Short Papers

Human Motion Tracking by Registering an Articulated Surface to 3D Points and Normals

Radu Horaud, Matti Niskanen, Guillaume Dewaele, and Edmond Boyer

Abstract—We address the problem of human motion tracking by registering a surface to 3D data. We propose a method that iteratively computes two things: maximum likelihood estimates for both the kinematic and free-motion parameters of a kinematic human-body representation, as well as probabilities that the data are assigned either to a body part or to an outlier class. We introduce a new metric between observed points and normals on one side and a parameterized surface on the other side, the latter being defined as a blending over a set of ellipsoids. We claim that this metric is well suited when one deals with either visual-hull or visual-shape observations. We illustrate the method by tracking human motions using sparse visual-shape data (3D surface points and normals) gathered from imperfect silhouettes.

Index Terms—Model-based tracking, human motion capture, articulated implicit surface, shape from silhouettes, robust surface registration, expectation-maximization.

1 Introduction

We address the problem of recovering articulated human-motion parameters using 3D data gathered from multiple image sequences. We advocate that this type of data has several advantages over 2D data: It is less ambiguous and it is less sensitive to self-occlusions. Three-dimensional features may be obtained by stereo [5], [6], [12]. Alternatively, one can capitalize on 3D shape from silhouettes. In general, 2D silhouettes are explicitly associated with a 3D smooth surface [9], [10], [12]. Another way to use silhouettes is to infer volumetric representations and to fit articulated models to the voxels thus obtained [4], [13] or to extract skeletal representations from these voxels [3]. It is also possible to infer 3D surfaces from silhouettes, namely, the *visual hull* [2] or the *visual shape* [8]. The advantage of surface-from-silhouettes is that it allows the recovery of both 3D surface points and surface normals. Moreover, there is no matching process associated with the reconstruction algorithm. Visual-hull algorithms have been proven to be extremely useful for recovering 3D meshes, which, in turn, are very useful for surface rendering. The drawback is that they need perfect silhouettes. Alternatively, visual-shape methods (such as the one described in [8]) produce *sparse* surface descriptions (points and normals) and can operate on imperfect silhouettes.

In this paper, we present a new method for tracking human motion based on fitting an articulated implicit surface to 3D points and normals. There are two important contributions. First, we introduce a new distance between an observation (a point and a normal) and an ellipsoid. We show that this can be used to define an implicit surface as a blending over a set of ellipsoids that are linked together to form a kinematic chain. Second, we exploit the analogy between the distance from a set of observations to the implicit surface and the negative log-likelihood of a mixture of Gaussian distributions. This allows us to cast the problem of implicit surface fitting into the problem of maximum likelihood (ML) estimation with hidden variables. We argue that outliers are best described by a uniform component that is added to the mixture and we formally derive the associated expectation-maximization (EM) algorithm.

Casting the data-to-model association problem into ML with hidden variables has already been addressed in the past within the framework of point registration [14], [11], [6]. In [5], observations are deterministically and iteratively assigned to each individual body part. We appear to be the first to apply a probabilistic data-to-model association framework to the problem of fitting a blending of ellipsoids to a set of 3D observations and to explicitly model outliers within this context.

The remainder of this paper is organized as follows: Section 2 describes how to compute a distance between a 3D observation (point and normal) and an ellipsoid, and how to build an implicit articulated surface based on this distance. Finally, it introduces the concept of a *probabilistic implicit surface*. Section 3 describes the formal derivation of the EM algorithm in the case of implicit surface fitting. Section 4 describes experiments with simulated data and with multiple-camera video data.

2 Modeling Articulated Objects

In order to model articulated objects such as human bodies, we must define a number of *open kinematic chains* that link the various *body parts*. We will use ellipsoids for modeling these parts. Since we measure 3D data (point and orientation vectors), we must properly define a metric that measures the discrepancy between the data and the model. This metric will be used to define a distance function as well as a probabilistic implicit surface.

2.1 The Distance from a 3D Datum to an Ellipsoid

One convenient way to describe 3D ellipsoids is to use an implicit equation and to embed the 3D Euclidean space into the 3D projective space. This yields a 4×4 full-rank symmetric matrix $\mathbf{Q}$:

$$\mathbf{Q} = \begin{bmatrix} \overline{\mathbf{Q}} & \boldsymbol{q} \\ \boldsymbol{q}^\top & q_{44} \end{bmatrix} = \begin{bmatrix} \mathbf{R}\mathbf{D}\mathbf{R}^\top & -\mathbf{R}\mathbf{D}\mathbf{R}^\top \boldsymbol{t} \\ -\boldsymbol{t}^\top \mathbf{R}\mathbf{D}\mathbf{R}^\top & \boldsymbol{t}^\top \mathbf{R}\mathbf{D}\mathbf{R}^\top \boldsymbol{t} - 1 \end{bmatrix}, \quad (1)$$

where $\mathbf{D} = \mathrm{Diag}[a^{-2}, b^{-2}, c^{-2}]$ is a 3×3 diagonal matrix, $\mathbf{R}$ is a 3×3 rotation matrix, and $\boldsymbol{t}$ is a 3D translation vector. In practice, $b = c$ and we choose $a \geq b$. We denote by $\boldsymbol{X}$ the homogeneous coordinates of a point $\boldsymbol{x}$ lying on the surface of the ellipsoid, $\boldsymbol{X}^\top \mathbf{Q} \boldsymbol{X} = 0$. The adjoint matrix $\mathbf{Q}^\star = \mathbf{Q}^{-\top}$ defines the *dual ellipsoid*. The family $\boldsymbol{P}$ of planes that are tangent to the ellipsoid $\mathbf{Q}$ satisfy the constraint $\boldsymbol{P}^\top \mathbf{Q}^{-1} \boldsymbol{P} = 0$ since $\mathbf{Q}^\top = \mathbf{Q}$. We denote by $\boldsymbol{p}$ the 3D vector that is orthogonal to the plane $\boldsymbol{P}$ (therefore, $\boldsymbol{p}$ is normal to the ellipsoid at point $\boldsymbol{x}$): $\boldsymbol{p} = \overline{\mathbf{Q}}\boldsymbol{x} + \boldsymbol{q}$, where the notations of (1) are used.

The *algebraic distance* from a 3D point Y to the surface of an ellipsoid was used in [12], defined by $q(Y) = \boldsymbol{Y}^\top \mathbf{Q} \boldsymbol{Y}$. The value of q varies from -1 at the center of the ellipsoid to 0 on its surface, and then to $+\infty$ outside the ellipsoid as the point is farther away from the surface. The *euclidean distance* from a point to an ellipsoid requires solving a six-degree polynomial. In [6], an approximation of the euclidean distance is used, i.e., a *pseudo-euclidean distance*, as shown in Fig. 1.

- *R. Horaud, G. Dewaele, and E. Boyer are with INRIA Grenoble-Rhone-Alpes, 655 avenue de l'Europe, 38330 Montonnot Saint-Martin, France. E-mail: {Radu.Horaud, Edmond.Boyer}@inrialpes.fr, Guillaume.Dewaele@ens-lyon.fr.*
- *M. Niskanen is with the Department of Electrical and Information Engineering, University of Oulu, Box 4500, FIN-90014 University of Oulu, Finland. E-mail: matti.niskanen@ee.oulu.fi.*

Manuscript received 18 Dec. 2007; accepted 24 Mar. 2008; published online 18 Apr. 2008.
Recommended for acceptance by T. Darrell.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-12-0834.
Digital Object Identifier no. 10.1109/TPAMI.2008.108.

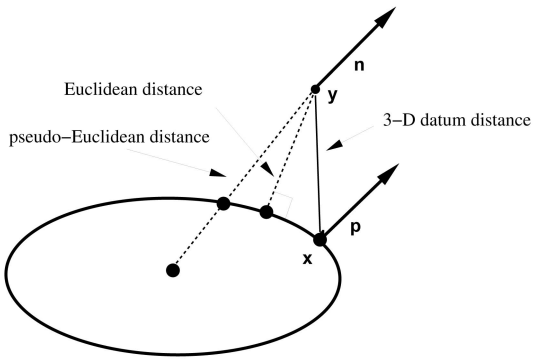

Fig. 1. The distance from the datum $\mathcal{Y} = (\boldsymbol{y}, \boldsymbol{n})$ to the ellipsoid $\mathbf{Q}$ is estimated by seeking the point $\boldsymbol{x} \in \mathbf{Q}$ such that the normal $\boldsymbol{p}$ at $\boldsymbol{x}$ is aligned with vector $\boldsymbol{n}$.

An observation will be referred to as a 3D *datum* and consists of both a 3D point and a 3D vector. We define a metric between such a 3D datum and an ellipsoid as follows: Let $\boldsymbol{Y}^\top = (\boldsymbol{y}^\top \; 1)$ be the homogeneous coordinates of an observed point and let $\boldsymbol{n}$ be a 3D observed vector. An observation or a 3D datum is denoted by $\mathcal{Y} = (\boldsymbol{y}, \boldsymbol{n})$. We seek an ellipsoid point $\boldsymbol{X} = (\boldsymbol{x}^\top \; 1)$ under the constraint that the vector $\boldsymbol{p}$ (normal to the ellipsoid at $\boldsymbol{x}$) is aligned with $\boldsymbol{n}$, e.g., Fig. 1. In other words, we seek an association between $\mathcal{X} = (\boldsymbol{x}, \boldsymbol{p})$ and $\mathcal{Y} = (\boldsymbol{y}, \boldsymbol{n})$. Fig. 2 compares the distance used in this paper with the euclidean distance from a point to an ellipsoid.

Let $d_{\mathcal{E}}(\mathcal{Y}, \mathcal{X})$ be the euclidean distance from the datum-point $\boldsymbol{y}$ to the ellipsoid-point $\boldsymbol{x}$ under the constraint that the datum-vector $\boldsymbol{n}$ and the ellipsoid-vector $\boldsymbol{p}$ are parallel:

$$d_{\mathcal{E}}(\mathcal{Y}, \mathcal{X}) = \|\boldsymbol{x} - \boldsymbol{y}\|_2 \quad \text{with} \quad \boldsymbol{n} \times \boldsymbol{p} = 0, \tag{2}$$

where $\|\boldsymbol{a}\|_2$ denotes the euclidean norm. We seek a solution for $\boldsymbol{x}$ under the constraints that $\boldsymbol{p}$ and $\boldsymbol{n}$ are parallel and yield the same orientation. Using (1), we obtain the following set of constraints:

$$\boldsymbol{x}^\top \overline{\mathbf{Q}} \boldsymbol{x} + 2\boldsymbol{x}^\top \boldsymbol{q} + q_{44} = 0, \tag{3}$$

$$\overline{\mathbf{Q}} \boldsymbol{x} + \boldsymbol{q} = \lambda \boldsymbol{n}. \tag{4}$$

From (4), we obtain $\boldsymbol{x} = \overline{\mathbf{Q}}^{-1}(\lambda \boldsymbol{n} - \boldsymbol{q})$. By substitution in (3), we obtain two solutions for λ. From $\boldsymbol{p}^\top \boldsymbol{n} > 0$, we have $\lambda > 0$ and $\lambda = (\boldsymbol{n}^\top \mathbf{R}\mathbf{D}^{-1}\mathbf{R}^\top \boldsymbol{n})^{-1/2}$. Therefore, the point onto the ellipsoid where its normal $\boldsymbol{p}$ is aligned with $\boldsymbol{n}$ is given by

$$\boldsymbol{x} = \lambda \, \mathbf{R}\mathbf{D}^{-1}\mathbf{R}^\top n + \boldsymbol{t}. \tag{5}$$

It will be convenient to use the Mahalanobis distance as follows:

$$d^2_{\mathcal{M}}(\boldsymbol{y}, \boldsymbol{x}) = (\boldsymbol{y} - \boldsymbol{x}(\mathbf{R}, \boldsymbol{t}, \boldsymbol{n}))^\top \Sigma^{-1} (\boldsymbol{y} - \boldsymbol{x}(\mathbf{R}, \boldsymbol{t}, \boldsymbol{n})). \tag{6}$$

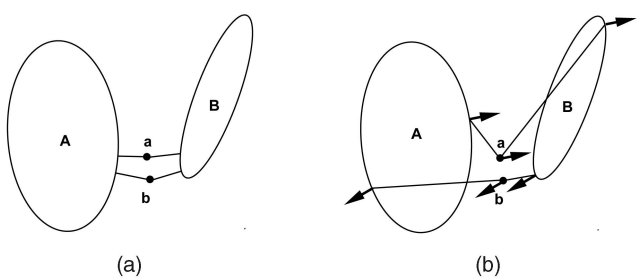

Fig. 2. (a) The classical euclidean distance from a point to an ellipsoid does not assign a point to an ellipsoid in an unambiguous way. (b) The 3D datum distance assigns without ambiguity $\boldsymbol{a}$ to $\mathbf{A}$ and $\boldsymbol{b}$ to $\mathbf{B}$.

Fig. 3. From left to right: The set of 21 ellipsoids used to model 14 body parts with 11 joints and two rotations per joint. The implicit surface defined as a blending of these ellipsoids. A set of 3D "surface" observations (points and normals) and the articulated implicit surface that has been fitted to these observations.

2.2 Kinematic Chains and Human-Body Modeling

Articulated motion has a long history in mechanics, biomechanics, robotics, and computer vision. A human body can be described by a number of *open kinematic chains* that share a common *root*. Such an open chain is composed of a number of rigid objects and two consecutive rigid objects in the chain are mechanically linked to form a joint. Rotational (or spherical) joints are the most convenient representations and they are well suited for human-body modeling. Each such joint may have one, two, or three rotational degrees of freedom. Therefore, within such a chain, a body part $\mathbf{Q}$ is linked to a *root* body part $\mathbf{Q}_r$ through a *constrained motion*, i.e., a kinematic chain with a number of rotational degrees of freedom. Since each joint may have several degrees of freedom, the total number of rotational parameters of a chain is larger than the number of rigid parts composing the chain. Moreover, the root body part undergoes a *free motion* itself, i.e., a rigid displacement with up to six degrees of freedom: three rotations and three translations.

Therefore, the motion of a body part (or ellipsoid) $\mathbf{Q}$ is composed of the root's free motion followed by the chain's constrained motion. We will denote the motion of $\mathbf{Q}$ by the 4×4 homogeneous matrix $\mathbf{T}$, which in turn is parameterized by the joint and free-motion parameter vector $\boldsymbol{\Lambda}$:

$$\mathbf{T}(\boldsymbol{\Lambda}) = \begin{bmatrix} \mathbf{R}(\boldsymbol{\Lambda}) & \boldsymbol{t}(\boldsymbol{\Lambda}) \\ \mathbf{0} & 1 \end{bmatrix}. \tag{7}$$

A complete human-body model may be described with five kinematic chains that share a common root body part. In this paper, we use the following *simplified* human-body model: There are 14 body parts and 11 joints (two ankles, two knees, two hips, two elbows, two shoulders, and a neck) with 22 rotational degrees of freedom (there are two degrees of freedom for each joint). We also consider three rotations and three translations for the free motion. Hence, there is a total of 26 degrees of freedom.

As detailed above, body parts are described by one or several ellipsoids: The feet and the thighs are described by two ellipsoids, the torso is described by three ellipsoids, and all of the other body parts are described by a single ellipsoid; hence, there are 21 ellipsoids, e.g., Fig. 3. The body parts are denoted by $\mathbf{Q}_p$, $1 \leq p \leq P$, where, for convenience, $\mathbf{Q}_1$ corresponds to the common root body part.

2.3 Articulated Implicit Surfaces

In addition to using a collection of kinematically linked ellipsoids, we will fuse them in order to define a smooth surface $\mathcal{S}$ over the entire body. This surface will be described by the implicit equation $f(\boldsymbol{y}) = C$ with $C = 1$. The *contribution* of an ellipsoid $\mathbf{Q}_p$ is defined by

$$f_p(\boldsymbol{y}) = \exp\left(-\frac{d^2_{\mathcal{M}}(\boldsymbol{y}, \boldsymbol{x}_p)}{\nu_p^2} \right), \tag{8}$$

where $d_{\mathcal{M}}$ is the Mahalanobis distance from $\boldsymbol{y}$ to the ellipsoid defined by (6), the point $\boldsymbol{x}_p$ lies onto the ellipsoid, and ν_p^2 is a parameter that tunes the spatial influence of the ellipsoid.

An implicit surface is defined as a level set of the following implicit function that is the fusion (or blending) of P ellipsoids verifying

$$f(\boldsymbol{y}) = \sum_{p=1}^{P} f_p(\boldsymbol{y}). \tag{9}$$

The class of implicit surfaces defined as above, i.e., $\boldsymbol{y} \in \mathcal{S} \Leftrightarrow f(\boldsymbol{y}) = C$, has successfully been used in computer graphics and in computer vision in conjunction with the algebraic distance [12] and with the pseudo-euclidean distance [6]. Within this paper, we extend this concept to the distance defined above. As will be detailed below, this is well suited to casting the problem of implicit surface fitting into the framework of ML in the presence of outliers.

In order to track articulated objects, the task at hand consists of fitting the articulated implicit surface just described to a set of observations. For this purpose, we first define a distance from a set of observations to the implicit surface. We have to solve the equation $f(\boldsymbol{y}) = 1$, where $\boldsymbol{y}$ is, as before, an observed 3D point. One may notice that the first-order Taylor expansion of $\ln a$ at $a = 1$ is $\ln a = a - 1 + O(a^2)$. We retain the following approximation of the distance from a set of I observations to the articulated implicit surface formed by P ellipsoids and parameterized by the kinematic variables $\boldsymbol{\Lambda}$:

$$F(\boldsymbol{\Lambda}) = -\nu^2 \sum_{i=1}^{I} \ln \sum_{p=1}^{P} \exp\left(-\frac{d_{ip}^2}{\nu^2}\right), \tag{10}$$

where

$$d_{ip}^2 = \big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}, \boldsymbol{n}_i)\big)^\top \Sigma_p^{-1} \big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}, \boldsymbol{n}_i)\big).$$

For convenience, we set $\nu = \nu_1 = \ldots = \nu_P$. The notation $\boldsymbol{x}_{ip}$ means that the 3D point $\boldsymbol{x}$ lies on ellipsoid p **and** is associated with observation i. It is worthwhile to notice that, whenever a set of observations is closed to one of the ellipsoids, the distance function is strictly equal to the sum of Mahalanobis distances from each such observation to the ellipsoid.

2.4 Probabilistic Implicit Surfaces

In this section, we introduce a probabilistic interpretation of (10). For this purpose, we denote by z_i a random variable that assigns an observation i to an ellipsoid p, namely, the notation $z_i = p$ means that the ith observation is assigned to the pth ellipsoid. There are as many *hidden* variables z_i as observations: $i \in \{1, \ldots, I\}$. The set of all the hidden variables is denoted by $\mathcal{Z} = \{z_1, \ldots, z_I\}$.

The likelihood of an observed 3D point, given its assignment to an ellipsoid and given an observed 3D normal, is drawn from a Gaussian distribution:

$$P(\boldsymbol{y}_i | z_i = p, \boldsymbol{n}_i) = \mathcal{N}\big(\boldsymbol{y}_i | \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}, \boldsymbol{n}_i), \Sigma_p\big). \tag{11}$$

In practice, the data are corrupted by noise and by errors and, therefore, there are observations that should not be assigned to an ellipsoid. For this reason, we introduce an *outlier class*, denoted by $P+1$, and we assume that the likelihood of an observation given that it is classified as an outlier is a uniform distribution over the volume V of the working space:

$$P(\boldsymbol{y}_i | z_i = P+1, \boldsymbol{n}_i) = \mathcal{U}(\boldsymbol{y}_i | V, 0) = \frac{1}{V}. \tag{12}$$

Therefore, one can write the likelihood of an observation as a mixture of P Gaussian components and one uniform component:

$$P(\boldsymbol{y}_i | \boldsymbol{n}_i) = \sum_{p=1}^{P+1} \pi_p P(\boldsymbol{y}_i | z_i = p, \boldsymbol{n}_i). \tag{13}$$

The notation

$$\pi_p = p(z_i = p | \boldsymbol{n}_i) \tag{14}$$

denotes the priors, the proportions, or the mixing parameters, and they obey the obvious constraint $\sum_{p=1}^{P+1} \pi_p = 1$. Notice that this prior probability depends on the observed vector $\boldsymbol{n}_i$. In this paper, we do not treat these observed vectors as random variables. By assuming independent and identically distributed observations, one can write the joint likelihood of all of the observations as

$$P(\mathcal{Y}_1, \ldots, \mathcal{Y}_I) = P(\boldsymbol{y}_1, \boldsymbol{n}_1, \ldots, \boldsymbol{y}_I, \boldsymbol{n}_I) = \prod_{i=1}^{I} P(\boldsymbol{y}_i | \boldsymbol{n}_i) P(\boldsymbol{n}_i).$$

Using Bayes' formula and the equations above, the negative log-likelihood is written as

$$-\ln P_{\boldsymbol{\Lambda}}(\mathcal{Y}_1, \ldots, \mathcal{Y}_I) = -\sum_{i=1}^{I} \ln \left(\sum_{p=1}^{P} \pi_p \mathcal{N}\big(\boldsymbol{y}_i | \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}, \boldsymbol{n}_i), \Sigma_p\big) + \pi_{P+1} \mathcal{U}(\boldsymbol{y}_i | V, 0) \right). \tag{15}$$

Notice that there is a strong analogy between (10) and (15): The former is a distance between a set of I observations and an articulated implicit surface, while the latter is the joint likelihood of the same observation set, where the likelihood is a mixture of P normal distributions plus a uniform distribution that captures the bad observations. This analogy will be exploited in Section 3 in order to cast the estimation of the kinematic parameters in the framework of ML with hidden variables via the EM algorithm.

3 Robust Tracking with the EM Algorithm

Because of the presence of the hidden variables, $\mathcal{Z} = \{z_1, \ldots, z_I\}$, the maximum-likelihood estimation problem, i.e., (15), does not have a simple solution. The most convenient way to maximize the likelihood of a mixture of distributions is to use the EM algorithm. The latter has been thoroughly studied in the context of data clustering [7]. In this paper, we formally derived an EM scheme in the particular case of *robustly* fitting an implicit surface to a set of 3D observations. It is worthwhile to notice that the formulas below are valid independently of the distance function being used, i.e., Fig. 1.

First, we derive the posterior class probabilities conditioned by the observations, namely,

$$P(z_i = p | \boldsymbol{y}_i, \boldsymbol{n}_i) = \frac{P(z_i = p, \boldsymbol{y}_i, \boldsymbol{n}_i)}{P(\boldsymbol{y}_i, \boldsymbol{n}_i)}.$$

We denote these posteriors by t_{ip} and, with the notations introduced in the previous section, we obviously obtain

$$t_{ip} = \frac{\pi_p P(\boldsymbol{y}_i | z_i = p, \boldsymbol{n}_i)}{P(\boldsymbol{y}_i | \boldsymbol{n}_i)}. \tag{16}$$

Second, we consider the joint probability of the set of observations $\mathcal{Y} = \{\mathcal{Y}_1, \ldots, \mathcal{Y}_I\}$ and of their assignments $\mathcal{Z}$, which yield the following expression:

$$P(\mathcal{Y}, \mathcal{Z}) = \prod_{i=1}^{I} \prod_{p=1}^{P+1} \big(P(\boldsymbol{y}_i | z_i = p, \boldsymbol{n}_i) P(z_i = p | \boldsymbol{n}_i)\big)^{\delta_p(z_i)} P(\boldsymbol{n}_i),$$

with the following definition for the function $\delta_p(z_i)$:

$$\delta_p(z_i) = \begin{cases} 1 & \text{if} \quad z_i = p \\ 0 & \text{otherwise.} \end{cases}$$

Third, we derive the expression of the *conditional expectation* of the log-likelihood taken over $\mathcal{Z}$, which, in this case, yields

$$\begin{aligned} &E[\ln P(\mathcal{Y}, \mathcal{Z})|\mathcal{Y}] \\ &\quad = \sum_{i=1}^{I}\sum_{p=1}^{P+1} E\big[\delta_p(z_i)|\mathcal{Y}\big]\big(\ln P(\boldsymbol{y}_i|z_i = p, \boldsymbol{n}_i) + \ln \pi_p\big) \\ &\qquad + (P+1)\sum_{i=1}^{I} \ln P(\boldsymbol{n}_i). \end{aligned}$$

One may notice that

$$E\big[\delta_p(z_i)|\mathcal{Y}\big] = \sum_{p=1}^{P+1} \delta_p(z_i = p) P(z_i = p|\boldsymbol{y}_i, \boldsymbol{n}_i) = t_{ip}.$$

By using the expressions of the normal and uniform distributions and by grouping constant terms, we obtain

$$\begin{aligned} &E[\ln P(\mathcal{Y}, \mathcal{Z})|\mathcal{Y}] \\ &\quad = -\frac{1}{2}\sum_{i=1}^{I}\left(\sum_{p=1}^{P} t_{ip}\big((\boldsymbol{y}_i - \boldsymbol{x}_{ip})^\top \Sigma_p^{-1}(\boldsymbol{y}_i - \boldsymbol{x}_{ip}) \right.\right. \\ &\qquad \left.\left. + \ln\det\Sigma_p - \ln\pi_p\big) + t_{iP+1}\ln\pi_{P+1}\right) + \text{const.} \end{aligned} \tag{17}$$

The maximization of (17) (or, equivalently, the minimization of its negative) will be carried out via the EM algorithm. There are, however, three notable differences between the standard EM for Gaussian mixtures [1] and our formulation:

- We added a *uniform-noise* component to the mixture. The role of this component is to "capture" outliers and hence to avoid having them influence the estimation of the model parameters.
- The means of the Gaussian components, $\boldsymbol{x}_{ip}$, are parameterized by the kinematic parameters that control the articulated motion of each ellipsoid; this has an important consequence because the M step of the algorithm will incorporate a nonlinear minimization procedure over the kinematic joints.
- At the start of the algorithm, each observation is associated with all of the ellipsoids. As the algorithm proceeds, each observation is eventually associated with one of the ellipsoids. Due to occlusions, missing data, etc., there may be ellipsoids with no associated observation. Therefore, there is a risk that the corresponding covariance becomes infinitely small. To overcome this problem, *we use a unique covariance matrix common to all of the densities in the mixture.*

Since we formally derived (17), the EM algorithm outlined below guarantees likelihood maximization. To summarize, the advantages of this formulation are 1) fast convergence properties of EM and 2) the fact that *it minimizes the negative log-likelihood* given by (15) and (17). In practice, the following EM procedure can be used for robust tracking of an articulated implicit surface:

1. **Initialization.** Compute the locations of the ellipsoid points $\boldsymbol{x}_{ip}^{(q)}$ from the current kinematic parameters $\boldsymbol{\Lambda}^{(q)}$ using (5). Similarly, initialize the covariance matrix $\Sigma^{(q)}$ common to all the ellipsoids. Initialize the priors or the mixing parameters $\pi_1^{(q)} = \ldots = \pi_{P+1}^{(q)} = 1/(P+1)$;
2. **E step.** Evaluate the posterior probabilities $t_{ip}^{(q)}$ using the current parameter values, through (16);
3. **M step.** Estimate new values for the kinematic parameters $\boldsymbol{\Lambda}^{(q+1)}$:

$$\arg\min_{\boldsymbol{\Lambda}} \frac{1}{2}\sum_{i=1}^{I}\sum_{p=1}^{P} t_{ip}^{(q)}\big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda})\big)^\top \Sigma^{(q)^{-1}}\big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda})\big).$$

Update the covariance matrix and the priors:

$$\begin{aligned} \Sigma^{(q+1)} &= \frac{1}{\sum_{p=1}^{P} T_p}\sum_{i=1}^{I}\sum_{p=1}^{P} t_{ip}^{(q)}\Big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}^{(q+1)})\Big) \\ &\qquad \times \Big(\boldsymbol{y}_i - \boldsymbol{x}_{ip}(\boldsymbol{\Lambda}^{(q+1)})\Big)^\top \\ \pi_p^{(q+1)} &= \frac{1}{I}\sum_{i=1}^{I} t_{ip}^{(q)}. \end{aligned}$$

4. **ML.** Evaluate the log-likelihood, i.e., (15), and check for convergence.

4 Experimental Results

The tracking algorithm described in Section 3 is not tight to any particular method for extracting the 3D data. In practice, we used 3D points and normals that lie on the *visual shape* [8]. Notice that the visual-shape algorithm does not require perfect silhouettes and provides as output a sparse set of 3D points and normals, not a 3D mesh. The visual-shape method uses multiple-view epipolar geometry in conjunction with the assumption that the object's surface is locally continuous and twice differentiable (see [8] for details).

4.1 Experiments with Simulated Data

We used an animation package to build a human body, to simulate various human motions, and to render image silhouettes. The simulator uses its own shape representation, which is different than ours, but it allows the user to define her/his own kinematic model. Therefore, we used the same kinematic model with the same number of degrees of freedom as the one described in Section 2.2. Nevertheless, we have not attempted to finely tune the shape parameters of our model to the simulated data. We simulated a setup composed of seven calibrated cameras. Sequences of image silhouettes were generated from the 3D model and rendered with the cameras' parameters. We computed 3D points and normals from these silhouettes and then we applied our method to these data sets. The articulated-motion parameters were recovered using our tracker. In order to assess the merits of the data-to-model fitting process, we added 20 percent of outliers uniformly distributed in the volume of the working space. These simulations allowed us to

1. assess the quality of the tracker with respect to ground-truth joint trajectories,
2. analyze the behavior of the method in the presence of various perturbations that alter the quality of the data,
3. quantify the merit of using both 3D points and normals instead of using only points, and
4. determine the optimal number of observations needed for reliable estimation of articulated motion.

We simulated a running sequence that involves both kinematic and free-motion parameters. The graphs of Fig. 4 illustrate the average error between the simulated angle values and their estimated values (measured in radians) for the kinematic parameters but not for the free-motion parameters. The first graph compares the ground-truth (simulated) joint trajectories of the left

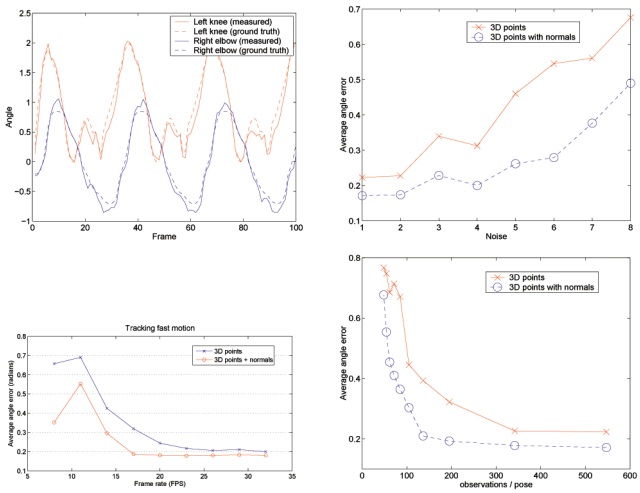

Fig. 4. The error between simulated angle values and estimated ones (measured in radians), from left to right: Ground-truth and measured trajectories over 100 frames. Average angle error as a function of silhouette noise for points and for points and normals. Average angle error as a function of frame rate. Average angle error as a function of the number of observations being used.

knee and of the right elbow (dashed curve) with the trajectories estimated with our method (solid curve) over 100 frames. The second graph illustrates the behavior of the method in the presence of silhouette noise. The results of using both 3D points and normals (dashed curve) are plotted against the results obtained using 3D points and the algebraic distance (solid line). The relatively large error corresponds to the fact that the shape model used by the animation package is not the same as our shape model. Hence, there is a systematic offset between the ground-truth kinematic parameters and the estimated parameters.

The third graph shows the average angle error as a function of the number of frames per second. The last graph shows the influence of the number of observations, where the latter varies from 50 to 550. The average angle error drops as the number of observations increases and our method (dashed) performs better than using 3D points alone (solid). From all of these experiments, one may conclude that tracking is improved when both points and normals are used instead of just points.

4.2 Experiments with Multiple-Video Data

The experimental data that we used for validating the human motion tracker was gathered with six calibrated and finely synchronized cameras. Each camera delivers 780 × 580 color images at 28 frames per second with a synchronization accuracy within 1 ms. The figures below show these image sequences sampled at 14 frames per second.

We applied articulated human motion tracking to two multiple-image sequences, the *taekwondo* sequence shown in Fig. 5 and the *leaning* sequence shown in Fig. 7. The first data set is composed of 6 × 700 frames, while the second one is composed of 6 × 200 frames. We used the same body-part dimensions for the two characters. One may notice that the silhouettes have holes and missing pieces, which results in the presence of 3D outliers. Fig. 6a shows the 3D points and normals that were reconstructed from the imperfect silhouettes; Fig. 6b shows the articulated implicit surface resulting from application of our method, while Fig. 6c shows the same surface resulting from application of our method in conjunction with the algebraic distance. Obviously, in this last case, there is a discrepancy between the data and the fitted model: The recovered motion of the right foot and the right thigh are

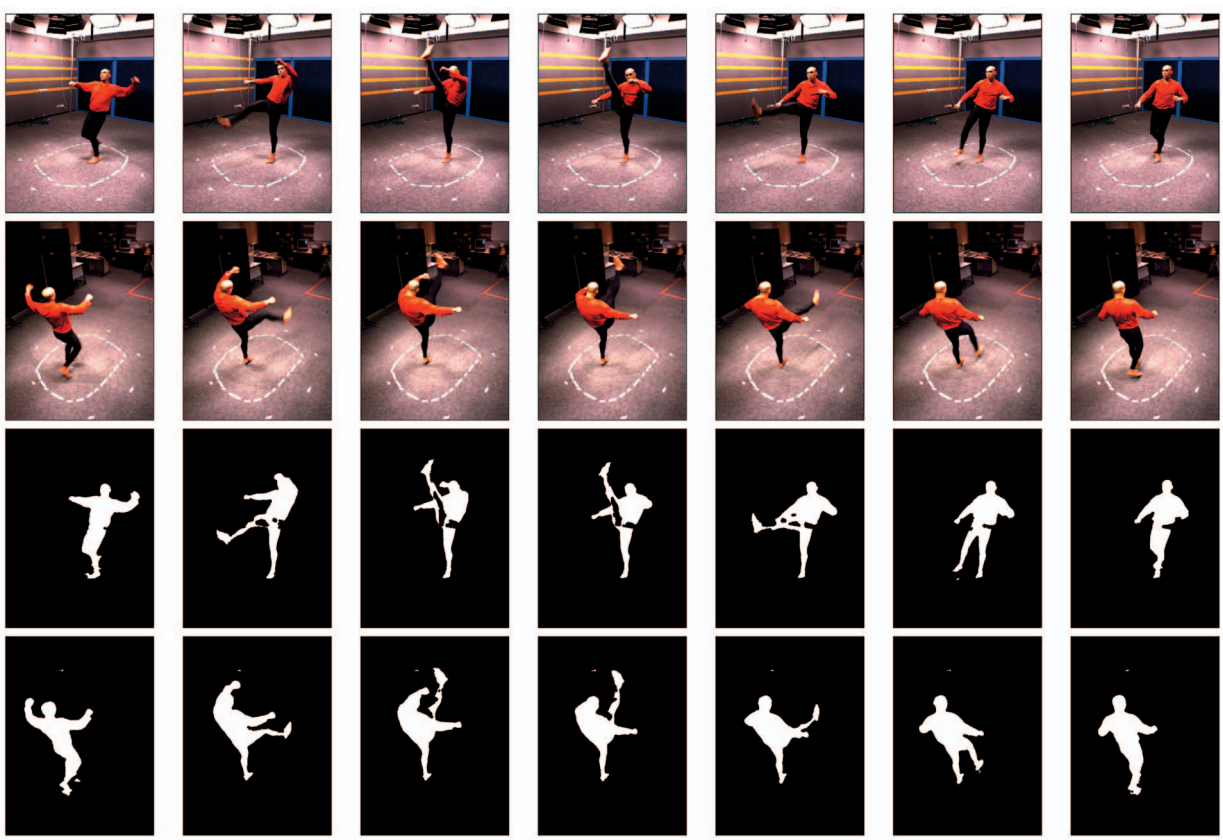

Fig. 5. The taekwondo sequence. The images and the associated silhouettes are from the first and fourth cameras. Imperfect silhouettes generate outlier data that are properly handled by our method.

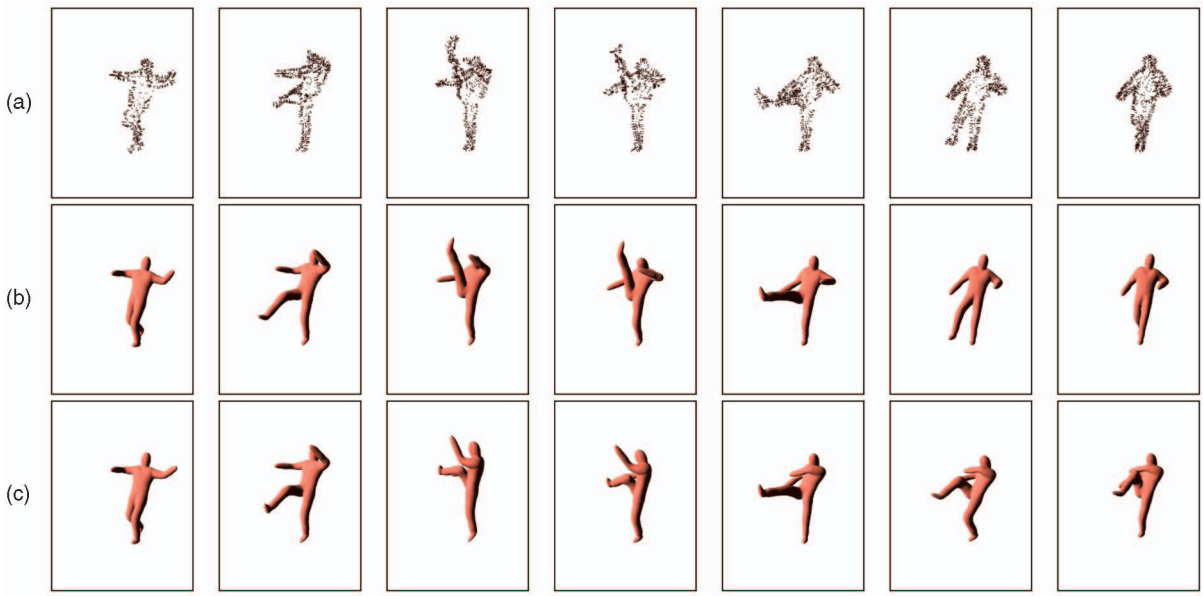

Fig. 6. (a) The 3D visual hull and (b) the implicit surface fitted to the taekwondo sequence using points and normals and (c) using points alone.

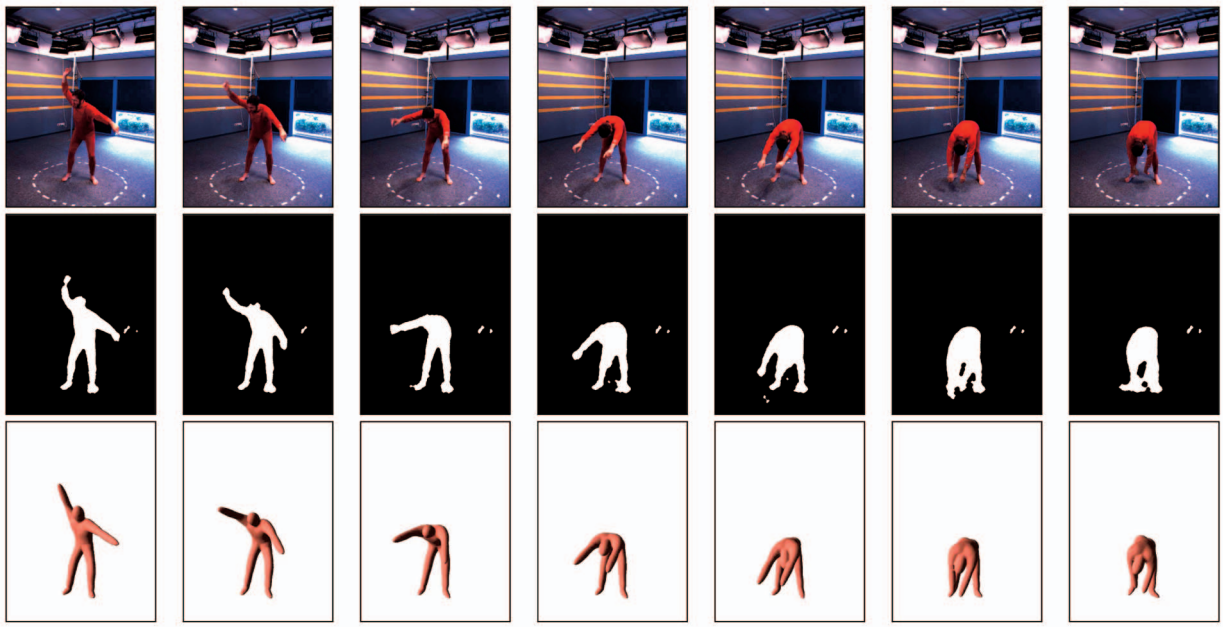

Fig. 7. The leaning sequence: Images and silhouettes associated with the first camera and the fitted implicit surface.

incorrect. Similarly, Fig. 7 shows an example of the *leaning* sequence, the corresponding silhouettes, and the fitted model using the proposed method.

5 Conclusions

In this paper, we have described a method for tracking articulated motion with several cameras. We have introduced a new metric that measures the discrepancy between observations (composed of both 3D points and 3D normals) and an articulated implicit surface. This metric is more powerful than previously used distance functions because it allows for less ambiguous associations between the data and the model. Moreover, it is well suited when one deals with either visual-hull or visual-shape representations of the data.

We cast the data-to-model fitting process into a robust probabilistic framework. We showed that there is a strong similarity between the mathematical representation of an implicit surface and a mixture of Gaussian distributions. We explored this similarity and we showed that the articulated motion tracking problem can be formulated as ML with hidden variables. We added a uniform component to the mixture to account for outliers. We formally derived an algorithm that computes ML estimates for the motion parameters within the framework of EM. Therefore, the tracker may well be interpreted in the framework of robust data clustering, where the observations are assigned to one of the ellipsoids or to an outlier component.

There are many questions that remain open and that we plan to investigate in the near future: The algorithm may be trapped in local minima if it is not properly initialized; there are some similarities between our robust tracker and the use of M-estimators and these similarities deserve further investigation. There are other interesting issues such as a thorough and quantitative evaluation of the results and their comparison with marker-based motion

capture systems and the possibility to capture several articulated motions at once.

REFERENCES

[1] C.M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
[2] E. Boyer and J.-S. Franco, "A Hybrid Approach for Computing Visual Hulls of Complex Objects," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 695-701, June 2003.
[3] G.J. Brostow, I.A. Essa, D. Steedly, and V. Kwatra, "Novel Skeletal Representation for Articulated Creatures," *Proc. Eighth European Conf. Computer Vision*, vol. 3, pp. 66-78, 2004.
[4] K.-M. Cheung, S. Baker, and T. Kanade, "Shape-from-Silhouette across Time. Part II: Applications to Human Modeling and Markerless Motion," *Int'l J. Computer Vision*, vol. 63, no. 3, pp. 225-245, 2005.
[5] D. Demirdjian, "Combining Geometric- and View-Based Approaches for Articulated Pose Estimation," *Proc. Eighth European Conf. Computer Vision*, vol. 3, pp. 183-194, 2004.
[6] G. Dewaele, F. Devernay, R. Horaud, and F. Forbes, "The Alignment between 3-D Data and Articulated Shapes with Bending Surfaces," *Proc. Ninth European Conf. Computer Vision*, vol. 3, pp. 578-591, May 2006.
[7] C. Fraley and A.E. Raftery, "Model-Based Clustering, Discriminant Analysis, and Density Estimation," *J. Am. Statistical Assoc.*, vol. 97, pp. 611-631, 2002.
[8] J.-S. Franco, M. Lapierre, and E. Boyer, "Visual Shapes of Silhouette Sets," *Proc. Third Int'l Symp. 3D Data Processing, Visualization, and Transmission*, 2006.
[9] S. Ilic, M. Salzmann, and P. Fua, "Implicit Meshes for Effective Silhouette Handling," *Int'l J. Computer Vision*, vol. 72, no. 2, pp. 159-178, 2007.
[10] D. Knossow, R. Ronfard, and R. Horaud, "Human Motion Tracking with a Kinematic Parameterization of Extremal Contours," *Int'l J. Computer Vision*, vol. 79, no. 2, pp. 247-269, Sept. 2008.
[11] B. Luo and E.R. Hancock, "Structural Graph Matching Using the EM Algorithm and Singular Value Decomposition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1120-1136, Oct. 2001.
[12] R. Plänkers and P. Fua, "Articulated Soft Objects for Multi-View Shape and Motion Capture," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1182-1187, Sept. 2003.
[13] J. Starck and A. Hilton, "Model-Based Multiple View Reconstruction of People," *Proc. Ninth IEEE Int'l Conf. Computer Vision*, Oct. 2003.
[14] W.M. Wells III, "Statistical Approaches to Feature-Based Object Recognition," *Int'l J. Computer Vision*, vol. 28, nos. 1/2, pp. 63-98, 1997.