CIT. Journal of Computing and Information Technology, Vol. 26, No. 1, March 2018, 29–43
doi: 10.20532/cit.2018.1003802

Body Part Extraction and Pose Estimation Method in Rowing Videos

Gábor Szűcs and Bence Tamás

Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary

This paper describes an image processing approach capable of estimating the pose of athletes exercising on indoor rowing machines in video sequences. The proposed algorithm finds and tracks the wrist, elbow, shoulder, ankle, knee, hip and head, and the line of the back also. Our contribution is twofold. The first contribution is a new background subtraction method, which can reliably separate the silhouette of athletes under some assumptions related to the videos. Furthermore, the paper introduces – as the second contribution – a skeleton fitting method to find the joints of the athletes based on the results of the background subtraction. This algorithm is based on anthropometric data and special movement patterns. The overall solution works on a real time setting in the test environment. Comparing the results, it is shown that our method surpasses the most accurate state-of-the-art general pose estimation solution for indoor rowing specific videos based on two commonly used metrics, as well.

ACM CCS (2012) Classification: Computing methodologies → Computer graphics → Image manipulation → Image processing

Applied computing → Life and medical sciences → Health informatics

Keywords: rowing machine, background subtraction algorithm, pose estimation method, contour of athlete's body, silhouette based adaptive background image, distance transformation

1. Introduction

The automatic analysis of different sports spreads increasingly, because such systems can help the athletes, especially in competitive sports. In the past decades, vast amount of visual information was gathered about sport events, trainings and tournaments. However the utilization of this information has not started yet in an organized, consequent way, except for some professional clubs of popular ball games. Analyzing the image content may help improve the motion patterns, thus it could have a great impact in many sport disciplines, such as in rowing.

The idea of our work was to provide a supporting tool with certain techniques of image processing, being part of a larger movement towards automatic movement detection in sports. The goal of our research work was to design a system, which can identify major body parts of the human body, i.e. head, back, hip, knees, ankles, elbows and wrist in video sequences taken from side view of athletes practicing on indoor rowing machines.

The first step in this information getting pipeline is to separate the rowers from the background. The *Related works* section presents kernel and GMM (Gaussian Mixture Model) based algorithms, which are compared with the suggested algorithm in the *Results* section. In this paper a new algorithm is presented, which takes advantage of the specialty of human's motion and the circumstances, e.g. the motion is a cyclic activity and it is recorded from a side view of a rower.

2. Related Works

2.1. Related Works in Sports

Biomechanical analysis deals with the motion patterns of various sports, examining the mechanical properties of the motion and suggesting modifications in this motion patterns in order to improve athletics performance. Although the field of sports analytics is quite young, basic biomechanical studies were carried out already in the 20th century [2]. Biomechanics is also often used in the sport of rowing, because rowing is a highly technical sport. Restrictions on the motion patterns lead to simple, but powerful mathematical models of rowing as suggested by [16].

There are various systems available for evaluating rowing technique (and sport technique in general). We can distinguish *sensor based* and *video based* systems. The former ones use different sensors such as simple accelerometers and GPSs of cheap mobile phones (for example: CrewNerd is a popular phone app) or more precise accelerometers and GPSs, with angle and force measurement tools (BioRow is the most popular system, at least in Europe). Video recordings are cheap alternatives, and they are often used in everyday coaching work (Wilson, 2008). Because of the huge interest in ball games, in other sports there was no significant research regarding complex processing of videos [17]. To the best of our knowledge, there are no available specialized video tools specially designed for rowing coaches. Kinovea is a free and open source software for multiple sports, which allows coaches to manually mark different points in videos. The disadvantage of this tool, that it needs manual assistance and it often "looses" the tracked points.

We can also distinguish the systems that provide *immediate feedback* from those enabling only *later* in-depth *evaluation*. A promising research area for giving immediate feedback for rowers is to use acoustic feedback information like in Sofirow online system [22]. For such a system, the big question is which features should be mapped to acoustic signals. There is another new research using sequential forward feature selection to identify the features that are most discriminative for individual rowers [11]. This paper describes a new solution capable of automatically extracting key body part positions from pure video data. We do not deal with interpreting the extracted data and giving immediate feedback, but as our system can work on a real time setting, we want to extend our system in the future.

2.2. Background Subtraction Problem

The first aim in our system is background subtraction. Separation of the background from foreground pixels in video sequences is a well established and heavily studied topic in video processing. This is a very important, but also a very difficult step in the processing pipeline. When the camera is fixed to a location, there is a reasonable supposition, that the background pixels of the image exhibit some regular behavior which can be described by statistical tools. The emerging foreground object moving through the scene will not fit into the model descibing behaviour of background pixels and is therefore detectable. After a parametric or nonparametric pixel based model (explained later) is applied, further improvements can be made using image based algorithms like meanshift [7], particle filters [19] or Kalman filters. The OpenCV 3.1 software library [15] provides some well-established statistical model based background subtraction algorithms as a built-in method.

Pixel based means that these algorithms consider only the previous values of a pixel at a given location, to decide whether the current value should be labeled as background (*BG*) or foreground (*FG*). A pixel is more probably in the background if its condtitional probability of beeing in the background is higher than the condition probability of beeing in the foreground. Thus, the ratio of the probabilities is greater than 1, which can be rewritten using the Bayes rule ($p\,(BG)$ is unconditional probability that the pixel belongs to background):

$$\frac{p\left(BG|x^{(t)}\right)}{p\left(FG|x^{(t)}\right)}=\frac{p\left(x^{(t)}\middle|BG\right)p\left(BG\right)}{p\left(x^{(t)}\middle|FG\right)p\left(FG\right)}>1 \qquad (1)$$

We can then rearrange the inequality and introduce a threshold value c_{thr} to decide if a pixel is background or not:

$$p\left(x^{(t)}|BG\right)>\frac{p\left(x^{(t)}|FG\right)p\left(FG\right)}{p\left(BG\right)}=c_{thr} \qquad (2)$$

The left side of the equation is referred to as the underlying background model as suggested by [28] and can be estimated from a training set $D_t = x\,(0), x\,(1), ..., x\,(t-1)$ created from the previous pixel values observed at this pixel location, and the pixel values $x(t)$ are d dimensional vectors.

There are two major techniques for estimating unknown statistical probability distributions:

(i) non-parametric and

(ii) parametric methods, which assume that the data follows a given probability distribution like normal distribution [26].

In this case, the goal is to estimate the parameters (e.g. mean and variance) of this distribution. The parameters can vary depending on the type of underlying distribution. However, the non-parametric methods do not assume any type of distribution of the training data. The simplest non-parametric method is the histogram method, which classifies the data into different bins and estimates the probability as the ratio of the count of examples in one bin over the total count. The big difference between parametric and non-parametric techniques is that, while the parametric methods use fixed number of parameters, the parameter number of non-parametric methods can grow with the size of the training data set.

Kernel density estimation method is a non-parametric statistical estimation based background subtractor described in [28]. The density estimation of a given vector x would be:

$$\hat{p}\left(x \middle| D_t, BG+FG\right) = \frac{1}{\left|D_t\right|}\sum_{D_t} K\left(\frac{x-x_i}{h}\right) \quad (3)$$

where in the left part of the equation, the common model is inferred from the training data set D_t, also considering the background BG and foreground FG behavior. This probability should be estimated since the true background and foreground classification is not known. K refers to the kernel function and h to the so called bandwidth, which will control the smoothness of the estimation. The kernel function can be of any type, the most common are Gaussian, uniform, triangular, biweight and triweight kernels, but the type of kernel has a minor impact on the performance of the overall background subtraction, as stated by [10].

In the practical implementation, if D_t is large (if the video is long), keeping the samples and calculating the estimations would require too much memory and processor time for average computer. Thus only a random subset of the last M samples is used. Simple random sampling would lead to too sparse sampling, thus a "short-term" and a "long-term" sample set is kept, and a denser and a sparser sampling are used respectively. The number of data samples can be controlled via a configurable parameter (e.g. half of the total number of samples in the subsample can be used as "short-term" and another half as "long-term" samples).

The most frequently used method of parametrized method family is Gaussian Mixture Model (GMM), which uses normal distribution. Improved adaptive Gaussian mixture model for background subtraction was proposed by [27]. If we have estimations for mean, variance, and weights of normal distribution (large $\mathcal{N}$ in the next equation, where I is the identity matrix), then the probability estimation will be:

$$\hat{p}\left(x^{(t)} \middle| D_t, BG+FG\right) = \sum_{i=1}^{N} \hat{\pi}_i \mathcal{N}\left(x^{(t)}, \hat{\mu}_i, \hat{\sigma}_i^2 I\right) \quad (4)$$

2.3. State-of-the-Art Methods for Human Pose Estimation

The second aim in our system is the human pose estimation, where the problem is localizing anatomical landmarks. This aim focuses on finding parts of individuals [23] and there are some new researches going on in this topic [4], [18], [20]. The method of the first paper [4] uses cascade detection heatmaps by convolutional neural networks and regression on these heatmaps. The next paper [18] also utilizes convolutional network architecture for human pose estimation, using various spatial relationships associated with the body. The authors of the third paper [20] proposed ConvNet architecture that is able to benefit from temporal context by combining information across the multiple frames using optical flow. Most approaches [14], [23] used a top-down strategy that first detects a person and then, on each detected region, the pose of the detected person is estimated. There is another recent method, Deepcut method [21], which gives good accuracy for this problem. Similarly to Deepcut, Insafutdinov *et al.* built a solution, the DeeperCut [13] with a stronger part detectors based on ResNet [12]. One of the drawbacks of the pairwise representations used in [13], which are offset vectors between every pair of body parts, is that a precise regression is difficult, and thus a separate logistic regression is required to convert the pairwise features into a probability score. Another drawback of these methods is that they assume a single person setting where the location and scale of the interested person are given.

In October 2016, there was a challenge called COCO 2016 Keypoint Challenge, that required localization of person keypoints in challenging, uncontrolled conditions. This keypoint competition involved simultaneously detecting people and localizing their keypoints. In this challenge, the Part Affinity Fields (PAF) [5] method won the first prize in all test sets. This directly exposes the association between anatomical parts of persons in an image. In this method, a two-branch network is built upon convolutional pose machines (CPM) architecture proposed by Wei *et al.* [24] to iteratively refine both confidence maps and PAFs with global image and spatial contexts. The architecture of the method is designed to jointly learn part locations and their association, via two branches of the same sequential prediction process. This prediction enables the part confidence maps and the association fields to encode global context while allowing an efficient bottom-up parsing step. Using an optimization and Part Affinity Fields, the PAF method achieves better accuracy than a graphcut optimization formula based on fully connected graph structure as in [13], [14], [21] and outperforms all methods mentioned in the previous paragraph.

3. Demand for a New Method

We selected two methods, the non-parametric kernel density estimation (kernel method) and the parametric GMM, as described in the previous section, and we used them for extraction of human body parts of a rowing athlete. One of the drawbacks of these methods is that they mark the shadows projecting to the background as foreground. There is an additional mechanism for detecting shadows, which can be connected to both methods. The idea behind the implementation is suggested by Cucchiara [8], [9]. The algorithm works in the HSV color space, because the shadows have not significant effect on hue values, but they will lower the value of the saturation. The next equation presents the calculation of the estimation of density function with kernels.

$$\hat{p}\left(x^{(t)} \middle| D_t, BG+FG\right) = \frac{1}{M} \sum_{m=M-1}^{t-1} K\left(\frac{\left\|x^{(t)} - x^{(m)}\right\|}{h}\right) \tag{5}$$

For estimation of a density function different kernels can be used. Parameters of the kernel are d dimensional vectors, so these should be handled either by crossproduct separable kernel extension or (circle or sphere) symmetric kernel extension. The type of the kernel functions has only a minor importance, however the value of the diameter h has a large impact on the estimation. Elgammal [10] suggested that $m/\left(0.68*\sqrt{2}\right)$ is a good choice, where m denotes the median of the absolute differences between consecutive vectors.

We have tried these methods with different variants and improved versions (with symmetric kernel extension), but the accuracy of segmentation was weak because of general assumptions in videos. So, the demand arised for a new method.

4. Suggested Methods for Rowing Detection

4.1. Preliminaries

In this section a real time video processing framework is presented, which is capable of reliable detection of an athlete performing a rowing workout on indoor rowing machines. The presented framework reports the position of the wrist, elbow, shoulder, ankle, knee, hip and head and the line of the back.

In our method we took advantage of the fact that rowing motion is a cyclic motion, which means that more or less the same motion is repeated several times. Rowing utilizes all body parts and all important muscle groups, resulting in complex patterns, and sometimes – in case of beginners – a lot of technique errors.

Rowers sit on a sliding seat in order to extend the movement to legs too. The rowing stroke begins at the catch position, where the legs are in full flexion, shins are vertical, the body leans slightly forward and the arms are extended, the shoulders are relaxed. Then the driving phase begins by pushing with the legs backward, while maintaining the same body position and keeping the arms straight. When the legs are almost extended, the body begins to swing to carry on accelerating the boat. Finally, with arms pull, the rower arrives at the finish position. Here, the movement (compared to the boat) changes direction and the recovery phase begins, with doing all the movements inversely: first the arms are straightened, then the body swings forward and finally the legs begin to bend and the rower arrives at the catch position again. Then this sequence is repeated several times. An example of rowing sequence is presented in [16], where the whole cycle of the rowing motion with different positions can be seen. In a rowing boat, the wrists travel along an arc, while in the case of an indoor rowing machine, the wrists move along a straight line. Except that, there is no difference between the two motions. This paper considers the case of indoor rowing, but the elaborated method can be extended to the outdoor rowing as well.

The framework relies on a couple of assumptions regarding the video and the motion. These assumptions are summarized in Table 1. The most important one is related to the viewpoint: we assume, that the plane of motion is parallel with the camera plane and the recorder is fixed. Furthermore, it is important that colors of the rower and of the background are not the same, the background and the lighting conditions are not varying during the recording. All the time, the rower's full body is captured by the recording, the rower is not too small in the images of the video (in order to able to detect all body parts), and the rower is not too close to the camera (avoiding large distortion of camera lenses). Additionally, the rower's shadow should be light or small, and other shadows should not be cast to the rower's body. The athlete should row without any extraordinary large mistake in rowing technique or any incidental motion and he should maintain average stroke rate, which is measured in water sports by a common unit, the stroke/min, abbr. spm (e.g. between 15 and 45 spm, which is actually not a strict requirement because rowers rarely do workouts below 20 spm or exceed the 40 spm stroke rate). The last assumption is that there are no other large moving objects captured by the camera.

Table 1. Video assumptions for rowing detection.

Criterion	Description/Status
lighting	not varying too much
shadow by athlete	no
shadow on athlete	no
other shadow	allowed
viewpoint	side view (camera plane and motion planes are parallel)
viewpoint	fixed
field of view	athlete is fully captured
field of view	bounding box area is not too small
rowing motion	no large rowing technique errors
stroke rate	between 15 and 45 spm
other objects	no large moving objects

Our suggested method – after the initialization – consists of two large blocks, as can be seen in Figure 1. The first block is a new background subtraction method, which can reliably separate the silhouettes of athletes, under certain assumptions related to the videos. In the next section we describe this area based on the adaptive body part extraction.

In the next section we present our suggested method for pose estimation as the second large block. This involves a new skeleton fitting method to find the joints of the athletes, based on the results of the background subtraction. The algorithm is based on the anthropometric data and special movement patterns. In the pose estimation block (Figure 1) the body parts (as points) are implemented, and there is another smaller block where the back curvature is calculated (as line). Both of them belong to the pose of the athlete's body.

4.2. Area Based Adaptive Body Part Extraction

The core idea behind the proposed method is that as the athlete moves on the scene, some of its points can be captured by subtracting the previous frame from the current frame, calcu-

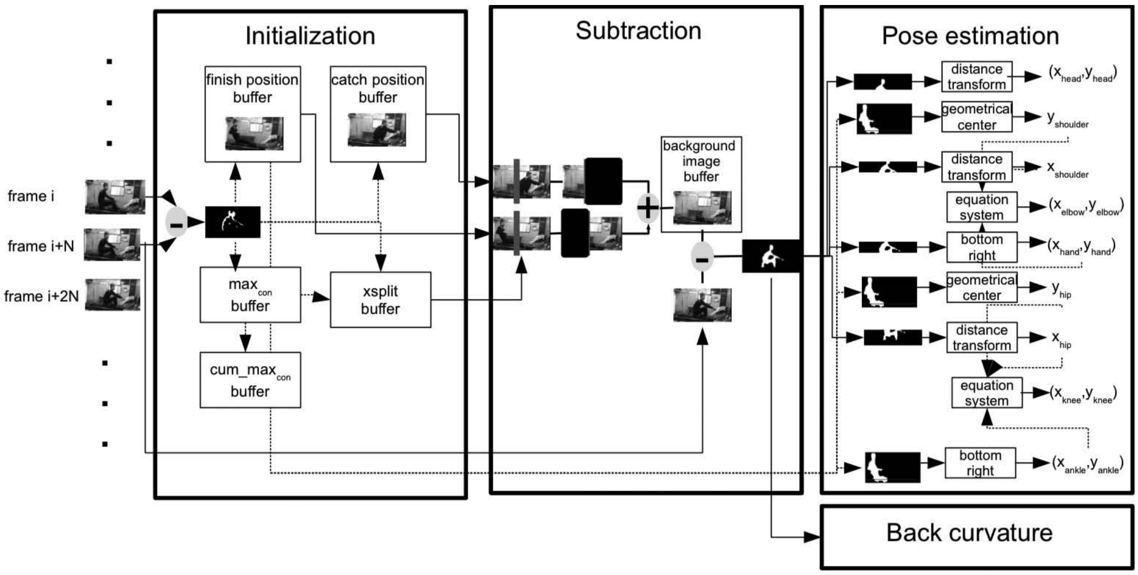


Figure 1. Block diagram of our suggested solution consisting of two contribution parts.

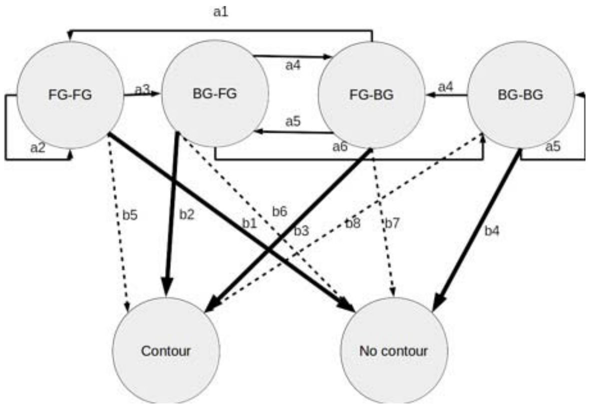


Figure 2. The states of our concept after the subtraction of neighboring frames at pixel level.

lating the absolute value, and then thresholding it. This will result in a binary image, where one value will indicate a huge change between frames, and zero value will indicate no or negligible change. We developed a concept with non-deterministic finite state machine presented in Figure 2, which shows the process on a pixel level. On the top of the figure, there are four states, which cannot be observed based on the thresholded difference image. The names denote if the pixel belongs to foreground or to background of the scene.

For example $BG-FG$ is the state where the pixel is now background pixel, but at the previous frame, the pixel belonges to the foreground. The arrows with a_1, a_2, ..., a_8 denote possible transition between the states and their probabilities. Note, that we do not know possible probabilities and do not have any method to estimate them.

The difference image encodes one of the two states from the bottom at every pixel. We do not know the probabilities of b_1, b_2, ..., b_8, but based on the assumptions summarized in the previous section, we can deduce some inequalities. For example b_1 is much larger than b_5, because if a pixel belongs to foreground at both the previous and the current frames, then the likelihood of a contour is very small (only small body parts, such as little finger or a lock of hair may need a contour between two consecutive frames in the video when this point belongs to foreground, then background, then foreground again during a unit of time), and there is a large probability that this pixel is not a contour point. Based on similar arguments, we can deduce the following inequalities:

$$b_1 \gg b_5, \quad b_2 \gg b_6, \quad b_3 \gg b_7, \quad b_4 \gg b_8 \qquad (6)$$

The $BG-FG$ and $FG-BG$ changes will occur near the athlete's contour. These states will cause most of the contours. So now if we calculate the center of the observed contour points, it is very likely that it will be in the bounding box of the rower. This is based on the assumption that the athlete does not change large position between two frames, and this will hold since we restricted the stroke rate to less than 45 spm, and a video recording contains at least 24 frames per second.

Consider Figure 3 from another aspect. If the rower is moving at the speed *v*, its height on the image plane is *h*, the time difference between consecutive frames is *Δt* and we calculate the thresholded difference of frames *i* and *i* + *N*, where *N* is a small number, then the number of the contour points (changed pixels in the *difference image*) is roughly proportional to the height of the rower and the distance he or she has moved, so the number of contour points (pixels) is:

$$|contour| \approx h \cdot v \cdot N \cdot \Delta t \tag{7}$$

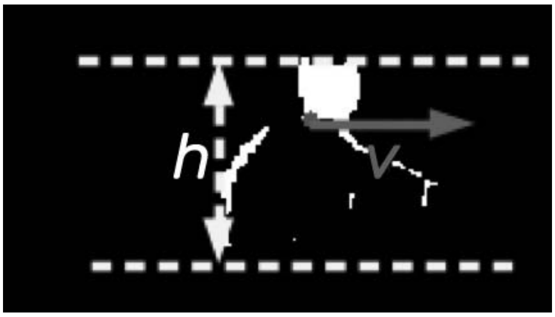


Figure 3. The subtracted contours of the athlete. Here the athlete is moving at the velocity *v* to the right, and *h* is her/his pixel height on the image.

If we observe a large number of contour points, then it means that the athlete was moving; otherwise, the rower did not change his position. In the latter case, we observe not only a small number of contour points, but these points are scattered and the order of magnitude of contour points is approximately equal to the magnitude of noise. In order to filter out these noisy observations, we will consider only appropriate (good) difference images where the number of contour points is high, due to a sufficiently large moving object (rower).

Based on these facts we suggest a new method, which is called *area based adaptive body part extraction* method consisting of two phases,

(i) producing an adaptive background image and

(ii) creating a rower's mask by background subtraction.

For the first phase we have elaborated two methods in order to get an adaptive background image – our aim was to get it because this provides a more general solution than a fixed background image does.

Our initially proposed method is *contour based adaptive background image* construction for the first phase. In this method we calculate *difference images* of frames *i* and *i* + *N*, where *N* is a small fixed number (so in the calculation of difference image, not only the current, but the previous frame is moving with *i* as well), and enumerate the number of the contour points. On the first predefined number of frames (FPS ∗ 4 would be enough, where FPS means frame per second) we measure the maximum number of contour pixels ($\max_{con}$) that appeared on the difference images. Let us consider the different images with more than $c_{thr} * \max_{con}$ contour points, where c_{thr} is an appropriate threshold (between 0 and 1). We call them *good* images, while the other images are *bad* images. The $\max_{con}$ parameter can be continuously updated with an exponentially weighted moving average scheme in order to reflect the newer observation with more weight.

For every predefined number of frames (FPS ∗ 4 frames) an independent new $\max'_{con}$ value is calculated. The cumulative $cum\_\max_{con}$ value can be computed from previous one:

$$cum\_\max_{con} = (1-\alpha) \cdot cum\_\max_{con} + \alpha \cdot \max{}'_{con} \tag{8}$$

For the good images (considered for cumulative $cum\_\max_{con}$) we calculate the center of the contours as the mean of their horizontal and vertical coordinates. To the series we apply a denoising (exponentially weighted moving average with high *α* value).

With this piece of information we can construct the (*contour based adaptive*) *background image*. We apply a morphological eroding operation (with a small rectangular structuring element to remove noise) to good images. Our method searches the catch position among these images automatically (based on extremum search in horizontal direction). Furthermore the method finds good image with finish position and stores this image.

After locating the leftmost pixel of the rowers' contour points (denote its *x* coordinate as *xsplit*), our method takes the left part of the chosen good image at catch position up to the coordinate *xsplit* and the right part of the stored good image at finish position beginning from the *xsplit* coordinate, and compose them. Note that, with this method, the shins still remain on the background image and cannot be eliminated, so the result will be noisy around them. As we acquired the background image, we can now get the foreground mask with a simple subtraction and a thresholding operation; this foreground mask will be the athlete's body. This process is shown in Figure 4, and the advantage of our method is the ability to detect the rower robustly when there are some small changes in the background.

If the rower is facing left instead of right, the described method would not allow so good segmentation (in case the method has not information about the direction), because the creation of background would be the same. To solve this issue avoiding the inaccurate background image, our method tracks the head position. If the head is further from the left side of the rower's bounding box, than from the right side of the bounding box at the (hypothesized) catch position, then the rower is facing left. Hence, the image is flipped and the whole background subtraction process is repeated resulting in a better background image.

If the background image is already available, when a new frame arrives only a subtraction and a thresholding operation are needed to compute the segmentation. Our method continuously recalculates the background image, so it can adopt to slow changes of the background. With the cost of more computation, even more sophisticated methods can be used, like modeling the background pixels with a normal distribution, or even with GMM.

We have elaborated an extended version of *contour based adaptive background image* construction, so called *silhouette based adaptive background image* construction, which consists of two steps:

(i) the first step is the *contour based adaptive background image* construction method itself, and

(ii) the second step uses the *background image* coming from the first step.

In the second step the method calculates the difference of the contour based background image (as the output of the first step) and the frame (instead of the difference between two frames as in the contour based method) which will result in a body silhouette instead of contours. Having difference images the method calculates new xsplit and thereafter a new background image is combined based on the same process shown in Figure 4. The idea behind this extended version

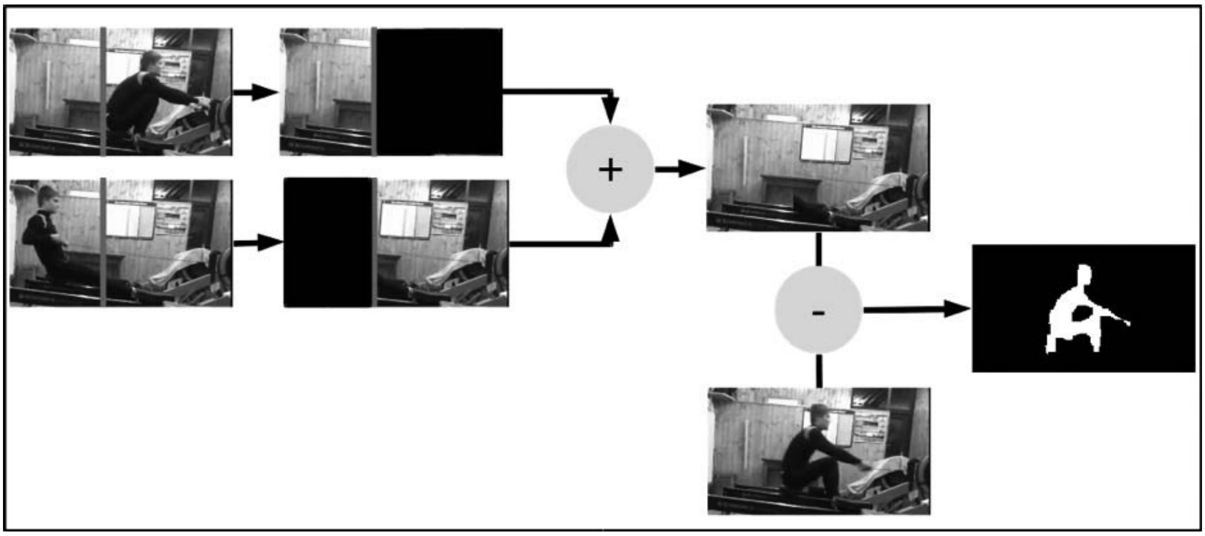


Figure 4. The process of background subtraction. On the left, good frames for catch and finish are acquired and the appropriate *xsplit x* coordinate is calculated. Then the two images are merged to form the background. With this background and the current frame a foreground mask is calculated.

was that using only one frame may lead to a better localized estimation; because in contour based background creation method (where two frames are subtracted) the contour was more "scattered". But we found that *silhouette based* solution did not improve the segmentation accuracy.

4.3. Suggested Method for Pose Estimation

The whole estimation is based on the rower's foreground mask and does not use any other visual information. It heavily relies on the anthropometric measurements published in [6] and on other temporal, spatial, behavioral and viewpoint information. The algorithm outputs the 2D image coordinates of ankle, knee, hip, elbow and shoulder, wrist, the center of head and also the line of the back, represented as series of straight lines.

The mean proportion of the head and neck-torso length in adult population is 0.339 [6]. To make sure that the whole head is captured, we use a slightly oversized, 0.38 proportion value, so we cut out the bottom region of the silhouette's bounding box, and search the center of the head in the top region. We find this point by *distance transform* [3] and finding extremum on the resulted image; where *distance transform* calculate the Euclidean distance between the investigated foreground point and the nearest background point, and the algorithm found the maximum as extremum.

The x coordinate of the hip is determined with similar calculations. However, for y coordinates we scan the frame at finish position and take the center of thighs; since thighs are horizontal, this is equal to the y coordinate of the hip, as demonstrated in Figure 5. The point of the shoulder can be determined in the same manner as in the case of head.

The ankle position can be determined as the bottom right foreground point at catch position. Since the ankle is fixed to the rowing machine, it moves only a little, and our method uses this estimation for all video frames.

For localizing the wrist, we crop the image above the thighs, then only the right side of the bounding box is kept. In this cropped image the center point is calculated by maximum point of distance transformed image.

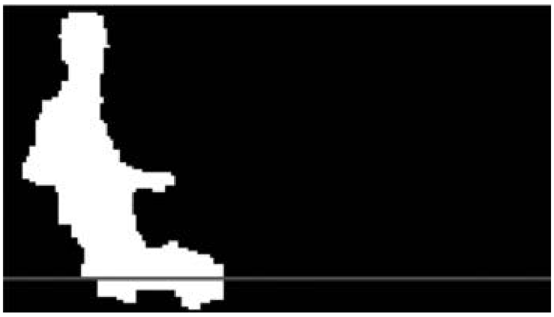


Figure 5. Determining the hip position with horizontal line, which is equal to the y coordinate of the hip.

Estimating the elbow and knee positions is more difficult. Because the wrist and shoulder or the ankle and hip positions are known and the elbow and knee move along a plane (this is true for the knee, but the elbow usually moves a little bit in perpendicular direction to the image plane which affects the accuracy of the estimation), we can use a linear equation system.

Let us assume that the hip is at position (hip $(t)x$, hip $(t)y$), and the ankle is at (ankle $(t)x$, ankle $(t)y$). The length of the thigh is l_{thigh} and the length of the shins is l_{shin} and the total leg length is l_{leg}. Now we can write

$$l_{\text{thigh}}^2 = \left(x_{\text{hip}}^{(t)} - x_{\text{knee}}^{(t)}\right)^2 + \left(y_{\text{hip}}^{(t)} - y_{\text{knee}}^{(t)}\right)^2 \quad (9)$$

$$l_{\text{shin}}^2 = \left(x_{\text{ankle}}^{(t)} - x_{\text{knee}}^{(t)}\right)^2 + \left(y_{\text{ankle}}^{(t)} - y_{\text{knee}}^{(t)}\right)^2 \quad (10)$$

$$l_{\text{leg}} = l_{\text{thigh}} + l_{\text{shin}} \quad (11)$$

From anthropometric measurements we get 0.77 for the $l_{\text{shin}} / l_{\text{thigh}}$ ratio. At the finish position the legs are straight, so the leg length can be calculated:

$$l_{\text{leg}}^2 = \left(x_{\text{hip}}^{(\text{finish})} - x_{\text{ankle}}^{(\text{finish})}\right)^2 + \left(y_{\text{hip}}^{(\text{finish})} - y_{\text{ankle}}^{(\text{finish})}\right)^2 \quad (12)$$

Now we can solve the linear equation system and get the (knee $(t)x$, knee $(t)y$) location. The elbow location is calculated in a similar manner, using the catch position for calculating the arm length.

To estimate the line of the back our method simply takes the outline of the silhouette from the bottom of the head to the line of the hip (we call it *outline method*). But at finish position the elbow will uncover the back. To solve the issue, a middle line (as several middle points found by maximum points in the distance transformed image) is applied to the torso, so the line of the spine is acquired. Then this line is shifted left to the position of the back; this is shown in Figure 6.

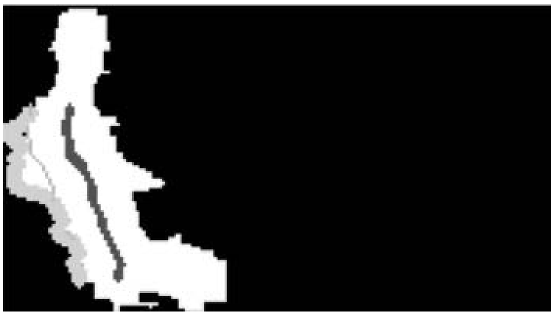


Figure 6. Determining the line of the back. If the elbow passes the back, instead of the original estimation (left thick line), a new estimation is used (middle thin line) with the help of the spine (right thick line).

5. Results

5.1. Results of the Background Subtraction

This section compares the proposed background subtraction algorithm with kernel and GMM background methods. The manual segmentation in pixel level would be a laborious work in all frames in the video, thus instead of quantitative accuracy we give only a qualitative comparison based on manual inspection of some sample videos.

Figure 7 presents an example of the foreground mask captured with the different methods. In the left, middle and right columns the *area based adaptive body part extraction* method, the GMM and the kernel method can be seen. The rows present different representative positions during the cyclic rowing activity. The middle images show bad, unrecognizable results made with GMM, especially in the top row. The kernel also achieved poor result in this case (at top right), while the area based method performed well. Our method always gives fair results in the test video sequences, while the others capture only contours or only a part of the contours (on the top middle image, the GMM achieved very poor accuracy).

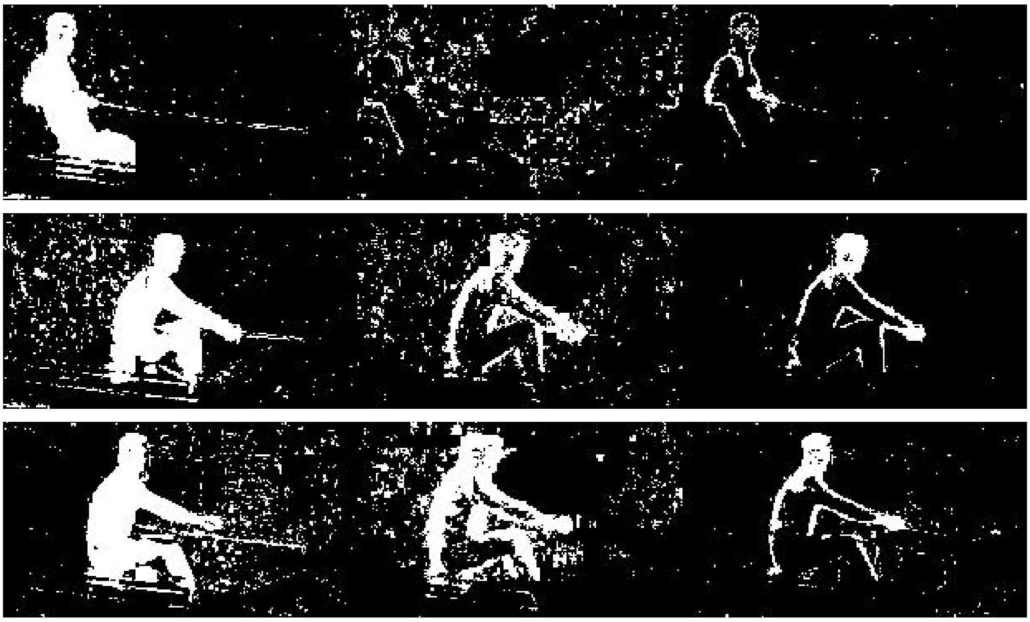


Figure 7. The results of different background subtraction algorithm on a test video. From left to right: masks acquired with the area based, GMM and kernel algorithms.

Table 2. Video assumptions for rowing detection.

Method	1	2	3	4	Average
Image resol.	206 × 168	320 × 240	212 × 117	270 × 152	
Kernel	6.7 ms	17.9 ms	5.53 ms	9.49 ms	9.89 ms
GMM	1.15 ms	3.72 ms	1.15 ms	1.79 ms	1.16 ms
Our method	0.12 ms	0.29 ms	0.13 ms	0.18 ms	0.18 ms
Our method (only subtraction)	0.06 ms	0.17 ms	0.07 ms	0.09 ms	0.09 ms

The only problem of our method is with the area near the shin. On the top left image, the shin is not captured, while on the bottom left image, "two shins" are detected, i. e. the white area below the knee is the shin, as it was approximately one second earlier. This is due to the design of the algorithm, and it can also cause problems behind the buttocks of the athlete, which can negatively affect the back estimation accuracy. But this is the trade off for fast and in other regions good and robust segmentation. Based on the qualitative test, the proposed method is more accurate, and we conducted a quantitative test for speed comparison.

Table 2 summarizes the average processing time of one video frame using different methods and various videos. As described above, after the initial phase, the proposed algorithm can reset the background generation phase, or it can use the calculated background image for a faster segmentation. In the latter case, the extra time spent on the background calculation of the first few frames will be amortized amongst all video frames. The table shows running time for both cases. Our method is implemented in c++ (OpenCV version 3.1.0.), and we get the results running on Intel Core i3 CPU with no GPU support.

5.2. Results of the Pose Estimations on Different Videos

Results of the pose estimations on different videos can be seen in Figure 8. The top three videos do not satisfy all the assumptions described in the beginning, but the estimation is still reasonably acceptable. In these cases the estimation

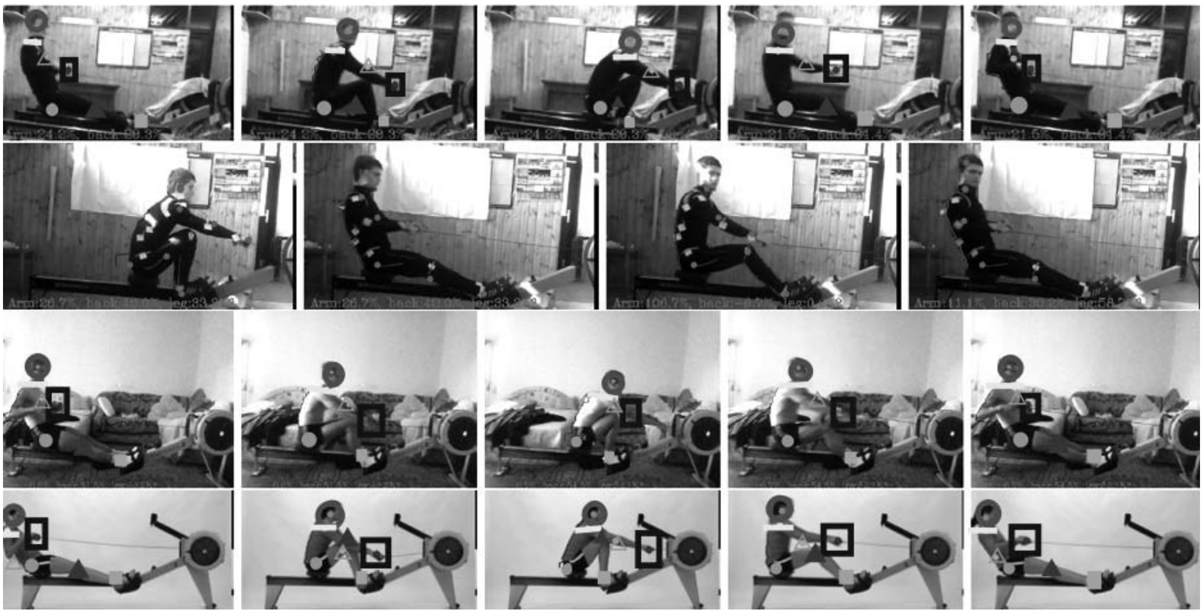


Figure 8. Results of the pose estimations on different videos. The head is marked with empty circle, the shoulder with a line, the elbow with empty triangle, the wrist with empty square, the hip with full circle, the knee with full triangle and the ankle with full square.

of the elbows, knees and sometimes back can get wrong. In the bottom video the proposed algorithm produces very accurate estimation. The head is marked with empty circle, the shoulder with a line, the elbow with empty triangle, the wrist with empty square, the hip with full circle, the knee with full triangle and the ankle with full square. In the second row, all points are marked with a circle, because of the colorful sticky notes put on the clothing. The line of the back is marked with a line (the left three pictures in the second row are the results of *outline based* back estimation, and the right picture is achieved by distance transform method).

For evaluating our results, we have collected a small testing dataset, containing six 15 minute long indoor rowing workouts, at various stroke rates and intensities and with four different people and two different backgrounds. Example frame from the video recording is shown in Figure 9. We have automatically collected the ground truth positions of the body parts using brisk sticky notes, as can be seen in the figure. The positions are derived from the images after a thresholding operation, separately for every individual color. These sticky notes do not affect the behavior and performance of our method, moreover, they enabled us to make an experimental dataset with ground truth positions. The colors had no impact on the process, since the predictions rely more on shapes than colors. For comparison, we have tested our dataset with the PAF method [5], which is one of the most accurate state-of-the-art general pose estimation solutions, as mentioned in the previous, Related works section.

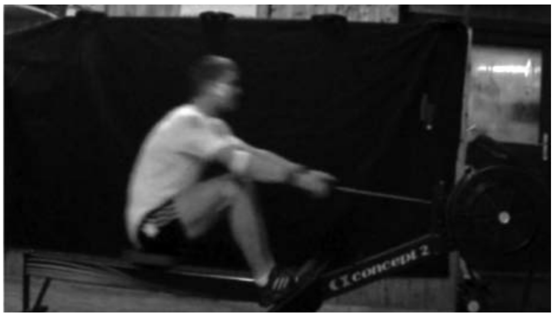


Figure 9. Example frame from the video recording for collect the ground truth positions.

Table 3 shows the RMSE (root mean squared error) of the predictions on different body parts (video resolution was 584 × 276) both for our method and the method by Cao *et al.* [5]. The error is defined as the Euclidean distance of the prediction and ground truth value (note that the back estimation is not included). The values are presented in pixel values and also in "torso lengths", i.e. compared to the length of the torso, which gives normalized values (this is independent of the image size). The average RMSE is 6.3 px for our method, while it is 7.6 px for Cao's method.

In order to compare the solution suggested in this paper with another one, we evaluated the results based on another metric, the so called PCK, as well. PCK (Percentage of Correct Keypoints) [1] is a widely used metric in the pose estimation literature. PCK@r is the percentage of predicted keypoints that lie within radius *r* of the ground truth value. It is usually also normalized to ground truth torso length for more comparable results (typical value of parameter

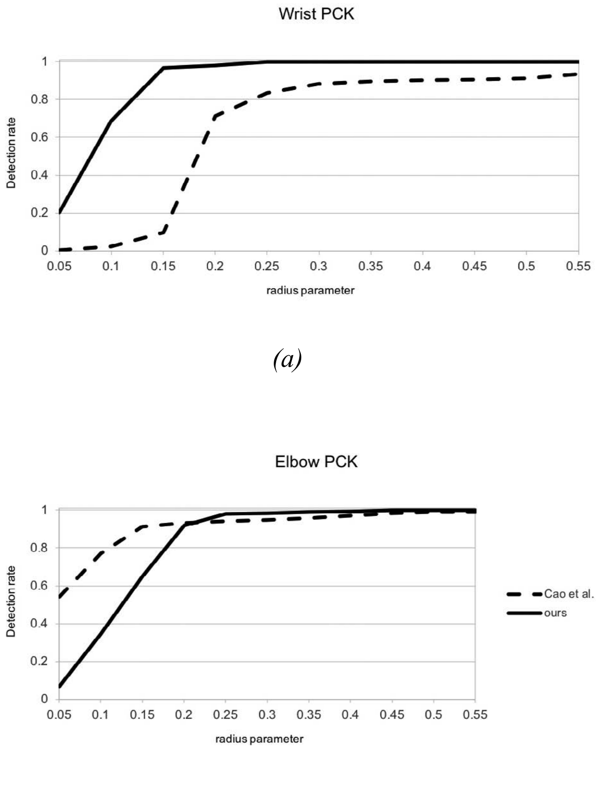
Table 3. Comparison of methods with measured RMSE of the pose estimation.

Body Part	**Cao *et al.***	**Our solution**	**Cao normalized**	**Our normalized**
Ankle	3.09	8.80	0.021	0.059
Knee	3.63	5.54	0.024	0.037
Hip	8.18	6.45	0.055	0.043
Wrist	18.81	5.17	0.113	0.034
Elbow	7.23	7.99	0.048	0.054
Shoulder	5.83	7.74	0.039	0.052
Head	8.21	2.45	0.055	0.016
Average	7.57	6.31	0.051	0.042

r is 0.2). In Table 4 our solution is compared with the method by Cao *et al.*, and it can be seen that our method reaches better average PCK@0.2 value than Cao's solution. Furthermore, at 3 body parts it reaches the maximum 1 value. Wrist and elbow can be important in the later analysis of the rowing, thus these body parts are investigated in more details, Figure 10 (a) and (b) shows the PCK values of the wrist and elbow predictions respectively.

Table 4. Comparison PCK@0.2 results at the pose estimation.

Body Part	**Cao *et al.***	**Our solution**
Ankle	0.999	1
Knee	0.998	0.984
Hip	0.879	1
Wrist	0.711	0.98
Elbow	0.931	0.918
Shoulder	1	0.825
Head	0.948	1
Average	0.924	0.958



(a)

(b)

Figure 10. PCK metrics for wrist and elbow evaluation.

Experimental results show that our average error in the pose estimation is only 4.2% of the length of torso (i.e. 0.042 normalized value) compared to 5.1% of the method (PAF) developed by Cao *et al*. In the comparison we selected PAF as one of the most accurate methods and we measured the PCK@0.2 results at the pose estimation, and our solution exceeds the currently best results (0.958 vs. 0.924). Based on these results we can conclude that our method surpasses the state-of-the-art method for indoor rowing specific videos.

6. Conclusion

In competitive sports, tracking of human body's parts and the whole pose of athlete is essential. In this paper new methods are presented, which are – under the given assumptions – capable of extracting the head, shoulder, elbow, wrist, hip, knee, ankle and back positions of a rowing athlete from video sequences. We propose a problem specific, fast and accurate background subtraction method, which surpasses two other methods widely used in literature; we also present a pose estimation approach to extract the interested points. The first contribution is a new background subtraction method, which can reliably separate the silhouette of athletes. By adaptive background image our method is able to detect rower robustly when there are some small changings in the background. This is a large advantage compared to the solutions using fixed background image, and another benefit is the speed of our method. So comparing it with GMM and kernel methods, we can conclude that our rowing specific method performs better than general background subtraction methods. The second contribution is a skeleton fitting method to find the joints of the athletes based on the results of the background subtraction. Our estimation is based on the rower's foreground mask and on the anthropometric measurements of human body. We also investigated the best state-of-the-art human pose estimating method, the so called Part Affinity Fields (PAF) method which also uses anatomical parts of the body, but this directly predicts them in the image. The test results show that our solution slightly outperforms the PAF method. Based on the comparison, the conclusion is that the whole procedure with our background-foreground segmentation and solving the linear equation system for anatomical parts is important and that these phases in the procedure can help each other (a good foreground mask can improve final results of the pose estimation as well).

References

[1] M. Andriluka *et al*. "2d Human Pose Estimation: New Benchmark and State of the Art Analysis", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3686–3693. http://dx.doi.org/10.1109/CVPR.2014.471

[2] W. Baumann, "Basics of Biomechanics" (in German), Verlag Karl Hofman, 1989.

[3] G. Borgefors, "Distance Transformations in Digital Images", *Computer Vision, Graphics, and Image Processing*, vol. 34, no. (3), pp. 344–371, 1986. http://dx.doi.org/10.1016/S0734-189X(86)80047-0

[4] A. Bulat and G. Tzimiropoulos, "Human Pose Estimation via Convolutional Part Heatmap Regression", in *European Conference on Computer Vision (ECCV)*, 2016, pp. 717–732. http://dx.doi.org/10.1007/978-3-319-46478-7_44

[5] Z. Cao *et al*., "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields", *CVPR* 2017. http://dx.doi.org/10.1109/CVPR.2017.143

[6] E. Churchill *et al*., "Anthropometric Source Book", Volume I: Anthropometry for Designers. 1978.

[7] D. Comaniciu and P. Meer, "Mean Shift: A Sobust Approach toward Feature Space Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002. http://dx.doi.org/10.1109/34.1000236

[8] R. Cucchiara *et al*., "The Sakbot System for Moving Object Detection and Tracking", in *Video-Based Surveillance Systems*, pp. 145–157, 2002. http://dx.doi.org/10.1007/978-1-4615-0913-4_12

[9] R. Cucchiara *et al*., "Detecting Moving Objects, Ghosts, and Shadows in Video Streams", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1337–1342, 2003. http://dx.doi.org/10.1109/TPAMI.2003.1233909

[10] A. M. Elgammal *et al*., "Non-parametric Model for Background Subtraction" in *Proceedings of the 6th European Conference on Computer Vision, part II*, 2000, pp. 751–767. http://dx.doi.org/10.1007/3-540-45053-X_48

[11] F. Gravenhorst *et al*., "Identifying Unique Biomechanical Fingerprints for Rowers and Correlations with Boat Speed-A Data-Driven Approach for Rowing Performance Analysis", *International Journal of Computer Science in Sport*, vol. 14, no. 1, 2015.

[12] K. He *et al*., "Deep Residual Learning for Image Recognition", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (ICCV)*, 2016, pp. 770–778. http://dx.doi.org/10.1109/CVPR.2016.90

[13] E. Insafutdinov *et al*., "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model", in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 34–50. http://dx.doi.org/10.1007/978-3-319-46466-4_3

[14] U. Iqbal and J. Gall, "Multi-Person Pose Estimation with Local Joint-to-Person Associations", in *Proceedings of European Conference on Computer Vision (ECCV)*, 2016, pp. 627–642. http://dx.doi.org/10.1007/978-3-319-48881-3_44

[15] A. Kaehler and G. Bradski, "Learning OpenCV", O'Reilly Media, Inc., 2014.

[16] V. Kleshnev, "Biomechanics of Rowing", Crowood Press Limited, 2016.

[17] T. W. Miller, "Sports Analytics and Data Science: Winning the Game with Methods and Models", FT Press, 2015.

[18] A. Newell *et al*., "Stacked Hourglass Networks for Human Pose Estimation" in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016, pp. 483–499. http://dx.doi.org/10.1007/978-3-319-46484-8_29

[19] K. Nummiaro *et al*., "An Adaptive Color-based Particle Filter", *Image and Vision Computing*, vol. 21, no. 1, 99–110, 2003. http://dx.doi.org/10.1016/S0262-8856(02)00129-4

[20] T. Pfister *et al*., "Flowing Convnets for Human Pose Estimation in Videos", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1913–1921. http://dx.doi.org/10.1109/ICCV.2015.222

[21] L. Pishchulin *et al*., "Deepcut: Joint Subset Partition and Labeling for Multi Person Pose Estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4929–4937. http://dx.doi.org/10.1109/CVPR.2016.533

[22] N. Schaffert, and K. Mattes, "Designing an Acoustic Feedback System for On-Water Rowing Training", *International Journal of Computer Science in Sport*, vol. 10, no. 2, pp. 71–76, 2011.

[23] M. Sun and S. Savarese, "Articulated Part-based Model for Joint Object Detection and Pose Estimation", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 723–730.
http://dx.doi.org/10.1109/ICCV.2011.6126309

[24] S. E. Wei *et al.*, "Convolutional Pose Machines", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4724–4732.
http://dx.doi.org/10.1109/CVPR.2016.511

[25] B. D. Wilson, "Development in Video Technology for Coaching Sports Technology", vol. 1, no. 1, pp. 34–40, 2008.
http://dx.doi.org/10.1002/jst.9

[26] R. S. Witte and J. S. "Statistics", Wiley, 9th edition, 2009.

[27] Z. Zivkovic, "Improved Adaptive Gaussian Mixture Model for Background Subtraction", in *Proceedings of the Pattern Recognition*, *17th International Conference on (ICPR'04)*, 2004, vol. 2, pp. 28–31.
http://dx.doi.org/10.1109/ICPR.2004.1333992

[28] Z. Zivkovic and F. van der Heijden, "Efficient Adaptive Density Estimation per Image Pixel for the Task of Background Subtraction", *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.
http://dx.doi.org/10.1016/j.patrec.2005.11.005

Received: October, 2017
Revised: April, 2018
Accepted: April, 2018

Contact addresses:
Gábor Szűcs
Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics,
Budapest, Hungary
e-mail: szucs@tmit.bme.hu

Bence Tamás
Department of Telecommunications and Media Informatics,
Budapest University of Technology and Economics,
Budapest, Hungary
e-mail: tamasbence92@gmail.com

Gábor Szűcs received the MSc in electrical engineering and PhD in computer science from the Budapest University of Technology and Economics (BME) in 1994 and in 2002, respectively. His research areas are data and multimedia mining, content based image retrieval, and semantic search, where he has published more than 100 publications. He is an associate professor at the Department of Telecommunications and Media Informatics of BME, and is the head of the research group DCLAB (Data Science and Content Technologies). Dr. Szűcs the recepient of the János Bolyai Research Scholarship, which is awarded by the Hungarian Academy of Science.

Bence Tamás is a graduate student at the Budapest University of Technology and Economics (BME), from which he received the BSc and MSc degrees in computer science (specialization in media informatics and later in data science). His research efforts cover image and video processing, sport analytics, IoT, and data science.