Paper

Dynamic Color Tracking Based on Probabilistic Data Association

Xin Lu* Student Member
Shunichiro Oe** Member

This paper presents a color tracking method based on probabilistic data association in order to resolve difficult and complicated visual tracking problem, such as a changing of target's representation, a clutter of environments and an interaction of target and camera. Because the probabilistic data association is flexible and suitable for ambiguous and missing data, which generates the difficulties of visual tracking, some methods of probabilistic data association could be combined and applied in this tracking method to find the solutions of these difficulties. Due to using sequential Monte Carlo framework, this tracking method is applied to tracking of changeful target by handling the related information between every frame in image sequences. In order to improve tracking accuracy, this method utilizes factorized sampling algorithm to express target characters as sample-set. Moreover, this method benefits from HSV color model and captures the color natures of object like human to enhance the color-sensing capability of computer. Hence, this method could be considered as self-learning system and imitate the based human vision function – tracking. The tracking system applying this method is implemented in real-time at around 15Hz with 640×480 pixels image. The results show that the self-learning and real-time system is able to track a target robustly with enough accuracy and automatically control the camera's pan, tilt and zoom to remain the object centered in the field of vision.

Keywords: visual tracking, sequential Monte Carlo, factorized sampling, HSV

1. Introduction

Visual tracking could be treated as target state representation and target state inference problem in an image sequences. Moreover,in cluttered and dynamic environments the better probabilities of accurate tracking depend on richer representation and more robust inference. The target state representation could be considered as color segmentation, contour detection and position mark. And the target state inference could be treated as an evaluation from old states to new one in fuzzy logic at every step of an image sequence. Hence, visual tracking is based on pattern recognition and probability theory. Real-time tracking in reality is difficult because there are much interference in environments and changes of target. Sometimes visual tracking even cannot be implemented due to that the successive inference of target states are interrupted and the unique representation of target doesn't exist. For example, when two very similar objects (the one is target, the other is interference) disappear and emerge simultaneously, the visual tracking could not continue. In fact human can not distinguish the target in this situation. This situation doesn't conform to the basic criteria of tracking. That is the present state of target must be predictable by previous one or matchable with template. Usually two factors (i.e., continuity and uniqueness) determine the success of prediction and matching. The continuity means the changes of target state must be continuous (little by little, step by step) and can be recognized by tracking system or human. The uniqueness means the matching result using template must be unique and can be represented in view. In this assumed situation that two very similar objects exist and one of them will be tracked, the continuity denotes that the two objects can move freely and at least one of them must be in the view. The uniqueness denotes that the two targets must have differences between each other, like color, shape, motion and so on. The conclusion is that when one of the two factors (continuity and uniqueness) exits, the accurate tracking can be remained for long time.

In visual tracking process, target representation and inference are two of the most important elements. Target representation consists of color distribution, shape distribution etc.. Many tracking algorithms assume fixed color distribution [(1)~(3)] for the target to enable efficient color segmentation. But in practice they are often invalid. therefore some methods [(4)~(6)] are used to replace assuming fixed color representation, in which a Gaussian is applied to represent both color and motion parameters. In shape representation, besides the Sobel method [(7)], a snake method as the projection of a continuous contour lying on a smooth surface onto the image has been presented by references [(8)(9)]. In order to provide a more constrained representation of target, the approaches [(1)(5)(13)] ap-

* Department of Information Science and Intelligent Systems, Faculty of Engineering, University of Tokushima
2-1, Minamijosanjima, Tokushima, Japan 770-8506
** Center for Advanced Information Technology, University of Tokushima
2-1, Minamijosanjima, Tokushima, Japan 770-8506

ply both shape and color distributions of target. In inference aspect, some methods [10] build target templates to predict and match the representation in advance. The Kalman filtering [11] has given a classical hypothesis generating under Gaussian assumption. Because in clutter which causes the target to be multi-model and non-Gaussian, CONDENSATION [12](factorized sampling)and ICONDENSATION [13](important factorized sampling) have been presented.

In this paper we present a new color tracking based on probabilistic data association. It is based on tracking criteria to build a sample-set representation and multi-inference model and applied to gradual changing targets. In cluttered and complicated environments, this approach expresses enough efficiency and accuracy.

Section **2.** describes the sequential Monte Carlo framework specially used in field of tracking. A factorized sampling algorithm is presented in Section **3.** In Section **4.** and **5.** we use the dynamical model and the HSV color model to enhance the tracking accuracy. There is tracking procedure in details in Section **6.** and the experiments results are shown in Section **7.**

2. Sequential Monte Carlo Framework

Tracking model could be taken as a graphic model that is a marriage between probability theory and graph theory. Usually the dynamic tracking model is assumed to be a sequential Monte Carlo framework. The sequential Monte Carlo framework must be set out in terms of discrete time slice t. The state of target at time slice t is $\mathbf{x}_t$ and its history is $\mathbf{X}_t = \{\mathbf{x}_1, ..., \mathbf{x}_t\}$. Similarly the state of observation at time slice t is $\mathbf{z}_t$ and its history is $\mathbf{Z}_t = \{\mathbf{z}_1, ..., \mathbf{z}_t\}$. In this tracking model the new target state $\mathbf{x}_t$ depends on the previous proceeding state $\mathbf{x}_{t-1}$ and independent of the earlier ones in Fig.1. So we can get

$$p(\mathbf{x}_t|\mathbf{X}_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}) \quad \cdots\cdots\cdots\cdots\cdots\cdots (1)$$

That is expressed the dynamic tracking model framework. Also the new observation is independent of previous states $\mathbf{X}_{t-1}$ and previous observations $\mathbf{Z}_{t-1}$. So that

$$p(\mathbf{z}_t|\mathbf{X}_t, \mathbf{Z}_{t-1}) = p(\mathbf{z}_t|\mathbf{x}_t) \cdots\cdots\cdots\cdots\cdots\cdots (2)$$

The prior $p(\mathbf{x}_t|\mathbf{Z}_{t-1})$ is actually a prediction taken from the posterior $p(\mathbf{x}_{t-1}|\mathbf{Z}_{t-1})$ at the previous time $t-1$. So that

$$p(\mathbf{x}_t|\mathbf{Z}_t) = Cp(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{Z}_{t-1}) \cdots\cdots\cdots\cdots (3)$$

$$p(\mathbf{x}_t|\mathbf{Z}_{t-1}) = \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{Z}_{t-1}) \cdots (4)$$

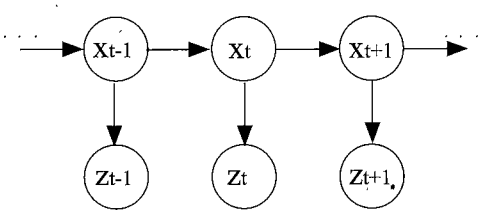


Fig. 1. A sequential Monte Carlo framework represents dynamic tracking.

and C is constant independent of $\mathbf{x}_t$. They are represented the propagations of state density at time t, where $p(\mathbf{z}_t|\mathbf{x}_t)$ denotes the measurement probability. Figure 1 shows the graphic model of tracking.

3. Factorized Sampling Algorithm

To improve tracking accuracy, the factorized sampling algorithm [12] is designed to address more general situations of target in sequential Monte Carlo framework. It has the noticeable feature that it is a considerably simpler algorithm than the Kalman filter. Despite random sampling is often considered to be computationally inefficient, the factorized sampling algorithm could run in near real-time by control of samples quantity. The reason of robustness is that tracking over time can maintain relatively robust distributions at successive time-steps, and self-learning model of color or shape is built at the same time.

By Bayesian rule, the posterior density $p(\mathbf{x}|\mathbf{z})$ could be represented as

$$p(\mathbf{x}|\mathbf{z}) = Cp(\mathbf{z}|\mathbf{x})p(\mathbf{x}) = \frac{p(\mathbf{z}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{z})} \quad \cdots\cdots\cdots (5)$$

where C is $1/p(\mathbf{z})$ independent of $\mathbf{x}$. By the factorized sampling algorithm a random variant $\mathbf{s}$ could be created from $p(\mathbf{x})$ to approximates the posterior $p(\mathbf{x}|\mathbf{z})$. Then the set of samples $\{\mathbf{s}^{(n)}, \pi^{(n)}\}$ could be generated from $p(\mathbf{x})$ with index $n \in \{1, ..., N\}$, where

$$\pi^{(n)} = \frac{p_z(\mathbf{s}^{(n)})}{\sum_{i=1}^{N} p_z(\mathbf{s}^{(i)})} \; where \; p_z(\mathbf{x}) = p(\mathbf{z}|\mathbf{x}) \cdots (6)$$

and the mean posteriors $\widetilde{p}(\mathbf{x}|\mathbf{z})$ can be generated from the set of samples $\{\mathbf{s}^{(n)}, \pi^{(n)}\}$. The more samples are selected, the more accurate result could be obtained.

$$p(\mathbf{x}|\mathbf{z}) \approx \widetilde{p}(\mathbf{x}|\mathbf{z}) = \frac{\sum_{n=1}^{N} \mathbf{s}^{(n)} p_z(\mathbf{s}^{(n)})}{\sum_{n=1}^{N} p_z(\mathbf{s}^{(n)})} \quad \cdots\cdots\cdots (7)$$

In CONDENSATION algorithm, the set of samples at time t denoted $\{\mathbf{s}_t^{(n)}, \pi_t^{(n)}, n = 1, ...N\}$ are drawn from prediction prior $p(\mathbf{x}_t|\mathbf{Z}_{t-1})$. So the prior could be replaced by the dynamical model $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and the set of samples $\{\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)}, n = 1, ...N\}$ at time $t-1$ of $p(\mathbf{x}_{t-1}|\mathbf{Z}_{t-1})$.

However, in fact the set of samples are difficult to be obtained from the prediction prior $p(\mathbf{x}_t|\mathbf{Z}_{t-1})$. So an important function $g(\mathbf{x})$ is generated and applied to select the samples. $g(\mathbf{x})$ means that which areas of state contain most posterior information. Generally there are some samples existing in these areas and the mean value of all the samples could very approximate the target. The method named importance sampling improves the efficiency of factorized sampling. The sample weight could be written as

$$\pi_t^{(n)} = \frac{f_t(\mathbf{s}_t^{(n)})}{g_t(\mathbf{s}_t^{(n)})} p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t^{(n)}) \quad \cdots\cdots\cdots\cdots (8)$$

$$f_t(\mathbf{s}_t^{(n)}) = \sum_{j=1}^{N} \pi_{t-1}^{(j)} p(\mathbf{x}_t = \mathbf{s}_t^{(n)}|\mathbf{x}_{t-1} = \mathbf{s}_{t-1}^{(j)}) \cdots (9)$$

In ICONDENSATION algorithm, some samples generated from standard factorized sampling $f(\mathbf{x})$ and some from important factorized sampling $g(\mathbf{x})$. We consider $f(\mathbf{x})$ to be equal with $g(\mathbf{x})$ in order to gain the fastest computation velocity in our approach.

The Fig.2 shows one time-step of factorized sampling algorithm. In the figure one blob signifies one sample $\mathbf{s}^{(n)}$ with its weight $\pi^{(n)}$. The algorithm includes three steps. The first step is stripping, which denotes dividing $\mathbf{s}_{t-1}^{(n)}$ from $\pi_{t-1}^{(n)}$ and obtaining sample set $\{\mathbf{s}_{t-1}^{(n)}\}$ without its weights $\{\pi_{t-1}^{(n)}\}$ in old state $\mathbf{x}_{t-1}$ for time-step $t-1$. The second step is drift/diffusion. Drift and diffusion separately express deterministic component and stochastic component in propagation $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ from old state $\mathbf{x}_{t-1}$ to new state $\mathbf{x}_t$. And the last step is measurement, which means generating weights $\{\pi_t^{(n)}\}$ from the observation density $p(\mathbf{z}_t|\mathbf{x}_t)$ to obtain the new sample-set $\{\mathbf{s}_t^{(n)}, \pi_t^{(n)}\}$ of new state $\mathbf{x}_t$ for time-step t. Section 4 and Section 5 define the models about $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $p(\mathbf{z}_t|\mathbf{x}_t)$. The complete procedure will be summarized in Section 6.

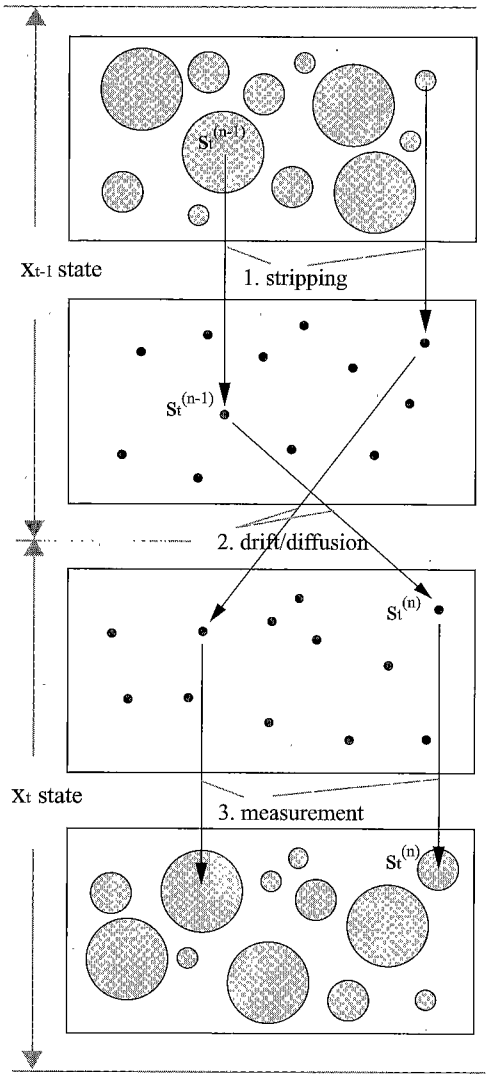


Fig. 2. One time-step in our factorized sampling algorithm.

4. Dynamical Model

The shape of tracking region is fixed by the definition of window W. It could be a rectangle or an ellipse in [2] [4]. Our approach doesn't restrict the type of shapes. We can use more complicated hand-drawn or learned regions in cases. In any case, tracking could amount to estimating the parameters of the transform in each frame to be used in W. In shape transform consideration, if the tracking region has enough characteristics of color information, the choice of a simple shape seems appropriate. Thus we consider the location $\mathbf{d} = (dx, dy)$ in the image coordinate system and the scale $\mathbf{e} = (ex, ey)$ as the estimative hidden variables as in [2] [4].

The second-order auto-regressive process (ARP) [14] is selected to calculate these parameters. Consistent with the first-order formalism described in the previous section, we define the state at time t as $\mathbf{s}_t = (\mathbf{d}_t, \mathbf{e}_t) = (dx_t, dy_t, ex_t, ey_t)$.Then dynamical model is assumed as

$$\mathbf{s}_{t+1} = P\mathbf{s}_t + Q\mathbf{s}_{t-1} + U\mathbf{v}_t + T, \mathbf{v}_t \sim \mathcal{N}(0, \Sigma). \quad \cdots\cdots\cdots\cdots\cdots\cdots (10)$$

where $\mathbf{v}_t$ are independent vectors of independent standard normal variables. They could be considered as Gaussian noises drawn from $\mathcal{N}(0, \Sigma)$, P and Q are matrices representing the deterministic components of the dynamical model respectively. U is metric representing stochastic component of the dynamic model. T is a fixed offset. They could be learned from a set of representative sequence where previous tracking results that have been obtained in some way. The specification of ARP in detail is described in appendix. We utilize a special model composed of the four relative dynamical matrices on dx_t, dy_t, ex_t and ey_t, and their respective standard deviations are 5 pixel/frame, 5 pixel/frame, 0.1 frame$^{-1/2}$ and 0.1 frame$^{-1/2}$.

5. HSV Color Model

Hue The color type (such as red, blue, or yellow). Measured in values of 0-360 by the central tendency of its wavelength

Saturation The 'intensity' of the color (or how much greyness is present). Measured in values of 0-100% by the amplitude of the wavelength

Value The brightness of the color. Measured in values of 0-100% by the spread of the wavelength

The Hue Saturation Value (or HSV) model defines a color space in terms of three constituent components; hue, saturation, and value. HSV is used in color progressions and is a non-linear transformation of the RGB color space.

Artists sometimes prefer to use the HSV color model over alternative models such as RGB and CMY, because of its similarities to the way humans tend to perceive color. RGB and CMY are additive and subtractive

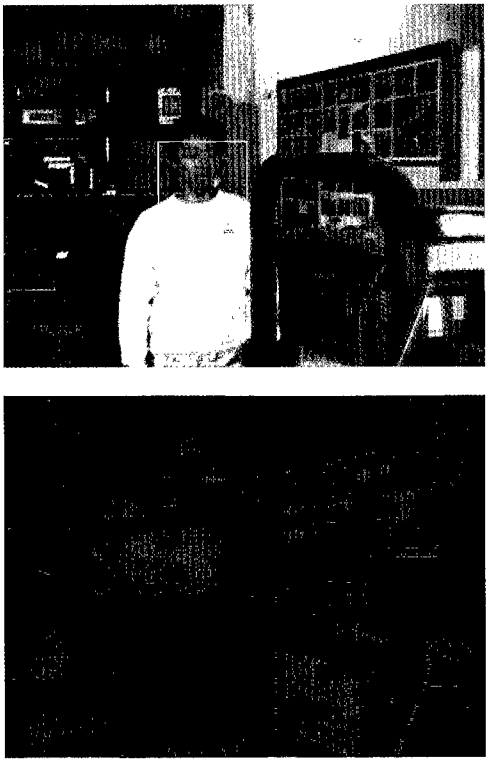


Fig. 3. A image conversion from RGB color space to HSV color space. the above image is in RGB color space, the following image is in HSV color space.

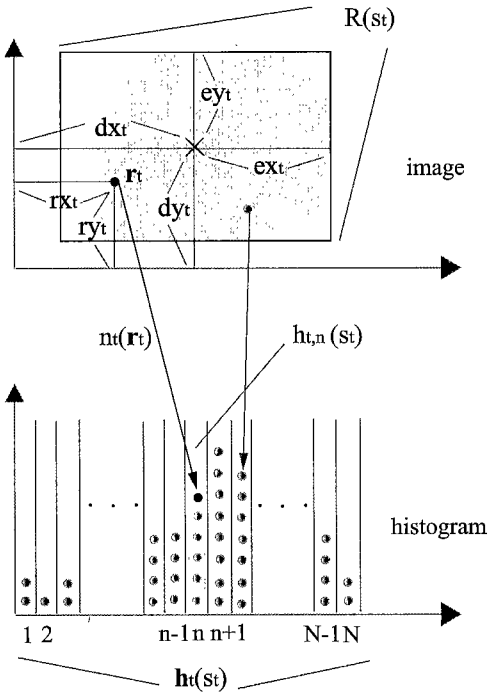


Fig. 4. The color histogram $\mathbf{h}_t(\mathbf{s}_t)$ in time-step t.

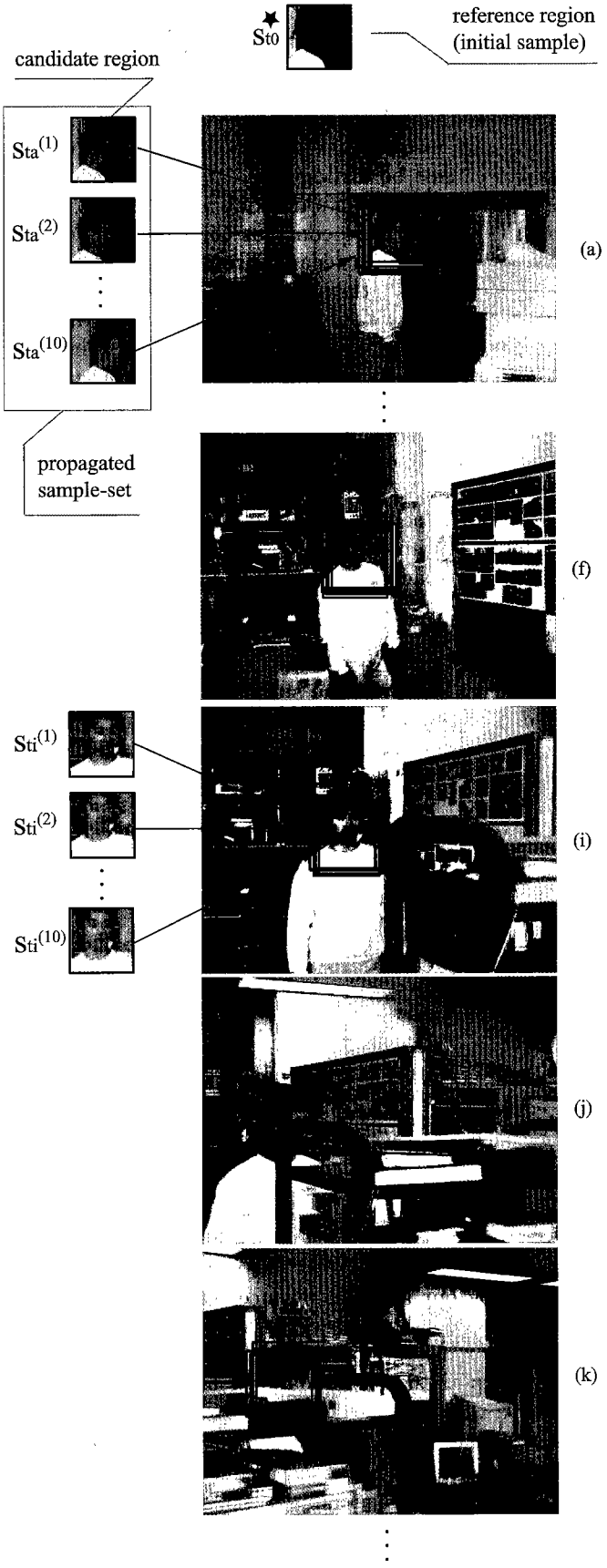


Fig. 5. Tracking multiple regions in frames.

models, respectively, defining color in terms of the wavelengths of light, whereas HSV encapsulates information about a color in terms that are more familiar to humans: What color is it, how intense is it and how light or dark is it? In this paper, the HSV model is used to obtain more chromatic information from shading effects. If the hue and the saturation are too small, color information of object cannot be distinguished from background. In natural environment, for a tracking object, its chromatic information is so more constant than its luminous information that we can regard chromatic information as foundation of visual tracking. Hence, the HS histogram is created with $N_h N_s$ bins and the pixels' hue and saturation are larger than the hue threshold and the saturation threshold set to 0.2 and 0.2. Whereas, the pixels, which have less color and approach white or black, retain important information when experiment environment is dark. Thus it is useful to create N_v additional value bins with them. The histogram consists of $N = N_h N_s + N_v$ bins. In our experiments, we set N_h, N_s and N_v to 10 by default. An image conversion from RGB color space to HSV color space is shown in Fig.3. It is apparent that the chromatic information is strengthened and the luminous information is weakened from RGB color space to HSV color space.

Within Fig.4, given the state vector $\mathbf{s}_t = (\mathbf{d}_t, \mathbf{e}_t) = (dx_t, dy_t, ex_t, ey_t)$ in time-step t, the state region in which color information will be gathered is defined as $R(\mathbf{s}_t) = \mathbf{d}_t + \mathbf{e}_t W$ (W is default scale of region). And the pixel location is defined as $\mathbf{r}_t = (rx_t, ry_t)$ in state region $R(\mathbf{s}_t)$. We assume $n_t(\mathbf{r}_t) \in \{1, ..., N\}$ as the bin index at pixel location $\mathbf{r}_t$ in time-step t. Within this region a kernel density estimate $\mathbf{h}_t(\mathbf{s}_t) = \{h_{t,n}(\mathbf{s}_t), n = 1...N\}$ of the color distribution is given by [4]

$$h_{t,n}(\mathbf{s}_t) = C \sum_{\mathbf{r}_t \in R(\mathbf{s}_t)} \delta[n_t(\mathbf{r}_t) - n] \gamma(|\mathbf{r}_t - \mathbf{d}_t|) \quad \cdots\cdots (11)$$

where δ is the Kronecker delta function, C is a normalization constant ensuring $\sum_{n-1}^{N} h_{t,n}(\mathbf{s}_t) = 1$, γ is a weighting function, and location $\mathbf{r}_t$ lies on the pixel grid, possibly sub-sampled for efficiency reasons. This model associates a probability to each of the N color bins. In [2][4] the weight function is smooth kernel such that the gradient computations required by the iterative optimization process can be performed. This is not required by our approach, hence we set $\gamma = 1$, which amounts to standard bin counting.

The color histogram $\mathbf{h}_t(\mathbf{s}_t)$ in time-step t associated with a hypothesized state $\mathbf{s}_t$ will be compared to the reference color histogram $\mathbf{h}^\star = \{h_n^\star, n = 1...N\}$,with $\sum_{n-1}^{N} h_n^\star = 1$. In our experiments, the reference distribution is gathered at an initial time at a location/scale $\mathbf{s}_{t_0}^\star$, which is either manually selected, as in [2][4], or automatically provided by a detection module. In either case:

$$\mathbf{h}^\star = \mathbf{h}_{t_0}(\mathbf{s}_{t_0}^\star) \quad \cdots\cdots (12)$$

The data probability must favor candidate color histogram close to the reference histogram, we therefore need to choose a distance on the HSV color distributions. Such a distance $\mathcal{D}$ is used in the deterministic techniques as [2][4] the criterion to be minimized at each time step. In [4], $\mathcal{D}$ is derived from the Bhattacharyya similarity coefficient, and defined as

$$\mathcal{D}[\mathbf{h}^\star, \mathbf{h}_t(\mathbf{s}_t)] = \left[1 - \sum_{n=1}^{N} \sqrt{h_n^\star h_t(\mathbf{s}_t)}\right]^{\frac{1}{2}} \quad \cdots\cdots (13)$$

with the argument that, contrary to Kullback-Leibler divergence, this distance between probability distribution is a proper one, is bound within [0,1], and empty bins are not a source of concern.

When gathering statistics on a number of window sequences obtained from successful tracking behaviors, we observed a consistent exponential behavior for the squared distance $\mathcal{D}^2$. when letting $p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t) \propto p(\mathcal{D}^2[\mathbf{h}^\star, \mathbf{h}_t(\mathbf{s}_t)])$, we thus set:

$$p(\mathbf{z}_t|\mathbf{x}_t = \mathbf{s}_t) \propto \exp\{-\lambda \mathcal{D}^2[\mathbf{h}^\star, \mathbf{h}_t(\mathbf{s}_t)]\} \quad \cdots\cdots (14)$$

In the experiments reported in the paper, it is difficult to estimate satisfactorily the parameter λ, so we fixed it to the same value $\lambda = 20$ by default. This value is in good agreement with the range of values estimated on the labelled sequences mentioned above.

6. Algorithms Summary

From above-mentioned theories our tracking method could be built and used to track target in clutter when the target and tracking camera are moving at the same time. In our experiment the algorithm of tracking approach includes two parts—initialization and iteration. They are summarized in the following table.

1 Initialization

(1) generate initial sample $\mathbf{s}_{t_0}^\star$ and computer its reference histogram $\mathbf{h}^\star = \{h^\star(n), n = 1...N\}$

(2) generate the first sample-set $\{\mathbf{s}_{t_0}^{(m)} = \mathbf{s}_{t_0}^\star, m = 1, ...M\}$

2 Iteration

(1) propagate each sample from the current set $\{\mathbf{s}_t^{(m)}, m = 1, ...M\}$ to generate a new set $\{\tilde{\mathbf{s}}_{t+1}^{(m)}, m = 1, ...M\}$ by second-order ARP

(2) compute candidate histograms $\{\mathbf{h}_{t+1}(\tilde{\mathbf{s}}_{t+1}^{(m)}), m = 1...M\}$

(3) compute each sample's probability $\{\pi_{t+1}^{(m)} = C \exp\{-\lambda \mathcal{D}^2[\mathbf{h}^\star, \mathbf{h}_{t+1}(\tilde{\mathbf{s}}_{t+1}^m)]\}, m = 1...M\}$ and C is a constant which let $\sum_{m=1}^{M} \pi_{t+1}^{(m)} = 1$

(4) generate index function $\{k = i(m), k = 1...M, m = 1...M\}$($i(m)$ is a function that arrange the new samples in descending order by their new probabilities)

(5) built new sample-set $\{\mathbf{s}_{t+1}^{(k)} = \tilde{\mathbf{s}}_{t+1}^{(i(m))}, k = 1...M, m = 1...M\}$, then goto step (1)

Fig. 6. Human head tracking in cluttered room.

7. Experiments

In experiment we draw a reference region as an initial sample $\mathbf{s}_{t0}^{\star}$ in the initial frame by computer mouse. And our tracking system automatically generates M same-size fixed-shape regions which are included in green windows and whose color information best matches the color reference region. In our experiment, M is set to 10 by default. As above algorithm, every region is considered as one sample $\mathbf{s}_t^{(m)}$ in sample-set $\{\mathbf{s}_t^{(m)}, m = 1, ...M\}$ for time-step t, and every region's distance with reference region in HSV color histogram is assumed in inverse proportion to every region's weight $\pi_t^{(m)}$ (probability). Starting from the last position in the previous frame, the new position of every region is calculated in the current frame by second-order ARP. And its weight is calculated

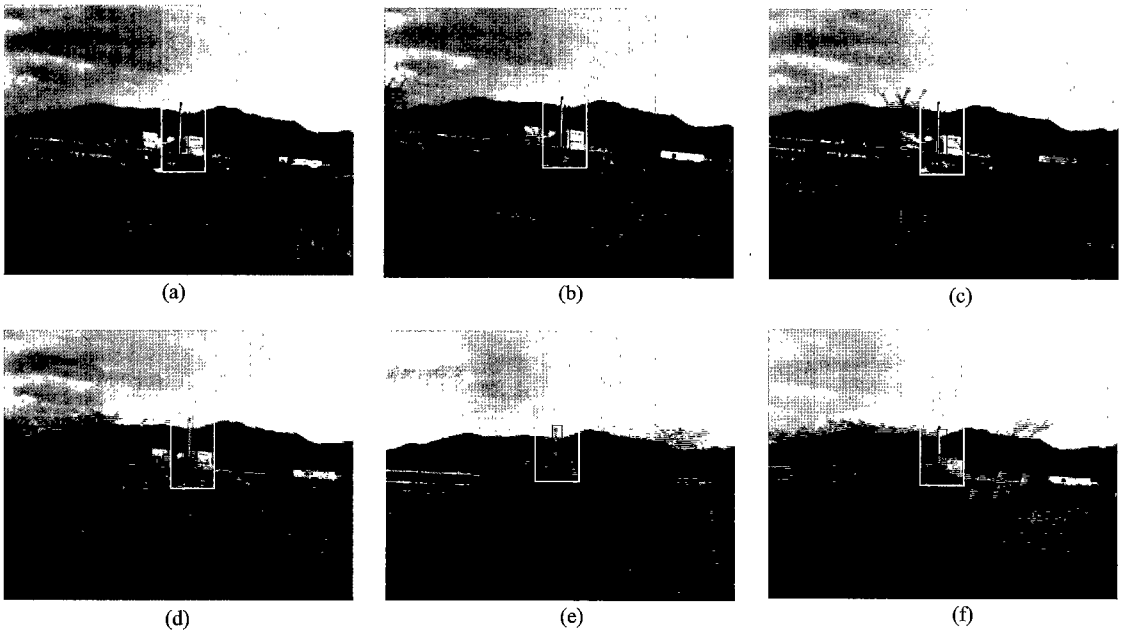


Fig. 7. Chimney tracking in bumpy road.

from its new distance with reference region in HSV color histogram. Moreover, the regions are arranged in descending order of their weights at the same time. The process is iterative at each frame to maintain the most suitable sample-set and use the arithmetical weighty average function to obtain the best region's position. Fig.5 has five example-images captured from experiment video at intervals of 10 frames.

Our tracking system is based on software. We have developed this tracking system on a 1-processor Intel Pentuim4 1.8GHz PC at around 15Hz with 640×480 pixels image sequences. The tracking results based on data association are robust.

Fig.6 is an image sequence of experiment whose some images have been mentioned in Fig.5 to explain our tracking algorithm. We use a camcorder to capture the person at the center of our office. A person is walking and turning round freely from one side of the office to another side. His head that is the tracking target is very similar with the boxes and instruments on shelf, cabinet and desk. The boxes' color is close to the skin and the dim instruments' color is close to the hair. The results show that the tracking system is able to find the head position in the image sequences and control the camera pan/tilt to adjust the head centered in the field of view, no matter when the head rotates or partially disappears and interferences are closed to it or partially cover it. So the system using dynamic color tracking method proposed in this paper is suitable for cluttered environments.

In Fig.7, we install tracking system in a car and set the camera towards window. A white chimney of factory is the tracking target. No matter how many the obstacles appear and no matter how violently the car shakes, the tracking target – chimney is adjusted in the center of image. All of the processes have run in real time.

Fig.8 illustrates the results of the human head tracking and the chimney tracking based on dynamic color tracking method proposed in this paper. The frame accuracies of each sequence are estimated at intervals of 10 frames. The average accuracy for human head tracking frames is about 87.7% with deviation 6.1%. And the average accuracy for chimney tracking frames is about 92.1% with deviation 2.3%. The human head are more changeful than chimney in color space when they are moving and rotating. Therefore the results of the chimney tracking are more accurate than the human head tracking in these experiments.

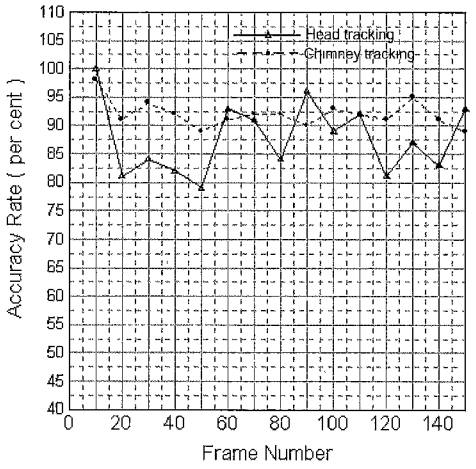


Fig. 8. The accuracies of the human head tracking and the chimney tracking based on dynamic color tracking method.

8. Conclusion

This paper has presented a dynamic color tracking method based on probabilistic data association. This method can be applied to changeful target tracking in

cluttered and dynamic environments. The factorized sampling theory is used in our system to choose the set of samples and let their mean distributions very approximate the target's ones. Because the method considers the relationship of target's representations and states between different time-steps by using sequential Monte Carlo framework, the proposed method is robust and efficient in real time with big image size.

The efficient tracking of visual features in complex environments is challenging task for the vision community. Real-time application such as surveillance and monitoring, perceptual user interfaces, smart rooms and video compression all require the abilities to track moving object. So we need to apply probabilistic data association methods to more extensive image processing fields.

(Manuscript received July 3, 2003,
revised Nov. 10, 2003)

References

(1) Stan Birchfied: "Ellitical head tracking using intensity gradient and color histograms", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp.232–237 (1998)
(2) G.R. Bradski: "Computer vision face tracking as a component of a perceptual user interface", *Workshop on Applications of Computer Vision*, pp.214–219 (1998)
(3) Ying Wu and Thomas S. Huang: "Color tracking by transductive learning", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol.I, pp.133–138 (2000-6)
(4) D. Comanciu, V. Ramesh, and P. Meer: "Real-time tracking of non-rigid objects using mean shift", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol.II, pp.142–149 (2000)
(5) C. Isard and G. Hager: "Joint probabilitic techniques for tracking multi-part objects", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 16–21 (1998)
(6) C. Wren, A. Azarbayejani, T. Darrel, and A. Pentland: "Real-time tracking of the human body", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol.9, pp.780–785 (1997)
(7) I. Sobel: "An Isotropic 3×3 image gradient operator", Machine Vision for Three-demensional Scences, Academic Press, pp.35–45 (1990)
(8) A. Blake, M. Isard, and D. Reynard: "Learning to track the visual motion of contours", *Artificial Intelligence*, No.78, pp.101–133 (1995)
(9) D. Terzopoulos, and R. Szeliski: "Tracking with Kalman Snakes", *Active Vision*, MIT Press, pp.3–20 (1992)
(10) H. Tao, H. Saehney, and R. Kumar: "Dynamic layer representation with applications to tracking", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol.II, pp.134–141 (2000)
(11) R. Kalman: "A new approach to linear filtering and prediction Problems", *J.Basic Eng.*, Vol.82, pp.35–45 (1960)
(12) M. Isard and A. Black: "Contour tracking by stochastic propagation of conditional density", *Proc. of European Conf. on Computer Vision*, pp.343–356 (1996)
(13) M. Isard and A. Black: "ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework", *Proc. of European Conf. on Computer Vision*, Vol.I, pp.767–781 (1998)
(14) A. Blake, and M. Isard: "Active contours", Springer (1998)

Appendix

A second order auto-regressive process model(ARP)[14] has been described by formula(10). Given a training set $\{\mathbf{x}_1, ..., \mathbf{x}_M\}$ from an image sequence, learn the parameters P, Q, U and T for a second-order ARP that describes the dynamics of tracking.

(1) First sums $R_i, i = 0, 1, 2$ and auto-correlation coefficients R_{ij} and $R'_{ij}, i, j = 0, 1, 2$ are computed:

$$\left.\begin{aligned} R_i &= \textstyle\sum_{k=3}^{M} \mathbf{x}_{k-i} \\ R_{ij} &= \textstyle\sum_{k=3}^{M} \mathbf{x}_{k-i}\mathbf{x}_{k-j}^T \\ R'_{ij} &= R_{ij} - \tfrac{1}{M-2} R_i R_j^T \end{aligned}\right\} \cdots\cdots\cdots \text{(A1)}$$

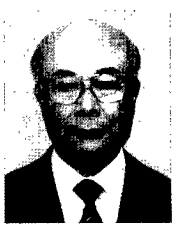
(2) Estimated parameters P, Q and T are given by

$$\left.\begin{aligned} Q &= (R'_{02} - R'_{01}R'^{-1}_{11}R'_{12}) \\ &\quad (R'_{22} - R'_{21}R'^{-1}_{11}R'_{12})^{-1} \\ P &= (R'_{01} - QR'_{21})R'^{-1}_{11} \\ T &= \tfrac{1}{M-2}(R_0 - QR_2 - PR_1) \end{aligned}\right\} \cdots\cdots \text{(A2)}$$

(3) The covariance coefficient U is estimated as a matrix square root $U = \sqrt{V}$ where

$$V = \frac{1}{M-2}(R_{00} - QR_{20} - PR_{10} - TR_0^T) \qquad \cdots\cdots\cdots\cdots\cdots\cdots \text{(A3)}$$

Xin Lu (Student Member) He received the B.S. degree in 2000 from Hohai University of China. Then he has been a software engineer in Fujitsu Ltd. in China for a year. In 2003, he received the M.S.Eng. degree from University of Tokushima. Now he is a Ph.D student at the Department of Information Science and Intelligent Systems in University of Tokushima. His research interests include image processing, pattern recognition, etc. He is a member of the Institute of Electrical Engineers of Japan.



Shunichiro Oe (Member) He received the B.S. and M.S.Eng degrees from University of Tokushima in 1967 and 1969. In 1980, he received the Ph.D. degree from the University of Osaka Prefecture. Between 1969 and 1974, he was research assistant at computer center of University of Tokushima. From 1974 to 2002 he was a lecture, associate professor and professor at the Department of Information Science and Intelligent Systems in University of Tokushima. Now he has been a head of the Center for Advanced Information Technology in University of Tokushima from 2002. His current research interests include image processing, remote sensing, pattern recognition, etc. He is a member of the Institute of Electrical Engineers of Japan.