A belief-theoretical approach to example-based pose estimation

Wenjuan Gong and Fabio Cuzzolin

***Abstract*—In example-based human pose estimation, the configuration of an evolving object is sought given visual evidence, having to rely uniquely on a set of sample images. We assume here that, at each time instant of a training session, a number of feature measurements is extracted from the available images, while ground truth is provided in the form of the true object pose. In this scenario, a sensible approach consists in learning maps from features to poses, using the information provided by the training set. In particular, multi-valued mappings linking feature values to set of training poses can be constructed. To this purpose we propose a Belief Modeling Regression (BMR) approach in which a probability measure on any individual feature space maps to a convex set of probabilities on the set of training poses, in a form of a belief function. Given a test image, its feature measurements translate into a collection of belief functions on the set of training poses which, when combined, yield there an entire family of probability distributions. From the latter either a single central pose estimate or a set of extremal ones can be computed, together with a measure of how reliable the estimate is. Contrarily to other competing models, in BMR the sparsity of the training samples can be taken into account to model the level of uncertainty associated with these estimates. We illustrate BMR's performance in an application to human pose recovery, showing how it outperforms our implementation of both Relevant Vector Machine and Gaussian Process Regression. Finally, we discuss motivation and advantages of the proposed approach with respect to its most direct competitors.**

***Index Terms*—Example-based pose estimation, feature-pose maps, theory of evidence, belief functions.**

I. Introduction

Pose estimation is a well studied problem in computer vision. Given an image sequence capturing the motion and evolution of an object of interest, the problem consists in estimating the position and orientation of the object at each time instant, along with its internal configuration or *pose*. Such estimation is typically based on two pillars: the extraction of salient measurements or *features* from the available images and, when present, a *model* of the structure and kinematics of the moving body. Pose estimation is, among others, a fundamental ingredient of motion capture, i.e., the accurate reconstruction of a person's motion, for instance for animation purposes or the medical analysis of posture and gait.

a) State of the art: Current methodologies for pose estimation can roughly be classified into *model-based*, *learning-based* and *example-based* approaches. In model-based methods [1], [2] human poses are represented by explicit body model parameters. Pose recovery is typically achieved via optimisation, whose aim is to match the pose variables of a forward rendered model with the extracted features. Initialization is often difficult, and the pose optimization process can be subject to local minima [3]. In contrast, learning-based approaches [14], [4], [5], [6] rest on the fact that typical (human) motions involve a far smaller set of poses than the kinematically possible ones, and learn a model that directly recovers pose estimates from observable image quantities. Such methods [7], [8], [9], [10] are generally faster, due to the lower dimensionality of the models employed, and often provide a better predictive performance whenever training and testing data are captured under similar conditions. This class of methods, however, requires heavy training to generate good quality predictions, and the resulting model may lack generalization power.

Example-based methods, which explicitly store a set of training examples whose 3D poses are known, estimate pose by searching for training image(s) similar to the given input image and, if required, by interpolating their poses [5], [11]. They can then be used to automatically initialize model-based methods, as in the monitoring of an automobile driver's head movements provided in [10]. No prior analytic structure of the pose space is incorporated in the estimation process, although the training data itself does amount to a rough approximation of the configuration space.

In this class of techniques, vectors of feature measurements (such as moments of silhouette images [12], edge direction histograms [13], distributions of shape contexts [14] or Harr-like features [15]) are first extracted from each individual image. The integration of multiple cues is exploited to increase both accuracy and robustness of the estimation [37], [38], [39], [40]. Then, the likely pose of the object is predicted by feeding the resulting feature vector to a learnt map from the feature space to the pose space, for instance using an efficient searching scheme such as random forests. Note that this map is (in general) one-to-many: more than one object configuration can generate the same feature observation(s), because of occlusions, self-occlusions and the ambiguities induced by the perspective image projection model itself.

Since only limited information is provided to the system in the training session, only an approximation of the true feature-pose mapping can be learned. In [12], for instance, an inverse mapping between image silhouette moments and 2D joint configurations is learned by fitting a Gaussian mixture to 2D joint configurations via the EM algorithm. The accuracy of the estimation depends on the forcibly limited size and

W. Gong is with the College of Computer and Communication Engineering, China University of Petroleum, Qingdao, China. Email: wenjuangong@upc.edu.cn.

F. Cuzzolin is with the Department of Computing and Communication Technologies, Oxford Brookes University, Oxford, UK. Email: fabio.cuzzolin@brookes.ac.uk.

distribution of the available examples, which are expensive and time-consuming to collect. This has suggested the adoption of a more activity-based setting to constrain the search space of poses. In [17], a number of exemplar 2D views of the human body is stored; the locations of the body joints are manually marked and labeled. The input image is then matched via shape context matching to each stored view, and the locations of the body joints in the matched exemplar view are transferred to the test image. Local Weighted Regression [5], BoostMap [11] and Bayesian Mixture of Experts [6] have also been applied. In example-based approaches queries can be potentially computationally expensive, and need to be performed quickly and accurately [5], [11]. In addition, these methods often have problems when working in high dimensional configuration s paces, a s i t i s d ifficult to collect enough examples to densely cover them.

b) Scenario: We consider here a situation in which:

- a set of training images of various poses of an object is available;
- the object's configuration in each image can be described by a vector $q \in \mathcal{Q}$ in a pose space $\mathcal{Q}$ which is a subset of $\mathbb{R}^D$, with D the dimensionality of the pose vector;
- a source of ground truth exists which provides for each training image I_k the pose configuration q_k of the object portrayed in the image;
- the location of the object within each training image is known, in the form of a minimal bounding box.

In the training session, the object explores its range of possible configurations, and a set of poses is collected to form a finite approximation $\tilde{\mathcal{Q}}$ of the true pose space $\mathcal{Q}$:

$$\tilde{\mathcal{Q}} \doteq \Big\{q_k, k = 1, ..., T\Big\}, \tag{1}$$

where T is the duration of the training session. At the same time N distinct features are extracted from the available image(s), within the available bounding box:

$$\tilde{\mathcal{Y}} \doteq \Big\{y_i(k), k = 1, ..., T\Big\}, \quad i = 1, ..., N. \tag{2}$$

In order to collect $\tilde{\mathcal{Q}}$ we need a source of ground truth poses at each time instant k of the training session. One option is to use a motion capture system, as in [12]. After applying a number of reflective markers in fixed positions of the moving object, the system is able to supply the 3D locations of the markers. Since we assume we do not know the object's actual pose space $\mathcal{Q}$, it is reasonable to use the collection of 3D marker locations as pose vector.

In the testing stage, a supervised localization algorithm (trained using the annotated image evidence and bounding box pairs, e.g. [27]) is employed to locate the object within each test image, so that image features are only extracted from within the resulting bounding box. Such features are exploited to estimate the object's configuration, with attached a measure of how reliable this estimate is.

c) Contributions: In this paper we propose a regression framework for the example-based pose estimation problem formulated above, based on the theory of *belief functions* [19], [20], [21]. Belief functions are non-additive measures which admit a number of interpretations: i) as random sets, i.e., probability distributions on the power set of all subsets; ii) as convex sets of conventional probability distributions; iii) as measures induced by the application of a multi-valued map to a probability measure [20], [22]. The most relevant interpretation for the proposed BMR model is the second one, for a belief function on the pose space is equivalent to a *set of linear constraints on the actual conditional pose distribution (given the features)*. Our Belief Modeling Regression (BMR) framework uses the finite amount of evidence provided in the training session to map any new feature value to a belief function on the set of training poses $\tilde{\mathcal{Q}}$, via a learnt *refining* map. This determines a convex set of distributions on $\tilde{\mathcal{Q}}$, which in turn generates an interval of pose estimates.

Multiple features are necessary to obtain a decent estimation accuracy. All single-feature refinings are collected in an *evidential model* of the object, and the information carried by individual features is fused in the belief theory framework [20], [23]. This allows even limited resolutions for the individual features to translate into relatively high estimation accuracy (in a similar way to tree-based classifiers [18] or boosting approaches, in which weak features are combined to form a strong classifier). The size of the resulting convex set of probabilities (*credal set*) reflects the amount of training information available: the larger and more densely distributed within the pose space the training set is, the narrower the resulting set of probabilities. A separate pose estimate can be computed for each vertex of the credal set, in a robust statistical fashion. In alternative, a central estimate can be extracted together with a measure of the associated uncertainty, as a function of the size of the estimated set of probabilities.

As we show in the last part of the paper, an evidential model provides a constraint on the family of admissible feature-to-pose maps, in terms of smooth upper and lower bounds. All mappings (even discontinuous, or 1-many) within those smooth bounds are admissible under the model. The width of this space of mappings reflects the uncertainty induced by the size and distribution of the available training set.

d) Paper outline: The paper is structured as follows. Firstly, the theory of belief functions is introduced in Section II, with a focus on their combination operators and the handling of evidence defined on distinct but related domains. In Section III the different elements of our Belief Modeling Regression approach are described in detail. Evidential model training is described in Section III-A. In Section III-B Dirichlet belief functions are proposed to model the uncertainty due to the scarcity of the training data. From the belief function resulting from their combination, either a pointwise estimate or a set of extremal estimates of the pose can be extracted. In Section III-C model assessment criteria are discussed, together with an analysis of the computational complexity of training and estimation algorithms. Section IV illustrates the performance of Belief Modeling Regression in a human pose recovery setting, showing how BMR outperforms our implementation of both Relevant Vector Machine and Gaussian Process Regression. Finally, Section V discusses motivation and advantages of the proposed approach in comparison with other competitors. Section VI concludes the paper.

II. Belief calculus

Suppose that we have a probability measure P for a question Q_1 whose possible outcomes form a set Ω, and that Q_1 is related to another question Q_2, whose outcomes are in a different set Θ, via a one-to-many map $\rho : \Omega \to 2^\Theta$ (a *multi-valued mapping*) Outcomes $\omega \in \Omega$ of Q_1 are mapped to *sets* of outcomes $B = \rho(\omega) \subset \Theta$ of Q_2. The probability value of $\omega \in \Omega$ thus supports the proposition that the true answer to Q_2 is in a subset A of Θ, whenever $\rho(\omega) \subseteq A$. As A. Dempster showed [20], the result of mapping a probability distribution via a multi-valued map is an object more general than a probability measure: a *belief measure* [22].

The *degree of belief* $b(A)$ with which $A \subseteq \Theta$ contains the answer to Q_2 is then the total probability of all the supporting outcomes ω of Q_1: $b(A) = P(\{\omega \in \Omega | \rho(\omega) \subseteq A\})$. A multi-valued mapping makes a probability distribution P on Ω into a distribution $m : 2^\Theta \to [0, 1]$, s.t. $\sum_{A \subset \Theta} m(A) = 1$, on the power set $2^\Theta = \{A \subseteq \Theta\}$ of the codomain Θ, called *basic probability assignment* (b.p.a.) [19]. The *belief function* (b.f.) $b : 2^\Theta \to [0, 1]$ induced by m has degree of belief on A

$$b(A) = \sum_{B \subseteq A} m(B), \tag{3}$$

i.e., the sum of the masses of all its subsets. The domain Θ of a belief function is called *frame of discernment* (FOD), and the non-zero mass subsets of Θ are said *focal elements* (f.e.s).

A popular (albeit criticized) interpretation of belief functions sees each focal element A with b.p.a. $m(A)$ as the indication of the existence of a mass $m(A)$ floating inside A, which can be assigned to any of its elements $x \in A$ [30]. A probability distribution *consistent* with b can then be obtained by redistributing the mass $m(A)$ of each focal element A to its elements $x \in A$. The resulting *credal set* [31] consistent with a belief function b is $\mathcal{P}[b] \doteq \big\{p \in \mathcal{P} : p(A) \geq b(A)\ \forall A \subseteq \Theta\big\}$, i.e., the set of probability measures whose values dominate that of b on all events A. This is a polytope in the simplex of all probabilities one can define on Θ. The vertices of this polytope are all the probability distributions p^π induced by an arbitrary permutation $\pi = \{x_{\pi(1)}, ..., x_{\pi(|\Theta|)}\}$ of the elements of Θ, of the form:

$$p^\pi[b](x_{\pi(i)}) = \sum_{\substack{A \ni x_\pi(i) \\ A \not\ni x_\pi(j)\ \forall j<i}} m(A). \tag{4}$$

The latter assigns to a singleton element in position $\pi(i)$ the mass of all the focal elements A containing it, while not containing any elements before it in the permutation order.

The combination of information obtained from different sources, and represented as belief functions, is a central theme of belief calculus [23], [41].

Definition 1. *The* conjunctive combination *of two belief functions $b_1, b_2 : 2^\Theta \to [0, 1]$ is a new belief function $b_1 \textcircled{\cap} b_2$ on the same FOD Θ whose focal elements are all the possible intersections of focal elements of b_1 and b_2 respectively, and whose b.p.a. is given by:*

$$m_{b_1 \textcircled{\cap} b_2}(A) = \sum_{B \cap C = A} m_{b_1}(B)\ m_{b_2}(C). \tag{5}$$

This definition can be trivially extended to the combination of an arbitrary number of belief functions.

While it is axiomatically justifiable as the only combination rule which meets a number of rationality requirements (such as least commitment, specialization, associativity and commutativity [42]), the conjunctive combination also amounts to assuming that the sources of evidence to merge are both reliable and independent. In general, the current consensus is that different combination rules are to be employed under different assumptions [42]. However, it is difficult to decide in which situations the sources of information can indeed be considered independent. An alternative point of view, supported by Shenoy, suggests that, rather than employing a battery of combination rules whose applicability to a given problem is difficult to establish, we should adopt models which do meet the assumption of independent sources, as it happens in probability theory. We support this view here, and test the adequacy of the assumption empirically.

In belief calculus a map between two FODs Θ and Ω of the form $\rho : 2^\Theta \to 2^\Omega$, $\rho(A) = \cup_{\theta \in A} \rho(\{\theta\})$, which maps Θ to a disjoint partition of its codomain Ω ($\rho(\{\theta_1\}) \cap \rho(\{\theta_2\}) = \emptyset$ for all distinct $\theta_1, \theta_2 \in \Theta$) is called a *refining*. The frame Ω is called a *refinement* of Θ, while Θ is said a *coarsening* of Ω. A FOD is called the *minimal refinement* [19] of a collection of frames $\Theta_1, ..., \Theta_N$ if it is a refinement of each of them (their *common refinement*), and no coarsening of it is still a common refinement. The minimal refinement of $\Theta_1, ..., \Theta_N$ is denoted by $\Theta_1 \otimes ... \otimes \Theta_N$. The frames $\Theta_1, ..., \Theta_N$ are said to be *independent* [19] if $\rho_1(A_1) \cap \cdots \cap \rho_N(A_N) \neq \emptyset$, (where $\forall \emptyset \neq A_i \subseteq \Theta_i, \forall i = 1, ..., N$, and ρ_i is the refining from Θ_i to $\bigotimes_i \Theta_i$) in which case the minimal refinement is their Cartesian product: $\bigotimes_i \Theta_i = \Theta_1 \times \cdots \times \Theta_N$. A belief function b' on Ω, a refinement of Θ, is called the *vacuous extension* of a second belief function b on Θ iff the focal elements of b' are images (via ρ) of focal elements of b.

III. Belief Modeling Regression

A. Model training

Consider an image feature function y, whose values lie in a *feature space* $\mathcal{Y}$, and denote by $\rho^* : \mathcal{Y} \to 2^\mathcal{Q}$ the unknown mapping from the feature space to the collection $2^\mathcal{Q} = \{Q \subseteq \mathcal{Q}\}$ of sets Q of object poses. We seek to learn from the training data $\{\tilde{\mathcal{Q}}, \tilde{\mathcal{Y}}\}$ (Equations (1) and (2)) an approximation of the unknown feature-pose mapping ρ^*.

EM clustering [29] is applied individually to each component of the training data $\{y_i(k), k = 1, ..., T\}$, $i = 1, ..., N$ to obtain a Mixture of Gaussians (MoG)[1] with n_i components:

$$\Big\{\Gamma_i^j, j = 1, ..., n_i\Big\}, \quad \Gamma_i^j \sim \mathcal{N}(\mu_i^j, \Sigma_i^j). \tag{6}$$

The MoG (6) induces an implicit partition of the i-th feature space (the range $\mathcal{Y}_i \subset \mathbb{R}^{d_i}$ of the unknown feature function y_i :

[1]MoG models are often employed in example-based pose estimation, as their parameters can be speedily estimated via the EM algorithm [29]. For instance, in [6] several experts predictions are combined in a Gaussian mixture model. In [26], conditional distributions are assumed to be Gaussian mixtures.

$\mathcal{I} \to \mathcal{Y}_i$ on the set of all images $\mathcal{I}$, d_i being the dimensionality of the i-th feature space):

$$\Theta_i \doteq \left\{\mathcal{Y}_i^1, \cdots, \mathcal{Y}_i^{n_i}\right\}, \tag{7}$$

where $\mathcal{Y}_i^j = \left\{y \in \mathcal{Y}_i \;\middle|\; \Gamma_i^j(y) > \Gamma_i^l(y)\ \forall l \neq j\right\}$ is the region of $\mathcal{Y}_i$ in which the j-th Gaussian component dominates (Figure 1). We call (7) the i-th *approximate feature space*.

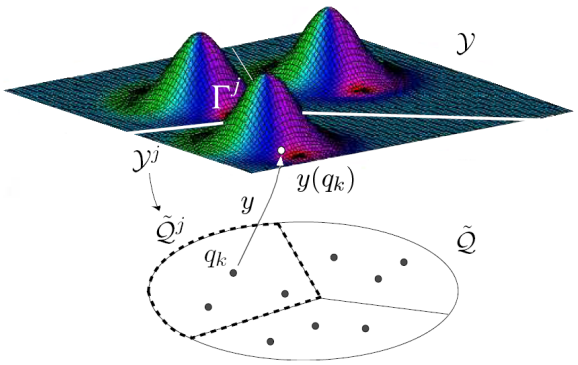

Fig. 1. A Mixture of Gaussians learned from training features defines an implicit partition (Equation (7)) of the feature space $\mathcal{Y}$. In turn, each feature region $\mathcal{Y}^j$ is in correspondence with the set $\tilde{\mathcal{Q}}^j$ of sample poses q_k whose feature value $y(q_k)$ falls inside $\mathcal{Y}^j$.

As training feature vectors are labelled by the true poses provided by the source of ground truth (cfr. the Introduction), each element $\mathcal{Y}_i^j$ of the approximate feature space is associated with the set of training poses whose i-th feature value falls in $\mathcal{Y}_i^j$ (see Figure 1 again):

$$\rho_i : \mathcal{Y}_i^j \mapsto \tilde{\mathcal{Q}}_i^j \doteq \left\{q_k \in \tilde{\mathcal{Q}} : y_i(k) \in \mathcal{Y}_i^j\right\}. \tag{8}$$

Applying EM clustering separately to each training feature sequence in Equation (2) thus yields both N approximate feature spaces $\Theta_i = \{\mathcal{Y}_i^1, \cdots, \mathcal{Y}_i^{n_i}\}$, $i = 1, ..., N$, and N maps as in Equation (8), from each approximate feature space to the set of training poses $\tilde{\mathcal{Q}}$.

Clearly, the maps (8) are multi-valued mappings linking the question Q_1 "to which Gaussian component of the MoG (6) does the new feature value y belong" to the question Q_2 "what is the object pose whose observed feature value is y".
The number of clusters n_i is set here to a fixed value for each feature space.

Training algorithm. In the training stage the object moves in front of the camera(s), exploring its configuration space, while a sequence of training poses $\tilde{\mathcal{Q}} = \{q_k, k = 1, ..., T\}$ is provided by a source of ground truth. Training images are annotated by a bounding box indicating the object's location within each image. At the same time:

1) for each time instant k, a number of feature values are computed from the region of interest in each training image: $\{y_i(k), k = 1, ..., T\}$, $i = 1, ..., N$;
2) EM clustering is applied to each feature sequence $\{y_i(k), k = 1, ..., T\}$ (after setting the number of clusters n_i), yielding:
   a) N approximate feature spaces $\Theta_i = \{\mathcal{Y}_i^j, j = 1, ..., n_i\}$, i.e., the implicit partitions of the feature ranges $\mathcal{Y}_i$ associated with the EM clusters;

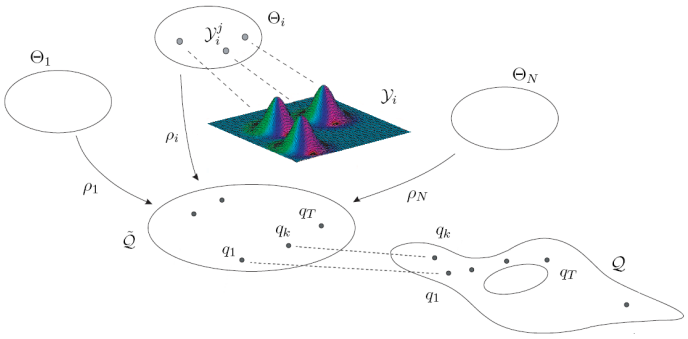

Fig. 2. Evidential model. The EM clustering of each feature set collected in the training stage yields an approximate feature space $\Theta_i = \{\mathcal{Y}_i^j, j = 1, ..., n_i\}$. Refining maps ρ_i between each approximate feature space and $\tilde{\mathcal{Q}} = \{q_1, ..., q_T\}$ (the training approximation of the unknown pose space $\mathcal{Q}$) are learned, allowing the fusion on $\tilde{\mathcal{Q}}$ of the evidence gathered on $\Theta_1, ..., \Theta_N$.

   b) N maps ρ_i (described in Equation (8)) mapping EM feature clusters to sets of sample training poses in the approximate pose space $\tilde{\mathcal{Q}}$.

As the learned application (Equation (8)) maps approximate feature spaces to disjoint partitions of the approximate pose space $\tilde{\mathcal{Q}}$, the latter is a common refinement (see Section II) of the collection of approximate feature spaces $\Theta_1, ..., \Theta_N$. The collection of FODs $\tilde{\mathcal{Q}}, \Theta_1, ..., \Theta_N$ along with their refinings $\rho_1, ..., \rho_N$ is characteristic of both the object to track, the chosen feature functions y_i, and the actual training data: we call it the *evidential model* (Figure 2) of the object.

B. Estimation

Once an evidential model has been learned from the available training set, it can be used to estimate the pose of the object given new visual evidence.

1) Belief functions induced by test feature values: when one or more new test images are acquired, new visual features $y_1, ..., y_N$ are extracted. Given the mixture in Equation (6), each new feature value y_i is associated with the vector of soft assignments to each mixture component:

$$y_i \mapsto \left[\Gamma_i^1(y_i), \Gamma_i^2(y_i), \cdots, \Gamma_i^{n_i}(y_i)\right]^T. \tag{9}$$

When normalized, the latter yields a probability distribution p_i on the approximate feature space Θ_i. Since each ρ_i is a multi-valued mapping, it follows (Section II) that p_i induces a belief function on the (approximate) pose range $\tilde{\mathcal{Q}}$. As a result, the test features values $y_1, ..., y_N$ are mapped to a collection of belief functions $b_1, ..., b_N$ on the set of training poses $\tilde{\mathcal{Q}}$.

2) Dirichlet belief function modeling: in fact, belief functions allow us to take into account the scarcity of the training samples by assigning a mass $m(\Theta_i)$ to the whole approximate feature space, prior to applying the refining ρ_i. This encodes the fact that training the model on a larger set of examples would alter the shape of the MoG approximation of $\mathcal{Y}_i$ in unpredictable ways. Namely, we map the soft assignment (9) to a *Dirichlet belief function* [43], with b.p.a.:

$$m_i : 2^{\Theta_i} \to [0, 1], m_i(\mathcal{Y}_i^j) = \frac{\Gamma_i^j(y_i)}{\sum_k \Gamma_i^k(y_i)}\big(1 - m_i(\Theta_i)\big). \tag{10}$$

The assignment in Equation (10) discounts the probability distribution obtained by simply normalizing the likelihoods (9) by assigning the mass $m_i(\Theta_i)$ to the entire FOD Θ_i. A reasonable choice for the mass function is: $m_i(\Theta_i) = \frac{1}{n_i}$, so that when $n_i \to \infty$ the discount factor tends to zero, and the approximate feature space converges to the Mixture of Gaussian representation of the actual feature space. In addition, as n_i cannot be greater than the number T of training samples, such a discounting factor also considers the limited size of the training set. If we set $m_i(\Theta_i) = 0$ the result is a probability distribution, called a *Bayesian* belief function [19].

3) Belief estimate: the Dirichlet belief functions $\{b_i : 2^{\Theta_i} \to [0,1], i = 1, ..., N\}$ inferred from the test feature values $y_1, ..., y_N$ via Equation (10) are then mapped to belief functions $\{b'_i : 2^{\tilde{\mathcal{Q}}} \to [0,1], i = 1, ..., N\}$ on the approximate pose space $\tilde{\mathcal{Q}}$ by vacuous extension: $\forall A \subset \tilde{\mathcal{Q}}$

$$m'_i(A) = \begin{cases} m_i(A_i) & \exists A_i \subset \Theta_i \; s.t. \; A = \rho_i(A_i); \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The resulting b.f.s on $\tilde{\mathcal{Q}}$ are combined by conjunctive combination, as in Equation (5). The result is a belief function $\hat{b} = b'_1 \textcircled{\cap} \cdots \textcircled{\cap} b'_N$ on $\tilde{\mathcal{Q}}$, which we call the *belief estimate* of the object's pose. As explained in Section II, the belief estimate is associated with an entire convex set of probabilities on the approximate pose space.

Example. Suppose that the training set of poses contains just three samples: $\tilde{\mathcal{Q}} = \{q_1, q_2, q_3\}$, and that the evidence combination process produces a belief estimate $\hat{b}$ with b.p.a.:

$$\hat{m}(\{q_1, q_2\}) = \frac{1}{3}, \hat{m}(\{q_3\}) = \frac{1}{6}, \hat{m}(\{q_1, q_2, q_3\}) = \frac{1}{2}. \quad (12)$$

According to Equation (4), the vertices of $\mathcal{P}[\hat{b}]$ are those probabilities which are generated by reassigning the mass of each focal element to any one of its singletons: there are $\prod_k |A_k|$ possible choices, where $\{A_k\}$ is the list of focal elements of $\hat{b}$. As the belief function (12) has 3 f.e.s of cardinality $2, 1$ and 3, respectively, the corresponding credal set $\mathcal{P}[\hat{b}]$ is the convex closure of $1 \cdot 2 \cdot 3 = 6$ probability distributions, namely:

$$p_1 = \begin{bmatrix} \frac{5}{6} & 0 & \frac{1}{6} \end{bmatrix}, \quad p_2 = \begin{bmatrix} \frac{1}{2} & \frac{1}{3} & \frac{1}{6} \end{bmatrix}, \quad p_3 = \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & \frac{1}{6} \end{bmatrix},$$
$$p_4 = \begin{bmatrix} 0 & \frac{5}{6} & \frac{1}{6} \end{bmatrix}, \quad p_5 = \begin{bmatrix} 0 & \frac{1}{3} & \frac{2}{3} \end{bmatrix}, \quad p_6 = \begin{bmatrix} \frac{1}{3} & 0 & \frac{2}{3} \end{bmatrix}.$$

4) Computing expected pose estimates: point-wise information on the object's pose can be extracted from a belief estimate $\hat{b}$ in two different ways.

a) Extracting a set of extremal point-wise estimates: each of the vertices (4) of the credal set associated with $\hat{b}$ is a probability distribution on the approximate pose space $\tilde{\mathcal{Q}}$. We can compute the associated expected pose as:

$$\hat{q} = \sum_{k=1}^{T} p(q_k) q_k. \quad (13)$$

The set of such extremal estimates describes therefore an entire polytope of expected pose values in the object's pose space $\mathcal{Q}$. In the above example, the expected poses for the vertices p_1, p_4, p_5, p_6 of $\mathcal{P}[b]$ are: $\hat{q}[p_1] = \frac{5}{6}q_1 + \frac{1}{6}q_3$, $\hat{q}[p_4] = \frac{5}{6}q_2 + \frac{1}{6}q_3$, $\hat{q}[p_5] = \frac{1}{3}q_2 + \frac{2}{3}q_3$, $\hat{q}[p_1] = \frac{1}{3}q_1 + \frac{2}{3}q_3$.

b) Extracting a single point-wise estimate: alternatively, we can approximate $\hat{b}$ with a probability $\hat{p}$ on $\tilde{\mathcal{Q}}$, before computing its mean value as above. The approach is widely supported by the belief function literature.
In particular, Smets' *pignistic function* [44]

$$BetP[b](x) = \sum_{A \supseteq x} \frac{m_b(A)}{|A|}, \forall x \in \Theta, \quad (14)$$

is the barycenter of the credal set $\mathcal{P}[b]$ associated with b. Although other transforms have been proposed [45], [48], [46], [47], empirically their performances in the human pose estimation experiments presented here have been proven comparable.

Pose estimation. Given an evidential model of the moving body with N feature spaces, and given at time t one or more test images, possibly coming from different cameras:

1) the object detector learned during training is applied to the test image(s), returning for each of them a bounding box roughly containing the object of interest;
2) the N feature values are extracted from the resulting bounding boxes, as during training;
3) the likelihoods $\{\Gamma_i^j(y_i(t)), j = 1, ..., n_i\}$ of each feature value $y_i(t)$ with respect to the learned Mixture of Gaussian distribution on $\mathcal{Y}_i$ are computed, as in (9);
4) for each feature $i = 1, ..., N$, a separate Dirichlet belief function $b_i(t) : 2^{\Theta_i} \to [0,1]$ on the appropriate feature space Θ_i is built from the set of likelihoods $\{\Gamma_i^j(y_i(t)), j = 1, ..., n_i\}$ as described in Section III-B2;
5) all the resulting b.f.s $\{b_i(t) : 2^{\Theta_i} \to [0,1], i = 1, ..., N\}$ are projected onto $\tilde{\mathcal{Q}}$ by vacuous extension (11), yielding a set of belief functions $\{b'_i : 2^{\tilde{\mathcal{Q}}} \to [0,1], i = 1, ..., N\}$;
6) their conjunctive combination $\hat{b}(t) \doteq b'_1(t) \textcircled{\cap} \cdots \textcircled{\cap} b'_N(t)$ is computed (Equation (5));
7) the object pose estimate(s) are computed:
   a) either the pignistic transform (14) is applied to $\hat{b}(t)$, yielding a probability distribution on $\tilde{\mathcal{Q}}$ from which an expected pose estimate $\hat{q}(t)$ is obtained by Equation (13);
   b) or, the vertices of the convex set of probabilities $\mathcal{P}[\hat{b}(t)]$ associated with the belief estimate $\hat{b}(t)$ are computed as in (4), and a mean pose estimate is obtained for each one of them as in (13).

C. Assessing evidential models

1) Robustness: as a consequence of computing the belief estimate by conjunctive combination, a non-zero mass may be assigned to the empty set. This happens when the focal elements of the belief functions to merge $\{b'_i(t)\}$ are disjoint. In the worst case scenario all the mass may be assigned to $\emptyset$, and no estimation is possible (*conflict*). Conflict may arise due to either the incorrect localization of the object of interest (due to limitations of the trained detector), so that background features conflicting with the foreground information are also extracted, or to the presence of occlusions, which generates conflict for similar reasons.

However, when adopting Dirichlet belief functions for inference (Section III-B2) this never materializes, for each individual b.f. has Θ_i as a focal element and some mass

is always assigned to non-empty focal elements, ensuring robustness to localization errors and occlusions.

2) Computational cost: at training time EM's computational cost is easy to assess, as the algorithm usually takes a constant number $c \sim 5 - 10$ of steps to converge. At each step the whole observation sequence of length T is processed, yielding a time complexity of $O(cNnT)$ (where N is the number of features, n the average number of EM clusters, T the number of training samples). This is acceptable for real-world applications, since this only needs to be done once in the training session. In the experiments of Section IV the whole training procedure in Matlab required 17.5 seconds for each run of EM on an outdated Athlon 2.2 GHz CPU with $N = 5$ features, $n_i = n = 5$ states per feature space, and $T = 1726$.

At test time, although the conjunctive combination in Equation (5) is exponential in complexity if naively implemented, fast implementations of $\textcircled{\cap}$ exist [50]. Numerous Monte-Carlo approximation schemes have been proposed [51]. Furthermore, Dirichlet b.f.s in Equation (10) have $n_i + 1$ non-zero focal elements, reducing the computational complexity of their pairwise combination from $O(2^{2n})$ (the mass multiplication of all the 2^n possible focal elements of the first b elief function by those of the second b.f.) to $O(n^2)$.

3) Self-consistency: an evidential model is *self-consistent* if it produces the correct ground truth pose values when presented with the training feature data $\{y_i(k), i = 1, ..., N\}$. Suppose that $y_1, ..., y_N$, the observed feature vector components, are such that: $y_1 \in \mathcal{Y}_1^{j_1}, \cdots, y_N \in \mathcal{Y}_N^{j_N}$. For $i = 1, ..., n$ the object's pose must lie within the subset $\rho_i(\mathcal{Y}_i^{j_i})$ of the training set $\tilde{\mathcal{Q}}$. Thus, the pose estimate must fall within:

$$\rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N}) \subset \tilde{\mathcal{Q}}. \tag{15}$$

Consequently, sample object poses in the same intersection of the above form are indistinguishable under the given evidential model. The collection of all the non-empty intersections of the form (15) is in fact the minimal refinement $\Theta_1 \otimes \cdots \otimes \Theta_N$ (Section II) of the FODs $\Theta_1, ..., \Theta_N$. It follows that:

Theorem 1. *Any two poses of the training set can be distinguished under the evidential model iff $\tilde{\mathcal{Q}}$ is the minimal refinement of $\Theta_1, ..., \Theta_N$.*

Proof. $\Rightarrow$: if any two sample poses can be distinguished under the model, i.e., for all k, k' $q_{k'} \notin \rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N}) \ni q_k$, it follows that each intersection in Equation (15) cannot contain more than one sample pose, otherwise there would exist a pair violating the above hypothesis (the intersection can, however, be empty). Furthermore, each sample pose q_k falls within such an intersection, the one associated with the feature regions $\mathcal{Y}_1^{j_1}, \cdots, \mathcal{Y}_N^{j_N}$ s.t. $y_1(q_k) \in \mathcal{Y}_1^{j_1}$, ..., $y_N(q_k) \in \mathcal{Y}_1^{j_N}$. Hence, all the elements of the minimal refinement of Θ_1,...,Θ_N are individual sample poses, $\tilde{\mathcal{Q}} = \Theta_1 \otimes \cdots \otimes \Theta_N$.
$\Leftarrow$: if $\tilde{\mathcal{Q}}$ is the minimal refinement of Θ_1,...,Θ_N then for all $q_k \in \tilde{\mathcal{Q}}$ we have that $\{q_k\} = \rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N})$ holds for some unique selection of feature regions $\mathcal{Y}_1^{j_1}, \cdots, \mathcal{Y}_N^{j_N}$, distinct for each training pose. Any two different sample poses thus belong to different intersections of the form (15), i.e., they can be distinguished under the model. □

It is then desirable to select, at training time, a collection of features which brings the minimal refinement $\Theta_1 \otimes \cdots \otimes \Theta_N$ as close as possible to $\tilde{\mathcal{Q}}$. The self-consistency of the model can be measured by the ratio between the cardinality of the minimal refinement of $\Theta_1, ..., \Theta_N$, and that of the approximate pose space $\tilde{\mathcal{Q}}$: $\frac{1}{T} \leq \frac{|\bigotimes_i \Theta_i|}{|\tilde{\mathcal{Q}}|} \leq 1$.

4) Model granularity and accuracy: the granularity $\{n_i, i = 1, ..., N\}$ and dependence of the feature spaces forming of an evidential model obviously affect the accuracy of the estimation process. Indeed, if the approximate feature spaces Θ_i are independent, for each combination of feature regions $\mathcal{Y}_1^{j_1}, \cdots, \mathcal{Y}_N^{j_N}$, there exists a unique sample pose q_k characterized by feature values in those regions: $\{q_k\} = \rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N})$. In this case different cues carry complementary information about the object's pose. When the opposite holds, instead, fewer than N feature values may be enough to resolve training poses, while in general each combination of feature values will yield a whole set of them.

Assuming the model is self-consistent ($|\tilde{\mathcal{Q}}| = |\otimes_i \Theta_i|$), the independence of its (approximate) feature spaces implies that $|\tilde{\mathcal{Q}}| = |\otimes_i \Theta_i| = \prod_i |\Theta_i|$), i.e.: $T = |\tilde{\mathcal{Q}}| \sim n_1 \times ... \times n_N$. Given a realistic sampling of the parameter space with $T = 20000$ examples, the use of $N = 9$ complementary features allows to require no more than $\sqrt[9]{20000} \sim 3$ MoG components for each feature space. This shows the advantage of encoding feature-pose maps separately: as long as the chosen features are independent (as described above), a relatively coarse MoG representation for each feature space permits a good accuracy of the pose estimate[2].

5) Approximate and actual pose space: finally, let us discuss the conditions under which the training set of poses $\tilde{\mathcal{Q}}$ is a proper approximation of the unknown parameter space $\mathcal{Q}$ (see Figure 2). Ideally, the set $\tilde{\mathcal{Q}}$ of training poses should be dense in $\mathcal{Q}$: $\forall q \in \mathcal{Q}$ there ought to exist a sample q_k such that $\|q - q_k\| < \epsilon$ for some ϵ small enough. Clearly, such a condition is hard to impose. The distribution of the training poses within $\mathcal{Q}$ has nevertheless a number of consequences on the estimation process.

Firstly, as the true pose space $\mathcal{Q}$ is typically non-linear, while the pose estimate is a linear combination of sample poses (see Section III-B3), the pointwise estimate may be non-admissible (fall outside $\mathcal{Q}$). This can be fixed by trying to make the feature spaces independent, as in that case every sample pose q_k is characterized by a different combination of feature clusters: $\{q_k\} = \rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N})$. Consequently, any set of test feature values $y_1 \in \mathcal{Y}_1^{j_1}, ..., y_N \in \mathcal{Y}_N^{j_N}$ will generate a belief estimate in which a single sample pose q_k is dominant: its credal set (Section III-B4a) is of limited extension around a single sample pose, and the risk of non-admissibility due to linear extrapolation of admissible poses is reduced.

Secondly, there may exist regions of $\mathcal{Q}$ characterized by combinations of approximate feature values not in the current evidential model – namely, object poses $q \in \mathcal{Q}$ such that:

$$\forall \mathcal{Y}_1^{j_1} \in \Theta_1, ..., \mathcal{Y}_N^{j_N} \in \Theta_N \quad q \notin \rho_1(\mathcal{Y}_1^{j_1}) \cap \cdots \cap \rho_N(\mathcal{Y}_N^{j_N}).$$

[2]Compare this point to what proposed in [15] or [18], where trees of classifiers are used for face pose estimation.

This would generate high levels of conflict $m(\emptyset)$ in the conjunctive feature combination (5), flagging the inadequacy of the model. In case new ground truth is provided, the model can be updated by adding the poses causing the problem.

IV. Experimental results

We tested our Belief Modeling Regression approach in a rather challenging setup, involving the pose estimation of human arms and legs from two well separated views. While the bottom line of BMR is doing the best we can with the available examples, regardless the dimensionality of the pose space, and without having at our disposal prior information on the object at hand, we ran test on articulated objects (one arm and a pair of legs) with a reasonably limited number of degrees of freedom to show what can be achieved in such a case. The results show that this technique outperforms competitors such as Relevant Vector Machines and Gaussian Process Regression.

A. Setup: two human pose estimation experiments

To collect the necessary ground truth we used a marker-based motion capture system [26], [24] built by E-motion, a Milan firm. The number of markers used was 3 for the arm (yielding a pose space $\mathcal{Q} \subset \mathbb{R}^9$, using as pose components the 3D coordinates of the marker), and 6 for the pair of legs ($\mathcal{Q} \subset \mathbb{R}^{18}$). The person was filmed by two uncalibrated DV cameras (Figure 3). In the training stage of the first experiment we asked the subject to make his arm follow a trajectory (approximately) covering the pose space of the arm itself, keeping his wrist locked and standing on a fixed spot on the floor to limit the intrinsic dimensionality of the pose space (resulting in 2 d.o.f.s for the shoulder and 3 for the elbow). In the second experiment we tracked the subject's legs, assuming that the person was walking normally on the floor, and collected a training set by sampling a random walk on a small section of the floor. This is similar to what is done in other works, where the set of examples are taken for a specific family of motions/trajectories, normally associated with action categories such as the walking gait. The length of the training sequence was 1726 frames for the arm experiment and 1952 frames for the legs test.

While the number of degrees of freedom was limited by constraining the articulated object (person) to perform motions of a specific class (walking versus brandishing an arm), the tests are sufficiently complex to allow us to illustrate the traits of the BMR approach to pose estimation. In addition, in both experiments the background was highly non-static, with people coming in and out the scene and flickering monitors; the object of interest would self-occlude itself a number of times on at least one of the two views (e.g. when one leg would occlude the other when seen from the left camera), making the experimental setup quite realistic.

Under the assumptions listed in the Introduction, in the training stage the images need to be annotated by a bounding box, to provide a rough localization of the unknown object. To simulate this annotation process, and isolate the performance of the proposed example-based estimation approach from that

Fig. 3. Two human body-part pose estimation experiments. Left: training images of a person standing still and brandishing his right arm. Right: training images of the person walking inside a rectangle on the floor.

of the object detector employed, in these tests we used color-based segmentation to separate the object of interest from the non-static background, implemented via a colorimetric analysis of the body of interest (Figure 4-middle). Pixels were clustered in the RGB space; the cluster associated with the yellow sweater (in the arm experiment) or the black pants (legs one) was detected, and pixels in that cluster assigned to the foreground; the minimal bounding box containing the silhouette of the segmented foreground pixels was finally detected. Note that this is just a way to automatically generate, rather than manually construct, the bounding box annotation required in the assumptions of the initial scenario.

B. Feature extraction and modeling

For these tests we decided to build an extremely simple feature vector for each image directly from the bounding box, as the collection $\max(row)$, $\min(row)$, $\max(col)$, $\min(col)$ of the row and column indexes delimiting it (Figure 4). As two views were available at all times, at each time instant two feature vectors of dimension 4 were computed.

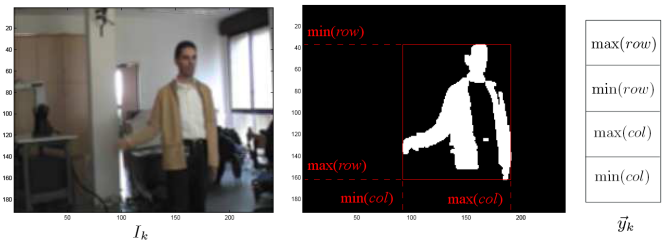

Fig. 4. Feature extraction process. Left: a training image I_k in the arm experiment. Middle: the object of interest is color-segmented and the bounding box containing the foreground is detected to simulate localization annotation. Right: the row and column indices of the vertices of the bounding box are collected in a feature vector $\vec{y}_k$.

In the arm experiment we built three different evidential models from these vectors. A *left* model was build using $N = 2$ features ($\max(row)$ and $\min(col)$) from the left view, and a Mixture of Gaussians with $n_i = n = 5$ components for both feature spaces. These feature components were selected as most discriminative for the motion observed (as $\max(col)$ and $\min(row)$ would remain almost constant during the arm's motion, we decided to neglect them). A second model was built for the right view only, with $N = 3$ feature spaces (associated with the components $\max(row)$, $\min(col)$ and $\max(col)$) and $n_i = n = 5$ MoG components for each feature space. This time we added $\max(col)$ to the selection to test the influence of an additional component. Finally, an overall model

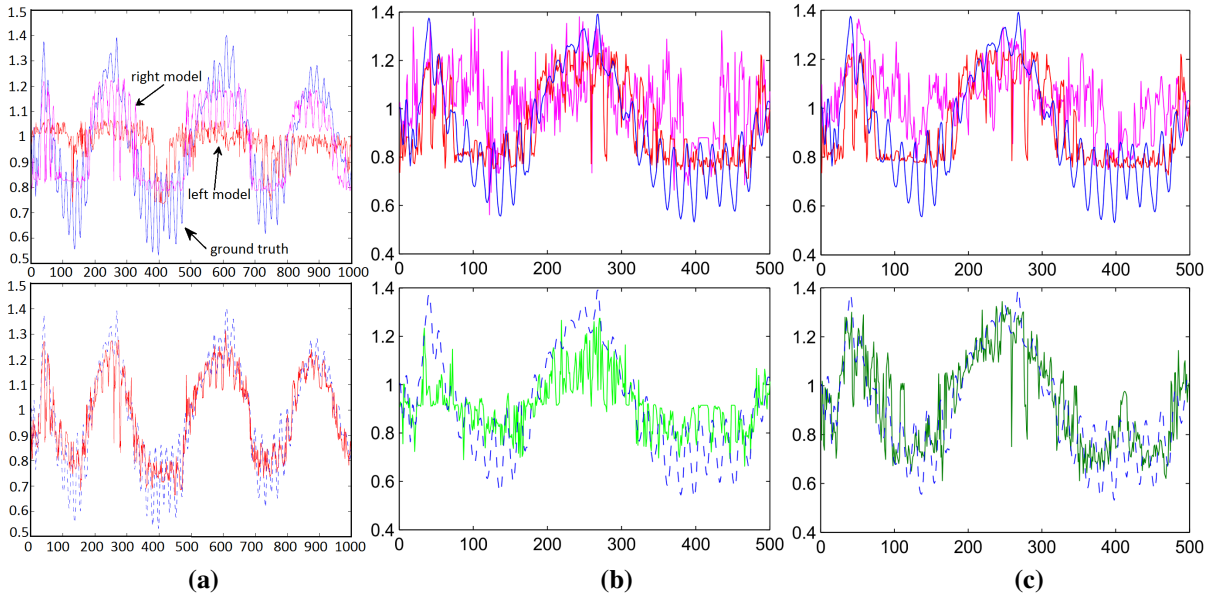

Fig. 5. Comparative performance on the arm experiment. **(a)** Performance of BMR. Top: pose estimates of component 9 of the pose vector (Y coordinate of the hand marker) produced by the left (in red) and right (in magenta) model compared to the ground truth (in blue), plotted against time. Bottom: the sequence of pose estimates yielded by the overall model which uses features computed in both left and right images is plotted (in solid red) against the ground truth (in dashed blue). **(b)**. Performance of RVM. Top: pose estimates of component 9 of the pose vector produced by a RVM using only the left (in red) and right (in magenta) features, compared to the ground truth (in blue), plotted against test time. Bottom: pose estimates yielded by RVM regression using features computed from both views, plotted in solid green against the ground truth (in dashed blue). **(c)**. Performance of GPR. Top: pose estimates of component 9 of the pose vector produced by a GPR model using only the left (in red) and right (in magenta) features, compared to the same ground truth (in blue). Bottom: pose estimates generated by a GPR regression model which using features from both views, plotted in solid green against the ground truth (in dashed blue).

was constructed from both features from the left view and features from the right one, with the same MoG representation.

For the legs experiment we built a model with $N = 6$ features ($\max(row)$, $\min(col)$ and $\max(col)$ for both views) and $n = 5$ Gaussian components per feature space.

C. Performance

To measure the accuracy of the estimates produced by different evidential models, we acquired a testing sequence for each of the two experiments and compared the results with the ground truth provided by the motion capture equipment. In both experiments we compared BMR's performance with that of Relevant Vector Machine and Gaussian Process Regression (see the Appendix for the relevant implementation details).

1) Arm experiment: in the arm experiment the test sequence was 1000 frames long. Pointwise pose estimates were extracted from belief estimates via pignistic transform as in Equation (14). As the anecdotal evidence of Figure 5-**(a)**-top indicates, the estimates of the single-view models were of rather poor quality. Indeed, recalling the discussion of Section III-C3, the minimal refinements $\bigotimes \Theta_i$ for the left-view and the right-view models were of size 22 and 80 respectively, signalling a poor model resolution. In opposition, the estimates obtained by exploiting image evidence from both views (Figure 5-**(a)**-bottom) were clearly better than a simple selection of the best partial estimate at each instant. This was confirmed by a minimal refinement $\bigotimes \Theta_i$ for the overall model with cardinality equal to 372 (the $N = 5$ features encoded by a MoG with $n = 5$ components were enough to resolve 372 of the 1700+ sample poses), with 139 sample poses individually resolved by some particular combination of the $N = 5$ feature values.

We also measured the Euclidean distance between real and expected 3D locations of each marker over the whole testing sequence. For the arm experiment, the average estimation errors were 17.3, 7.95, 13.03, and 2.7 centimeters for the markers "hand", "wrist", "elbow" and "shoulder", respectively (see Table IV-C6). As during testing the features were extracted from the estimated foreground, and no significant occlusions were present, the conflict between the different feature components was negligible throughout the test sequence.

2) Lower and upper estimates: as the belief estimate $\hat{b}(t)$ at time t amounts to a convex set $\mathcal{P}[\hat{b}(t)]$ of probability distributions on $\tilde{\mathcal{Q}}$, an expected pose estimate can be computed for each of its vertices (Equation (4)). The BMR approach can therefore provide a robust pose estimate, by computing for each instant t the minimal and maximal expected value (over $\mathcal{P}[\hat{b}(t)]$) of each component q^c of the pose vector:

$$\hat{q}^c_{\min}(t) = \min_{p \in \mathcal{P}[\hat{b}(t)]} \sum_{k=1}^{T} p(q_k) q^c_k, \;\; \hat{q}^c_{\max}(t) = \max_{p \in \mathcal{P}[\hat{b}(t)]} \sum_{k=1}^{T} p(q_k) q_k. \tag{16}$$

Figure 6 plots these lower and upper bounds to the expected pose values in the arm experiment, for three different components of the pose vector, over three subsequences of the test sequence. As it can be observed, even for the rather poor (feature-wise) evidential model built here, most of the time the true pose falls within the provided interval of expected pose estimates. Quantitatively, the percentage of test frames in which this happens for the twelve pose components is 49.25%, 44.92%, 49.33%, 50.50%, 48.50%, 48.33%, 49.17%, 54.42%,

49.67%, 51.50%, 39.33% and 43.50%, respectively. We can also measure the average Euclidean distance between the true pose estimate and the *boundary* of the interval of expected poses, for the four markers and along the entire test sequence: we obtain average 3D distances of 7.84cm, 3.85cm, 5.78cm and 2.07cm for the four markers, respectively. These figures give a better indication of the robustness of BMR than the errors associated with the central expected pose estimate given by the pignistic function (which we collected in Table IV-C6).

Note that in these tests the pose estimate interval was computed *using just a subset of the true vertices* of the belief estimate for computational reasons: the true interval is indeed wider, and amounting to even lower average estimation errors.

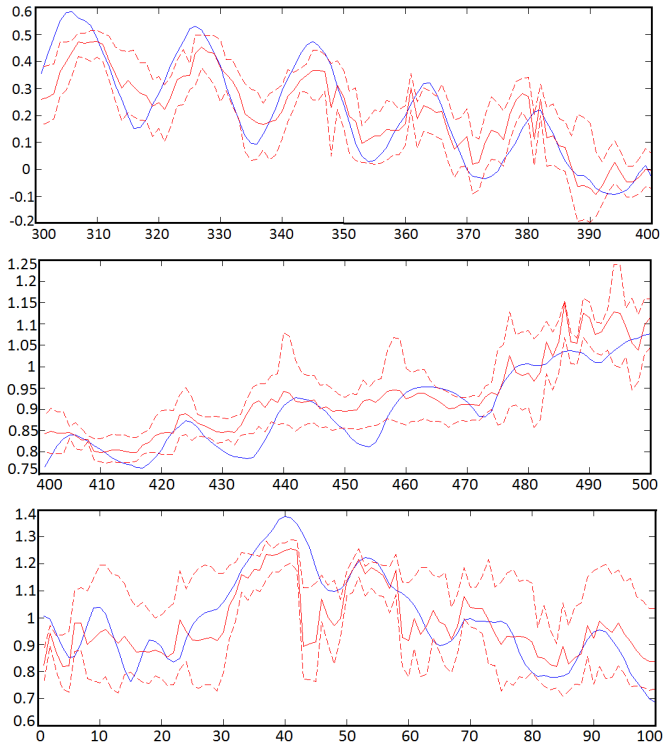

Fig. 6. Plots of lower and upper expected pose estimates (Equation (16), in dashed red) generated by the credal sets associated with the sequence of belief estimates $\hat{b}(t)$, versus the pignistic estimate (in solid red) and the ground truth (in blue). Top: component 1 of the pose vector, test frames from $t = 300$ to $t = 399$. Middle: component 6, test frames from $t = 400$ to $t = 499$. Bottom: component 9, test frames from $t = 1$ to $t = 100$.

3) Legs experiment: Figure 7-**(a)** shows instead BMR's performance in the legs experiment, for a 200-frame-long test sequence. As in Section IV-C1, the pignistic transform was adopted to extract a pointwise pose estimate at each time instant. The results were a bit less impressive but still good, mainly due to the difficulty of automatically segmenting a pair of black pants against a dark background (see Figure 3-right). Again, this cannot be considered an issue with BMR, as annotation is supposed to be given in the training stage. A quantitative assessment returned average estimation errors (for the pignistic expected pose estimate and $n = 5$) of 25.41, 19.29, 21.84, 19.88, 23.00, and 22.71 centimeters, respectively, for the six markers (located on thigh, knee and toe of each leg). These are collected again in Table IV-C6. The cameras were located at a distance of about three meters. As in the arm experiment, no significant conflict was reported.

4) Comparison with Relevance Vector Machine: Figure 5-**(b)** shows the estimates produced by a RVM on the same test sequences and components as in column **(a)**. From the top diagram, we see that the left model performs better than the right model for RVM, while Figure 5-**(a)** shows that for BMR the right model seems closer to the ground truth than the left one. In both cases, however, combining left and right features boosts the estimation's accuracy. From a visual comparison of columns **(a)** and **(b)**, it is easy to observe that our evidential model significantly outperforms a standard RVM implementation. BMR predictions also appear less noisy than RVM outputs.

Figure 7-**(b)** shows the estimates produced by RVM in the legs experiment. Here, the use of individual (left or right) images gives noisy and imprecise estimates, while combining left and right images causes the prediction task to fail completely. From the top diagram we can observe that the left model tracks the trend of variation of the joint location to some extent, while the right model gives random and noisy estimates. When combining left and right images (bottom), RVM considers all data as noise and is not able to correctly model the feature-pose mapping.

5) Comparison with Gaussian Process Regression: Figure 5-**(c)** shows the estimates produced by GPR for the same experimental setting as in columns **(a)** (for BMR) and **(b)** (for RVM). With only left or right images as inputs, the model is already able to learn the data's pattern of variation. When merging features from both views, the model performs much better. GPR estimates appear more accurate than RVM's, but are noisier and exhibit severe oscillations compared to BMR's. A visual inspection of Figure 5 shows a rather comparable performance with that of the BMR approach.

Figure 8 plots the confidence intervals of the estimates produced by GPR for the same test sequences as in Figure 6. A confidence level of 95% (corresponding to an interval of two standard deviations) is used. It should be clear, however, the difference between the confidence band (shown in Figure 8) associated with a *single* Gaussian distribution on the outputs (poses) (such as the prediction function $p(q|y, \tilde{Q}, \tilde{y})$ of a GPR) which is characterized by a *single* mean estimate and a (co)-variance, and the *interval* of *expected (mean) poses* associated with a belief estimate (which amounts to entire family of probability distributions) shown in Figure 6.
This is the consequence of the second-order uncertainty encoded by belief functions, as opposed to single classical probability distributions. Indeed, for each vertex of the credal estimate produced by BMR we could also compute (besides an expectation) a covariance and a confidence band: the cumulated confidence bands for all Probability Distribution Functions (PDFs) in the credal estimate would be a fairer comparison for the single confidence band depicted in Figure 8, and would better illustrate the approach's robustness.

Finally, Figure 7-**(c)** shows our GRP estimates for the legs experiment. The model is able to track the ground truth when combining left and right images as inputs. However, due to the higher dimensionality ($D = 18$) of the targets compared

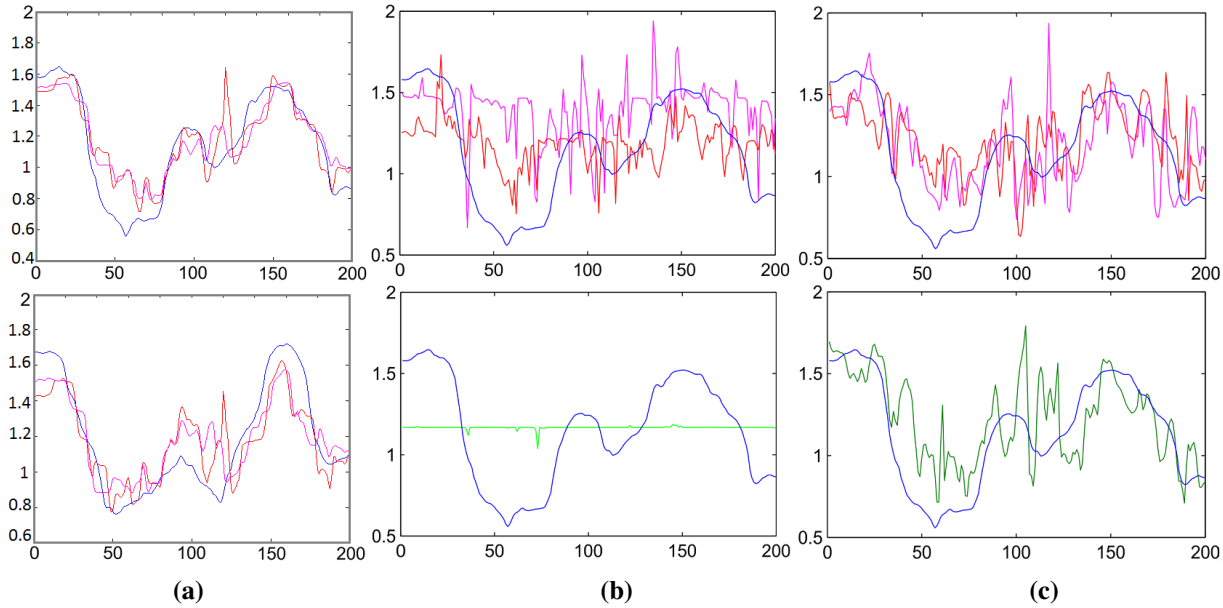

Fig. 7. Comparative performance on the legs experiment. **(a)** Performance of BMR. Performance of two versions of the two-view evidential model with $N = 6$ feature spaces, in the legs experiment, on a test sequence of length 200. The pignistic expected pose is computed for two models with $n_i = n = 5$ (in red) and $n_i = n = 4$ (in magenta) MoG components for each feature space, respectively, and plotted versus the ground truth (in blue). The estimates for components 9 (top) and 12 (bottom) of the 18-dimensional pose vector (the 3D coordinates of each of the 6 markers) are shown. **(b)**. Performance of RVM. Top: pose estimates of component 9 of the pose vector (Z coordinate of the third marker) produced by a RVM using only the left (in red) and right (in magenta) features, compared to the ground truth (in blue), plotted against time. Bottom: pose estimates yielded by a RVM regression model using features from both views, plotted (in green) against the ground truth (in blue). **(c)**. Performance of GPR. Top: pose estimates of component 9 generated by GPR using only the left (in red) and right (in magenta) features, compared to the usual ground truth (in blue). Bottom: estimates yielded by GPR regression when computing features from both views, plotted (in green) against the ground truth (in blue).

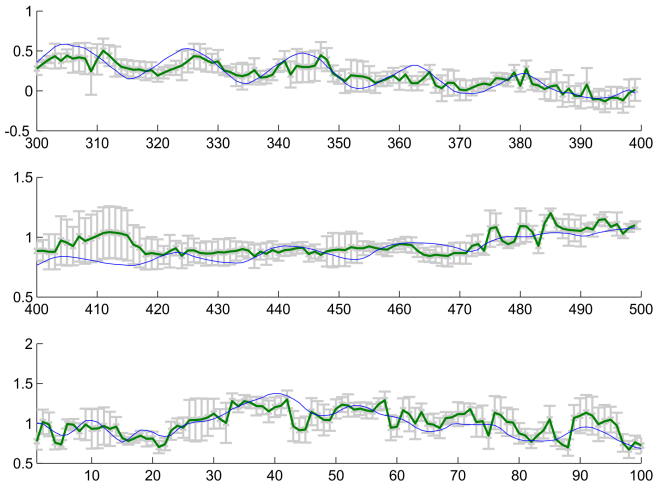

Fig. 8. Confidence intervals (two standard deviations, drawn in gray) associated with the GPR estimates (in solid deep green), plotted against the ground truth (in blue). Top: component 1 of the pose vector, test frames from $t = 300$ to $t = 399$. Middle: component 6, test frames from $t = 400$ to $t = 499$. Bottom: component 9, frames $t = 1$ to $t = 100$.

to the arm experiment ($D = 12$), neither the GPR model nor the RVM one are capable of providing accurate estimates.

6) Summary and quantitative comparison: Quantitatively, the performance of RVM, GPR and BMR are compared and shown in Table IV-C6. Pose estimation errors are calculated as average Euclidean distances (in centimeters) between real and estimated 3D location over the whole testing sequence. Estimation errors are calculated separately for each marker. From the table, we can see that the proposed BMR method gives the best result for all body joints. For all three methods, we can see that estimation errors related to joints connected to the torso, such as 'Shoulder', are much lower than those further out. This can be explained by observing that, under this experimental setting, the degree of freedom of the 'Shoulder' joint is lower than that of other joints, resulting in a less complex mapping between image features and joint locations. As an additional remark, in the tests conducted the left thigh is sometimes occluded by the right thigh, affecting the accuracy of its location's estimates.

V. Discussion

We wish to conclude by discussing the methodological justification of the proposed regression framework, in the light of the problem to solve and in comparison with similar Bayesian approaches, in particular Gaussian Process regression and Relevance Vector Machine.

A. BMR's smooth lower and upper pose estimates

Given the training data $\{\tilde{\mathcal{Q}}, \tilde{\mathcal{Y}}\}$, Belief Modeling Regression addresses the problem of estimating an unknown feature-to-pose mapping $y \mapsto q$ by providing smooth lower and upper bounds to the latter, in order to capture the inherent ambiguities associated with occlusions and perspective projection.

These bounds can be easily computed for an evidential model with a single scalar feature y. Given a probability distribution $p = \{p_k, k = 1, ..., T\}$, $\sum_k p_k = 1$ on the set of training poses $\tilde{\mathcal{Q}} = \{q_k, k = 1, ..., T\}$, the expectation

Models	Arm experiment				Legs experiment					
	Hand	Wrist	Elbow	Shoulder	Left Thigh	Left Knee	Left Toe	Right Thigh	Right Knee	Right Toe
RVM	31.2	13.6	23.0	4.5	50.5	41.7	47.2	42.7	45.0	46.4
GPR	25.0	10.6	18.6	7.0	44.3	35.0	37.2	36.5	35.3	37.3
BMR	**17.3**	**7.95**	**13.03**	**2.7**	**25.41**	**19.29**	**21.84**	**19.88**	**23.00**	**22.71**

TABLE I
ESTIMATION ERRORS (IN CENTIMETERS) OF RVM, GPR AND BMR IN BOTH ARM AND LEG EXPERIMENTS.

function (13) maps any arbitrary feature value y to a pose vector $\hat{q}(y)$. A belief estimate $\hat{b}(y)$ induced by a test feature value y on $\tilde{\mathcal{Q}}$, however, amounts to an entire convex set of probability distributions $\mathcal{P}[\hat{b}(y)]$ on $\tilde{\mathcal{Q}}$ (Section III-B3). For each scalar component q^c of the pose vector q, the pose estimate associated with y admits then the bounds (16). These are smooth functions of $y \in \mathcal{Y}$, due to the smoothness of the Gaussian likelihoods Γ we employ to learn the approximate feature space (Section III-A).

Theorem 2. [3] *When using Bayesian belief functions for inference, the lower and upper bounds (16) to the pose estimates under a single-feature evidential model are both smooth functions of $y \in \mathcal{Y}$.*

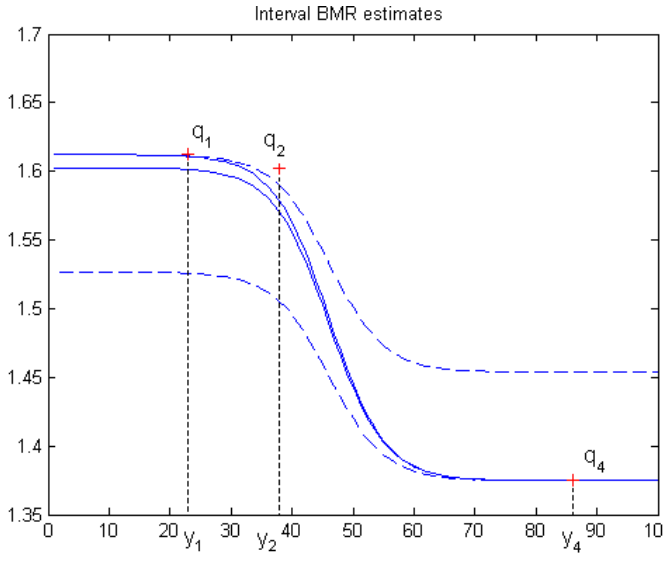

Fig. 9. The lower and upper bounds generated by Bayesian belief functions in the toy situation concerning a single-feature evidential model described in the text are depicted as solid blue curves. Using Dirichlet belief functions delivers wider, more cautious bounds (dashed blue).

The bounds are depicted as solid blue lines in the example of Figure 9, where we picked component $c = 2$ of the sample poses q_1, q_2 and q_4 of the training sequence of the arm experiment (Section IV), and built a single-feature evidential model using $y_1 = 23$, $y_2 = 38$ and $y_4 = 86$ as training feature values and $n = 2$ as EM clusters (mapped to $\{q_1, q_2\}$ and $\{q_4\}$, respectively, by the learnt refining). Within those smooth bounds, any feature-to-pose mapping is admissible, even discontinuous ones – a quite realistic situation, for the actual pose space $\mathcal{Q}$ can have holes composed by non-reachable poses, causing discontinuities in the feature-pose map. When Dirichlet b.f.s are used (Section III-B2) and a mass $m(\Theta)$ is assigned to the whole approximate feature space Θ, lower and upper bounds to pose estimates remain smooth (see Figure 9, dashed blue lines) but are more widely separated. It can be shown that the conjunctive combination of more than one feature produces rather complex (but still smooth) upper and lower boundaries to the admissible feature-pose map.

[3]Please refer to Appendix for the proof of this theorem.

B. Critical comparison with GPR and RVM

Summarising, RVM, GPR and BMR all model a family of feature-to-pose mappings, albeit of a different nature. RVM is actually a special case of GPR under sparsity constrains. In GPR and RVM, mappings are one-to-one, and probability distributions are defined over the set of mappings. The form of the family of mappings actually modeled is determined by the choice of a covariance function, which also determines a number of their features such as periodicity, continuity, etc.. After conditioning a Gaussian Process by the training data, we obtain a prediction function (17) on $\mathcal{Q}$ which follows a Gaussian distribution, given a test observation and the trained model parameters. The predicted mean and variance vary according to the test observations. In particular, the training samples are assumed correct and trustworthy – as a result, the posterior GP has zero uncertainty there.

In opposition, Belief Modeling Regression produces a random set, an entire convex set of discrete but arbitrary PDFs on the set of sample poses $\tilde{\mathcal{Q}}$, rather than on $\mathcal{Q}$. This random set (Section V-A) corresponds to a constrained family of mappings, rather than a distribution over the possible maps as in GPR. The resulting mappings are arbitrary and one-to-many, as long as they generate the learned refinings under the training data. A trait of BMR is that uncertainty is modelled even in correspondence of sample feature values: compare Figure 9, where the lower and upper mappings are separated even in correspondence of y_1, y_2 and y_4.

Different is the treatment of the uncertainty induced by the scarcity of samples (i.e., far from the samples). In GPR the standard deviation of the prediction function is influenced by both the type of prior GP selected and the distance from the samples. In BMR the width of the interval of pose estimates is influenced by both the number n_i of EM feature clusters, and the mass $m(\Theta_i)$ Dirichlet belief functions assign to the whole (approximate) feature space.

VI. CONCLUSIONS

In this paper we presented a novel approach to example-based pose estimation, in which the available evidence comes in the form of a training set of images containing sample poses of an unspecified object, whose location within those images is provided. Ground truth is available in the training stage in the form of the configurations of these sample poses. An

evidential model of the object is learned from the training data, under weak likelihood models built separately for each feature, and is exploited to estimate the pose of the object in any test image. Framing the problem within belief calculus is natural as feature-pose maps induce belief functions in the pose space, and it allows to exploit the available, limited evidence without additional assumptions, with the goal of producing the most sensible possible estimate with an attached degree of reliability. The approach was tested in a fairly challenging human pose recovery setup where it was shown to outperform popular competitors, demonstrating its potential even in the presence of poor feature representations. These results open a number of interesting directions: a proper empirical testing of object localization algorithms in conjunction with the proposed Belief Modeling Regression approach; an efficient conflict resolution mechanism able to discriminate as much as possible foreground from background features; the testing of the framework in higher-dimensional settings; the development of a fully-fledged 'evidential tracking' approach to exploit temporal information.

APPENDIX

A. Relevance Vector Machines

A *Relevance Vector Machine* (RVM) [52] is a sparse Bayesian model, learning a probabilistic mapping between inputs and targets (such as feature-pose maps in human pose estimation [14], [16]). In RVMs, given a set of example vectors $\{\mathbf{x}_n\}_{n=1}^N$ and the corresponding (scalar) targets $\{t_n\}_{n=1}^N$, the latter are modeled as linearly-weighted sums of M basis functions $\Phi(\mathbf{x}) = [\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_M(\mathbf{x})]^{\mathrm{T}}$: $t(\mathbf{x};\boldsymbol{\omega}) = \sum_{m=1}^M \omega_m \phi_m(\mathbf{x}) = \boldsymbol{\omega}^{\mathrm{T}}\Phi(\mathbf{x})$, with weights $\boldsymbol{\omega} = [\omega_1, \omega_2, \dots, \omega_M]^{\mathrm{T}}$. Sparsity can be obtained by imposing a prior on the weights as follows: $p(\boldsymbol{\omega}|\alpha_1, ..., \alpha_M) = (2\pi)^{-\frac{M}{2}} \prod_{m=1}^M \alpha_m^{\frac{1}{2}} \exp\left\{-\frac{1}{2}\sum_{m=1}^M \alpha_m \omega_m^2\right\}$, where $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_M]$ are hyperparameters, each α_m independently controlling the variance of each weight ω_m. Hyperpriors over α are defined via Gamma distributions: $p(\boldsymbol{\alpha}) = \prod_{m=1}^M \mathrm{Gamma}(\alpha_m|a,b)$, $p(\beta) = \mathrm{Gamma}(\beta|c,d)$, where $\beta \equiv \sigma^{-2}$, $\mathrm{Gamma}(\alpha|a,b) = \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}$, and $\Gamma(a) = \int_0^\infty t^{a-1}e^{-t}dt$ is the gamma function. The relevance of each basis function is represented by one hyperparameter.

Assuming the outputs are observed with a Gaussian noise of zero mean and standard deviation equal to σ, the vectors $\mathbf{t} = [t_1, ..., t_N]'$ of target values are also normally distributed around the sample mean $\overline{\mathbf{t}}$: $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) \sim N(\overline{\mathbf{t}}, \sigma)$. Given data and hyperparameters, the Bayesian posterior of the weights is: $p(\boldsymbol{\omega}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2) = \frac{p(\mathbf{t}|\boldsymbol{\omega},\sigma^2)p(\boldsymbol{\omega}|\boldsymbol{\alpha})}{p(\mathbf{t}|\boldsymbol{\alpha},\sigma^2)} = (2\pi)^{-\frac{N+1}{2}}|\Sigma|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\boldsymbol{\omega}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\boldsymbol{\omega}-\boldsymbol{\mu})\right\}$, where $\Sigma = (\sigma^{-2}\Phi^{\mathrm{T}}\Phi + \mathrm{A})^{-1}$, $\boldsymbol{\mu} = \sigma^{-2}\Sigma\Phi^{\mathrm{T}}\mathbf{t}$ and $\mathrm{A} = \mathrm{diag}(\alpha_1, \alpha_2, \dots, \alpha_M)$. The corresponding marginal likelihood can be computed by integrating out the weights: $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) = \int p(\mathbf{t}|\boldsymbol{\omega}, \sigma^2)p(\boldsymbol{\omega}|\boldsymbol{\alpha})d\boldsymbol{\omega} = \frac{|\sigma^2\mathrm{I} + \Phi\mathrm{A}^{-1}\Phi^{\mathrm{T}}|^{-\frac{1}{2}}}{(2\pi)^{\frac{N}{2}}} \exp\left\{-\frac{1}{2}\mathbf{t}^{\mathrm{T}}(\sigma^2\mathrm{I} + \Phi\mathrm{A}^{-1}\Phi^{\mathrm{T}})^{-1}\mathbf{t}\right\}$.

In this work, we use Mike Tipping's standard RVM implementation[4]. In the training stage, the marginal likelihood is maximized to find the optimal hyperparameters $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP} via the following update equations: $\alpha_m^{New} = \frac{\gamma_m}{\mu_m^2}$, $(\sigma^2)^{New} = \frac{\|\mathbf{t}-\Phi\boldsymbol{\mu}\|^2}{N-\sum_{m=1}^M \gamma_m}$, until a certain number of iterations is reached, or changes are below a threshold. Here $\gamma_m = 1 - \alpha_m\Sigma_{mm}$, $\gamma_m \in [0,1]$ is a measure of 'well-determinedness' of the parameter ω_m, and Σ_{mm} denotes the diagonal values of the posterior weight covariance matrix.

[4] http://www.relevancevector.com

B. Gaussian Process Regression

Gaussian Process Regression (GPR) [28], [33], [34] assumes that any finite set of observations $X = [\mathbf{x}_1, ..., \mathbf{x}_N]$ is drawn from a multivariate Gaussian distribution. A Gaussian process can be seen as a distribution over functions, and the distribution of the vector of target values $\mathbf{t} = [t_1, ..., t_N]'$ is completely specified by the mean $m(X)$ and the covariance matrix $K(X,X)$ of the input matrix X: $\mathbf{t} \sim \mathcal{GP}(m(X), K(X,X))$. The covariance matrix $K(X,X) = [k(\mathbf{x}_p, \mathbf{x}_q), p, q = 1, ..., N]$ is frequently defined as a Gaussian function, with: $k(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \exp\left\{-\frac{1}{2}(\mathbf{x}_p - \mathbf{x}_q)^T M (\mathbf{x}_p - \mathbf{x}_q)\right\} + \sigma_n^2\delta_{pq}$, where $\delta_{pq} = 1$ iff $p = q$ and 0 otherwise. Its hyperparameters are the standard deviation σ_f of the noise-free observations, that of the Gaussian noise (σ_n), and the parameters $\{M\}$ of the symmetric matrix M, all collected in a vector $\theta = [\{M\}, \sigma_f^2, \sigma_n^2]$.

Given a training set of noisy observations $\{(\mathbf{x}_k, t_k)\}_{k=1,...,N}$ (where N denotes the number of training samples, and t_k is a scalar target value $\forall k$), we can find the optimal hyperparameters of the Gaussian Process $\mathcal{GP}$ which best fits the data by maximizing the log marginal likelihood (see [28] for more details): $\log p(\mathbf{t}|X, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{t}^{\mathrm{T}}K_t^{-1}\mathbf{t} - \frac{1}{2}\log|K_t| - \frac{N}{2}\log(2\pi)$, where $K_t = K(X,X) + \sigma_n^2\mathrm{I}$. Given the optimal hyperparameters, GPR predicts the distribution of a test output vector $\mathbf{t}^*$ from the matrix of the test inputs X^* as follows:

$$p(\mathbf{t}^*|X, \mathbf{t}, X^*) \sim \mathcal{N}(\overline{\mathbf{t}^*}, K^*(X^*, X^*)), \tag{17}$$

with predicted mean: $\overline{\mathbf{t}^*} = K(X^*, X)[K(X,X) + \sigma_n^2 I]^{-1}\mathbf{t}$, and predicted covariance matrix: $K^*(X^*, X^*) = K(X^*, X^*) - K(X^*, X)[K(X,X) + \sigma_n^2 I]^{-1}K(X, X^*) + \sigma_n^2$. This amounts to having an entire family of regression models, all of which agree with the sample observations.

Our implementation is based on the standard GPR implementation by Carl Edward Rasmussen and Hannes Nickisch (http://www.gaussianprocess.org/gpml/code/matlab/doc/). For training multiple hyperparameters, a line search strategy is utilized for iterative optimization.

C. Proof of Theorem 2

Proof. We prove the statement for the upper bound – a dual proof holds for the lower bound. Let n be the number of EM clusters in the feature space $\mathcal{Y}$. As we encode feature values y as Bayesian belief functions on the approximate feature space Θ, the belief estimate $\hat{b}(y)$ has n disjoint focal elements $\tilde{\mathcal{Q}}^1, ..., \tilde{\mathcal{Q}}^n$ (each the image of an EM cluster in $\mathcal{Y}$) with mass $m(\tilde{\mathcal{Q}}^j) = \Gamma^j(y)/Z$, Z a normalization factor. Therefore, we can decompose the upper bound as: $\max \hat{q}^c(y) =$

$\max \sum_{q_k \in \tilde{\mathcal{Q}}} p_k(y) q_k^c = \max(\sum_{j=1}^n \sum_{q_k \in \tilde{\mathcal{Q}}^j} p_k(y) q_k^c) = \sum_{j=1}^n \max(\sum_{q_k \in \tilde{\mathcal{Q}}^j} p_k(y) q_k^c)$. By definition, every distribution $p \in \mathcal{P}[\hat{b}(y)]$ is such that $\sum_{q_k \in \tilde{\mathcal{Q}}^j} p_k(y) = m(\tilde{\mathcal{Q}}^j) = \frac{\Gamma^j(y)}{Z}$. Hence: $\max(\sum_{q_k \in \tilde{\mathcal{Q}}^j} p_k(y) q_k^c) = \frac{\Gamma^j(y)}{Z} \max_{q_k \in \tilde{\mathcal{Q}}^j} q_k^c$, for the $\max$ is obtained by assigning all mass $\frac{\Gamma^j(y)}{Z}$ to the sample with the largest pose component value. The quantity $\max_{q_k \in \tilde{\mathcal{Q}}^j} q_k^c$ does not depend on the test feature value y, but is a function of the samples in the considered focal element (set of training poses) $\tilde{\mathcal{Q}}^j$. Thus, $\max \hat{q}^c(y) = \frac{1}{Z} \sum_{j=1}^n \Gamma^j(y) \max_{q_k \in \tilde{\mathcal{Q}}^j} q_k^c$ is a smooth function, as a linear combination of the smooth functions $\Gamma^j(y)$. □

ACKNOWLEDGMENTS

This work was supported in part by grants from Natural Science Foundation of Shandong (ZR2015FL015), Qingdao Technology Plan (15-9-1-69-jch), National 973 Program (2015CB352502), Ministry of Science and Technology of China (2015IM010300) and Fundamental Research Funds for the Central Universities. The work was also partly supported by the UK Engineering and Physical Sciences Research Council (EPSRC), under Grant EP/I018719/1.

REFERENCES

[1] J. Deutscher, A. Blake and I. Reid, "Articulated body motion capture by annealed particle filtering," *CVPR'00*, pp. 126–133.
[2] H. Sidenbladh et al, "Stochastic tracking of 3D human figures using 2d image motion," *ECCV'00*, pp. 702-718.
[3] C. Sminchisescu and B. Triggs, "Kinematic jump processes for monocular 3d human tracking," *CVPR'03*, pp. 69—76.
[4] A. Elgammal and C. Lee, "Inferring 3d body pose from silhouettes using activity manifold learning," *CVPR'04*, vol. 2, pp. 681–688.
[5] G. Shakhnarovich, P. Viola, and T. Darrell, "Fast pose estimation with parameter-sensitive hashing," *ICCV'03*, pp. 750–757.
[6] C. Sminchisescu et al, "Discriminative density propagation for 3d human motion estimation," *CVPR'05*, vol. 1, pp. 390–397.
[7] T.-P. Tian, R. Li and S. Sclaroff, "Articulated pose estimation in a learned smooth space of feasible solutions," *CVPR'05*.
[8] R. Poppe and M. Poel, "Comparison of silhouette shape descriptors for example-based human pose recovery," *AFGR'06*, pp. 541–546.
[9] Y. Zheng et al, "Example based non-rigid shape detection," *ECCV'06*, vol. 4, pp. 423–436.
[10] S. Niyogi and W. T. Freeman, "Example-based head tracking," *AFGR'96*, pp. 374–378.
[11] V. Athitsos et al, "Boostmap: A method for efficient approximate similarity rankings," *CVPR'04*, vol. 2, pp. 268–275.
[12] R. Rosales and S. Sclaroff, "Specialized mappings and the estimation of human body pose from a single image," *IEEE Workshop on Human Motion*, 2000, pp. 19–24.
[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *CVPR'05*, pp. 886–893.
[14] A. Agarwal and B. Triggs, "Recovering 3D human pose from monocular images," *PAMI* 28(1) (2006), pp. 44–58.
[15] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *CVPR'01*, vol. 1, pp. 511–518.
[16] A. Thayananthan et al, "Multivariate relevance vector machines for tracking," *ECCV'06*, vol. 3, pp. 124–138.
[17] G. Mori and J. Malik, "Recovering 3D human body configurations using shape contexts," *PAMI* 28(7), pp. 1052–1062.
[18] J. Meynet, T. Arsan, J. C. Mota and J.-P. Thiran, "Fast multi-view face tracking with pose estimation," *EUSIPCO'08*.
[19] G. Shafer, *A Mathematical Theory of Evidence*. Princeton University Press, 1976.
[20] A. Dempster, "Upper and lower probabilities induced by a multivariate mapping," *Ann. Math. Statist.* 38 (1967), pp. 325–339.
[21] F. Cuzzolin, *Visions of a generalized probability theory*. Lambert Academic Publishing, September 24, 2014.
[22] G. Shafer, "Perspectives on the theory and practice of belief functions," *IJAR* 4 (1990), pp. 323–362.
[23] P. Smets and R. Kennes, "The transferable belief model," *Artificial Intelligence* 66 (1994), pp. 191–234.
[24] N. Howe et al, "Bayesian reconstruction of 3D human motion from single-camera video," *NIPS'99*.
[25] H. Sidenbladh et al, "A framework for modeling the appearance of 3D articulated figures," *AFGR'00*.
[26] R. Rosales and S. Sclaroff, "Learning and synthesizing human body motion and posture," *AFGR'00*.
[27] P. Felzenszwalb, R. Girshick, D. McAllester and D. Ramanan, "Object detection with discriminatively trained part based models," *PAMI* 32(9) (2010), pp. 1627–1645.
[28] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
[29] A. Moore, "Very fast EM-based mixture model clustering using multiresolution KD-trees," *NIPS'99*, pp. 543–549.
[30] F. Cuzzolin, *The geometry of uncertainty*. Springer-Verlag, 2017.
[31] I. Levi, *The enterprise of knowledge*. MIT Press, 1980.
[32] W. Gong, J. Brauer, M. Arens and J. Gonzalez, "Modeling vs. learning approaches for monocular 3D human pose estimation," *ICCV'11 - PERHAPS Workshop*.
[33] L. Bo and C. Sminchisescu, "Structured output-associative regression," *CVPR'09*, pp. 2403–2410.
[34] O. Rudovic and M. Pantic, "Shape-constrained Gaussian process regression for facial-point-based head-pose normalization," *ICCV'11*, pp. 1495–1502.
[35] G. Shafer, "Allocations of probability," *Annals of Probability* 7(5) (1979), pp. 827–839.
[36] H. Nguyen, "On random sets and belief functions," *J. Mathematical Analysis and Applications* 65 (1978), pp. 531–542.
[37] T. Darrell, G. Gordon, M. Harville and J. Woodfill, "Integrated person tracking using stereo, color, and pattern detection," *CVPR'98*, pp. 601–608.
[38] T. B. Moeslund and E. Granum, "3D human pose estimation using 2D-data and an alternative phase space representation," *CVPR'00 - Workshop on Human Modeling, Analysis and Synthesis*.
[39] C. Sminchisescu and B. Triggs, "Covariance scaled sampling for monocular 3D body tracking," *CVPR'01*.
[40] H. Sidenbladh and M. Black, "Learning the statistics of people in images and video," *IJCV* 54 (2003), pp. 189–209.
[41] F. Cuzzolin, "Geometry of Dempster's rule of combination," *IEEE Tr. SMC-B* 34(2) (2004), pp. 961–977.
[42] P. Smets, "Analyzing the combination of coflicting belief functions," *Information Fusion* 8(4) (2007), pp. 387–412.
[43] A. Jsang and S. Pope, "Normalising the consensus operator for belief fusion," *AJCAI'06*.
[44] P. Smets, "Decision making in the TBM: the necessity of the pignistic transformation," *IJAR* 38(2) (2005), pp. 133–147.
[45] F. Voorbraak, "A computationally efficient approximation of Dempster-Shafer theory," *Int. J. Man-Machine Studies* 30 (1989), pp. 525–536.
[46] F. Cuzzolin, "Geometry of relative plausibility and relative belief of singletons," *AMAI* 59(1) (2010), pp. 47–79.
[47] ——, "On the relative belief transform," *IJAR* 53(5) (2012), pp. 786–804.
[48] ——, "Two new Bayesian approximations of belief functions based on convex geometry," *IEEE Tr. SMC-B* 37(4) (2007), pp. 993-1008.
[49] J. Schubert, "Fast Dempster-Shafer clustering using a neural network structure," *IPMU'98*, pp. 1438-1445.
[50] T. Denoeux and A. B. Yaghlane, "Approximating the combination of belief functions using the fast Moebius transform in a coarsened frame," *IJAR* 31(1–2) (2002), pp. 77-101.
[51] S. Moral and A. Salmeron, "A Monte-Carlo algorithm for combining Dempster-Shafer belief based on approximate pre-computation," *ECSQARU'99*, pp. 305–315.
[52] M. E. Tipping, "Sparse Bayesian learning and the Relevance Vector Machine," *JMLR* 1 (2001), pp. 211–244.