

Research Article

Large-Screen Interactive Imaging System with Switching Federated Filter Method Based on 3D Sensor

Lei Yu ^{1,2,3} and Junyi Hou¹

¹School of Mechanical and Electric Engineering, Soochow University, Suzhou, China

²Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education (Nanjing University of Science and Technology), Nanjing, China

³Collaborative Innovation Center of Industrial Energy-Saving and Power Quality Control (Anhui University), Hefei, China

Correspondence should be addressed to Lei Yu; yu_lei@suda.edu.cn

Received 8 August 2018; Revised 5 November 2018; Accepted 22 November 2018; Published 4 December 2018

Academic Editor: Carlos F. Aguilar-Ibáñez

Copyright © 2018 Lei Yu and Junyi Hou. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Large-screen human-computer interaction technology is reflected in all aspects of daily life. The dynamic gesture tracking algorithm commonly used in recent large-screen interactive technologies demonstrates compelling results but suffers from accuracy and real-time problems. This paper systematically addresses these issues by a switching federated filter method that combines particle filtering and Mean Shifting algorithms based on a 3D sensor. Compared with several algorithms, the results show that the one-hand and two-hand large-screen gesture tracking based on the switched federated filtering algorithm work well, and there is no tracking failure and loss of target. Therefore, the switching federated tracking positioning algorithm can be well applied to the design of human-computer interaction system and provide new ideas for future human-computer interaction.

1. Introduction

The definition of human-computer interaction technology is based on a certain software program through the corresponding input and output devices, and the organic combination of computer and human operations is employed to realize the related technology of interactive communication [1–17]. The current large-screen human-computer interaction technology is mainly divided into the following categories: (1) infrared frame touch interaction; (2) capacitive touch-based electronic screen touch interaction; (3) speech recognition interaction; (4) computer vision interaction. These types of technologies have their own advantages and disadvantages in terms of cost, accuracy, interactivity, etc. [3–7]. Touch interactions based on infrared frame and capacitive touch-based electronic screens are all based on interactive hardware fixed at a certain place. Therefore, not only are such man-machine interaction operations costly, but also the range and accuracy of the interaction are affected by the hardware constraints [4–8]. Also, interactions based on speech recognition are often limited by complex noisy environments which are mostly used in the small-scale civil-computer

interaction field and therefore have a smaller application area. The human-machine interaction system based on machine vision is mainly based on sensors and other devices to collect images or signals and then completes the signal preprocessing operations on the image signals collected by the sensors or digital signals through an independently designed human-computer interaction software. The targets to be tracked are segmented from the background, and a series of operations such as target tracking and motion recognition are used to complete the corresponding human-computer interaction [4, 7–11]. Therefore, the cost of human-computer interaction system designed based on computer vision is relatively low, and the interaction effect can be well obtained.

Hand gesture recognition methods based on computer vision are mainly divided into two categories: one is an analysis method based on a 3D model, and the other is based on a 2D image. The analysis method with the 3D model needs to establish a parametric model that describes gestures. Because it can provide 3D data, a more accurate gesture model can be established. However, the method has many parameters and high computational complexity, and it is difficult to achieve real-time results in current

technologies. Also, with the two-dimensional image method, it is mainly to analyze the performance of the image and extract the effective hand features for identification. Because of the loss of the third 3D space information, the gesture model cannot be effectively established for the described features. Besides, its parameters are less, so it can meet the requirements of real-time processing. These two methods can hardly guarantee the balance between system parameters and real-time performance.

This paper proposes a large-screen interactive imaging system with switching federated filter method based on 3D sensor and validates the tracking results with the independently developed gesture position tracking platform. The main innovations of this paper are as follows: (1) An improved switching federated filter algorithm combining particle filtering and Mean Shift is introduced into the field of large-screen gesture tracking to track dynamic gestures; (2) the self-developed gesture tracking platform and 3D interactive software are combined to observe the gesture tracking effects; (3) the large-screen single-hand and two-hand gesture tracking based on the switching federated filtering algorithm work well and there is no phenomenon of tracking failure and losing the target.

The structure of this paper is as follows: An improved switching federated filter algorithm combining particle filtering and Mean Shift is employed to track and locate the dynamic gestures in Section 2. Section 3 presents the dynamic gesture tracking and positioning experiment verification based on switching federated filter method. Finally, conclusion is given in Section 4.

2. Dynamic Gesture Tracking Algorithm with Switching Federated Filter

Common gesture tracking algorithms cannot handle the dynamic gesture tracking problem in complex environments. This paper has developed an improved switching federated filter algorithm that combines particle filtering and Mean Shift algorithm. In the case of slow movement of the human hand, after the uniform displacement of the particles, the particles drift to the dynamic gesture area except for a small number of particles. The gesture position can then be obtained without subsequent prediction of the particles, so that the running time of algorithm can be saved. The average value of the particles will drift when the movement of the hand is fast or there is occlusion. If the region of the gesture cannot be detected, the particle will return to the condition before the drift, waiting for the algorithm to perform corresponding processing on the next frame. Therefore, how to select the corresponding filtering algorithm under different conditions is the key problem. In this paper, the switching system scheme is introduced into the filter for the first time, and applied to the large-screen interactive imaging system.

A switching system consists of a series of sequential or discrete differential equations subsystems and switching rules or a switching strategy [4, 18–34]. If the entire filtering process is regarded as a hybrid dynamic system, then each

filter algorithm can be regarded as a subsystem of the system. The proposed system is constructed by coupling the switching system with arbitrary switching rules and using two subsystems composed of arbitrary switching filter system. In the case when the movement of the human hands is fast, the filter subsystems of particle filter algorithm will be chosen; when the human hand moves slowly, the filter subsystems of Mean Shift algorithm will be selected. The meaning of the switching signal is

$$\sum = \{x_0; (i_0, t_0), (i_1, t_1), \dots, (i_j, t_j), \dots, | t_j \in M, j \in N\}, \quad (1)$$

where (i_j, t_j) denotes that the i_j -th subsystem is switched on at t_j and the i_{j+1} -th subsystem is switched off at t_{j+1} . Thus, in finite time, the switching sequence \sum is finite, and there exists no state transition during the switching moment, where t_0 is the initial time and $t_j > 0$ is the j -th switching time. When $t \in [t_j, t_{j+1})$, the trajectory of the switched nonlinear system is produced by the i_{j+1} -th subsystem, defining $T_{\min} = \min\{T = t_j - t_{j-1}\}$, as the minimum interval time of the i_k -th subsystem.

The current common particle filter implementation framework is based on resampling and sequential importance sampling, which can be called sampling importance resampling filter [35–37]. Therefore, it mainly consists of three basic operation steps to form an iterative cycle as follows:

Step 1 (sampling). Based on the Bayesian posteriori estimation and state transfer equation at the previous moment to achieve the purpose of updating the particle state, the predicted distribution $p(x_k | y_{1:k-1})$ (i.e., prior distribution) can be achieved.

Step 2 (weight update). Based on the latest observed information y_k , employ the likelihood function to perform a corresponding weight update operation for the weights:

$$w_{kj}^{(i)} \propto w_{(k-1),j}^{(i)} \frac{p_j(y_k | x_k^{(i)}) p_j(x_k^{(i)} | x_{k-1}^{(i)})}{q_j(x_k^{(i)})}, \quad (2)$$

where w represents the weight of the particle, q represents the importance distribution function, and p represents the posterior probability density.

Step 3 (resampling). Based on the principle of identity distribution, the resampling operation is performed on the updated particle combinations after the weights update operation is completed. A new set of particles with most of the particle weights can then be obtained. The number of times N that particles weights $w_k^{(i)}$ with nonbiased weight values are sampled is expected to be

$$E_j(N_{kj}^{(i)} | w_{kj}^{(i)}, N_j) = N_j w_{kj}^{(i)}. \quad (3)$$

Meanwhile, the Mean Shift algorithm as the other filter subsystems is a process that uses nonparametric density estimation, which was used to perform iterative search based on feature spatial gradient directions and then obtain sample

data with local density maximum [12–14]. Compared with other tracking methods based on optimized matching search, the advantage of the Mean Shift filter algorithm is that the method does not need to know the characteristics of the feature space in advance; the only requirement is the relevant sample data. The filtering is performed according to these sample points. Therefore, this kind of algorithm has less computational complexity. The implementation of the algorithm is not difficult, and the algorithm has strong real-time performance to meet the systems with high real-time requirements. In the Mean Shift algorithm, the most critical thing is to calculate the offset mean for each point and then update the position of the point based on the newly calculated offset mean.

For n sample points x_i , $i = 1, \dots, n$, in a given d -dimensional space R^d , the basic form of the Mean Shift vector for point x is

$$M_{hj}(x) = \frac{1}{k} \sum_{x_i \in S_h} (x_i - x), \quad (4)$$

where S_h is a high-dimensional sphere with a radius of h . S_h is defined as

$$S_{hj}(x) = \{y \mid (y - x)(y - x)^T \leq h^2\}. \quad (5)$$

The form of Mean Shift has a fundamental problem: in the region of S_h , each point has the same effect on x . In fact, this effect is related to the distance between x and each point. The importance of different samples is different. Aiming at the abovementioned considerations, the kernel function and the sample weight are added to the basic Mean Shift vector form, and the following modified Mean Shift vector form is obtained:

$$M_{hj}(x) = \frac{\sum_{i=1}^n G_{Hj}(x_i - x) w_j(x_i) (x_i - x)}{\sum_{i=1}^n G_{Hj}(x_i - x) w_j(x_i)}. \quad (6)$$

Among them,

$$G_{Hj}(x_i - x) = |H_j|^{-1/2} G_j(H_j^{-1/2}(x_i - x)). \quad (7)$$

$$\begin{aligned} \nabla \hat{f}_j(x) &= \frac{2 \sum_{i=1}^n (x_i - x) G_j(\|(x_i - x)/h\|^2) w_j(x_i)}{h^{d+2} \sum_{i=1}^n w_j(x_i)} \\ &= \frac{2}{h^2} \left[\frac{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i)}{h^d \sum_{i=1}^n w_j(x_i)} \right] \cdot \left[\frac{\sum_{i=1}^n (x_i - x) G_j(\|(x_i - x)/h\|^2) w_j(x_i)}{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i)} \right]. \end{aligned} \quad (12)$$

Among them, the second square brackets mean the Mean Shift vector, which is proportional to the probability density gradient. Mean Shift vector correction results are as follows:

$$M_{hj}(x) = \frac{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i) x_i}{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i)} - x. \quad (13)$$

$G(x)$ is a unit of kernel function. H is a positive definite symmetric $d \times d$ matrix, called the bandwidth matrix, which is a diagonal matrix. $w(x_i) \geq 0$ is the weight of each sample. The form of the diagonal matrix H is

$$H = \begin{pmatrix} h_1^2 & 0 & \cdots & 0 \\ 0 & h_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_d^2 \end{pmatrix}_{d \times d}. \quad (8)$$

The above Mean Shift vector can be rewritten as

$$M_{hj}(x) = \frac{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i) (x_i - x)}{\sum_{i=1}^n G_j(\|(x_i - x)/h\|^2) w_j(x_i)}. \quad (9)$$

Mean Shift vector $M_h(x)$ is a normalized probability density gradient. In the Mean Shift algorithm, the probability density is actually used to obtain the local optimal solution of the probability density. For a probability density function $f(x)$, the known d -dimensional space of n sampling points x_i , $i = 1, \dots, n$, the kernel function of $f(x)$ is estimated:

$$\hat{f}_j(x) = \frac{\sum_{i=1}^n K_j(\|(x_i - x)/h\|^2) w_j(x_i)}{h^d \sum_{i=1}^n w_j(x_i)}, \quad (10)$$

where $w(x_i \geq 0)$ is a weight assigned to the sample point x_i and $K(x)$ is a kernel function. The estimate of the gradient $\nabla f(x)$ of the probability density function $f(x)$ is

$$\nabla \hat{f}_j(x) = \frac{2 \sum_{i=1}^n (x - x_i) k'(\|(x_i - x)/h\|^2) w_j(x_i)}{h^{d+2} \sum_{i=1}^n w_j(x_i)}. \quad (11)$$

Let $g(x) = -k'(x)$ and $G(x) = g(\|x\|^2)$; there are

Considering $m_h(x) = \frac{\sum_{i=1}^n G(\|(x_i - x)/h\|^2) w_j(x_i) x_i}{\sum_{i=1}^n G(\|(x_i - x)/h\|^2) w_j(x_i)}$, then the abovementioned formula becomes

$$M_{hj}(x) = m_{hj}(x) + x. \quad (14)$$

The process after the integration of the specific algorithm is as follows.

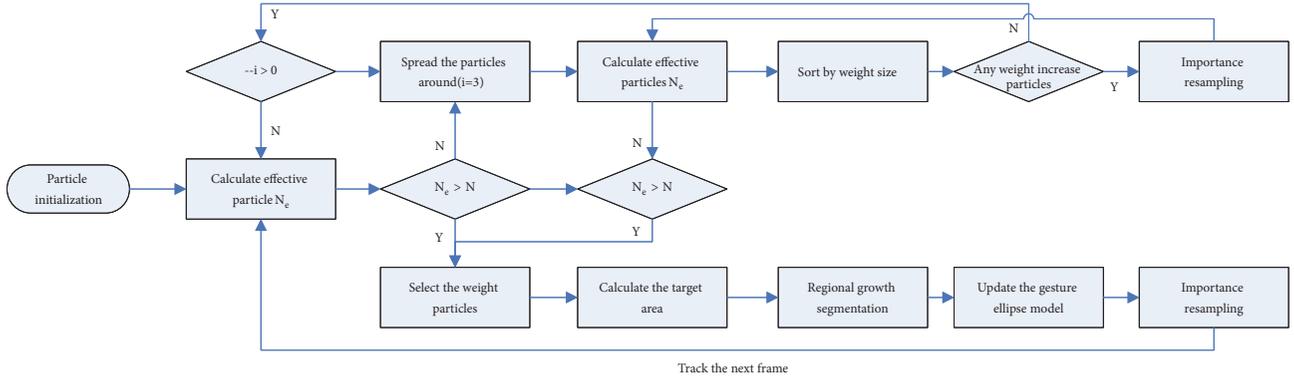


FIGURE 1: The algorithm flowchart with switching federated filter method.

Step 1. In the initial frame, all the particles are distributed in the gesture area according to the Gaussian distribution.

Step 2. In the next frame, all particles are mean shifted, the Pasturian after shifting is taken as the weight of the particle, the weights are normalized, and the number of effective particles (N_e) is calculated.

Step 3. Determine whether the number of effective particles is greater than the threshold N .

Step 4. If the effective number of particles is greater than the set threshold ($N_e > N$), the target area is in the particle distribution area. Then, all the particles are sorted by weight from the highest to the lowest. Pick a particle whose weight is greater than the threshold to calculate the location of the gesture area.

Step 5. Define the seed as the center of the region calculated in Step 4, the area growth is segmented, the ellipse model of the gesture is established with the separated area, and resample the significance of the particles.

Step 6. If the effective number of particles is less than the set threshold ($N_e < N$), particles are spread by using a four-week Gaussian distribution.

Step 7. Use Mean Shift algorithm on all particles, the Pasturian after shifting is taken as the weight of the particle, the weights are normalized, and the number of the effective particles (N_e) is calculated.

Step 8. If the effective number of particles is greater than the set threshold ($N_e > N$), go to Step 4.

Step 9. If the effective number of particles is less than the set threshold ($N_e < N$), determine whether the particle weight has increased.

Step 10. If the weight of some particle increased, it means that a small part of the particles have spread to the gesture area. Resample the significance of the particles and then go to Step 7. If the particle weight do not increase, and the particles

spread to the surrounding less than 3 times, then go to Step 6. If they have been spread 3 times, it is considered that there is no target area in the frame, the particles are restored to the state before being shifted, and then go to the next frame.

The algorithm flowchart is shown in Figure 1.

3. Dynamic Gesture Tracking and Positioning Experiment Verification Based on Switching Federated Filter Method

In this paper, two experiments are designed to verify the gesture tracking and positioning effect of the proposed switched federated filtering algorithm. The first experiment is based on the self-developed human-computer interaction positioning software platform and observes the filtering effect of the switching federated filter algorithm. The second one is the gesture tracking and positioning experiment based on 3D interactive software. It compares the switching federated filter algorithm with several algorithms and observes the performance of the different algorithms in tracking accuracy and tracking time.

3.1. Gesture Tracking and Positioning Experiment Based on Self-Developed Human-Computer Interactive Platform. The self-developed human-computer interactive positioning software platform system generates an interactive operation interface through a projector on an arbitrary wall or curtain. The Kinect sensor is generally fixed at a range of 1-3 meters directly above the screen and 5-6 centimeters far from the wall surface [4, 6, 8]. The projector is placed in front of the wall, and the size of the projection surface can be changed by adjusting the distance. The physical map is shown in Figure 2. The operator stands in the interactive area and can use the limb to click, swing, rotate, and grab in front of the projection surface.

Figures 3 and 4 show the effect of gesture positioning before filtering and after switching federated filtering, respectively. As shown in Figure 3, the image on the left taken by Kinect sensor is used to recognize the position of the gesture, and the position and relative coordinates of the gesture obtained after being calculated by the human-computer

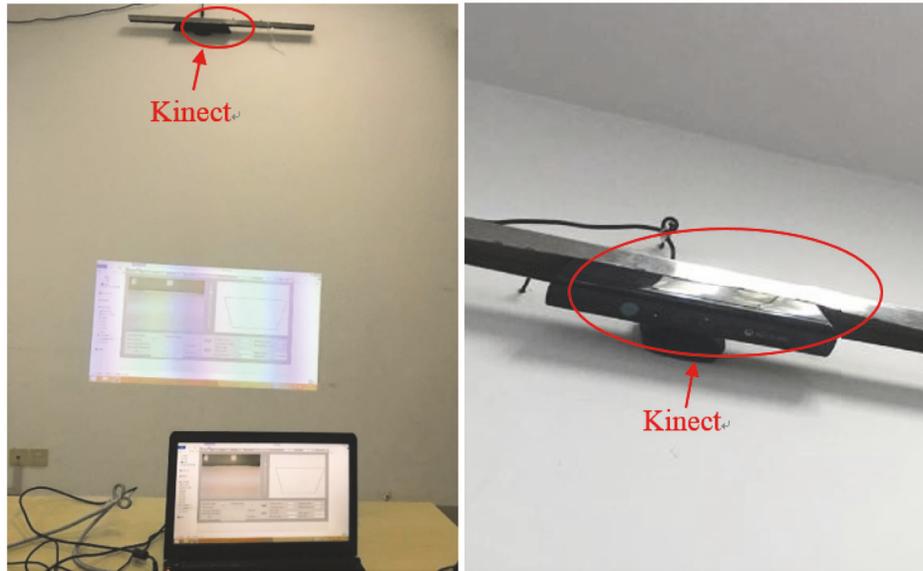


FIGURE 2: The self-developed human-computer interactive positioning software platform.

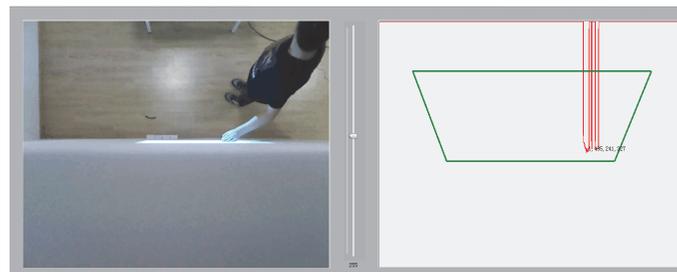


FIGURE 3: Gesture tracking and positioning effect without filter algorithm.

interaction software platform can be seen on the right side. The green trapezoidal area represents the interactive area recognized by the sensor. The red line represents the position of the recognized gesture. The black numbers in Figure 3 next to the cusp is the relative 2D coordinate of the gesture in the interactive area.

From Figures 3 and 4, the multiple red line cusps in Figure 3 indicate that the platform is unable to filter out interference to determine the relative gesture position. After the switching federated filter algorithm is added, only one red line appears in Figure 4. So the platform can accurately determine the relative gesture position. Therefore, the proposed switching filter algorithm can improve the accuracy of the gesture tracking algorithm.

3.2. Gesture Tracking and Positioning Experiment Based on 3D Interactive Software. There are many commonly used target tracking algorithms. This paper uses the typical Cam-shift algorithm to compare the actual application with the algorithm on the large screen. Cam-shift algorithm is the commonly used gesture tracking algorithm, which is good for tracking solid objects in a black-and-white background. However, the contrast between the background color and

the target is not obvious, and the tracking effect is poor [15–17, 38–40]. To verify the effectiveness of the switching federated filter algorithm, a gesture tracking and positioning experiment based on the 3D interactive software is presented. The specific operation method is to change the large-screen projection system based on the independent developed interactive platform to a large-screen splicing display system and use the 3D interactive software for real-time display.

This section compares the Cam-shift algorithm with the switching federated tracking algorithm presented in the simulation. Gesture trajectories tracked by two different algorithms are transmitted to the computer. The mouse function of trace tracking in the 3D interactive software can convert the gesture trace map into the trace of the mouse origin and display it on the mosaic screen. The tracking effect of different algorithms can be observed according to the mouse trajectory on the splicing screen. The interactive software based on the software platform can operate the position of the gesture as the mouse position, so the mouse track is the gesture tracking trajectory. Figure 5 is an actual effect diagram of one-hand gesture tracking using the proposed switching federated filter algorithm, and Figure 6 is the corresponding computer screenshot for Figure 5. Figure 7 is the one-hand gesture

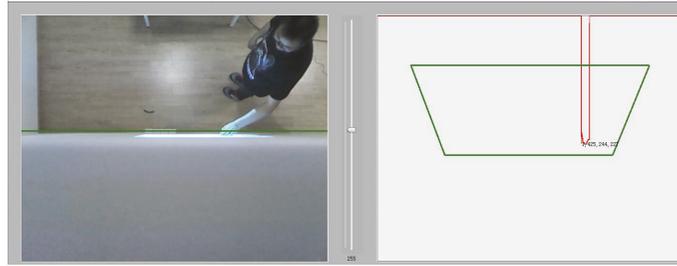


FIGURE 4: Gesture tracking and positioning effect with the federated tracking filter algorithm.



FIGURE 5: The one-hand gesture tracking effect based on switching federated filter algorithm.

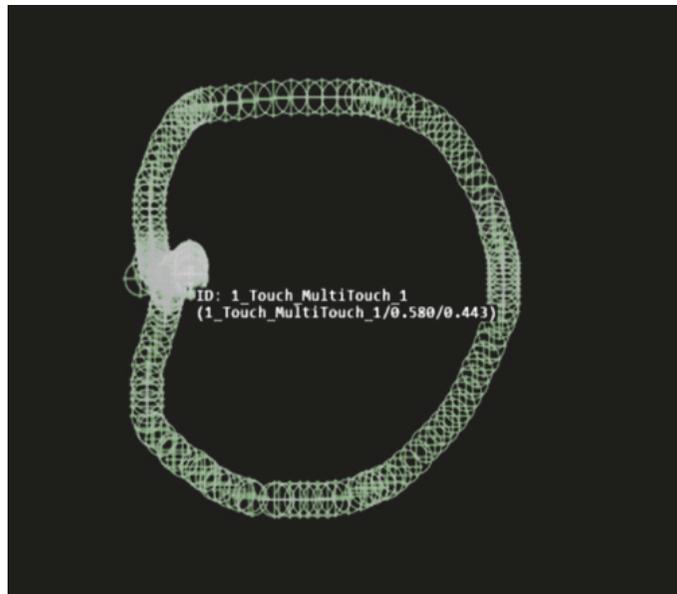


FIGURE 6: The computer screenshot of one-hand gesture tracking based on switching federated filter algorithm.

tracking effect based on Cam-shift algorithm, and Figure 8 is the corresponding computer screenshot for Figure 7.

As can be seen from Figure 5, when one hand moves in front of the screen, the gesture tracking effect is well obtained. The 3D interactive software can accurately track the gesture track and display the gesture track. From Figure 6, there is

no tracking failed point on the tracking trajectory (the failed trajectory point is displayed as a red exclamation point), so the gesture tracking based on the switching federated filter algorithm works well. Figure 7 is a schematic diagram of a single-handed operation based on the Cam-shift algorithm. When one single hand moves in front of the screen, a tracking

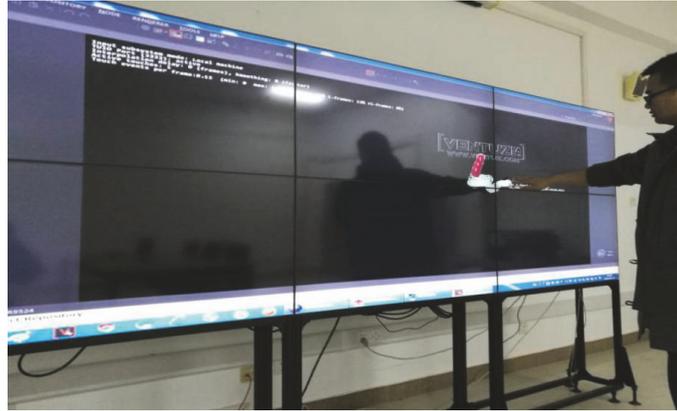


FIGURE 7: The one-hand gesture tracking effect based on Cam-shift algorithm.

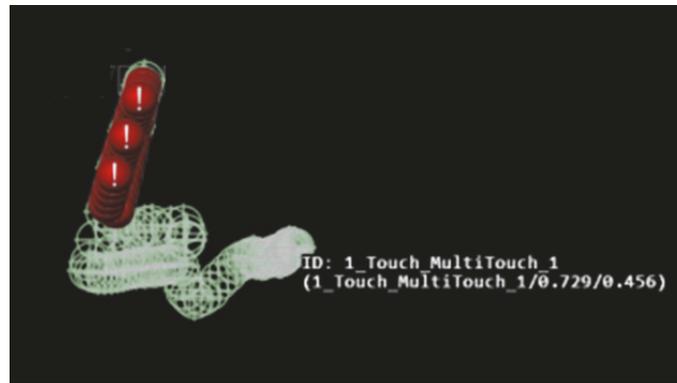


FIGURE 8: The computer screenshot of one-hand gesture tracking based on Cam-shift algorithm.

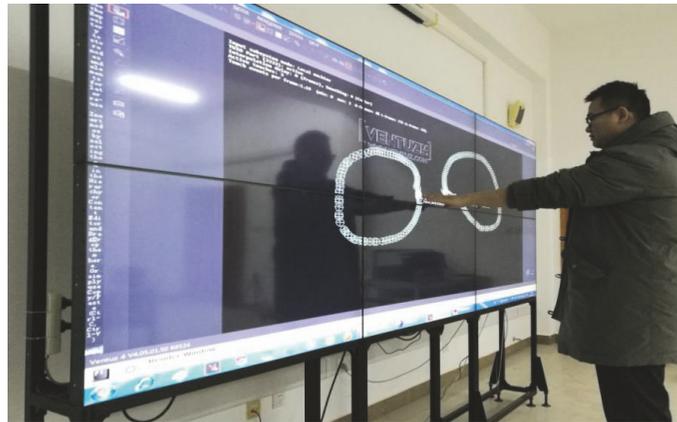


FIGURE 9: The hands gesture tracking effect based on switching federated filter algorithm.

failure phenomenon occurs in the gesture tracking effect. The computer screenshot of the one-hand gesture trajectory is shown in Figure 8. It can be seen from Figure 8 that the tracking trajectory has failed to follow the track point, so the gesture tracking effect based on the Cam-shift algorithm cannot be well obtained.

The hands gesture track trajectory diagrams are shown in the following. Figure 9 is the actual effect diagram of

two-hand gesture tracking based on the switching federated filter algorithm, and Figure 10 is the computer screenshot of hands gesture tracking in Figure 9. From Figure 9, the gesture tracking effect is better when both hands are moved in front of the screen. The 3D interactive software can accurately track the gesture track and display the gesture track. From Figure 10, there is no tracking failure point on the tracking trajectory, so the gesture tracking based on the switching

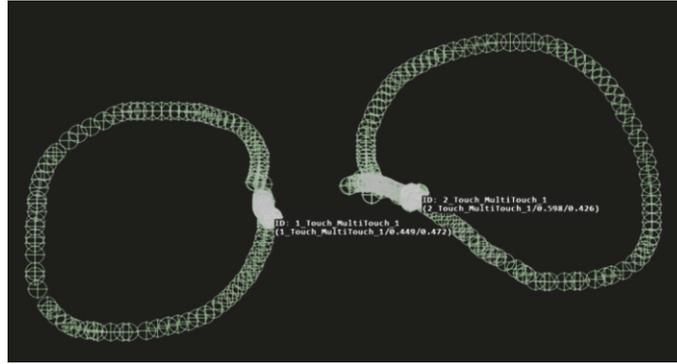


FIGURE 10: The computer screenshot of hands gesture tracking based on switching federated filter algorithm.

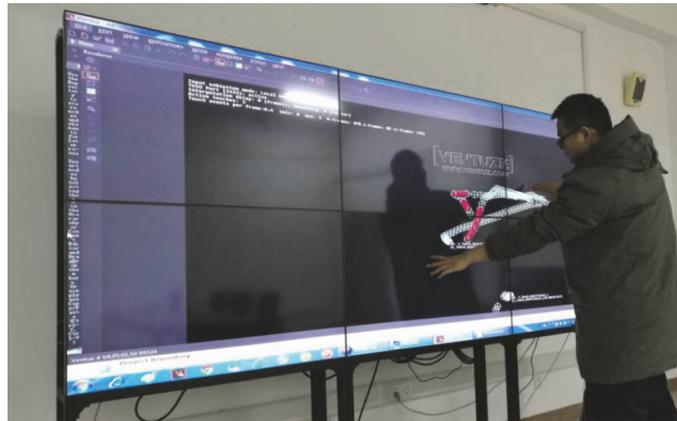


FIGURE 11: The hands gesture tracking effect based on Cam-shift algorithm.

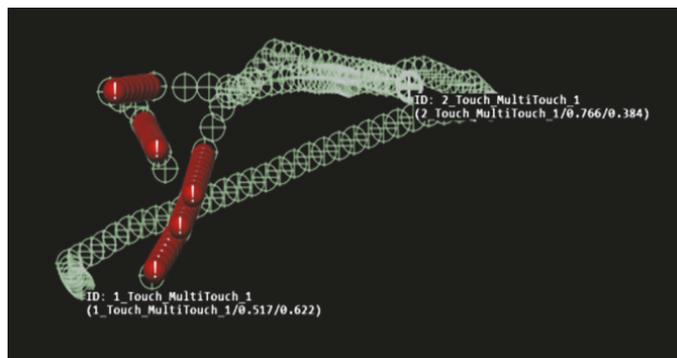


FIGURE 12: The computer screenshot of hands gesture tracking based on Cam-shift algorithm.

federated filter algorithm works well. Figure 11 is a schematic diagram of the two-hand operation based on the Cam-shift algorithm. When both hands are moved in front of the screen, the tracking effect of the gesture tracking fails. The computer screenshot of the hand gesture trajectory is shown in Figure 12, and the track trajectory fails to follow the trajectory point (the tracking failed trajectory point is displayed as a red exclamation point), so the effect of gesture tracking based on the Cam-shift algorithm cannot be well obtained.

To verify the accuracy and tracking time at different sampling points based on the federated tracking algorithm and several commonly used algorithms [41], this paper traces the image sequence of 200 different sampling points and calculates the tracking accuracy and tracking time of the algorithm, respectively. Particle filter (PF) is a Monte Carlo approximation to the optimal Bayesian filter and provides robust tracking of moving objects in a cluttered environment, especially in the case of nonlinear and non-Gaussian problems where the interest lies in the detection and tracking of

TABLE 1: Relationship between target tracking accuracy (%) and algorithms.

MP	Cam-shift	Particle Filter	Velocity-Adaptive Particle Filter	Federated Filter Tracking
20	4.8	5.2	5.8	7.1
60	52.5	55.3	58.6	63.1
100	82.5	84.7	89.1	91.7
200	90.1	91.2	92.7	94.9

TABLE 2: Relationship between tracking time (ms) and algorithms.

MP	Cam-shift	Particle Filter	Velocity-Adaptive Particle Filter	Federated filter tracking
20	18.3	16.5	15.7	14.7
60	30.2	27.4	23.2	16.9
100	48.5	39.2	31.5	21.8
200	82.8	64.3	52.1	26.2

moving objects. The velocity adaptive model (VAPF) updates the propagation distance according to the temporal difference of previous frames by calculating the average velocity.

Table 1 shows the tracking accuracy of different algorithms at different sampling points. The correct tracking rate (%) are below the various algorithm names. TA indicates the tracking algorithm. MP indicates the number of sampling points.

Table 2 shows the tracking time of the several algorithms at different sampling points. Below the various algorithm names are tracking times (ms). TA indicates the tracking algorithm. MP indicates the number of sampling points.

From Tables 1 and 2, it can be found that the switching federated tracking algorithm proposed in this paper improves the tracking accuracy and tracking time of the algorithm compared to the several filter algorithms. Therefore, the proposed switching federated filter algorithm has greatly improved the real-time performance and accuracy of tracking gestures and can be well applied in large-screen interactive imaging systems.

4. Conclusions

Recent gesture tracking algorithms of large-screen interactive imaging systems are faced with inaccuracies and low real-time performance. This paper proposed a switching federated gesture tracking algorithm combining particle filtering and Mean Shift algorithm. The federated filter algorithm first averages the particles to make most of the particles drift into the scope of the gesture region. The subsequent step of particle prediction can be omitted, thus saving the running time of the algorithm. In the aspect of experimental simulation, this paper first compares the effect graph before and after the filtering with the self-designed software platform and shows that the switching federated filter has good effect in removing the positioning interference. Then, the trajectory display function is invoked in the 3D interactive software. Compared with the Cam-shift algorithm for the gesture tracking image in actual large screen, whether it is tracking one-hand gesture or two-hand gesture, none of the federated filter algorithms will fail to track or lose the target. Therefore, the switching federated filter algorithm can solve

the problem of low accuracy and real-time performance in dynamic gesture tracking. This algorithm effectively reduces the impact of complex environments on tracking effects, which can be applied to interactive imaging systems such as large screens.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work is supported by the National Natural Science Foundation of China (61403268, 61873176); Natural Science Foundation of Jiangsu Province, China (BK20181433); Fundamental Research Funds for the Central Universities (30918014108); Natural Science Fund for Colleges and Universities in Jiangsu Province (16KJB120005); Open Fund for Collaborative Innovation Center of Industrial Energy-Saving and Power Quality Control, Anhui University (KFKT201806).

References

- [1] T. E. Hutchinson, K. P. White, W. N. Martin, K. C. Reichert, and L. A. Frey, "Human-computer interaction using eye-gaze input," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1527–1534, 1989.
- [2] Yang Liu, Zhaoyang Lu, Jing Li, Chao Yao, and Yanzheng Deng, "Transferable feature representation for visible-to-infrared cross-dataset human action recognition," *Complexity*, vol. 2018, Article ID 5345241, 20 pages, 2018.
- [3] L. A. Alexandre, "Gender recognition: A multiscale decision fusion approach," *Pattern Recognition Letters*, vol. 31, no. 11, pp. 1422–1427, 2010.
- [4] L. Yu, C. Li, and S. Fei, "Any-wall touch control system with switching filter based on 3-D sensor," *IEEE Sensors Journal*, vol. 18, no. 11, pp. 4697–4703, 2018.

- [5] H. Hasan and S. Abdul-Kareem, "Human-computer interaction using vision-based hand gesture recognition systems: A survey," *Neural Computing and Applications*, vol. 25, no. 2, pp. 251–261, 2014.
- [6] S.-C. Hsu, J.-Y. Huang, W.-C. Kao, and C.-L. Huang, "Human body motion parameters capturing using kinect," *Machine Vision and Applications*, vol. 26, no. 7-8, pp. 919–932, 2015.
- [7] H. Xu, L. Yu, and S. Fei, "Hand-held 3-D reconstruction of large-scale scene with kinect sensors based on surfel and video sequences," *IEEE Geoscience and Remote Sensing Letters*, pp. 1–5.
- [8] G. Dai, L. Yu, H. Xu, Z. W. Hou, and S. M. Fei, "Federated filter algorithm with positioning technique based on 3D sensor," *Circuits Systems Signal Processing*, vol. 37, no. 6, pp. 2613–2628, 2018.
- [9] V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677–695, 1997.
- [10] S. Wu, W. Guo, S. Xu, Y. Huang, L. Wang, and T. Tan, "Coupled topic model for collaborative filtering with user-generated content," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 6, pp. 908–920, 2016.
- [11] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang, "Semisupervised learning of classifiers: Theory, algorithms, and their application to human-computer interaction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 12, pp. 1553–1567, 2004.
- [12] W. Feng, Y. Yang, L. Wan, and C. Yu, "Tone-mapped mean-shift based environment map sampling," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 9, pp. 2187–2199, 2016.
- [13] H. Cho, S.-J. Kang, and Y. H. Kim, "Image segmentation using linked mean-shift vectors and global/local attributes," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 10, pp. 2132–2140, 2017.
- [14] Y. A. Ghassabeh, T. Linder, and G. Takahara, "On some convergence properties of the subspace constrained mean shift," *Pattern Recognition*, vol. 46, no. 11, pp. 3140–3147, 2013.
- [15] J.-H. Lee, H.-J. Jung, and J. Yoo, "A real-time face tracking algorithm using improved camshift with depth information," *Journal of Electrical Engineering & Technology*, vol. 12, no. 5, pp. 2067–2078, 2017.
- [16] S.-H. Yen, C.-H. Wang, and J.-C. Chien, "Accurate and robust ROI localization in a camshift tracking application," *Multimedia Tools and Applications*, vol. 74, no. 23, pp. 10291–10312, 2015.
- [17] Z. W. Wang, X. K. Yang, Y. Xu, and S. Y. Yu, "CamShift guided particle filter for visual tracking," *Pattern Recognition Letters*, vol. 30, no. 4, pp. 407–413, 2009.
- [18] Z. Sun and S. S. Ge, "Analysis and synthesis of switched linear control systems," *Automatica*, vol. 41, no. 2, pp. 181–195, 2005.
- [19] M. S. Branicky, "Multiple Lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.
- [20] Y. Sun, Z. Wu, and F. Meng, "Common weak linear copositive lyapunov functions for positive switched linear systems," *Complexity*, vol. 2018, Article ID 1365960, 7 pages, 2018.
- [21] H. Sun, Y. Li, G. Zong, and L. Hou, "Disturbance attenuation and rejection for stochastic Markovian jump system with partially known transition probabilities," *Automatica*, vol. 89, pp. 349–357, 2018.
- [22] J. Yu, P. Shi, W. Dong, and C. Lin, "Command filtering-based fuzzy control for nonlinear systems with saturation input," *IEEE Transactions on Cybernetics*, vol. 47, no. 9, pp. 2472–2479, 2017.
- [23] G. Zong, H. Ren, and L. Hou, "Finite-time stability of interconnected impulsive switched systems," *IET Control Theory & Applications*, vol. 10, no. 6, pp. 648–654, 2016.
- [24] J. Yu, H. Yu, C. Lin, and W. Dong, "Fuzzy finite-time command filtered control of nonlinear systems with input saturation," *IEEE Transactions on Cybernetics*, vol. 48, no. 8, pp. 2378–2387, 2018.
- [25] H. Xu, L. Yu, J. Hou, and S. Fei, "Automatic reconstruction method for large scene based on multi-site point cloud stitching," *Measurement*, vol. 131, pp. 590–596, 2019.
- [26] H. Ren, G. Zong, and T. Li, "Event-triggered finite-time control for networked switched linear systems with asynchronous switching," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 11, pp. 1874–1884, 2018.
- [27] L. Yu, S. Fei, L. Sun, J. Huang, and G. Yang, "Design of robust adaptive neural switching controller for robotic manipulators with uncertainty and disturbances," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 3-4, pp. 571–581, 2014.
- [28] L. Yu, S. Fei, L. Sun, J. Huang, and Y. Zhu, "Fuzzy approximation of a novel nonlinear adaptive switching controller design," *Circuits, Systems and Signal Processing*, vol. 34, no. 2, pp. 377–391, 2015.
- [29] G. Zong, R. Wang, W. X. Zheng, and L. Hou, "Finite-time stabilization for a class of switched time-delay systems under asynchronous switching," *Applied Mathematics and Computation*, vol. 219, no. 11, pp. 5757–5771, 2013.
- [30] J. Yu, P. Shi, and L. Zhao, "Finite-time command filtered backstepping control for a class of nonlinear systems," *Automatica*, vol. 92, pp. 173–180, 2018.
- [31] J. Hou, L. Yu, and S. Fei, "Four-point trapezoidal calibration algorithm for human-computer interaction system based on 3D sensor," *Measurement*, vol. 134, pp. 730–738, 2019.
- [32] L. Yu, J. Huang, and S. Fei, "Robust switching control of the direct-drive servo control systems based on disturbance observer for switching gain reduction," *IEEE Transactions on Circuits and Systems II: Express Briefs*, p. 1, 2018.
- [33] G. Zong, R. Wang, W. Zheng, and L. Hou, "Finite-time H_∞ control for discrete-time switched nonlinear systems with time delay," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 6, pp. 914–936, 2015.
- [34] L. Yu, "Image noise preprocessing of interactive projection system based on switching filtering scheme," *Complexity*, vol. 2018, Article ID 1258306, 10 pages, 2018.
- [35] R. Xiong, Y. Zhang, H. He, X. Zhou, and M. G. Pecht, "A double-scale, particle-filtering, energy state prediction algorithm for lithium-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1526–1538, 2018.
- [36] L. Martino, J. Read, V. Elvira, and F. Louzada, "Cooperative parallel particle filters for online model selection and applications to urban mobility," *Digital Signal Processing*, vol. 60, pp. 172–185, 2017.
- [37] Z. Zhao, P. Feng, J. Guo et al., "A hybrid tracking framework based on kernel correlation filtering and particle filtering," *Neurocomputing*, vol. 297, pp. 40–49, 2018.
- [38] L. Rong, X. Peng, L. Liu, and B. Zhang, "Improved reduced-order fault detection filter design for polytopic uncertain discrete-time markovian jump systems with time-varying delays," *Complexity*, vol. 2018, Article ID 9489620, 15 pages, 2018.

- [39] L. Yu, G. Dai, and S. Fei, "Contactless multi-touch interactive technology based on adaptive filtering algorithm," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 231, no. 9, pp. 685–692, 2017.
- [40] L. Song, Z. Duan, B. He, and Z. Li, "Application of federal kalman filter with neural networks in the velocity and attitude matching of transfer alignment," *Complexity*, vol. 2018, Article ID 3039061, 7 pages, 2018.
- [41] M. Rollason and D. Salmond, "Particle filter for track-before-detect of a target with unknown amplitude viewed against a structured scene," *IET Radar Sonar Navigation*, vol. 12, no. 6, pp. 603–609, 2018.

