EXPERIMENTAL VALIDATION

The proposed algorithms were tested and evaluated using eight datasets: four simulated datasets and four publicly available datasets that are widely used for benchmarking clustering methods. The main characteristics of these datasets are summarized in Table I. The simulated datasets (SIM) are designed to evaluate the robustness of the proposed method with respect to outliers. The simulated inliers are drawn from Gaussian mixtures while the simulated outliers are drawn from a uniform distribution, e.g. Fig. 1. The SIM datasets have different cluster configurations in terms of separability, shape and compactness. The eight datasets that we used are the following:

- **SIM-Easy:** Five clusters that are well separated and compact.
- **SIM-Unbalanced:** Four clusters of different size and density.
- **SIM-Overlapped:** Four clusters, two of them overlap.
- **SIM-Mixed:** Six clusters of different size, compactness and shape.
- **MNIST** contains instances of handwritten digit images normalized to the same size [25]. We preprocessed these data with PCA to reduce the dimension from 784 to 141, by keeping 95% of the variance.

TABLE I: Datasets used for benchmarking and their characteristics: n is the number of data points, d is the dimension of the data space, and K is number of clusters.

Data Set	n	d	K
SIM-Easy	600	2	5
SIM-Unbalanced	600	2	4
SIM-Overlapped	600	2	4
SIM-Mixed	600	2	6
MNIST [25]	10,000	141	10
Wav [26]	5,000	21	3
BCW [27]	569	30	2
Letter Recognition [28]	20,000	16	26

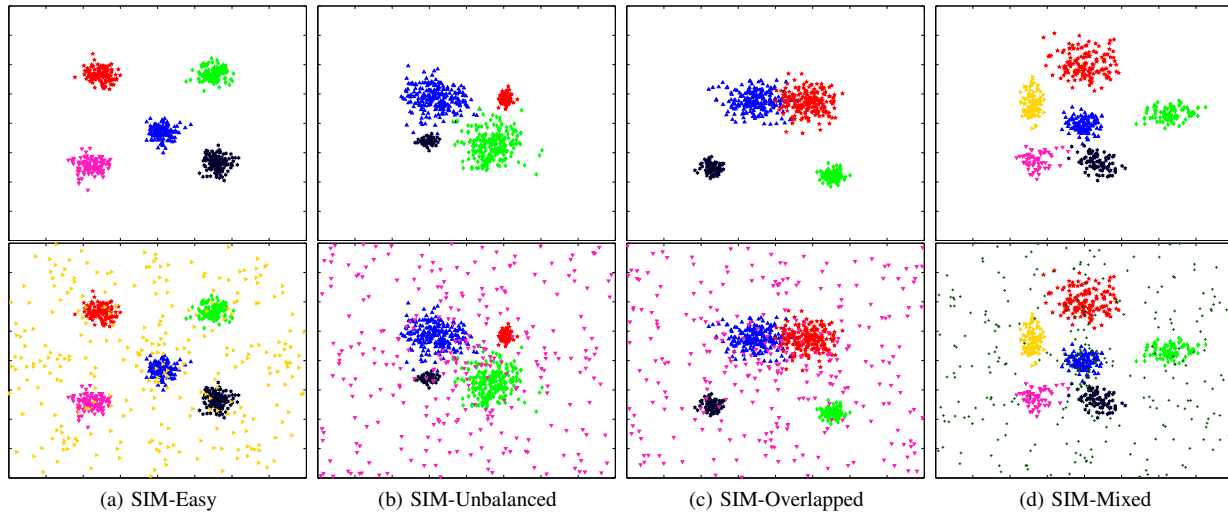| (a) SIM-Easy | (b) SIM-Unbalanced | (c) SIM-Overlapped | (d) SIM-Mixed |

Fig. 1: Samples of the SIM dataset with no outliers (top row) and contaminated with 50% outliers (bottom row). The 600 inliers are generated from Gaussian mixtures while the 300 outliers are generated from a uniform distribution.

- **Wav** is the Waveform Database Generator [26].
- **BCW** refers to the Breast Cancer Wisconsin data set [27], in which each instance represents a digitized image of a fine needle aspirate (FNA) of breast mass.
- **Letter Recognition** contains $20,000$ single-letter images that were generated by randomly distorting the images of the 26 uppercase letters from 20 different commercial fonts [28]. Each letter/image is described by 16 features. This dataset is available through the UCI machine learning repository.

In addition to the two proposed methods (FWD-EM and WD-EM) we tested the following algorithms:

- **GMM** uses EM with the standard Gaussian mixture model, implemented as described in [29];
- **GMM+U** uses EM with a GMM and with an additional uniform component, [30];
- **FM-uMST** stands for the *finite mixture of unrestricted multivariate skew t-distribution* algorithm of [8];
- **IGMM** stands for the *infinite Gaussian mixture model* [19];
- **I^{2}GMM** stands for the *infinite mixture of infinite Gaussian mixtures* [21];
- K**-Means** is the standard K-means algorithm;
- KK**-Means** is the kernel K-means algorithm of [31];
- **NCUT** is the spectral clustering algorithm of [32].
- **HAC** is the hierarchical agglomerative clustering algorithm of [33].

All the above algorithms need proper initialization. All the mixture-based algorithms, WD-EM, FWD-EM, GMM, GMM+U, FM-uMST, IGMM and I^{2}GMM start from the same proportions, means, and covariances which are estimated from the set of clusters provided by K-means. The latter is randomly initialized several times to find a good initialization. Furthermore, algorithms WD-EM, FWD-EM, GMM, GMM+U and FM-uMST are iterated until convergence, i.e, the log-likelihood difference between two consecutive iterations is less

than 1%, or are stopped after 400 iterations.

To quantitatively evaluate all the tested methods, we chose to use the Davies-Bouldin (DB) index [34]:

$$DB = \frac{1}{K} \sum_{k=1}^{K} R_k, \qquad (39)$$

where $R_k = \max_{k,k \neq l}\{(S_k + S_l)/d_{kl}\}$, $S_k = n_k^{-1} \sum_{\boldsymbol{x} \in C_k} \|\boldsymbol{x} - \boldsymbol{\mu}_k\|$ is the cluster scatter, n_k is the number of samples in cluster k, $\boldsymbol{\mu}_k$ is the cluster center, and $d_{kl} = \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_l\|$. A low value of the DB index means that the clusters are far from each other with respect to their scatter, and therefore the discriminative power is higher. Since the algorithms are randomly initialized, we repeat each experiment 20 times and compute the mean and standard deviation of the DB index for each experiment. Table II summarizes the results obtained with the MNIST, WAV, BCW, and Letter Recognition datasets. The proposed WD-EM method yields the best results for the WAV and BCW data, while the I^{2}GMM method yields the best results for the MNIST data. It is interesting to notice that the non-parametric methods K-means, NCUT and HAC yield the best and second best results for the Letter Recognition data.

For completeness we also provide the micro F_1 scores (also used in [21]) obtained with the MNIST, WAV, BCW and Letter Recognition datasets in Table III. Based on this classification score, the proposed WD-EM method yields the best results for the WAV and BCW data, while the I^{2}GMM yields the best results for the Letter Recognition data, and the IGMM method yields the best results for the MNIST data. This comparison also shows that I^{2}GMM, GMM and GMM+U yield similar scores.

An interesting feature of the proposed weighted-data clustering algorithms is their robustness in finding good clusters in the presence of outliers. To illustrate this ability we ran a large number of experiments by adding outliers, drawn from a uniform distribution, to the four simulated datasets, e.g. Table IV

TABLE II: Results obtained with the MNIST, WAV, BCW, and Letter Recognition datasets. The clustering scores correspond to the Davies-Bouldin (DB) index. The best results are shown in underlined **bold**, and the second best results are shown in **bold**. The proposed method yields the best results for the WAV and BCW datasets, while I^{2}GMM yields the best results for the MNIST dataset. Interestingly, the non-parametric methods (K-means, HAC and Ncut) yield excellent results for Letter Recognition.

Dataset	WD-EM	FWD-EM	GMM	GMM+U	FM-uMST	IGMM	I^{2}GMM	K-Means	KK-Means	Ncut	HAC
MNIST	2.965(0.15)	3.104(0.21)	3.291(0.14)	3.245(0.09)	**2.443(0.00)**	3.555(0.06)	**2.430(0.14)**	2.986(0.01)	2.980(0.02)	4.760(0.08)	3.178(0.00)
WAV	**0.975(0.00)**	1.019(0.00)	1.448(0.03)	1.026(0.04)	1.094(0.10)	1.028(0.02)	2.537(0.35)	1.020(0.00)	**0.975(0.05)**	2.781(0.06)	1.089(0.00)
BCW	**0.622(0.00)**	0.687(0.00)	0.714(0.00)	0.689(0.00)	0.727(0.00)	0.719(0.00)	0.736(0.09)	0.659(0.00)	**0.655(0.00)**	0.838(0.00)	0.685(0.00)
Letter Recognition	1.690(0.00)	1.767(0.01)	2.064(0.06)	2.064(0.06)	1.837(0.00)	2.341(0.11)	1.724(0.03)	**1.450(0.02)**	1.504(0.03)	**1.626(0.00)**	**1.626(0.00)**

TABLE III: Micro F$_1$ scores obtained on the real data sets (MNIST, WAV, BCW and Letter Recognition). The number in parenthesis indicates the standard deviation of 20 repetitions. Based on this classification score, I^{2}GMM yields the best result.

Data set	WD-EM	FWD-EM	GMM	GMM+U	FM-uMST	IGMM	I^{2}GMM	K-Means	KK-Means	Ncut	HAC
MNIST	0.524(0.01)	0.455(0.01)	**0.573(0.00)**	0.549(0.01)	0.519(0.00)	**0.689(0.02)**	0.545(0.06)	0.497(0.02)	0.507(0.02)	0.402(0.00)	0.532(0.00)
WAV	**0.774(0.00)**	0.534(0.00)	0.535(0.00)	0.552(0.00)	**0.632(0.08)**	0.543(0.01)	0.493(0.00)	0.521(0.00)	0.522(0.00)	0.387(0.00)	0.597(0.00)
BCW	**0.965(0.00)**	0.907(0.00)	0.885(0.00)	0.915(0.00)	**0.927(0.00)**	0.914(0.00)	0.682(0.00)	0.907(0.00)	0.910(0.00)	0.859(0.00)	0.879(0.00)
Letter Recognition	0.315(0.01)	0.323(0.00)	**0.423(0.00)**	**0.423(0.00)**	0.379(0.00)	0.306(0.02)	**0.466(0.01)**	0.340(0.00)	0.343(0.01)	0.347(0.00)	0.347(0.00)

TABLE IV: DB scores obtained on the SIM-X dataset (**best** and **second best**).

	Outliers	WD-EM	FWD-EM	GMM	GMM+U	FM-uMST	IGMM	I^{2}GMM	K-Means	KK-Means	Ncut	HAC
SIM-Easy	10%	**0.229(0.01)**	0.295(0.01)	0.295(0.01)	**0.222(0.02)**	0.307(0.02)	1.974(0.12)	0.500(0.16)	0.291(0.01)	0.330(0.07)	0.283(0.01)	0.266(0.00)
	20%	**0.266(0.02)**	0.338(0.01)	0.342(0.01)	**0.233(0.01)**	0.349(0.02)	1.564(0.43)	0.626(0.28)	0.344(0.01)	0.420(0.10)	0.335(0.01)	0.330(0.01)
	30%	**0.330(0.01)**	0.385(0.01)	0.384(0.02)	**0.227(0.02)**	0.501(0.04)	1.296(0.12)	0.570(0.27)	0.372(0.01)	0.381(0.03)	0.366(0.02)	0.376(0.01)
	40%	**0.358(0.01)**	0.445(0.04)	0.453(0.05)	**0.211(0.02)**	0.585(0.06)	1.259(0.16)	0.534(0.21)	0.417(0.01)	0.411(0.01)	0.409(0.01)	0.401(0.01)
	50%	**0.380(0.01)**	0.455(0.02)	0.459(0.02)	**0.195(0.01)**	0.568(0.05)	1.107(0.06)	0.626(0.21)	0.422(0.01)	0.439(0.03)	0.422(0.01)	0.438(0.01)
SIM-Unbalanced	10%	**0.270(0.01)**	0.954(0.72)	1.354(1.02)	**0.277(0.01)**	1.104(0.76)	1.844(0.29)	0.491(0.17)	0.405(0.02)	0.433(0.05)	0.402(0.02)	0.427(0.02)
	20%	**0.329(0.03)**	4.503(4.33)	3.003(1.85)	**0.269(0.01)**	1.181(0.44)	1.278(0.45)	0.591(0.13)	0.512(0.02)	0.515(0.03)	0.477(0.03)	0.529(0.02)
	30%	**0.399(0.03)**	3.502(3.09)	2.034(1.22)	**0.252(0.03)**	1.414(0.88)	1.272(0.35)	0.601(0.10)	0.548(0.03)	0.540(0.03)	0.531(0.02)	0.570(0.03)
	40%	**0.534(0.13)**	2.756(2.33)	2.097(1.15)	**0.251(0.02)**	1.650(0.94)	1.239(0.36)	0.615(0.05)	0.557(0.03)	0.567(0.02)	0.563(0.02)	0.597(0.02)
	50%	**0.557(0.10)**	2.400(1.44)	1.520(0.38)	**0.268(0.01)**	1.612(0.69)	1.144(0.36)	0.665(0.10)	0.580(0.03)	0.585(0.03)	0.583(0.03)	0.636(0.02)
SIM-Overlapped	10%	**0.305(0.02)**	0.693(0.31)	1.510(0.97)	**0.307(0.02)**	1.373(0.63)	2.168(0.20)	0.554(0.14)	0.395(0.03)	0.428(0.06)	0.385(0.01)	0.427(0.01)
	20%	**0.368(0.03)**	1.562(0.45)	1.881(0.50)	**0.293(0.01)**	2.702(1.28)	1.837(0.37)	0.608(0.08)	0.467(0.02)	0.532(0.07)	0.440(0.02)	0.502(0.01)
	30%	**0.472(0.04)**	1.825(0.55)	2.209(0.64)	**0.294(0.03)**	5.101(1.99)	1.568(0.61)	0.586(0.15)	0.532(0.02)	0.521(0.03)	0.508(0.01)	0.557(0.01)
	40%	0.549(0.04)	2.372(0.54)	2.597(0.73)	**0.322(0.01)**	4.569(1.72)	1.320(0.40)	0.687(0.11)	0.546(0.02)	0.556(0.03)	**0.541(0.03)**	0.593(0.02)
	50%	0.641(0.06)	2.269(0.44)	2.247(0.60)	**0.298(0.02)**	5.762(3.34)	1.174(0.25)	0.815(0.12)	0.563(0.03)	0.576(0.02)	**0.560(0.03)**	0.618(0.02)
SIM-Mixed	10%	**0.282(0.01)**	0.443(0.11)	0.448(0.11)	0.290(0.01)	0.951(0.35)	2.032(0.46)	0.414(0.12)	0.358(0.01)	0.418(0.06)	0.359(0.01)	0.355(0.01)
	20%	**0.351(0.02)**	0.857(0.52)	1.325(0.79)	**0.286(0.01)**	1.062(0.38)	1.782(0.44)	0.462(0.08)	0.413(0.02)	0.476(0.06)	0.409(0.01)	0.428(0.01)
	30%	**0.396(0.02)**	1.368(0.74)	1.524(0.64)	**0.278(0.01)**	1.693(0.56)	1.627(0.54)	0.483(0.07)	0.454(0.02)	0.464(0.04)	0.449(0.01)	0.468(0.01)
	40%	**0.449(0.03)**	1.100(0.61)	1.188(0.59)	**0.277(0.02)**	1.609(0.43)	1.456(0.34)	0.483(0.05)	0.478(0.02)	0.504(0.04)	0.478(0.01)	0.508(0.02)
	50%	**0.492(0.03)**	1.364(0.59)	1.513(0.67)	**0.265(0.01)**	1.972(0.86)	1.366(0.29)	0.562(0.04)	0.501(0.01)	0.515(0.02)	0.499(0.02)	0.546(0.02)

and Fig. 2. A comparison between WD-EM, FWD-EM, and the state-of-art clustering techniques mentioned above, with different percentages of outliers, is provided. As it can be easily observed in these tables, GMM+U performs extremely well in the presence of outliers, which is not surprising since the simulated outliers are drawn from a uniform distribution. Overall, the proposed WD-EM method is the second best performing method. Notice the very good performance of the Ncut method for the SIM-overlapped data. Among all these methods, only GMM+U and WD-EM offer the possibility to characterize the outliers using two very different strategies. The GMM+U model simply pulls them in an *outlier class* based on the posterior probabilities. The WD-EM algorithm iteratively updates the posterior probabilities of the weights, and the final posteriors, (18), allow to implement a simple outlier detec-tion mechanism. Another important remark is that WD-EM systematically outperforms FWD-EM, which fully justifies the proposed weighted-data model. Fig. 2 shows results of fitting the mixture models to SIM-mixed data drawn from a Gaussian mixture and contaminated with 50% outliers drawn from a uniform distribution. These plots show that GMM, IGMM, and I^{2}GMM find five components corresponding to data clusters while they also fit a component onto the outliers, roughly centered on the data set.

IX. Audio-Visual Clustering

In this section we illustrate the effectiveness of our method to deal with audio-visual data which belong to the heteroge-nous type of data, i.e. gathered with different sensors, having

(a) WD-EM   (b) FWD-EM   (c) GMM   (d) GMM+U

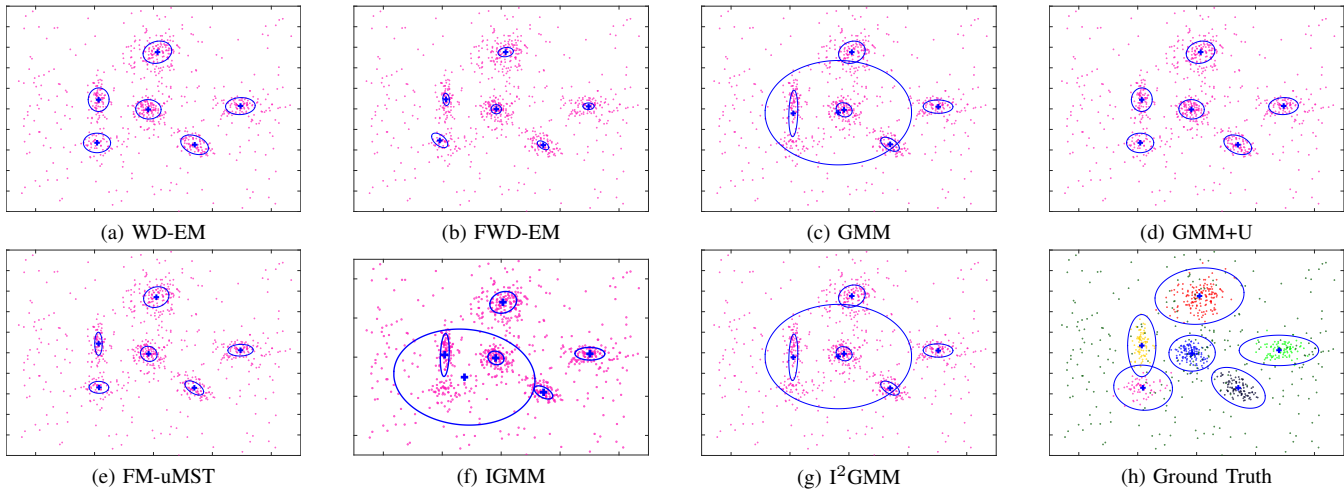(e) FM-uMST   (f) IGMM   (g) I²GMM   (h) Ground Truth

Fig. 2: Results obtained by fitting mixture models to the SIM-Mixed data in the presence of 50% outliers (see Table IV).

different noise statistics, and different sources of errors. The challenges of clustering audio-visual data were enumerated in Section I. Prior to clustering one needs to represent audio and visual observations in the same Euclidean space, e.g. Fig. 3. Without loss of generality we adopt the sound-source localization method of [35] that performs 2D direction of arrival (DOA) estimation followed by mapping the estimated sound-source direction onto the image plane: a DOA estimate therefore corresponds to a pixel location in the image plane. To find visual features, we use an upper-body detector [36] that provides an approximate localization of human heads, followed by lip localization using facial landmark detection [37]. The rationale of combining upper-body detection with facial landmark localization is that, altogether this yields a detection and localization algorithm that is much more robust to head pose than the vast majority of face detection methods.

Let $\boldsymbol{A} = \{\boldsymbol{a}_j\}_{j=1}^{n_a} \in \mathbb{R}^2$ and $\boldsymbol{V} = \{\boldsymbol{v}_j\}_{j=1}^{n_v} \in \mathbb{R}^2$ denote the set of auditory and visual observations respectively. To initialize the weight variables, we use (38) in the following way. An auditory sample is given a high initial weight if it has many visual samples as neighbors, or $w_{a_i} = \sum_{\boldsymbol{v}_j \in \boldsymbol{V}} \exp(-d^2(\boldsymbol{a}_i, \boldsymbol{v}_j)/\sigma)$. Visual weights are initialized in an analogous way, $w_{v_i} = \sum_{\boldsymbol{a}_j \in \boldsymbol{A}} \exp(-d^2(\boldsymbol{v}_i, \boldsymbol{a}_j)/\sigma)$. As illustrated below, this *cross-modal* weighting scheme favors clusters composed of both auditory and visual observations. We recorded three sequences:

- The *fake speaker* (FS) sequence, e.g. first and second rows of Fig. 4, consists of two persons facing the camera and the microphones. While the person onto the right emits speech signals (counting from "one" to "ten") the person onto the left performs fake lip, facial, and head movements as he would speak.
- The *moving speakers* (MS) sequence, e.g. third and fourth rows of Fig. 4, consists of two persons that move around while they are always facing the cameras and microphones. The persons take speech turns but there is

a short overlap between the two auditory signals.
- The *cocktail party* (CP) sequence, e.g. fifth and sixth rows of Fig. 4, consists of four persons engaged in an informal dialog. The persons wander around and turn their heads towards the active speaker; occasionally two persons speak simultaneously. Moreover the speakers do
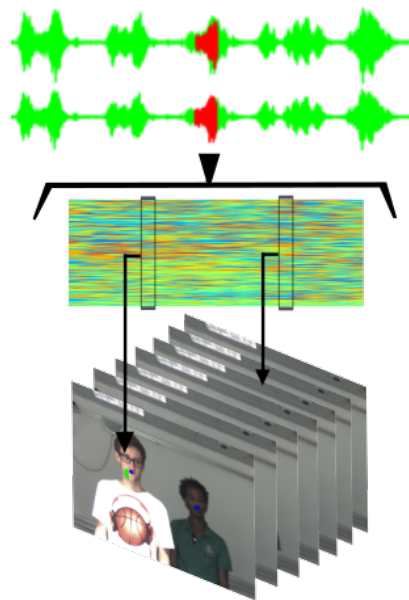

Fig. 3: Audio-visual data acquisition and alignment. *Top*: left- and right-microphone signals. A temporal segment of 0.4 s is outlined in red. *Middle*: Binaural spectrogram that corresponds to the outlined segment. This spectrogram is composed of 50 binaural vectors, each one being associated with an audio frame (shown as a vertical rectangle). *Bottom*: video frames associated with a segment. A sound-source direction of arrival (DOA) is extracted from each binaural vector and mapped onto the image plane, hence each green dot in the image plane corresponds to a DOA.

not always face the camera, hence face and lip detection/localization are unreliable.

The visual data are gathered with a single camera and the auditory data are gathered with two microphones plugged into the ears of an acoustic dummy head, referred to as *binaural audition*. The visual data are recorded at 25 video frames per second (FPS). The auditory data are gathered and processed in the following way. First, the short-time Fourier transform (STFT) is applied to the left- and right-microphone signals which are sampled at 48 KHz. Second, the left and right spectrograms thus obtained are combined to yield a binaural spectrogram from which a sound-source DOA is estimated. A spectrogram composed of 512 frequency bins is obtained by applying the STFT over a sliding window of width 0.064 s and shifted along the signal with 0.008 s hops. An audio frame, or 512 frequency bins, is associated with each window, hence there are 125 audio frames per second (with 0.056 ms overlap between consecutive frames). Both the visual and audio frames are further grouped into temporal segments of width 0.4 s, hence there are 10 visual frames and 50 audio frames in each segment.

As already mentioned, we follow the method of [35] to extract a sound-source DOA from each audio frame. In order to increase the robustness of audio localization, a voice activity detector (VAD) [38] is first applied to each frame, such that not all the frames have DOA estimates associated with them. On an average there are 40 audio DOA observations per segment. The FS sequence contains 28 segments, the MS sequence contains 43 segments, while the CP sequence contains 115 segments. The left hand sides of Fig. 4 show the central frame of a segment with all the visual features (blue) and auditory features (green) available within that segment.

We tested the proposed WD-EM algorithm on these audio-visual data as well as the GMM+U [30] and FM-uMST [8] algorithms. We chose to compare our method with these two methods for the following reasons. Firstly, all three methods are based on finite mixtures and hence they can use a model selection criterion to estimate the number of components in the mixture that best approximates clusters in the data. This is important since the number of persons and of active speakers among these persons are not known in advance. Secondly, as demonstrated in the previous section, these three methods yield robust clustering in the presence of outliers.

WD-EM uses the MML criterion for model selection as described in Section VI. We implemented a model selection criterion based on BIC to optimally select the number of components with GMM+U and FM-uMST. While each algorithm yields an optimal number of components for each audio-visual segment, not all them contain a sufficient number of audio and visual observations, such that the component can be associated with an active speaker. Therefore, we apply a simple two-step strategy, firstly to decide whether a component is *audio-visual*, *audio-only*, or *visual-only*, and secondly to select the best audio-visual components. Let n_v and n_a be the total number of visual and audio observations in a segment. We start by assigning each observation to a component: let n_a^k and n_v^k

TABLE V: The correct detection rates (CDR) obtained with the three methods for three scenarios: fake speaker (FS), moving speakers (MS), and cocktail party (CP).

Scenario	# Segments	WD-EM	GMM-U [30]	FM-uMST [8]
FS	28	100.00%	100.00%	71.43%
MS	43	83.87%	61.90%	72.22%
CP	115	65.66%	52.48%	49.57%

be the number of audio and visual observations associated with component k. Let $r_k = \min\{n_a^k, n_v^k\}/(n_a + n_v)$ measure the audio-visual relevance of a component. If $r_k \geq s$ then component k corresponds to an active speaker, with s being a fixed threshold.

Fig. 4 shows examples of applying the WD-EM, GMM+U and FM-uMST algorithms to the three sequences. One may notice that, while the visual observations (blue) are very accurate and form small *lumps* around the moving lips of a speaker (or of a fake speaker), audio observations (green) are very noisy and have different statistics; this is due to the presence of reverberations (the ceiling in particular) and of other sound sources, such as computer fans. The ground-truth active speaker is shown with a yellow frame. The data clusters obtained by the three methods are shown with red ellipses. A blue disk around a cluster center designates an audio-visual cluster. Altogether, one may notice that the proposed method outperforms the two other methods. An interesting feature of WD-EM is that the weights give more importance to the accurate visual data (because of the low-variance groups of observations available with these data) and hence the audio-visual cluster centers are pulled towards the visual data (lip locations in these examples).

To further quantify the performance of the three methods, we carefully annotated the data. For each segment, we identified the active speaker and we precisely located the speaker's lips. Let x_g be the ground-truth lip location. We assign x_g to a component by computing the maximum responsibility (16) of x_g. When x_g is assigned to an audio-visual cluster, an active speaker is said to be correctly detected if the posterior probability of x_g is equal or greater than $1/K$, where K is the number of components. Table V summarizes the results obtained with the three methods.

X. CONCLUSIONS

We presented a weighted-data Gaussian mixture model. We derived a maximum-likelihood formulation and we devised two EM algorithms, one that uses fixed weights (FWD-EM) and another one with weights modeled as random variables (WD-EM). While the first algorithm appears to be a straightforward generalization of standard EM for Gaussian mixtures, the second one has a more complex structure. We showed that the expectation and maximization steps of the proposed WD-EM admit closed-form solutions and hence the algorithm is extremely efficient. Moreover, WD-EM performs much better than FWD-EM which fully justifies the proposed
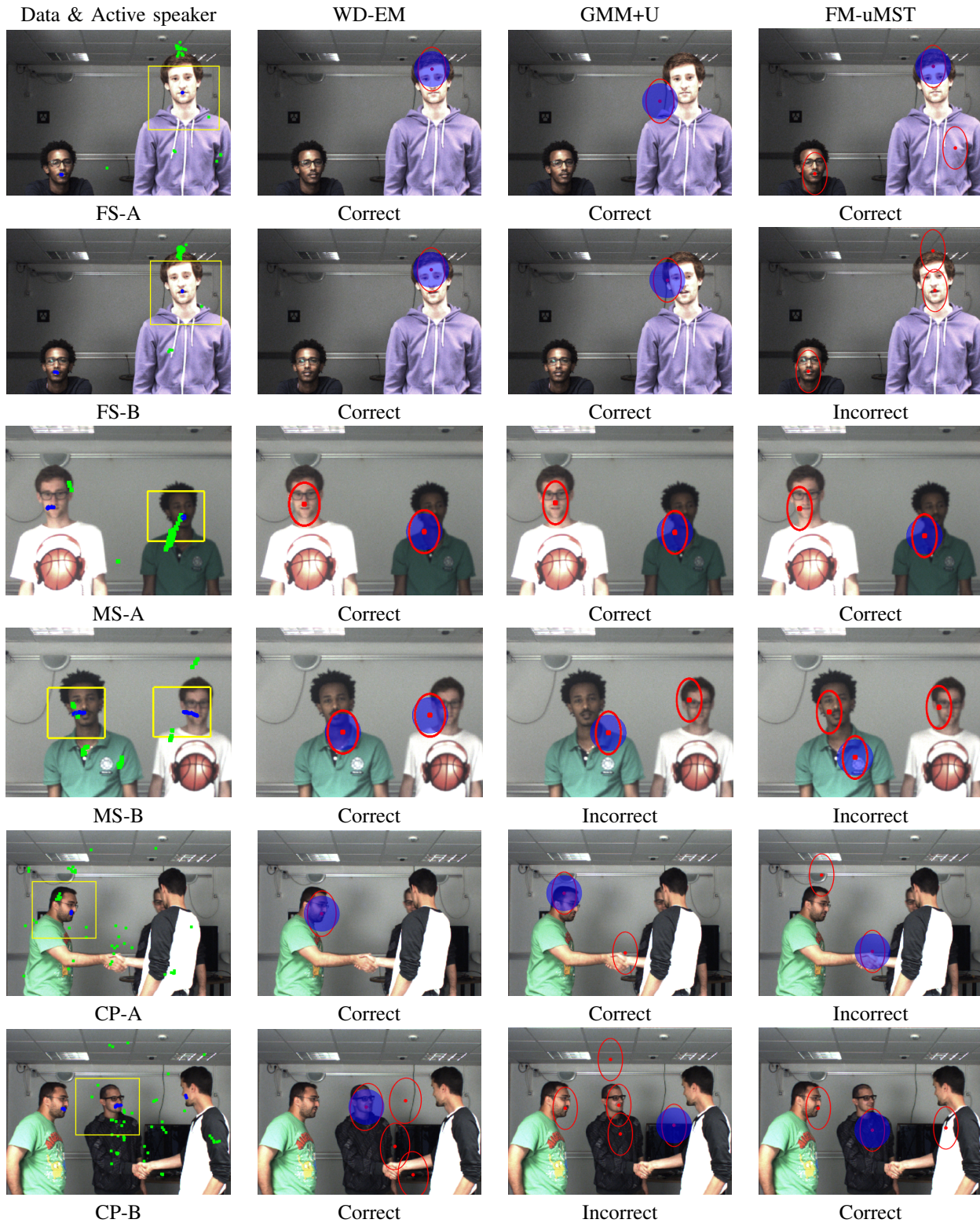
Fig. 4: Results obtained on the *fake speaker* (FS), *moving speaker* (MS) and *cocktail party* (CP) sequences. The first column shows the audio (green) and visual (blue) observations, as well as a yellow bounding box that shows the ground-truth active speaker. The second, third and fourth columns show the mixture components obtained with the WD-EM, GMM+U and FM-uMST methods, respectively. The blue disks mark components that correspond to correct detections of active speakers, namely whenever there is an overlap between a component and the ground-truth bounding box.

generative probabilistic model for the weights. We extended the MML-based model selection criterion proposed in [15] to the weighted-data Gaussian mixture model and we proposed an algorithm that finds an optimal number of components in the data. Interestingly, the WD-EM algorithm compares favorably with several state-of-the-art parametric and non-parametric clustering methods: it performs particularly well in the presence of a large number of outliers, e.g. up to 50% of outliers. Hence, the proposed formulation belongs to the robust category of clustering methods.

We also applied WD-EM to the problem of clustering heterogenous/multimodal data sets, such as audio-visual data. We briefly described the audio-visual fusion problem and how it may be cast into a challenging audio-visual clustering problem, e.g. how to associate human faces with speech signals and how to detect and localize active speakers in complex audio-visual scenes. We showed that the proposed algorithm yields better audio-visual clustering results than two other finite-mixture models, and this for two reasons: (i) it is very robust to noise and to outliers and (ii) it allows a cross-modal weighting scheme. Although not implemented in this paper, the proposed model has many other interesting features when dealing with multimodal data: it enables to balance the importance of the modalities, to emphasize one modality, or to use any prior information that might be available, for example by giving high weight priors to visual data corresponding to face/lip localization.

REFERENCES

[1] G. McLachlan and D. Peel, *Finite Mixture Models*. Wiley, 2000.

[2] ——, "Robust mixture modelling using the t distribution," *Statistics and Computing*, vol. 10, no. 4, pp. 339–348, 2000.

[3] C. M. Bishop and M. Svensen, "Robust Bayesian mixture modelling," *Neurocomputing*, vol. 64, pp. 235–252, 2005.

[4] C. Archambeau and M. Verleysen, "Robust Bayesian clustering," *Neural Networks*, vol. 20, no. 1, pp. 129–138, 2007.

[5] J. Sun, A. Kabán, and J. M. Garibaldi, "Robust mixture clustering using Pearson type VII distribution," *Pattern Recognition Letters*, vol. 31, no. 16, pp. 2447–2454, 2010.

[6] J. L. Andrews and P. D. McNicholas, "Model-based clustering, classification, and discriminant analysis via mixtures of multivariate t-distributions," *Statistics and Computing*, vol. 22, no. 5, pp. 1021–1029, 2012.

[7] F. Forbes and D. Wraith, "A new family of multivariate heavy-tailed distributions with variable marginal amounts of tailweight: application to robust clustering," *Statistics and Computing*, vol. 24, no. 6, pp. 971–984, November 2014.

[8] S. Lee and G. McLachlan, "Finite mixtures of multivariate skew t-distributions: some recent and new results," *Statistics and Computing*, vol. 24, no. 2, pp. 181–202, 2014.

[9] S. Kotz and S. Nadarajah, *Multivariate t Distributions and their Applications*. Cambridge, 2004.

[10] B. Long, Z. M. Zhang, X. Wu, and P. S. Yu, "Spectral clustering for multi-type relational data," in *Proceedings of the 23rd International Conference on Machine learning*. ACM, 2006, pp. 585–592.

[11] G. Tseng, "Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data," *Bioinformatics*, vol. 23, no. 17, pp. 2247–2255, 2007.

[12] M. Ackerman, S. Ben-David, S. Branzei, and D. Loker, "Weighted clustering," in *Proceedings of AAAI*, 2012.

[13] D. Feldman and L. Schulman, "Data reduction for weighted and outlier-resistant clustering," in *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2012, pp. 1343–1354.

[14] F. Forbes, S. Doyle, D. Garcia-Lorenzo, C. Barillot, and M. Dojat, "A weighted multi-sequence Markov model for brain lesion segmentation," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, vol. 9, Sardinia, Italy, 2010, pp. 225–232.

[15] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 381–396, 2002.

[16] C. Hennig, "Methods for merging gaussian mixture components," *Advances in Data Analysis and Classification*, vol. 4, no. 1, pp. 3–34, 2010.

[17] J. P. Baudry, E. A. Raftery, G. Celeux, K. Lo, and R. Gottardo, "Combining mixture components for clustering," *Journal of Computational and Graphical Statistics*, vol. 19, no. 2, 2010.

[18] V. Melnykov, "Merging mixture components for clustering through pairwise overlap," *Journal of Computational and Graphical Statistics*, 2014.

[19] C. E. Rasmussen, "The infinite Gaussian mixture model." in *NIPS*, vol. 12, 1999, pp. 554–560.

[20] D. Gorur and C. Rasmussen, "Dirichlet process gaussian mixture models: Choice of the base distribution," *Journal of Computer Science and Technology*, vol. 25, no. 4, pp. 653–664, 2010.

[21] H. Z. Yerebakan, B. Rajwa, and M. Dundar, "The infinite mixture of infinite Gaussian mixtures," in *Advances in Neural Information Processing Systems*, 2014, pp. 28–36.

[22] X. Wei and C. Li, "The infinite student t-mixture for robust modeling," *Signal Processing*, vol. 92, no. 1, pp. 224–234, 2012.

[23] G. Schwarz *et al.*, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[24] G. Celeux, S. Chrétien, F. Forbes, and A. Mkhadri, "A component-wise EM algorithm for mixtures," *Journal of Computational and Graphical Statistics*, vol. 10, no. 4, 2001.

[25] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[26] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*. CRC Press, 1984.

[27] W. Street, W. Wolberg, and O. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*. International Society for Optics and Photonics, 1993, pp. 861–870.

[28] P. W. Frey and D. J. Slate, "Letter recognition using Holland-style adaptive classifiers," *Machine Learning*, vol. 6, no. 2, pp. 161–182, 1991.

[29] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[30] J. Banfield and A. E. Raftery, "Model-based Gaussian and non-Gaussian clustering," *Biometrics*, vol. 49, no. 3, pp. 803–821, 1993.

[31] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel K-means: spectral clustering and normalized cuts," in *Proceedings of the Tenth International Conference on Knowledge Discovery and Data Mining*. ACM, 2004, pp. 551–556.

[32] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[33] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proceedings of the Eleventh International Conference on Information and Knowledge Management*. ACM, 2002, pp. 515–524.

[34] D. Davies and D. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 224–227, 1979.

[35] A. Deleforge, R. Horaud, Y. Y. Schechner, and L. Girin, "Co-localization of audio sources in images using binaural features and locally-linear regression," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 4, pp. 718–731, Apr. 2015.

[36] V. Ferrari, M.-J. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[37] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[38] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Processing Letters*, vol. 6, no. 1, pp. 1–3, 1999.

Israel Dejene Gebru received the B.Sc. degree in Computer Engineering from Addis Ababa University, Ethiopia in 2008 and the M.Sc. degree in Telecommunication Engineering from University of Trento, Italy in 2011. From 2008 to 2011, he worked for ZTE Cooperation as an Engineer. He is currently a PhD candidate in Université Joseph Fourier and works in Perception Team, at INRIA Grenoble Rhône-Alpes, France. His research interests include development of statistical methods and machine learning algorithms for multimodal signal analysis, speaker tracking and computer vision problems. He is particularly interested in joint processing of audio and visual data for applications in human-robot interaction.

Xavier Alameda-Pineda received the M.Sc. degree in mathematics and telecommunications engineering from the Universitat Politècnica de Catalunya – BarcelonaTech in 2008 and 2009 respectively, the M.Sc. degree in computer science from the Université Joseph Fourier and Grenoble INP in 2010, and the Ph.D. degree in mathematics/computer science from the Université Joseph Fourier in 2013. He worked towards his Ph.D. degree in the Perception Team, at INRIA Grenoble Rhône-Alpes. He currently holds a postdoctoral position at the Multimodal Human Understanding Group at University of Trento. His research interests are machine learning and signal processing for scene understanding, speaker diaritzation and tracking, sound source separation and behavior analysis.

Florence Forbes received the B.Sc. and M.Sc. degrees in computer science and applied mathematics from the Ecole Nationale Supérieure d'Informatique et Mathématiques Appliquées de Grenoble (ENSIMAG), France, and the PhD degree in applied probabilities from the University Joseph Fourier, Grenoble, France. Since 1998, she has been a research scientist with INRIA, where she currently holds a position of director of research. She founded the MISTIS team and has been the team head since 2003. Her research activities include Bayesian analysis, Markov and graphical models, and hidden structure models.

Radu Horaud received the B.Sc. degree in electrical engineering, the M.Sc. degree in control engineering, and the Ph.D. degree in computer science from the Institut National Polytechnique de Grenoble, France. Currently he holds a position of director of research with INRIA Grenoble, Montbonnot Saint-Martin, France, where he is the founder and head of the PERCEPTION team. His research interests include computer vision, machine learning, audio signal processing, audiovisual analysis, and robotics. He is an area editor of the *Elsevier Computer Vision and Image Understanding*, a member of the advisory board of the *Sage International Journal of Robotics Research*, and an associate editor of the *Kluwer International Journal of Computer Vision*. In 2013 Radu Horaud was awarded an ERC Advanced Grant for his project *Vision and Hearing in Action* (VHIA).