Video-Based People Tracking

Marcus A. Brubaker, Leonid Sigal and David J. Fleet

1 Introduction

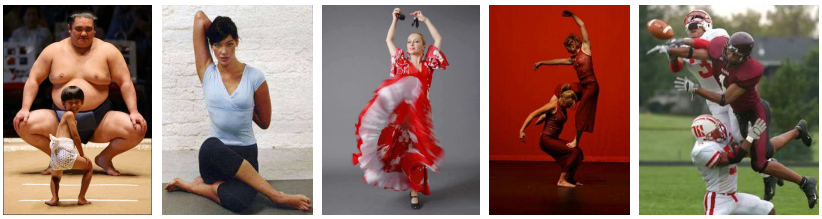
Vision-based human pose tracking promises to be a key enabling technology for myriad applications, including the analysis of human activities for perceptive environments and novel man-machine interfaces. While progress toward that goal has been exciting, and limited applications have been demonstrated, the recovery of human pose from video in unconstrained settings remains challenging. One of the key challenges stems from the complexity of the human kinematic structure itself. The sheer number and variety of joints in the human body (the nature of which is an active area of biomechanics research) entails the estimation of many parameters. The estimation problem is also challenging because muscles and other body tissues obscure the skeletal structure, making it impossible to directly observe the pose of the skeleton. Clothing further obscures the skeleton, and greatly increases the variability of individual appearance, which further exacerbates the problem. Finally, the imaging process itself produces a number of ambiguities, either because of occlusion, limited image resolution, or the inability to easily discriminate the parts of a person from one another or from the background. Some of these issues are inherent, yielding ambiguities that can only be resolved with prior knowledge; others lead to computational burdens that require clever engineering solutions.

The estimation of 3D human pose is currently possible in constrained situations, for example with multiple cameras, with little occlusion or confounding background clutter, or with restricted types of movement. Nevertheless, despite a decade of active research, monocular 3D pose tracking remains largely unsolved. From a single

Marcus A. Brubaker
Department of Computer Science, University of Toronto, e-mail: mbrubake@cs.toronto.edu

Leonid Sigal
Department of Computer Science, University of Toronto, e-mail: ls@cs.toronto.edu

David J. Fleet
Department of Computer Science, University of Toronto, e-mail: fleet@cs.toronto.edu



(a) (b) (c) (d) (e)

Fig. 1 Challenges in human pose estimation. Variation in body size and shape (a), occlusions of body parts (b), inability to observe the skeletal motion due to clothing (c), difficulty segmenting the person from the background (d), and complex interactions between people in the environment (e), are challenges that plague the recovery of human pose in unconstrained scenes.

view it is hard to escape ambiguities in depth and scale, reflection ambiguities where different 3D poses produce similar images, and missing observations of certain parts of the body because of self-occlusions.

This chapter introduces the basic elements of modern approaches to pose tracking. We focus primarily on monocular pose tracking with a probabilistic formulation. While multiview tracking in constrained settings, e.g., with minimal occlusion, may be relatively straightforward (Kakadiaris and Metaxas, 2000; Corazza, Muendermann, Chaudhari, Demattio, Cobelli, and Andriacchi, 2006) the problems faced in monocular tracking often arise in the general multiview case as well. This chapter is not intended to be a thorough review of human tracking but rather a tutorial introduction for practitioners interested in applying vision-based human tracking systems. For a more exhaustive review of the literature we refer readers to (Forsyth, Arikan, Ikemoto, O'Brien, and Ramanan, 2006; Moeslund, Hilton, and Krüger, 2006).

1.1 Tracking as Inference

Because of the inescapable uncertainty that arises due to ambiguity, and the prevalence of noisy or missing observations of body parts, it has become common to formulate human pose tracking in probabilistic terms. As such, the goal is to determine the posterior probability distribution over human poses or motions, conditioned on the image measurements (or observations).

Formally, let $\mathbf{s}_t$ denote the state of the body at time t. It represents the unknown parameters of the model we wish to estimate. In our case it typically comprises the joint angles of the body along with the position and orientation of the body in world coordinates. We also have observations at each time, denoted $\mathbf{z}_t$. This might simply be the image at time t or it might be a set of image measurements (e.g., edge locations or optical flow). Tracking can then be formulated as the problem of inferring the probability distribution over state sequences, $\mathbf{s}_{1:t} = (\mathbf{s}_1, \ldots, \mathbf{s}_t)$, conditioned on

the observation history, $\mathbf{z}_{1:t} = (\mathbf{z}_1, \ldots, \mathbf{z}_t)$; that is, $p(\mathbf{s}_{1:t}|\mathbf{z}_{1:t})$. Using Bayes' rule, it is common to express the posterior distribution as

$$p(\mathbf{s}_{1:t}|\mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_{1:t}|\mathbf{s}_{1:t})p(\mathbf{s}_{1:t})}{p(\mathbf{z}_{1:t})} . \tag{1}$$

Here, $p(\mathbf{z}_{1:t}|\mathbf{s}_{1:t})$ is called the likelihood. It is the probability of observing the image measurements given a state sequence. In effect the likelihood provides a measure of the consistency between a hypothetical motion and the given image measurements. The other major factor in (1) is the prior probability of the state sequence, $p(\mathbf{s}_{1:t})$. In effect this prior distribution captures whether a given motion is plausible or not. During pose tracking we aim to find motions that are both plausible and consistent with the image measurements. Finally, the denominator in (1), $p(\mathbf{z}_{1:t})$, often called the partition function, does not depend on the state sequence, and is therefore considered to be constant for the purposes of this chapter.

To simplify the task of approximating the posterior distribution over human motion (1), or of finding the most probable motion (i.e., the MAP estimate), it is common to assume that the likelihood and prior models can be factored further. For example, it is common to assume that the observations at each time are independent given the states. This allows the likelihood to be rewritten as a product of simpler likelihoods, one at each time:

$$p(\mathbf{z}_{1:t}|\mathbf{s}_{1:t}) = \prod_{i=1}^{t} p(\mathbf{z}_i|\mathbf{s}_i) . \tag{2}$$

This assumption and resulting factorization allows for more efficient inference and easier specification of the likelihood. Common measurement models and likelihood functions are in Section 3.

The prior distribution over human motion also plays a key role. In particular, ambiguities and noisy measurements often necessitate a prior model to resolve uncertainty. The prior model typically involves a specification of which poses are plausible or implausible, and which sequences of poses are plausible. Often this involves learning dynamical models from training data. This is discussed in Section 4.

The last two elements in a probabilistic approach to pose tracking are inference and initialization. Inference refers to the process of finding good computational approximations to the posterior distribution, or to motions that are most probable. This is discussed in Section 5. Furthermore, tracking most often requires a good initial guess for the pose at the first frame, to initialize the inference. Section 6 discusses methods for automatic initialization of tracking and for recovery from tracking failures.

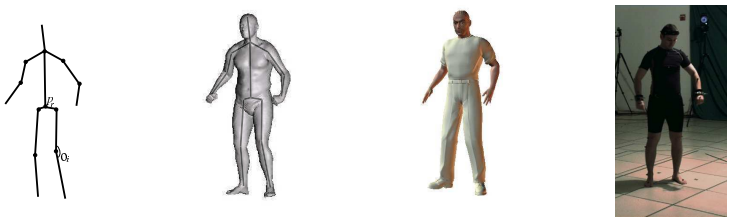


Fig. 2 Simple image formation model for a human pose. The skeleton of the human body is overlaid with soft tissue and clothing. Furthermore, the formation of an image of the person (on the right) also depends on lighting and camera parameters.

2 Generative Model for Human Pose

To begin to formulate pose tracking in more detail, we require a parameterization of human pose. While special parameterizations might be required for certain tasks, most approaches to pose tracking assume an articulated skeleton, comprising connected, rigid parts. We also need to specify the relation between this skeleton and the image observations. This is complex since we do not observe the skeleton directly. Rather, as illustrated in Figure 2, the skeleton is overlaid with soft tissue, which in turn is often covered by clothes. The image of the resulting surface also then depends on the viewpoint of the camera, the perspective projection onto the image plane, the scene illumination and several other factors.

2.1 Kinematic Parameterization

An articulated skeleton, comprising rigid parts connected by joints, can be represented as a tree. One part, such as the upper torso, is defined to be the root node and all remaining parts are either a child of the root or of another part. In this way, the entire pose can be described by the position and orientation of the root node in a global coordinate frame, and the position and orientation of each part in the coordinate frame of its parent. The state $\mathbf{s}$ then comprises these positions and orientations.

If parts are rigidly attached at joints then the number of degrees of freedom (DOFs) required will be less than the full 6 DOFs necessary to represent pose in a 3D space. The precise number of degrees of freedom varies based on the type of joint. For instance, a hinge joint is commonly used to represent the knee and has one rotational DOF while a ball-and-socket joint, often used to represent the hip, has three rotational DOFs. While real joints in the body are significantly more complex, such simple models greatly reduce the number of parameters to estimate.

One critical issue when designing the state space is the parameterization of rotations. Formally, rotations in $\mathbb{R}^3$ are 3×3 matrices with determinant 1, the set of which is denoted $SO(3)$. Unfortunately, 3×3 matrices have significantly more pa-

rameters than necessary to specify the rotation, and it is extremely difficult to keep a matrix in $SO(3)$ as it changes over time. Lower dimensional parameterizations of rotations are therefore preferred. Most common are Euler angles, which represent a rotation as a sequence of 3 elementary rotations about fixed axes. Unfortunately, Euler angles suffer from several problems including ambiguities, and singularities known as Gimbal lock. The most commonly used alternatives include exponential maps (Grassia, 1998) and quaternions (Kuipers, 2002).

2.2 Body Geometry

The skeleton is overlaid with soft tissue and clothing. Indeed we do not observe the skeleton but rather the surface properties of the resulting 3D volume. Both the geometry and the appearance of the body (and clothing) are therefore critical factors in the estimation of human pose and motion.

Body geometry has been modeled in many ways and remains a largely unexplored issue in tracking and pose estimation. A commonly used model treats the segments of the body as rigid parts whose shapes can be approximated using simple primitives such as cylinders or ellipsoids. These geometric primitives have the advantage of being simple to design and efficient to work with under perspective projection (Stenger, 2004; Wachter and Nagel, 1999). Other, more complex shape models have been used such as deformable super-quadrics (Metaxas and Terzopoulos, 1993), and implicit functions comprising mixtures of Gaussian densities to model 3D occupancy (Plankers and Fua, 2001). The greater expressiveness allows one to more accurately model the body, which can improve pose estimation, but it increases the number of parameters to estimate, and the projection of the body onto the image plane becomes more computationally expensive.

Recent efforts have been made to build detailed models of shape in terms of deformable triangulated meshes that are anchored to a skeleton. A well-known example of which is the SCAPE model (Anguelov, Srinivasan, Koller, Thrun, Rodgers, and Davis, 2005). By using dimensionality reduction, the triangulated mesh is parameterized using a small number of variables, avoiding the potential explosion in the number of parameters. Using multiple cameras one can accurately recover both the shape and pose (Balan, Sigal, Black, Davis, and Haussecker, 2007). However, the computational cost of such models is high, and may only be practical with offline processing. Good results on 3D monocular hand tracking have also been reported, based on a mesh-based surface model with approximately 1000 triangular facets (de la Gorce et al, 2008).

However, even deformable mesh body models cannot account for loose fitting clothing. Dresses and robes are extreme examples, but even loose fitting shirts and pants can be difficult to handle, since the relationship between the surface geometry observed in the image and the underlying skeleton is very complex. In most current tracking algorithms, clothing is assumed to be tight fitting so that the observed geometry is similar to the underlying body. To handle the resulting errors due to these

assumptions, the observation models (and the likelihood functions) must be robust to the kinds of appearance variations caused by clothing. Some have attempted to explicitly model the effects of clothing and its interaction with the body to account for this, but these models are complex and computationally costly (Balan and Black, 2008; Rosenhahn, Kersting, Powel, and Seidel, 2006). This remains a challenging research direction.

2.3 Image Formation

Given the pose and geometry of the body, the formation of an image of the person depends on several other factors. These include properties of the camera (e.g., the lens, aperture and shutter speed), the rest of the scene geometry (and perhaps other people), surface reflectance properties of clothing and background objects, the illumination of the scene, etc. In practice much of this information is unavailable or tedious to acquire. The exception to this is the geometric calibration of the camera. Standard methods exist (e.g., Forsyth and Ponce (2003)) which can estimate camera parameters based on images of calibration targets.[1] With fixed cameras this need only be done once. If the camera moves then certain camera parameters can be included in the state, and estimated during tracking. In either case, the camera parameters define a perspective projection, $P(\mathbf{X})$, which maps a 3D point $\mathbf{X} \in \mathbb{R}^3$ to a point on the 2D image plane.

3 Image Measurements

Given the skeleton, body geometry and image formation model, it remains to formulate the likelihood distribution $p(\mathbf{z}|\mathbf{s})$ in (1).[2] Conceptually, the observations are the image pixels, and the likelihood function is derived from ones generative model that maps the human pose to the observed image. As suggested in Figure 2, this involves modeling the surface shape and reflectance properties, the sources of illumination in the scene, and a photo-realistic rendering process for each pixel. While this can be done for some complex objects such as the human hand (de la Gorce, Paragos, and Fleet, 2008), this is extremely difficult for clothed people and natural scenes in general. Many of the necessary parameters about scene structure, clothing, reflectance and lighting are unknown, difficult to measure and not of direct interest. Instead, approximations are used that explain the available data while being (to varying degrees) independent of many of these unknown parameters. Toward that end it is common to extract a collection of image measurements, such as edge locations,

[1] Standard calibration code and tools are available as part of OpenCV (The Open Computer Vision Library), available from `http://sourceforge.net/projects/opencvlibrary/`.

[2] In this section we drop the time subscript for clarity.

which are then treated as the observations. This section briefly introduces the most common measurements and likelihood functions that are often used in practice.

3.1 2D Points

One of the simplest ways to constrain 3D pose is with a set image locations that are projections of known points on the body. These 3D points might be joint centers or points on the surface of the body geometry. For instance, it is easy to show that one can recover 3D pose up to reflection ambiguities from the 2D image positions to which the joint centers project (Taylor, 2000).

If one can identify such points (e.g., by manual initialization or ensuring that subjects wear textured clothing that produce distinct features), then the observation $\mathbf{z}$ comprises a set of 2D image locations, $\{\mathbf{m}_i\}_{i=1}^{M}$, where measurement $\mathbf{m}_i$ corresponds to location ℓ_i on part $j(i)$. If we assume that the 2D image observations are corrupted by additive noise then the likelihood function can be written as

$$p(\{\mathbf{m}_i\}_{i=1}^{M}|\mathbf{s}) = \prod_{i=1}^{M} p_i\left(\mathbf{m}_i - P(K_{j(i)}(\ell_i|\mathbf{s}))\right) \tag{3}$$

where $P(\mathbf{X})$ is the 2D camera projection of the 3D point $\mathbf{X}$, and $K_j(\ell|\mathbf{s})$ is the 3D position in the global coordinate frame of the point ℓ on part j given the current state $\mathbf{s}$. The function $p_i(d)$ is the probability density function of mean-zero additive noise on point i. This is often chosen to be Gaussian with a standard deviation of σ_i, i.e.,

$$p_i(d) = \frac{1}{\sqrt{2\pi}\,\sigma_i} \exp\left(-\frac{||d||^2}{2\sigma_i^2}\right) . \tag{4}$$

However, if it is believed that some of the points may be unreliable, for instance if they are not tracked reliably from the image sequence, then it is necessary to use a likelihood density with *heavy tails*, such as a Student's t-distribution. The greater probability density in the tails reflects our belief that measurement outliers exist, and reduces the influence of such outliers in the likelihood function.

One way to find the image locations to which the joint centers project is to detect and track a 2D articulated model (Felzenszwalb and Huttenlocher, 2005; Rehg and Kanade, 1995; Sigal and Black, 2006); unfortunately this problem is almost as challenging as the 3D pose estimation problem itself. Another approach is to find image patches that are projections of points on the body (possibly joint centers), and can be reliably tracked over time, e.g., by the KLT tracker (Tomasi and Kanade, 1991) or the WSL tracker (Jepson, Fleet, and El-Maraghi, 2003). Such a likelihood is easy to implement and has been used effectively (Urtasun, Fleet, Hertzmann, and Fua, 2005; Urtasun, Fleet, and Fua, 2006a). Nevertheless, acquiring 2D point tracks frequently requires hand initialization and tuning of the tracking algorithm. Further,

(a) Image $I(x,y)$ (b) Background $B(x,y)$ (c) Log Probability of BG $\log p_B(I(x,y))$ (d) Mask $M(x,y)$ (e) Clean Mask

Fig. 3 Background subtraction. Original image is illustrated in (a); the corresponding background image of the scene, $B(x,y)$, in (b); (c) shows the log probability of each pixel $I(x,y)$ belonging to the background (with light color corresponding to high probability); (d) illustrates the foreground mask (silhouette image) obtained by thresholding the probabilities in (c); in (e) a cleaned up version of the foreground mask in (d) obtained by simple morphological operations.

patch trackers often fail when parts are occluded or move quickly, requiring reinitialization or other modifications to maintain a reliable set of tracks.

3.2 Background Subtraction

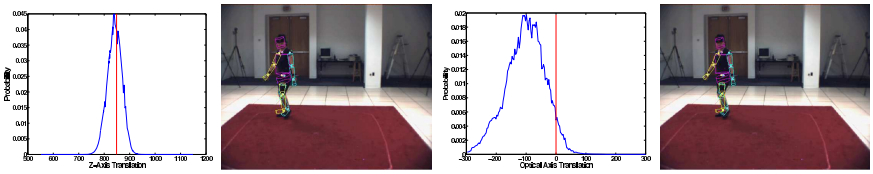
If the camera is in a fixed location and the scene is relatively static, then it is reasonable to assume that a background image $B(x,y)$ of the scene can be acquired (see Figure 3 (b)). This can then be subtracted from an observed image $I(x,y)$ and thresholded to determine a mask that indicates which pixels correspond to the foreground person (e.g., Horprasert, Harwood, and Davis (1999); Prati, Mikic, Trivedi, and Cucchiara (2003)). That is, $M(x,y) = 1$ if $||I(x,y) - B(x,y)|| > \varepsilon$ and $M(x,y) = 0$ otherwise (e.g. Figure 3 (d)). The mask can be used to formulate a likelihood by penalizing discrepancies between the observed mask $M(x,y)$ and a mask $\hat{M}(x,y|\mathbf{s})$ predicted from the image projection of the body geometry. For instance Deutscher and Reid (2005) used

$$p(M|\mathbf{s}) = \prod_{(x,y)} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|M(x,y) - \hat{M}(x,y|\mathbf{s})|}{2\sigma^2}\right) \tag{5}$$

where σ controls how strongly disagreements are penalized. Such a likelihood is attractive for its simplicity but there will be significant difficulty in setting the threshold ε to an appropriate value; there may be no universally satisfactory value.

One can also consider a probabilistic version of background subtraction which avoids the need for a threshold (see Figure 3 (c)). Instead, it is assumed that background pixels are corrupted with mean-zero, additive Gaussian noise. This yields the likelihood function

$$p(I|\mathbf{s}) = \prod_{(x,y)} p_B(I(x,y))^{1-\hat{M}(x,y|\mathbf{s})} \tag{6}$$



(a) (b) (c) (d)

Fig. 4 Background likelihood. The behavior of the background subtraction likelihood described by Equation (5) is illustrated. A true pose, consistent with the pose of the subject illustrated in Figure 3, is taken and the probability of that pose as a function of varying a single degree of freedom in the state are illustrated in (a) and (c); in (a) the entire body is shifted up and down (along the Z-axis), in (d) along the optical axis of the camera. In (b) and (d) poses corresponding to the strongest peak in the likelihood of (a) and (c) respectively are illustrated. While ideally one would prefer the likelihood to have a single global maxima at the true value (designated by the vertical line in (a) and (c)), in practice, the likelihoods tend to be noisy, multi-modal and may not have a peak in the desired location. In particular in (c), due to the insensitivity of monocular likelihoods to depth, noise in the obtained foreground mask and inaccuracies in the geometric model of the body lead to severe problems. Also note that, in both figures, the noise in the likelihood indicates that simple search methods are likely to get stuck in local optima.

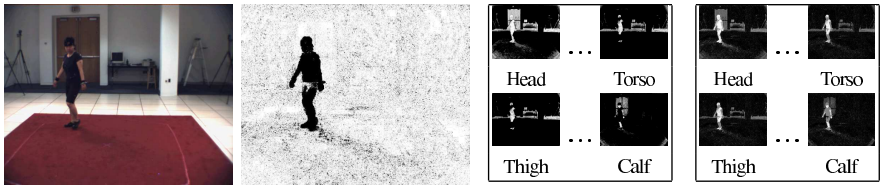
where $p_B(I(x,y))$ is the probability that pixel $I(x,y)$ is consistent with the background. For instance, a hand specified Gaussian model can be used or more complex models such as mixtures of Gaussians can be learned in advance or during tracking (Stauffer and Grimson, 1999). Such a likelihood will be more effective than one based on thresholding.

Nevertheless, background models will have difficulty coping with body parts that appear similar to the background; in such regions, like the lower part of the torso in Figure 3, the model will be penalized incorrectly. Problems also arise when limbs occlude the torso or other parts of the body, since then one cannot resolve them from the silhouette. Finally, background models often fail when the illumination changes (unless an adaptive model is used), when cameras move, or when scenes contain moving objects in the background.

3.3 Appearance Models

In order to properly handle uncertainty, e.g., when some region of the foreground appears similar to the background, it is useful to explicitly model the foreground appearance. Accordingly, the likelihood becomes

$$p(I|\mathbf{s}) = \prod_{(x,y)} p_B(I(x,y))^{1-\hat{M}(x,y|\mathbf{s})} p_F(I(x,y)|\mathbf{s})^{\hat{M}(x,y|\mathbf{s})} \tag{7}$$



(a) Image $I(x,y)$ (b) Log Probability of BG $\log p_B(I(x,y))$ (c) Log Probability of FG $\log p_F(I(x,y)|\mathbf{s})$ (d) Log FG/BG Ratio $log \frac{p_F(I(x,y)|\mathbf{s})}{p_B(I(x,y))}$

Fig. 5 Modeling appearance. The appearance likelihood described by Equation (8) is illustrated. The observed image is shown in (a); the log probability of a pixel belonging to the background in (b); the probability of the pixel belonging to a foreground model (modeled by a mixture of Gaussians) for a given body part in (c); the final log ratio of the foreground to background probability is illustrated in (d). Notice that unlike the background likelihood, the appearance likelihood is able to attribute parts of the image to individual segments of the body.

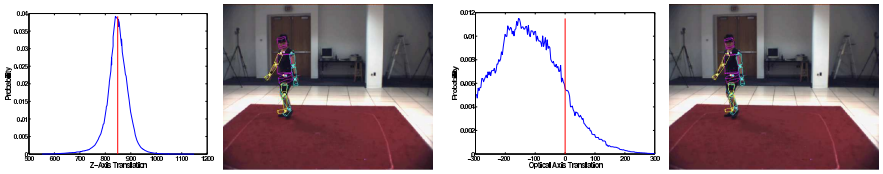
where $p_F(I(x,y)|\mathbf{s})$ is the probability of pixel $I(x,y)$ belonging to the foreground. Notice that if a uniform foreground model is assumed, i.e., $p_F(\cdot) \propto 1$, then (7) simply becomes the probabilistic background subtraction model of (6).

An accurate foreground model $p_F(I(x,y)|\mathbf{s})$ is often much harder to develop than a background model, because appearance varies depending on surface orientation with respect to the light sources and the camera, and due to complex non-rigid deformation of the body and clothing over time. It therefore requires offline learning based on a reasonable training ensemble of images (e.g., see Isard and MacCormick (2001); Ramanan, Forsyth, and Zisserman (2007)) or it can be updated online (e.g., Wren, Azarbayejani, Darrell, and Pentland (1997)). Simple foreground models are often learned from the image pixels to which the body projects to in one or more frames. For example one could learn the mean RGB color and the its covariance for the body, or for each part of the body if they differ in appearance. One can also model the statistics of simple filter outputs (e.g., gradient filters).

One important consideration about likelihoods is computational expense, as evaluating every pixel in the image can be burdensome. Fortunately, this can usually be avoided as a likelihood function typically needs only be specified up to a multiplicative constant. By dividing the likelihood by the background model for each pixel terms cancel out leaving

$$p(I|\mathbf{s}) \propto \prod_{(x,y) \text{ s.t. } \hat{M}(x,y|\mathbf{s})=1} \frac{p_F(I(x,y)|\mathbf{s})}{p_B(I(x,y))} \tag{8}$$

where the product is only over the foreground pixels, allowing a significant savings in computation. This technique can be more generally used to speed up other types of likelihood functions.



(a) (b) (c) (d)

Fig. 6 Appearance likelihood. The behavior of the appearance likelihood described by Equation (8) is illustrated. Similarly to Figure 4 a true pose, consistent with the pose of the subject illustrated in Figure 3, is taken and the probability of that pose as a function of varying a single degree of freedom in the state are illustrated in (a) and (c); as before in (a) the entire body is shifted up and down (along the Z-axis), in (d) along the optical axis of the camera. In (b) and (d) poses corresponding to the strongest peak in the likelihood of (a) and (c) respectively are illustrated. Notice that due to the strong separation between foreground and background in this image sequence, appearance likelihood performs similarly to the background likelihood model (illustrated in Figure 4); in sequences where foreground and background contain similar colors appearance likelihoods tend to produce superior performance.

3.4 Edges and Gradient Based Features

Unfortunately foreground and background appearance models have several problems. In general they have difficulty handling large changes in appearance such as those caused by varying illumination and clothing. Additionally, near boundaries they can become inaccurate since most foreground models do not capture the shading variations that occur near edges, and the pixels near the boundary are a mixture of foreground and background colors due to limited camera resolution. For this reason, and to be relatively invariant to lighting and small errors in surface geometry, it has been common to use edge-based likelihoods (e.g., Wachter and Nagel (1999)). These models assume that the projected edges of the person should correspond to some local structure in image intensity.

Perhaps the simplest approach to the use of edge information is the Chamfer distance (Barrow, Tenenbaum, Bolles, and Wolf, 1977). or the Hausdorff distance (Huttenlocher, Klanderman, and Rucklidge, 1993). Edges are first extracted from the observed image using standard edge detection methods (Forsyth and Ponce, 2003) and a distance map is computed where $d(\mathbf{x})$ is the squared Euclidean distance from pixel $\mathbf{x}$ to the nearest edge pixel. The outline of the subject in the image is computed and the boundary is sampled at a set of points $\{\mathbf{b}_i\}_{i=1}^{M}$. In the case of Chamfer matching the likelihood function is

$$p(d\,|\,\mathbf{s}) = \exp\left(-\frac{1}{M}\sum_{i=1}^{M} d(\mathbf{b}_i)\right). \tag{9}$$

Chamfer matching is fast, as the distance map need only be computed once and is evaluated only at edge points. Additionally, it is robust to changes in illumination and other appearance changes of the subject. However it can be difficult to obtain

a clean set of edges as texture and clutter in the scene can produce spurious edges. Gavrila and Davis (1996) successfully used a variant of Chamfer matching for pose tracking. To minimize the impact of spurious edges they performed an outlier rejection step on the points $\mathbf{b}_i$.

Chamfer matching is also robust to inaccuracies in the geometry of the subject. If the edges of the subject can be predicted with a high degree of accuracy, then predictive models of edge structure can be used. Kollnig and Nagel (1997) built hand specified models which predicted large gradient magnitudes near outer edges of the target. Later, this work was extended to predict gradient orientations and applied to human pose tracking by Wachter and Nagel (1999). Similarly, Nestares and Fleet (2001) learned a probabilistic model of local edge structure which was used by Poon and Fleet (2002) to track people. Such models can be effective however sufficiently accurate shape models can be difficult to build.

3.5 Discussion

There is no consensus as to which form of likelihood is best. However, some cues are clearly more powerful than others. For instance, if 2D points are practical in a given application then they should certainly be used as they are an extremely strong cue. Similarly, some form of background model is invaluable and should be used whenever it is available.

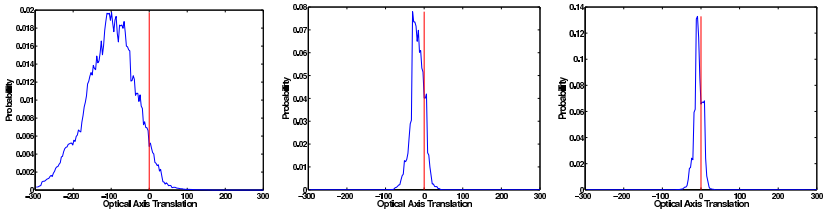
Another effective technique is to use multiple measurements. To correctly combine measurements, the joint probability of the two observations $p(\mathbf{z}^{(1)}, \mathbf{z}^{(2)} | \mathbf{s})$ needs to be specified. This is often done by assuming the conditional independence of the observations

$$p(\mathbf{z}^{(1)}, \mathbf{z}^{(2)} | \mathbf{s}) = p(\mathbf{z}^{(1)} | \mathbf{s}) p(\mathbf{z}^{(2)} | \mathbf{s}). \tag{10}$$

This assumption, often referred to as *naïve Bayes*, is unlikely to hold as errors in one observation source are often correlated with errors in others. However, it is reasonable when, for instance, the two observations are from different cameras or when one set of observations is explaining edges and the other is explaining pixels not at the boundary. The behavior of the background likelihood (previously illustrated in Figure 4) as a function of image measurements combined from multiple views is illustrated in Figure 7.

4 Motion Models

Prior information about human pose and motion is essential for resolving ambiguity, for combining noisy measurements, and for coping with missing observations. A prior model biases pose estimation toward plausible poses, when pose



(1 view) (2 views) (3 views)

Fig. 7 Number of views. Effect of combining measurements from a number of image views on the (background) likelihood. With a single view the likelihood exhibits a wide noisy mode relatively far from the true value of the translation considered (denoted by the vertical line); with more views contributing image measurements the ambiguity can be resolved, producing a stronger peak closer to the desired value.

might otherwise be under-constrained. In principle one would like to have priors that are weak enough to admit all (or most) allowable motions of the human body, but strong enough to constrain ambiguities and alleviate challenges imposed by the high-dimensional inference. The balance between these two competing goals is often elusive. This section discusses common forms of motion models and introduces some emerging research directions.

4.1 Joint Limits

The kinematic structure of the human body permits a limited range of motion in each joint. For example, knees cannot hyperextend and the torso cannot tilt or twist arbitrarily. A central role of prior models is to ensure that recovered poses satisfy such biomechanical limits. While joint limits can be encoded by thresholds imposed on each rotational DOF, the true nature of joint limits in the human body is more complex. In particular, the joint limits are dynamic and dependant on other joints (Herda, Urtasun, and Fua, 2005). Unfortunately, joint limits by themselves do not encode enough prior knowledge to facilitate tractable and robust inference.

4.2 Smoothness and Linear Dynamical Models

Perhaps the simplest commonly used prior model is a low-order Markov model, based on an assumption that human motion is smooth (e.g., Wachter and Nagel (1999); Sidenbladh, Black, and Fleet (2000); Poon and Fleet (2002)). A typical first-order model specifies that the pose at one time is equal to the previous pose up to additive noise:

$$\mathbf{s}_{t+1} = \mathbf{s}_t + \eta \tag{11}$$

where the *process noise* η is usually taken to be Gaussian $\eta \sim \mathcal{N}(0,\Sigma)$. The resulting prior is then easily shown to be

$$p(\mathbf{s}_{t+1}|\mathbf{s}_t) = G(\mathbf{s}_{t+1};\mathbf{s}_t,\Sigma) \tag{12}$$

where $G(x;m,C)$ is the Gaussian density function with mean m and covariance C, evaluated at x. Second-order models express $\mathbf{s}_{t+1}$ in terms of of $\mathbf{s}_t$ and $\mathbf{s}_{t-1}$, allowing one to use velocity in the motion model. For example, a common, damped second-order model is

$$\mathbf{s}_{t+1} = \mathbf{s}_t + \kappa(\mathbf{s}_t - \mathbf{s}_{t-1}) + \eta \tag{13}$$

where κ is a damping constant which is typically between zero and one.

Equations (11) and (13) are instances of linear models, the general form of which is $\mathbf{s}_{t+1} = \sum_{n=1}^{N} A_n \mathbf{s}_{t-n+1} + \eta$, i.e., an N-th order linear dynamical model. In many cases, as in (11) and (13), it is common to set the parameters of the transition model by hand, e.g., setting A_n, assuming a fixed diagonal covariance matrix Σ, or letting the diagonal elements of the covariance matrix in (12) be proportional to $||\mathbf{s}_t - \mathbf{s}_{t-1}||^2$ (Deutscher and Reid, 2005). One can also learn dynamical models from motion capture data (e.g., North and Blake (1997)). This would allow one, for example, to capture the coupling between different joints. Nevertheless, learning good parameters is challenging due to the high-dimensionality of the state space, for which the transition matrices, $A_n \in \mathbb{R}^{N\times N}$, can easily suffer from over-fitting.

Smoothness priors are relatively weak, and as such allow a diversity of motions. While useful, this is detrimental when the model is too weak to adequately constrain tracking in monocular videos. In constrained settings, where observations from 3 or more cameras are available and occlusions are few, such models have been shown to achieve satisfactory performance (Deutscher and Reid, 2005).

It is also clear that human motion is not always smooth, thereby violating smoothness assumptions. Motion at ground contact, for example, is usually discontinuous. One way to accommodate this is to assume a heavy-tailed model of process noise that allows occasional, large deviations from the smooth model. One might also consider the use of switching linear dynamical models, which produce piece-wise linear motions (Pavolvic, Rehg, Cham, and Murphy, 1999).

4.3 Activity Specific Models

Assuming that one knows or can infer the type of motion being tracked, or the identity of the person performing the motion, one can apply stronger prior models that are specific to the activity or subject (Lee and Elgammal, 2007). The most common approach is to learn models off-line (prior to tracking) from motion capture data. Typically one is looking for some low-dimensional parameterization of the pose and motions.

To introduce the idea, consider a dataset $\Psi = \{\psi^{(i)}\}$ consisting of K kinematic poses $\psi^{(i)}$, $i \in (1,...,K)$ obtained, for example, using a motion capture system.

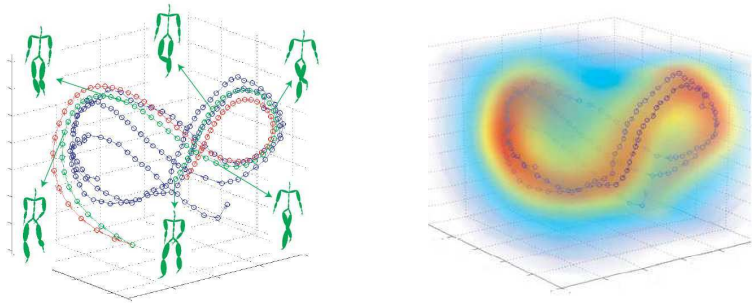


Fig. 8 Illustration of the latent space motion prior model. Results of learning a Gaussian Process Dynamical Model that encodes both the non-linear low-dimensional latent pose space and the dynamics in that space. On the left a few walking motions are shown embedded in the 3D latent space. Each point on a trajectory is an individual pose. For six of the points the corresponding mean pose in the full pose space is shown. On the right the distribution over plausible poses in the latent space is shown. This figure is re-printed from (Wang, Fleet, and Hertzmann, 2006).

Since humans often exhibit characteristic patterns of motion, these poses will often lie on or near a low-dimensional manifold in the original high-dimensional pose space. Using such data for training, methods like Principle Component Analysis (PCA) can be used to approximate poses by the linear combination of a mean pose $\mu_\Psi = \frac{1}{K}\sum_{i=1}^{K}\psi^{(i)}$ and a set of learned principal directions of variation. These principle directions are computed using the singular value decomposition (SVD) of a matrix S whose i-th row is $\psi^{(i)} - \mu_\Psi$. Using SVD, matrix S is decomposed into two orthonormal matrices U and V ($U = [u_1, u_2, ..., u_m]$ consisting of the eigenvectors, (a.k.a., *eigen-poses*) and a diagonal matrix Λ containing ordered eigenvalues such that $S = U\Lambda V^T$.

Given this learned model, a pose can be approximated by

$$\psi \approx \mu_\Psi + \sum_{i=1}^{q} u_i c_i \tag{14}$$

where c_i is the set of scalar coefficients and $q \ll m$ controls the amount of variance accounted for by the model. As such, the inference over the pose can be replaced by the inference over the coefficients $\mathbf{s} = [c_1, c_2, ..., c_q]$. Since q is typically small (e.g. $2-5$) with respect to the dimensionality of the pose space this transformation facilitates faster pose estimation. However, the new low-dimensional state space representation also requires a new model of dynamics that has to operate on the coefficients. The models of dynamics in the linear latent-space such as the one obtained using the eigen-decomposition are typically more complex then those in the original pose space and are often nonlinear. One alternative to simplifying the motion models is to learn the eigen-decomposition for entire trajectories of motion rather then the individual poses (Sidenbladh, Black, and Fleet, 2000; Urtasun, Fleet,

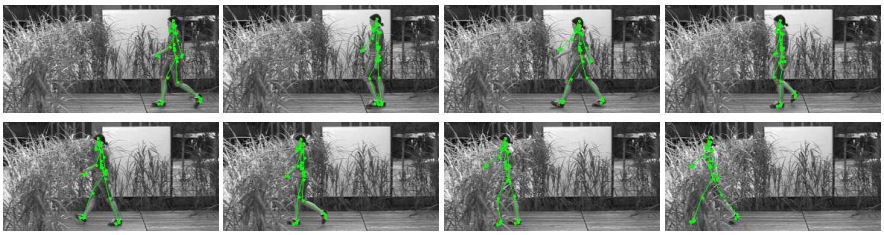


Fig. 9 Tracking with the GPDM. 56 frames of a walking motion that ends with almost total occlusion (just the head is visible) in a cluttered and moving background. Note how the prior encourages realistic motion as occlusion becomes a problem. This figure is re-printed from (Urtasun, Fleet, and Fua, 2006a).

and Fua, 2006b). Regardless, linear models such us the one described here are typically insufficient to capture intricacies of real human poses or motion.

More recent methods have shown that non-linear embeddings are more effective (Sminchisescu and Jepson, 2004). Gaussian Processes Latent Variable Models (GPLVMs) have became a popular choice since they have been shown to generalize from small amounts of training data (Urtasun, Fleet, Hertzmann, and Fua, 2005). Furthermore, one can learn a low-dimensional embedding that not only models the manifold for a given class of motions, but also captures the dynamics in that learned manifold (Li, Tian, and Sclaroff, 2007; Urtasun, Fleet, and Fua, 2006a). This allows the inference to proceed entirely in the low-dimensional space alleviating complexities imposed by the high-dimensional pose space all together. An example of a 3D latent space for walking motions is illustrated in Figure 8 and results of tracking with that model is shown in Figure 9.

Alternatively, methods that use motion capture directly to implicitly specify stronger priors have also been proposed. These types of priors make the assumption that the observed motion should be akin to the motion exhibited in the database of exemplar motions. Simply said, given a pose at time t such approaches find an exemplar motion from the database that contains a closely resembling pose, and uses that motion to look up the next pose in the sequence. Priors of this form can also be formulated probabilistically (e.g. Sidenbladh, Black, and Sigal (2002)).

All of these methods have proven effective for monocular pose inference in specific scenarios for relatively simple motions. However, due to their action specific nature, learning models that successfully generalize and represent multiple motions and transitions between those motions has been limited.

4.4 Physics-based Motion Models

Recently, there has been preliminary success in using physics-based motion models as priors. Physics-based models have the potential to be as generic as simple smoothness priors but more informative. Further, they may be able to recover sub-

tleties of more realistic motion which would be difficult, if not impossible, with existing motion models. The use of physics-based prior models in human tracking dates back to the early 1990's with pioneering work by Metaxas and Terzopoulos (1993) and Wren and Pentland (1998). However, it is only recently that success with monocular imagery has been shown.

The fundamental motivation for physics-based motion models is the possibility that motions are best described by the forces which generated them, rather than a sequence of kinematic poses. These forces include not only the internal (e.g., muscle generated) forces used to propel limbs, but also external forces such as gravity, ground reaction forces, friction and so on. Many of these forces can be derived from first principles and provide important constraints on motion. Modeling the remaining forces, either deterministically or stochastically is then the central difficulty of physics-based motion models. This class of models remains a promising but relatively unexplored area for future research.

The primary difficulty with physics-based models is the instability of complex dynamical systems. Sensitivity to initial conditions, discontinuities of motion and other non-linearities have made robust, realistic control of humanoid robots an elusive goal of robotics research. To address this in the context of tracking Brubaker, Fleet, and Hertzmann (2007) used a simplified, physical model that is stable and easy to control. While this model was restricted to simple walking motions, the work was extended by Brubaker and Fleet (2008) to a more complex physical model, capable of a wider range of motions. An alternative strategy employed by Vondrak, Sigal, and Jenkins (2008) used a motion capture database to guide the dynamics. Using inverse dynamics, they solved for the forces necessary to mimic motions found in the database.

5 Inference

In a probablistic framework our goal is to compute some approximation to the distribution $p(\mathbf{s}_{1:t} \,|\, \mathbf{z}_{1:t})$. Often this is formulated as online inference, where the distribution is computed one frame at a time as the observations arrive, exploiting the well-known recursive form of the posterior (assuming conditional independence of the observations):

$$p(\mathbf{s}_{1:t} \,|\, \mathbf{z}_{1:t}) \;\propto\; p(\mathbf{z}_t \,|\, \mathbf{s}_t)\, p(\mathbf{s}_t \,|\, \mathbf{s}_{1:t-1})\, p(\mathbf{s}_{1:t-1} \,|\, \mathbf{z}_{1:t-1}) \,. \tag{15}$$

The motion model, $p(\mathbf{s}_t|\mathbf{s}_{1:t-1})$, is often a first order Markov model which simplifies to $p(\mathbf{s}_t|\mathbf{s}_{t-1})$. While this is not strictly necessary for the inference methods presented here, it is important because the motion model then depends only on the last state as opposed to the entire trajectory.

The classic, and perhaps simplest, approach to this problem is the Kalman filter (e.g., Wachter and Nagel (1999)). However, the Kalman Filter is not suitable for human pose tracking where the dynamics are non-linear and the likelihood func-

tions are non-Gaussian. As a consequence, Sequential Monte Carlo techniques are amongst the most commonly used to perform this inference. Sequential Monte Carlo methods were first applied to visual tracking with the CONDENSATION algorithm of Isard and Blake (1998) but were applied earlier for time series analysis by Gordon, Salmond, and Smith (1993) and Kong, Liu, and Wong (1994). For a more detailed discussion of Sequential Monte Carlo methods, we refer the reader to the review article by Doucet, Godsill, and Andrieu (2000).

In this section, we present a very simple algorithm, particle filtering, in which stochastic simulation of the motion model is combined with weighting by the likelihood to produce weighted samples which approximate the posterior. We also present two variants which attempt to work around the deficiencies of the basic particle filter.

5.1 Particle Filter

A particle filter represents a distribution with a weighted set of sample states, denoted $\{(\mathbf{s}_{1:t}^{(i)}, w_{1:t}^{(i)}) | i = 1, \ldots, N\}$. When the samples are *fairly weighted*, then sample statistics approximate expectation under the target distribution, i.e.,

$$\sum_{i=1}^{N} \hat{w}_{1:t}^{(i)} f(\mathbf{s}_{1:t}^{(i)}) \approx E[f(\mathbf{s}_{1:t})] \tag{16}$$

where $\hat{w}_{1:t}^{(i)} = \left(\sum_{j=1}^{N} w_{1:t}^{(j)}\right)^{-1} w_{1:t}^{(i)}$ is the normalized weight. The sample statistics approach that of the target distribution as the number of samples, N, increases.

In a simple particle filter, given a fairly weighted sample set from time t, the samples at time $t+1$ are obtained with *importance sampling*. First samples are drawn from a proposal distribution $q(\mathbf{s}_{t+1}^{(i)} | \mathbf{s}_{1:t}^{(i)}, \mathbf{z}_{1:t+1})$. Then the weights are updated

$$w_{1:t+1}^{(i)} = w_{1:t}^{(i)} \frac{p(\mathbf{z}_{t+1} | \mathbf{s}_{t+1}^{(i)}) p(\mathbf{s}_{t+1}^{(i)} | \mathbf{s}_{1:t}^{(i)})}{q(\mathbf{s}_{t+1}^{(i)} | \mathbf{s}_{1:t}^{(i)}, \mathbf{z}_{1:t+1})} \tag{17}$$

to be fairly weighted samples at time $t+1$. The proposal distribution must be non-zero everywhere the posterior is non-zero, but it is otherwise largely unconstrained. The simplest and most common proposal distribution is motion model, $p(\mathbf{s}_{t+1} | \mathbf{s}_{1:t})$, which simplifies the weight update to be $w_{1:t+1}^{(i)} = w_{1:t}^{(i)} p(\mathbf{z}_{t+1} | \mathbf{s}_{t+1}^{(i)})$.

This simple procedure, while theoretically correct, is known to be degenerate. As t increases, the normalized weight of one particle approaches 1 while the others approach 0. Weights near zero require a significant amount of computation but contribute very little to the posterior approximation, effectively reducing the posterior approximation to a point estimate.

Algorithm 1 Particle Filtering.

Initialize the particle set $\{(\mathbf{s}_1^{(i)}, w_1^{(i)}) | i = 1,\ldots,N\}$
for $t = 1,2,\ldots$ **do**
  Compute the normalized weights $\hat{w}_{1:t}^{(i)}$ and the effective number of samples N_{eff} as in (18)
  if $N_{eff} < N_{thresh}$ **then**
    for $i = 1,\ldots,N$ **do**
      Randomly choose an index $j \in (1,...,N)$ with probability $p(j) = \hat{w}_{1:t}^{(j)}$
      Set $\tilde{\mathbf{s}}_{1:t}^{(i)} = \mathbf{s}_{1:t}^{(j)}$ and $\tilde{w}_{1:t}^{(i)} = 1$
    end for
    Replace the old particle set $\{(\mathbf{s}_{1:t}^{(i)}, w_{1:t}^{(i)}) | i = 1,\ldots,N\}$ with $\{(\tilde{\mathbf{s}}_{1:t}^{(i)}, \tilde{w}_{1:t}^{(i)}) | i = 1,\ldots,N\}$
  end if
  for $i = 1,\ldots,N$ **do**
    Sample $\mathbf{s}_{t+1}^{(i)}$ from the proposal distribution $q(\mathbf{s}_{t+1} | \mathbf{s}_{1:t}^{(i)}, \mathbf{z}_{1:t+1})$
    Construct the new state trajectory $\mathbf{s}_{1:t+1}^{(i)} = (\mathbf{s}_{1:t}^{(i)}, \mathbf{s}_{t+1}^{(i)})$
    Update the weights $w_{1:t+1}^{(i)}$ according to equation (17)
  end for
end for

To mitigate this problem, a resampling step is introduced where particles with small weights are discarded. Before the propagation stage a new set of samples $\{\tilde{\mathbf{s}}_{1:t}^{(i)} | i = 1,\ldots,N\}$ is created by drawing an index j such that $p(j) = \hat{w}_{1:t}^{(j)}$, and then setting $\tilde{\mathbf{s}}_{1:t}^{(i)} = \mathbf{s}_{1:t}^{(j)}$. The weights for this new set of particles are then $\tilde{w}_{1:t}^{(i)} = 1/N$ for all i. This resampling procedure can be done at every frame, at a fixed frequency, or only when heuristically necessary. While it may seem good to do this at every frame, as done by Isard and Blake (1998), it can cause problems. Specifically, resampling introduces bias in finite sample sets, as the samples are no longer independent and can even exacerbate particle depletion over time.

Resampling when necessary balances the need to avoid degeneracy without introducing undue bias. One of the most commonly used heuristics is an estimate of the *effective sample size*

$$N_{eff} = \left(\sum_{i=1}^{N} (\hat{w}_{1:t}^{(i)})^2 \right)^{-1} \tag{18}$$

which takes values from 1 to N. Intuitively, this can be thought of as the average number of independent samples that would survive a resampling step. Notice that after resampling, N_{eff} is equal to N. With this heuristic, resampling is then performed when $N_{eff} < N_{thresh}$, otherwise it is skipped. The particle filtering algorithm, with this heuristic resampling strategy, is outlined in Algorithm 1.

5.2 Annealed Particle Filter

Algorithm 2 Annealed Particle Filtering.

Initialize the weighted particle set $\{(\mathbf{s}_{1,L}^{(i)}, w_{1,L}^{(i)}) | i = 1, \ldots, N\}$.
for $t = 1, 2, \ldots$ **do**
  for $i = 1, \ldots, N$ **do**
    Sample $\mathbf{s}_{t+1}^{(i)}$ from the proposal distribution $q(\mathbf{s}_{t+1} | \mathbf{s}_{1:t,L}^{(i)}, \mathbf{z}_{1:t+1})$
    Construct the new state trajectory $\mathbf{s}_{1:t+1,0}^{(i)} = (\mathbf{s}_{1:t,L}^{(i)}, \mathbf{s}_{t+1}^{(i)})$
    Assign the weights $w_{1:t+1,0}^{(i)} = \frac{W_0(\mathbf{s}_{1:t+1,0}^{(i)} | \mathbf{z}_{1:t+1})}{q(\mathbf{s}_{t+1,0}^{(i)} | \mathbf{s}_{1:t,L}^{(i)}, \mathbf{z}_{1:t+1})}$
  end for
  for $\ell = 1, \ldots, L$ **do**
    for $i = 1, \ldots, N$ **do**
      Compute the normalized weights $\hat{w}_{1:t+1,\ell-1}^{(i)} = \left(\sum_{j=1}^{N} w_{1:t+1,\ell-1}^{(j)} \right)^{-1} w_{1:t+1,\ell-1}^{(i)}$
      Randomly choose an index $j \in (1, \ldots, N)$ with probability $p(j) = w_{1:t+1,\ell-1}^{(j)}$
      Sample $\mathbf{s}_{t+1,\ell}^{(i)}$ from the diffusion distribution $T_\ell(\mathbf{s}_{t+1,\ell} | \mathbf{s}_{1:t+1,\ell-1}^{(j)})$
      Construct the new state trajectory $\mathbf{s}_{1:t+1,\ell}^{(i)} = (\mathbf{s}_{1:t}^{(i)}, \mathbf{s}_{t+1,\ell}^{(i)})$
      Compute the annealed weights $w_{1:t+1,\ell}^{(i)} = W_\ell(\mathbf{s}_{1:t+1,\ell}^{(i)} | \mathbf{z}_{1:t+1})$
    end for
  end for
end for

Unfortunately, resampling does not solve all the problems of the basic particle filter described above. Specifically, entire modes of the posterior can still be missed, particularly if they are far from the modes of the proposal distribution or if modes are extremely peaked. One solution to this problem is to increase the number of particles, N. While this will solve the problem in theory, the number of samples theoretically needed is generally computationaly untenable. Further, many samples will end up representing uninteresting parts of the space. While these issues remain challenges, several techniques have been proposed in an attempt to improve the efficiency of particles filters. One approach, inspired by simulated annealling and continuation methods, is Annealed Particle Filtering (APF) (Deutscher and Reid, 2005).

The APF algorithm is outlined in Algorithm 2. At each time t, the APF goes through L levels of annealing. For each particle i, annealing level ℓ, begins by choosing a sample from the previous annealing level, $\mathbf{s}_{1:t+1,\ell-1}^{(j)}$, with probability $p(j) = \hat{w}_{1:t+1,\ell-1}^{(i)}$. The state at time $t+1$ of sample j is then diffused to create a new hypothesis $\mathbf{s}_{t+1,\ell}^{(i)}$ according to

$$T_\ell(\mathbf{s}_{t+1,\ell} | \mathbf{s}_{1:t+1,\ell-1}) = \mathcal{N}(\mathbf{s}_{t+1,\ell} | \mathbf{s}_{t+1,\ell-1}, \alpha^\ell \Sigma) \,, \tag{19}$$

and weights the new hypothesis by

$$W_\ell(\mathbf{s}_{1:t+1,\ell} | \mathbf{z}_{1:t+1}) = \left(p(\mathbf{z}_{t+1} | \mathbf{s}_{t+1,\ell}) p(\mathbf{s}_{t+1,\ell} | \mathbf{s}_{t,\ell}) \right)^{\beta_\ell} \,. \tag{20}$$

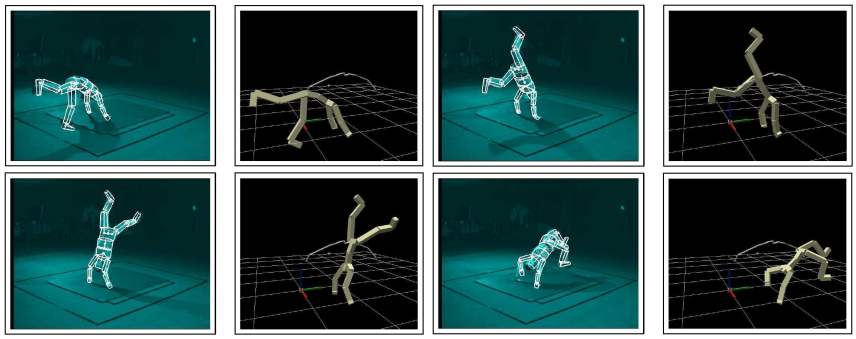


Fig. 10 Annealed Particle Filter (APF) tracking from multi-view observations. This figure is re-printed from (Deutscher and Reid, 2005).

In the above $\alpha \in (0,1)$ is called the annealing rate and is used to control the scale of covariance, Σ, in the diffusion process. The β_ℓ is the temperature parameter, derived based on the annealing rate, α, and the survival diagnostics of the particle set (for details see Deutscher and Reid (2005)) to ensure that a fixed fraction of samples survive from one stage of annealing to the next.

The sequence $\beta_0, \ldots, \beta_L$ is a gradually increasing sequence between zero and 1, ending with $\beta_L = 1$. When β_ℓ is small, the difference in height between peaks and troughs of the posterior are attenuated. As a result it is less likely that one mode, by chance, will dominate and attract all the particles, thereby neglecting other, potentially important modes. In this way the APF allows particles to broadly explore the posterior in the early stages. This means that the particles are better able to find different potential peaks, which then attract the particles more strongly as the likelihood becomes more strongly peaked (as β_ℓ increases). It is worth noting that, with $L = 1$, the APF reduces to the standard particle filter discussed in the previous section.

While often effective in finding significant modes of the posterior, the APF does not produce fairly weighted samples from the posterior. As such it does not accurately represent the posterior and the sample statistics of (16) are not representative of expectations under the posterior. Recent research has shown that by restricting the form of the diffusion and properly weighting the samples, one can obtain fairly weighted samples (Gall, Potthoff, Schnorr, Rosenhahn, and Seidel, 2007; Neal, 2001).

Algorithm 3 Markov Chain Monte Carlo Filtering.

Initialize $\{(\mathbf{s}_1^{(i)}, w_1^{(i)}) | i = 1, \ldots, N\}$
for $t = 1, 2, \ldots$ **do**
  for $i = 1, \ldots, N$ **do**
    Sample $\tilde{\mathbf{s}}_{t+1}^{(i)}$ from the proposal distribution $q(\tilde{\mathbf{s}}_{t+1} | \mathbf{s}_{1:t}^{(i)}, \mathbf{z}_{1:t+1})$
    Construct the new state trajectory $\tilde{\mathbf{s}}_{1:t+1}^{(i)} = (\mathbf{s}_{1:t}, \tilde{\mathbf{s}}_{t+1}^{(i)})$
    Update the weights $w_{1:t+1}^{(i)}$ according to equation (17) with $\mathbf{s}_{t+1}^{(i)} = \tilde{\mathbf{s}}_{t+1}^{(i)}$
  end for
  Compute the normalized weights $\hat{w}_{1:t+1}^{(i)} = \left(\sum_{j=1}^{N} w_{1:t+1}^{(j)}\right)^{-1} w_{1:t+1}^{(i)}$
  for $i = 1, \ldots, N$ **do**
    Randomly choose an index $j \in (1, ..., N)$ with probability $p(j) = \hat{w}_{1:t+1}^{(j)}$.
    Set the target distribution to be $\mathscr{P}(\mathbf{q}) \propto p(\mathbf{z}_{t+1} | \mathbf{q}) p(\mathbf{q} | \tilde{\mathbf{s}}_{1:t}^{(j)})$
    Set the initial state of the Markov Chain to $\mathbf{q}_0 = \tilde{\mathbf{s}}_{t+1}^{(j)}$
    for $r = 1, \ldots, R$ **do**
      Sample $\mathbf{q}_r$ from the MCMC transition density $T(\mathbf{q}_r | \mathbf{q}_{r-1})$, e.g., using Hybrid Monte Carlo as described in Algorithm 4
    end for
    Set $\mathbf{s}_{1:t+1}^{(i)} = (\tilde{\mathbf{s}}_{1:t}^{(i)}, \mathbf{q}_R)$ and $w_{1:t+1}^{(i)} = 1$
  end for
end for

5.3 Markov Chain Monte Carlo Filtering

Another way to improve the efficiency of particle filters is with the help of Markov Chain Monte Carlo (MCMC) methods to explore the posterior. A Markov Chain[3] is a sequence of random variables $\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2, \ldots$ with the property that for all i, $p(\mathbf{q}_i | \mathbf{q}_0, \ldots, \mathbf{q}_{i-1}) = p(\mathbf{q}_i | \mathbf{q}_{i-1})$. In MCMC, the goal is to construct a Markov Chain chain such that, as i increases, $p(\mathbf{q}_i)$ approaches the desired target distribution $\mathscr{P}(\mathbf{q})$. In the context of particle filtering at time t, the random variables, $\mathbf{q}$, are hypothetical states $\mathbf{s}_t$ and the target distribution, $\mathscr{P}(\mathbf{q})$, is the posterior $p(\mathbf{s}_{1:t} | \mathbf{z}_{1:t})$. The key to MCMC is the definition of a suitable transition density $p(\mathbf{q}_i | \mathbf{q}_{i-1}) = T(\mathbf{q}_i | \mathbf{q}_{i-1})$. To this end there are several properties that must be satisfied, one of which is

$$\mathscr{P}(\mathbf{q}) = \int T(\mathbf{q} | \hat{\mathbf{q}}) \mathscr{P}(\hat{\mathbf{q}}) d\hat{\mathbf{q}} \,. \tag{21}$$

This means that the chain has the target distribution $\mathscr{P}(\mathbf{q})$ as its stationary distribution. For a good review of the various types of Markov transition densities used, and a more thorough introduction to MCMC in general, see (Neal, 1993).

A general MCMC-filtering algorithm is given in Algorithm 3. It begins by propogating samples through time and updating their weights according to a conventional particle filter. These particles are then are chosen with probability proportional to their weights, as the initial states in N independant Markov chains. The target dis-

[3] A full review of MCMC methods is well beyond the scope of this chapter and only a brief introduction will be presented here.

Algorithm 4 Hybrid Monte Carlo Sampling.

Given a starting state $\mathbf{q}_0 \in \mathbb{R}^m$ and a target distribution $\mathscr{P}(\mathbf{q})$, define $E(\mathbf{q}) = -\log \mathscr{P}(\mathbf{q})$.
Draw a momentum vector $\mathbf{p}_0 \in \mathbb{R}^m$ from a Gaussian distribution with mean 0 and unit variance.
for $\ell = 1, \ldots, L$ **do**
  $\mathbf{p}_{\ell-0.5} = \mathbf{p}_{\ell-1} - \frac{1}{2}\Delta \frac{\partial E(\mathbf{q}_{\ell-1})}{\partial \mathbf{q}}$
  $\mathbf{q}_\ell = \mathbf{q}_{\ell-1} + \Delta \mathbf{p}_{\ell-0.5}$
  $\mathbf{p}_\ell = \mathbf{p}_{\ell-0.5} - \frac{1}{2}\Delta \frac{\partial E(\mathbf{q}_\ell)}{\partial \mathbf{q}}$
end for
Compute the acceptance probability $a = \min(1, e^{-c})$ where c is computed according to (22)
Set u to be a uniformly sampled random number between zero and one
if $u < a$ **then**
  return $\mathbf{q}_L$
else
  return $\mathbf{q}_0$
end if

tribution for each chain is the posterior $p(\mathbf{s}_t|, \mathbf{z}_{1:t})$. The final states of each chain are then taken to be fair samples from the posterior.

Choo and Fleet (2001) used an MCMC method known as Hybrid Monte Carlo. The Hybrid Monte Carlo algorithm (Algorithm 4) is an MCMC technique, based on ideas developed for molecular dynamics, which uses the gradient of the posterior to efficiently find high probability states. A single step begins by sampling a vector $\mathbf{p}_0 \in \mathbb{R}^m$ from a Gaussian distribution with mean zero and unit variance. Here m is the dimension of the state vector $\mathbf{q}_0$. The randomly drawn vector, known as the momentem, is then used to perform a simulation of the system of differential equations $\frac{\partial \mathbf{p}}{\partial \tau} = -\frac{\partial E(\mathbf{q})}{\partial \mathbf{q}}$ and $\frac{\partial \mathbf{q}}{\partial \tau} = \mathbf{p}$ where τ is an artificial time variable and $E(\mathbf{q}) = -\log \mathscr{P}(\mathbf{q})$. The simulation begins at $(\mathbf{q}_0, \mathbf{p}_0)$ and proceeds using a leapfrog step which is explicitly given in Algorithm 4. There, L is the number of steps to simulate for and Δ is a diagonal matrix whos entries specify the size of step to take in each dimension of the state vector $\mathbf{q}$. At the end of the simulation, the ending state of the physical simulation $\mathbf{q}_L$ is accepted with probability $a = \min(1, e^{-c})$ where

$$c = (E(\mathbf{q}_L) + \frac{1}{2}||\mathbf{p}_0||^2) - (E(\mathbf{q}_0) + \frac{1}{2}||\mathbf{p}_L||^2). \tag{22}$$

The specific form of the simulation procedure and the acceptance test at the end are designed such that $\mathscr{P}(\mathbf{q})$ is the stationary distribution of the transition distribution. The parameters of the algorithm, L and the diagonals of Δ are set by hand. As a rule of thumb Δ and L should be set so that roughly 75% of transitions are accepted. An important caveat is that the values of L and Δ cannot be set based on $\mathbf{q}$ and must remain constant throughout a simulation. For more information on Hybrid Monte Carlo see (Neal, 1993).

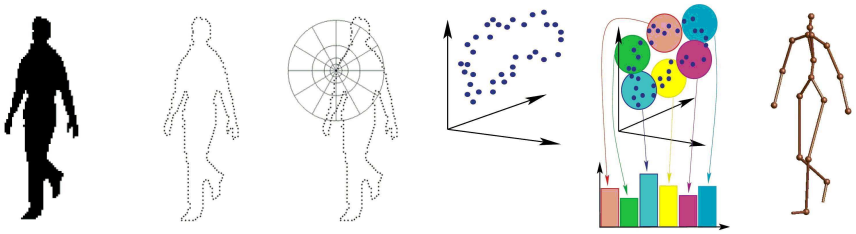


Fig. 11 Illustration of the simple discriminative model. The model introduced by Agarwal and Triggs (2006) is illustrated. From left to right the figure shows (1) silhouette image, (2) contour of the silhouette image, (3) shape context feature descriptor for a point on a contour, (4) a set of shape context descriptors in the high (60-dimensional) space, (5) a 100-dimensional vector quantized histogram of shape descriptors that is used to obtain (6) the pose of the person through linear regression. This figure is re-printed from (Agarwal and Triggs, 2006).

6 Initialization and Failure Recovery

The final issue we must address concerns initialization and the recovery from tracking failures. Because of the large number of unknown state variables one cannot assume that the filter can effectively search the entire state space without a good prior model (or initial guess). Fortunately, over the last few years, progress on the development of discriminative methods for detecting people and pose inference has been encouraging.

6.1 Introduction to Discriminative Methods for Pose Estimation

Discriminative approaches aim to recover pose directly from a set of measurements, usually through some form of regression applied to a set of measurements from a single frame. Discriminative techniques are typically learned from a set of training exemplars, $\mathscr{D} = \{(\mathbf{s}^{(i)}, \mathbf{z}^{(i)}) \sim p(\mathbf{s}, \mathbf{z}) | i = 1...N\}$), which are assumed to be fair samples from the joint distribution over states and measurements. The goal is to learn to predict an output for a given input. The inputs, $\mathbf{z} \in \mathbb{R}^M$, are generic image measurements, [4] and outputs $\mathbf{s} \in \mathbb{R}^N$, as above, represent the 3D poses of the body.

The simplest discriminative method is Nearest-Neighbor (NN) lookup (Howe, 2007; Mori and Malik, 2002), where, given a set of features observed in an image, the exemplar from the training database with the closest features is found, i.e., $k^* = \arg\min_k d(\tilde{\mathbf{z}}, \mathbf{z}^{(k)})$. The pose $\mathbf{s}^{(k^*)}$ for that exemplar is returned. The main challenge is to define a useful similarity measure $d(\cdot, \cdot)$, and a fast indexing scheme. One such approach was proposed by Shakhnarovich, Viola, and Darrell (2003). Unfortunately,

[4] For instance, histograms-of-oriented-gradients, vector quantized shape contexts, HMAX, spatial pyramids, vocabulary trees and so on. See Kanaujia, Sminchisescu, and Metaxas (2007a) for details.

Fig. 12 Discriminative Output. Pose estimation results obtained using the discriminative method introduced by Kanaujia, Sminchisescu, and Metaxas. This figure is re-printed from (Kanaujia, Sminchisescu, and Metaxas, 2007a).

this simple approach has three drawbacks: (1) large training sets are required, (2) all the training data must be stored and used for inference, and (3) it produces unimodal predications and hence ambiguities (or multi-modality) in image-to-pose mappings cannot be accounted for (e.g., see Sminchisescu, Kanaujia, Li, and Metaxas (2005)).

To address (1) and (2) a variety of global (e.g., Agarwal and Triggs (2006)) and local (e.g., Rosales and Sclaroff (2002)) parametric models have been proposed. These models learn a functional mapping from image features to 3D pose. While these methods have been demonstrated successfully on restricted domains, and with moderately large training sets, they do not provide one-to-many mappings, and therefore do not cope with multi-modality.

Multi-modal mappings have been formulated in a probabilistic setting, where one explicitly models multi-modal conditional distributions, $p(\mathbf{s}|\mathbf{z},\Theta)$, where Θ are parameters of the mapping, learned by maximizing the likelihood of the training data $\mathscr{D}$. One example is the conditional Mixture of Experts (cMoE) model introduced by Sminchisescu, Kanaujia, Li, and Metaxas (2005), which takes the form

$$p(\mathbf{s}|\mathbf{z},\Theta) = \sum_{k=0}^{K} g_k(\mathbf{z}|\Theta) e_k(\mathbf{s}|\mathbf{z},\Theta), \tag{23}$$

where K is the number of experts, g_k are positive gating functions which depend on the input features, and e_k are experts that predict the pose (e.g., kernel regressors). This model under various incarnations has been shown to work effectively with large datasets (Bo, Sminchisescu, Kanaujia, and Metaxas, 2008) and with partially labeled data[5] (Kanaujia, Sminchisescu, and Metaxas, 2007a).

The MoE model, however, still requires moderate to large amounts of training data to learn parameters of the gates and experts. Recently, methods that utilize an intermediate low dimensional embedding have been shown to be particularly effec-

[5] Since joint samples span a very high dimensional space, $\mathbb{R}^{N+M}$, obtaining a dense sampling of the joint space for the purposes of training is impractical. Hence, incorporating samples from marginal distributions $p(\mathbf{s})$ and $p(\mathbf{z})$ is of great practical benefit.

tive in dealing with little training data in this setting (e.g., Navaratnam, Fitzgibbon, and Cipolla (2007); Kanaujia, Sminchisescu, and Metaxas (2007b)). Alternatively, non-parametric approaches for handling large amounts of training data efficiently that can deal with multi-modal probabilistic predictions have also been recently proposed by Urtasun and Darrell (2008). Similar in spirit to the simple NN method above, their model uses the local neighborhood of the query to approximate a mixture of Gaussian Process (GP) regressors.

6.2 Discriminative Methods as Proposals for Inference

While discriminative methods are promising alternatives to generative inference, it is not clear that they will be capable of solving the pose estimation problem in a general sense. The inability to generalize to novel motions, deal with significant occlusions and a variety of other realistic phenomena seem to suggest that some generative component is required.

Fortunately, discriminative models can be incorporated within the generative setting in an elegant way. For example, multimodal conditional distributions that are the basis of most recent discriminative methods (e.g., Bo, Sminchisescu, Kanaujia, and Metaxas (2008); Navaratnam, Fitzgibbon, and Cipolla (2007); Sminchisescu, Kanaujia, Li, and Metaxas (2005); Urtasun and Darrell (2008)) can serve directly as proposal distributions (i.e., $q(\mathbf{s}_{t+1}|\mathbf{s}_{1:t}, \mathbf{z}_{1:t+1})$) to improve the sampling efficiency of the Sequential Monte Carlo methods discussed above. Some preliminary work on combining discriminative and generative methods in this and other ways has shown promise. It has been shown that discriminative components provide for effective initialization and the recovery from transient failures, and that generative components provide effective means to generalize and better fit image observations (Sigal, Balan, and Black, 2007; Sminchisescu, Kanaujia, Li, and Metaxas, 2005; Sminchisescu, Kanajujia, and Metaxas, 2006).

7 Conclusions

This chapter introduced the basic elements of modern approaches to pose tracking. Using the probabilistic formulation introduced in this chapter one should be able to build a state-of-the-art framework for tracking relatively simple motions of single isolated subjects in a compliant (possibly instrumented) environment. The more general problem of tracking arbitrary motion in monocular image sequences of unconstrained environments remains a challenging and active area of research. While many advances have been made, and the progress is promising, no system to date can robustly deal with all the complexities of recovering the human pose and motion in an entirely general setting.

While the need to track human motion from images is motivated by a variety of applications, currently there have been relatively few systems that utilize the image-based recovery of the articulated body pose for higher-level tasks or consumer applications. This to a large extent can be attributed to the complexity of obtaining an articulated pose in the first place. Nevertheless, a few very promising applications in biomechanics (Corazza, Muendermann, Chaudhari, Demattio, Cobelli, and Andriacchi, 2006) and human computer interfaces (Demirdjian, Ko, and Darrell, 2005; Ren, Shakhnarovich, Hodgins, Pfister, and Viola, 2005; Sukel, Catrambone, Essa, and Brostow, 2003) have been developed. The articulated pose has also proved useful as a front end for action recognition applications (Ning, Xu, Gong, and Huang, 2008). We believe that as the technologies for image-based recovery of articulated pose grows over the next years, so will the applications that utilize that technology.

References

Agarwal A, Triggs B (2006) Recovering 3d human pose from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1):44–58

Anguelov D, Srinivasan P, Koller D, Thrun S, Rodgers J, Davis J (2005) SCAPE: Shape Completion and Animation of People. ACM Transactions on Graphics 24(3):408–416

Balan A, Black MJ (2008) The naked truth: Estimating body shape under clothing. In: IEEE European Conference on Computer Vision

Balan AO, Sigal L, Black MJ, Davis JE, Haussecker HW (2007) Detailed human shape and pose from images. In: IEEE Conference on Computer Vision and Pattern Recognition

Barrow HG, Tenenbaum JM, Bolles RC, Wolf HC (1977) Parametric correspondence and chamfer matching: Two new techniques for image matching. In: International Joint Conference on Artificial Intelligence, pp 659–663

Bo L, Sminchisescu C, Kanaujia A, Metaxas D (2008) Fast algorithms for large scale conditional 3d prediction. In: IEEE Conference on Computer Vision and Pattern Recognition

Brubaker M, Fleet DJ (2008) The kneed walker for human pose tracking. In: IEEE Conference on Computer Vision and Pattern Recognition

Brubaker M, Fleet DJ, Hertzmann A (2007) Physics-based person tracking using simplified lower-body dynamics. In: IEEE Conference on Computer Vision and Pattern Recognition

Choo K, Fleet DJ (2001) People tracking using hybrid Monte Carlo filtering. In: IEEE International Conference on Computer Vision, vol II, pp 321–328

Corazza S, Muendermann L, Chaudhari A, Demattio T, Cobelli C, Andriacchi T (2006) A markerless motion capture system to study musculoskeletal biomechanics: visual hull and simulated annealing approach. Annals of Biomedical Engineering 34(6):1019–1029

de la Gorce M, Paragos N, Fleet DJ (2008) Model-based hand tracking with texture, shading and self-occlusions. In: IEEE Conference on Computer Vision and Pattern Recognition

Demirdjian D, Ko T, Darrell T (2005) Untethered gesture acquisition and recognition for virtual world manipulation. Virtual Reality 8(4):222–230

Deutscher J, Reid I (2005) Articulated body motion capture by stochastic search. International Journal of Computer Vision 61(2):185–205

Doucet A, Godsill S, Andrieu C (2000) On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and Computing 10(3):197–208

Felzenszwalb P, Huttenlocher DP (2005) Pictorial structures for object recognition. International Journal of Computer Vision 61(1):55–79

Forsyth DA, Ponce J (2003) Computer Vision: A Modern Approach. Prentice Hall

Forsyth DA, Arikan O, Ikemoto L, O'Brien J, Ramanan D (2006) Computational studies of human motion: Part 1, tracking and motion synthesis. Foundations and Trends in Computer Graphics and Vision 1(2&3):1–255

Gall J, Potthoff J, Schnorr C, Rosenhahn B, Seidel HP (2007) Interacting and annealing particle filters: Mathematics and a recipe for applications. Journal of Mathematical Imaging and Vision 28:1–18

Gavrila DM, Davis LS (1996) 3-D model-based tracking of humans in action: a multi-view approach. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 73–80

Gordon N, Salmond DJ, Smith AFM (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings Part F Radar and signal processing 140:107–113

Grassia FS (1998) Practical parameterization of rotations using the exponential map. Journal of Graphics Tools 3(3):29–48

Herda L, Urtasun R, Fua P (2005) Hierarchical implicit surface joint limits for human body tracking. Computer Vision and Image Understanding 99(2):189–209

Horprasert T, Harwood D, Davis L (1999) A statistical approach for real-time robust background subtraction and shadow detection. In: FRAME-RATE: Frame-rate applications, methods and experiences with regularly available technology and equipment

Howe N (2007) Silhouette lookup for monocular 3d pose tracking. Image and Vision Computing 25:331–341

Huttenlocher DP, Klanderman GA, Rucklidge WJ (1993) Comparing images using the Hausdorf distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(9):850–863

Isard M, Blake A (1998) CONDENSATION - conditional density propagation for visual tracking. International Journal of Computer Vision 29(1):5–28

Isard M, MacCormick J (2001) BraMBLe: a bayesian multiple-blob tracker. In: IEEE International Conference on Computer Vision, vol 2, pp 34–41

Jepson AD, Fleet DJ, El-Maraghi TF (2003) Robust Online Appearance Models for Visual Tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(25):1296–1311

Kakadiaris L, Metaxas D (2000) Model-based estimation of 3D human motion. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(12):1453–1459

Kanaujia A, Sminchisescu C, Metaxas D (2007a) Semi-supervised hierarchical models for 3d human pose reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition

Kanaujia A, Sminchisescu C, Metaxas D (2007b) Spectral latent variable models for perceptual inference. In: IEEE International Conference on Computer Vision

Kollnig H, Nagel HH (1997) 3d pose estimation by directly matching polyhedral models to gray value gradients. International Journal of Computer Vision 23(3):283–302

Kong A, Liu JS, Wong WH (1994) Sequential imputations and bayesian missing data problems. Journal of the American Statistical Association 89(425):278–288

Kuipers JB (2002) Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality. Princeton University Press

Lee CS, Elgammal A (2007) Modeling view and posture manifolds for tracking. In: IEEE International Conference on Computer Vision

Li R, Tian TP, Sclaroff S (2007) Simultaneous learning of non-linear manifold and dynamical models for high-dimensional time series. In: IEEE International Conference on Computer Vision

Metaxas D, Terzopoulos D (1993) Shape and nonrigid motion estimation through physics-based synthesis. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(6):580–591

Moeslund TB, Hilton A, Krüger V (2006) A survey of advances in vision-based human motion capture and analysis. Computer Vision and Image Understanding 104(2-3):90–126

Mori G, Malik J (2002) Estimating human body configurations using shape context matching. In: IEEE European Conference on Computer Vision, pp 666–680

Navaratnam R, Fitzgibbon A, Cipolla R (2007) The joint manifold model for semi-supervised multi-valued regression. In: IEEE International Conference on Computer Vision

Neal RM (1993) Probabilistic inference using markov chain monte carlo methods. Tech. Rep. CRG-TR-93-1, Department of Computer Science, University of Toronto

Neal RM (2001) Annealed importance sampling. Statistics and Computing 11:125–139

Nestares O, Fleet DJ (2001) Probabilistic tracking of motion boundaries with spatiotemporal predictions. In: IEEE Conference on Computer Vision and Pattern Recognition, vol II, pp 358–365

Ning H, Xu W, Gong Y, Huang TS (2008) Latent pose estimator for continuous action recognition. In: IEEE European Conference on Computer Vision

North B, Blake A (1997) Using expectation-maximisation to learn dynamical models from visual data. In: British Machine Vision Conference

Pavolvic V, Rehg J, Cham TJ, Murphy K (1999) A dynamic bayesian network approach to figure tracking using learned dynamic models. In: IEEE International Conference on Computer Vision, pp 94–101

Plankers R, Fua P (2001) Articulated soft objects for video-based body modeling. In: IEEE International Conference on Computer Vision, vol 1, pp 394–401

Poon E, Fleet DJ (2002) Hybrid Monte Carlo filtering: edge-based people tracking. In: Workshop on Motion and Video Computing, pp 151–158

Prati A, Mikic I, Trivedi MM, Cucchiara R (2003) Detecting moving shadows: Algorithms and evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(7):918–923

Ramanan D, Forsyth DA, Zisserman A (2007) Tracking people by learning their appearance. IEEE Transactions on Pattern Analysis and Machine Intelligence 29:65–81

Rehg J, Kanade T (1995) Model-based tracking of self-occluding articulated objects. In: IEEE International Conference on Computer Vision, pp 612–617

Ren L, Shakhnarovich G, Hodgins J, Pfister H, Viola P (2005) Learning silhouette features for control of human motion. ACM Transactions on Graphics 24(4):1303–1331

Rosales R, Sclaroff S (2002) Learning body pose via specialized maps. In: Advances in Neural Information Processing Systems

Rosenhahn B, Kersting U, Powel K, Seidel HP (2006) Cloth X-Ray: MoCap of people wearing textiles. In: Pattern Recognition, DAGM

Shakhnarovich G, Viola P, Darrell TJ (2003) Fast pose estimation with parameter-sensitive hashing. In: IEEE International Conference on Computer Vision, pp 750–757

Sidenbladh H, Black M, Fleet D (2000) Stochastic tracking of 3d human figures using 2d image motion. In: IEEE European Conference on Computer Vision, vol 2, pp 702–718

Sidenbladh H, Black MJ, Sigal L (2002) Implicit probabilistic models of human motion for synthesis and tracking. In: IEEE European Conference on Computer Vision, vol 1, pp 784–800

Sigal L, Black MJ (2006) Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition, vol 2, pp 2041–2048

Sigal L, Balan A, Black MJ (2007) Combined discriminative and generative articulated pose and non-rigid shape estimation. In: Advances in Neural Information Processing Systems

Sminchisescu C, Jepson A (2004) Generative modeling for continuous non-linearly embedded visual inference. In: International Conference on Machine Learning, pp 759–766

Sminchisescu C, Kanaujia A, Li Z, Metaxas D (2005) Discriminative density propagation for 3d human motion estimation. In: IEEE Conference on Computer Vision and Pattern Recognition, vol 1, pp 390–397

Sminchisescu C, Kanajujia A, Metaxas D (2006) Learning joint top-down and bottom-up processes for 3d visual inference. In: IEEE Conference on Computer Vision and Pattern Recognition, vol 2, pp 1743–1752

Stauffer C, Grimson W (1999) Adaptive background mixture models for real-time tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, vol 2, pp 246–252

Stenger BDR (2004) Model-based hand tracking using a hierarchical bayesian filter. PhD thesis, University of Cambridge

Sukel K, Catrambone R, Essa I, Brostow G (2003) Presenting movement in a computer-based dance tutor. International Journal of Human-Computer Interaction 15(3):433–452

Taylor CJ (2000) Reconstruction of articulated objects from point correspondences in a single uncalibrated image. Computer Vision and Image Understanding 80(10):349–363

Tomasi C, Kanade T (1991) Detection and tracking of point features. Tech. Rep. CMU-CS-91-132, Carnegie Mellon University

Urtasun R, Darrell T (2008) Local probabilistic regression for activity-independent human pose inference. In: IEEE Conference on Computer Vision and Pattern Recognition

Urtasun R, Fleet DJ, Hertzmann A, Fua P (2005) Priors for people tracking from small training sets. In: IEEE International Conference on Computer Vision, vol 1, pp 403–410

Urtasun R, Fleet DJ, Fua P (2006a) 3D people tracking with gaussian process dynamical models. In: IEEE Conference on Computer Vision and Pattern Recognition, vol 1, pp 238–245

Urtasun R, Fleet DJ, Fua P (2006b) Motion models for 3D people tracking. Computer Vision and Image Understanding 104(2-3):157–177

Vondrak M, Sigal L, Jenkins OC (2008) Physical simulation for probabilistic motion tracking. In: IEEE Conference on Computer Vision and Pattern Recognition

Wachter S, Nagel HH (1999) Tracking persons in monocular image sequences. Computer Vision and Image Understanding 74(3):174–192

Wang JM, Fleet DJ, Hertzmann A (2006) Gaussian process dynamical models. In: Advances in Neural Information Processing Systems 18, pp 1441–1448

Wren CR, Pentland A (1998) Dynamic models of human motion. In: IEEE International Conference on Automatic Face and Gesture Recognition, pp 22–27

Wren CR, Azarbayejani A, Darrell T, Pentland AP (1997) Pfinder: Real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(7):780–785