# PARAMETRIC CONTOUR TRACKING USING UNSCENTED KALMAN FILTER

*Yunqiang Chen, Thomas Huang*

chenyq@ifp.uiuc.edu, huang@ifp.uiuc.edu
University of Illinois at Urbana-Champaign
Urbana, IL, USA 61801

*Yong Rui*

yongrui@microsoft.com
Microsoft Research
Redmond, WA 98052

## ABSTRACT

This paper presents an efficient method to integrate various spatial-temporal constraints to regularize the contour tracking. The global shape of the contour is represented in a parametric form. Based on the parametric shape prior, a causal smoothness constraint can be developed. The causality nature of the constraint allows us to do efficient probabilistic contour detection using the powerful Hidden Markov Model (HMM). The contour parameters are then updated according to the detected contour points and the object dynamics by an Unscented Kalman filter (UKF). Due to the comprehensive spatial-temporal constraints, the algorithm is very robust to severe distractions. Real-time performance is also achieved. To validate the efficacy and robustness of the proposed approach, we apply this approach to track people in bad illumination and cluttered environments and promising results are reported.

## 1. INTRODUCTION

Visual tracking has become more and more important. Real-time applications such as video surveillance, video conferencing and human-computer interface in virtual environment all require the ability to track moving objects. It is a very challenging task to do efficient and robust visual tracking in complex environments, especially in some virtual environments where illumination and background might change dramatically between frames.

Contour-based tracking methods [7, 11, 9, 4] have been proved to be a powerful tool for boundary delineation. To handle the observation noise, local spatial constraints (e.g. contour smoothness) are enforced during contour evolution. However, for tracking contours in cluttered environments, more constraints are necessary. For example, the parametric shape constraint can greatly reduce the solution space and object dynamics should be considered to take the temporal constraints (e.g. object dynamics) into account.

Blake and Isard [5] developed a sampling-based algorithm (CONDENSATION) to explore the parametric shape prior and object dynamics. The MAP result is achieved by propagating the conditional probability densities over time. However, this non-parametric representation (i.e. using a set of discrete samples) of the probability density function requires a large number of samples which grows exponentially with the dimension of the state space. Furthermore, the assumption that the observations are independent to each other could hamper the performance of the algorithm.

To deal with the problems in previous methods, we propose a novel contour tracking algorithm to combine the global shape prior, object dynamics and local spatial constraints to achieve robust contour tracking. The object contours are modeled by parametric shape, e.g. ellipse. The contour tracking is divided into two stages.

The first step is to find the contour points of the object from the noisy image. Unlike traditional snake model, we simplified the traditional non-causal smoothness constraint to a causal form based on the parametric shape prior. Therefore, a Hidden Markov Model (HMM) is designed to enforce this causal smoothness constraint when detecting contour points. A probabilistic solution can be obtained using efficient forward-backward algorithm.

The second step is to estimate the shape parameters based on the detected contour points and the object dynamics. Unlike the non-parametric methods (e.g. CONDENSATION [5]), we assumes uni-mode Gaussian distribution and resort to the efficient Kalman filtering technique. Because the detected contour points are non-linearly related to the contour parameters, the Unscented Kalman Filter (UKF) [6] is used to estimate the contour parameters over time.

The rest of the paper is organized as follows. In Section 2, we explain the HMM for the contour detection. In Section 3, the UKF is adapted to update the contour parameters through time. We test our algorithm with real sequences and report promising results in Section 4. Concluding remarks and future works are in Section 5.

## 2. CONTOUR DETECTION USING HMM

To detect contour in noisy images, the dependency between neighboring contour points, i.e. contour smoothness constraint, is exploit in the active contour model [7, 12, 9]. A common difficulty in these traditional methods is its non-causal smoothness constraint and the deficiency in recursively refining the contours in the 2D image plane [1]. Only the local minimum can be obtained instead of a probabilistic solution. In this section, we design a causal smoothness constraint based on parametric shape prior, which can be represented by a Hidden Markov Model. Hence, an efficient probabilistic contour detection can be achieved using the forward-backward algorithm.

### 2.1. Hidden Markov model

Due to the aperture problem, we can restrict the contour searching to a set of normal lines of the predicted contour (Figure 1) at no big loss. Let $\phi = 1, ..., M$, be the index of the normal lines and $\lambda = -N, ..., N$, be the index of the pixels along a normal line. To track the object is to find out the true contour points on all the normal lines. To detect the contour points accurately, we use HMM [10] to model the dependencies between neighboring normal lines (contour smoothness constraint).

The hidden states of the HMM are the true contour points on all normal lines, (denoted as $\mathbf{s} = \{s_1, ..., s_\phi, ..., s_M\}$). So, the number of states are the number of pixels on each normal line (i.e. $2N + 1$). Because the contours are signified by the sharp intensity changes, we do the edge detection along the normal lines. The observations, $\mathbf{O} = \{O_1, ..., O_\phi, ..., O_M\}$, are the edge detection results on each normal line $\phi$. The HMM is fully specified by the observation model $P(O_\phi|s_\phi)$, and the transition probability $p(s_\phi|s_{\phi-1})$, which we described as follows.

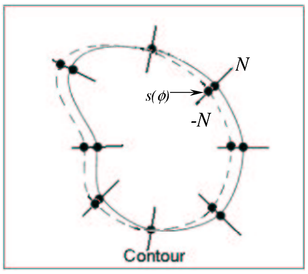


**Fig. 1**. Illustration of the 1D contour model: *At frame t, the solid curve is the predicted contour. The dashed curve is the true contour that we want to find. The measurements are collected along the $M$ normal lines of the predicted contour. $s(\phi)$ is the true contour point on the $\phi$th normal line. The true contour is found if we detect all the $s(\phi), \phi \in [1, M]$.*

### 2.2. Observation likelihood of HMM

The observation likelihood function based on the edge detection (i.e. $\mathbf{z}_\phi$) is similar to the one used in [5]. Because of noise and image clutter, there can be multiple edges along each normal line. Let $J$ be the number of detected edges ($\mathbf{z}_\phi = (z_1, z_2, ..., z_J)$). Of the $J$ edges, at most one is the true contour. With the assumption that the clutter is a Poisson process along the line with spatial density $\gamma$ and the true target measurement is normally distributed with standard deviation $\sigma_z$, we can obtain the observation likelihood model as follows:

$$p(\mathbf{z}_\phi|s_\phi = \lambda_\phi) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma_z q\gamma} \sum_{m=1}^{J} \exp(-\frac{(z_m - \lambda_\phi)^2}{2{\sigma_z}^2}) \tag{1}$$

where $q$ is the prior probability of the contour point not being detected by edge detection.

### 2.3. States transition probabilities

Another important component in HMM is the states transition probabilities which encode the spatial dependencies of the contour points on neighboring normal line. To achieve this, the smoothness constraint has to be represented in a causal form. In Figure 1, we can see when the normal lines are dense (e.g., 30 in our experiments), the true contour points on adjacent normal lines tend to have the same displacement from the predicted contour position (indexed as 0 on each normal line). This correlation is causal and can be captured by transition probabilities $p(s_\phi|s_{\phi-1})$:

$$p(s_\phi|s_{\phi-1}) = c \cdot e^{-(s_\phi - s_{\phi-1})^2/\sigma_s^2} \tag{2}$$

where $c$ is a normalization constant and $\sigma_s$ is a predefined constant that regulates the smoothness of the contour. This transition probability will penalize the roughness between neighboring contour points, hence resulting in a smooth contour.

### 2.4. Probabilistic contour detection

Given the observation sequence $\mathbf{O} = \{O_\phi, \phi \in [1, M]\}$ and the transition probabilities $a_{i,j} = p(s_{\phi+1} = j|s_\phi = i)$, we can get a probabilistic contour detection using the forward-backward algorithm. The ”forward probability distribution” and the ”backward probability distribution” [10] are defined as follow:

$$\alpha_\phi(s) = p(O_1, O_2, ..., O_\phi, s_\phi = s) \tag{3}$$

$$\beta_\phi(s) = p(O_{\phi+1}, O_{\phi+2}, ..., O_M, s_\phi = s) \tag{4}$$

After computing the forward and backward probability, we can compute the probability of each state at line $\phi$;

$$P(s_\phi = s|\mathbf{O}) = \frac{\alpha_\phi(s)\beta_\phi(s)}{\sum_u \alpha_\phi(u)\beta_\phi(u)}, \quad s \in [-N, N] \tag{5}$$

where $P(s_\phi = s|\mathbf{O})$ represents the probability of having the true contour point at $s$ on the normal line $\phi$. Because

Kalman filter can only handle uni-mode Gaussian distribution, we then estimate the expected position and the variance of the true contour point on each normal line:

$$y_t^\phi = \sum_{s_\phi=-N}^{N} s * P(s_\phi|\mathbf{O}), \quad \sigma_{y^\phi} = \sum_{s_\phi=-N}^{N} s^2 * P(s_\phi|\mathbf{O}) \tag{6}$$

which are then used in the UKF as the measurements to estimate the contour parameters in the next section.

## 3. TRACKING WITH OBJECT DYNAMICS

To increase the robustness of tracking, we also impose the parametric shape prior and the object dynamics. The shape prior greatly reduces the solution space. In our experiment, we use ellipse to approximate the human's heads. For the contour point on each normal line $\phi$ $(x_\phi, y_\phi)$, we have:

$$ax_\phi^2 + by_\phi^2 + cx_\phi y_\phi + dx_\phi + ey_\phi - 1 = 0 \tag{7}$$

Let vector $X_t = [a, b, c, d, e]^T$ denotes the parameters of the contour at time $t$. The true contour point on each normal line is the intersection of the normal line with the parametric contour $X_t$. We denote it by vector $Y_t$. Considering the temporal correlations, the image formation process can be formulated as follows:

$$X_t = f(X_{t-1}, m_{t-1}) \tag{8}$$

$$Y_t = h(u_t, X_t, n_t) \tag{9}$$

where $f(.)$ is the system dynamics, $h(.)$ is the observation model (i.e. how are the contour points related to the current contour parameters), $u_t$ is the system input, $m_t$ and $n_t$ are the process noise and observation noise, respectively.

In this paper, we adopt the Langevin process to model the object dynamics [13]:

$$\begin{bmatrix} \mathbf{X_t} \\ \dot{\mathbf{X}}_\mathbf{t} \end{bmatrix} = \begin{bmatrix} 1 & \tau \\ 0 & a \end{bmatrix} \begin{bmatrix} \mathbf{X_{t-1}} \\ \dot{\mathbf{X}}_\mathbf{t-1} \end{bmatrix} + \begin{bmatrix} 0 \\ b \end{bmatrix} m_t \tag{10}$$

where $a = \exp(-\beta_\theta \tau)$, $b = \bar{v}\sqrt{1-a^2}$. $\beta_\theta$ is the rate constant, $m$ is a thermal excitation process drawn from Gaussian distribution $N(0, Q)$, $\tau$ is the discretization time step and $\bar{v}$ is the steady-state root-mean-square velocity.

Because the observation model $h(.)$ is non-linear, we have to resort to non-linear estimation technique such as extended Kalman filter. The UKF is proposed as an alternative in [6, 8], which is provably superior to the EKF. The UKF does not need to explicitly calculate the Jacobians or Hessians. Therefore, the UKF not only outperforms the EKF in accuracy (second order approximation vs. first order approximation), but also is computationally efficient. Its superior performance has been demonstrated in many applications [6, 8]. Also different from the random sampling methods such as particle filtering, only a small number of carefully chosen sample points are propagated in each estimation step, which provide a compact parameterization of the underlying distribution. Hence it is much more efficient than the sampling methods. The UKF can be summarized as follows (for more details, please refer to [6]:

1. Select $2n+1$ sigma points $X_{t|t}$ according to:

$$X_{l,t|t} = \begin{cases} \hat{X}_{t|t} & l = 0 \\ \hat{X}_{t|t} - \sigma_t^l & l = 1, ..., n \\ \hat{X}_{t|t} + \sigma_t^l & l = n+1, ..., 2n \end{cases} \tag{11}$$

where $\sigma_t^l$ is the $l$th column of the matrix $\sqrt{n\Sigma_X(t|t)}$

2. Compute $X_{t+1|t}$ by applying the system dynamics equation (8) to $X_{t|t}$.

3. Compute the predicted state $\hat{X}_{t+1|t}$ (and the error covariance matrix)

$$\hat{X}_{t+1|t} = \frac{1}{2n+1}\sum_{l=0}^{2n} X_{l,t+1|t} \tag{12}$$

4. Compute $Y_{t+1|t}$ by applying the observation equation (9) to $X_{t+1|t}$.

5. Compute the predicted observation $\hat{Y}_{t+1|t}$ as

$$\hat{Y}_{t+1|t} = \frac{1}{2n+1}\sum_{l=0}^{2n} Y_{l,t+1|t} \tag{13}$$

6. Compute the innovation $v_{t+1} = Y_{t+1} - \hat{Y}_{t+1|t}$ from the measurement $Y_{t+1} = [y_{t+1}^0, ..., y_{t+1}^\phi, ..., y_{t+1}^M]^T$ (obtained from equation (6)) and the predicted observation $\hat{Y}_{t+1|t}$.

7. Update the Kalman gain matrix $K_{t+1}$.

8. Update the estimate of the state vector (and the error covariance matrix)

$$\hat{X}_{t+1|t+1} = \hat{X}_{t+1|t} + K_{t+1}v_{t+1} \tag{14}$$

## 4. EXPERIMENTS

To validate the efficacy and robustness of the proposed algorithm, we test our algorithm in a cluttered office sequence [2]. There are 499 frames in this sequence, with 30 frames per second. We use gray value only. Note that the blinds and the door (sharp edges and cluster) impose a great challenge to the visual tracking algorithms. Also note that the sequence was captured by a pan/tilt/zoom camera that moves


**Fig. 2**. Tracking with traditional contour model. The contour tracking results are severely distracted by the sharp edges on the background.


**Fig. 3**. Tracking with our new contour model. The tracker survives all the severe distractions on the background.

all the time. For the traditional contour tracking without object dynamics, the contour can be easily distracted by the cluttered background. The tracking results of four frames are shown in Figure 2.

When the person moved close to the door, the sharp edge along the door greatly distracted the tracking. The edge is smooth and strong – local smoothness constraints alone are not sufficient to avoid the distraction. Our new tracking algorithm survives all the severe distractions by integrating the shape prior and object dynamics.

## 5. CONCLUSION

In this paper, we present a real-time parametric contour tracking algorithm. Two main contributions are made to improve the efficiency and robustness of the contour tracking. First, the local spatial constraint (e.g. contour smoothness) is represented in a causal form based on the parametric shape prior. Hence, it can be enforced by a Hidden Markov model and efficient probabilistic contour detection is achieved. Second, to handle the non-linearity between the detected contour points and the contour parameters, an UKF is used to update the contour parameters according to the object dynamics. Because the efficiency of both the HMM and the UKF, the algorithm can run comfortably real-time on a PIII 833 computer.

## 6. REFERENCES


[1] A. Amini, T. Weymouth, and R. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Trans. on Pattern Analysis & Machine Intell.*, 12(9):855–867, September 1990.

[2] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *Proc. IEEE Int'l Conf. on Comput. Vis. and Patt. Recog.*, pages 232–237, 1998.

[3] Y. Chen, T. S. Huang, and Y. Rui. Optimal radial contour tracking by dynamic programming. In *Proc. IEEE Int'l Conf. on Image Processing*, 2001.

[4] C. Chesnaud, P. Refregier, and V. Boulet. Statistical region snake-based segmentation adapted to different physical noise models. *IEEE Trans. on Pattern Analysis & Machine Intell.*, 21(11):1145–1157, November 1999.

[5] M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):5–28, 1998.

[6] S. J. Julier and J. K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. Technical report RRG, Dept. of Engineering Science, University of Oxford.

[7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. Computer Vision*, 1(4):321–331, 1988.

[8] R. Merwe, A. Doucet, N. Freitas, and E. Wan. The unscented particle filter. Technical report CUED/F-INFENG/TR 380, Cambridge University Engineering Department, August 2000.

[9] N. Peterfreund. Robust tracking of position and velocity with kalman snakes. *IEEE Trans. on Pattern Analysis & Machine Intell.*, 21(6):564–569, June 1999.

[10] L. R. Rabiner and B. H. Juang. An introduction to hidden Markov models. *IEEE ASSP Mag.*, pages 4–15, January 1986.

[11] D. Terzopoulos and R. Szeliski. Tracking with kalman snakes. In A. Blake and A. Yuille, editors, *Active Vision*, pages 3–20. MIT Press, 1992.

[12] D. Terzopoulos and R. Szeliski. Tracking with kalman snakes. In *Active Vision*, pages 3–20, 1992.

[13] J. Vermaak and A. Blake. Nonlinear filtering for speaker tracking in noisy and reverberant environments. In *Proc. IEEE Int'l Conf. Acoustic Speech Signal Processing*, 2000.