Fingertip Tracking Based on Edge Feature and Pixel Ratio in Gesture

Bo-tao ZENG\textsuperscript{1,2}, Zhi-quan FENG\textsuperscript{1,2,*}, Tao XU\textsuperscript{1,2}, Xiao-hui YANG\textsuperscript{1,2}, Chang-sheng AI\textsuperscript{3} and Wei XIE\textsuperscript{4}

\textsuperscript{1}School of information Science and Engineering, University of Jinan, 250022 Jinan, China
\textsuperscript{2}Shandong Provincial Key Laboratory of Network Based Intelligent Computing, 250022 Jinan, China
\textsuperscript{3}School of Mechanical Engineering, University of Jinan, 250022 Jinan, China
\textsuperscript{4}School of Information and Electrical Engineering, Harbin Institute of Technology at Weihai, 264209 Weihai, China

\textsuperscript{*}Corresponding author

Keywords: Edge feature, Particle filtering, Kinect, Fingertip tracking, Pixel ratio.

Abstract. Fingertip recognition and tracking is a key problem in gesture recognition. The current fingertip locating and tracking method is complicated, or needs to be labelled artificially. In this paper, a particle filtering method based on edge feature and pixel ratio is proposed to track the target finger in a complex background. We set the region of each particle as a fixed value, and calculate the edge orientation histograms and the proportion of the body pixels of each particle area. The similarity degree of edge features is measured by the Bhattacharyya distance, and a new similarity measure is defined to measure the similarity degree of pixel ratio. These two feature similarities will be linearly combined to track the target model. And then we calculate the farthest point from the center of the contour in the predicted model and update the target model again. The results show that the method can track the target fingertip accurately and effectively in real time, under the condition of interference.

Introduction

In the rapid development of the Internet era, artificial intelligence has come to the stage of the time, as an important part of artificial intelligence, human-computer interaction has also been used more and more widely, in which gestures are the main form of human-computer interaction. At present, the recognition and interaction of gestures have also received great attention, and many researchers have devoted themselves to interaction based gestures. But compared with the diversity and polysemy of gestures, the pointing gesture of a single finger is more intuitive and simple. Even the finger tracking is used on mobile phones to achieve real-time human-computer interaction\cite{1} or used in real-time music playing \cite{2}. It is also an ideal human-computer interaction way to gain a person's region of interest by pointing gesture. In order to achieve real-time interaction, how to effectively, quickly and accurately get the fingertip position is the key problem \cite{3}. The common ways to get fingertips at present are: wearing data glove, or wearing some mark of calibrated color on fingertips. These methods belong to non-natural interaction. The method of extracting hand area based on skin-color detection is easy to be disturbed by skin-like color \cite{4}. Analysis of image depth information obtained by external devices such as Kinect, and using K-curvature algorithm to obtain fingertip position, the process is relatively complex \cite{5}.

In this paper, we will integrate the two characteristics of the background edge orientation and the percentage of the finger pixels in the region of interest, and get the joint feature probability by linear combination, and track it with particle filter algorithm. Kinect has the function of identifying the human skeleton, Kinect has the function of identifying the human skeleton, and this article will simply use the Kinect's human skeleton recognition function to divide the human body and the background environment. After the morphological operation, we will get more efficient and non-interfering finger pixel ratio information. To achieve more accurate tracking, the farthest point of
the edge contour center is calculated for each prediction result, and the fingertip’s position will be update.

**Feature Extraction**

The appropriateness of the feature selection has a significant impact on the quality of the tracking effect [6]. Color features are invariable to the translation, rotation, and scale changes of the target, and are often used as the first choice for image processing, but it is easily affected by light and color changes. For a moving target in a complex background, it often loses the target or causes serious deviation. Edge features are less dependent on illumination and color changes, but are more sensitive to the deformation of the target. The article [7] implements the tracking of the grayscale moving target with a combined edge feature. At present, the combination of different features and the integration of the advantages of each feature to achieve the target tracking is a common method [8]. But for small target tracking [9], what features should be used is still a matter to be considered. In this paper, the edge features of the target area and the body pixel ratio in the fixed area will be used as new features. From the depth of information obtained by Kinect, we can get a binary image containing only the human pixel information. Compared with the commonly used skin-color detection, the problem of obtaining the hand region is difficult to segment. After the morphological operation, the depth map can effectively segment the background and accurately represent the percentage of human pixels in a certain area. The two features of the body pixel ratio and the edge information of the target area will be combined to be used to track the target in real time.

**Feature Description**

In this paper, we set the region of each particle as a fixed value and extracted the two features from two template with different sizes [10]. And the two types of templates are centered on fingertips. We extract the edge orientation features from the template of 80 * 90, and the pixel ratio feature from template of 40 * 40.

![Figure 1. Grey-scale image and binary image.](image)

**Figure 1.** Two features are extracted from different sizes of templates, respectively. Edge orientation histogram extracted from the template of 80*90 (a), and the pixel ratio feature extracted from the template of 40*40 (b).

**Fingertip Relocation**

Observing the contour of the binary image of the target model, and the position of fingertip can be defined as the farthest point from the contour center.

![Figure 2. Contour image.](image)

**Figure 2.** The contour image (b) of the finger binary image (a).

The set of the contour points of the finger can be expressed as:

\[ X_c = \frac{\sum_{i=1}^{n} x_i}{n} \]  

(1)
The coordinates of the fingertip point are defined as \((x_p, y_p)\).

\[ p = \arg \max_{i=1}^{n} \left( (x_i - X_c)^2 + (y_i - Y_c)^2 \right) \]  

**Particle Filter Tracking Algorithm**

**Target Model Description**

The tracked target is defined as the target model, and the target of the current frame is candidate target, and the target model is described with the edge feature and the pixel ratio feature respectively.

Set \(q_u\) represents the probability density of each level of the target model’s edge orientation feature, \(n_1\) represents the quantization progression of edge direction value, and the edge orientation characteristic of target model can be described as:

\[ q_u = C \sum_{i=1}^{n} k(||x_i^e||) \delta(b_{EOH}(x_i)-u) \]  

\[ \hat{q}_{EOH} = \{ q_u \}_{u=1...n_1}, \text{and} \sum_{u=1}^{n_1} \hat{q}_u = 1 \]  

Where \(k(\cdot)\) is the section function of the Epanechnikow kernel, \(\delta(\cdot)\) is the Dirac function. And \(x_i^e\) represents the gradient amplitude of the edge point of the target model. \(N\) represent the number of edges of the edge pixels in the target model. \(b_{EOH}(\cdot)\) is used to judge the value of the direction of the edge point whether equal to the eigenvalue. \(C\) is the normalization coefficient.

Set \(q_v\) represent the probability density of human body pixels ratio of target model. \(T_i\) express the number of body pixels of the target model, and \(C_i\) represents the total number of pixels of the target model. The experiment shows that the size of the selected region of interest is set to \(40 \times 40\) will has good tracking performance.

\[ \hat{q}_v = \frac{T_i}{C_i} \]  

**Similarity Measure**

In order to measure the similarity between the target model and the candidate models, Bhattacharyya distance is used to represent the similarity of edge orientation characteristics between the target model and candidate models.

For pixel ratio features, a new representation of similarity is set as:

\[ \rho(y) = 1 - \frac{||\hat{p}_y(y) - \hat{q}_l||}{\hat{q}_l} \]  

The bigger the \(\rho\) is, the better the similarity of the pixel ratio feature is. In the changing scene, the performance of the two features is different. In order to integrate the advantages of the two, a linear weighting method is used to combine the weights of the two features.

\[ \alpha_1 = \frac{1 - \rho_1}{(1 - \rho_1) + \rho_2} \]  

\[ \alpha_2 = \frac{\rho_2}{(1 - \rho_1) + \rho_2} \]  

The better the performance of a feature is, the more similar it is to the target model, and it get the greater weight. The joint characteristic probability density can be expressed as:

\[ \rho_{sum}(y) = \alpha_1 (1 - \rho_1) + \alpha_2 \rho_2 \]
Fingertip Tracking Algorithm

The program is written in Visual studio 2013 and read into the video through the Kinect device.

Input Video → Observation Model → Feature Calculation → Particle Filter

Update Prediction Model ← Corner Detection ← Prediction Model

Figure 3. Algorithm flow chart.

Initialize the target area.

Initialize the region of interest when the finger into the calibration area, and determine the coordinates of the center point, half width and half height of the particle. The weight of each particle are initially set to 1/N, make sure each particle have an equal opportunity.

Tracking Procedure.

Step 1: According to the state equation of the system, the state prediction value is obtained, and update each state to get the state prediction parameter.

Step 2: For each state prediction parameter, calculate the EOH and pixel ratio of the candidate region in which it is located. Updating the weights and calculating the probability density of the EOH $\rho_1$ and pixel ratio $\rho_2$ of each particle area, the weight ratio $\alpha_1$ and $\alpha_2$.

Step 3: According to the updated weight, a state parameter is estimated as the tracking output.

Step 4: When the similarity is greater than the threshold, the corner position of the region is calculated and the position of the corner is re updated as the prediction position. And the similarity between the predicted state and target model will be calculated again. When the joint feature similarity is greater than the threshold (initial setting is 0.86), update the edge histogram of the target model by the forgetting factor and recalculate the pixel ratio. The value range of forgetting factor is 0.1-0.3.

Step 5: Calculating the maximum weight value and judging whether the tracking is normal with the predictive position information. If normal, then output the target position and turn to Step 1. Otherwise, the tracking is lost, and program end.

Experimental Results and Analysis

For the different situation like the movement of the finger, the change of the background, and the interference of the finger of the non-target human, the subtest is carried out by this algorithm.

Figure 4. Experimental result.

Figure 4. In the first period of the initial run of the program (a), the tracking effect was good (b). A slight offset occurred at the 132th frame (c), but it soon recovered in the 137th frame (d). There is an
interference with a non-target person in (e), and background change in (f). There are good tracking performance in all two situations.

Experimental results show that in the process of finger movement, this algorithm can accurately track fingertip position. When slight error occurs, it can also find the precise fingertip position in a very short time. For the change of the background, when the non-target person or non-target finger appears in the picture, the algorithm can still track the fingertips correctly, and there will be no false tracking.

Conclusions

A particle filter tracking algorithm based on edge direction information and pixel duty ratio is proposed in this paper. The method of tracking the target with a single color and edge features does not apply to a small target such as the fingertip. In this paper, considering the fingertip is small target, we proposed the concept of pixel ratio which the effectiveness of target tracking is measured by the degree of change in the ratio of target pixels in the fixed size area. Combined with the edge direction information of the image, the tracking of the target is realized. When the tracking position is trusted, the fingertip position in the prediction area are recalculated, and update the target model. The experiment shows that the algorithm can effectively track the fingertips in the motion, and can track the target fingertip well when the background is disturbed.

Acknowledgement

This paper was supported by the National Key R&D Program of China (No. 2016YFB1001403) and the National Natural Science Foundation of China (No. 61472163).

References


