Auto-Generation of Runner's Stroboscopic Image and Measuring Landing Points Using a Handheld Camera

Kunihiro HASEGAWA, Hideo SAITO
Graduate School of Science and Technology
Keio University
3-14-1, Hiyoshi, Kohoku-Ku, Yokohama, Kanagawa, Japan
{hiro, saito}@hvrl.ics.keio.ac.jp

Abstract

This paper discusses a method for automatically generating a stroboscopic image and measurement of landing points from a video image using a handheld camera. The purpose of this method is training of amateur runners. For the training, generating a stroboscopic image which has an arbitrary frame's runner and measurement of landing points are important to measure stride length and speed of the runner from the stroboscopic image. The proposed method for generating a stroboscopic image has two steps. First, a stitched background image is generated from the input video image. Then, an arbitrary frame's runner image is overlapped to the stitched background image. Thus, this method can generate a stroboscopic image including the runners in arbitrary frames without overwriting them in some frames. These processes can be executed automatically. The method for measurement of landing points uses a homography which is a matrix to project and transform a plane to other plane. We demonstrate the effectiveness of the proposed method by showing some experiments under various environments. The Result of the measurement as an input to the stroboscopic image obtained the measurement result whose accuracy was sufficient for analysis.

Keywords: Sports Vision, Stroboscopic Image, Runner, Overlap, Handheld Camera

1 Introduction and Motivation

In various sports, a variety of scientific analysis for them has been performed. These results contribute to players of sports for improving their skills. The information given by the scientific analysis is also useful to make a TV program interesting. Computer vision has also been used for analyzing sports. For example, Hamid et al.[1] visualized an offside line by tracking of multiple persons in football. Lu et al.[2] proposed the method for motion analysis for basketball. Atmosukarto et al.[3] performed the study of the recognition method for a formation of American Football. Beetz et al.[4] proposed the system it creating an analysis models by tracking the ball and multiple players from a video image assumed to be used in various sports.

Athletics is one of the targets of an analysis as described above. Athletics have a huge variety of items to be analyzed. A number of studies for these analyses have also been performed. For example, Yang et al.[5] proposed the method of discrimination a runner's running state by recognizing the difference of the motion of the foot from the input video image. Furthermore, studies using variety of sensors have also been performed. For example, Strohrmann et al.[6] analyzed the kinematic change with the fatigue in running. This method measures the displacement of the position and angle etc. of a hip and each joint using 12 wearable orientation sensors in the whole body of the runners initially. Analysis results are obtained by comparing these measured data. Oliveira et al.[7] visualized the change in a heart rate of runners during running with a heartbeat sensors.

Such scientific analyses have been used only for professionals. However, amateur runners' interest in such analyses is also growing with the growing popularity of a running in recent year. In targets of athletics, a runner's stride length and speed have advantages as ease visualization and clarity of the results. Thus, these items are suitable to support amateur runners. Existing methods for analyzing them are using instrument by laser or wearing GPS sensors and so on. However, using a special instrument or a trouble to wear sensors are a problem in these methods. In addition, the system, such as above study[4], does not have a simplicity of use for amateur runners. Further, it does not fit for the analysis of an athletics stride length and so on.

For these problems, we have decided to develop the method to measure a runner's stride length and speed using only one handheld camera. For realizing this goal, we need to automate a judging the landing and measuring landing points. In this paper, we discuss to create follow images from the video image captured by handheld camera and measure landing points as first step for the goal.

1. The stitched image extracted only background(Figure 1 (a))

2. The stitched image projected runners in each frame(Figure 1 (b))

In above items, creating the stroboscopic image which has an arbitrary frame's runner shown as Figure 1 (b) and measurement landing points are main outcome. These images are necessary for an automation to judge landing timings. In addition, a measurement of landing points uses the stroboscopic image. We can create the image extracted only the runner by taking the difference between them. In addition, we think that we can solve the problem to judge landing timings.

In this paper, Section 2 details our proposed method. The result of experiments are described in section 3. Section 4 discuss the result. Finally, conclusion is described in section 5.

(a) Stitched image extracted only background (b) Stitched image projected runners in each frame

Figure 1 Creating result images

2 Proposed Method

2.1 Overview of the System

Figure 2 is the overview of the system for measuring runner's speed and stride length using only one hand-held video camera. As shown in this figure, this system has three elements, generating stroboscopic image of the runner, a judgment of the landing timing and a measurement of the landing points. Among them, we developed the method of generating the stroboscopic image and a measurement of landing points in this paper. As described above, a judgment of landing timings will uses the stroboscopic image generated by the method proposed in this paper. For measurement, we input landing points manually this time because a judgment of landing section have not been complete yet.

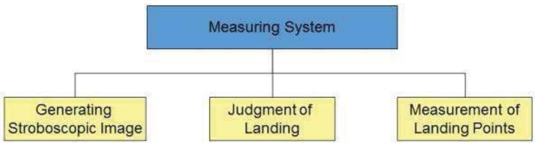

Figure 2 The overview of measuring system

2.2 Overview of Method

The goal of this method is to generate a runner's stroboscopic image and measure landing points using this stroboscopic image. For former section of this goal, generated stitched image needs to have runners of arbitrary frames. However, existing method[8] erases runners in some frames shown as Figure 3. In this figure, the runner in the second frame of input images is erased in the output stitched image. It is because the current frame overwrites the area which have the runner in a previous frame if this area have an overlap with a current frame regardless of whether it has a runner or not. Thus, we tried to solve the problem in this study. This problem is solved by two steps. The first step is generating of the stitched image extracting only background. The next step is a projection of a runner to this stitched image. The first step uses the generating method of a background image using Mean-Shift proposed by Cho et al.[9]. The second step uses the extraction of runners using HOG descriptor[10] and a projection of this runners to the stitched image.

For latter section of this goal, we developed the method using a homography which is a matrix to project and transform a plane to other plane between an image coordinate system and a world coordinate system.

2.3 Generation of Background Stitched Image

The first step is a generation of the stitched image extracting only background. We use the method proposed by Cho et al.[9] as described above. The first step is extraction of feature points from each frame of an input video. Then, a calculation of a homography between each frame uses these feature points. Finally, a stitched image is generated by this homography. Feature points are SURF[11] in this method.

The next step is a generation of the stitched image extracting only background. The extraction of a background uses Mean-Shift[12][13] for each pixel of the stitched image. Each pixel of original images projected to the same pixel in the stitched image are sample points for Mean-shift in this step. Pixels of the background area projected to the stitched image more than that of a runner. This method bases the assumption that the period existing runners on a certain location is shorter than the period there are no runner.

In the other words, pixels of the background area projected to the stitched image more than that of runner. Thus, using Mean-shift can extract a background.

In Mean-Shift processing, a pixel value of each original image is assigned as the initial value. The bandwidth is 10. The kernel formula is Epanechinikov kernel just like Cho et al. In addition, the weight for decision the result is calculated for each candidate value. Finally, the value whose weight is biggest is a background pixel value. The weight is the Gaussian in this calculation for easy processing.

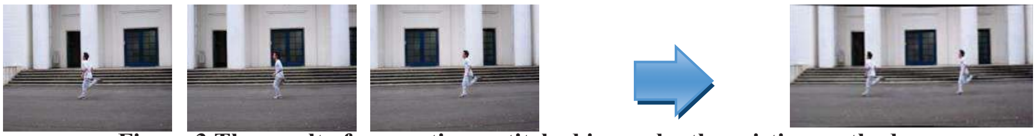

Figure 3 The result of generating a stitched image by the existing method

2.4 Overlapping of Runners

This section explains how to overlap runners in each frame to stitched image as the former section's goal of our proposed method. This process generates a runner's stroboscopic image which is main outcome of this method. Figure 4 is flowchart of this process. First, HOG descriptor[10] extracts the area of runner in every frame from input video image.

Then, this area is projected to background stitched image generated in 2.3. Calculation for this projection uses the homography of each frame used for generating stitched image. The correspondence of projection position between the stitched image and original image is also same after generating background stitched image. Thus this calculation can also use this homography. Figure 5 is the image of this process. This figure shows the example of generating the stitched image from an input video image which has n frames. If an existing method creates a stitched image, runners in some frames are overwritten all or partially shown as the lower left area of Figure 5. Our proposed method extracts the area of a runner first. The area surrounded by a frame is it in this figure. The projected area of a runner is overwritten on the stitched image. Finally, this method can create the stroboscopic image of a runner shown as the lower right area of Figure 5.

2.5 The Method of Measuring Landing Points

This section explains how to measure a runner's landing points as the latter section's goal of our proposed method. Figure 6 is the image of this method. A runner runs between two rows markers placed in a straight line at regular intervals. For measurement, we calculate the homography between the image coordinate system and the world coordinate system using these markers as known feature points. Landing points inputted manually are projected to the world coordinate by this homography. Only 4 feature points set at four corners need for calculate the homography at least. Thus, other feature points are for landmark for runner to run straight.

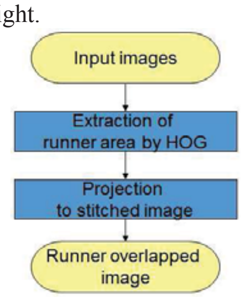

Figure 4 The flowchart of overlapping runners

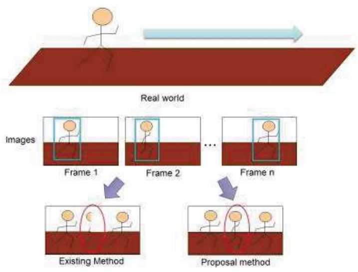

Figure 5 The image of overlapping runner process

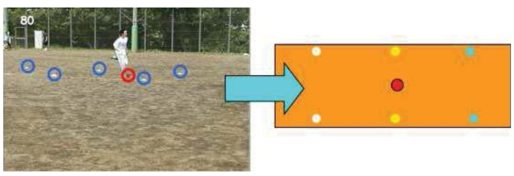

Figure 6 The image of measuring landing point

3 Experiment and Result

3.1 Overview of Experiments

This section explains the experiments about three methods described in the previous section. We experimented three methods separately to clarify each process. We used some frames extracted from the video image capturing the person is running by a video camera or digital still camera having video capture function as input for each case. As mentioned initially, the person who uses a camera holds it without using a tripod and so on and only moves a camera so as to follow runner. Further, we captured video images in three scenes that the ground surface is a concrete, dirt and grass for a generation of the background and overlapping of the runner in order to show that this method can be used in various environments. Figure 7 is examples of input images. In addition, Figure 3 is example of input images that a ground surface is a concrete.

In the experiment of measurement, Figure 8 is the examples of input image. We use the stroboscopic image created from these images for input. We needed the ground truth of landing points to confirm the accuracy of our method. For this purpose, we use the scene that the runner ran 30m on the ground whose surface is a dirt. We could get footprints made by spike shoes. We used Microsoft Visual Studio 2010 as the IDE, C++ as the programing language, OpenCV 2.4.6 as the image processing library for implementation

(a) Grass (b) dirt

Figure 7 Examples of input image for two experiments

Figure 8 Examples of input image for measurement

3.2 Generation of Background Stitched Image

In this section, we describe the result of generating the stitched image extracting only background. Figure 9 is the result of this experiments. Figure 9(a), (c), (e) are the stitched images on a concrete, grass and dirt scene respectively using "stitch" function in OpenCV. These results disappear some runners. This reason is same as Figure 3. Figure 9(b), (d), (f) are the results of using Cho's method in same scenes. However each image has a few noises influence of calculation for projection, they have no runner. Thus, we confirmed that this process is successful.


(a) Background image by OpenCV on a concrete (b) Background image by Cho's method on a concrete


(c) Background image by OpenCV on a grass (d) Background image by Cho's method on a grass


(e) Background image by OpenCV on a dirt (f) Background image by Cho's method on a dirt

Figure 9 Stitched images by extracting only background

3.3 Overlapping of Runners

We describe the result of overlapping of runners in this section as the former section's goal of our proposed method. Figure 10 is the result of this experiments. We used frames in Figure 3 and Figure 7 for overlapping of runners. In addition, the stitched image generated by the existing method for comparison are same as the previous section. Figure 10(a), (b), (c) are the results of using the proposed method on a concrete, grass and dirt scene respectively. Stitched images generated by the existing method show only runners in a part of frames. In contrast, images generated by our proposed method show all runners without disappeared. From this results, we confirmed that our proposed method can generate the stroboscopic image of runners.

3.4 Measurement of Landing Points

Finally, we describe the result of measuring landing points with the stroboscopic image as the latter section’s goal of our proposed method. Figure 11 is visualized landing points as footprints of runner generated by this experiment. We can get an appearance of landing. From this result, we can also create the graph showing a change of stride length. Figure 12 is the graph. According to this graph, for example, we can understand that this runner’s stride length extends gradually. The errors of the measurement results obtained by this method ware that the average was 0.14m, maximum was 0.61m, minimum was 0.01m, and the standard deviation was 0.14m. Some errors occur by the accuracy of the click of the landing points. However, according to preliminary experiments, the error of approximately pair of the shoes length (≒0.30m) does not affect for analyses. Thus, we confirmed that our method can get a sufficient accuracy of a measurement.


(a) Proposed method on concrete


(b) Proposed method on grass


(c) Proposed method on dirt

Figure 10 Runner's stroboscopic images synthesized by the proposed method

Figure 11 Footprint image generated by our method

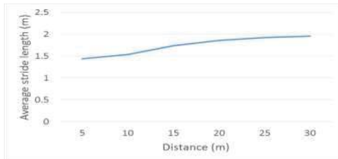

Figure 12 The graph showing stride length

4 Discussion

We confirmed that our proposed method solved the problem described in section 3 from the results of experiments. By these results, generating of a stroboscopic image part in the system for measuring a runner’s stride length and speed was completed. Measuring runner’s landing points part can measure with an accuracy which does not affect the analysis. Then we think that an automatic judgment of landing timing can be realized by using these stroboscopic images and background stitched images. As a point that should be noted, this proposed method has failure cases. The concrete examples are following three cases.

I. Too fast to move a video camera.
II. Rate of a runner on the image is too large.
III. Rate of a runner on the image is too small.

We explain briefly each case. I is that feature points are not taken by blurring occurs in the image, or not appearing a same area between frames because a camera is moved too fast. However, use cases of this method are the case that the person who stops at a certain place captures a runner. Thus, we can consider not to occur this failure. II is that feature points not obtained between frames to generate the stitched image because the area of the runner in image is too large. On the other hand, III is that area of the runner is not extracted by the HOG descriptor because the area of the runner is too small. II and III are scenes that can occur depending on the environment. Simultaneously, a technical solutions are difficult. Thus, we need to be careful at capturing.

Further, the frame rate of a video image is also important because we cannot get enough measurement accuracy for analyses if that is too small. For this point, we can use the result of preliminary experiments for to develop measurement system. In this experiments, the frame rate of video is 14.985fps and we could get the good result of analyses. Thus, we consider that there is no problem if the frame rate of video image is 14.985fps which is a half of 29.97fps (a frame rate of NTSC) or more. We used the video image deactivated interlace mode. So, the frame rate of this video image is 14.985fps.

Finally, we describe the theory of how to obtain landing timings using the result of this proposed method. First, we take the difference between the background stitched image overlapped the area of a runner and background stitched image. Only overlapped runner on each frame remains as a trajectory of the runner by this process. Of course, the existing method for extracting the human contours have been proposed are various. However, it is possible to extract a runner easily and more accurately using this method. Further, it is possible to obtain ground timings by analyzing this trajectory of runner. We think that there are to look the up-and-down motion of the runner or overlap of the runner between a current frame and a previous one for the analysis method. For example, in the former case, the change timing from descending to rising of runner's movement can be regarded as the landing timing. For this purpose, the changing point is searched in the trajectory of runner. In the latter case, we use that the runner's feet does not move almost at the landing timing. From this fact, the landing timing can be regarded as the timing that the overlapped area of the runner's feet is large between frames. We consider that these methods are applicable to judge the landing timings except failure cases described above. We are going to consider their detail in the future.

5 Conclusion

In this paper, we proposed the method of automatic generating stroboscopic image of runner and measuring landing points that they are necessary for the realization of the system for measuring runner's speed and stride length. To realize the method, we develop three sub methods, the generation of stitched image extracting a background using the Mean-Shift, overlapping of the extracted runners region using the HOG descriptor to the stitched image and measurement of landing points using a homography. The usefulness of our proposed methods is confirmed by results of experiments. Finally, we are going to develop the method of judgment the landing timings automatically using the stroboscopic image generated by this proposed method in the future.

References

[1] R. Hamid, R. K. Kumar, M. Grundmann, K. Kim, I. Essa and J. Hodgins (2010). Player localization using multiple static cameras for sports visualization. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 731-738.

[2] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy (2013). Learning to Track and Identify Players from Broadcast Sports Video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(7), 1704-1716.

[3] I. Atmosukarto, B. Ghanem, S. Ahuja, K. Muthuswamy, and N. Ahuja (2013). Automatic Recognition of Offensive Team Formation in American Football Plays. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 991-998.

[4] M. Beetz, N. v. Hoyningen-Huene, B. Kirchlechner, S. Gedikli, F. Siles, M. Durus and M. Lames (2009). ASpoGAMo: Automated Sports Game Analysis Models, International Journal of Computer Science in Sport, 8(1).

[5] Yang J. Y., Xu G. L. and Li Y. (2013). Running state recognition in videos via frames' frequency and positions of two feet. Fourth International Conference on Intelligent Control and Information Processing (ICICIP), 310-313.

[6] C. Strohrmann, H. Harms, C. Kappeler-Setz and G.Troster (2012). Monitoring Kinematic Changes With Fatigue in Running Using Body-Worn Sensors. IEEE Transactions on Information Technology in Biomedicine, 16(5), 983-990

[7] G. Oliveira, J. Comba, R. Torchelsen, M. Padilha and C.Silva (2013). Visualizing Running Races through the Multivariate Time-Series of Multiple Runners. 26th SIBGRAPI - Conference on Graphics, Patterns and Images, 99-106.

[8] Microsoft Corporation (2011). Microsoft Research Image Composite Editor (ICE), http://research.microsoft.com/en-us/um/redmond/groups/ivm/ICE/

[9] Cho, S.-H. ; Kang, H.-B. (2011). Panoramic background generation using mean-shift in moving camera environment. Proceedings of the international conference on image processing, computer vision, and pattern recognition (IPCV), 829-835.

[10] N. Dalal and B. Triggs (2005). Histograms of oriented gradients for human detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 886-893.

[11] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool (2008). SURF: Speeded Up Robust Features. Computer Vision and Image Understanding (CVIU), 110(3), 346-359.

[12] K. Fukunaga and L. Hostetler (1975). The estimation of the gradient of a density function. IEEE Transactions on Information Theory, 21(1), 32-40.

[13] D. Comaniciu and P. Meer (2002). Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), 603-619.