Proposal

Topic : Background Subtraction in Freely Moving Camera

Air Force Island

Guanfang Dong, Yingnan Ma

1.Abstract

Background subtraction is to recognise background from foreground and remove the background from the video. Previous method is finding the coarse foreground area by [1]particle trajectory. The result can be improved by using [2]Otsu's thresholding.

In order to get a more accurate result, we apply some creative methods, which are different from the previous method. After we get course foreground, instead of using Otsu's thresholding, we remove the optical flow by analyzing the texture of the video. After removing the optical flow, we can avoid the negative effect caused by optical flow when recognising background from foreground. Also, we implement DCT in our project in order to cancel the noise on each frame of the video. With the combination of above methods, the accuracy of background subtraction result can be improved dramatically.

2.Introduction

In real world, the background subtraction technology is used in many areas. When we try to implement background subtraction from videos, the detection of moving objects is the challenge of the project, no matter for stationary camera or for freely moving camera. For freely moving camera, it is more challenging when we implement the background subtraction. Due to the procedure of camera, the position of moving objects and whole background are changed. In order to avoid recognising moving background as foreground, we need a more accurate algorithm, which can recognise the moving objects from the "moving" background. We cannot simply calculate the background image and use this image to implement background subtraction. Also, since the camera is freely moving, the video recorded by moving camera will make background and foreground move together. In addition, since we live in a three dimensional world, camera will actually project the 3d object into 2d space. This also leads to another problem. The objects may look different due to the camera procedure. Therefore, we cannot just simply create a very big background image and implement background subtraction. Since most videos are recorded by freely moving camera or by monitors, it is essential to implement an algorithm about moving object detection in freely moving camera and extend the algorithm in a creative way.

During the process of research, we find that the paper [2]*Moving Objects Detection with Freely Moving Camera via Background Motion Subtraction* is related to the background subtraction in freely moving camera. This paper suggests that we could start at the point of particle trajectory, which means [2]"the moving path of a particle across a video clip". Based on particle trajectory, we could use probabilistic graphical model to receive pixel wise foreground or background labeling. After that, a coarse foreground may be showed so that we can edit this coarse foreground in detail to get a more accurate foreground.

Previous methods such as Otsu's thresholding may lead to bad performance for changed-velocity moving objects. To improve the performance, we apply some creative methods. We remove the optical flow by analyzing the texture of the video. After removing the optical flow, we can avoid the negative effect caused by optical flow when recognising background from foreground. Also, we implement DCT in our project in order to cancel the noise on each frame of the video. With the combination of above methods, the accuracy of background subtraction result can be improved dramatically.

3.Work Summary

Our work is inspired by the instruction of the paper *Moving Object Detection With Freely Moving Camera via Background Motion Subtraction*. Firstly, a video clip will be imported and we compute particle advection to find particle trajectories. [4]We will build measurable matrixes from velocity vector instead of position vectors.

$$\mathbf{W}_{2F\times P} = [\mathbf{w}_1^T \cdots \mathbf{w}_P^T]^T = \begin{pmatrix} u_{11} & \cdots & u_{1P} \\ v_{11} & \cdots & v_{1P} \\ \vdots & & \vdots \\ u_{F1} & \cdots & u_{FP} \\ v_{F1} & \cdots & v_{FP} \end{pmatrix}.$$

Figure1. Measurable matrixes for particle trajectory

Then, based on the particle trajectories, we could distinguish the moving object from the background by a reasonable thresholding, [1] because the inconsistency of motion in foreground block is higher compared on in background block. This reasonable thresholding can be found by some methods like mean squared error. Then, we could get a coarse foreground.

In order to generate a more precise foreground, we reconstruct background motion. This paper uses geometry-oriented method to inpaint image. After that, we obtain [1]“background motion of particles in CFG” by their publicly available code. Now we import the length of objective motion vector in CFG and find the best thresholding. Paper also uses [3]Otsu’s thresholding method, which follows the patterns of moving object in constant motion. However, as the result of experiment that made by author of paper, Otsu’s thresholding method may generate a wrong value for object moving in different velocity. For example, [3]Otsu’s thresholding will “mis-classfied some football players running at low speeds as background”. To deal with this problem, we analyze the optical flow and the consequence that influenced by optical flow. Based on the analysis of the video textures under the different lighting degrees, we generate a reasonable offset and remove the negative effects caused by optical flow. In addition, we implement DCT method to generate a smoother image for each frame since some videos have hot pixel when these videos are shot in a low brightness condition. Combination of above methods improves accuracy a lot.

$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos\left[\frac{\pi(2x+1)u}{16}\right]\cos\left[\frac{\pi(2y+1)v}{16}\right]$$

Figure2. DCT forward transformation

We also use some normalization methods such as Min-Max, Log and Arctan normalizations. For example, Min-Max will [1] “preserve the relationship among the original data values”. Log and Arctan normalizations will help in scattering the small values and clustering the large values. Finally, some background objects may block the foreground objects.

$$\mathcal{F}_{\mathrm{MM}}(x) = \frac{x - X_{\min}}{X_{\max} - X_{\min}}$$

$$\mathcal{F}_{\mathrm{LOG}}(x) = \frac{\log(x - X_{\min} + 1)}{\log(X_{\max} - X_{\min} + 1)}$$

$$\mathcal{F}_{\mathrm{ATAN}}(x) = \frac{\arctan(x - X_{\min})}{\arctan(X_{\max} - X_{\min})}$$

Figure3. Formula of Min-Max, Log and Arctan Normalizations

We find the color differences between background objects and moving objects. Then, we use mean-shift segmentation method to refine foreground pixels by dividing image into patches. In the end, the background can be subtracted from the video with the guarantee of high accuracy.

4.Creative ideas

- Remove the optical flow by analyzing the texture of the video
- Implement DCT to achieve the noise canceling

Based on the course foreground, we remove the optical flow by analyzing the texture of the video. After removing the optical flow, we can avoid the negative effect caused by optical flow when recognising background from foreground. Also, we implement DCT in our project in order to cancel the noise on each frame of the video. With the combination of above methods, the accuracy of background subtraction result can be improved dramatically.

5.Plan to do

- Import the video clip and compute particle advection to find particle trajectories.
- Distinguish the moving object from the background by a reasonable thresholding, based on the particle trajectories.
- Subtract background to get a coarse foreground, followed by reconstructing background motion to prepare for generating a more precise foreground.
- Ensamble Motion in Coarse foreground, followed by implementing the motion propagation to get background motion in coarse foreground.
- Implement DCT to achieve the noise canceling.
- Remove the optical flow by analyzing the texture of the video.
- Generate an optimized foreground by implementing Mean-shift Segmentation.

6.Timeline & Lab Arrangement

- By 02.24 (Lab 6)
  Import the video clip and compute optical flow and particle advection to find particle trajectories.
- By 03.03 (Lab 6)
  Distinguish the moving object from the background by a reasonable thresholding, based on the particle trajectories.

- By 03.10 (Lab 7)
  Subtract background to get a coarse foreground, followed by reconstructing background motion to prepare for generating a more precise foreground.
- By 03.17 (Lab 8)
  Ensamble motion in coarse foreground. Then, ensamble motion in coarse background, followed by implementing the motion propagation to get background motion in coarse foreground.
- By 03.24 (Lab9)
  Implement DCT to achieve the noise canceling and remove the optical flow by analyzing the texture of the video.
- By 03.31 (Lab10)
  Generate an optimized foreground by implementing Mean-shift Segmentation
- By 04.07
  Final Report

7.References

[1]Sheikh, Y., Javed, O., & Kanade, T. (2009, September). Background subtraction for freely moving cameras. *In Computer Vision, 2009 IEEE 12th International Conference on* (pp. 1219-1225). IEEE.

[2]Wu, Y., He, X., & Nguyen, T. Q. (2017). Moving object detection with a freely moving camera via background motion subtraction. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(2), 236-248.

[3]Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66.

[4]Wu, S., Oreifej, O., & Shah, M. (2011, November). Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *Computer Vision (ICCV), 2011 IEEE International Conference* on (pp. 1419-1426). IEEE.

[5]Sajid, H., & Cheung, S. C. S. (2015, September). Background subtraction for static & moving camera. In *Image Processing (ICIP), 2015 IEEE International Conference* on (pp. 4530-4534). IEEE.

[6]Heikkila, M., & Pietikainen, M. (2006). A texture-based method for modeling the background and detecting moving objects. *IEEE transactions on pattern analysis and machine intelligence*, *28*(4), 657-662.

[7]Mahadevan, V., & Vasconcelos, N. (2008, June). Background subtraction in highly dynamic scenes. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*(pp. 1-6). IEEE.

[8]Ren, Y., Chua, C. S., & Ho, Y. K. (2003). Motion detection with nonstationary background. *Machine Vision and Applications*, *13*(5-6), 332-343.

[9]Ahmed, N., Natarajan, T., & Rao, K. R. (1974). Discrete cosine transform. *IEEE transactions on Computers*, *100*(1), 90-93.

[10]Yalcin, H., Hebert, M., Collins, R., & Black, M. J. (2005, June). A flow-based approach to vehicle detection and background mosaicking in airborne video. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (Vol. 2, pp. 1202-vol). IEEE.