Conjugate Mixture Models for Clustering Multimodal Data

Vasil Khalidov, Florence Forbes, Radu Horaud

LETTER Communicated by Hagai Attias

Conjugate Mixture Models for Clustering Multimodal Data

Vasil Khalidov
vasil.khalidov@inrialpes.fr
Florence Forbes
florence.forbes@inrialpes.fr
Radu Horaud
radu.horaud@inrialpes.fr
INRIA Grenoble Rhône-Alpes, 38330 Montbonnot Saint-Martin, France

The problem of multimodal clustering arises whenever the data are gathered with several physically different sensors. Observations from different modalities are not necessarily aligned in the sense there there is no obvious way to associate or compare them in some common space. A solution may consist in considering multiple clustering tasks independently for each modality. The main difficulty with such an approach is to guarantee that the unimodal clusterings are mutually consistent. In this letter, we show that multimodal clustering can be addressed within a novel framework: conjugate mixture models. These models exploit the explicit transformations that are often available between an unobserved parameter space (objects) and each of the observation spaces (sensors). We formulate the problem as a likelihood maximization task and derive the associated conjugate expectation-maximization algorithm. The convergence properties of the proposed algorithm are thoroughly investigated. Several local and global optimization techniques are proposed in order to increase its convergence speed. Two initialization strategies are proposed and compared. A consistent model selection criterion is proposed. The algorithm and its variants are tested and evaluated within the task of 3D localization of several speakers using both auditory and visual data.

1 Introduction

The unsupervised clustering of multimodal data is a key capability whenever the goal is to group observations that are gathered using several physically different sensors. A typical example is the computational modeling of biological multisensory perception. This includes the issues of how a human detects objects that are both seen and touched (Pouget, Deneve, & Duhamel, 2002; Ernst & Banks, 2002), seen and heard (Anastasio, Patton, & Belkacem-Boussaid, 2000; King, 2004, 2005), or how a human localizes one source of sensory input in a natural environment in the presence of competing

stimuli and a variety of noise sources (Haykin & Chen, 2005). More generally, multisensory fusion (Hall & McMullen, 2004; Mitchell, 2007) is highly relevant in various other research domains, such as target tracking (Smith & Singh, 2006) based on radar and sonar data (Naus & van Wijk, 2004; Coiras, Baralli, & Evans, 2007), mobile robot localization with laser range finders and cameras (Castellanos & Tardos, 1999), robot manipulation and object recognition using both tactile and visual data (Allen, 1995; Joshi & Sanderson, 1999), underwater navigation based on active sonar and underwater cameras (Majumder, Scheding, & Durrant-Whyte, 2001), audiovisual speaker detection (Beal, Jojic, & Attias, 2003; Perez, Vermaak, & Blake, 2004; Fisher & Darrell, 2004), speech recognition (Heckmann, Berthommier, & Kroschel, 2002; Nefian, Liang, Pi, Liu, & Murphy, 2002; Shao & Barker, 2008), and so forth.

When the data originate from a single object, finding the best estimates for the object's characteristics is usually referred to as a pure fusion task, and it reduces to combining multisensor observations in some optimal way (Beal et al., 2003; Kushal, Rahurkar, Fei-Fei, Ponce, & Huang, 2006; Smith & Singh, 2006). For example, land and underwater robots fuse data from several sensors to build a 3D map of the ambient space regardless of the number of objects present in the environment (Castellanos & Tardos, 1999; Majumder et al., 2001). The problem is much more complex when several objects are present and when the task implies their detection, identification, and localization. In this case, one has to consider two processes simultaneously: segregation (Fisher, Darrell, Freeman, & Viola, 2001), which assigns each observation to either an object or an outlier category, and estimation which computes the parameters of each object based on the group of observations that were assigned to that object. In other words, in addition to fusing observations from different sensors, multimodal analysis requires the assignment of each observation to one of the objects.

This observation-to-object association problem can be cast into a probabilistic framework. Recent multisensor data fusion methods able to handle several objects are based on particle filters (Checka, Wilson, Siracusa, & Darrell, 2004; Chen & Rui, 2004; Gatica-Perez, Lathoud, Odobez, & McCowan, 2007). Notice, however, that the dimensionality of the parameter space grows exponentially with the number of objects, causing the number of required particles to increase dramatically and augmenting computational costs. A number of efficient sampling procedures were suggested (Chen & Rui, 2004; Gatica-Perez et al., 2007) to keep the problem tractable. Of course, this is done at the cost of loss in model generality, and hence these attempts are strongly application dependent. Another drawback of such models is that they cannot provide estimates of the accuracy and importance of each modality with respect to each object. The sampling and distribution estimation are performed in the parameter space, but no statistics are gathered for the observation spaces. Recently Hospedales and Vijayakumar (2008) extended the single-object model of Beal et al. (2003)

to multiple objects: several single-object models are incorporated into the multiple-object model, and the number of objects is selected by an additional hidden node, which thus accounts for model selection. This method also suffers from exponential growth in the number of possible models.

In the case of unimodal data, the problems of grouping observations and associating groups with objects can be cast into the framework of standard data clustering, which can be solved using a variety of parametric or nonparametric techniques. The problem of clustering multimodal data raises the difficult question of how to group together observations that belong to different physical spaces with different dimensionalities (e.g., how to group visual data with auditory data). When the observations from two different modalities can be aligned pairwise, a natural solution is to consider the Cartesian product of two unimodal spaces. Unfortunately, such an alignment is not possible in most practical cases. Different sensors operate at different frequency rates, and hence the number of observations gathered with one sensor can be quite different from the number of observations gathered with another sensor. Consequently, there is no obvious way to align the observations pairwise. Considering all possible pairs would result in a combinatorial blow-up and typically create an abundance of erroneous observations corresponding to inconsistent solutions.

Alternatively, one may consider several unimodal clusterings, provided that the relationships between a common object space and several observation spaces can be explicitly specified. Multimodal clustering then results in a number of unimodal clusterings jointly governed by the same unknown parameters characterizing the object space.

The original contribution of this letter is to show how the problem of clustering multimodal data can be addressed within the framework of mixture models (McLachlan & Peel, 2000). We propose a variant of the EM algorithm (Dempster, Laird, & Rubin, 1977; McLachlan & Krishnan, 1996) specifically designed to estimate object-space parameters that are indirectly observed in several sensor spaces. The convergence properties of the proposed algorithm are thoroughly investigated, and several efficient implementations are described in detail. The proposed model is composed of a number of modality-specific mixtures. These mixtures are jointly governed by a set of common object-space parameters (which will be referred to as the tying parameters), thus esnsuring consistency between the sensory data and the object space being sensed. This is done using explicit transformations from the unobserved parameter space (object space) to each of the observed spaces (sensor spaces). Hence, the proposed model is able to deal with observations that live in spaces with different physical properties such as dimensionality, space metric, and sensor sampling rate. We believe that linking the object space with the sensor spaces based on object-space-to-sensor-space transformations has more discriminative power than existing multisensor fusion techniques and hence performs better in terms of multiple object identification and localization. To the best of our knowledge,

there has been no previous attempt to use a generative model, such as ours, for the task of multimodal data interpretation.

In section 2 we formally introduce the concept of conjugate mixture models. Standard gaussian mixture models (GMM) are used to model the unimodal data. The parameters of these gaussian mixtures are governed by the object parameters through a number of object-space-to-sensor-space transformations (one transformation for each sensing modality). Through the letter, we will assume a very general class of transformations: nonlinear Lipschitz continuous functions (see below). In section 3 we cast the multimodal data clustering problem in the framework of maximum likelihood, and we explicitly derive the expectation and maximization steps of the associated EM algorithm. While the E-step of the proposed algorithm is standard, the M-step implies nonlinear optimization of the expected complete-data log likelihood with respect to the object parameters. We investigate efficient local and global optimization methods. More specifically, in section 4 we prove that, provided that the object-to-sensor functions as well as their first derivatives are Lipschitz continuous, the gradient of the expected complete-data log likelihood is Lipschitz continuous as well. The immediate consequence is that a number of recently proposed optimization algorithms specifically designed to solve Lipschitzian global optimization problems can be used within the M-step of the proposed algorithm (Zhigljavsky & Žilinskas, 2008). Several of these algorithms combine a local maximum search procedure with an initializing scheme to determine, at each iteration, good initial values from which the local search should be performed. This implies that the proposed EM algorithm has guaranteed convergence properties. Section 5 discusses several possible local search initialization schemes, leading to different convergence speeds. In section 6 we propose and compare two possible strategies to initialize the EM algorithm. Section 7 is devoted to a consistent criterion to determine the number of objects. Section 8 illustrates the proposed method with the task of audiovisual object detection and localization using binocular vision and binaural hearing. Section 10 analyzes in detail the performances of the proposed model under various practical conditions with both simulated and real data. Finally, section 11 concludes the letter and provides directions for future work.

2 Mixture Models for Multimodal Data

We consider N objects $n = 1, \ldots, N$. Each object n is characterized by a parameter vector of dimension d, denoted by $\boldsymbol{s}_n \in \mathbb{S} \subseteq \mathbb{R}^d$. The set $\mathbf{s} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n, \ldots, \boldsymbol{s}_N\}$ corresponds to the unknown tying parameters. The objects are observed with a number of physically different sensors. Although, for clarity, we consider two modalities, generalization is straightforward. Therefore, the observed data consist of two sets of observations denoted, respectively, by $\mathbf{f} = \{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m, \ldots, \boldsymbol{f}_M\}$ and $\mathbf{g} = \{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_k, \ldots, \boldsymbol{g}_K\}$ lying

in two different observation spaces of dimensions r and p, $\boldsymbol{f}_m \in \mathbb{F} \subseteq \mathbb{R}^r$ and $\boldsymbol{g}_k \in \mathbb{G} \subseteq \mathbb{R}^p$.

One key ingredient of our approach is that we consider the transformations

$$\begin{cases} \mathcal{F} : \mathbb{S} \to \mathbb{F} \\ \mathcal{G} : \mathbb{S} \to \mathbb{G} \end{cases} \tag{2.1}$$

that map $\mathbb{S}$, respectively, into the observation spaces $\mathbb{F}$ and $\mathbb{G}$. These transformations are defined by the physical and geometric properties of the sensors, and they are supposed to be known. We treat the general case when both $\mathcal{F}$ and $\mathcal{G}$ are nonlinear.

An assignment variable is associated with each observation, thus indicating the object that generated the observation: $\boldsymbol{A} = \{A_1, \ldots, A_m, \ldots, A_M\}$ and $\boldsymbol{B} = \{B_1, \ldots, B_k, \ldots, B_K\}$. Hence, the segregation process is cast into a hidden variable problem. The notation $A_m = n$ (resp. $B_k = n$) means that the observation $\boldsymbol{f}_m$ (resp. $\boldsymbol{g}_k$) was generated by object n. In order to account for erroneous observations, an additional $N+1$-th fictitious object is introduced to represent an outlier category. The notation $A_m = N+1$ (resp. $B_k = N+1$) means that $\boldsymbol{f}_m$ (resp. $\boldsymbol{g}_k$) is an outlier. Note that we will also use the following standard convention: uppercase letters for random variables ($\boldsymbol{A}$ and $\boldsymbol{B}$) and lowercase letters for their realizations ($\boldsymbol{a}$ and $\boldsymbol{b}$). The usual conditional independence assumption leads to

$$P(\mathbf{f}, \mathbf{g}|\boldsymbol{a}, \boldsymbol{b}) = \prod_{m=1}^{M} P(\boldsymbol{f}_m|a_m) \prod_{k=1}^{K} P(\boldsymbol{g}_k|b_k). \tag{2.2}$$

In addition, all assignment variables are assumed to be independent:

$$P(\boldsymbol{a}, \boldsymbol{b}) = \prod_{m=1}^{M} P(a_m) \prod_{k=1}^{K} P(b_k). \tag{2.3}$$

As discussed in section 11, more general cases, could be considered. However, we focus on the independent case, for it captures most of the features relevant to the conjugate clustering task and because more general dependence structures could be reduced to the independent case by using appropriate variational approximation techniques (Jordan, Ghahramani, Jaakkola, & Saul, 1998; Celeux, Forbes, & Peyrard, 2003).

Next we define the following probability density functions for all $n = 1 \ldots N, N+1$, for all $\boldsymbol{f}_m \in \mathbb{F}$, and for all $\boldsymbol{g}_k \in \mathbb{G}$:

$$P_n^{\mathbb{F}}(\boldsymbol{f}_m) = P(\boldsymbol{f}_m|A_m = n), \tag{2.4}$$

$$P_n^{\mathbb{G}}(\boldsymbol{g}_k) = P(\boldsymbol{g}_k|B_k = n). \tag{2.5}$$

More specifically, the likelihoods for an observation to belong to an object n are gaussian distributions whose means $\mathcal{F}(\boldsymbol{s}_n)$ and $\mathcal{G}(\boldsymbol{s}_n)$ correspond to the object's parameter vector $\boldsymbol{s}_n$ mapped to the observation spaces by the transformations $\mathcal{F}$ and $\mathcal{G}$:

$$P_n^{\mathbb{F}}(\boldsymbol{f}_m) = \mathcal{N}(\boldsymbol{f}_m;\ \mathcal{F}(\boldsymbol{s}_n), \boldsymbol{\Sigma}_n), \tag{2.6}$$

$$P_n^{\mathbb{G}}(\boldsymbol{g}_k) = \mathcal{N}(\boldsymbol{g}_k;\ \mathcal{G}(\boldsymbol{s}_n), \boldsymbol{\Gamma}_n), \tag{2.7}$$

with

$$\mathcal{N}(\boldsymbol{f}_m;\ \mathcal{F}(\boldsymbol{s}_n), \boldsymbol{\Sigma}_n) = \frac{1}{(2\pi)^{r/2}|\boldsymbol{\Sigma}_n|^{1/2}} \exp\left(-\frac{1}{2}\|\boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}_n)\|^2_{\boldsymbol{\Sigma}_n}\right), \tag{2.8}$$

where the notation $\|\boldsymbol{v} - \boldsymbol{w}\|^2_{\boldsymbol{\Sigma}}$ stands for the Mahalanobis distance $(\boldsymbol{v} - \boldsymbol{w})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{v} - \boldsymbol{w})$ and $^\top$ stands for the transpose of a matrix. The likelihoods of outliers are taken as two uniform distributions:

$$P_{N+1}^{\mathbb{F}}(\boldsymbol{f}_m) = \mathcal{U}(\boldsymbol{f}_m; V), \tag{2.9}$$

$$P_{N+1}^{\mathbb{G}}(\boldsymbol{g}_k) = \mathcal{U}(\boldsymbol{g}_k; U), \tag{2.10}$$

where V and U denote the respective support volumes. We also define the prior probabilities $\pi = (\pi_1, \ldots, \pi_n, \ldots, \pi_{N+1})$ and $\lambda = (\lambda_1, \ldots, \lambda_n, \ldots, \lambda_{N+1})$:

$$\pi_n = P(A_m = n), \quad \forall m = 1, \ldots, M, \tag{2.11}$$

$$\lambda_n = P(B_k = n), \quad \forall k = 1, \ldots, K. \tag{2.12}$$

Therefore, $\boldsymbol{f}_m$ and $\boldsymbol{g}_k$ are distributed according to two $(N+1)$-component mixture models, where each mixture is made of N gaussian components and one uniform component:

$$P(\boldsymbol{f}_m) = \sum_{n=1}^{N} \pi_n \mathcal{N}(\boldsymbol{f}_m;\ \mathcal{F}(\boldsymbol{s}_n), \boldsymbol{\Sigma}_n) + \pi_{N+1}\mathcal{U}(\boldsymbol{f}_m; V), \tag{2.13}$$

$$P(\boldsymbol{g}_k) = \sum_{n=1}^{N} \lambda_n \mathcal{N}(\boldsymbol{g}_k;\ \mathcal{G}(\boldsymbol{s}_n), \boldsymbol{\Gamma}_n) + \lambda_{N+1}\mathcal{U}(\boldsymbol{g}_k; U). \tag{2.14}$$

The log likelihood of the observed data can then be written as

$$\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta}) = \sum_{m=1}^{M} \log\left(\sum_{n=1}^{N} \pi_n \mathcal{N}(\boldsymbol{f}_m;\ \mathcal{F}(\boldsymbol{s}_n), \boldsymbol{\Sigma}_n) + \pi_{N+1}\mathcal{U}(\boldsymbol{f}_m; V)\right) + \\ + \sum_{k=1}^{K} \log\left(\sum_{n=1}^{N} \lambda_n \mathcal{N}(\boldsymbol{g}_k;\ \mathcal{G}(\boldsymbol{s}_n), \boldsymbol{\Gamma}_n) + \lambda_{N+1}\mathcal{U}(\boldsymbol{g}_k; U)\right), \tag{2.15}$$

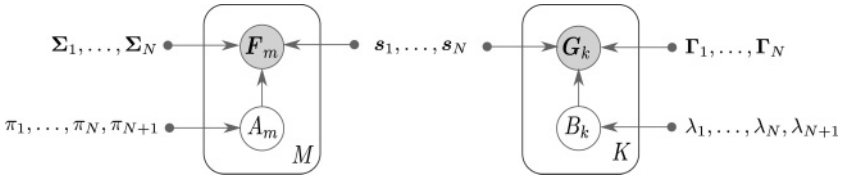


Figure 1: Graphical representation of the conjugate mixture model. Circles denote random variables and the plates (rectangles) around them represent multiple similar nodes, their number being given in the plates.

where

$$\boldsymbol{\theta} = \{\pi_1, \ldots, \pi_N, \pi_{N+1}, \lambda_1, \ldots, \lambda_N, \lambda_{N+1}, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_N, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_N, \boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_N\} \tag{2.16}$$

denotes the set of all unknown parameters to be estimated using a maximum likelihood principle.

The graphical representation of our conjugate mixture model is shown in Figure 1. We adopted the graphical notation introduced in Bishop (2006) to represent similar nodes in a more compact way: the M (resp. K) similar nodes are indicated with a plate. The two sensorial modalities are linked by the tying parameters $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N$ shown between the two plates.

3 Generalized EM for Clustering Multimodal Data

Given the probabilistic model just described, we wish to determine the parameter vectors associated with the objects that generated observations in two different sensory spaces. It is well known that direct maximization of the observed-data log likelihood equation 2.15, is difficult to achieve. The expectation-maximization (EM) algorithm (Dempster et al., 1977; McLachlan & Krishnan, 1996) is a standard approach to maximize likelihood functions of type 2.15. It is based on the following representation, for two arbitrary values of the parameters $\boldsymbol{\theta}$ and $\tilde{\boldsymbol{\theta}}$:

$$\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta}) = Q(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) + H(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}), \tag{3.1}$$

$$\text{with} \quad Q(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) = \mathbb{E}[\log P(\mathbf{f}, \mathbf{g}, \boldsymbol{A}, \boldsymbol{B}; \boldsymbol{\theta}) \mid \mathbf{f}, \mathbf{g}; \tilde{\boldsymbol{\theta}}], \tag{3.2}$$

$$\text{and} \quad H(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) = -\mathbb{E}[\log P(\boldsymbol{A}, \boldsymbol{B} \mid \mathbf{f}, \mathbf{g}; \boldsymbol{\theta}) | \mathbf{f}, \mathbf{g}; \tilde{\boldsymbol{\theta}}], \tag{3.3}$$

where the expectations are taken over the hidden variables $\boldsymbol{A}$ and $\boldsymbol{B}$. Each iteration q of EM proceeds in two steps:

- *Expectation*. For the current values $\boldsymbol{\theta}^{(q)}$ of the parameters, compute the conditional expectation with respect to variables $\boldsymbol{A}$ and $\boldsymbol{B}$:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = \sum_{\boldsymbol{a} \in \{1 \dots N+1\}^M} \sum_{\boldsymbol{b} \in \{1 \dots N+1\}^K} P(\boldsymbol{a}, \boldsymbol{b} | \mathbf{f}, \mathbf{g};\ \boldsymbol{\theta}^{(q)}) \times \log P(\mathbf{f}, \mathbf{g}, \boldsymbol{a}, \boldsymbol{b};\ \boldsymbol{\theta}). \tag{3.4}$$

- *Maximization*. Update the parameter set $\boldsymbol{\theta}^{(q)}$ by maximizing equation 3.4 with respect to $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}^{(q+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}). \tag{3.5}$$

It is well known that the EM algorithm increases the target function $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta})$ in equation 2.15, that is, the sequence of estimates $\{\boldsymbol{\theta}^{(q)}\}_{q \in \mathbb{N}}$ satisfies $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta}^{(q+1)}) \geq \mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta}^{(q)})$. Standard EM deals with the parameter estimation of a single mixture model, and a closed-form solution for equation 3.5 exists in this case. When the maximization 3.5 is difficult to achieve, various generalizations of EM are proposed. The M step can be relaxed by requiring just an increase rather than an optimum. This yields generalized EM (GEM) procedures (McLachlan & Krishnan, 1996, see Boyles, 1983), for a result on the convergence of this class of algorithms). The GEM algorithm searches for some $\boldsymbol{\theta}^{(q+1)}$ such that $Q(\boldsymbol{\theta}^{(q+1)}, \boldsymbol{\theta}^{(q)}) \geq Q(\boldsymbol{\theta}^{(q)}, \boldsymbol{\theta}^{(q)})$. Therefore, it provides a sequence of estimates that still verifies the nondecreasing likelihood property, although the convergence speed is likely to decrease. In the case of conjugate mixture models, we describe in more detail the specific forms of the E and M steps in the following sections.

3.1 The Expectation Step. Using equations 2.3 to 2.12, we can decompose the conditional expectation 3.4 as

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = Q_{\mathcal{F}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) + Q_{\mathcal{G}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}), \tag{3.6}$$

with

$$Q_{\mathcal{F}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = \sum_{m=1}^{M} \sum_{n=1}^{N+1} \alpha_{mn}^{(q)} \log \left(\pi_n P(\boldsymbol{f}_m | A_m = n;\ \boldsymbol{\theta}) \right), \tag{3.7}$$

$$Q_{\mathcal{G}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = \sum_{k=1}^{K} \sum_{n=1}^{N+1} \beta_{kn}^{(q)} \log \left(\lambda_n P(\boldsymbol{g}_k | B_k = n;\ \boldsymbol{\theta}) \right), \tag{3.8}$$

where $\alpha_{mn}^{(q)}$ and $\beta_{kn}^{(q)}$ denote the posterior probabilities $\alpha_{mn}^{(q)} = P(A_m = n | \boldsymbol{f}_m; \boldsymbol{\theta}^{(q)})$ and $\beta_{kn}^{(q)} = P(B_k = n | \boldsymbol{g}_k; \boldsymbol{\theta}^{(q)})$. Their expressions can be derived

straightforwardly from Bayes' theorem, $\forall n = 1, \ldots, N$:

$$\alpha_{mn}^{(q)} = \frac{\pi_n^{(q)} \mathcal{N}(\boldsymbol{f}_m; \mathcal{F}(\boldsymbol{s}_n^{(q)}), \boldsymbol{\Sigma}_n^{(q)})}{\sum_{i=1}^{N} \pi_i^{(q)} \mathcal{N}(\boldsymbol{f}_m; \mathcal{F}(\boldsymbol{s}_i^{(q)}), \boldsymbol{\Sigma}_i^{(q)}) + V^{-1} \pi_{N+1}^{(q)}}, \tag{3.9}$$

$$\beta_{kn}^{(q)} = \frac{\lambda_n^{(q)} \mathcal{N}(\boldsymbol{g}_k; \mathcal{G}(\boldsymbol{s}_n^{(q)}), \boldsymbol{\Gamma}_n^{(q)})}{\sum_{i=1}^{N} \lambda_i^{(q)} \mathcal{N}(\boldsymbol{g}_k; \mathcal{G}(\boldsymbol{s}_i^{(q)}), \boldsymbol{\Gamma}_i^{(q)}) + U^{-1} \lambda_{N+1}^{(q)}}, \tag{3.10}$$

and $\alpha_{m,N+1}^{(q)} = 1 - \sum_{n=1}^{N} \alpha_{mn}^{(q)}$ and $\beta_{k,N+1}^{(q)} = 1 - \sum_{n=1}^{N} \beta_{kn}^{(q)}$. When equations 2.6 to 2.10 are used, the expressions above further lead to

$$Q_{\mathcal{F}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = -\frac{1}{2} \sum_{m=1}^{M} \sum_{n=1}^{N} \alpha_{mn}^{(q)} \left(\| \boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}_n) \|_{\boldsymbol{\Sigma}_n}^2 + \log((2\pi)^r |\boldsymbol{\Sigma}_n| \pi_n^{-2}) \right) -$$

$$-\frac{1}{2} \sum_{m=1}^{M} \alpha_{m,N+1}^{(q)} \log(V^2 \pi_{N+1}^{-2}), \tag{3.11}$$

$$Q_{\mathcal{G}}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)}) = -\frac{1}{2} \sum_{k=1}^{K} \sum_{n=1}^{N} \beta_{kn}^{(q)} \left(\| \boldsymbol{g}_k - \mathcal{G}(\boldsymbol{s}_n) \|_{\boldsymbol{\Gamma}_n}^2 + \log((2\pi)^p |\boldsymbol{\Gamma}_n| \lambda_n^{-2}) \right) -$$

$$-\frac{1}{2} \sum_{k=1}^{K} \beta_{k,N+1}^{(q)} \log(U^2 \lambda_{N+1}^{-2}). \tag{3.12}$$

3.2 The Maximization Step. In order to carry out the maximization, equation 3.5, of the conditional expectation equation 3.4, its derivatives with respect to the model parameters are set to zero. This leads to the standard update expressions for priors, more specifically, $\forall n = 1, \ldots, N+1$:

$$\pi_n^{(q+1)} = \frac{1}{M} \sum_{m=1}^{M} \alpha_{mn}^{(q)}, \tag{3.13}$$

$$\lambda_n^{(q+1)} = \frac{1}{K} \sum_{k=1}^{K} \beta_{kn}^{(q)}. \tag{3.14}$$

The covariance matrices are governed by the tying parameters $\boldsymbol{s}_n^{(q+1)} \in \mathbb{S}$ through the functions $\mathcal{F}$ and $\mathcal{G}$, $\forall n = 1, \ldots, N$:

$$\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s}_n^{(q+1)}) = \frac{1}{\sum_{m=1}^{M} \alpha_{mn}^{(q)}} \sum_{m=1}^{M} \alpha_{mn}^{(q)} (\boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}_n^{(q+1)}))(\boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}_n^{(q+1)}))^\top, \tag{3.15}$$

$$\mathbf{\Gamma}_n^{(q+1)}(\boldsymbol{s}_n^{(q+1)}) = \frac{1}{\sum_{k=1}^{K} \beta_{kn}^{(q)}} \sum_{k=1}^{K} \beta_{kn}^{(q)} (\boldsymbol{g}_k - \mathcal{G}(\boldsymbol{s}_n^{(q+1)}))(\boldsymbol{g}_k - \mathcal{G}(\boldsymbol{s}_n^{(q+1)}))^{\top}. \tag{3.16}$$

For every $n = 1, \ldots, N$, $\boldsymbol{s}_n^{(q+1)}$ is the parameter vector such that

$$\boldsymbol{s}_n^{(q+1)} = \operatorname*{argmax}_{\boldsymbol{s}} Q_n^{(q)}(\boldsymbol{s}), \tag{3.17}$$

where

$$\begin{aligned} Q_n^{(q)}(\boldsymbol{s}) = &- \sum_{m=1}^{M} \alpha_{mn}^{(q)} (\| \boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}) \|^2_{\mathbf{\Sigma}_n(\boldsymbol{S})} + \log |\mathbf{\Sigma}_n(\boldsymbol{s})|) - \\ &- \sum_{k=1}^{K} \beta_{kn}^{(q)} (\| \boldsymbol{g}_k - \mathcal{G}(\boldsymbol{s}) \|^2_{\mathbf{\Gamma}_n(\boldsymbol{S})} + \log |\mathbf{\Gamma}_n(\boldsymbol{s})|). \end{aligned} \tag{3.18}$$

We stress that the covariances $\mathbf{\Sigma}_n(\boldsymbol{s})$ and $\mathbf{\Gamma}_n(\boldsymbol{s})$ in equations 3.15 and 3.16 are considered functions of $\boldsymbol{s} \in \mathbb{S}$. Hence, at each iteration of the algorithm, the overall update of the tying parameters can be split into N identical optimization tasks of the form 3.18. These tasks can be solved in parallel. In general, $\mathcal{F}$ and $\mathcal{G}$ are nonlinear transformations, and hence there is no simple closed-form expression for the estimation of the tying parameters.

3.3 Generalized EM for Conjugate Mixture Models. The initial parameters selection of the proposed EM algorithm for conjugate mixture models uses the procedure *Initialize* that is given in section 6. The maximization step uses two procedures, referred to as *Choose* and *Local Search*, which are explained in detail in sections 4 and 5. To determine the number of objects, we define the procedure *Select* that is derived in section 7. The overall EM procedure is outlined below:

1. Apply procedure *Initialize* to initialize the parameter vector:

$$\begin{aligned} \boldsymbol{\theta}^{(0)} = \Big\{ &\pi_1^{(0)}, \ldots, \pi_{N+1}^{(0)}, \lambda_1^{(0)}, \ldots, \lambda_{N+1}^{(0)}, \boldsymbol{s}_1^{(0)}, \ldots, \boldsymbol{s}_N^{(0)}, \mathbf{\Sigma}_1^{(0)}, \ldots, \\ &\mathbf{\Sigma}_N^{(0)}, \mathbf{\Gamma}_1^{(0)}, \ldots, \mathbf{\Gamma}_N^{(0)} \Big\}. \end{aligned}$$

2. *E step*: Compute $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(q)})$ using equations 3.9 to 3.12
3. *M step*: Estimate $\boldsymbol{\theta}^{(q+1)}$ using the following substeps:
   (a) *The priors.* Compute $\pi_1^{(q+1)}, \ldots, \pi_{N+1}^{(q+1)}$ and $\lambda_1^{(q+1)}, \ldots, \lambda_{N+1}^{(q+1)}$ using equations 3.13 and 3.14;
   (b) *The tying parameters.* For each $n = 1, \ldots, N$:

- Apply procedure *Choose* to determine an initial value, denoted by $\tilde{\boldsymbol{s}}_n^{(0)}$, as proposed in section 5;
- Apply procedure *Local Search* to each $Q_n^{(q)}(\boldsymbol{s})$ as defined in equation 3.18 starting from $\tilde{\boldsymbol{s}}_n^{(0)}$ and set the result to $\boldsymbol{s}_n^{(q+1)}$ using the equation 3.19 specified below.

(c) *The covariance matrices.* For every $n = 1 \dots N$, use equations 3.15 and 3.16 to compute $\boldsymbol{\Sigma}_n^{(q+1)}$ and $\boldsymbol{\Gamma}_n^{(q+1)}$.

4. *Check for convergence*: Terminate; otherwise go to step 2.
5. Apply procedure *Select*. Use equation 7.1, specified below, to determine the best N.

This algorithm uses the following procedures:

- *Initialize*. This procedure aims at providing the initial parameter values $\boldsymbol{\theta}^{(0)}$. Its performance has a strong impact on the time required for the algorithm to converge. In section 6, we propose different initialization strategies based on single-space cluster detection.
- *Select*. This procedure applies the BIC-like criterion to determine the number of objects N. In section 7, we propose the consistent criterion for the case of conjugate mixture models.
- *Choose*. The goal of this procedure is to provide at each M-step initial values $\tilde{\boldsymbol{s}}_1^{(0)}, \dots, \tilde{\boldsymbol{s}}_N^{(0)}$ which are likely to be close to the global maxima of the functions $Q_n^{(q)}(\boldsymbol{s})$ in equation 3.18. The exact form of this procedure is important to ensure the ability of the subsequent *Local Search* procedure to find these global maxima. We will use results on global search algorithms (Zhigljavsky & Žilinskas, 2008) and propose different variants in section 5.
- *Local Search*. An important requirement of this procedure is that it finds a local maximum of the $Q_n^{(q)}(\boldsymbol{s})$'s starting from any arbitrary point in $\mathbb{S}$. In this work, we will consider procedures that consist in iterating a local update of the form (ν is the iteration index)

$$\tilde{\boldsymbol{s}_n}^{(\nu+1)} = \tilde{\boldsymbol{s}_n}^{(\nu)} + \mathbf{H}_n^{(q,\nu)} \boldsymbol{\nabla} Q_n^{(q)}(\tilde{\boldsymbol{s}_n}^{(\nu)}), \tag{3.19}$$

with $\mathbf{H}_n^{(q,\nu)}$ being a positive definite matrix that may vary with ν. When the gradient $\boldsymbol{\nabla} Q_n^{(q)}(\boldsymbol{s})$ is Lipschitz continuous with some constant $L_n^{(q)}$, an appropriate choice that guarantees the increase of $Q_n^{(q)}(\tilde{\boldsymbol{s}}^{(\nu)})$ at each iteration ν is to choose $\mathbf{H}_n^{(q,\nu)}$ such that it verifies $\|\mathbf{H}_n^{(q,\nu)}\| \leq 2/L_n^{(q)}$. Different choices for $\mathbf{H}_n^{(q,\nu)}$ are possible, and they correspond to different optimization methods that in general belong to the variable metric class. For example $\mathbf{H}_n^{(q,\nu)} = \frac{2}{L_n^{(q)}}\mathbf{I}$ leads to gradient ascent, while taking $\mathbf{H}_n^{(q,\nu)}$ as a scaled inverse of the Hessian matrix would lead to a Newton-Raphson optimization step.

Other possibilities include Levenberg-Marquardt and quasi-Newton methods.

4 Analysis of the *Local Search* Procedure

Each instance of equation 3.18 for $n = 1, \ldots, N$ can be solved independently. In this section, we focus on providing a set of conditions under which each iteration of our algorithm guarantees that the objective function $Q_n^{(q)}(\boldsymbol{s})$ in equation 3.18 is increased. We start by rewriting equation 3.18 more conveniently in order to perform the optimization with respect to $\boldsymbol{s} \in \mathbb{S}$. To simplify the notation, the iteration index q is sometimes omitted. We simply write $Q_n(\boldsymbol{s})$ for $Q_n^{(q)}(\boldsymbol{s})$.

Let $\bar{\alpha}_n = \sum_{m=1}^{M} \alpha_{mn}^{(q)}$ and $\bar{\beta}_n = \sum_{k=1}^{K} \beta_{kn}^{(q)}$ denote the average object weights in each one of the two modalities. We introduce $\alpha_n = \bar{\alpha}_n^{-1}(\alpha_{1n}^{(q)}, \ldots, \alpha_{Mn}^{(q)})$ and $\beta_n = \bar{\beta}_n^{-1}(\beta_{1n}^{(q)}, \ldots, \beta_{Kn}^{(q)})$, the discrete probability distributions obtained by normalizing the object weights. We denote by $\boldsymbol{F}$ and $\boldsymbol{G}$ the random variables that take their values in the discrete sets $\{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m, \ldots, \boldsymbol{f}_M\}$ and $\{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_k, \ldots, \boldsymbol{g}_K\}$. It follows that the expressions for the optimal variances 3.15 and 3.16 as functions of $\boldsymbol{s}$ can be rewritten as

$$\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s}) = \mathbb{E}_{\alpha_n}[\big(\boldsymbol{F} - \mathcal{F}(\boldsymbol{s})\big)\big(\boldsymbol{F} - \mathcal{F}(\boldsymbol{s})\big)^\top], \tag{4.1}$$

$$\boldsymbol{\Gamma}_n^{(q+1)}(\boldsymbol{s}) = \mathbb{E}_{\beta_n}[\big(\boldsymbol{G} - \mathcal{G}(\boldsymbol{s})\big)\big(\boldsymbol{G} - \mathcal{G}(\boldsymbol{s})\big)^\top], \tag{4.2}$$

where $\mathbb{E}_{\alpha_n}$ and $\mathbb{E}_{\beta_n}$ denote the expectations with respect to the distributions α_n and β_n. Using some standard projection formula, it follows that the covariances are

$$\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s}) = \mathbf{V}_f + \boldsymbol{v}_f \boldsymbol{v}_f^\top, \tag{4.3}$$

$$\boldsymbol{\Gamma}^{(q+1)}(\boldsymbol{s}) = \mathbf{V}_g + \boldsymbol{v}_g \boldsymbol{v}_g^\top, \tag{4.4}$$

where $\mathbf{V}_f$ and $\mathbf{V}_g$ are the covariance matrices of $\boldsymbol{F}$ and $\boldsymbol{G}$, respectively, under distributions α_n and β_n, and $\boldsymbol{v}_f$ and $\boldsymbol{v}_g$ are vectors defined by

$$\boldsymbol{v}_f = \mathbb{E}_{\alpha_n}[\boldsymbol{F}] - \mathcal{F}(\boldsymbol{s}), \tag{4.5}$$

$$\boldsymbol{v}_g = \mathbb{E}_{\beta_n}[\boldsymbol{G}] - \mathcal{G}(\boldsymbol{s}). \tag{4.6}$$

For convenience, we omit the index n for $\mathbf{V}_f$, $\mathbf{V}_g$, $\boldsymbol{v}_f$, and $\boldsymbol{v}_g$. Let $\bar{\boldsymbol{f}}_n = \mathbb{E}_{\alpha_n}[\boldsymbol{F}]$ and $\bar{\boldsymbol{g}}_n = \mathbb{E}_{\beta_n}[\boldsymbol{G}]$. This yields

$$\bar{\boldsymbol{f}}_n = \bar{\alpha}_n^{-1} \sum_{m=1}^{M} \alpha_{mn}^{(q)} \boldsymbol{f}_m, \tag{4.7}$$

$$\bar{\boldsymbol{g}}_n = \bar{\beta}_n^{-1} \sum_{k=1}^{K} \beta_{kn}^{(q)} \boldsymbol{g}_k, \tag{4.8}$$

$$\mathbf{V}_f = \bar{\alpha}_n^{-1} \sum_{m=1}^{M} \alpha_{mn}^{(q)} \boldsymbol{f}_m \boldsymbol{f}_m^\top - \bar{\boldsymbol{f}}_n \bar{\boldsymbol{f}}_n^\top, \tag{4.9}$$

$$\mathbf{V}_g = \bar{\beta}_n^{-1} \sum_{k=1}^{K} \beta_{kn}^{(q)} \boldsymbol{g}_k \boldsymbol{g}_k^\top - \bar{\boldsymbol{g}}_n \bar{\boldsymbol{g}}_n^\top. \tag{4.10}$$

Next we derive a simplified expression for $Q_n(\boldsymbol{s})$ in equation 3.18 in order to investigate its properties. Notice that one can write equation 3.18 as the sum $Q_n(\boldsymbol{s}) = Q_{n,\mathcal{F}}(\boldsymbol{s}) + Q_{n,\mathcal{G}}(\boldsymbol{s})$, with

$$Q_{n,\mathcal{F}}(\boldsymbol{s}) = -\sum_{m=1}^{M} \alpha_{mn}^{(q)} (\| \boldsymbol{f}_m - \mathcal{F}(\boldsymbol{s}) \|^2_{\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})} + \log |\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})|), \tag{4.11}$$

and a similar expression for $Q_{n,\mathcal{G}}(\boldsymbol{s})$. Equation 4.11 can be written as

$$\begin{aligned} Q_{n,\mathcal{F}}(\boldsymbol{s}) = &-\bar{\alpha}_n (\mathbb{E}_{\alpha_n}[(\boldsymbol{F} - \mathcal{F}(\boldsymbol{s}))^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} (\boldsymbol{F} - \mathcal{F}(\boldsymbol{s}))] \\ &+ \log |\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})|). \end{aligned} \tag{4.12}$$

The first term of equation 4.12 can be further divided into two terms:

$$\begin{aligned} &\mathbb{E}_{\alpha_n}[(\boldsymbol{F} - \mathcal{F}(\boldsymbol{s}))^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} (\boldsymbol{F} - \mathcal{F}(\boldsymbol{s}))] = \\ &= \mathbb{E}_{\alpha_n}[(\boldsymbol{F} - \bar{\boldsymbol{f}}_n)^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} (\boldsymbol{F} - \bar{\boldsymbol{f}}_n)] + \boldsymbol{v}_f^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} \boldsymbol{v}_f. \end{aligned} \tag{4.13}$$

The Sherman-Morrison formula applied to equation 4.3 leads to

$$\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} = \mathbf{V}_f^{-1} - \mathbf{V}_f^{-1} \boldsymbol{v}_f \boldsymbol{v}_f^\top \mathbf{V}_f^{-1} / (1 + D_{n,\mathcal{F}}(\boldsymbol{s})), \tag{4.14}$$

with

$$D_{n,\mathcal{F}}(\boldsymbol{s}) = \| \mathcal{F}(\boldsymbol{s}) - \bar{\boldsymbol{f}}_n \|^2_{\mathbf{V}_f}. \tag{4.15}$$

It follows that equation 4.13 can be written as the sum of

$$\mathbb{E}_{\alpha_n}[(\boldsymbol{F} - \bar{\boldsymbol{f}}_n)^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} (\boldsymbol{F} - \bar{\boldsymbol{f}}_n)] = C_f - \frac{D_{n,\mathcal{F}}(\boldsymbol{s})}{1 + D_{n,\mathcal{F}}(\boldsymbol{s})}, \tag{4.16}$$

and of

$$\boldsymbol{v}_f^\top \boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})^{-1} \boldsymbol{v}_f = \frac{D_{n,\mathcal{F}}(\boldsymbol{s})}{1 + D_{n,\mathcal{F}}(\boldsymbol{s})}. \tag{4.17}$$

Hence the first term of equation 4.12, namely 4.13, is equal to C_f, which is constant with respect to $\boldsymbol{s}$. Moreover, applying the matrix determinant lemma to the second term of equation 4.12, we successively obtain

$$\begin{aligned} \log|\boldsymbol{\Sigma}_n^{(q+1)}(\boldsymbol{s})| &= \log|\mathbf{V}_f + \boldsymbol{v}_f \boldsymbol{v}_f^\top| = \log|\mathbf{V}_f| + \log(1 + \boldsymbol{v}_f^\top \mathbf{V}_f^{-1} \boldsymbol{v}_f) = \\ &= \log|\mathbf{V}_f| + \log(1 + D_{n,\mathcal{F}}(\boldsymbol{s})). \end{aligned} \tag{4.18}$$

It follows that there is only one term depending on $\boldsymbol{s}$ in equation 4.12:

$$Q_{n,\mathcal{F}}(\boldsymbol{s}) = -\bar{\alpha}_n \left(C_f + \log|\mathbf{V}_f| + \log(1 + D_{n,\mathcal{F}}(\boldsymbol{s})) \right). \tag{4.19}$$

Repeating the same derivation for the second sensorial modality, we obtain the following equivalent form of equation 3.18:

$$Q_n(\boldsymbol{s}) = -\bar{\alpha}_n \log(1 + D_{n,\mathcal{F}}(\boldsymbol{s})) - \bar{\beta}_n \log(1 + D_{n,\mathcal{G}}(\boldsymbol{s})) + C, \tag{4.20}$$

where C is some constant not depending on $\boldsymbol{s}$.

Using this form of $Q_n(\boldsymbol{s})$, we can now investigate the properties of its gradient $\nabla Q_n(\boldsymbol{s})$. It appears that under some regularity assumptions on $\mathcal{F}$ and $\mathcal{G}$, the gradient $\nabla Q_n(\boldsymbol{s})$ is bounded and Lipschitz continuous. The corresponding theorem is formulated and proved. First, we establish as a lemma some technical results required to prove the theorem. In what follows, for any matrix $\mathbf{V}$, the matrix norm used is the operator norm $\|\mathbf{V}\| = \sup_{\|\boldsymbol{v}\|=1} \|\mathbf{V}\boldsymbol{v}\|$. For simplicity, we further omit the index n.

Lemma 1. *Let* $\mathbf{V}$ *be a symmetric positive definite matrix. Then the function*

$$\varphi(\boldsymbol{v}) = \|\boldsymbol{V}\boldsymbol{v}\|/(1 + \boldsymbol{v}^\top \boldsymbol{V}\boldsymbol{v})$$

is bounded by $\varphi(\boldsymbol{v}) \leq C_\varphi(\boldsymbol{V})$ *with* $C_\varphi(\boldsymbol{V}) = \sqrt{\|\boldsymbol{V}\|}/2$ *and is Lipschitz continuous:*

$$\forall \boldsymbol{v}, \tilde{\boldsymbol{v}} \quad \|\varphi(\boldsymbol{v}) - \varphi(\tilde{\boldsymbol{v}})\| \leq L_\varphi(\boldsymbol{V})\|\boldsymbol{v} - \tilde{\boldsymbol{v}}\|,$$

where $L_\varphi(\boldsymbol{V}) = \|\boldsymbol{V}\|(1 + \mu(\boldsymbol{V})/2)$ *is the Lipschitz constant and* $\mu(\boldsymbol{V}) = \|\boldsymbol{V}\|\|\boldsymbol{V}^{-1}\|$ *is the condition number of* $\boldsymbol{V}$.

Proof. We start by introducing $\boldsymbol{w} = \mathbf{V}\boldsymbol{v}$ so that $\varphi(\boldsymbol{v}) = \tilde{\varphi}(\boldsymbol{w}) = \|\boldsymbol{w}\|/(1 + \boldsymbol{w}^\top \mathbf{V}^{-1}\boldsymbol{w})$. As soon as $\boldsymbol{w}^\top \mathbf{V}^{-1}\boldsymbol{w} \geq \lambda_{\min}\|\boldsymbol{w}\|^2$ (where we denoted by $\lambda_{\min}$ the smallest eigenvalue of $\mathbf{V}^{-1}$, so that in fact $\lambda_{\min} = \|\mathbf{V}\|^{-1}$), to find the maximum of $\tilde{\varphi}(\boldsymbol{w})$ we should maximize the expression $t/(1 + \lambda_{\min} t^2)$ for $t = \|\boldsymbol{w}\| \geq 0$. It is reached at the point $t^* = \lambda_{\min}^{-1/2}$. Substituting this value into the original expressions gives $\varphi(\boldsymbol{v}) \leq \sqrt{\|\mathbf{V}\|}/2$.

To compute the Lipschitz constant L_φ we consider the derivative,

$$\|\nabla\tilde{\varphi}'(\boldsymbol{w})\| = \frac{\left\|(1 + \boldsymbol{w}^\top \mathbf{V}^{-1}\boldsymbol{w})\boldsymbol{w} - 2\|\boldsymbol{w}\|^2 \mathbf{V}^{-1}\boldsymbol{w}\right\|}{\|\boldsymbol{w}\|(1 + \boldsymbol{w}^\top \mathbf{V}^{-1}\boldsymbol{w})^2} \leq 1 + \frac{2\|\mathbf{V}^{-1}\|\|\boldsymbol{w}\|^2}{(1 + \boldsymbol{w}^\top \mathbf{V}^{-1}\boldsymbol{w})^2},$$

from where we find that $\|\nabla\tilde{\varphi}'(\boldsymbol{w})\| \leq 1 + \mu(\mathbf{V})/2$, and so $L_\varphi = \|\mathbf{V}\|(1 + \mu(\mathbf{V})/2)$.

This lemma yields the following main result for the gradient ∇Q:

Theorem 1. *Assume functions $\mathcal{F}$ and $\mathcal{G}$ and their derivatives $\mathcal{F}'$, and $\mathcal{G}'$, are Lipschitz continuous with constants $L_\mathcal{F}$, $L_\mathcal{G}$, $L'_\mathcal{F}$, and $L'_\mathcal{G}$, respectively. Then the gradient ∇Q is bounded and Lipschitz continuous with some constant L.*

Proof. From equation 4.20, the gradient ∇Q can be written as

$$\begin{aligned}\nabla Q(\boldsymbol{s}) &= \nabla Q_\mathcal{F}(\boldsymbol{s}) + \nabla Q_\mathcal{G}(\boldsymbol{s}) = \\ &= \frac{2\bar{\alpha}\mathcal{F}'^\top(\boldsymbol{s})\mathbf{V}_f^{-1}(\bar{\boldsymbol{f}} - \mathcal{F}(\boldsymbol{s}))}{1 + D_\mathcal{F}(\boldsymbol{s})} + \frac{2\bar{\beta}\mathcal{G}'^\top(\boldsymbol{s})\mathbf{V}_g^{-1}(\bar{\boldsymbol{g}} - \mathcal{G}(\boldsymbol{s}))}{1 + D_\mathcal{G}(\boldsymbol{s})}. \qquad (4.21)\end{aligned}$$

It follows from lemma 1 that $\|\nabla Q_\mathcal{F}(\boldsymbol{s})\| \leq 2L_\mathcal{F}\bar{\alpha}C_\varphi(\mathbf{V}_f^{-1})$ and $\|\nabla Q_\mathcal{G}(\boldsymbol{s})\| \leq 2L_\mathcal{G}\bar{\beta}C_\varphi(\mathbf{V}_g^{-1})$. The norm of the gradient is then bounded by

$$\left\|\nabla Q(\boldsymbol{s})\right\| \leq 2L_\mathcal{F}\bar{\alpha}C_\varphi(\mathbf{V}_f^{-1}) + 2L_\mathcal{G}\bar{\beta}C_\varphi(\mathbf{V}_g^{-1}). \qquad (4.22)$$

Considering the norm $\|\nabla Q_\mathcal{F}(\boldsymbol{s}) - \nabla Q_\mathcal{F}(\tilde{\boldsymbol{s}})\|$, we introduce $\boldsymbol{v}_1 = \bar{\boldsymbol{f}} - \mathcal{F}(\boldsymbol{s})$ and $\boldsymbol{v}_2 = \bar{\boldsymbol{f}} - \mathcal{F}(\tilde{\boldsymbol{s}})$. Then we have

$$\begin{aligned}\left\|\nabla Q_\mathcal{F}(\boldsymbol{s}) - \nabla Q_\mathcal{F}(\tilde{\boldsymbol{s}})\right\| \leq 2\bar{\alpha}\Bigg(&\left\|\frac{(\mathcal{F}'(\boldsymbol{s}) - \mathcal{F}'(\tilde{\boldsymbol{s}}))^\top \mathbf{V}_f^{-1}\boldsymbol{v}_1}{1 + \|\boldsymbol{v}_1\|^2_{\mathbf{V}_f}}\right\| + \\ &+ \left\|\frac{\mathcal{F}'^\top(\tilde{\boldsymbol{s}})\mathbf{V}_f^{-1}\boldsymbol{v}_2}{1 + \|\boldsymbol{v}_2\|^2_{\mathbf{V}_f}} - \frac{\mathcal{F}'^\top(\tilde{\boldsymbol{s}})\mathbf{V}_f^{-1}\boldsymbol{v}_1}{1 + \|\boldsymbol{v}_1\|^2_{\mathbf{V}_f}}\right\|\Bigg). \qquad (4.23)\end{aligned}$$

Using lemma 1 with $\mathbf{V}_f^{-1}$ we have

$$\left\|\nabla Q_{\mathcal{F}}(\boldsymbol{s}) - \nabla Q_{\mathcal{F}}(\tilde{\boldsymbol{s}})\right\| \leq 2\bar{\alpha}\left(L'_{\mathcal{F}} C_{\varphi}(\mathbf{V}_f^{-1}) + L^2_{\mathcal{F}} L_{\varphi}(\mathbf{V}_f^{-1})\right)\|\boldsymbol{s} - \tilde{\boldsymbol{s}}\|.$$

The same derivations can be performed for $\nabla Q_{\mathcal{G}}(\boldsymbol{s})$, so that finally we get

$$\left\|\nabla Q_{\mathcal{G}}(\boldsymbol{s}) - \nabla Q_{\mathcal{G}}(\tilde{\boldsymbol{s}})\right\| \leq L\|\boldsymbol{s} - \tilde{\boldsymbol{s}}\|, \tag{4.24}$$

where the Lipschitz constant is given by

$$L = 2\bar{\alpha}\left(L'_{\mathcal{F}} C_{\varphi}(\mathbf{V}_f^{-1}) + L^2_{\mathcal{F}} L_{\varphi}(\mathbf{V}_f^{-1})\right) + 2\bar{\beta}\left(L'_{\mathcal{G}} C_{\varphi}(\mathbf{V}_g^{-1}) + L^2_{\mathcal{G}} L_{\varphi}(\mathbf{V}_g^{-1})\right). \tag{4.25}$$

To actually construct the nondecreasing sequence in equation 3.19, we make use of the following fundamental result on variable metric gradient ascent algorithms.

Theorem 2 (Polyak, 1987). *Let the function $Q : \mathbb{R}^d \to \mathbb{R}$ be differentiable on $\mathbb{R}^d$ and its gradient ∇Q be Lipschitz continuous with constant L. Let the matrix $\boldsymbol{H}$ be positive definite, such that $\|\boldsymbol{H}\| \leq \frac{2}{L}$. Then the sequence $Q(\tilde{\boldsymbol{s}}^{(\nu)})$, defined by $\tilde{\boldsymbol{s}}^{(\nu+1)} = \tilde{\boldsymbol{s}}^{(\nu)} + \boldsymbol{H}\nabla Q(\tilde{\boldsymbol{s}}^{(\nu)})$, is nondecreasing.*

This result shows that for any functions $\mathcal{F}$ and $\mathcal{G}$ that verify the conditions of theorem 1, using equation 3.19 with $\mathbf{H} = \frac{2}{L}\mathbf{I}$, we are able to construct a nondecreasing sequence and an appropriate *Local Search* procedure. Notice however, that its guaranteed theoretical convergence speed is linear. It can be improved in several ways.

First, the optimization direction can be adjusted. For certain problems, the matrix $\mathbf{H}$ can be chosen as in variable metric algorithms, such as the Newton-Raphson method, quasi-Newton methods, or Levenberg-Marquardt method, provided that it satisfies the conditions of theorem 2. Second, the optimization step size can be increased based on local properties of the target function. For example, at iteration ν, if when considering the functions $\mathcal{F}$ and $\mathcal{G}$ on some restricted domain $\mathbb{S}^{(\nu)}$ there exist smaller local Lipschitz constants $L_{\mathcal{F}}^{(\nu)}$, $L_{\mathcal{G}}^{(\nu)}$, $L_{\mathcal{F}}^{\prime(\nu)}$ and $L_{\mathcal{G}}^{\prime(\nu)}$, $\mathbf{H}$ can be set to $\mathbf{H} = \frac{2}{L^{(\nu)}}\mathbf{I}$ with $L^{(\nu)}$ smaller than L. It follows that $\|\tilde{\boldsymbol{s}}^{(\nu+1)} - \tilde{\boldsymbol{s}}^{(\nu)}\| \leq \frac{2}{L^{(\nu)}}\|\nabla Q(\tilde{\boldsymbol{s}}^{(\nu)})\|$, which means that one can take the local constants, $L_{\mathcal{F}}^{(\nu)}$, $L_{\mathcal{G}}^{(\nu)}$, $L_{\mathcal{F}}^{\prime(\nu)}$, and $L_{\mathcal{G}}^{\prime(\nu)}$ if they are valid in the ball $\mathbb{B}_{\rho^{(\nu)}}(\tilde{\boldsymbol{s}}^{(\nu)})$ with

$$\rho^{(\nu)} = \frac{2}{L^{(\nu)}}\left(2L_{\mathcal{F}}^{(\nu)}\bar{\alpha}C_{\varphi}(\mathbf{V}_f^{-1}) + 2L_{\mathcal{G}}^{(\nu)}\bar{\beta}C_{\varphi}(\mathbf{V}_g^{-1})\right). \tag{4.26}$$

5 Global Search and the *Choose* Procedure

Theorem 1 allows us to use the improved global random search techniques for Lipschitz continuous functions (Zhigljavsky, 1991). These algorithms are known to converge, in the sense that generated point sequences fall infinitely often into an arbitrarily small neighborhood of the optimal points set. (For more details and convergence conditions see theorem 3.2.1 and the discussion that follows in Zhigljavsky, 1991.) A proper choice of the initial value $\tilde{\mathbf{s}}^{(0)}$ not only guarantees finding the global maximum, but can also be used to increase the convergence speed. A basic strategy is to draw samples in $\mathbb{S}$, according to some sequence of distributions over $\mathbb{S}$, that verifies the convergence conditions of global random search methods. However, the speed of convergence of such an algorithm is quite low.

Global random search methods can also be significantly improved by taking into account some specificities of the target function. Indeed, in our case, function 4.20 is made of two parts for which the optimal points are known and are, respectively, $\bar{\boldsymbol{f}}$ and $\bar{\boldsymbol{g}}$. If there exists $\tilde{\mathbf{s}}^{(0)}$ such that $\tilde{\mathbf{s}}^{(0)} \in \mathcal{F}^{-1}(\bar{\boldsymbol{f}}) \cap \mathcal{G}^{-1}(\bar{\boldsymbol{g}})$, then it is the global maximum, and the M step solution is found. Otherwise one can sample $\mathbb{S}$ in the vicinity of the set $\mathcal{F}^{-1}(\bar{\boldsymbol{f}}) \cup \mathcal{G}^{-1}(\bar{\boldsymbol{g}})$ to focus on a subspace that is likely to contain the global maximum. This set is generally a union of two manifolds. For sampling methods on manifolds, we refer to Zhigljavsky (1991). An illustration of this technique is given in section 8.

Another possibility is to use a heuristic that function 4.20 does not change much after one iteration of the EM algorithm. Then the initial point $\tilde{\mathbf{s}}^{(0)}$ for the current iteration can be set to the optimal value computed at the previous iteration. However, in general, this simple strategy does not yield the global maximum, as can be seen from the results in section 9.

6 Algorithm Initialization and the *Initialize* Procedure

In this section we focus on the problem of selecting the initial values $\boldsymbol{\theta}^{(0)}$ for the model parameters. As is often the case with iterative optimization algorithms, the closer $\boldsymbol{\theta}^{(0)}$ is to the optimal parameter values, the less time the algorithm would require to converge. Within the framework of conjugate mixture models, we formulate two initialization strategies, namely, the observation space candidates (OSC) strategy and the parameter space candidates (PSC) strategy, that attempt to find a good initialization.

The OSC strategy consists of searching for cluster centers in single modality spaces $\mathbb{F}$ and $\mathbb{G}$ to further map them into the parameter space $\mathbb{S}$ and select the best candidates. More specifically, we randomly select an observation $\boldsymbol{f}_m$ (or $\boldsymbol{g}_k$) and run the mean shift algorithm (Comaniciu & Meer, 2002) in the corresponding space to find local modes of the distribution, which are called *candidates*. The sets of candidate points $\{\hat{\boldsymbol{f}}_i\}_{i\in I}$ and $\{\hat{\boldsymbol{g}}_j\}_{j\in J}$ are

further rarefied, that is, if $\|\hat{\boldsymbol{f}}_{i_1} - \hat{\boldsymbol{f}}_{i_2}\| \leq \varepsilon_{\mathrm{f}}$ for some $i_1 \neq i_2$ and for some threshold $\varepsilon > 0$, we eliminate one of these points. These rarefied sets are then mapped to $\mathbb{S}$. If one of the observation space mappings, for example $\mathcal{F}$, is noninjective, for each $\hat{\boldsymbol{f}}_i$ we need to select a point $\boldsymbol{s}_i \in \mathcal{F}^{-1}(\hat{\boldsymbol{f}}_i)$ that is the best in some sense. We consider observation density in the other observation spaces around an image of $\boldsymbol{s}_i$ as the optimality measure of $\boldsymbol{s}_i$. This can be estimated through calculation of the kth nearest neighbor distance (k-NN) in the corresponding observation space. The final step is to choose N points out of these candidates to initialize the cluster centers $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N\}$, so that the intercluster distances are maximized. This can be done using, for example, hierarchical clustering. The variances $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_N$ and $\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_N$ are then calculated by standard empirical variance formulas based on observations that are closest to the corresponding class center. The priors $\pi_1, \ldots, \pi_{N+1}$ and $\lambda_1, \ldots, \lambda_{N+1}$ are set to be equal.

The PSC strategy consists of mapping all the observations to the parameter space $\mathbb{S}$ and performing subsequent clustering in that space. More specifically, for every observation $\boldsymbol{f}_m$ and $\boldsymbol{g}_k$, we find an optimal point from the corresponding preimage $\mathcal{F}^{-1}(\boldsymbol{f}_m)$ and $\mathcal{G}^{-1}(\boldsymbol{g}_k)$. The optimality condition is the same as in the previous strategy, that is, we compare the local observation densities using k-NN distances. Then one proceeds with selecting local modes in space $\mathbb{S}$ using the mean-shift algorithm and initializing N cluster centers $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N\}$ from all the candidates thus calculated. The estimation of variances and priors is exactly the same as in the previous strategy.

The second strategy proved to be better when performing simulations (see section 10). This can be explained by possible errors in finding the preimage of an observation space point in the parameter space. Thus, mapping a rarefied set of candidates to the parameter space is less likely to make a good guess in that space than mapping all the observations and finding the candidates directly in the parameter space.

7 Estimating the Number of Components and the *Select* Procedure

To choose the N that best corresponds to the data, we perform model selection based on a criterion that resembles the BIC criterion (Schwarz, 1978). We consider the score function of the form

$$\mathrm{BIC}_N = -2\mathcal{L}(\mathbf{f}, \mathbf{g}, \hat{\boldsymbol{\theta}}_N) + D_N \log(M + K), \tag{7.1}$$

where $\hat{\boldsymbol{\theta}}_N$ is the ML estimate obtained by the proposed EM algorithm, $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta})$ is given by equation 2.15 and $D_N = N\left(d + 2 + \frac{1}{2}(r^2 + p^2 + r + p)\right)$ is the dimensionality of the model.

As in the case of (nonconjugate) gaussian mixture models, we cannot derive the criterion from the Laplace approximation of the probability

$P(\mathbf{f}, \mathbf{g}|N = N_0)$ because of the Hessian matrix of $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta})$ that is not necessarily positive definite (Aitkin & Rubin, 1985; Quinn, McLachlan, & Hjort, 1987). Nevertheless, we can use the same arguments as those used in Keribin (2000) for gaussian mixture models to show that the criterion is consistent, that is, if N_* is the number of components in the real model that generated $\mathbf{f}$ and $\mathbf{g}$, then

$$N_{\text{BIC}} \to N_* \quad \text{a.s.}, \quad \text{when} \quad M, K \to \infty, \tag{7.2}$$

provided variances $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_N, \boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_N$ are nondegenerate and the sequence $\frac{M}{M+K}$ has only one accumulation point (i.e., has a limit).

The BIC-like criterion 7.1 shows good performance on both simulated and real data (see section 10), choosing correctly the number of objects in all the cases.

8 Clustering Using Auditory and Visual Data

We illustrate the method in the case of audiovisual (AV) objects. Objects could be characterized by both their locations in space and their auditory status, that is, whether they are emitting sounds or not. These object characteristics are not directly observable and hence need to be inferred from sensor data (e.g., cameras and microphones). These sensors are based on different physical principles, they operate with different bandwidths and sampling rates, and they provide different types of information. On one side, light waves convey useful visual information only indirectly, on the premise that they reflect onto the objects' surfaces. A natural scene is composed of many objects and surfaces, and hence the task of associating visual data with objects is a difficult one. On the other side, acoustic waves convey auditory information directly from the emitter to the receiver, but the observed data are perturbed by the presence of reverberations, of other sound sources, and of background noise. Moreover, very different methods are used to extract information from these two sensor types. A wide variety of computer vision principles exists for extracting 3D points from a single image or from a pair of stereoscopic cameras (Forsyth & Ponce, 2003), but practical methods are strongly dependent on the lighting conditions and the properties of the objects' surfaces (e.g., presence or absence of texture, color, shape, reflectance). Similarly, various algorithms were developed to locate sound sources using a microphone pair based on interaural time differences (ITD) and on interaural level differences (ILD) (Wang & Brown, 2006; Christensen, Ma, Wrigley, & Barker, 2007), but these cues are difficult to interpret in natural settings due to the presence of background noise and other reverberant objects. A notable improvement consists in the use a larger number of microphones (Dibiase, Silverman, & Brandstein, 2001). Nevertheless, the extraction of 3D sound source positions from several

microphone observations results in inaccurate estimates. We show below that our method can be used to combine visual and auditory observations to detect and localize objects. A typical example where the conjugate mixture models framework may help is the task of locating several speaking persons.

Using the same notations as above, we consider two sensor spaces. The multimodal data consist of M visual observations $\mathbf{f}$ and of K auditory observations $\boldsymbol{g}$. We consider data that are recorded over a short time interval $[t_1, t_2]$, such that one can reasonably assume that the AV objects have a stationary spatial location. Nevertheless, it is not assumed here that the AV objects (e.g., speakers), are static: lip movements and head and hand gestures are tolerated. We address the problem of estimating the spatial locations of all the objects that are both seen and heard. Let N be the number of objects, and in this case each object is described by a three-dimensional parameter vector $\boldsymbol{s}_n = (x_n, y_n, z_n)^\top$.

The AV data are gathered using a pair of stereoscopic cameras and a pair of omnidirectional microphones (i.e., binocular vision and binaural hearing). A visual observation vector $\boldsymbol{f}_m = (u_m, v_m, d_m)^\top$ corresponds to a 2D image location (u_m, v_m) and to an associated binocular disparity d_m. Considering a projective camera model (Faugeras, 1993), it is straightforward to define an invertible function $\mathcal{F} : \mathbb{R}^3 \to \mathbb{R}^3$ that maps $\boldsymbol{s} = (x, y, z)^\top$ onto $\boldsymbol{f} = (u, v, d)^\top$:

$$\mathcal{F}(\boldsymbol{s}) = \left(\frac{x}{z}, \frac{y}{z}, \frac{1}{z}\right)^\top \quad \text{and} \quad \mathcal{F}^{-1}(\boldsymbol{f}) = \left(\frac{u}{d}, \frac{v}{d}, \frac{1}{d}\right)^\top. \tag{8.1}$$

This model corresponds to a rectified camera pair (Hartley & Zisserman, 2000) and it can be easily generalized to more complex binocular geometries (Hansard & Horaud, 2007, 2008). Without loss of generality, one can use a sensor-centered coordinate system to represent the object locations.

Similarly one can use the auditory equivalent of disparity, namely, the interaural time difference (ITD) widely used by auditory scene analysis methods (Wang & Brown, 2006). The function $\mathcal{G} : \mathbb{R}^3 \to \mathbb{R}$ maps $\boldsymbol{s} = (x, y, z)^\top$ onto a 1D audio observation:

$$g = \mathcal{G}(\boldsymbol{s}) = \frac{1}{c}\Big(\|\boldsymbol{s} - \boldsymbol{s}_{M_1}\| - \|\boldsymbol{s} - \boldsymbol{s}_{M_2}\|\Big). \tag{8.2}$$

Here c is the sound speed, and $\boldsymbol{s}_{M_1}$ and $\boldsymbol{s}_{M_2}$ are the 3D locations of the two microphones in the sensor-centered coordinate system. Each isosurface defined by equation 8.2 is represented by one sheet of a two-sheet hyperboloid in 3D. Hence, each audio observation g constrains the location of the auditory source to lie on a 2D manifold.

In order to perform audiovisual clustering based on the conjugate EM algorithm, theorem 1 must hold for both equations 8.1 and 8.2 namely, the

functions $\mathcal{F}$ and $\mathcal{G}$ and their derivatives are Lipschitz continuous. We prove the following theorem:

Theorem 3. *The functions $\mathcal{F}$, $\mathcal{F}'$, $\mathcal{G}$, and $\mathcal{G}'$ are Lipschitz continuous with constants $L_{\mathcal{F}} = z_{min}^{-1}\sqrt{3}$, $L'_{\mathcal{F}} = z_{min}^{-2}$, $L_{\mathcal{G}} = \|\boldsymbol{s}_{M_1} - \boldsymbol{s}_{M_2}\|(cR)^{-1}$ and $L'_{\mathcal{G}} = 3(cR)^{-1}$ in the domain $\mathbb{S} = \{|z| > z_{min} > 1\} \cap \{min\{\|\boldsymbol{s} - \boldsymbol{s}_{M_1}\|, \|\boldsymbol{s} - \boldsymbol{s}_{M_2}\|\} > R > 1\}$.*

Proof. The derivatives of $\mathcal{F}$ and $\mathcal{G}$ are given by

$$\mathcal{F}'(\boldsymbol{s}) = \frac{1}{z}\begin{bmatrix} 1 & 0 & -x/z \\ 0 & 1 & -y/z \\ 0 & 0 & -1/z \end{bmatrix} \tag{8.3}$$

$$\mathcal{G}'(\boldsymbol{s}) = \frac{1}{c}\left(\frac{\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}}{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}\|} - \frac{\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}}{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}\|}\right). \tag{8.4}$$

The eigenvalues of $\mathcal{F}'(\boldsymbol{s})$ are $1/z$ and $-1/z^2$, so $\|\mathcal{F}'(\boldsymbol{s})\| \leq \max\{z^{-1}, z^{-2}\} \leq z_{\min}^{-1}$, from which it follows that $L_{\mathcal{F}}$ can be taken as $L_{\mathcal{F}} = z_{\min}^{-1}\sqrt{3}$. Also $\|\mathcal{F}'(\boldsymbol{s}) - \mathcal{F}'(\tilde{\boldsymbol{s}})\| \leq \max\{|z^{-1} - \tilde{z}^{-1}|, |z^{-2} - \tilde{z}^{-2}|\} \leq z_{\min}^{-2}\|\boldsymbol{s} - \tilde{\boldsymbol{s}}\|$, so that $L'_{\mathcal{F}}$ can be set to $L'_{\mathcal{F}} = z_{\min}^{-2}$.

Introducing $\boldsymbol{e}_1 = \frac{\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M1}}}{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}\|}$ and $\boldsymbol{e}_2 = \frac{\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M2}}}{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}\|}$, it becomes $\|\boldsymbol{e}_1\| = \|\boldsymbol{e}_2\| = 1$ and $\mathcal{G}'(\boldsymbol{s}) = \frac{1}{c}(\boldsymbol{e}_1 - \boldsymbol{e}_2)$. Provided that $\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}\|$ and $\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}\|$ are both greater than R, it follows that $\|\mathcal{G}'(\boldsymbol{s})\| = \frac{1}{c}\|\boldsymbol{e}_1 - \boldsymbol{e}_2\| \leq \|\boldsymbol{s}_{\mathrm{M}_1} - \boldsymbol{s}_{\mathrm{M}_2}\|(cR)^{-1}$, and so $L_{\mathcal{G}} = \|\boldsymbol{s}_{\mathrm{M}_1} - \boldsymbol{s}_{\mathrm{M}_2}\|(cR)^{-1}$. Then, the second derivative of $\mathcal{G}$ is given by

$$\mathcal{G}''(\boldsymbol{s}) = \frac{1}{c\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}\|}(\mathbf{I} - \boldsymbol{e}_1\boldsymbol{e}_1^\top) - \frac{1}{c\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}\|}(\mathbf{I} - \boldsymbol{e}_2\boldsymbol{e}_2^\top).$$

so that

$$\|\mathcal{G}''(\boldsymbol{s})\| \leq \left|\frac{1}{c\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M1}}\|} - \frac{1}{c\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M2}}\|}\right| + \sup_{\|\boldsymbol{v}\|=1} \frac{2\boldsymbol{e}_1\boldsymbol{e}_1^\top\boldsymbol{v}}{c\,\min\{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M1}}\|, \|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M2}}\|\}} \leq 3(cR)^{-1},$$

and $L'_{\mathcal{G}}$ can be set to $L'_{\mathcal{G}} = 3(cR)^{-1}$.

This result shows that under some natural conditions (the AV objects should not be too close to the sensors), the conjugate EM algorithm described in section 3.3 can be applied. The constant L given by lemma 1 guarantees a certain (worst-case) convergence speed. In practice, we can use the techniques mentioned in sections 4 and 5 to accelerate the algorithm. First, to speed up the local optimization step,

local Lipschitz constants can be computed based on the current value of parameter $\tilde{\mathbf{s}}^{(\nu)}$. Equation 4.26 gives the largest possible step size $\rho^{(\nu)}$, so setting $z_{\min}^{(\nu)} = z^{(\nu)} - \rho^{(\nu)}$ and $R^{(\nu)} = \min\{\|\tilde{\mathbf{s}}^{(\nu)} - \boldsymbol{s}_{\mathrm{M}_2}\|, \|\tilde{\mathbf{s}}^{(\nu)} - \boldsymbol{s}_{\mathrm{M}_1}\|\} - \rho^{(\nu)}$, provides local Lipschitz constants that ensure the update will not quit $\mathbb{S}^{(\nu)} = \{|z| > z_{\min}^{(\nu)}\} \cap \{\min\{\|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_1}\|, \|\boldsymbol{s} - \boldsymbol{s}_{\mathrm{M}_2}\|\} > R^{(\nu)}\}$. Second, we propose four possibilities to set the initial object parameter values $\tilde{\boldsymbol{s}}_n^{(0)}$: (1) it can be taken to be the previously estimated object position $\boldsymbol{s}_n^{(q-1)}$, (2) it can be set to $\mathcal{F}^{-1}(\bar{\boldsymbol{f}})$ (as soon as $\mathcal{F}$ is injective in $\mathbb{S}$), (3) it can be found through sampling of the manifold $\mathcal{G}^{-1}(\bar{\boldsymbol{g}})$ by selecting the sampled value that gives the largest Q value, or (4) similarly through sampling directly in $\mathbb{S}$. Comparisons are reported in the following sections.

9 Experiments with Simulated Data

Our algorithm is first illustrated on simulated data. For simplicity, we consider (u, d) and (x, z) coordinates so that $\mathbb{F} \subseteq \mathbb{R}^2$ and $\mathbb{S} \subseteq \mathbb{R}^2$. Notice, however, that this preserves the projective nature of the mapping $\mathcal{F}$; it does not qualitatively affect the results and leads to a better understanding of the algorithm performance. We consider three objects defined in $\mathbb{S}$ by $\boldsymbol{s}_n$, $n = 1, 2, 3$. We simulated three cases: well-separated objects (GoodSep), partially occluded objects (PoorSep), and poor precision in visual observations for well-separated objects (PoorPrec). The ground-truth object locations (x, z) for the GoodSep and PoorPrec cases are the same: $\boldsymbol{s}_1 = (-300, 1000)$, $\boldsymbol{s}_2 = (10, 800)$ and $\boldsymbol{s}_3 = (500, 1500)$. In the PoorSep case, the coordinates are, respectively, $\boldsymbol{s}_1 = (-300, 1000)$, $\boldsymbol{s}_2 = (10, 800)$, and $\boldsymbol{s}_3 = (100, 1500)$. The data in both observation spaces $\mathbb{F}$ and $\mathbb{G}$ were simulated from a mixture model with three gaussian components and a uniform component that models the outliers. The means of the gaussian components are computed using $\mathcal{F}(\boldsymbol{s}_n)$ and $\mathcal{G}(\boldsymbol{s}_n)$, $n = 1, 2, 3$. An example of simulated data for the three mentioned configurations is shown in Figure 2: (u, d) locations of the visual observations and ITD values of the auditory observations.

9.1 Initialization. We compared two strategies, OSC and PSC, proposed in section 6. Their performance is summarized in Figure 3. It shows the mean and variance of the likelihood value $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta})$ for initial parameters $\boldsymbol{\theta}_{\mathrm{OSC}}^{(0)}$ and $\boldsymbol{\theta}_{\mathrm{PSC}}^{(0)}$ chosen by OSC and PSC strategies, respectively. For the total number of clusters $N = 1, \ldots, 5$ and different object configurations, we calculate the statistics based on 10 initializations. The analysis shows that the PSC strategy performs at least as well as the OSC strategy, or even better in some cases. Our explanation is that mappings from observation spaces to parameter space are subject to absolute (and, in our case, bounded) noise. Mapping all the observations and calculating a candidate point in the parameter space has an averaging effect and reduces the absolute error, compared to the strategy with candidate calculation being performed in

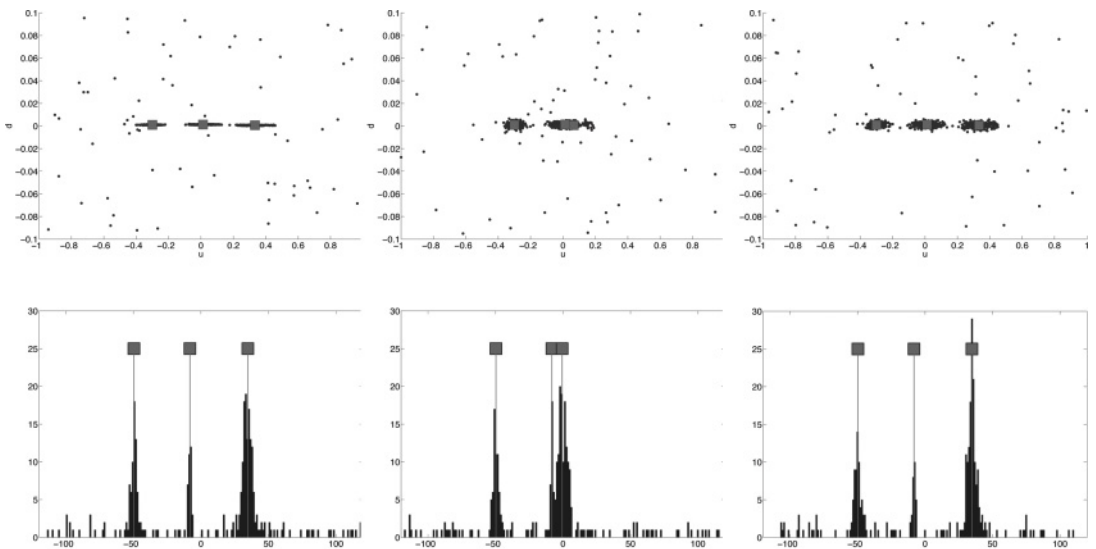


Figure 2: Simulated data in visual (top) and audio (bottom) observation spaces for three cases: (a) well-separated objects, (b) partially occluded objects, and (c) poor precision of visual observations. The small squares correspond to the ground-truth parameter values. Each one of the two mixtures models (associated with each sensorial modality) contains four components: three objects and one outlier class.

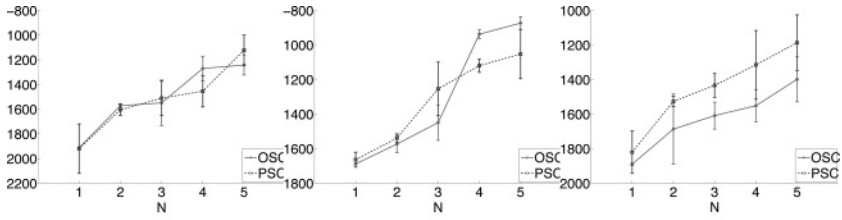


Figure 3: Means and variances of log-likelihood values $\mathcal{L}(\mathbf{f}, \mathbf{g}, \boldsymbol{\theta})$ for initial parameters $\boldsymbol{\theta}^{(0)}_{\text{OSC}}$ and $\boldsymbol{\theta}^{(0)}_{\text{PSC}}$ chosen by OSC (solid line) and (PSC, dashed line) strategies respectively, for different numbers of clusters N and different data configurations.

an observation space with subsequent mapping to the parameter space. Therefore, in what follows, all the results are obtained based on the PSC initialization strategy.

9.2 Optimization. We compared several versions of the algorithm based on various *Choose* and *Local Search* strategies. For the initial values $\tilde{s_n}^{(0)}$, we considered the following possibilities: the optimal value computed at a previous run of the algorithm (IP), the value predicted from visual data (IV), the value predicted from audio data (IA), and the value obtained by global random search (IG). More specifically:

- When initializing from visual data (IV), the average value $\bar{f}_n$, calculated in the current E-step of the algorithm for every n, was mapped to the parameter space, and $\tilde{s_n}^{(0)}$ was set to $\tilde{s_n}^{(0)} = \mathcal{F}^{-1}(\bar{f}_n)$ using the injectivity of $\mathcal{F}$.
- When initializing from audio data (IA), $\mathcal{G}^{-1}(\bar{g}_n)$ defines a manifold. The general strategy here would be to find the optimal point that lies on this surface. We achieved this through random search based on a uniform sampling on the corresponding part of the hyperboloid (see Zhigljavsky, 1991, for details on sampling from an arbitrary distribution on a manifold). In our experiments we used 50 samples to select the one providing the largest Q (likelihood) value.
- The most general initialization scheme (IG) was implemented using global random search in the whole parameter space $\mathbb{S}$; 200 samples were used in this case.

Local optimization was performed using either basic gradient ascent (BA) or the locally accelerated gradient ascent (AA). The latter used the local Lipschitz constants to augment the step size, as described in section 4. Each algorithm run consisted of 70 iterations of the EM algorithm, with 10 nondecreasing iterations during the M step.

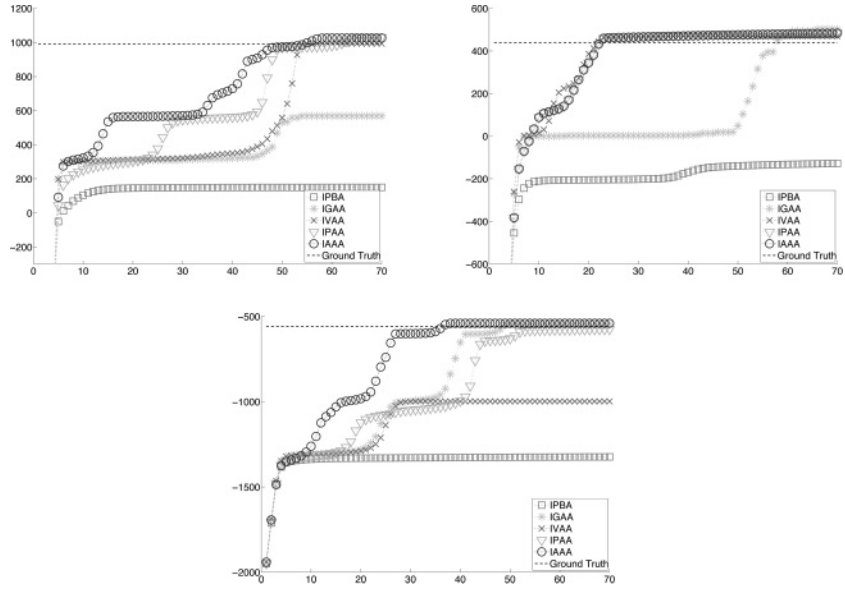


Figure 4: Likelihood function evolution for five variants of the algorithm in three cases. (Top left) Well-separated objects. (Top right) Poorly separated objects. (Bottom) Well-separated object but poor observation precision.

Table 1: Acronyms Used for Five Variants of the Conjugate EM Algorithm.

Acronym	$\tilde{\mathbf{s}}^{(0)}$ Initialization (*Choose*)	Local Optimization (*Search*)
IPBA	Previous iteration value	Basic gradient ascent
IGAA	Global random search	Accelerated gradient ascent
IVAA	Predicted value from visual data	Accelerated gradient ascent
IPAA	Previous iteration value	Accelerated gradient ascent
IAAA	Audio predicted manifold sampling	Accelerated gradient ascent

Note: Variants correspond to different choices for the *Choose* and *Local search* procedures.

To check the convergence speed of different versions of the algorithm for the three object configurations, we compared the likelihood evolution graphs presented in Figure 4. Each graph contains several curves that correspond to five different versions of the algorithm. The acronyms we use to refer to the different versions (for example, IPAA) consist of two parts encoding the initialization (IP) and the local optimization (AA) types. The black dashed line on each graph shows the ground-truth likelihood level, that is, the likelihood value for the parameters used to generate the data. The meaning of the acronyms is set out in Table 1.

As expected, the simplest version IPBA that uses none of the proposed acceleration techniques appears to be the slowest. The other variants using basic gradient ascent are then not reported. Predicting a single object parameter value from visual observations (IVAA) does not give any improvement over IPAA, where $\tilde{\mathbf{s}}^{(0)}$ is taken from the previous EM iteration. When $\tilde{\mathbf{s}}^{(0)}$ is obtained by sampling the hyperboloid predicted from audio observations (IAAA), a significant impact on the convergence speed is observed, especially on early stages of the algorithm, where the predicted value can be quite far from the optimal one. However, blind sampling of the whole parameter space does not bring any advantage: it is much less efficient regarding the number of samples required for the same precision. This suggests that in the general case, the best strategy would be to sample the manifolds $\mathcal{F}^{-1}(\bar{\boldsymbol{f}}_n)$ and $\mathcal{G}^{-1}(\bar{\boldsymbol{g}}_n)$ with possible small perturbations to find the best $\tilde{\mathbf{s}}^{(0)}$ estimate and perform an accelerated gradient ascent afterward (IAAA). We note that IAAA succeeds in all the cases to find parameter values that are well fitted to the model in terms of likelihood function (likelihood is greater or equal than that of real parameter values).

Parameter evolution trajectories for the IAAA version of the algorithm in the Good-Sep case are shown in Figures 5 and 6. The estimate changes are reflected by the node sizes (from smaller to bigger) and shading (from darker to lighter). The final values are very close to the real cluster centers in all three audio, visual, and object spaces. The convergence speed is quite dependent on the initialization. In the provided example, the algorithm spent almost half of useful iterations to disentangle the estimates trying to decide which one corresponds to which class. Another possibility here would be to predict the initial values through sampling in the audio domain. We demonstrate this strategy further when working with real data.

We compared the performance of our algorithm for the three object configurations. For each of them, we computed absolute and relative errors for the object parameter estimations in the different coordinate systems (object, audio, and visual spaces). The averages were taken over 10 runs of the algorithm for different PSC initializations, as described above. The results are reported in Table 2. We give object location estimates $\hat{\mathbf{s}} = (\hat{x}, \hat{z})$, $\hat{\boldsymbol{f}} = (\hat{u}, \hat{d})$ and $\hat{\boldsymbol{g}}$ in parameter, visual, and audio spaces respectively. It appears that the localization precision is quite high. In a realistic setting such as that of section 10, the measurement unit can be set to a millimeter. In that case, the observed precision, in a well-separated objects configuration, is at worse about 6 cm. However, precision in the z coordinate is quite sensible to the variance of the visual data and the object configuration. To get a better idea of the relationship between the variance in object space and the variance in visual space, $\mathcal{F}^{-1}$ can be replaced by its linear approximation given by a first-order Taylor expansion. Assuming then that visual data are distributed according to some probability distribution with mean $\mu_{\mathcal{F}}$ and variance $\Sigma_{\mathcal{F}}$, it follows that through the linear approximation of $\mathcal{F}^{-1}$, the variance in object space is $\frac{\partial \mathcal{F}^{-1}(\mu_{\mathcal{F}})}{\partial \boldsymbol{f}} \boldsymbol{\Sigma}_{\mathcal{F}} \frac{\partial \mathcal{F}^{-1}(\mu_{\mathcal{F}})}{\partial \boldsymbol{f}}^{\top}$. Then the z coordinate

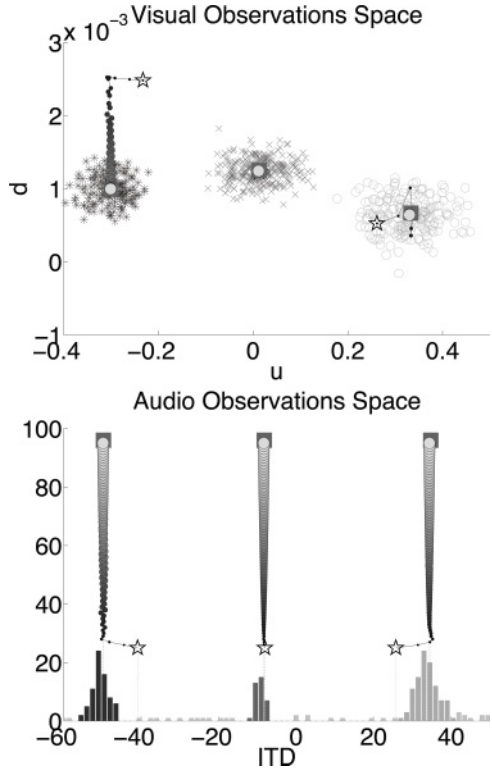


Figure 5: IAAA algorithm: Parameter evolution and assignment results for the GoodSep case in audio and visual spaces (note the scale change, which corresponds to a zoom on the cluster centers). The initialization (white stars) is based on the PSC strategy. Ground truth means are marked with squares. The evolution is shown by circles from smaller to bigger, from darker to brighter. Observations assignments are depicted by different markers ($\circ$, $*$, and $\times$ for the three object classes) in visual space and are coded by shading in audio space. Due to the zoom, outliers are not visible on these figures.

covariance for an object n is approximately proportional to the d covariance for the object multiplied by z_n^4. For distant objects, a very high precision in d is needed to get a satisfactory precision in z. At the same time, we observe that the likelihood of the estimate configuration often exceeds the likelihood for real parameter values. This suggests that the model performs well for the given data, but cannot get better precision than that imposed by the data.

9.3 Selection. To select the optimal number of clusters N, we applied the BIC criterion 7.1 to the models, trained for that N. The BIC score graphs are shown on Figure 7. The total number of objects N is correctly determined

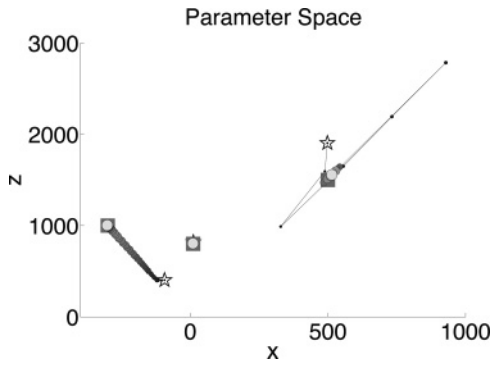


Figure 6: IAAA algorithm: Parameter evolution for the GoodSep case in object space. The initialization (white stars) is based on the PSC strategy. Ground truth means are marked with squares. The evolution is shown by circles from smaller to bigger, from darker to brighter.

in all three cases of object configurations, from which we conclude that the BIC criterion provides reliable model selection in our case.

10 Experiments with Real Data

In this section we evaluate the effectiveness of our algorithms in estimating the 3D locations of AV objects—a person localization task. The examples used below are from a database of realistic AV scenarios described in detail in Arnaud et al. (2008).

The experimental setup consists of a mannequin equipped with a pair of microphones fixed into its ears and a pair of stereoscopic cameras mounted on its forehead (this device was developed within the POP project).[1] Each data set comprises two audio tracks, two image sequences, and the calibration information. All the recordings were performed in an ordinary room with no special adjustments to its acoustics or appearance. Thus, the data contain both visual background information and auditory noise—reverberations in particular. This configuration best mimics what a person would hear and see in a standard indoor environment.

We tested our multimodal clustering method with two scenarios: a meeting and a cocktail party (see Table 3):

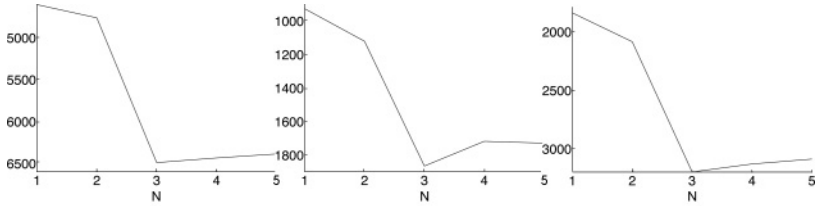
- The meeting scenario is a recording of a discussion held by five persons sitting around a table, only three of them being visible.[2] It lasts 25 seconds and contains a about 8000 visual and 600 audio observations.

[1] http://perception.inrialpes.fr/POP/.
[2] http://perception.inrialpes.fr/CAVA_Dataset/Site/data.html#M1.

Table 2: IAAA Algorithm: Object Location Estimates in Parameter, Visual, and Audio Spaces for GoodSep, PoorSep, and PoorPrec Object Configurations.

Parameter Space	Ground Truth $\boldsymbol{s} = (x, z)$	Estimates Mean $\hat{\boldsymbol{s}} = (\hat{x}, \hat{z})$	Absolute Error $e_{\mathrm{a}} = \|\hat{\boldsymbol{s}} - \boldsymbol{s}\|$	Relative Error $e_{\mathrm{r}} = \|\hat{\boldsymbol{s}} - \boldsymbol{s}\|/\|\boldsymbol{s}\|$
GoodSep				
Object 1	$(-300, 1000)$	$(-300.13, 997.81)$	2.2	2.1×10^{-3}
Object 2	$(10, 800)$	$(9.28, 804.46)$	4.52	5.7×10^{-3}
Object 3	$(500, 1500)$	$(513.56, 1555.23)$	56.86	3.5×10^{-2}
PoorSep				
Object 1	$(-300, 1000)$	$(-307.47, 1028.38)$	29.35	2.8×10^{-2}
Object 2	$(10, 800)$	$(14.19, 895.69)$	95.79	1.2×10^{-1}
Object 3	$(100, 1500)$	$(105.02, 1447.49)$	52.75	3.5×10^{-2}
PoorPrec				
Object 1	$(-300, 1000)$	$(-208.86, 698.51)$	314.97	0.3
Object 2	$(10, 800)$	$(8.44, 703.97)$	96.04	1.2×10^{-1}
Object 3	$(500, 1500)$	$(507.65, 1533.8)$	34.66	2.2×10^{-2}
Visual Space	$\boldsymbol{f} = (u, d)$	$\hat{\boldsymbol{f}} = (\hat{u}, \hat{d})$	$e_{\mathrm{a}} = \|\hat{\boldsymbol{f}} - \boldsymbol{f}\|$	$e_{\mathrm{r}} = \|\hat{\boldsymbol{f}} - \boldsymbol{f}\|/\|\boldsymbol{f}\|$
GoodSep				
Object 1	$(-0.3, 0.001)$	$(-0.3008, 0.001)$	7.87×10^{-4}	2.6×10^{-3}
Object 2	$(0.0125, 0.00125)$	$(0.0115, 0.00124)$	9.59×10^{-4}	7.6×10^{-2}
Object 3	$(0.3333, 0.00067)$	$(0.3302, 0.00064)$	31.21×10^{-4}	9.3×10^{-3}
PoorSep				
Object 1	$(-0.3, 0.001)$	$(-0.299, 0.001)$	1.02×10^{-3}	3.4×10^{-3}
Object 2	$(0.0125, 0.00125)$	$(0.0159, 0.00112)$	3.36×10^{-3}	2.6×10^{-1}
Object 3	$(0.6667, 0.00067)$	$(0.7131, 0.00238)$	4.95×10^{-3}	7.4×10^{-2}
PoorPrec				
Object 1	$(-0.3, 0.001)$	$(-0.299, 0.0014)$	10.8×10^{-4}	3.5×10^{-3}
Object 2	$(0.0125, 0.00125)$	$(0.012, 0.00142)$	5.38×10^{-4}	4.3×10^{-2}
Object 3	$(0.3333, 0.00067)$	$(0.331, 0.00065)$	23.56×10^{-4}	7.1×10^{-3}
Audio Space	g	$\hat{g}$	$e_{\mathrm{a}} = \|\hat{g} - g\|$	$e_{\mathrm{r}} = \|\hat{g} - g\|/\|g\|$
GoodSep				
Object 1	-49.71	-49.8	0.09	1.9×10^{-3}
Object 2	-8.22	-8.35	0.13	1.6×10^{-2}
Object 3	34.75	34.37	0.38	1.1×10^{-2}
PoorSep				
Object 1	-49.71	-49.59	0.12	2.3×10^{-3}
Object 2	-8.22	-7.76	0.46	5.6×10^{-2}
Object 3	-0.66	-0.02	0.65	9.7×10^{-1}
PoorPrec				
Object 1	-49.71	-49.49	0.22	4.4×10^{-3}
Object 2	-8.22	-8.28	0.06	7.6×10^{-3}
Object 3	34.75	34.47	0.29	8.3×10^{-3}

Note: The estimates are calculated based on 10 runs of the algorithm with PSC initializations.



(a) GoodSep (b) PoorSep (c) PoorPrec

Figure 7: BIC score graphs for the three object configurations, evaluated for models trained for different total number of clusters N.

Table 3: Summary of the Main Characteristics of the Two Scenarios Used to Evaluate the Multimodal Clustering Algorithm.

Scenario	Visible Persons	Speaking Persons	Visual Background	Audio Noise	Occluded Speakers	Audio Overlap
Meeting	3	5	Yes	Yes	No	Yes
Cocktail party	3	3	Yes	Yes	Yes	Yes

The three visible persons perform head and body movements while taking speech turns. Sometimes two persons (visible or not) speak simultaneously.

- The cocktail party scenario shows a dynamic scene with three persons walking in a room and taking speech turns.[3] Occasionally one speaker is hidden by another person and two persons may speak simultaneously. Speakers may go in and out of the two cameras' field of view. Moreover, there are sounds emitted by the persons' steps. The recording lasts 30 seconds and contains about 12,500 visual and 3400 audio observations.

10.1 Preprocessing and Algorithm Initialization. Visual observations, $\mathbf{f}$, are obtained as follows. First, we detect points of interest (POI) in both the left and right images and select those points that correspond to a moving scene object. Second, we perform stereo matching such that a disparity value is associated with each matched point.

In practice we used the POI detector described in Harris and Stephens (1988). This detector is known to have high repeatability in the presence of texture and to be photometric invariant. We analyze each image point detected this way and select points associated with a significant motion pattern. Motion patterns are obtained in a straightforward manner. A temporal intensity variance σ_t is estimated at each POI. Assuming stable lighting

[3] http://perception.inrialpes.fr/CAVA_Dataset/Site/data.html#CTMS3.

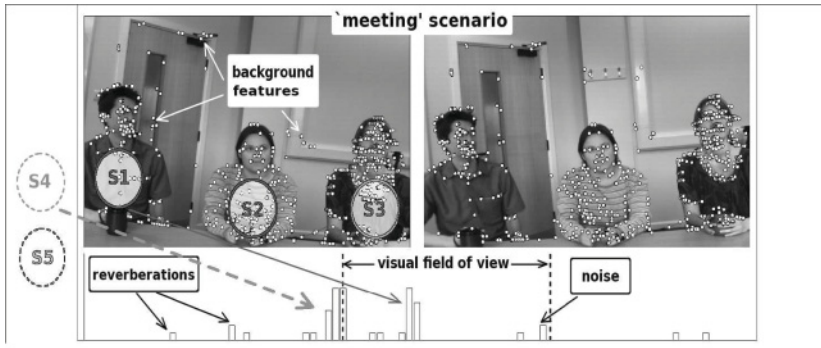
conditions, the POI belongs to a static scene object if its temporal intensity variance is low and nonzero due to a camera noise only. For image points belonging to a dynamic scene object, the local variance is higher and depends on the texture of the moving object and the motion speed. In our experiments, we estimated the local temporal intensity variance σ_t at each POI from a collection of five consecutive frames. The point is labeled "motion" if $\sigma_t > 5$ (for 8-bit gray-scale images); otherwise it is labeled as "static." The motion-labeled points are then matched and the associated disparities estimated using standard stereo methods. In practice, the results shown in this letter are obtained with the method described in Hansard and Horaud (2007) using the INTEL's OpenCV camera calibration software.[4] Overall, this provides the $(u, v, d)^\top$ to $(x, y, z)^\top$ mapping (see equation 8.1). Examples are shown in Figure 8. Alternatively, we could have used the spatiotemporal point detector described in Laptev (2005). This method is designed to detect points in a video stream having large local variance in both the spatial and temporal domains, thus representing abrupt events in the stream. However, such points are quite rare in our data set.

Auditory observations, $\mathbf{g}$, are obtained as follows. Our method uses interaural time differences (ITD), which are detected through the analysis of the cross-correlogram of the filtered left- and right-microphone one-dimensional signals for every frequency band (Christensen et al., 2007). Like any other audiovisual fusion method, one needs to perform audiovisual calibration to estimate the positions of the microphones and the positions and orientations of the cameras in a common world coordinate system. This is done using the method described in Arnaud et al. (2008).

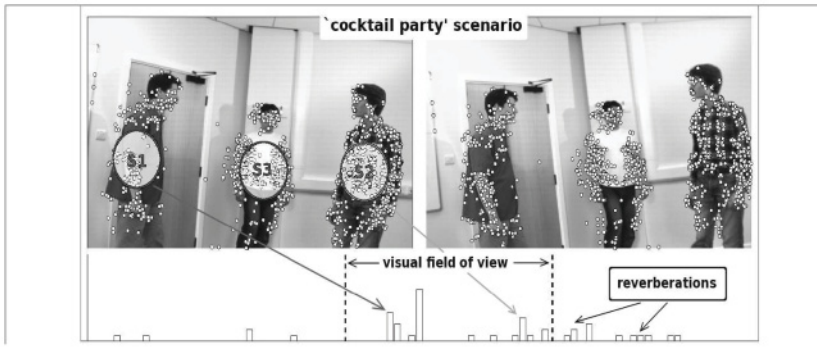
In order to initialize the algorithm's parameter values, we used the PSC initialization strategy described in section 6. Although real-data distributions do not strictly correspond to the case of gaussian mixtures, the initialization strategy that we have adopted remains relevant. This originates from the fact that parameter space sampling with configuration restrictions plays the role of a global optimization method similar to Monte Carlo sampling in the method of generations (Zhigljavsky & Žilinskas, 2008). It helps to avoid local maxima and allows quickly finding a set of appropriate initial parameters. Local distribution density modes occur to be good candidates to initialize cluster centers. As in the case of simulated data, we used the BIC score (see section 7) to select the optimal number of audiovisual clusters.

10.2 Results and Discussion. The experimental validation described below was performed with two goals in mind. First, we wanted to check that our method was stable and robust with real data gathered in complex situations, that it correctly finds the number of clusters, and that it efficiently

[4] http://www.intel.com/technology/computing/opencv.



(a) Meeting: There are five speakers but only three are visible.



(b) Cocktail party: The three speakers walk in the room.

Figure 8: This figure illustrates how the audiovisual data are preprocessed. Visual points of interest (POI) associated with scene motion are matched between the left and right images. The histograms of the interaural time difference (ITD) observations correspond to a segment of 0.3 seconds. The audiovisual calibration allows us to filter out auditory data that fall outside the field of view of the two cameras. Notice the large number of auditory perturbations corresponding to noise, reverberations, as well as to speakers outside the visual field of view. In these examples, there are two simultaneous speakers: (a) S1 and S4 and (b) S1 and S2. Notice that S4 is easily eliminated because its associated ITD falls outside the visual field of view.

determines the model's parameters—the 3D positions of the audiovisual objects composing a scene. Second, we wanted to test the model's capability to deal with dynamic changes in the scene, yet in the presence of acoustic noise and reverberations and visually occluded persons, for example. Below we provide a detailed account of the results obtained with the meeting and cocktail party audiovisual sequences.

The audiovisual recordings are split into segments, each segment lasting 0.3 seconds. At 25 frames per second, this corresponds to approximately

eight video frames. The initialization method described in section 6 and the model selection method described in section 7 are combined and applied to the first segment in order to find initial parameter values and estimate the number of components (the number of audiovisual objects) to be used by the conjugate EM algorithm. Consequently, the parameters estimated for one segment are used to initialize the parameters for the next segment, while the number of components remains constant:

- *Quasi-static scene*. The meeting situation corresponds to the well-separated case referred to as *GoodSep* in the previous section. The initialization strategy performs well, and the candidate configuration obtained by the initialization step is relatively close to the optimal one found by the EM algorithm described in detail in section 3.3. In fact, the likelihood evolution reported in Figure 9 shows that convergence is reached in about 20 iterations of EM, three times faster than in the simulated GoodSep case reported in Figure 4. The 3D position estimates are quite accurate; in particular, the natural alignment of the speakers along the table is clearly seen in the XZ plane. Although in practice, the data are not piecewise gaussian and the outliers are not uniformly distributed, our method performs quite well, which illustrates its robustness when dealing with real data distributions. Figure 10 shows sequential results obtained in this case. The speech sources are correctly detected even in the case when two persons are simultaneously active.
- *Dynamic scene*. The cocktail party situation corresponds to the partially occluded case, referred to as *PoorSep* in the previous section. In this case, the locations of the audiovisual objects vary over time, as well as their number. Nevertheless, we assume that these changes are rather slow. We did not attempt to tune our algorithm to the dynamic case. Hence, we use the same initialization strategy as in the quasi-static case which is briefly summarized above. Figure 11 shows the results obtained in this case.

Overall, the proposed method performs well on data collected in a natural environment. The initialization strategy and the model selection criterion proved to be robust to noise and minor deviations from the gaussian distribution assumption. It possesses the features of a global optimization method that enables finding initial parameter values that are close to optimal ones. In both examples, the parameter initialization and model selection were performed on the first audiovisual data segment. This certainly biases the overall results. Indeed, in both cases, the initialization and model selection algorithms dealt with a case where the objects were well separated. One could rerun initialization and model selection on every data segment at the cost of a less efficient procedure.

The conjugate clustering method automatically weights the auditory and visual modalities, in terms of precision and amount of observations,

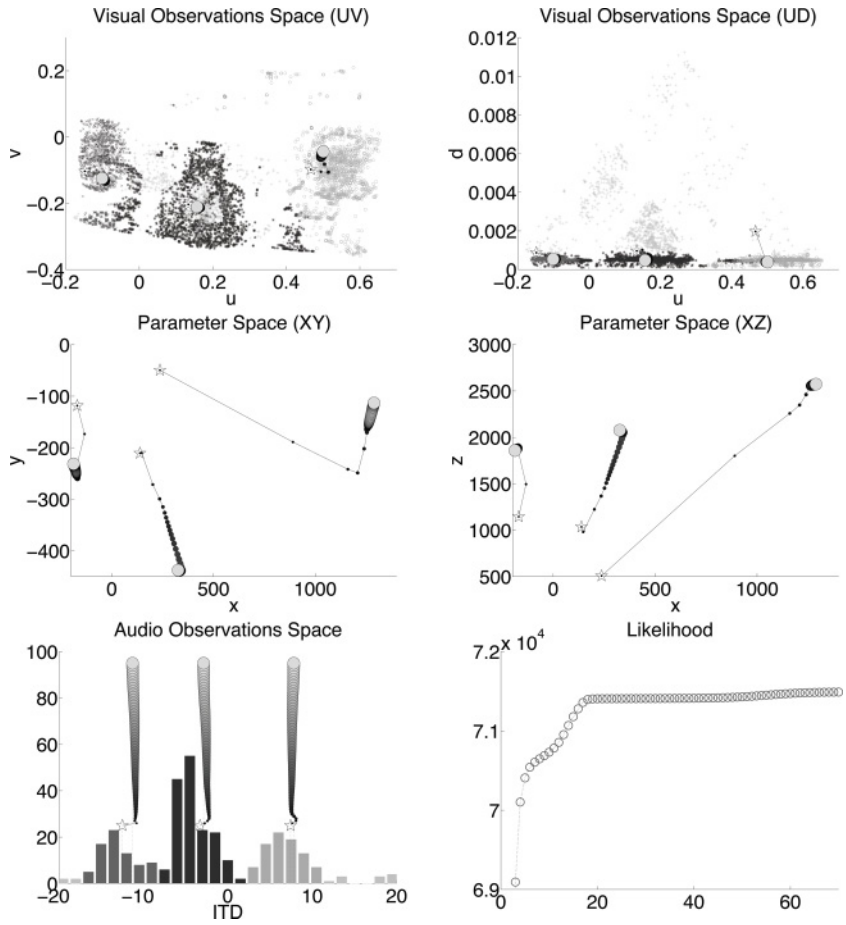


Figure 9: An example of applying the proposed EM algorithm to a time interval of 20 seconds of the meeting scenario. The results are shown in the visual and auditory observation spaces as well as in the parameter space. The initial parameter values are shown with three stars, while the parameter evolution trajectories are shown with circles of increasing size. The final observation-to-cluster assignments are shown in dark colors for the three gaussian components and in light color for the outlier component. The log-likelihood curve (bottom right) shows that the algorithm converged after 20 iterations.

to infer the parameter values. We noticed that in general, the visual data are considered by the algorithm as more reliable. This can be explained by the fact that in practice, the auditory signals are contaminated with noise and reverberations. This typically smooths the histogram peaks in the ITD

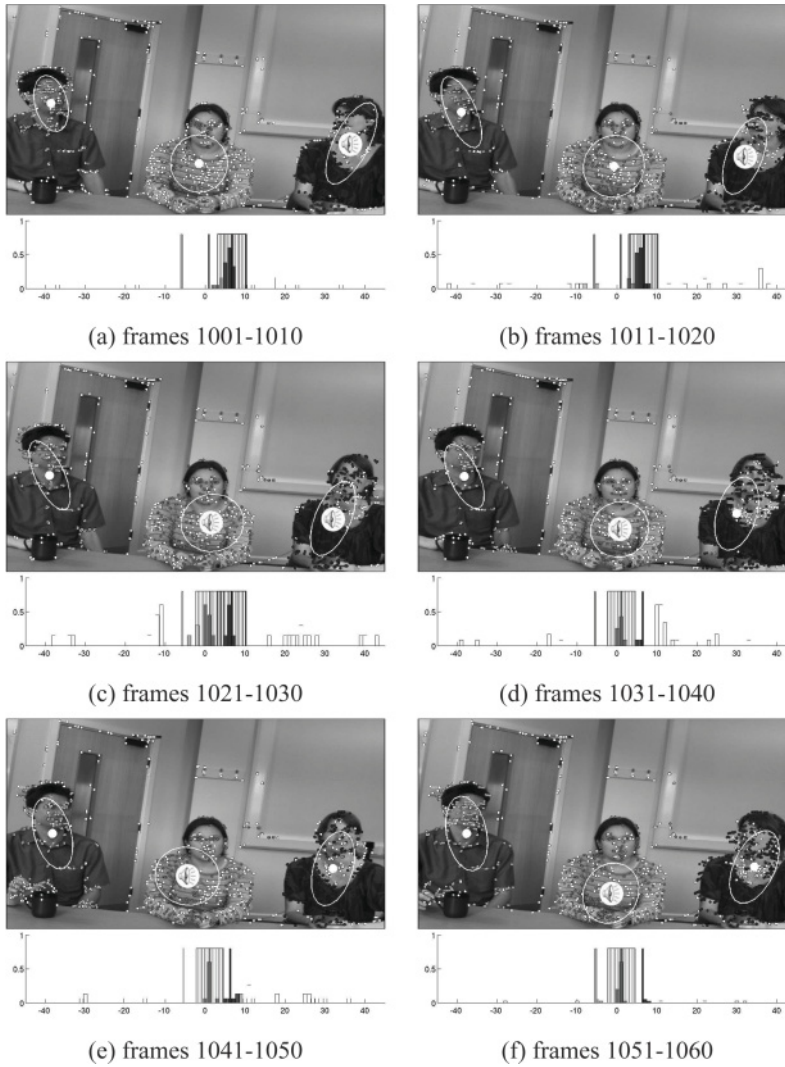


Figure 10: Results obtained in the case of the meeting scenario shown overlapped onto the left image. Sixty frames (1001 to 1060) were split into six segments. Parameter initialization and model selection were performed on the first segment (frames 1–10) and are not shown. The "visual" covariance matrices associated with the three gaussian components are projected onto the image plane. The white dots correspond to the projected 3D locations estimated by the algorithm. The gray-level dots encode the observation-to-cluster assignments, and the active speaker is marked with a corresponding symbol. The algorithm correctly estimates speech sources, even in the case when two speakers are active.

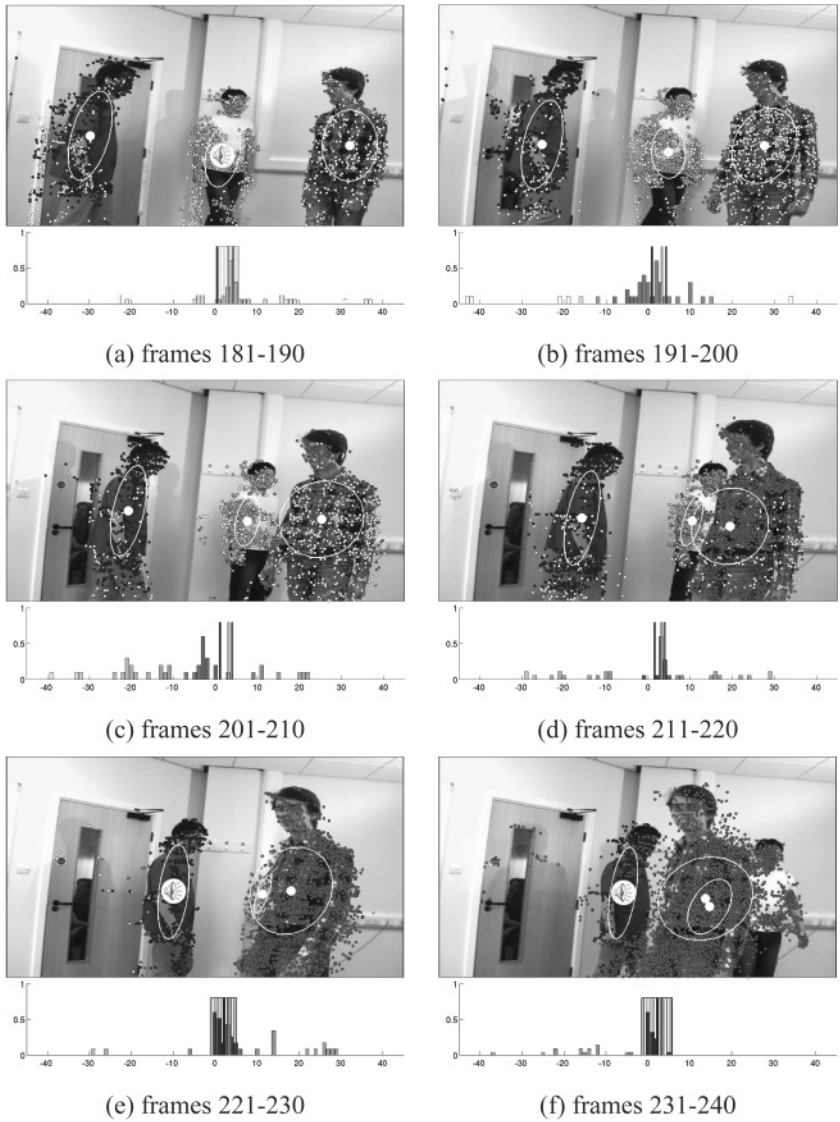


Figure 11: Results obtained in the case of the cocktail party scenario shown overlapped onto the left image. As in the previous case, sixty frames (181 to 240) were split into six segments. Parameter initialization and model selection were performed on the first segment (frames 1–10) and are not shown. As expected, well-separated objects, (a–c), are correctly handled. Although partial occlusion, (d–e) is also handled correctly, the algorithm fails to deal with a complete occlusion, (f).

domain and adds false peaks, as can be seen in Figures 10 and 11. As reverberations are natural for most of the environments and sound sources, we added auditory cluster variances to model the local smoothing effect, as well as an outlier category to treat false peaks. In general, if the data are gathered using a small time interval, reverberations and noise have a higher effect, the observations are scattered, and auditory spatial localization is poor. At the same time, widening the time interval would result in sharper peaks for sound sources that are smoothed due to reverberations and dynamics of the scene, and hence the auditory temporal localization will be less accurate. Thus, the auditory data are typically sparse in both time and space. The temporal discontinuity of the auditory data, together with the lack of resolution, makes it less reliable than the visual data.

Although our multimodal clustering model has no built-in dynamic capability, as is the case with target-tracking methods based on the Kalman filter, the implemented algorithm performs quite well in the case of partial visual occlusions, as illustrated in the cocktail party scenario.

11 Conclusion

We proposed a novel framework to cluster heterogeneous data gathered with physically different sensors. Our approach differs from other existing approaches in that it combines in a single statistical model a number of clustering tasks while ensuring the consistency of their results. In addition, the fact that the clustering is performed in observation spaces allows one to get useful statistics on the data, an advantage of our approach over particle filtering models. The task of simultaneous clustering in spaces of different nature, related through known functional dependencies to a common parameter space, was formulated as a likelihood maximization problem. Using the ideas underlying the classical EM algorithm, we built the conjugate EM algorithm to perform the multimodal clustering task while keeping attractive convergence properties. The analysis of the conjugate EM algorithm and, more specifically, of the optimization task arising in the M-step, revealed several possibilities for increasing convergence speed. We proposed to decompose the M-step into two procedures, the *Local Search* and *Choose* procedures, which allowed us to derive a number of acceleration strategies. We exhibited appealing properties of the target function, which induced several implementations of these procedures, resulting in a significantly improved convergence speed. We introduced the *Initialize* and *Select* procedures to efficiently choose initial parameter values and determine the number of clusters in a consistent manner, respectively. A nontrivial audiovisual localization task was considered to illustrate the conjugate EM performance on both simulated and real data. Simulated data experiments allowed us to assess the average method behavior in various configurations.

They showed that the obtained clustering results were precise regarding the observation spaces under consideration. They also illustrated the theoretical dependency between the precisions in observation and parameter spaces. Real data experiments then showed that the observed data precision was high enough to guarantee high precision in the parameter space.

One of the strong points of the formulated model is that it is open to different useful extensions. It can be easily extended to an arbitrary number J of observation spaces $\mathbb{F}_1, \ldots, \mathbb{F}_J$. The main results, including *Local Search* and *Choose* acceleration strategies, stay valid with minor changes. The sum of two terms, related to spaces $\mathbb{F}$ and $\mathbb{G}$, would have to be replaced by a sum of J terms corresponding to $\mathbb{F}_1, \ldots, \mathbb{F}_J$ in the formulas of section 3.

In particular, adding gaussian priors on parameters (i.e., priors, covariance matrices, and object locations) would not essentially change the formulas. For a large class of dynamics equations, the update expressions 3.13 to 3.16 for priors and variances will remain in closed form, whereas the function $Q_n^{(q)}(\boldsymbol{s})$ in equation 3.18 will receive an additional term $\log P(\boldsymbol{s})$. For instance, multimodal dynamic inference of parameter values for Brownian dynamics (van Kampen, 2007) can be performed by means of the formulated model. Gaussian priors would add a quadratic term similar to the others in equation 3.18 that can be viewed as an observation from the ambient space modality. Thus, the optimization algorithm would not require any changes and would give an unbiased estimate.

Also, the assumption that assignment variables $\boldsymbol{a}$ and $\boldsymbol{b}$ are independent could be relaxed. An appropriate approach to perform inference in a nonindependent case would be to consider variational approximations (Jordan et al., 1998) and in particular a variational EM (VEM) framework. The general idea would be to approximate the joint distribution $P(\boldsymbol{a})$ by a distribution from a restricted class of probability distributions that factorize as $\tilde{P}(\boldsymbol{a}) = \prod_{m=1}^{M} \tilde{P}(a_m)$. For any such distribution, our model would be applicable without any changes so that for a variational version of the conjugate EM algorithm, all the results from section 3 would hold.

It appears that as a generalization of gaussian mixture models, our model has larger modeling capabilities. It is entirely based on a mathematical framework in which each step is theoretically well founded. Its ability to provide good results in a nontrivial multimodal clustering task is particularly promising for applications requiring the integration of several heterogeneous information sources. Therefore, it has advantages over other methods that include ad hoc processing while being open to the incorporation of more task dependent information.

Acknowledgments

We thank Miles Hansard (INRIA Grenoble Rhône-Alpes) and Heidi Christensen (Department of Computer Science, University of Sheffield) for providing their software for visual and auditory feature detection. This work

was supported by the Perception-on-Purpose project, under EU grant FP6-IST-2004-027268.

References

Aitkin, M., & Rubin, D. (1985). Estimation and hypothesis testing in finite mixture models. *Journal of the Royal Statistical Society. Series B (Methodological), 47*(1), 67–75.

Allen, P. (1995). Integrating vision and touch for object recognition tasks. In R. C. Luo & M. G. Kay (Eds.), *Multisensor integration and fusion for intelligent machines and systems* (pp. 407–440). New York: Ablex.

Anastasio, T. J., Patton, P. E., & Belkacem-Boussaid, K. E. (2000). Using Bayes' rule to model multisensory enhancement in the superior colliculus. *Neural Computation, 12*(5), 1165–1187.

Arnaud, E., Christensen, H., Lu, Y., Barker, J., Khalidov, V., Hansard, M., et al. (2008). The CAVA corpus: Synchronized stereoscopic and binaural datasets with head movements. In *Proceedings of the ACM/IEEE Tenth International Conference on Multimodal Interfaces* New York: ACM.

Beal, M., Jojic, N., & Attias, H. (2003). A graphical model for audiovisual object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25*(7), 828–836.

Bishop, C. (2006). *Pattern recognition and machine learning*. New York: Springer.

Boyles, R. (1983). On the convergence of EM algorithms. *Journal of the Royal Statistical Society: Series B, 45*(1), 47–50.

Castellanos, J., & Tardos, J. (1999). *Mobile robot localization and map building: A multisensor fusion approach*. Norwood, MA: Kluwer.

Celeux, G., Forbes, F., & Peyrard, N. (2003). EM procedures using mean field–like approximations for Markov model-based image segmentation. *Pattern Recognition, 36*(1), 131–144.

Checka, N., Wilson, K., Siracusa, M., & Darrell, T. (2004). Multiple person and speaker activity tracking with a particle filter. In *Proc. of IEEE Conference on Acoustics, Speech, and Signal Processing*, (pp. 881–884). Piscataway, NJ: IEEE.

Chen, Y., & Rui, Y. (2004). Real-time speaker tracking using particle filter sensor fusion. *Proceedings of IEEE, 92*(3), 485–494.

Christensen, H., Ma, N., Wrigley, S., & Barker, J. (2007). Integrating pitch and localisation cues at a speech fragment level. In *Proc. of Interspeech* (pp. 2769–2772) New York: ACM.

Coiras, E., Baralli, F., & Evans, B. (2007). Rigid data association for shallow water surveys. *IET Radar, Sonar and Navigation, 1*(5), 354–361.

Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach towards feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(5), 603–619.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society: Series B, 39*(1), 1–38.

Dibiase, J., Silverman, H., & Brandstein, M. (2001). Robust localization in reverberant rooms. In M. Brandstein & D. Ward (Eds.), *Microphone Arrays: Signal Processing Techniques and Applications*. New York: Springer.

Ernst, M. O., & Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature, 415*, 429–433.

Faugeras, O. D. (1993). *Three dimensional computer vision: A geometric viewpoint*. Cambridge, MA: MIT Press.

Fisher III, J. W., & Darrell, T. (2004). Speaker association with signal-level audiovisual fusion. *IEEE Transactions on Multimedia, 6*(3), 406–413.

Fisher III, J. W., Darrell, T., Freeman, W. T., & Viola, P. (2001). Learning joint statistical models for audio-visual fusion segregation. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems, 14*. Cambridge, MA: MIT Press.

Forsyth, D., & Ponce, J. (2003). *Computer vision: A modern approach*. Upper Saddle River, NJ: Prentice Hall.

Gatica-Perez, D., Lathoud, G., Odobez, J.-M., & McCowan, I. (2007). Audiovisual probabilistic tracking of multiple speakers in meetings. *IEEE Transactions on Audio, Speech, and Language Processing, 15*(2), 601–616.

Hall, D. L., & McMullen, S. A. H. (2004). *Mathematical techniques in multisensor data fusion*. New York: Artech House.

Hansard, M., & Horaud, R. (2007). Patterns of binocular disparity for a fixating observer. In *Proc. of Second International Symposium of Advances in Brain, Vision, and Artificial Intelligence*, (pp. 308–317). New York: Springer.

Hansard, M., & Horaud, R. (2008). Cyclopean geometry of binocular vision. *Journal of the Optical Society of America A, 25*(9), 2357–2369.

Harris, C., & Stephens, M. (1988). A combined corner and edge detector. In *Proc. of Fourth Alvey Vision Conference*, (pp. 147–151). Sheffield: University of Sheffield.

Hartley, R., & Zisserman, A. (2000). *Multiple view geometry in computer vision*. Cambridge: Cambridge University Press.

Haykin, S., & Chen, Z. (2005). The cocktail party problem. *Neural Computation, 17*, 1875–1902.

Heckmann, M., Berthommier, F., & Kroschel, K. (2002). Noise adaptive stream weighting in audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing, 11*, 1260–1273.

Hospedales, T., & Vijayakumar, S. (2008). Structure inference for Bayesian multisensory scene understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 30*(12), 2140–2157.

Jordan, M., Ghahramani, Z., Jaakkola, T., & Saul, L. (1998). An introduction to variational methods for graphical models. In M. Jordan (Eds.), *Learning in graphical models*, (pp. 105–162). Norwell, MA: Kluwer.

Joshi, R., & Sanderson, A. (1999). *Multisensor fusion: A minimal representation framework*. Singapore: World Scientific.

Keribin, C. (2000). Consistent estimation of the order of mixture models. *Sankhya: the Indian Journal of Statistics, Series A, 62*(1), 49–66.

King, A. J. (2004). The superior colliculus. *Current Biology, 14*, 335–338.

King, A. J. (2005). Multisensory integration: strategies for synchronization. *Current Biology, 15*, 339–341.

Kushal, A., Rahurkar, M., Fei-Fei, L., Ponce, J., & Huang, T. (2006). Audio-visual speaker localization using graphical models. In *Proc. of the Eighteenth*

International Conference on Pattern Recognition, (pp. 291–294). Piscataway, NJ: IEEE.

Laptev, I. (2005). On space-time interest points. *Int. J. Comp. Vis., 64*(2–3), 107–123.

Majumder, S., Scheding, S., & Durrant-Whyte, H. (2001). Multisensor data fusion for underwater navigation. *Robotics and Autonomous Systems, 35*, 97–108.

McLachlan, G., & Krishnan, T. (1996). *The EM algorithm and extensions*. Hoboken, NJ: Wiley.

McLachlan, G. J., & Peel, D. (2000). *Finite mixture models*. Hoboken, NJ: Wiley.

Mitchell, H. (2007). *Multi-sensor data fusion*. New York: Springer.

Naus, H., & van Wijk, C. (2004). Simultaneous localization of multiple emitters. *IEE Proceedings Radar Sonar and Navigation, 151*, 65–70.

Nefian, A. V., Liang, L., Pi, X., Liu, X., & Murphy, K. (2002). Dynamic Bayesian networks for audio-visual speech recognition. *EURASIP Journal of Applied Signal Processing, 2002*(1), 1274–1288.

Perez, P., Vermaak, J., & Blake, A. (2004). Data fusion for visual tracking with particles. *Proceedings of IEEE, 92*(3), 495–513.

Polyak, B. (1987). *Introduction to optimization*. New York: Optimization Software.

Pouget, A., Deneve, S., & Duhamel, J.-R. (2002). A computational perspective on the neural basis of multisensory spatial representations. *Nature Review Neuroscience, 3*, 741–747.

Quinn, B., McLachlan, G., & Hjort, N.L. (1987). A note on the Aitkin-Rubin approach to hypothesis testing in mixture models. *Journal of the Royal Statistical Society. Series B (Methodological), 49*(3), 311–314.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6*, 461–464.

Shao, X., & Barker, J. (2008). Stream weight estimation for multistream audio-visual speech recognition in a multispeaker environment. *Speech Communication, 50*(4), 337–353.

Smith, D., & Singh, S. (2006). Approaches to multisensor data fusion in target tracking: A survey. *IEEE Transactions on Knowledge and Data Engineering, 18*, 1696–1710.

van Kampen, N. (2007). *Stochastic processes in physics and chemistry* (3rd ed.). Dordrecht: North-Holland.

Wang, D., & Brown, G. J., (Eds.). (2006). *Computational auditory scene analysis: Principles, algorithms, and applications*. Hoboken, NJ: Wiley–IEEE Press.

Zhigljavsky, A. (1991). *Theory of global random search*. Norwell, MA: Kluwer.

Zhigljavsky, A., & Žilinskas, A. (2008). *Stochastic global optimization*. New York: Springer.

Received October 30, 2009; accepted July 2, 2010.