Automatic Detection of Pronation Type of Runners Using Computer Vision

Pečar, Anže; Dolenc, Aleš; Jaklič, Aleš; Peer, Peter and Kovač, Jure

Abstract: ***Running is a popular form of recreation. Running clubs and shops commonly provide services related to video analysis of runners. The most common service is to determine the pronation type of a runner. This is done by manually measuring the eversion angle of an ankle from a backside video of a runner on a treadmill. Therefore, there exists a need to develop applications for automatic gait analysis of runners. We have developed methods for automatic measurement of eversion angle of an ankle for subjects running on a treadmill. They measure the angle and pronation type based on human body skeleton model segmented with computer vision techniques. Tested on a group of 15 runners, these methods produced reasonably accurate results and demonstrated the potential for wider use in commercial and sport research applications. Nevertheless, since the accuracy can be further improved, we also provide some guidelines for improvements to eliminate the shortcomings.***

Index Terms: ***Gait analysis, running, motion detection, human skeleton, pronation, supination, automated video analysis, computer vision***

1. Introduction

THE increased popularity of running also increased the demand for running gait analysis.

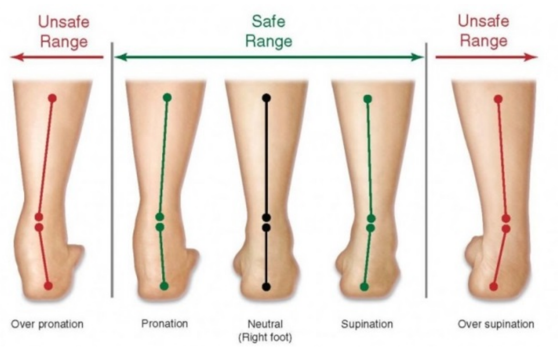

Figure 1. Pronation types

Manuscript received Sep. 2017 and accepted for the journal in Dec. 2017. The authors acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0214 (B) Computer Vision).

A. Pečar, A. Jaklič (ales.jaklic@fri.uni-lj.si), P. Peer and J. Kovač are with the University of Ljubljana, Faculty of Computer and Information Science, Večna pot 113, 1000 Ljubljana, Slovenia.

A. Dolenc is with the University of Ljubljana, Faculty of sports.

The most common part of running gait analysis is determining the pronation type of a runner (Fig 1.). This is done by measuring the eversion angle of an ankle on a video of a runner running on a treadmill. The measurements are most commonly made by hand, therefore a need for development of automatic methods of running gait subjects exist. The methods could be used for broad use in sport stores, for the purposes of running gait analysis for running trainers and also as an aid in sport research.

The aim of this study is to develop automatic methods for measuring the ankle eversion angle in runners from a video of a runner running on a treadmill filmed from behind. The aim is to determine the angle without any aid of special markers or other features that might obstruct the runners.

In the next Section we present a sequence of processing steps used to provide automatic pronation type detection in runners.

In Section 3 we experimentally evaluate the proposed solution. Section 4 summarizes the conclusions.

2. Methods

2.1 Capturing the Video Material

For the purposes of this study we have filmed 14 different runners while running on a treadmill. We filmed one runner in two different running shoes, thus obtaining 15 different videos.

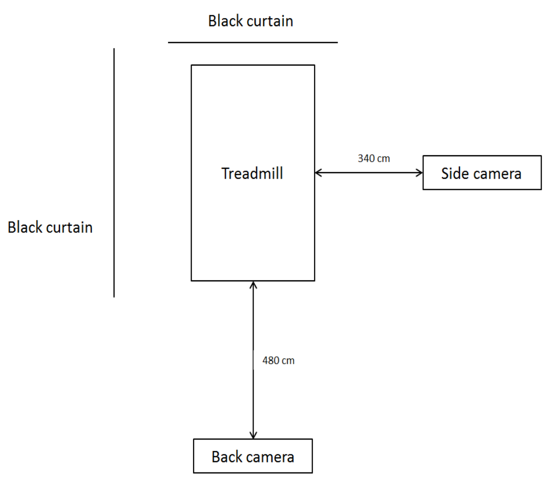

Figure 2. Cameras spatial layout

The runners have been filmed from behind. We filmed all the runners using a Lumix FZ200 camera. Camera positions can be seen in Fig. 2. The camera was positioned behind the runner on a stand, approximately 480 cm away from the runner and at a height of 50 cm.

The runners were filmed at 200 Hz and at a resolution of 640x480 pixels. We filmed at least 50 steps of running for each runner, running at a steady pace of 12 km/h. Every recording started with at least 10 seconds long clip of an empty treadmill. This was done for the purposes of learning the background for the foreground detection algorithm. All the recordings where then cut to appropriate length. They were consisted by the initial 10 s of empty scene (2000 frames), followed by 20 steps of running at 12 km/h (approximately 3500 – 4200 frames). We filmed 10 male runners and 5 female runners. The runners differed by the style of running, form and pronation type and were all recreational runners.

2.2 Automatic Pronation Type Detection

Our solution consists of 5 phases presented in Fig. 3. In the first phase, the foreground detection algorithm [1] is used to produce a silhouette of a runner without the background. This silhouette is than further processed by morphological operations on images to produce a cleaner silhouette. On this cleaner silhouette we then compute the center lines of the skeleton and the point representing the ankle. This produces a video equipped with lines, representing the runner's skeleton and the information about the eversion ankle for each foot of a runner in every step [2]. Afterwards we process the obtained signals to calculate the average step and average values through all the steps of a runner. On these values we then use 3 different methods for determining the support phase. Finally, we calculate the average eversion angle in the ankle during the support phase.

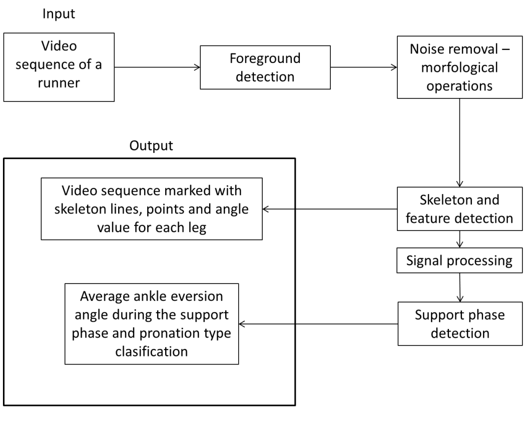

Figure 3. Processing phases

2.3 Foreground Detection

Foreground detection obtains the model of the background. The model of the background is obtained by adaptive background mixture models algorithm [1]. The number of training frames used for the function was 15 and learning rate was set to 10^{-8}. All the other parameters were set to their default values. The foreground detection was done in RGB color space.

2.4 Morphological Operations

Morphological operations on images where used to obtain a cleaner mask of a runner. We first used morphological close [3], to fill smaller holes in the mask Then morphological open [3] operation was used to remove any smaller noise in the mask. This operation was followed by dilation [4] to fill any bigger holes in the mask. For the structural element a square of size 6 is used. We remove all objects smaller than 7500 pixels from the frames. The resulting mask after the morphological operations is shown in Fig. 4.

Figure 4. Foreground mask of the runner

2.5 Skeleton Detection

We segment the obtained mask of a runner into 3 segments (Fig. 5) that where determined manually based on the estimated heights of runner's joints.

The upper segment contains the runner's hips, the middle segment contains the upper part of runner's legs and the bottom segment contains the lower part of the legs We processed each segment individually [2]. In the upper segment only the line through the center of the segment was calculated using Hough line detection algorithm [5].

In the middle segment we obtained the lines running down the center of each leg. We calculated the center point for each part of the segment representing the individual leg. Only the points close to the average center point of each leg where used for the line calculation, because the human legs are fairly straight, therefore the points cannot vary significantly from the average center point.

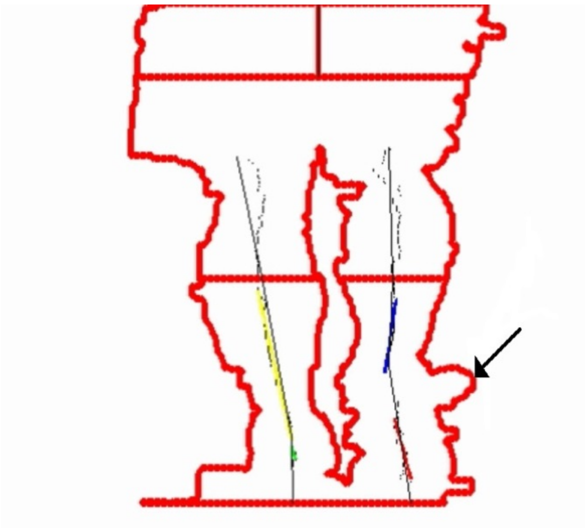

Figure 5. The 3 segments of the mask

In the lower segment, also the center points for each leg where obtained by similar methods as in the middle segment. The points where then divided into two groups, separated by the point representing the ankle.

2.6 Determining the Ankle Point

The ankle point for each leg was calculated with more care. The human leg has usually the minimum width a little above the ankle. This fact was used to determine the location of the ankle. We calculated the height at which the leg had the minimum width. We lowered that height for 20 pixels and obtained the center point for that height. This point then represented the ankle point.

2.7 Ankle Angle Computation

The center points of the lower segment have been split into two groups. The points above the ankle and the points below the ankle point. From each group a line was calculated, representing the skeleton.

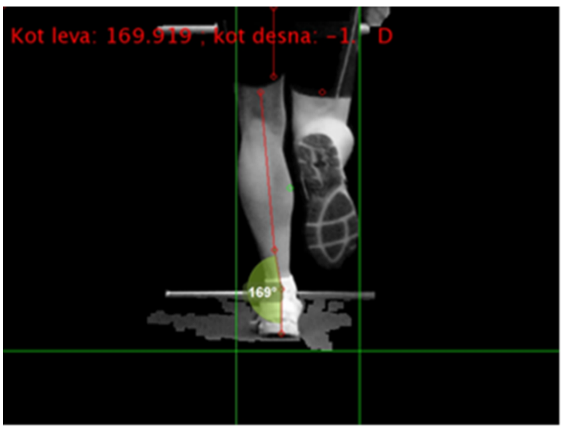

Figure 6. The calculated angle

We then calculated the angle between those two lines, to obtain the eversion angle of an ankle. The angle was calculated using the standard formula for angle between lines using the line coefficients k_1 and k_2 :

$$\tan\varphi = \left|\frac{k_1 - k_2}{1 + k_1 k_2}\right|$$

The calculated angle is marked in Fig. 6.

2.8 Signal Processing

We obtain the end point of each segment of the skeleton and the angles in these points. This is the raw data [6]. We add the silhouette mass, silhouette center point, maximum length of the silhouette and maximum width of the silhouette in the surrounding of knees for easier processing. We obtain those signals before the silhouette is segmented into segments. Because running is a cyclic motion, those signals are then processed to detect the gait cycle. We use the mass of the silhouette for gait cycle determination (Fig. 7). When the cycles are obtained, we use discrete Fourier transform to obtain the frequency specter and determine the main gait frequency. Gait signals are prone to erratic detections, which are seen as spikes in gait signals [6]. The spikes are detected by applying a moving median filter through entire signal. Moving median filter enables local detection of spikes. When a spike is detected, the values are set to undefined NaN.

Fig. 7 presents the periodic time evolution of computed features.

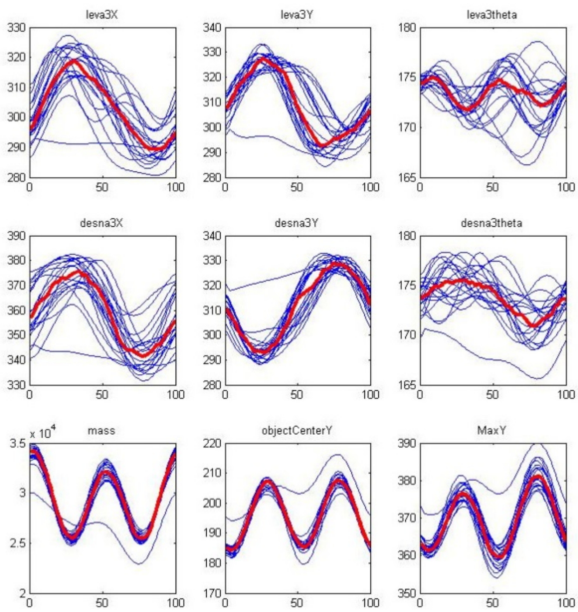

Figure 7. Periodic nature of detected features

Features leva3X, leva3Y and leva3theta represent x, y and angle of a left ankle respectively. The same goes for features desna3X, desna3Y and desna3theta for a right ankle. The feature mass represents the binary silhouette mass, objectCenterY the y coordinate of center of mass, while MaxY represents the maximum y coordinate of the silhouette.From the obtained gait cycles we then calculate the average gait cycle for all variables [6].

2.9 Support Phase Detection

We use three different methods for support phase detection.

The first method is manual. We manually determine in which part of the cycle the foot is in contact with the floor, to determine the support phase [7].

The second method is based on the height of the ankle point. We specify a fixed interval to which the height of the angle point has to correspond.

The third method is also based on the height of the ankle point. We search for spikes in the signal of Y coordinate of the ankle point, obtaining the point where the ankle point is at its lowest. We than set the fixed surrounding round that point as the support phase. If both legs have spikes roughly in the same area, we use the second spike for the other leg, at least one third of a gait cycle length away.

After we obtain the part of the cycle representing the support phase for each leg, we calculate the average eversion angle of an ankle during the support phase for each leg.

3. Results

We obtained results for all 15 runners manually by measuring the maximum angle in the ankle for all steps of running using Kinovea software [8]. Then we calculated the average angle for each leg for each runner. Those results served as the comparison for automatic method results. We also classified the runners into pronation types based on the measured angle. The runners with angles smaller than 168^{o} where classified as over-pronators, the runners having angles greater than 173^{o} where classified as under-pronators (supinators) and the runners having angles between those values where classified as neutral runners.

All three methods used automatic skeleton and angle detection, they differed regarding the method used for support phase detection:

1. The first method used the manual support phase detection.
2. The second method used automatic support phase detection by selecting an interval of the angle point height in which the point has to fall in.
3. The third method used automatic support phase detection, using a fixed interval around a detected spike, where the ankle point is at its lowest height.

We compared the results with the manual measurements and calculated the absolute and relative error for all methods. Absolute error was calculated using a formula:

$$abs_i = | V_i - R_i |,$$

where V_i is the value obtained by the automatic method and R_i is the value obtained by the manual measuring. The relative error was calculated using the formula:

$$rel_i = | V_i - R_i | / R_i .$$

The relative errors for all methods are displayed in Table 1.

Table 1 Relative error for all methods

Method	L leg	R leg	Both legs
1st	1,16%	1,28%	1,22%
2nd	1,06%	1,15%	1,10%
3rd	1,28%	1,28%	1,28%

3.3 Classification Accuracy

We classified the runners into pronation types based on the automatic methods results and compared them to the manual classification. We achieved the average classification accuracy of approximately 69%. The results can be seen in Table 2.

Table 2 Classification accuracy for automatic methods

Method	L leg	R leg	Both legs
1st	67 %	73 %	70 %
2nd	67 %	73 %	70 %
3rd	67 %	67 %	67 %

However, if we only consider the runners that were not near a limit between different classifications, the classification accuracy is higher as can be seen in Table 3. In this classification result only the runners with manually measured angles that differed for minimum of 1^{o} from the classification limits were used.

Table 3 Classification accuracy for automatic methods for runners not near the class limits

Method	L leg	R leg	Both legs
1st	78%	83%	80%
2nd	78%	100%	87%

3rd	89%	83%	87%

4. CONCLUSIONS

In this study we propose some prototypes of algorithms and methods for automatic eversion angle measurement in runners running on a treadmill and automatic classification into pronation types. The work can be classified as Specialization (S) of work [6, 10] according to classification proposed in [9].

We filmed 14 different runners while running on a treadmill and obtained 15 different videos on which we tested our algorithms. We developed algorithms that measure the eversion angle during the support phase on videos of the runners from behind without any additional aids and markers or additional lighting. The results are in form of the average angle during the support phase for each leg and also as a visual result of a video of a runner enhanced with the information about the current angle in the ankle of each foot and also with the lines representing the skeleton. The obtained results where then analyzed and compared with the manual measurements.

The proposed methods gave some promising and adequate results. The smallest relative error was less than 1%, but the accuracy should be improved to 0.5% or less. The classification accuracy was higher in runners that weren’t on the verge of classification classes. With the runners on the verge of individual classes, the accuracy is worse, because the borders between classes are very strict and offer little tolerance and the detection error is too large.

REFERENCES

[1] Bowden R., KaewTraKulPong P., "An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection," 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01, 2001.

[2] Yoo J., Nixon M. S., "Automated Markerless Analysis of Human Gait Motion for Recognition and Classi," ETRI Journal, 33, pp. 259 - 266, 2011.

[3] Laganiere R., OpenCV 2 Computer Vision Application Programming, Packt Publishing, 2011.

[4] Bradski G., Kaehler A., Learning OpenCV, O'Reilly Media, 2008.

[5] Hart P. E., Duda R. O., "Use of The Hough Transformation to Detect Lines and Curves in Pictures," Comm. ACM 15, pp. 11 - 15, 1972.

[6] Kovač J., Peer P., "Human Skeleton Model Based Dynamic Features for Walking Speed Invariant Gait Recognition," Mathematical Problems in Engineering, 2014.

[7] Čoh M., et al, Biomehanika atletike, Ljubljana: Faculty of Sport, University of Ljubljana, 2001.

[8] "Kinovea - a microscope for your videos," 2014. [Online]. Available: http://www.kinovea.org/.

[9] Blagojević, V., et al, “A Systematic Approach to Generation of New Ideas for PhD Research in Computing,” Advances in Computers, Elsevier, Vol. 104, 2017, pp. 1-31.

[10] Kovač, J., “Vision Based Techniques for Human Gait Analysis and Their use in Biometrics,” PhD thesis, Faculty of Computer and Information Science, University of Ljubljana, 2014

A. Pečar earned a BSc degree from University of Ljubljana, Faculty of Computer and Information Science, Slovenia in 2014.

A. Dolec is an assistant professor at University of Ljubljana, Faculty of Sports, Slovenia.

A. Jaklič is a teaching assistant at University of Ljubljana, Faculty of Computer and Information Science, Slovenia. His main research interest is computer vision.

P. Peer is an associate professor at University of Ljubljana, Faculty of Computer and Information Science, Slovenia. His research spans computer vision, image-based biometry, and computer vision applications.

J. Kovač earned a PhD degree from University of Ljubljana, Faculty of Computer and Information Science, Slovenia in 2014.