Motion Estimation on Fish-Eye Images

Arief Suryadi Satyawan[1] Junichi Hara[2] Hiroshi Watanabe[1,2]

[1] Graduate School of Fundamental Science and Engineering, Waseda University
[2] Global Information and Telecommunication Institute, Waseda University

1. Introduction

Motion estimation is one of important techniques in the image or video processing used for many applications such as, object tracking, 3D reconstruction, image, and video compression, etc. Many studies related to this technique have been developed by researchers, and most of them tend to reveal motion vectors occurred between successive 2D images corresponding to 3D phenomenal movements captured by conventional cameras. Therefore, variations on block-matching, phase correlation and optical flow algorithm have grown significantly to increase the capability and performance of the above applications. However, it is interesting to be re-observed whether those algorithms could be used well for images taken from recent omni-directional cameras since the images undergo non-linear distortion caused by the characteristic of these cameras [1].

In this paper, study of applying optical flow algorithms for estimating motion from omni-directional images will be described. The algorithms are based on Lucas and Kanade and Horn and Schunck concept, while the images are captured by using a fish-eye camera.

2. Proposed Method

2.1 Optical Flow

Basic idea of the optical flow comes from assumption that the apparent brightness pattern of moving objects remains constant [3]. Hence, the optical flow constrain is stated as I(x,y,t) = I(x+u,y+v,t+1), where u and v is the optical flow vector of a pixel at (x,y) from time t to t+1. The linearized version (first-order Taylor) of this then yields the optical flow constrain become $I_t+I_xu+I_yv =0$, which I_x, I_y and I_t now are partial derivatives of image function and also the inherent dependency on image position. This constrain then solved in different ways by Lucas and Kanade (LK) and Horn and Schunck (HS) [3].

2.2 Experimental Design

In this experiment, images came from a video sequence captured by fish-eye Ricoh Theta camera sized 720x640 pixels including some black area outside the eye circle. By applying LK and HS algorithm written in Matlab [4], the optical flows between two successive images are calculated. As a result, observed motion vector fields from the first image corresponding to the motion of pixels as if these move toward to the second image are obtained. From then on, to be able to evaluate how far the motions flow through correctly, we apply motion-compensated configuration for determining qualitative and quantitative performance [2]. We firstly reconstruct the first image moved by the motion vectors. Since the movements of the pixels are mostly non-integer steps (new pixels are estimated move to area among other pixels), correlation method can be used to generate the new image, and linear method is preferred. Finally, the reconstructed image is compared with the next/second image to calculate error (peak to peak signal to noise ratio/PSNR). The general algorithm developed in this research is provided in Table 1.

3. Results

3.1 Image Source

In this research, there are 126 fish-eye images (one-eye side) obtained from video sequence showing a hand with an eraser moved slowly from left to right, and four of them (image 17th, 18th, 75th and 76th) are shown in Figure 1.

3.2 Lucas and Kanade (LK) Optical Flow

In this experiment, window size (w) does not affect to the result when the object is located close to the border inside the circle area, as shown in Figure 2. In this condition, for every size of window, the reconstructed image is produced very well and the PSNR is high at around 40 dB, as depicted in Figure 4. However, the result change when the object moves in the middle area of circle, as shown in Figure 3. In this condition, the reconstructed image become worst for small size of window, while for the large one, the result is better. Interestingly, the PSNR for each window size remain the same at about 22 dB, as depicted in Figure 4. This result will occur when the image reconstruction process does not perform well due to interpolation scheme used.

3.3 Horn and Schunck (HS) Optical Flow

HS optical flow generally performs well for each number of iteration (ite) according to the reconstructed image given from the Figure 5 and 6, respectively, in spite of the PSNR decrease to around 28 dB for each number of iteration used, as shown in Figure 4.

4. Conclusion

In this project, first study on applying optical flow concept for estimating motion from omni-directional images has been done successfully. Results show that the performance of HS's optical flow is better than LK's optical flow. HS is also greater around 8 dB than LK. Nevertheless, it is important to consider the use of non-linear characteristics of fish-eye image into motion model and appropriate size of window throughout the image for


(a) (b) (c) (d)

Fig 1. Images Source a. Image 17th, b. Image 18th, c. Image 75th, and d. Image 76th.

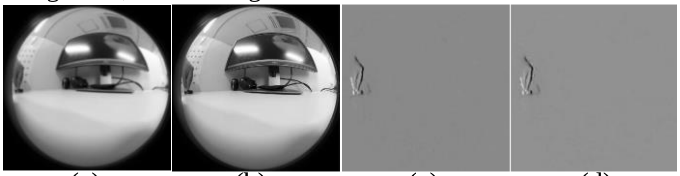
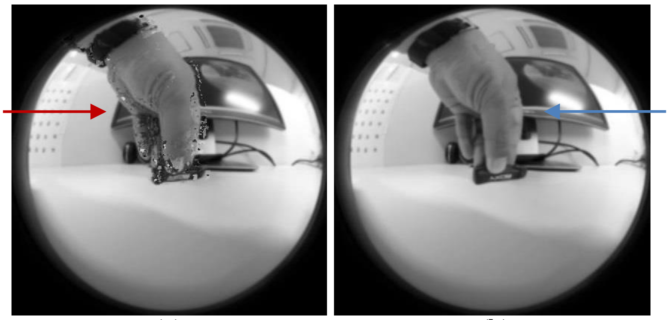

(a) (b) (c) (d)

Fig 2. Reconstructed Image 17th (a. w=2 and b. w=8) and Motion Compensated Image 17th (c. w=2 and b. w=8)


(a) (b)

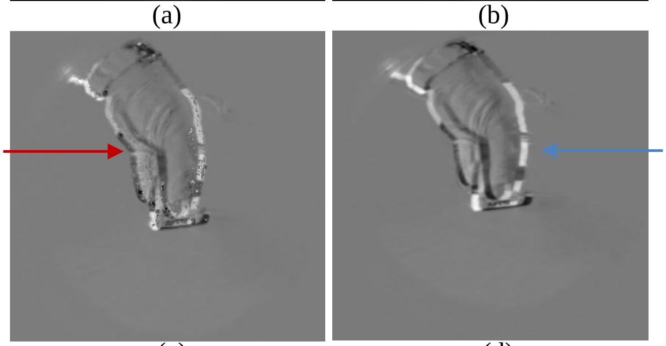

(c) (d)

Fig 3. Reconstructed Image 75th (a. w=2 and b. w=8) and Motion Compensated Image 75th (c. w=2 and b. w=8)

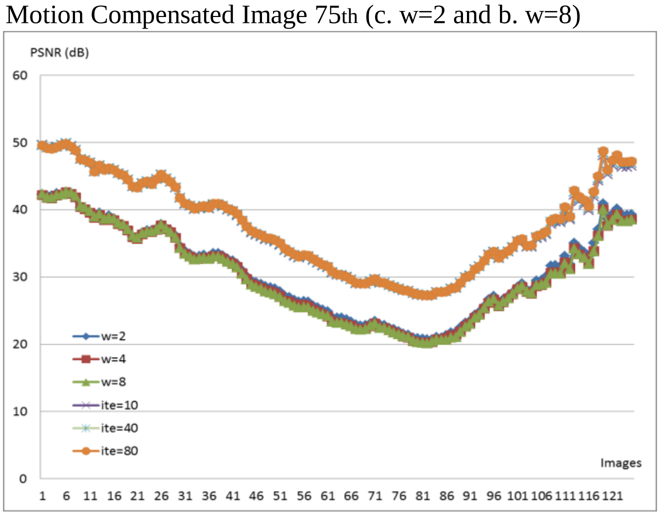


Fig 4. PSNR vs Images for LK (w=2, 4 or 8) and HS (Ite=10, 40 or 80)

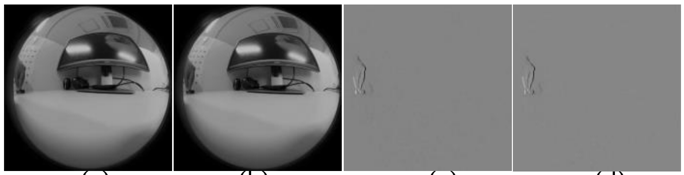

(a) (b) (c) (d)

Fig 5. Reconstructed Image 17th (a. ite=10, b. ite=80) and Motion Compensated Image 17th (c. ite=10, d. ite=80).

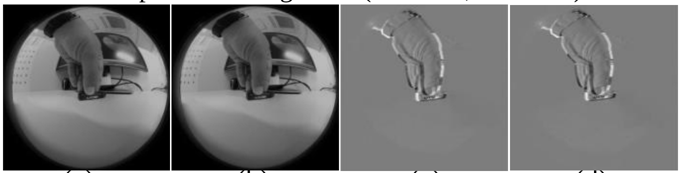

(a) (b) (c) (d)

Fig 6. Reconstructed Image 75th (a. ite=10 and b. ite=80) and Motion Compensated Image 75th (c. ite=10 and d. ite=80).

further experiments. Moreover, according to the image reconstruction used for motion-compensated based evaluation performance, it is important to investigate further the possibility of using other schemes of interpolation, therefore pixels assigned by each vectors can be reconstructed more accurately.

Table 1. Experimental Design Algorithm

Algorithm for optical flow (LK or HS) calculation and the performance evaluation

Input: 126 one side of fish-eye images
Output: reconstructed image, vector flows (u & v), motion compensated, & PSNR
1: ***for** each two successive images, **do***
2: *$im1_{st}$ & $im2_{nd}$ ← convert the two images to grayscale*
3: *$im1_{st}$ & $im2_{nd}$ ← smooth the two grayscale images*
4: *I_x, I_y & I_t ← obtaining spatial and temporal gradient of Images*
5: *u & v ← calculating vector flows*
6: *im1R ← $im1_{st}$ moved by u & v*
7: *im_prediction ← subtract $im2_{nd}$ by im1R*
8: *PSNR ← 10 $\log_{10}$ im_prediction*
9: ***end for***
10: ***Return***

5. References

1. Shree K. Nayar, “Omnidirectional Vision”, Chapter Robotic Research, pp 195-202, DOI 10.1007/978-1-4471-1580-9_18, Springer-Verlag London Limited.
2. A. Djamal, K. Djemaa, “An Adapted Block-Matching Method for Optical Flow Estimation in Catadioptric Images”, International Conference on Multimedia Computing and Systems (ICMCS 2014), pp 69-74, DOI: 10.1109/ICMCS.2014.6911139, IEEE, 2014.
3. Wedel, D. Cremers, “Stereo Scene Flow for 3D Motion Analysis”, DOI 10.1007/978-0-85729-965-9_2, © Springer-Verlag London Limited 2011.
4. Matlab : https://www.mathworks.com/