공학석사 학위논문

Feature-based Particle Filter for Multiple Objects Tracking

다중객체추적을 위한 특징 기반의 파티클 필터

2012년 8월

서울대학교 대학원

전기·컴퓨터공학부

서 보 경

Feature based Particle Filter for Multiple Objects Tracking

지도 교수 유 석 인

이 논문을 공학석사 학위논문으로 제출함

2012 년 8 월

서울대학교 대학원

전기·컴퓨터공학부

서 보 경

서보경의 석사 학위논문을 인준함

2012 년 8 월

위 원 장 _____ (인)

부위원장 _____ (인)

위 원 _____ (인)

Abstract

Feature-based Particle Filter for Multiple Objects Tracking

Bo-Kyung Seo

School of Computer Science and Engineering

The Graduate School

Seoul National University

This paper proposes an advanced particle filter for multi-target tracking with speed-up robust features. In this study, a mixture of the Gaussian Background Model and the SURF algorithm are used for target representation and localization. This approach transforms an image into a large collection of local feature vectors, each of which is invariant to the image's translation, scaling, and rotation. Additionally, it is also partially invariant to illumination changes and affine or 3D projection. Lastly, NN algorithm is used for segmenting multiple objects into a single-object state space.

Several experimental results show that the proposed algorithm has good performance for object tracking in the presence of object translation, rotation and partial occlusion. Overall, this approach makes the system robust to occlusions and allows false positive detections in the background to be identified and removed.

Keywords: Feature point, SURF, Particle Filter, Tracking
Student Number: 2009-23113

Contents

List of Figures

Chapter 1. Introduction

Object tracking is an essential task within the field of computer vision. Especially, the detection and tracking of multiple targets problem has arisen in a wide variety of contexts. For example, tracking of aircraft by means of radar, tracking of sea animals or submarines based on sonar and also the tracking of people using videos for surveillance or security purposes are included.

The goal of object tracking is to determine the position of the object in images continuously and reliably against dynamic scenes [1]. To achieve this aim, a number of algorithms have been established. As a promising solution, the particle filter that allows for an approximate solution to the sequential estimation, has been successfully implemented in current tracking technologies. Other tracking strategies can also be found as Multiple Hypothesis Tracking [4, 5], Kernel-Based Tracking [6, 7] and Optical Flow-Based Tracking [8].

Particle Filtering provides a convenient framework for estimating and propagating the density of state variables regardless of the underlying distribution and the given system. These schemes [2, 3] recruited particles for computing a sampled representation of the posterior probability distribution over scene properties of interest, based on image observations.

Our proposed approach explores the particle filter's ability of tracking multiple objects in a single-object state space with speed-up robust features. First, correspondence between objects and observations, namely the

motion correspondence problem has to be established. In this work, the mixture of the Gaussian Background Model and the SURF algorithm are used for target representation and localization. Object localization aims to find the location and scale of an object in an image. This approach transforms an image into a large collection of local feature vectors, each of which is invariant to image translation, scaling, rotation, and also partially invariant to illumination changes and affine or 3D projection. Additionally, the NN algorithm is used after the localization for segmenting multiple objects into a single-object state space. Experiments show that the proposed approach renders good performance for multiple objects tracking in the presence of object translation, rotation and partial occlusion.

The principal portion of this paper is composed of the following. Section 2 introduces the previous works, Section 3 presents how to figure out and represent the object region. The object recognition is also explained in this section. In Section 4, we present the particle filter and explain how it works for object tracking. Finally, sections 5 and 6 discuss experimental results and conclusions, respectively.

.

Chapter 2. Related Work

2.1 Object Detection

2.1.1 Geometry-based Object Detection

The geometric description of a 3D object allows the projected shape to be predicated accurately in a 2D image under projective projection. Therefore, the recognition using edge or boundary information is main idea. Much attention was made to extract geometric primitives such as lines, circles that are invariant to viewpoint change [9]. Nevertheless, it has been shown that such primitives can only be reliably extracted under limited conditions controlled variation in lighting and viewpoint with certain occlusion.

2.1.2 Appearance-based Object Detection

Most recent efforts have been centered on appearance-based techniques as advanced feature descriptors and pattern recognition algorithms are developed [10]. Most notably, the eigenface methods have attracted much attention as it is one of the first face recognition systems that are computationally efficient and relatively accurate [11]. The underlying idea of this approach is to compute eigenvectors from a set of vectors where each one represents one face image as a raster scan vector of gray-scale pixel values. Each eigenvector captures certain variance among all the vectors, and a small set of eigenvectors captures almost all the appearance variation of face images in the training set.

Given a test image represented as a vector of gray-scale pixel values, its identity is determined by finding the nearest neighbor of this vector after being projected onto a subspace spanned by a set of eigenvectors. In other

words, each face image can be represented by a linear combination of eigen faces with minimum error, and this linear combination constitutes a compact reorientation. The eigen face approach has been adopted in recognizing generic objects across different viewpoints [12] and modeling illumination variation [13].

As the goal of object recognition is to tell one object apart from the others, discriminative classifiers have been used to exploit the class specific information. Classifiers such as k-nearest neighbor, neural networks with radial basis function (RBF), dynamic link architecture, Fisher linear discriminant, support vector machines (SVM), sparse network of Winnows (SNoW), and boosting algorithms have been applied to recognize 3D objects from 2D images [13, 14, 15, 16, 17, 18]. While appearance-based methods have shown promising results in object recognition under viewpoint and illumination change, they are less effective in handling occlusion. In addition, a large set of exemplars needs to be segmented from images for generative or discriminative methods to learn the appearance characteristics. These problems are partially addressed with parts-based representation schemes.

2.1.3 Feature-based Object Detection

Finding interest points is central idea of feature-based object recognition algorithms. Interest point are invariant to change due to scale, illumination and affine transformation. The scale-invariant feature transform descriptor, proposed by Lowe, is arguably one of the most widely used feature representation schemes for vision applications [10]. The SIFT approach uses extrema in scale space for automatic scale selection with a pyramid of difference of Gaussian filters, and keypoints with low contrast or poorly localized on an edge are removed. Next, a consistent orientation is assigned to each keypoint and its magnitude is computed based on the local image gradient histogram, thereby achieving invariance to image rotation. At each

keypoint descriptor, the contribution of local image gradients are sampled and weighted by a Gaussian, and then represented by orientation histograms. For example, the 16 x 16 sample image region and 4 x 4 array of histograms with 8 orientation bins are often used, thereby providing a 128-dimensional feature vector for each keypoint. Objects can be indexed and recognized using the histograms of keypoints in images. Numerous applications have been developed using the SIFT descriptors, including object retrieval [19, 20], and object category discovery [21].

Although the SIFT approach is able to extract features that are robust to certain scale and illumination change, vision applications with large base line change entail the need of affine invariant point and region operators [22].

A performance evaluation among various local descriptors can be found in [23], and a study on affine region detectors is presented in [22]. Finally, SIFT-based methods are expected to perform better for objects with rich texture information as sufficient number of keypoints can be extracted. On the other hand, they also require sophisticated indexing and matching algorithms for effective object recognition [10, 24].

2.2 Data Association

The purpose of data association in multiple targets tracking problem is to recover the correct correspondence between observations and targets. Once data association is established, filtering is applied to estimate the state of targets.

Most existing data association algorithms consider a one-to-one mapping between targets and observations, which assumes that at a given time instant one observation can be associated with at most one target and vice versa: one target corresponds to at most one observation. This assumption is reasonable when the considered observations are punctual. However, in the visual tracking problem, the observations correspond to blobs or meaningful

regions which cannot be faithfully modeled by a single point. Moreover, erroneous detections due to occlusion and spurious motion segmentation provide a set of observations where a single moving object is often detected as multiple moving regions, or multiple moving regions are merged into a single blob. Therefore, the one-to-one association is usually violated in real environments.

Methods to solve the occlusion problem in multiple interacting objects tracking have been previously presented. Shiloh [25], Chang [26] and Dockstader [27] overcame occlusion in multiple objects tracking by used fusing multiple camera inputs. Cucchiara [28] proposed probabilistic masks and appearance models to cope with frequent shape changes and large occlusions. Eng [29] developed a Bayesian segmentation approach that fused a region-based background subtraction and a human shape model for people tracking under occlusion. Wu [30] proposed a dynamic Bayesian network which accommodates an extra hidden process for partial occlusion handling. Andrew [31, 32], Siebel [33], Hieu [34] and Alper [35] used appearance models to track occluded objects. Tao [36] presented a dynamic background layer model and model each moving object as a foreground layer, together with the foreground ordering, the complete information necessary for reliably tracking objects through occlusion is included.

2.3 Multi-targets Tracking

Although many algorithms have been proposed in the literature, the problem of multiple objects tracking in dynamic scene is still far from being solved. Multiple camera based tracking methods [25, 26, 27] still cannot handle complete occlusion. Precise model based algorithms [31, 32] are sensitive to background clutter, and they are at the cost of computationally more expensive schemes because model estimation for the number of model parameters is usually large. Moreover, many of those algorithms are designed

to deal with short-duration partial occlusion, and fail at severe occlusions or when a partial occlusion lasts for a long time. Probabilistic approaches like the Monte Carlo filter is useful for dealing with the problem of background clutter as it allows for the tracking of multiple hypotheses [37, 38, 39, 40]. However, the measure of object has to be detected by an independent technique that may not be acquired in heavy occlusion. Several methods using motion model to perform robust tracking can deal with some instances of occlusion. These methods require precise motion modeling [40] and fail at the non-linear motion of interacting objects.

2.3.1 Bayesian Filtering

The Bayesian approach to dynamic state estimation problems involves the construction of the probability density function of the current state of an evolving system, given the accumulated observation history. For linear Gaussian models where the probability density functions can be summarized by means and covariance, the calculation is carried out in terms of the familiar updating equations of the Kalman filter. In general, for nonlinear, non-Gaussian models, there is no simple way to proceed. Two difficulties must be resolved: how to represent a general probability density function using finite computer storage and how to perform the integrations involved in updating the probability density function when new data are acquired.

Several approximate methods have been proposed. These include the extended Kalman filter [41, 42], the Gaussian sum filter, approximating the first two moments of the probability density functions and numerical integration over a grid of points in the state space. However, none of these methods can be applied automatically.

Chapter 3. Object Detection and Matching

3.1 Fast Interest Point Detection

The SURF feature detector is based on the Hessian matrix. Given a point $X = (x, y)^T$ in an image I, the Hessian matrix $H(X, \sigma)$ in X at scale σ is defined as follows.

$$H(X, \sigma) = \begin{bmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{xy}(X, \sigma) & L_{yy}(X, \sigma) \end{bmatrix} \quad (1)$$

Where $L_{xx}(X, \sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image I in point X, and similarly for $L_{xy}(X, \sigma)$ and $L_{yy}(X, \sigma)$.

In contrast to SIFT, which approximates Laplacian of Gaussian with Difference of Gaussians, SURF approximates second order Gaussian derivatives with box filters, see Figure 1. Image convolutions with these box filters can be computed rapidly by using integral images as defined in [43]. The entry of an integral image $I_{\Sigma}(X)$ at location $X = (x, y)^T$ represents the sum of all pixels in the base image I of a rectangular region formed by the origin and X.

$$I_{\Sigma}(X) = \sum_{i=0}^{i \le x} \sum_{j=0}^{j \le y} I(i, j) \quad (2)$$

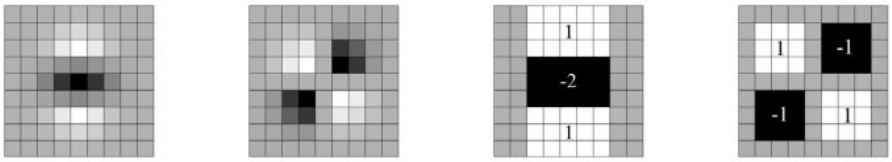


Figure1. Approximation of the second order derivatives. Left to right: the Gaussian second order partial derivatives in y-direction and xy-direction, and approximations there of using box filters. The grey regions are equal to zero.

Once we have computed the integral image, it is straight forward to calculate the sum of the intensities of pixels over any upright, rectangular area.

The location and scale of interest points are selected by relying on the determinant of the Hessian. Here, the approximation of the second order derivatives is denoted as D_{xx}, D_{yy} and D_{xy}. By choosing the weights for the box filters adequately, Hessian's determinant can be approximated.

$$\mathrm{Det}(H_{approx}) = D_{xx}D_{yy} - \left(0.9D_{xy}\right)^2 \qquad (3)$$

Interest points are localized in scale and image space by applying non-maximum suppression in a 3x3x3 neighborhood. Finally, the resultants maxima of the determinant of the approximated Hessian matrix are interpolated with the scale and image space.

3.2 Descriptor of Interest Point

In the first step, SURF constructs a circular region around the detected interest points in order to assign a unique orientation to the former and thus gain invariance to image rotations. The orientations is computed using Haar wavelet responses in both x and y direction as shown in the middle of Figure 2. The Haar wavelets can be easily computed via integral images, similar to the Gaussian second order approximated box filters. Once the Haar wavelet responses are computed, they are weighted with a Gaussian with $\sigma = 2.5$ s centered at the interest points. In a next step, the dominant orientation is estimated by summing the horizontal and vertical wavelet responses within a rotating wedge, covering an angle of $\frac{\pi}{3}$ in the wavelet response space. The resulting maximum is then chosen to describe the orientation of the interest point descriptor.

In the second step, the SURF descriptors are constructed by extracting square regions around the interest points. These are oriented in the directions assigned in the previous step. Some example windows are shown on the right side of Figure 2. The windows are split up into 4 x 4 sub-regions in order to retain some spatial information. In each sub-region, Haar wavelets are extracted at regularly spaced sample points. In order to increase robustness to geometric deformations and localization errors, the responses of the Haar wavelets are weighted with a Gaussian, centered at the interest point. Finally, the wavelet responses in horizontal d_x and vertical directions d_y are summed up over each sub-region. Furthermore, the absolute values $|d_x|$ and

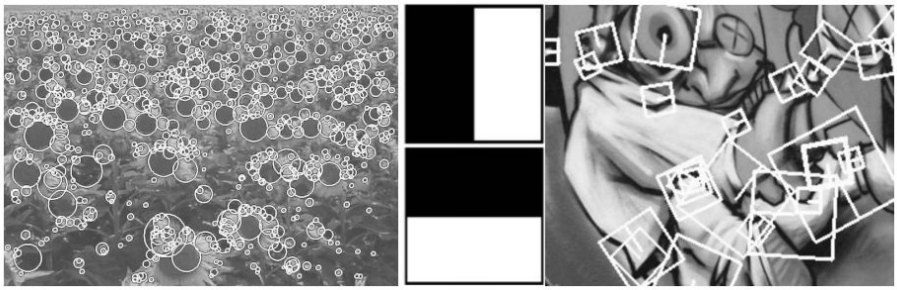


Figure 2. Detecting interest points using SURF. This kind of scenes shows clearly the nature of the features from Hessian-based detectors. Middle: Haar wavelet types used for SURF. Right: Detail of the Graffiti scene showing the size of the descriptor window at different scales.

$|d_y|$ are summed in order to obtain information about the polarity of the image intensity changes. Hence, the underlying intensity pattern of each sub-region in described by a vector.

$$V = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right) \tag{4}$$

The resulting descriptor vector for all 4 x 4 sub-regions is of length 64. See Figure 3 for an illustration of the SURF descriptor for three different image intensity patterns. Notice that the Haar wavelets are invariant to illumination bias and additional invariance to contrast is achieved by normalizing the descriptor vector to unit length.

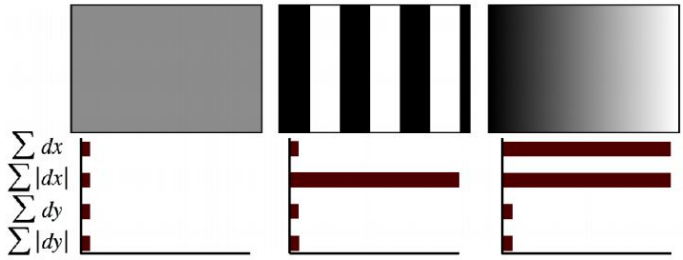


Figure 3. The descriptor entries of a sub-region. It represents the nature of the underlying intensity pattern. Left: In case of a homogeneous region, all values are relatively low. Middle: In presence of frequencies in x direction, the value of $\sum |d_x|$ is high, but all others remain low. If the intensity is gradually increasing in x direction, both values $\sum d_x$ and $\sum |d_x|$ are high.

3.3 Object Matching

Object recognition is archived by image matching. Extracted interest points of the input image are compared to the interest points of all model images. In order to create a set of interest point correspondences M, the nearest neighbor ratio matching strategy [41, 6, 42, 43] is used. This describes that a matching pair is detected if its Euclidean distance in descriptor space is closer than 0.6 times the distance to the second nearest neighbor.

The selected object is the one figuring in the model image with the highest recognition score S_R. This score is traditionally the number of total matches in M. However, the presence of mismatches often leads to false detections. This can be avoided with the help of the following new alternative

for estimation of the recognition score. Hereby, we calculate the mean Euclidean distance to the individual nearest neighbors for each image pair. This value is typically smaller for corresponding image pairs than for non-corresponding ones, and it does not depends on the number of extracted features in the individual images. Hence, we maximize the following recognition score.

$$S_R = \underset{i}{\operatorname{argmax}} \left(0.5 \times \frac{N_i}{\sqrt{\sum_{j=1}^{N_i} d_{ij}^2}} + 0.5 \times \frac{1}{(cx_i - cx_j)^2 + (cy_i - cy_j)^2} \right) \quad (5)$$

Then we choose the object which the mean distance of its matches is smallest. N_i denotes the number of matches in image i. Furthermore, d_{ij} is the Euclidean distance in the descriptor space between a matching pair of keypoints.

Chapter 4. Particle Filter for Object Tracking

The simplest particle filter consists of essentially two steps: prediction and update. Given all available $y_{1:t-1} = \{y_1, \dots, y_{t-1}\}$ observations up to time t-1, the prediction stage uses a probabilistic transition model $p(x_t|x_{t-1})$ to predict the posterior at time t as

$$p(x_t|y_{t-1}) = \int p(x_t\,|x_{t-1})p(x_{t-1}|y_{1:t-1})\,dx_{t-1} \quad (6)$$

At time t, the observation y_t is available, the state can be updated using Bay`s rule

$$P(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t})} \quad (7)$$

Where $p(y_t|x_t)$ is described by the observation equation.

In the particle filter, the posterior $p(x_t|y_{1:t})$ is approximated by a finite set of N samples $\{x_t^i\}_{i=1,\dots,N}$ with importance weights w_t^i. The candidate samples $\tilde{x}_t^i$ are drawn from an importance distribution $q(x_t|x_{1:t-1}, y_{1:t})$ and the weight of the samples are

$$w_t^i = w_{t-1}^i \frac{p(y_t|\widetilde{x_t^i})p(\widetilde{x_t^i})|x_{t-1}^i}{q(\widetilde{x_t}|x_{1:t-1},y_{1:t})} \quad (8)$$

The samples are resampled to generate an unweighted particle set

according to their importance weights to avoid degeneracy. In the case of the bootstrap filter, $q(x_t|x_{1:t-1}, y_{1:t}) = p(x_t|x_{t-1})$ and the weights become the observation likelihood $p(y_t|x_t)$.

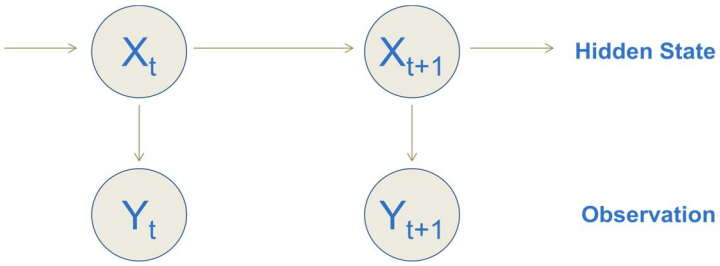


Figure 4. Probabilistic frame work for tracking system

For the implementation of the particle filter, the following mathematical model is used:

1. Transition model / state motion model $p(x_t|x_{t-1})$: this specifies how objects move between frames.
2. Observation model $p(y_t|x_t)$: this specifies the likelihood of an object being in a specific state (i.e. at the specific location).
3. Initial state Est (1) / prior distribution model $p(x_0)$: describes initial distribution of object states.

Chapter 5. Experimental Result

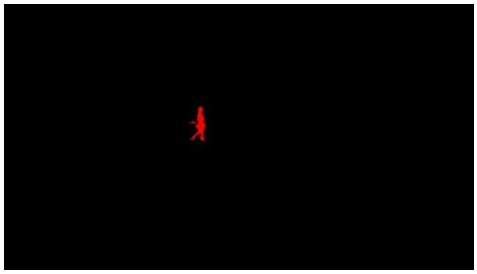
5.1 Environment

In order to evaluate the performance of the proposed approach, we implemented and tested our system in many different dynamic environments. In the experiments, we track the location of a pedestrian under various circumstances such as background clutter, overlapped multiple objects, hidden object, object deformation and so on. We test our method on various videos. The image size of the video is 640x480 pixels and all the experiments are performed on a desktop computer with a 2.67GHz core CPU. The algorithm is implemented in C++ and it takes about 50 milliseconds to handle each frame. The experimental results are showed in figure 4-9.

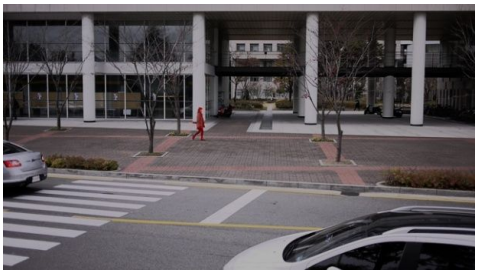
5.2 Result

5.2.1 Single Object Detection

(a)



(b)



(c)

Figure 5. Extracting and Detecting Single Object

Figure 5 illustrates that a single object is extracted successfully by a Mixture of Gaussian Background model. Figure 5 (a) is the original image, and (b), (c) represent the extracted single object.

5.2.2 Key Points Extraction

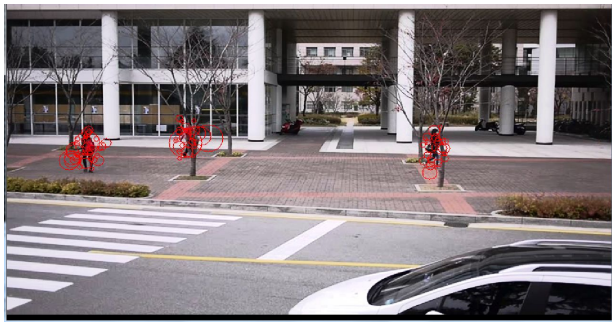


Figure 6. Extracting Key Points

Figure 6 shows that each object can be composed of a number of feature points which are extracted using SURF.

5.2.3 Key Points Matching

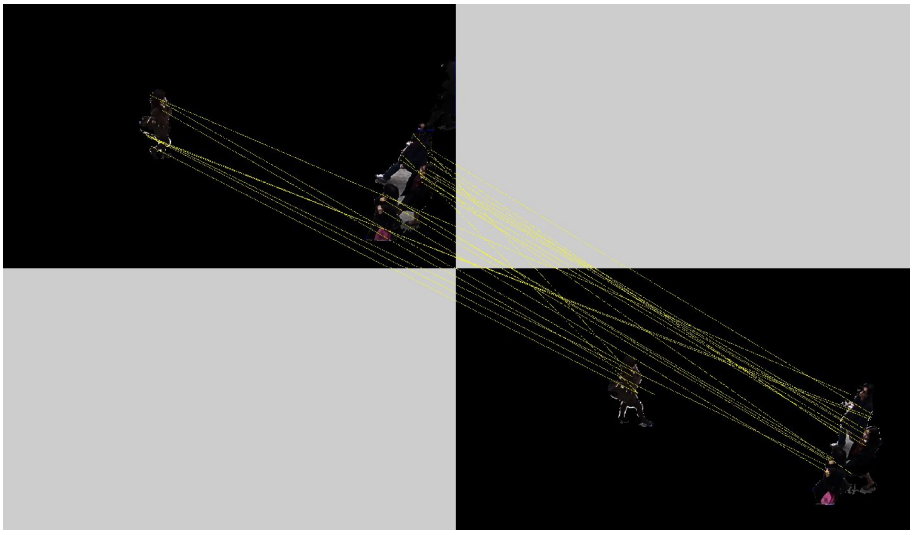


Figure 7. Finding Key Pairs

Figure 7 illustrates that key pair can be created by matching feature points between previous and current frame. The mean Euclidean distance to the individual nearest neighbors for each image pair is calculated for making correspondence between objects and observations.

5.2.4 Multi-Objects Tracking

Figure 8. Tracking multi-objects on a complex background. The sequence has 600 frames and the frames 224, 245, 291 and 306 are shown.

Figure 8 presents the multi-object tracking with advanced particle filter using SURF. On the left of figure 8, it shows that the Objects are extracted from background and marked with different colors. The right of figure 8 shows that our method is able to track multi-objects successfully.

5.2.5 Comparing performance

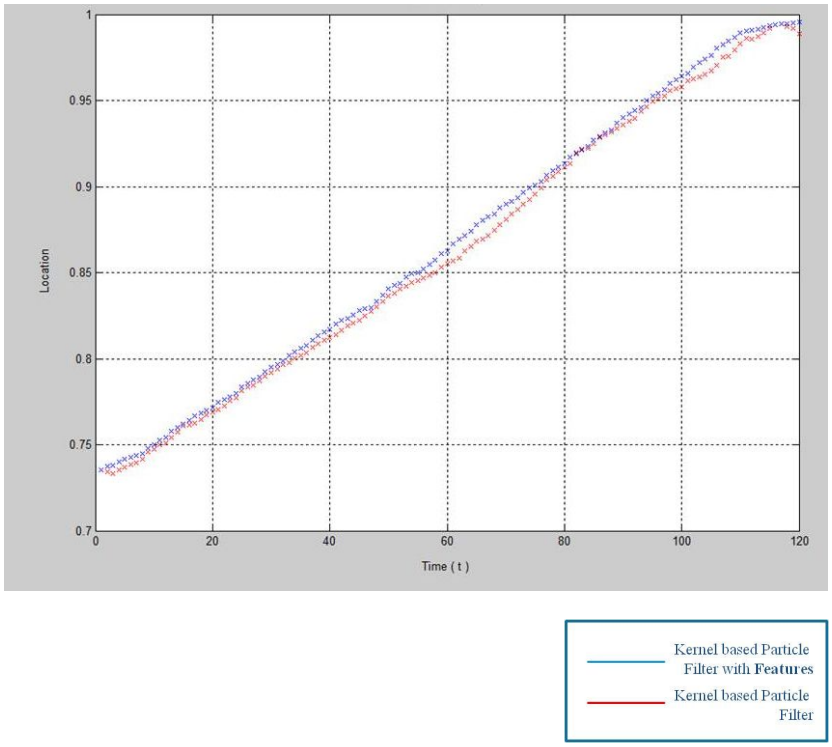


Figure 9. Trajectory of object with Features and without Features.

The performance for estimating the object's location between the particle filter with features and without features is shown in Figure 9. It can be seen that using the particle filter with features is more accurate than the one

without features.

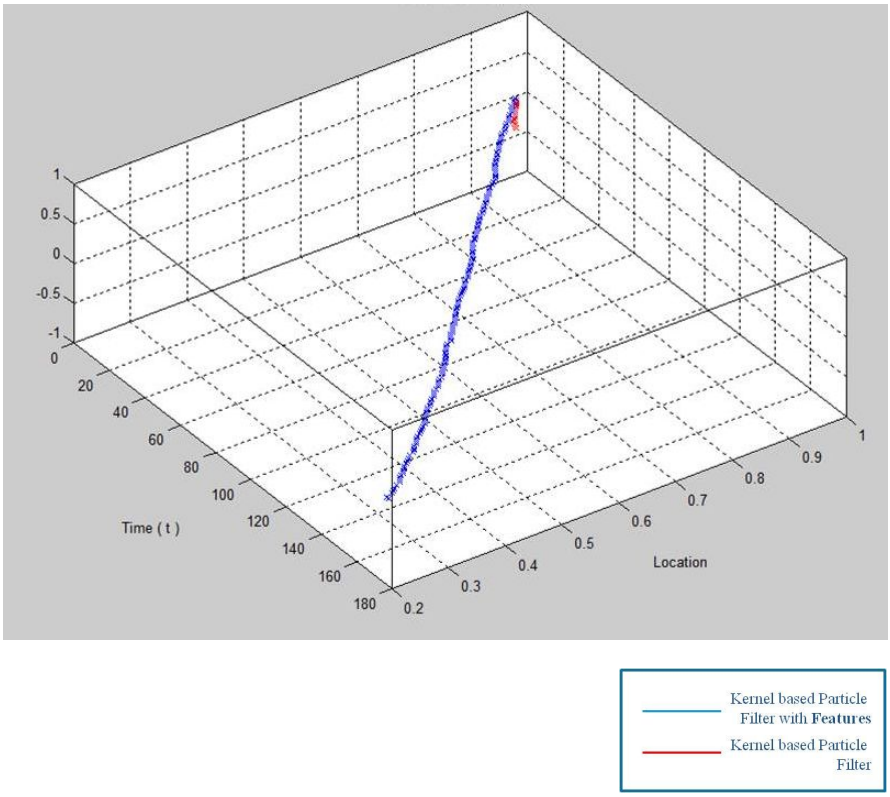


Figure 10. Trajectory of Object in Partial Occlusion

Figure 10 illustrates that particle filter without features can fail to track object, on the other hand, the one with features can keep on tracking.

Chapter 6. Conclusions

In this paper, we have described and demonstrated the advanced particle filter using SURF Features. Several experimental results show that the proposed algorithm has good performance for object tracking in the presence of object translation, rotation and partial occlusion. This approach makes the system robust to occlusions and allows false positive detections in the background to be identified and removed. Our system can be adapted to many potential tracking applications such as video surveillance, visual navigation and monitoring, content-based indexing and retrieval, object-based coding, traffic monitoring and so on.

Bibliography

1. T.-L. Liu and H.-T. Chen, “Real-time tracking using trust region methods,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 3, pp. 397–402, 2004.

2. B. Ristic, S. Arulampalam, and N. Gordon, Beyond the Kalman filter: Particle filters for tracking applications, Artech House, 2004.

3. K. Nummiaro, E. Koller-Meier, and L. Van-Gool, “An adaptive colorbased particle filter,” Image and Vision Computing, vol. 21, pp. 99–110, 2003.

4. C. Rasmussen, G. Hager, Probabilistic data association methods for tracking complex visual objects, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 560–576.

5. T. Cham and J. Rehg, “A Multiple Hypothesis Approach to Figure Tracking,” Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. II, pp. 239-219, 1999.

6. D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 5, pp. 564–577, 2003.

7. D. Xu, Y. Wang, and J. An, “Applying a new spatial color histogram in mean-shift based tracking algorithm,” in Proceedings of the Image and Vision Computing New Zealand (IVCNZ ’05), University of Otago, Dunedin, New Zealand, November 2005.

8. Lucena, M., Fuertes, J.M., de la Blanca, N.P., Evaluation of three optical flow based observation models for tracking. ICPR (2004) 236–239

9. J. Mundy and A. Zisserman, editors. Geometric invariance in computer vision. MIT Press, 1992.

10. D. Lowe. Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 2004.

11. M. Turk and A. Pentland. Eigenfaces for recognition. Journal of Cognitive Neuroscience,1991.

12. H. Murase and S. K. Nayar. Visual learning and recognition of 3-D objects from appearance. International Journal

13. P. Belhumeur and D. Kriegman. What is the set of images of an object under all possible illumination conditions, International Journal of Computer Vision,1998.

14. T. Poggio and S. Edelman. A network that learns to recognize 3D objects. Nature,1990.

15. M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. IEEE Transactions on Computers,1993.

16. P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class speci_c linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence,1997.

17. M. Pontil and A. Verri. Support vector machines for 3d object recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998.

18. D. Roth, M.-H. Yang, and N. Ahuja. Learning to recognize objects. Neural Computation, 2002

19. J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. In Proceedings of IEEE International Conference on Computer Vision.

20. D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2006.

21. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2003.

22. K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, A comparison of affine region detectors. International Journal of Computer Vision, 2006.

23. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.

24. J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, editors. Toward category-level object recognition. Springer-Verlag, 2006.

25. Dockstader et al, Multiple camera tracking of interacting and occluded human motion, Proceedings of the IEEE , 2001.

26. Ting-Hsun Chang, Shaogang Gong, and Eng-Jong, Tracking multiple people under occlusion using multiple cameras. In Proc.11th British Machine Vision Conference, 2000.

27. S.L. Dockstader and A.M. Tekalp, Multiple camera fusion for multi-object tracking. In Proc. IEEE Workshop on Multi-Object Tracking, 2001.

28. R.Cucchiara, C.Grana, G.Tardini, R.Vezzani, Probabilistic people tracking for occlusion handling, Proceedings of the 17th International Conference on ICPR 2004.

29. How-Lung Eng, et al. A bayesian framework for robust

human detection and occlusion handling using human shape model, ICPR, 2004.

30. Ying Wu, Ting Yu, Gang Hua, Tracking appearances with occlusions, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003.

31. A. Senior, et al. Appearance Models for Occlusion Handling. PETS, 2001.

32. A. Senior, Tracking with Probabilistic Appearance Models, Proc. ECCV workshop on Performance Evaluation of Tracking and Surveillance Systems, 2002.

33. N. T. Siebel, S. Maybank, Fusion of Multiple Tracking Algorithms for Robust People Tracking, 7th European Conf. on Computer Vision, 2002.

34. Hieu T. Nguyen and Arnold W.M. Smeulders, Fast Occluded Object Tracking by a Robust Appearance Filter, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004.

35. A. Yilmaz and M. Shah, Contour-Based Object Tracking with Occlusion Handling in Video Acquired Using Mobile Cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004.

36. H. Tao, H. S. Sawhney, and R. Kumar, Dynamic Layer

Representation with Applications to Tracking, CVPR, 2000.

37. M. Isard and A. Blake, CONDENSATION – Conditional Density Propagation for Visual Tracking, International Journal on Computer, 1998.

38. M. Isard and A. Blake, Contour Tracking by Stochastic Propagation of Conditional Density, ECCV, 1996.

39. A. Doucet, N. Freitas, N. Gordon, Sequential Monte Carlo Methods in Practice, Springer. 2001.

40. R. Rosales and S. Sclaroff, Improved Tracking of Multiple Humans with Trajectory Prediction and Occlusion Modeling, CVPR, 1998.

41. JAZWINSKI, A.H., 'Stochastic processes and filtering theory' (Academic Press, 1973)

42. MOON, J.R. and STEV'ENS, C.F., 'An approximate linearisation approach to bearings-only tracking'. IEEE colloquium on Target tracking and data fusion, 1996.

43. ALSPACH, D.L. and SORENSON, H.W.: 'Non-linear Bayesian estimation using Gaussian sum approximation', ZEEE Trans.,

44. Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Computer

Vision and Image Understanding (CVIU), 2008

초 록

Feature-based Particle Filter for Multiple Objects Tracking

서울대학교 대학원
컴퓨터공학부
서 보 경

본 논문에서는 다중객체추적을 위하여 SURF(Speed-up Robust Features)을 이용한 Particle Filter 알고리즘을 제안한다. 객체를 표현하기 위하여 Mixture of Gaussian Background Model을 이용하여 전경과 후경을 분리하였고, SURF(Speed-up Robust Features) 을 이용하여 각각의 객체를 표현하기 위한 특징을 추출하였다. 이를 통해 하나의 영상을 지역적 특징을 담은 벡터들의 집합으로 표현할 수 있었고, 영상의 크기, 회전, 밝기 변화, 왜곡 등에 덜 민감한 반응을 보였다. 또한 NN 알고리즘은 여러 개의 객체를 하나의 객체 상태 공간으로 표현하는데 사용되었다. 마지막으로 Particle Filter가 보다 정확한 객체의 위치를 예측하는데 적용되었다.

실험 결과, 본 논문에서 제안한 알고리즘은 객체의 모양이

회전이나 혹은 보는 각도에 의해 왜곡될 경우에도 비교적 정확하게 객체를 인식하였고, 부분적인 가림이나 배경이 객체로 잘못 인식된 상황에서도 정상적으로 동작하는 모습을 보였다.

주요어: 특징점, SURF, 파티클 필터, 다중객체추적

학 번: 2009-23113