A FEATURE GUIDED PARTICLE FILTER FOR ROBUST HAND TRACKING

Matti-Antero Okkonen
VTT Technical Research Centre of Finland, P.O. Box 1100, FI-90571 Oulu, Finland

Janne Heikkilä, Matti Pietikäinen
Department of Electrical and Information Engineering, P.O. Box 4500, FI-90014, University of Oulu, Finland

Keywords: Particle filtering, hand tracking, importance sampling, adaptive color model.

Abstract: Particle filtering offers an interesting framework for visual tracking. Unlike the Kalman filter, particle filters can deal with non-linear and non-Gaussian problems, which makes them suitable for visual tracking in presence of real-life disturbance factors, such as background clutter and movement, fast and unpredictable object movement and unideal illumination conditions. This paper presents a robust hand tracking particle filter algorithm which exploits the principle of importance sampling with a novel proposal distribution. The proposal distribution is based on effectively calculated color blob features, propagating the particles robustly through time even in unideal conditions. In addition, a novel method for conditional color model adaptation is proposed. The experiments show that using these methods in the particle filtering framework enables hand tracking with fast movements under real world conditions.

1 INTRODUCTION

Visual hand tracking is an important part of the growing field of automatic human activity recognition. Hand gestures are a natural part of human communication and automatic interpretation of them would have numerous applications, for example in natural user interfaces, in automatic sign language recognition, in virtual reality and even in emotion recognition. In addition, visual tracking of hands offers a non-contact modality, where no special hardware is needed. However, in order to work in real environments, tracking algorithms must tackle convoluting factors such as fast and complex hand movements, changing object appearance, unideal illumination conditions and background movement and clutter.

Previous work on hand tracking can be divided by how the hand is modeled: Some approaches treat the hand as a deformable object and try to solve the parameters that define the hand's articulation, whereas other methods aim for bare trajectory information by estimating the spatial location of the hand through time. The earliest attempts for hand tracking used data gloves or visual markers for the task, but gradually the focus moved to purely visual based systems, such as the DigitEyes (Rehg and Kanade, 1993). Another important milestone in this field was the introduction of particle filtering for visual tracking, which was done by Isard and Blake with the Condensation algorithm (Isard and Blake, 1996). Particle filters are sophisticated statistical estimation techniques that can deal with non-linear and non-Gaussian problems, which makes them well suited for visual tracking. Many variations of the basic particle filtering algorithms have been proposed for both spatial and articulated hand tracking since the Condensation algorithm. More information about the earlier work on hand tracking can be found in (Pavlovic et al., 1997; Mahmoudi and Parviz, 2006).

In spatial hand tracking, a group of methods have augmented the basic particle filtering algorithms with an enhancement step, where the spatial locations of the particles are improved using some optimization technique. Pantrigo et al. have used the local search and the scatter search algorithms (Pantrigo et al., 2005b; Pantrigo et al., 2005a), whereas Shan et al. have utilized the Mean Shift algorithm (Shan et al., 2004). The last-mentioned method exploits mean shift with color and motion cues for all particles in

Okkonen M., Heikkilä J. and Pietikäinen M. (2008).
A FEATURE GUIDED PARTICLE FILTER FOR ROBUST HAND TRACKING.
In *Proceedings of the Third International Conference on Computer Vision Theory and Applications*, pages 368-374
DOI: 10.5220/0001078503680374

each time step. An optimization approach have also been proposed to articulated hand tracking by Lin et al. using the Nelder Mead algorithm (Lin et al., 2004). These optimization techniques aim for more efficient use of the particles by relocating them to spaces where they get bigger weights, thus reducing the variance of the state estimate and enabling tracking with fewer particles.

In this paper, a particle filtering algorithm for spatial hand tracking is proposed. The algorithm employs a feature based proposal distribution for optimal grouping of the particles through time. This way the particles can be propagated based on current observations rather than a predefined dynamic model and therefore there is no need for an additional indeterministic optimization scheme as in (Pantrigo et al., 2005b; Pantrigo et al., 2005a; Shan et al., 2004; Lin et al., 2004). Furthermore, the impoverishment of the particles can be avoided by adjusting the variance of the proposal distribution, which is not possible with the methods relying on the optimization schemes. The proposal distribution is built on color cue, since it is more perennial compared to motion and edge information that may not always be observable due to e.g. motion blur or lack of motion. In addition, a novel technique for color model adaptation is presented, which restrains the model to falsely adapt to background when tracking temporally fails, which is largely ignored by previous methods. The proposed method is referred as the Feature Guided Particle Filter (FGPF) troughout the paper.

The rest of the paper is organized as follows. Section 2 gives an overview about the theoretical basis of particle filtering and Section 3 explains the proposed method in detail. Section 4 reports the experimental results, and finally, discussion and conclusions are given in Section 5.

2 PARTICLE FILTERING

Particle filtering arises as an approximate solution to the sequential Bayesian estimation framework. The underlying problem is to estimate the pdf of hidden parameters $\mathbf{x}_t$ at discrete time steps, given a set of observations $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$.

2.1 Recursive Bayesian Filter

In the Bayesian approach the posteriori pdf $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ is obtained recursively in two steps. In the prediction step, the prior pdf is created by predicting the distribution from the previous step

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{x}_{t-1})d\mathbf{x}_{t-1}, \quad (1)$$

which is updated in the update step when the current observation is available

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{\int p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})}{p(\mathbf{z}_t|\mathbf{z}_{t-1})}. \quad (2)$$

Closed form solution exist to the problem in restricted cases. The Kalman filter, for example, restricts the model dynamics $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and the measurement density $p(\mathbf{z}_t|\mathbf{x}_t)$ to be linear and Gaussian. Particle filter algorithms solve the above problem via Monte Carlo simulations and rather than operating with analytic expressions the distributions are represented with a set of discrete samples of the parameter space (i.e. particles) $\{\mathbf{x}^i, w^i\}_{i=1}^{N_S}$, with associated weights. Although particle filters only approximate the solution, they can deal with non-linear and non-Gaussian system and measurement models. Especially the ability to deal with arbitrary distributions is an advantage when applying them to visual tracking in cluttered environments. A more detailed introduction to particle filtering can be found in (Arulampalam et al., 2002).

3 FEATURE GUIDED PARTICLE FILTER (FGPF)

This section gives a detailed description of the proposed method.

3.1 Object Model

The human hand is a complex object and modeling its deformations leads inevitably to high-dimensional models that require learning the model parameters (Rehg and Kanade, 1993; Isard and Blake, 1996). Since the goal here is to estimate only the location of the hand, it is modeled with rectangles for their generality and simplicity. The model consists of two concentric rectangles, R_c and R_m, for evaluating both the color and the motion cue, respectively. The dimensions of the model remain fixed throughout processing, which leaves the state of the model to be determined by the central coordinates

$$\mathbf{x} = (x_c, y_c)^T. \quad (3)$$

3.2 Measurement Model

The measurement model is based on a fusion of color and motion cues. Edge density is omitted here since it requires detailed information about the object's geometry and is easily degraded by motion blur.

Skin color forms a compact cluster in chromaticity space, which makes it a popular feature for hand tracking. In our approach, skin color is modeled with a two dimensional histogram in (r,g) space of Normalized Color Coordinates (NCC). This way the model can present arbitrary distributions based purely on chromatic values. Similar to (Pérez et al., 2004), the color likelihood of a particle is defined as

$$p(\mathbf{z}_c|\mathbf{x}) \propto \exp(-D_c^2/2\sigma_c^2), \quad (4)$$

where

$$D_c = 1 - \sum_{r=1}^{N_b}\sum_{g=1}^{N_b}\sqrt{P(r,g)C(r,g)} \quad (5)$$

is built on the Bhattacharyya distance between the object histogram $P(r,g)$ and the color reference histogram $C(r,g)$ in (r,g) color space.

The motion cue is computed by following the approach in (Shan et al., 2004) by evaluating the absolute difference of luminosity between successive frames $\Delta I_t = |I_t - I_{t-1}|$. Using this, a binary image I_m is computed, where each pixel $I_m(x,y)$ is assigned with 1 if the average difference in the 3×3 neighborhood of $\Delta I_t(x,y)$ exceeds a predetermined threshold T_m and 0 otherwise. The motion likelihood of a particle is defined as the color likelihood, where the motion distance is defined as

$$D_m = \frac{1}{A(R_m)}\sum_{x,y\in R_m} I_m(x,y). \quad (6)$$

$A(R_m)$ denotes the area of R_m in above equation.

Using the assumption that the color and motion cues are statistically independent, the measurement model can be seen as their product

$$p(\mathbf{z}|\mathbf{x}) \propto \exp(-D_c^2/2\sigma_c^2 - D_m^2/2\sigma_m^2). \quad (7)$$

3.3 The Proposal Distribution

The performance of a particle filtering algorithm is dependent on how the particles are distributed in the state space. Importance sampling is a technique used in Monte Carlo methods that aims for assigning the particles to spaces where the measurement density is high (Arulampalam et al., 2002). This improves efficiency and reduces the variance of the parameter estimate since particles with small weights would have negligible impact on the estimate. The idea is to draw the particles from a proposal distribution $q(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{z}_t)$ that can exploit current measurements $\mathbf{z}_t$ rather than using the predetermined prior $p(\mathbf{x}_t|\mathbf{x}_{t-1})$.

Here, a color based proposal function for efficient distribution of the particles is presented. Color is chosen because it is more permanent than motion and gradient cues, that can be easily lost due lack of motion or motion blur. Furthermore, the use of gradient information would require object shape information, which would complicate the system beyond the objectives of the spatial tracking. To make the color cue as discriminative as possible, the color model is created from a small portion of the object and is updated along the image sequence. Color based blob features have been earlier used in articulated hand tracking, for example, for evaluating the object likelihood (Bretzner et al., 2002). Color segmentation has also been used as a coarse level proposal distribution in (Isard and Blake, 1996).

To create the proposal function, first a color likelihood image I_c is created from the color model $C(r,g)$ by *histogram backprojection* (Swain and Ballard, 1991). The blobs are detected from the likelihood image following the method of (Bretzner and Lindeberg, 1998) by evaluating the squared Laplacian

$$\nabla^2 L(x,y) = \frac{\partial^2}{\partial x^2}L(x,y) + \frac{\partial^2}{\partial y^2}L(x,y). \quad (8)$$

where $L(x,y) = g(\cdot) * I_c(x,y)$ and $g(\cdot)$ is a Gaussian kernel. The partial derivatives $\frac{\partial^2}{\partial x^2}L(x,y)$ can be expressed as a convolution of $\frac{\partial^2}{\partial x^2}g(\cdot)$ with the likelihood image $I_c(x,y)$. Straightforward application of the convolution would be computationally demanding, which is avoided here by approximating the second order derivative Gaussian $\frac{\partial^2}{\partial x^2}g(\cdot)$ with *box filters*, as in the SURF descriptor (Leonardis et al., 2006). This way the convolution can be computed very fast using *integral images*. For detailed descriptions about the box filters and the integral images, we refer to (Leonardis et al., 2006) and (Viola and Jones, 2001), respectively.

As an illustration, a 9×9 box filter approximation of the second order Gaussian derivative $\frac{\partial^2}{\partial x^2}g(\cdot)$ is shown in Fig. 1.

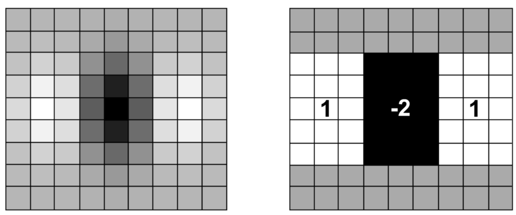

Figure 1: A cropped second order gaussian derivative $\frac{\partial^2}{\partial x^2}g(\cdot)$ and its box filter approximation.

However, in practice the filter size must be considerably larger for detecting hand sized blobs. Note that the computational burden of convolution is invariant to the size of the box filter.

The blobs are detected as local maxima in $-\nabla^2 L(x,y)$, yielding in a set of blob centers

$$\mathcal{B} = \{\mathbf{c}_b^i, \xi^i\}_{i=1}^{N_b} \tag{9}$$

corresponding to weights $\xi^i \propto -\nabla^2 L(\mathbf{c}_b^i)$. Points with weights smaller than a given threshold T_b are considered insignificant and are therefore excluded from the set.

The dynamics of the system is modeled with a second order autoregressive model

$$\mathbf{x}_t = 2\mathbf{x}_{t-1} - \mathbf{x}_{t-2} + \upsilon, \tag{10}$$

where υ equals to Gaussian noise component with zero mean and variance σ_d. The model is kept simple to achieve a level of generality, since human hand motion can be diverse and learned models are inevitably biased to their training data.

Finally, the proposal distribution is defined as

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}^j, \mathbf{x}_{t-2}^j, \mathcal{B}_t) \propto \sum_{i=1}^{N_b} \xi^i p(\mathbf{c}_b^i|\mathbf{x}_{t-1}{}^j, \mathbf{x}_{t-2}^j) \mathcal{N}(\mathbf{x}_t; \mathbf{c}_b^i, \sigma_p^2 I) \tag{11}$$

where $\mathcal{N}(\cdot)$ and I denote isotropic Gaussian distribution and identity matrix, respectively, and $p()$ stands for the dynamic model. Using blob features, the particles are distributed around skin colored areas. In addition, the Gaussians in (11) are weighted not only with the blob weights ξ^i, but also with the dynamic model to favor blobs that induce smoothness into the particle trajectories. Note that both the proposal distribution (11) and the measurement density (7) need to be defined only up to proportionality (Arulampalam et al., 2002). An illustration of the proposal distribution is given in Fig. 2.

3.4 Location Estimate and Color Model Adaptation

As stated in Section 2, the posterior pdf is not necessary unimodal. This situation arises often when the background contains clutter and additional movement. In such cases the location estimate calculated as the weighted mean of the particles is biased by the "outlier particles". To avoid this, the location estimate is calculated here as a weighted mean of a subset

$$\{\{\mathbf{x}', w'\} \in \{\mathbf{x}, w\} : ||\mathbf{x}' - \hat{\mathbf{x}}|| < R_e\}, \tag{12}$$

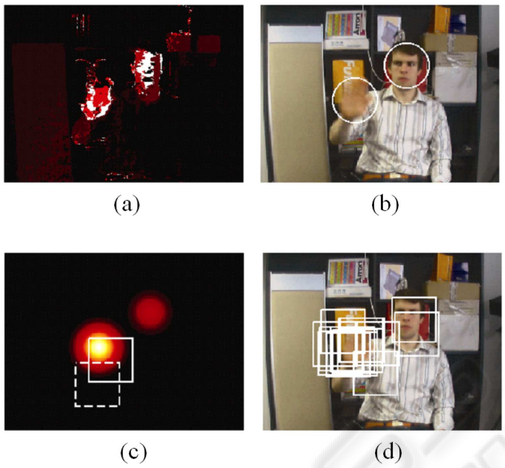

Figure 2: Illustration of the proposal distribution. (a) A color likelihood image I_c at time t and (b) the detected blobs, marked with circles. (c) The proposal distribution for an individual particle, where the continuous and the dashed rectangles mark the state of the particle at $t-1$ and $t-2$ respectively. (d) The resulting particle set.

where $\hat{\mathbf{x}}$ equals to the particle in the full set $\{\mathbf{x}, w\}$ with the biggest weight and R_e defines the radius of the area from which the subset is collected.

In each time step, a color histogram $H_t(r,g)$ is computed in the Normalized Color Coordinates from the pixels inside the smaller object rectangle R_c of the location estimate. The histograms are stored in a buffer of length N_{buff}, from which the reference color model $C_t(r,g)$ is composed as their mean, following smoothly the changes in the hand's appearance. If the location estimate should fail, the color model should not be adapted. To recognize such events, two steps are taken: In the first step, the candidate histogram $H_t(r,g)$ and the current reference model $C_t(r,g)$ are tested for similarity using the Bhattacharyya distance

$$D(H_t, C_t) = \sum_{r=1}^{N_b} \sum_{g=1}^{N_b} \sqrt{H_t(r,g) C_t(r,g)}. \tag{13}$$

The candidate is added to the buffer only if $D(H_t, C_t) > \tau_c$. An abrupt change between the current reference and the candidate is taken to be an indication of a false estimate. The second step is to evaluate the amount of ambiguity which each new histogram $H_t(r,g)$, that passes the first step, adds to the model. For every frame, a scatter measure for the likelihood image I_c is computed as

$$\mathcal{S}(I_c) = \frac{\sqrt{\mu_{20}'^2 + \mu_{02}'^2}}{X \cdot Y}, \tag{14}$$

where X and Y denote the dimension of the image. The moments $\mu'_{20} = \mu_{20}/\mu_{00}$ and $\mu'_{02} = \mu_{02}/\mu_{00}$ are constructed from central moments

$$\mu_{pq} = \sum_x \sum_y (x-\overline{x})^p (y-\overline{y})^q I_c(x,y), \qquad (15)$$

where $(\overline{x},\overline{y})$ is the center of gravity of the image. Since the size of the hand should not change too much between consecutive frames, a sharp increase in the scatter of the color likelihood image is considered to be caused by a false estimate. Thus, if $\mathcal{S}_t - \mathcal{S}_{t-1} > \tau_s$, the previously added histogram $H_{t-1}(r,g)$ is discarded from the buffer. This situation occurs, for example, when the model falsely adapts to an background object that meets the condition in the first step.

3.5 The Algorithm

According to the importance sampling theory, the particle weights must be augmented with a correction factor when sampling from the proposal distribution. As derived in (Arulampalam et al., 2002), the particle weight update equation then becomes

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i,\mathbf{x}_{t-2}^i)}{q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i,\mathbf{x}_{t-1}^i,\mathbf{z}_t)} p(\mathbf{z}_t|\mathbf{x}_t^i), \qquad (16)$$

where the factor $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i,\mathbf{x}_{t-2}^i)/q(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i,\mathbf{x}_{t-1}^i,\mathbf{z}_t)$ emphasizes particles toward the dynamic model. This factor is omitted here since the proposal distribution is considered to be a better representation of the true state than the generic dynamic model, in which case the factor would degrade the estimate calculated from the particles. This has been shown in our experiments.

The hand tracking algorithm used in the experiments is built on the Sampling Importance Resampling (SIR) algorithm and is presented in Table 1.

Table 1: Feature Guided Particle Filter (FGPF) for hand tracking.

To create the particle set $\{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_p}$, process each particle in the previous set $\{\mathbf{x}_{t-1}^i, w_{t-1}^i\}_{i=1}^{N_p}$ as follows:

1. Extract the feature set $\mathcal{B} = \{\mathbf{c}_b^i, \xi^i\}_{i=1}^{N_b}$
2. draw $\mathbf{x}_t^i \sim q(\mathbf{x}_t|\mathbf{x}_{t-1}^i, \mathbf{x}_{t-2}^i, \mathcal{B})$
3. update weight: $w_t^i \propto w_{t-1}^i p(\mathbf{z}_t|\mathbf{x}_t^i)$
4. Estimate the effective sample size $\hat{N_{eff}} = \left\{\sum_{i=1}^{N_p}(w_t^i)^2\right\}^{-1}$
5. If $\hat{N_{eff}} < T_{eff}$, draw $\mathbf{x}_t^{*i} \sim \{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_p}$ so that $p(\mathbf{x}_t^{*i} = \mathbf{x}_t^j) \propto w_t^j$ and replace $\{\mathbf{x}_t^i, w_t^i\} \leftarrow \{\mathbf{x}_t^{*i}, N_p^{-1}\}$
6. Estimate the location of the object as the mean of the subset defined in (12).

4 EXPERIMENTS

The experiments were made with a set of twelve test sequences, where a user moved his hand randomly. The sequences had 15 fps, a resolution of 320x240 and a mean length of 240 frames. Eight sequences were made by varying three parameters: background clutter (simple/cluttered), hand speed (slow/fast) and the presence of the user (only the hand and arm in the frames/user's upper torso in the background). Another four sequences were made varying the first two parameters, with the user and an additional person moving in the background. Furthermore, to simulate real-life incidents, the hand of the user occasionally moved outside of the camera's field of view for a few frames. The center of the palm was manually labeled as the reference point for each frame of the sequences.

For a comparison, the tests were performed also with the Mean Shift Embedded Particle Filter (MSEPF), implemented as presented in (Shan et al., 2004). The parameters for the tests were found manually for both methods, since global optimization over the test sequences would have been infeasible. Each parameteres was tested with few values and the optimal over all sequences was chosen. The selected parameters are presented in Table 2. For the evaluation, the average tracking rate was computed for both methods, which is defined as the proportion of frames where the estimate is within 20 pixels of the reference point. Since both methods are stochastic in nature, the tests were repeated 10 times for each sequence. Table 3 shows the results for each sequence parameter value, averaged over the others.

To verify the advantage of the presented proposal distribution, the experiments were also performed by replacing the proposal distribution with the dynamic model. This produced an overall tracking rate of 0.65, which is considerably lower than using the proposal distribution. In addition, tests were also carried out using the weighting equation (16), which yielded in an overall tracking rate of 0.90.

As the results show, the presented method outperforms the MSEPF with the given parameters. The tests showed that the Feature Guided Particle Filter was able to deal with convoluting factors, such as background movement and clutter with fast and diverse motion. Moreover, the proposed method was able to recover when the object was momentarily out of sight. These factors also produced the major differences between the two methods, whereas the tracking rates were relatively high for both methods with the less complicated sequences. The main shortcoming of MSEPF was its inability to recover when tracking was lost, for example when image features were mo-

Table 2: Parameters used in the experiments.

N_p, number of particles	30	30
σ_d^2, variance of the dynamic model	60	60
σ_m^2, variance of the measurement model	0.1	0.1
R_m, rectangle size for the motion cue	61×61	61×61
R_c, rectangle size for the color cue	11×11	11×11
Number of bins in color histograms	64	64
T_m, threshold for the motion cue	15	15
T_{eff} ’threshold for resampling	15	15
S_{bf}, size of the box filters	91×91	-
(τ_c, τ_s), thresholds for color model adaptation	(0.5,0.01)	-
σ_p^2, variance of the proposal distribution	70	-
R_e, the radius for the particle subset	50	-
k, constant for the velocity weight	-	0.07
Number of mean shift iterations	-	2

Table 3: Average tracking rates.

	FGPF	MSEPF
Hand speed (slow/fast)	0.96 / 0.91	0.89 / 0.67
Background (simple/cluttered)	0.93 / 0.95	0.79 / 0.76
User in the background (no/yes)	0.97 / 0.93	0.73 / 0.81
Additional person in the background (no/yes)	0.96 / 0.90	0.80 / 0.73
Average over all sequences	0.94	0.78

mentarily ambigious. Example frames of the tests are given in Fig. 3.

Figure 3: Example frames from the test set, where the continuous and the dashed rectangles mark the estimates of the FGPF and MSEPF, respectively (frames 207, 209, 212, 230).

5 CONCLUSIONS

In this paper a particle filter approach for hand tracking is proposed. The method uses computationally efficient color based blob features for effective propagation of the particles. Since color cue alone might be ambiguous, it is augmented with motion cue in the measurement model. Furthermore, a novel technique for conditional color model adaptation is presented, which makes the method invariant to momentarily faults in the location estimate. The experiments show that the method is able to track the hand in the presence of complicating factors, such as fast hand movements and background clutter and movement. For the future work, the method could be enhanced to track with a moving camera. At this stage, fast camera movements would degrade the motion cue and thus thus make the measurement model less accurate.

REFERENCES

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.

Bretzner, L., Laptev, I., and Lindeberg, T. (2002). Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. In *Proc. of FGR*, pages 423–428.

Bretzner, L. and Lindeberg, T. (1998). Feature tracking with automatic selection of spatial scales. *Computer Vision and Image Understanding: CVIU*, 71(3):385–392.

Isard, M. and Blake, A. (1996). Contour tracking by stochastic propagation of conditional density. In *Proc. of ECCV (1)*, pages 343–356.

Leonardis, A., Bischof, H., and Pinz, A. (2006). Surf: Speeded up robust features. In *Proc. of ECCV (1)*, volume 3951 of *Lecture Notes in Computer Science*. Springer.

Lin, J. Y., Wu, Y., and Huang, T. S. (2004). 3d model-based hand tracking using stochastic direct search method. In *Proc. of FGR*, pages 693–698.

Mahmoudi, F. and Parviz, M. (2006). Visual hand tracking algorithms. In *Geometric Modeling and Imaging–New Trends*, pages 228 – 232.

Pantrigo, J. J., Montemayor, A. S., and Cabido, R. (2005a). Scatter search particle filter for 2d real-time hands and face tracking. In *Proc. of ICIAP*, pages 953–960.

Pantrigo, J. J., Montemayor, A. S., and Sanchez, A. (2005b). Local search particle filter applied to human-computer interaction. In *Proc. of ISPA*, pages 279 – 284.

Pavlovic, V., Sharma, R., and Huang, T. S. (1997). Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695.

Pérez, P., Vermaak, J., and Blake, A. (2004). Data fusion for visual tracking with particles. *Proceedings of the IEEE (issue on State Estimation)*, 92:495 – 513.

Rehg, J. M. and Kanade, T. (1993). Digiteyes: Vision-based human hand tracking. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA.

Shan, C., Wei, Y., Tan, T., and Ojardias, F. (2004). Real time hand tracking by combining particle filtering and mean shift. In *Proc. of FGR*, pages 669–674.

Swain, M. J. and Ballard, D. H. (1991). Color indexing. *Int. J. Comput. Vision*, 7(1):11–32.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proc. of CVPR*.