

Feature-point Tracking Evaluation Method

Xiang RUAN^{*}, Haihong ZHANG^{**} and Hiromitsu HAMA^{***}

(Received September 30,2000)

Synopsis

Feature point tracking is a very important problem in 3D reconstruction research area. We propose here a new method that expand the algorithm adopted by Carlo Tomasi and Takeo Kanade. In this paper, we first introduce the tracking method adopted by Tomasi and Kanade. The method still has something that should be improved. The main problem is real-time-checking wrong tracking. Our method can expand their method, and can increase the precision of their method. It is based on histogram theory and image gradient theory. We also show some simulation result in this paper to prove the validity of our method.

Keywords: Feature points, Tracking, 3D reconstruction

1. Introduction

Feature point tracking is a very important problem in 3D reconstruction research area, since the efficiency will do great influence on the reconstruction. So much effort has been adopted to resolve point-tracking problem. As mentioned by Tomasi and Kanade^[1], feature-tracking problem should be divided into two basic problems. That is, 'how to select the features and how to track them from frame to frame. Actually, most researches mainly concentrate on the latter one^{[2][3][4]}.

Tomasi and Kanade^[5] tried to resolve two problems synchronously. Firstly, they do the work of feature selection basing on some conditional criteria. Then the latter problem was resolved by minimize the sum of squared intensity differences between a past and a current window. They approximate the current window by a translation of the old one, and then write a linear 2×2 system whose unknown is the displacement vector between the two windows. But the problem is, when we do a wrong tracking in frame I, the error will continue to the frames next to I.

In this paper, we develop a new method to extent Tomasi and Kanade's research. We try to find good criteria to check the correctness of the tracking process. We check the validity of tracking result step by step, which can increase the precision, and reduce the infection of noise during the experiment.

First, we review the tracking method adopted by Tomasi and Kanade in Section 2. After analyzing the method, a new algorithm is proposed to check the validity of tracking in run time. At last, we show some experiment data, and discuss them.

2. Feature Tracking

The patterns of image intensities change in a complex way when the camera moves. We always use three variables x, y and t to describe an image sequence $I(x, y, t)$. Here, x and y are the space variables, and t is the time variable. In fact, images taken at near time instants are usually strongly related to each other, because they refer to the same scene taken from only slightly different viewpoints. We express this relationship as follows:

$$I(x, y, t + \tau) = I(x - \xi, y - \eta, t). \quad (1)$$

That means a later image taken at time $t + \tau$ can be obtained by moving every point in the current image, which is taken at time t , by a small amount. So, we define this amount as $\vec{d} = (\xi, \eta)$, named *displacement* of the point at $\vec{x} = (x, y)$ between time t and $t + \tau$. It is obviously that the main problem of track is finding displacement \vec{d} . However, it is quite difficult to track one pixel feature point since the lack of relative information, so instead of one pixel, we use a window of pixels.

Here we meet two problems: first, how to know that we are following the same window if its contents change over time, and second, how to combine the different velocities to give the one resulting vector if we measure displacement of the window. In fact, the algorithm, will be adopted in this paper, is a solvent of the first problem. And Tomasi and Kanade solve the latter problem with an algorithm introduced in the following.

In principle, rather than describing window changes as simple translations, Tomasi and Kanade model the changes as a more complex transformation, such as an affine map. In this way, different velocities can be associated to different points of the window. Here, we define $J(x) = I(x, y, t + \tau)$ $I(\bar{x} - \bar{d}) = I(x - \xi, y - \eta, t)$, and τ is so small here, then we get local image model:

$$J(\bar{x}) = I(\bar{x} - \bar{d}) + n(\bar{x}), \quad (2)$$

where n is noise. The displacement vector \bar{d} is then chosen so as to minimize the residue error defined by the following double integral over the given window ω

$$\varepsilon = \int_{\omega} [I(\bar{x} - \bar{d}) - J(\bar{x})]^2 w dx, \quad (3)$$

where w is a weight function that emphasizes the central area of the window, and we always set it to 1 for simplicity.

2.1 Solving Displacement Vector

Because the displacement vector \bar{d} is quite small, we can approximate the intensity function by its Taylor series and truncate it to linear term as follows:

$$I(\bar{x} - \bar{d}) = I(\bar{x}) - \bar{g} * \bar{d}$$

Rewrite Eq. (3) by using the equation above, then we obtain

$$\begin{aligned} \varepsilon &= \int_{\omega} [I(\bar{x}) - \bar{g} * \bar{d} - J(\bar{x})]^2 w d\bar{x} \\ &= \int_{\omega} (h - \bar{g} * \bar{d})^2 w d\bar{x}, \end{aligned} \quad (4)$$

where $h = I(\bar{x}) - J(\bar{x})$. Differentiate Eq. (4) with respect to \bar{d} and set the result to zero, then

$$\int_{\omega} (h - \bar{g} * \bar{d}) \bar{g} w dA = 0.$$

Because $(\bar{g} * \bar{d}) \bar{g} = (\bar{g} \bar{g}^T) \bar{d}$ and \bar{d} is nearly constant within ω , we get

$$\left(\int_{\omega} \bar{g} \bar{g}^T w dA \right) \bar{d} = \int_{\omega} h \bar{g} w dA.$$

Rewrite the equation listed above, we get the basic equation for tracking:

$$G \bar{d} = \bar{e}, \quad (5)$$

where the coefficient matrix is the symmetric 2×2 matrix

$$G = \int_{\omega} \bar{g} \bar{g}^T w dA,$$

and

$$\bar{e} = \int_{\omega} (I - J) \bar{g} w dA.$$

2.2 Defects of the Algorithm

Tomasi and Kanade made some experiments, and got good results. In fact, it is really a valid method to track feature points, but it still has some problems. In Eq.(3), we just set weight function w to 1. Actually, if we can find a valid weight function, the precision will be increased rapidly. This is the first problem of the algorithm left unsolved.

In this paper, we try to resolve the second problem, that is, how to select problem of the algorithm. Since we take photos from various visual angles, a certain feature point is not always visible throughout the image stream. In the other hand, the feature point may be lead to wrong direction during tracking process. It is also obvious that the method introduced above is not sensitive to noise. The problem obtained here is how to ensure that the tracker is always tracking the right feature point throughout the tracking process. We adopt a new method to resolve the problem. In our method, we check the feature point frame by frame. Basing on the proposed evaluating criteria, the method judges that the feature point has been lost, then we abandon the feature point. This ensure us always track the right feature point, and increase the accuracy of the tracking.

3. Feature-point Tracking Evaluation

Our method is based on two facts. First is, background image is much more unitary than object image. Second is, in a certain image stream, visual angels between two contiguous images is very small, thus, will be similar with each other. For a certain feature, there should be some intensity properties of which is fix throughout the whole image stream. We use these properties to check the right features.

As mentioned before, one feature point is difficult to be tracked. We use small windows around the feature points to present each feature points. We regard the properties of the windows as the properties of the feature points.

In the other hand, background of image is always unitary, and the texture of object is abundant. So gradient is another property to identify features. We suppose all pixels of frame A is H_A and all pixels of frame B is H_B . If feature point windows f in frame A has an intensity I_a and in frame B has an intensity I_b (We use the intensity if the window center as the intensity of the window). The amount of pixel at I_a is H_{I_a} and H_{I_b} is the pixel amount at I_b . Then we define the ratio R_a and R_b as:

$$R_a = \frac{\sum_{i=I_a-1}^{I_a+1} H}{H_A}, \quad R_b = \frac{\sum_{i=I_b-1}^{I_b+1} H}{H_B}.$$

In physical meaning, if we use S1 to present the area of whole histogram, and S2 present the area between $I_a - 1$ and $I_a + 1$ in the histogram, then R_a is the ratio of S1 and S2 (Fig. 1).

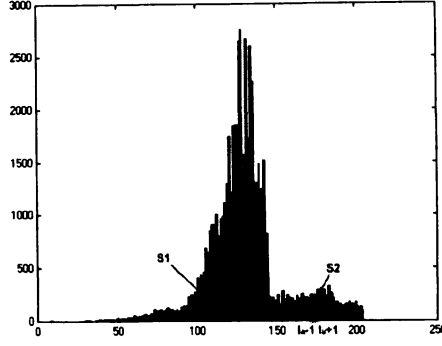


Fig.1 Physical meaning of R_a and R_b .

If the range of intensity of the image is 255, then we define ε as follow:

$$\varepsilon = \sqrt{\frac{\sum_{j=1}^{255} (H_{ia} - H_{ib})^2}{254}}.$$

On the other hand, we block the image with 3×3 windows to present every point. We define g_j as the average gradient of all 9 points in block windows j . If we are tracking feature window i in frame a , then define:

$$G_{ia} = \sqrt{\frac{\sum_{j=1}^n (g_i - g_j)^2}{n-1}},$$

here we suppose images are blocked into n blocks. Certainly, in next frame b :

$$G_{ib} = \sqrt{\frac{\sum_{j=1}^n (g_i - g_j)^2}{n-1}}.$$

It is obvious that if combining with Tomasi and kanade' method with the evaluation criteria we proposed, a robust tracking result can be obtained. When tracking feature window i from frame a to frame b , first, we calculate displacement \vec{d} from frame a , get the position of feature window i in the frame b . Then we use formula in the follow to judge the rightness of the tracking:

$$\text{If } (|R_a - R_b| < \varepsilon) \text{ and } (|G_{ia} - G_{ib}| < \lambda) \Rightarrow \text{Good Tracking}$$

$$\text{Otherwise} \Rightarrow \text{Bad Tracking}$$

If the tracking is good, we continue to next step, if not, we try again to find a good result or judge that the feature window lost in the image stream. And we set a threshold $\lambda = 0.5$.

4. Simulations

In this section, we show some simulation results. We use an image stream with 30 frames, and try to track 50 features from frame 1 to frame 30(Fig. 2).

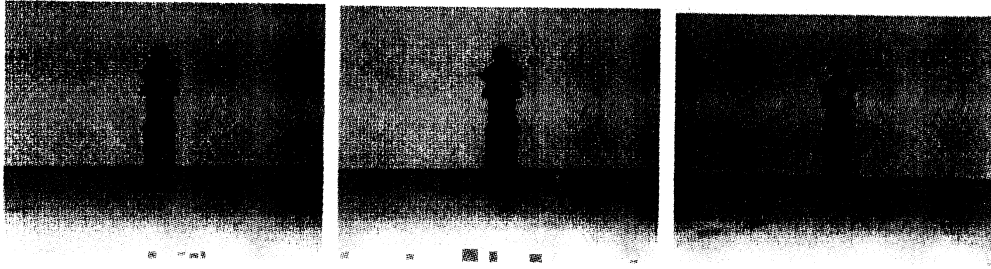


Fig.2 Frame 1, Frame 20 and Frame 30.

We first use tracking method without our new check method to track the features. We got 5 errors tracking at the last frame. And then, we use tracking method with our proposed check method to track again, got only one error tracking at the last frame.



Fig.3 Tracking result without using check method.

In Fig.3, we show the tracking result of tracking without our new method. Feature 20 and 35 are miss tracked in the frame 30, because feature 20 is not visible in the frame 30, and feature 35 changed so much is the image stream. But when we tracked with our new check method, feature 35 is right tracked, and we delete feature 20 in the frame 30, because it is not visible throughout the image stream (Fig. 4). From the simulation, we can see that precision is increased 8%. (we got 10% error in former simulation and 2% error in the latter simulation)

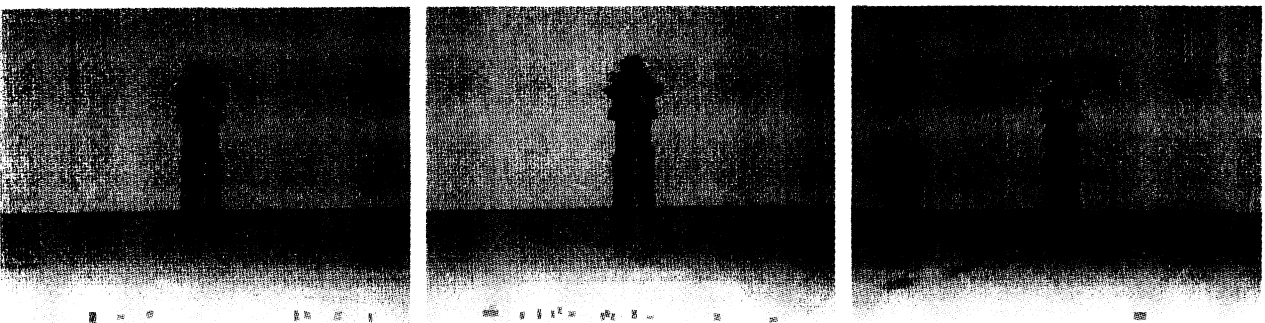


Fig.4 Tracking result with using check method.

5. Conclusions

In this paper, we adopt a new method to expand Tomasi and Kanade' track algorithm. How to choose good threshold (ϵ, λ) is the crux of resolving the point-tracking problem. If background is not unitary and object has no abundant texture, then result in not satisfied. This will be improved in the further works.

References

- [1] Carlo Tomasi, Takeo Kanade, "Shape and Motion from Image Streams: a Factorization Method—Part 3 Detection and Tracking of Point Features", CMU-CS-91-132, April 1991,

http://www.ri.cmu.edu/people/person_136_pubs.html.

[2] Jianbo Shi, Carlo Tomasi, "Good Features to Track", IEEE Conference on Computer Vision and Pattern Recognition ,(CVPR94) Seattle, June 1994,

http://www.ri.cmu.edu/people/person_136_pubs.html.

[3] D.J Connor and J.O.Limb, "Properties of frame difference signals generated by moving images", IEEE Trans., COM-22 (10), pp.1564-1575, 1974.

[4] C.Cafforio and F.Rocca, "Methods for measuring small displacements in television images", IEEE Trans., IT-22, pp.573-579, 1976.

[5] T.W.Ryan, R.T.Gray, and B.R.Hunt, "Prediction of correlation errors in stereo-pair images", Optical Engineering, 19(3), pp.312-322,1980.