Object Tracking Based on Extended SURF and Particle Filter

Min Niu[1], Xiaobo Mao[1], Jing Liang[1], and Ben Niu[2]

[1] School of Electrical Engineering, Zhengzhou University, Zhengzhou, China
angel12318@126.com, {mail-mxb,liangjing}@zzu.edu.cn
[2] College of Management, Shenzhen University, Shenzhen, China
drniuben@gmail.com

Abstract. Under complex environment, it is difficult to track target successfully by single feature. To solve this problem, the paper propose a novel object tracking approach which fuses color and SURF(Speeded Up Robust Features) in the frame of particle filter. SURF remain invariant for illumination, scale and affine. Add color to make up for the shortcoming(SURF is based on image gray scale information.). It not only maintains the characteristics of SURF, but also makes use of the image color information. The experimental results prove that the proposed method is real-time and robust in different scenes.

Keywords: SURF, Color Feature, Particle Filter, Object Tracking.

1 Introduction

Video target tracking is an important research subject in the field of machine vision. Many scholars has carried on the extensive research to target tracking, a variety of algorithms are proposed, but most algorithms are based on single feature, which are effective in a particular environment. It may result in decline in accuracy even failure when the target or environment changes. For the situation, multiple features have been widely used in tracking systems. However, the performance still depend on single feature even the fusion[1].

Compared with color, shape and texture etc features, SIFT (scale invariant feature transform) proposed by D.G. Lowe in 2004 is more capable to remain invariable for illumination, scale and affine [2-4]. SIFT is the local characteristics of the image, and although it has more advantages over other features, the matching algorithm based on SIFT is obliged to deal with complex computing problems and long time consuming. Even though Grabner etc put forwards an idea that increases the calculating velocity by integral image at cost of the superiority [5]. In 2006, Bay etc provided SURF (speeded up robust features) [6] algorithm on the foundation of SIFT. SURF has good adaptability for zooming, small perspective changes, noise, and brightness changes, like SIFT. In addition, it can take short time to accomplish SURF matching algorithm. So SURF has excellent performance not only in speed but also in accuracy [7-10].

D.-S. Huang et al. (Eds.): ICIC 2013, LNAI 7996, pp. 649–657, 2013.
© Springer-Verlag Berlin Heidelberg 2013

However, SURF only uses the gray information of images, ignores color information, so it is difficult to identify and match the target with similar texture. The paper fuses color vector and build extend SURF descriptor to solve the problem.

When tracking object, it needs to process a large of invalid information to determine the optimal matching position if we calculate all the pixels in images directly, and it is heavy computation and time-consuming. Therefore, it can improve the efficiency of tracking algorithm if we adopt a certain method to reduce the search area of candidate target. Due to the particle filter can predict target location in the next frame image and realize the state estimation of nonlinear non-Gaussian systems, it is commonly used in tracking algorithm at present. The paper uses extend SURF descriptor to set up the target model and particle filter to search candidate target location and the novel method is robust and real-time.

2 Particle Filter

For particle filter, its fundamental is Monte Carlo method. It uses a set of particles with weights $\{(x^i_k, \omega^i_k), i=1...n\}$ to estimate the posterior density $p(x_k \mid y_{1:k})$. x^i_k describes the particle, namely the target state. ω^i_k represents the weight, and $\sum_{i=1}^{N} \omega_k^i = 1$.

In ideal conditions, all the weights should be $1/N$, the particles should be random sampled from the posterior density, but it is impossible in the actual situation. So sampling new particles from the proposed density $\pi\ (x^i_k \mid x^i_{k-1}, y_{1:k})$ which is similar with posterior density, and computing weights again to make up the difference between the proposed density and the posterior density. Then normalizing the weights, the posterior density is computed as:

$$p(x_k | y_{1:k}) \approx \sum_{i=1}^{N} \omega_k^i \delta(x_k - x_k^i) \tag{1}$$

Where, δ(*) is DE carat function. Weights updating formula as follows:

$$\omega_k^i \propto \omega_{k-1}^i \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{\pi(x_k^i | x_{k-1}^i, y_{1:k})} \tag{2}$$

Where, it is significant how to choose the proposed density $\pi\ (x^i_k \mid x^i_{k-1}, y_{1:k})$. It is easy to make $\pi\ (x^i_k \mid x^i_{k-1}, y_{1:k}) = p(x^i_k \mid x^i_{k-1})$ for practical application. Weights updating is computed as:

$$\omega_k^i \propto \omega_{k-1}^i p(y_k | x_k^i) \tag{3}$$

3 SURF Algorithm

3.1 Feature Point Detection

Image pyramid is used to express scale space usually in the field of vision. It is variable for image size when using the traditional way to conduce a scale space. Next, reuse Gaussian filer to smooth each layer image. In order to speed up, we adopt the box filters increasing gradually which approximates second-order Gaussian filter and the integral image to make the convolution to form different scales of image pyramid.

SURF algorithm detects the extreme points in images by computing the Hessian matrix. To a point (x,y) in image I, at the scale σ the Hessian matrix is defined as follows:

$$H(x,y,\sigma)=\begin{bmatrix} L_{xx}(x,y,\sigma) & L_{xy}(x,y,\sigma) \\ L_{xy}(x,y,\sigma) & L_{yy}(x,y,\sigma) \end{bmatrix} \tag{4}$$

Where, $L_{xy}(x,y,\sigma)$ is the convolution between Gaussian second order derivative $\frac{\partial}{\partial x^2} g(x,y,\sigma)$ and the pixel value I(x,y) in image I. Similarly for L_{xy} and L_{yy}.. Because the box filter approximate second order Gaussian filter, for convenience of calculation, we use the convolution of I(x,y) with box filter instead, the results are described by D_{xx}, D_{xy}, D_{yy}. Introduce weight to reduce the error between the approximation and the accurate value. So the determinant of Hessian matrix is computed as follows:

$$\det H = D_{xx}D_{yy} - (0.9D_{xy})^2 \tag{5}$$

The extreme point obtained by calculating the determinant of Hessian matrix, 8 pixel points at the same scale and 18 pixel ones at upper and lower adjacent scales form a 3*3*3 neighborhood. If the extreme point larger or smaller than the remaining 26 pixel ones, it is the feature point. Then localize feature point and over scale by interpolating it in scale and image space with the method proposed by Brown et al.

3.2 SURF Feature Descriptor

Firstly, calculating the Harr wavelet response (Harr response side length is 4σ, σ is scale at which the feature point.) in X and Y direct in a circular neighborhood of radius 6σ around the feature point, then the responses are weighted with a Gaussian (2σ) centered on the feature point. Next computing the weighted sum of all responses within a sliding orientation window covering 60° angle to form a new vector, the longest vector over all windows is defined as orientation of the feature point, the process is shown in figure 1.

Constructing a square region centered on the feature point and oriented along the orientation confirmed before, with the size 20σ.The region is divided into 4*4 square sub-regions evenly. In each of sub-regions, we compute the Harr responses at 5σ scale in orientation selected before and its vertical direction, which are represented by d_x and d_y respectively. In order to increase the robustness, d_x and d_y are weighted with a Gaussian(3.3σ) centered on the feature point. Every sub-region has a four-dimensional descriptor vector.

$$V = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right) \tag{6}$$

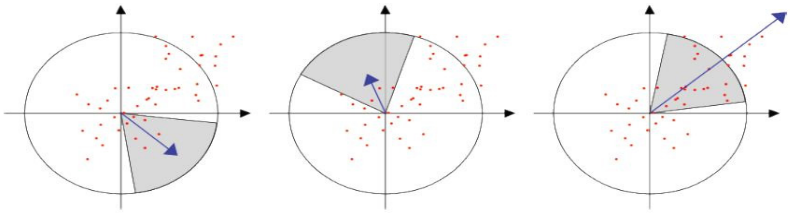

Fig. 1. Diagram for confirming orientation of the feature point

For every feature point, its descriptor vector is 64-dimensional.Figure 2 shows the descriptor vector building.

3.3 Extend SURF Descriptor

SURF descriptor only uses the gray information of images, ignores color information, so it is difficult to identify and match the target with similar texture. So the paper adopts a new kind of descriptor based on color. Namely, fuses color vector to build extend SURF descriptor to solve the problem.

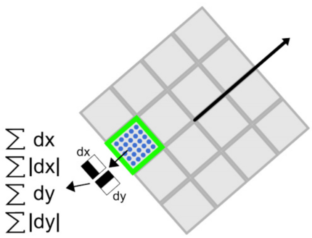

Fig. 2. Diagram for building descriptor vector

Because SURF has good adaptability for zooming, small perspective changes, noise, and brightness changes, and color descriptor can not change structure of SURF descriptor, so extend SURF one is more robust. To reduce the influence of illumination change during target tracking, the paper uses HSV color model [11]. Where, H is hue, S is saturation, they are not sensitive to light. V is value and opposite. And three components are independent mutually. So we only compute the sum of H and S to obtain $\sum h$ and $\sum s$. Therefore, every sub-region has a 2-dimensional color vector.

$$C = \left[\sum h \quad \sum s \right] \tag{7}$$

16 sub-regions form 32-dimensional color descriptor vector. Add color vector C to SURF descriptor vector V to construct the extend SURF descriptor vector with 96-dimensional.

3.4 Feature Point Matching

The paper adopts the nearest neighbor matching vector method to match feature point [12]. Given A and B are the set of feature points in two images I_1 and I_2, a_j is a point in A, compute the Euclidean distances between a_j and all the points in B, call d_1 and d_2 the nearest distance and next nearest distance regularly, corresponding to two feature points b_j and b^*_j, if $d_1 \leqslant \mu d_2$ (μ=0.65), a_j and b_j match. Otherwise, we discard the feature point. Traversing all the feature points in A can find out all possible matching feature point pairs.

4 Target Tracking

4.1 Target Motion Model

Supposing object keeps a constant velocity, its movement is small for the same object of adjacent frames. The paper adopts a rectangle to represent the object area, $X=(x,y,v_x,v_y,h_x,h_y)^T$ is described as motional state of an object, (x,y) are the center coordinates of the object, (v_x,v_y) is velocity of movement in x and y direction, (h_x,h_y) are height and width of object area. The dynamic equation of the system is shown in formula 8.

$$X_k = AX_{k-1} + v_k \tag{8}$$

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Where, A is state transfer matrix of the system, v_k is system noise.

4.2 Similarity Measurement

The paper adopts the extend SURF feature to track object, we draw a conclusion that similarity between reference target and candidate one is related to the number of matching feature point pairs and the Euclidean distance between matching feature points.Supposing m is the number of matching feature point pairs between candidate target S_k and reference one S_0 at time k, dis_t is the Euclidean distance between a matching feature point pair, then the average Euclidean distance between two objects is defined as:

$$D_k = \sum_{t=1}^{m} dis_t / m \tag{9}$$

We judge the similarity degree between targets in two frames by combining the number and Euclidean distance, and observation likelihood function is defines as:

$$p\left(y_t \middle| x_t\right) = 1 - \exp\left(-mD_k\right) \tag{10}$$

4.3 Algorithm Description

(1) In the initial frame, we choose the object tracking area manually, calculate the extend SURF feature vector contained in the object as reference template, and sample a set of particles with weights $\{(x^i_0, 1/N), i=1...N\}$ from the proposed density $\pi(x^i_k \mid x^i_{k-1}, y_{1:k})$, for k=0. According to the actual conditions, set the initialization parameter, such as target coordinates, velocity and rectangular frame size;
(2) Input next frame image, exploit formula 7 to acquire a new particle collection for time k;
(3) Compute the extend SURF feature vector in image for time k.

(4) Utilize formula 3 and 9 to calculate the weight of each particle and to normalize it as follows:

$$\omega_k^i = \omega_k^i / \sum_{i=1}^{N} \omega_k^i \tag{11}$$

(5) If $1/\sum_{i=1}^{N}\left(\omega_k^i\right)^2 < 2N/3$, resample to generate a new set of particles: $\{(x^i_k, 1/N), i=1...N\}$.

(6) Output the state estimation: $x_k = \sum_{i=1}^{N} \omega_k^i x_k^i$;

(7) $k=k+1$, if go on tracking, switch(2); otherwise end.

5 Experiment Analysis

To validate the superiority of the algorithm above, several video of different scenes are tested, and the results are compared with SURF algorithm. The environment of experiment is personal computer, AMD Athlon, 2.8GHz, 2GB Memory, OS Windows XP, and the proposed algorithm is programmed by VC++6.0 and OpenCV.

The first video sequences are people tracking on a simple scenes, the experiment result is as in figure 3. We choose the 120th frame, the 150th frame, the 170th frame, the 480th frame to analyze the tracking performance. During tracking, he human body target selected is interfered with similar color can, moreover the distance between target and vidicon is bigger gradually when human is walking. So some changes have taken place in scale with target appearing in the lens. The experiment result shows the extend SURF algorithm overcomes the interference with similar color, and retains invariant for scale, achieves good tracking performance.

Fig. 3. Human tracking on a simple scene

(a) Human tracking based on SURF

(b) Human tracking based on the algorithm proposed

Fig. 4. Human tracking under occlusions

The second video is human tracking with occlusion. Two persons are running in the playground, a person in brown was occluded by the other one in blue now and then. From the tracking results in figure 4, we can see that the algorithm base on SURF is distracted seriously and lose the target when occlusion occurs, but the algorithm based on the extend SURF works well throughout the whole sequence. Reasons for the case were analyzed basing on experimental phenomenon. SURF feature only uses the gray information of images, ignores color information, so it is so difficult to identify and match the target with similar texture that miss real target. Extend SURF feature not only maintains the characteristics of SURF, but also makes use of color information, so the proposed algorithm can track the object selected accurately when the above situation exists.

The third video is car tracking in a complex environment, the light is dark and there are some other ones near the object. We analyze the 210th frame, the 232nd frame, the 301st frame, the 340th in video as shown in figure 5. The experiment result indicates the proposed algorithm can track object accurately.

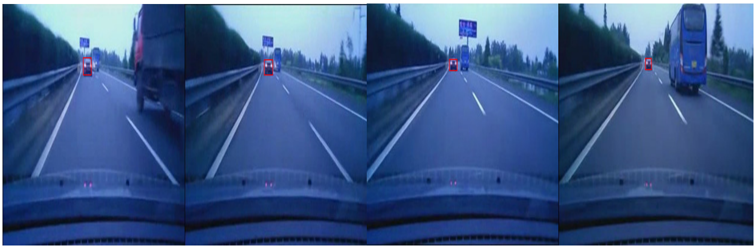

Fig. 5. Car tracking under the complex environment

6 Conclusions

The paper proposed a novel object tracking algorithm by integrating color and SURF (Speeded Up Robust Features) in the frame of particle filter. The proposed algorithm has been tested on different scenes and proved stability and robustness. The results show it is stable and robust. In our future work, we also want to use some new swam intelligence based algorithms [13,14] to solve the problem of object tracking.

Acknowledgment. This research is partially supported by the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20114101110005), Science and technology research plan funded key projects province department of education of Henan (Grant No. 12A410002), the National Natural Science Foundation of China(Grants nos.71001072, 71271140, 71240015), and the Natural Science Foundation of Guangdong Province (Grants nos. S2012010008668, 9451806001002294).

References

1. Wang, H., Wang, J.T., Ren, M.W.: A new robust object tracking algorithm by fusing multi-features. Journal of Image and Graphics 14, 489–495 (2009)
2. David, G.L.: object recognition from local scale-invariant features. In:7th IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157 (1999)

3. David, G.L.: Distinctive image features from scale-invariant keypoints. International Jornal of Computer Vision 60, 91–110 (2004)
4. Zeng, L., Wang, Y.Q., Tan, J.B.: Improved algorithm for SIFT feature extraction and matching. Optics and Precision Engineering 19, 1392–1397 (2011)
5. Tong, R.Q., Huang, Y.Q., Tian, R.J.: SURF algorithm and its Detection effect on object tracking. Journal of Southwest University of Science and Technology 26, 63–67 (2011)
6. Grabner, M., Grabner, H., Bischof, H.: Fast approximated SIFT. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3851, pp. 918–927. Springer, Heidelberg (2006)
7. Bay, H., Ess, A., Tuytelaars, T.: Speeded-up robust features(SURF). Computer Vision and Image Understanding 110, 404–417 (2008)
8. Dong, H., Han, D.Y.: Research of image matching algorithm based on SURF features. In: International Conference on Computer science and Information Processing, pp. 1140–1143 (2012)
9. Li, J.G., Wang, T., Zhang, Y.M.: Face detection using SURF cascade. In: IEEE International Conference on Computer Vision Workshops, pp. 2183–2190 (2011)
10. Sun, X., Fu, K., Wang, H.J.: High resolution remote sensing image understanding, 68–74 (2011)
11. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. Pattern Analysis and Machine Intelligence 24, 603–619 (2002)
12. Marius, M., David, G.L.: Fast approximate nearest neighbors with automatic algorithm configuration. In: International Conference on Computer Vision Theory and Applications, VISAPP, pp. 331–340 (2009)
13. Niu, B., Wang, H.: Bacterial colony optimization. Discrete Dynamics in Nature and Society, 1–28 (2012)
14. Niu, B., Fan, Y., Xiao, H., Xue, B.: Bacterial foraging-Based approaches to portfolio optimization with liquidity risk. Neurocomputing 98(3), 90–100 (2012)