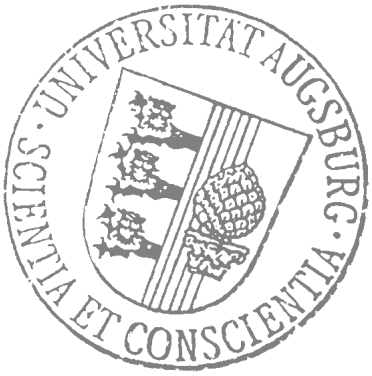
Swimmer detection and pose estimation for continuous stroke rate determination

Dan Zecha, Thomas Greif, Rainer Lienhart

Angaben zur Veröffentlichung / Publication details:

Zecha, Dan, Thomas Greif, and Rainer Lienhart. 2011. "Swimmer detection and pose estimation for continuous stroke rate determination." Augsburg: Universität Augsburg.

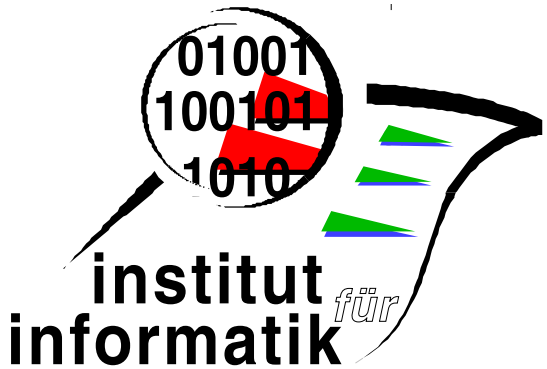
Universität Augsburg



Swimmer Detection and Pose Estimation for Continuous Stroke Rate Determination

D. Zecha, T. Greif, R. Lienhart

Report 2011-13 Juli 2011



Institut für Informatik
D-86135 Augsburg

Swimmer Detection and Pose Estimation for Continuous Stroke Rate Determination

Dan Zecha, Thomas Greif, Rainer Lienhart

Multimedia Computing Lab, University of Augsburg, Germany

ABSTRACT

In this work we propose a novel approach to automatically detect a swimmer and estimate his/her pose continuously in order to derive an estimate of his/her stroke rate given that we observe the swimmer from the side. We divide a swimming cycle of each stroke into several intervals. Each interval represents a pose of the stroke. We use specifically trained object detectors to detect each pose of a stroke within a video and count the number of occurrences per time unit of the most distinctive poses (so-called key poses) of a stroke to continuously infer the stroke rate. We extensively evaluate the overall performance and the influence of the selected poses for all swimming styles on a data set consisting of a variety of swimmers.

Keywords: Object detection, pose estimation, stroke rate estimation, swimming channel

1. INTRODUCTION

One of the most essential tasks in competitive sports nowadays is the assessment and active improvement of an athlete's technique using video assisted performance diagnostics. Here, the athlete is recorded during training sessions and competitions, while the footage is manually evaluated afterward in order to analyze the overall performance of the athlete and the contributing parts. Such an approach is extremely helpful in swimming, where the stroke rates, the body postures, and the different phases of a stroke cycle are evaluated extensively, especially for world class athletes.[1–3]

Although such an extensive performance diagnostic is indispensable for improving a swimming technique, this time-consuming and exhausting task is nowadays still done manually. Recent progresses in image processing as well as computer vision, however, can help to automate such tedious tasks in order to save time and resources.

In this work we introduce a novel approach to measure a swimmer's stroke rate in a swimming channel. We train a pictorial structure model for object detection[4] in a specific way such that not only a swimmer but also his/her specific recurring key poses can be detected. We show that with suitable post–processing of the raw pose signals we can easily recover a swimmer's stroke rate from the temporal occurrence of such characteristic poses. We extensively evaluate the proposed approach for the four major swimming styles on a data set including a large variety of swimmers such as male/female adult and age group swimmers at different swimming speeds.

There are two main reasons why we restrict ourselves to videos from a swimming channels. (1) Most pools, where competitions are held, either don't feature suitably mounted cameras or one does not have access to the footage, e.g., due to exclusive broadcast contracts. Thus, videos are mostly recorded by the coaching staff using a hand-held camcorder, which only shows the swimmer above water. Clearly, even for manual performance diagnostics the possibilities are limited in such videos. Using footage of a swimming channel where the swimmer can be observed over and under water, however, allows for a far more sophisticated analysis. (2) Having control over environmental parameters like stream velocity, illumination, and other influences causing additional noise allows us to evaluate our approach in great detailer and also to determine its limits.

1.1 Related work

While automatic performance diagnostics from videos are already successfully applied to other sports like soccer,[5] running or skiing, previous work on swimming has focused mainly on the detection of people in aquatic environments.[6,7] Such approaches are used to automatically detect drowning incidents[8,9] for swimming pool surveillance.[10] Other works carry out performance diagnostics from data captured by wearable hardware sensors[11,12] rather than from videos. Ries et al.[13] detect a single key pose of swimmers in various aquatic environments.

These detections in turn can be used to initialize motion models for cyclic motions learned for the different strokes.[14] These approaches, however, were never applied to analyzing specific phases of a swim stroke nor to extracting stroke frequency measurements based upon the detection of recurring poses. To our best knowledge there has been no prior work on automatically recovering the stroke raten directly from video sequences.

2. APPROACH

In this section we describe how the stroke rate is automatically computed using a set of specifically trained object detectors. We also introduce the training data and explain in detail how it was prepared to train our object models for pose detection.

2.1 Overview

We exploit the recently by Felzenszwalb et al. proposed approach for object detection[4] to estimate a swimmer's pose and subsequently a swimmer's stroke rate in a swimming channel. This approach treats each object as a combination of deformable parts that are connected by some underlying structure. Since the appearance of many objects varies significantly when observed from a different point of view, each model consists of a mixture of submodels. Each submodel is trained on a specific view of the object. All trained submodels are combined to form an overall mixture model that can be used to for robust object detection in images.

We take advantage of the above mentioned mixture models and train one model per swimming style. As the original approach each model consists of submodels. However, in our case these submodels do not represent different viewing points, but instead each submodel is responsible for the detection of a different pose of a stroke cycle. Poses that can be detected reliably are called *key poses*. In order to determine the stroke rate, we identify key poses within a swimming cycle throughout a video. Counting the number of their occurrences per time unit yields the stroke rate.

We will use the terms *submodel* and *pose model* interchangeably throughout this work and denote a mixture model made up of n pose models by an *n-pose model*.

2.2 Training set preparation

Four major swimming styles have been established in traditional swimming competitions: breaststroke, butterfly, backstroke, and freestyle. We assume that the swimming style of the swimmer is known beforehand and will train one model for each style due to obvious differences between these styles. Like running or rowing, swimming is a cyclic motion sequence consisting of self–repeating motions.

However, there is an essential difference between the major swimming styles: the arm movement in freestyle and backstroke swimming alternates from side to side; while one arm is pulling under water, the other one is recovering above the water line. Hence every pose of the left body half occurs half a cycle later on the right body half of the swimmer (see Figure 1). Throughout this work we will call these swimming styles *anti–symmetrical*. In contrast to that, in a butterfly and breaststroke cycle both arms and legs of a swimmer have semantically the same posture at every time. We therefore denote them as *symmetrical* swimming styles.

2.2.1 Data set

We created our dataset from videos of swimmers in a swimming channel. The footage was taken at the IAT[15] in Leipzig, Germany, and covers all swimming styles with male and female adult and age group swimmers at different stream velocities. We only collected footage of swimmers observed from a side view. Using footage at various stream velocities is beneficial due to covering a wider range of the variations that occur in swimming. For instance, faster streams produce more bubbles in the water, which results in noisier images and thus helps the learning approach to better tolerate noise. In addition, the swimmer is forced to swim at different speeds, which in turn will result in changes of how a stroke in executed as well as the level of motion noise generated by the extremities. Noise in an image may substantially blur the silhouette of a swimmer, while at the same time result in more edge information (from the churned water) when transformed to feature space. We assume that the usage of noisier images in training as well as a larger variation in posture and body tense due to different stream velocities will reflect on the detector's robustness against noise.

All videos have a frame rate of 50 frames per second, and we manually annotated every swimmer with a corresponding bounding box. The final training set consists of the images of ten cycles per swimming style which results in about 3000 positive training images. We have added the same amount of negative images depicting empty swimming channels, and another 4000 annotated positive and negative images for testing.

2.2.2 Data set partition

As described in the previous section 2.1, we want to detect characteristic poses within a cycle of each swimming style. Therefore, each pose model has to be trained on these poses. Hence, we partition the training images for each style according to different criteria, and for each partition a pose model specific to this partition will be trained.

It might at first seem appropriate to train pose models for all possible poses within a cycle of a certain swimming style. This, however, is impractical because it would result in a very large number of pose models making the approach computationally infeasible. Moreover, each pose model should be invariant to changes within the key pose, because different swimmers will most likely not feature identical poses. Therefore, selecting only a single significant key pose for each swimming style is also discouraged. Under these aspects we partition a cycle into several intervals and all consecutive images within an interval contribute to the same pose model and therefore resemble the same key pose. Since the stroke rate measurements rely completely on the performance of the pose models, it is crucial that these are trained properly. Therefore, care has to be taken to quantize the cycles in an appropriate manner.

We first quantize all training cycles uniformly into eight consecutive subsets, following our intuition that too many and equally too few pose models will decrease performance. The resulting subsets consist of 75 to 150 images each, depending on the velocity of the swimmer. Each subset contains poses that all contribute to the same key pose that we later aim to detect by training a pose model. Figure 1 shows example images of all eight intervals.

A closer inspection of the separated cycles and the trained models shows that such a simple quantization might not be suitable for all swimming styles, and that one has to carefully partition the cycles with respect to the swimming style. In order to evaluate the performance according to different training set quantizations, we therefore introduce additional specific partitions for each swimming style.

Anti–symmetrical swimming styles: Since it is hard to distinguish the left and right body half if observed from the side, we additionally quantize these cycles into only four uniformly distributed subsets that cover only half of a cycle. That is, each subset contains similar poses of the left and right body halves.

Symmetrical swimming styles: For symmetrical swimming styles we observe that, especially for slower swimmers, phases within the cycle often are longer than one eighth of a cycle, which results in more or less similar poses for two consecutive subsets. This can be seen in the right hand side of Figure 1. For breaststroke we thus additionally partition the training images into only four subsets. We adhere to the actual semantic phases and create subsets for the outsweep, insweep, recovery and diving. Since butterfly doesn't feature four semantically different phases, we use a uniform quantization here. Later experiments will show that, especially for slow swimmers swimming butterfly, most of the pose models failed to detect the characteristic poses of such a four–pose model. Therefore, only for butterfly, we use an additional partitioning into only two subsets. One subset contains a very specific pose and therefore a very small part of the cycles, whereas the other contains all remaining poses.

Table 1 contains an overview over all training set partitions used depending on the swimming style.

2.3 Swimmer detection and pose estimation

Swimmers are detected in images by applying the trained mixture models at all scales and positions of the image. The swimmer is detected at the scale and position where the combined confidence of all pose models is above a model–specific threshold. Multiple detections of the same swimmer are eliminated using non–maximum suppression.[4] When we evaluate the scores of the submodels of an n–pose model at the detected location, we can

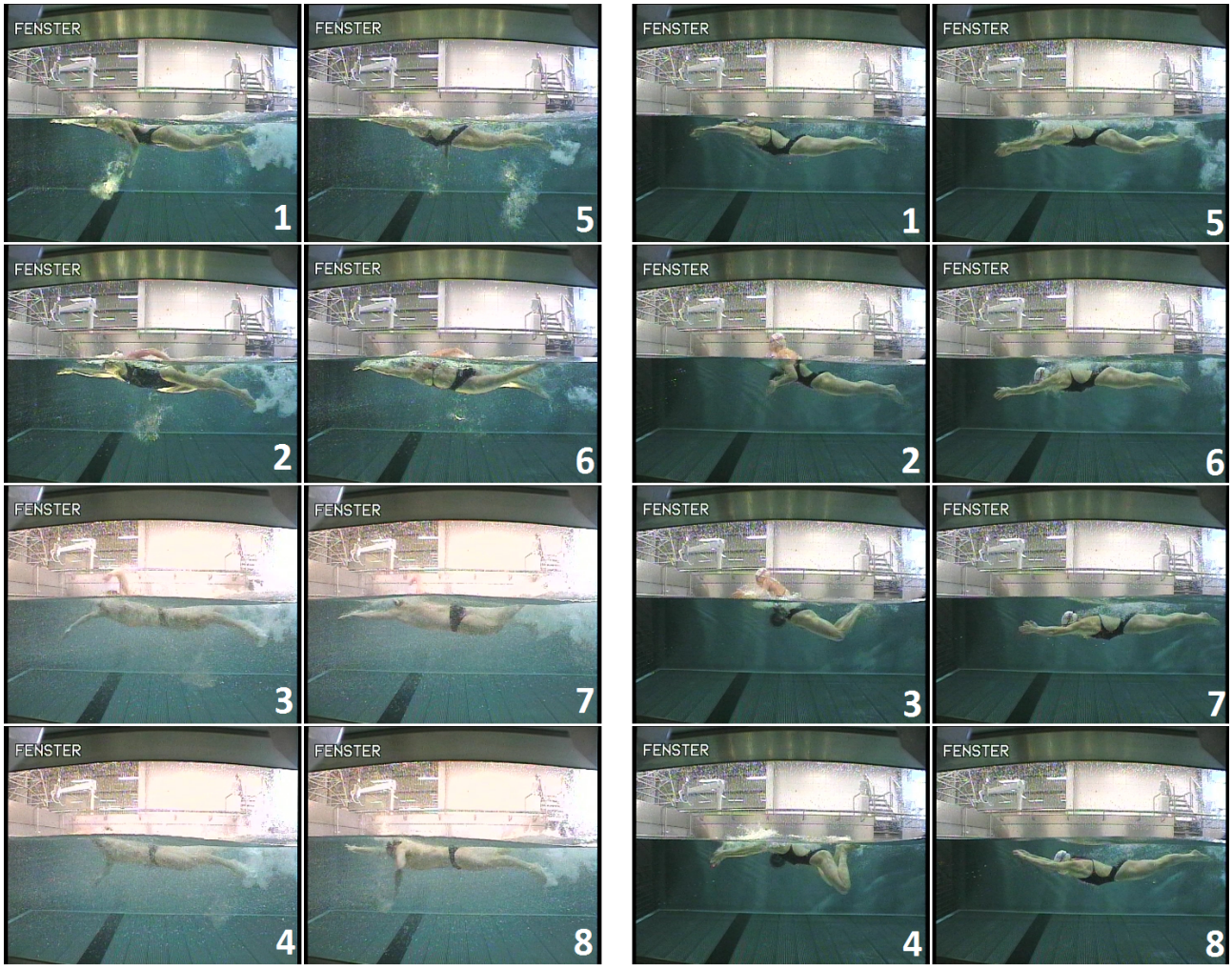


Figure 1. A freestyle (left side) and breaststroke (right side) cycle. Each image depicts a representative for a single subset of poses. The first four freestyle swimming poses show the cycle of the left body half, while pose 5-8 are their counterparts. The division of the breaststroke cycle shows a long recovery phase (1,6,7,8). (Source of original images: IAT, Leipzig[15])

assume that a pose model i has higher confidence values than all other $n-1$ pose models if the swimmer features a pose where pose model i was trained on. Hence, for each pose model we record all frames of a video where it scored highest. This yields a sequence of unit impulses that in turn can be interpreted as a signal over time resembling the occurrence of a key pose. The left hand side of Figure 2 shows an example of such a *pose signal*. As can be seen, there are incorrect detections within a video where the swimmer is detected correctly, but not by the pose model that would suit the actual pose but by another one. Such detections often occur in transition areas where two subsequent pose models both compute high scores. We eliminate this noise by convolving each signal with a zero-mean Gaussian

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{1}$$

where the standard deviation was set empirically to $\sigma = 6$. The result is a regularly oscillating function whose local maxima yield the stroke rate of a swimmer provided that the pose models are well trained.

swimming style	four-pose model	eight-pose model	two-pose model
freestyle	uniformly, joined	uniformly	–
backstroke	uniformly, joined	uniformly	–
butterfly	uniformly	uniformly	semantically
breaststroke	semantically	uniformly	–

Table 1. An overview over all trained models, listing the kind of training set quantization for each model and each swimming style. Uniformly quantized training sets are created by simply dividing a cycle into equally sized subsets. While semantically quantized sets are bound to actual phases of a swimming cycle, specifically quantized sets are manually sorted into groups of specifically chosen key poses. The training sets of the four-pose freestyle and backstroke models are joined from the uniformly quantized eight-pose sets, taking similar poses of left and right body halves into account.

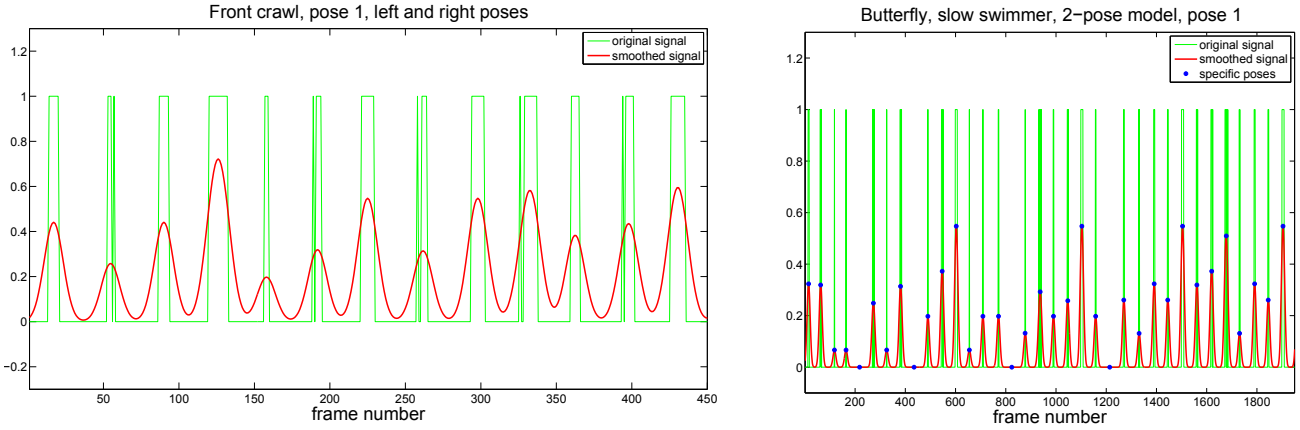


Figure 2. Left: A raw pose signal (green graph) for a specific pose i. The signal is 1 if submodel i had the highest score and 0 otherwise. To deal with noise the signal is smoothed (red graph). Right: Example signal showing the result of interpolating missing peaks. The original red signal features several gaps, where the pose wasn't detected. Blue dots depict frames where peaks are detected after the interpolation.

2.4 Pose signal refinement

The smoothed raw pose signal may still exhibit two types of errors: accidental spurious false detections due to signal noise and gaps that result from missed detections.

2.4.1 Eliminating noise

In the ideal case each local maximum of the pose signal should correspond to a frame depicting a pose the pose model was trained on. For very noisy signals Gaussian smoothing does not eliminate all false detections which leads to multiple peaks around the true location. We thus apply non–maximum suppression[4] to the smoothed signal. The distance between two consecutive peaks resembles the number of frames between the occurrence of two detected key poses. We define the signal Δ of distances between two consecutive detections by $\Delta = \{\Delta_1, ..., \Delta_{i-1}, \Delta_i, \Delta_{i+1}, ..., \Delta_n\}$. We assume that more than half of these distances are not affected by noise and compute the median distance to obtain an estimate of the actual stroke rate. We then assume that all distances smaller than 0.7 times this median distance are the result of noise and eliminate the corresponding peaks. Since swimmers might change their frequency throughout the video, we apply this form of non–maximum suppression to small windows of 15 seconds.

2.4.2 Eliminating gaps

The previous refinement step deletes all spurious detections (i.e., all false alarms) of a given pose. However, it does not fill in sporadically missed detections. The right hand side of Figure 2 shows four instances of misses around frame number 220, 440, 820 and 1220. These misses can be identified and eliminated again by analyzing the signal Δ. We run a sliding window of size 3 over this sequence in order to check for gaps. A single miss is

identified at position i if $1.8 \cdot \Delta_{i-1} < \Delta_i < 2.2 \cdot \Delta_{i-1}$ and $1.8 \cdot \Delta_{i+1} < \Delta_i < 2.2 \cdot \Delta_{i+1}$. If a miss was identified, an additional pose detection is inserted into the middle of the interval as shown in Figure 2 by the blue dots at the location of the original misses.

3. EXPERIMENTAL RESULTS

In this section we evaluate and discuss the performance of the swimmer detection and pose estimation as well as of the temporally continuously derived stroke rate. For the detection task we empirically analyze how performance depends on the preparation of the training data, while for the pose estimation we also investigate for each stroke which poses throughout the stroke cycle are temporally and visually distinctive and can thus be identified reliably.

3.1 Swimmer detection performance

3.1.1 Test set and performance measures

We constructed four test sets, one for each swimming style. Each test set consisted of 500 negative images depicting an empty swimming channel and 500 positive images with a single swimmer swimming the respective swim stroke in the swimming channel. Each swimmer was annotated by his/her bounding box. In contrast to the training set which included only male and female adult swimmers at different water flow velocities, we also added images of age group (i.e., teenage) swimmers to the test set in order to check the robustness of the detectors.

We use the same performance measures as the VOC challenge for object detection.[16] Therefore a detection is considered correct if the overlap a_0 between the detection area and the ground truth bounding box exceeds 50%. The overlap[16] is computed by

$$a_0 = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \tag{2}$$

whereas B_p is the predicted and B_{gt} the ground truth bounding box. Recall and precision of a swimmer model are then given by

$$recall = \frac{\#correct\_detections}{\#positive\_examples} \tag{3}$$

$$precision = \frac{\#correct\_detections}{\#detections}. \tag{4}$$

Recall measures how many swimmers were detected correctly, whereas *precision* measures how many detections are indeed swimmers. A complete recall/precision curve for each swimming style model is generated by associating with each detection a confidence value and varying the threshold for declaring a detection. *Average precision* (AP) summarizes the recall/precision curve into a single score by computing the integral from 0 to 1 over this curve (see Figure 3).

3.1.2 Results

Figure 3 shows the recall vs. precision curves of the four-pose and eight-pose mixture models for each swimming style (see also Table 1). For backstroke and freestyle – the two anti-symmetrical swim strokes where each body half performs the same cyclic motion but offset by half a cycle – the four-pose models for half a cycle are best (effectively eight-pose model for a full cycle), while for breaststroke and butterfly – the two symmetrical swim strokes – the eight-pose models are best. This behavior is in line with what we expect for these swimming styles. While half the models have an average precision higher than 0.89, the models for backstroke and butterfly have a lower average precision. As we will discover below, this does not necessarily imply that the swimmer wasn't found in the images or that the models scored a lot of false detections. Visually examining the resulting detections showed that the swimmer was found in almost all images.

While the sizes and positions of the bounding boxes are evaluated through the average precision of a model, the positions of the parts are not. Hence we discuss next a few sample detections of freestyle and butterfly swimmers.

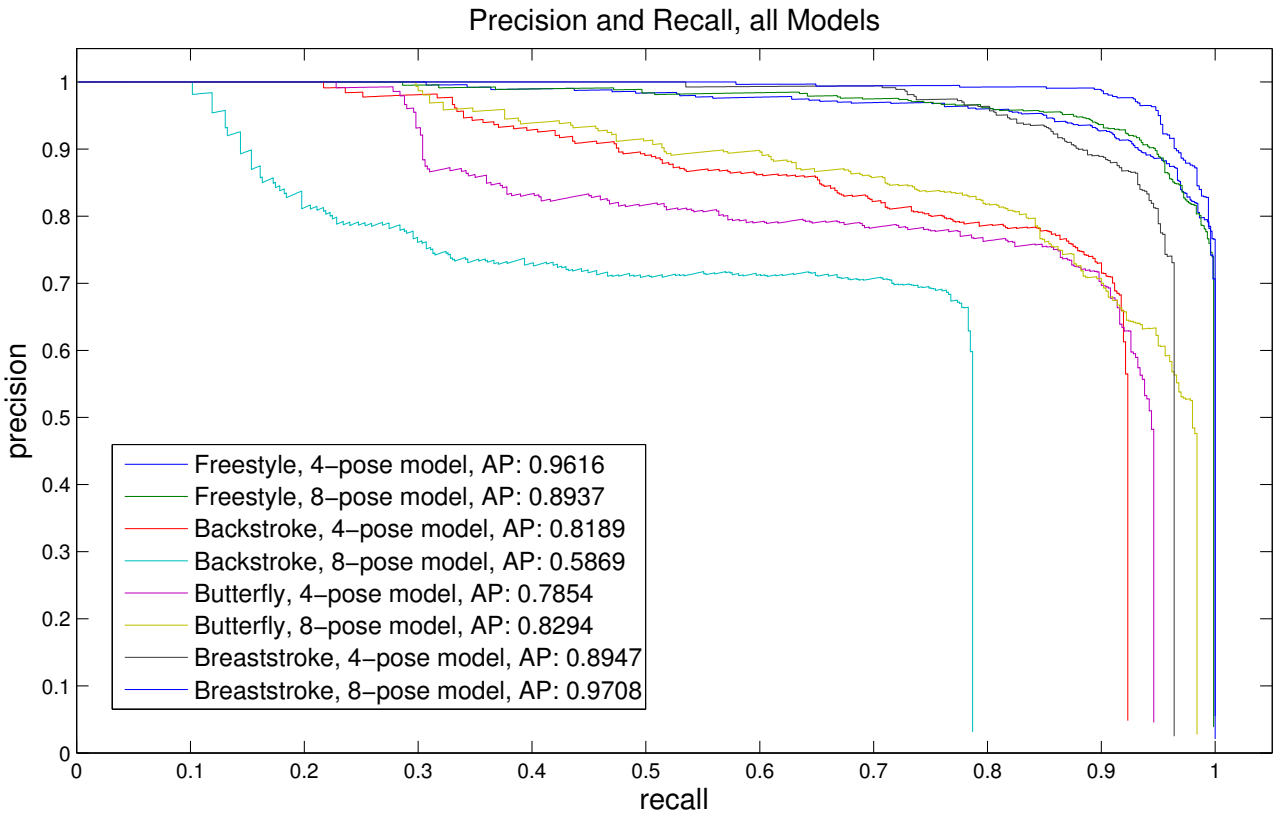


Figure 3. Precision vs. recall and average precision score (AP) for all eight models.

One discovery we made was that in almost every detection of any swimming style, the learned model positioned at least one part on the hip of the swimmer. We assume that the gradient information under this part of the swimmer is very dominant, so the training algorithm places and trains one or more filters in this area. As the hip is found in most detections we also assume that this part of the swimmer often has lower deformation costs and thereby a higher score compared to other parts, even high enough to be a dominant factor for the whole detection score of a swimmer.

Our initial intuition was that the automatically determined part filters should correspond to real parts of a person like the head or the arms. However, it can be seen in the sample detections of Figure 4 that the model parts very often are not confined to a part of the swimmer, but also include water below and/or above a part. This may be intuitive if a part covers an area of the image with a lot of noise (e.g. bubbles) generated by the swimmer. However, Figure 4-2 also shows that the left-bottom part covers an area without any part of the swimmer. We assume that the corresponding pose model learned the background as context information, i.e., that for the pose the arm is not allowed to be in this area, while for other poses of the stroke it should.

A problem we observed with some detections can be seen in Figure 4-5. The bounding boxes are far too big for the swimmer. This is due to the large amount of water bubbles produced by the strong leg kicks of the swimmer. These bubbles appear as co-occurring noise with the swimmer in the feature pyramid, and the detector obviously interprets this noise as a part of the swimmer. We found that such artifacts appear often with swimmers whose posture is stretched or who produce waves/bubbles in the channel. We suspect that the water line is a very dominant feature and good orientation for a detector.

An unwanted effect of over-sized bounding boxes is that the overlap a_0 between the predicted bounding box and the ground truth bounding box does not exceed 50%, thus counting it as an incorrect detection although

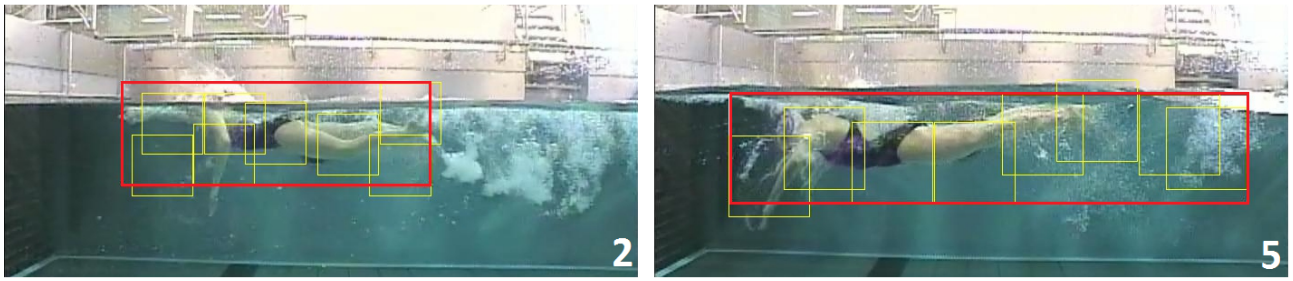


Figure 4. Some sample detections of four-pose models. The red bounding box depicts the detection of the root filter while the yellow boxes depict subwindows of the highest scoring parts. (Source of video frames: IAT, Leipzig[15])

the swimmer is depicted in the window of the predicted box. We assume this being an important reason for the worse average precision values of some models.

There are various factors that influence the performance of a pose model: flow velocity, different kinds of image noise, illumination, gender, posture and body tense. We found that the age of a swimmer is another important factor that influences the detection and pose estimation performance. Age group swimmers do have a different physique compared to older swimmers, which may result in a considerably different execution of a swimming stroke.

3.2 Pose estimation performance

3.2.1 Test set and performance measure

As we want to analyze how performance of the pose estimation depends on the stroke rate and flow velocity, we recorded additional videos of swimmers in the swimming channel under controlled water flow velocities: two swimmers of age twelve and fourteen (different from the adults in our training set) were recorded with a constant slow stream velocity for all four swimming styles and a constant fast stream velocity for freestyle, butterfly and backstroke. Additionally we increased slowly the flow velocity during two takes for freestyle and breaststroke, in order to see how our system adapts to changes in stroke rate. The lengths of the videos vary from 20 to 60 seconds, and the videos were recorded at 50 fps.

As manually annotating each video frame with the ground truth pose of the swimmer is prohibitively expensive, we decided to measure the pose estimation quality indirectly by means of the stroke rate. This significantly releases the annotation burden as only a single key pose has to be identified per stroke cycle in all nine test videos. For instance, the key pose for backstroke was where one arm pointed upright to the ceiling. As this is also the moment of highest arm speed, it could be easily determined in time.

The stroke rate $f_{p_i,j}$ in strokes per minute derived from a pose model p_i at the j-th detected occurrence and the stroke rate $f_{gt,j}$ derived at the j-th annotated ground truth key pose is given by

$$f_{p_i,j} = \frac{60}{\frac{1}{v_{in}} \cdot \left(\frac{\delta_{p_i,j} + \delta_{p_i,j-1} + \delta_{p_i,j-2}}{3} \right)} = \frac{60 \cdot 3 \cdot v_{in}}{\delta_{p_i,j} + \delta_{p_i,j-1} + \delta_{p_i,j-2}} \tag{5}$$

where v_{in} is the frame rate of the test video and $\delta_{p_i,j}$ the frame distance between the occurrences j and $j-1$ of two consecutive key poses. Thus the stroke rate is always the average over the last three cycles. The averaged frame distance divided by the frame rate of the video computes the time per stroke in seconds, of which the reciprocal multiplied by 60 gives the stroke rate as the number of strokes per minute.

Given that we only have the ground truth stroke rate at a single position per stroke cycle, we compute the fit of the actually detected stroke rate sequence $f_{p_i} = \{f_{p_i,j}\}$ with the ground truth stroke rate sequence $f_{gt} = \{f_{gt,j}\}$ of a test video by means of the divergence in the expected stroke rates $d_E = |E(f_{p_i}) - E(f_{gt})|$ and stroke rate variances $d_V = |Var(f_{p_i}) - Var(f_{gt})|$. The quality q_{p_i} of the pose detection with pose model p_i is then measured by

$$q_{p_i} = \exp\left(-d_E \cdot d_V\right). \tag{6}$$

The quality equals 1 if a pose model returns the same rates as the ground truth and decreases the more inaccurate the model gets. We assume that models with a measured quality value smaller than 0.02 cannot reproduce the rates well enough and therefore do not use them for rate measurements.

3.2.2 Results for butterfly and breaststroke

Figure 5 depicts the measured quality for the butterfly and breaststroke models: left the four-pose models and right the eight-pose models. It can be seen that both the four-pose model and the eight-pose model for breaststroke have a very high quality. Only pose models two, six and seven of the 8-pose model do not capture the frequency very well. We found that the training data of model six and seven is very similar to the training data of pose model number five due to long diving phases of the swimmer in this part of the stroke cycle (see Figure 1, right most image column). This pecularity of breaststroke is the reason why the four-pose breaststroke model is the only four-pose model which was not trained on sets of temporally uniformly quantized stroke cycles. We combined those images of a cycle into the same set that depict an actual phase of breaststroke swimming, i.e. outsweep, insweep, recovery and diving. This may be the reason why the breaststroke four-pose model outperforms every other model we trained with uniformly divided stroke cycles.

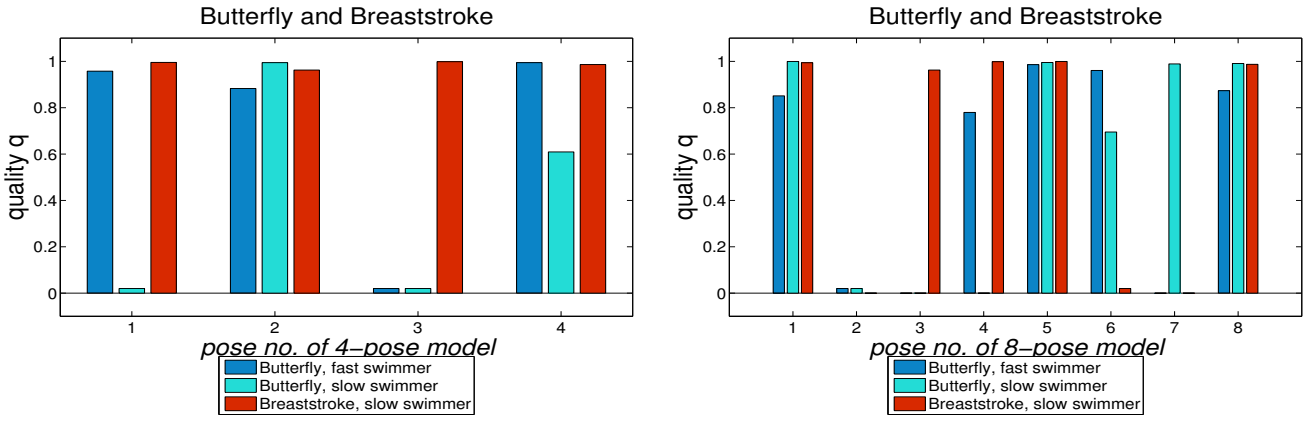


Figure 5. Measured quality of the four-pose (left) and eight-pose (right) butterfly and breaststroke models.

While the pose detection performance was high for most poses of fast butterfly swimmers, fewer pose models scored high for slow swimmers, indicating again that defining poses based on temporally uniformly spaced time intervalls throughout the cycles needs to be reconsidered also for butterfly. To improve the performance of the butterfly detector we trained another swimmer model consisting of only two poses. The first pose model was trained on a very specific, only shortly lasting, and manually selected pose, while the second pose model was trained on the rest of the cycle. The quality measure of this model is greater than $q > 0.95$ for slow swimmers, for fast swimmers though it fails to detect the stroke rate.

3.2.3 Results for freestyle and backstroke

An important property of freestyle and backstroke is that they are anti–symmetrical, i.e., the motion of arms and legs of both body sides are offset by half of a cycle. A direct result of this characteristic can be seen in Figure 6 for the eight-pose models: apart from the first freestyle and the sixth backstroke pose model, none of the eight-pose submodels, neither backstroke nor freestyle, captures the frequency of any swimmer correctly as aubmodel n and $n+4$ for $n \in \{1, 2, 3, 4\}$ describes visually the same pose from the side view. Thus the resulting pose signals are so noisy that there is no suitable information left after post-processing.

Taking this into account, the four pose models for both freestyle and backstroke were trained with the training sets for left and right facing poses combined in order to capture the half of a cycle. It can be seen in Figure 6 that these models basically capture the frequency very well. They effectively represent eight-pose models for a complete cycle. In conclusion of Figures 5 and 6, we will use the four-pose model versions for all swimming styles to depict the stroke frequencies in the next subsection.

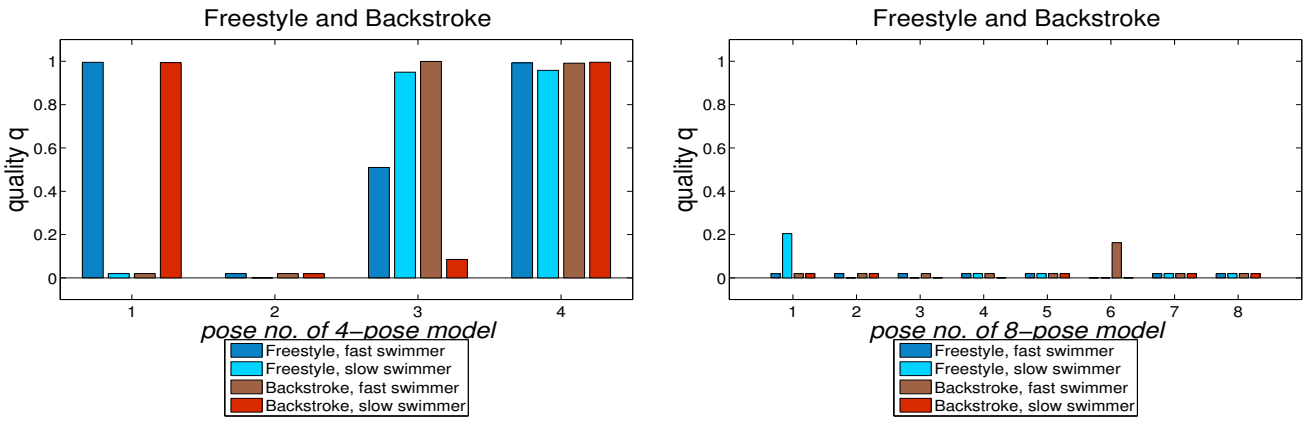


Figure 6. Measured quality of the four-pose and eight-pose freestyle and backstroke models.

3.3 Stroke rate

An important goal of this work is to automatically measure the stroke rate of a swimmer over time in a swimming channel. We trained pose models and evaluated their quality with respect to this task in the previous section. Here we want to discuss sample traces in order to give insights into the actual shape of the frequency curves using the four-pose models.

Figure 7 depicts the stroke rate graphs of every major swimming style, excluding all pose models with a quality less than 0.02. The ground truth, which was obtained manually by labeling distinctive poses in each cycle of a test video, can be reproduced by at least one fixed (over all test videos) pose model of every swimmer style. As can be seen, in almost all cases we are able to measure the stroke rate reliably. In cases we have divergence one could even argue from the graph, that it would be more plausible to assume that the ground truth was incorrectly determined (see e.g. top left in Figure 7) as it is sometimes even difficult for a human to exactly determine the ground truth.

In order to capture an increase in a swimmer's stroke rate we recorded two videos (freestyle and breaststroke) where the stream velocity was increased slowly (see Figure 8). It can be seen that most of the pose models capture the increase well.

4. CONCLUSION

In this work we have shown that we can robustly detect swimmers and their poses to estimate their stroke rates with high precision from videos showing swimmers from the side. Key poses have been defined as intervals of swimming cycles that are easy to detect by an object detection model. We have shown that it is crucial to prepare the training data for each swimming style in an appropriate manner. The approach was evaluated under different aspects on a data set containing a wide variety of swimmers such as male/female adult and age group swimmers at different swimming speeds.

REFERENCES

[1] Craig, A. and Pendergast, D., "Relationships of stroke rate, distance per stroke, and velocity in competitive swimming," *Med Sci Sports* **11**(3), 278–283 (1979).

[2] Seifert, L., Chollet, D., and Rouard, A., "Swimming constraints and arm coordination," *Human movement science* **26**(1), 68–86 (2007).

[3] Seifert, L., Boulesteix, L., Chollet, D., and Vilas-Boas, J., "Differences in spatial-temporal parameters and arm-leg coordination in butterfly stroke as a function of race pace, skill and gender," *Human movement science* **27**(1), 96–111 (2008).

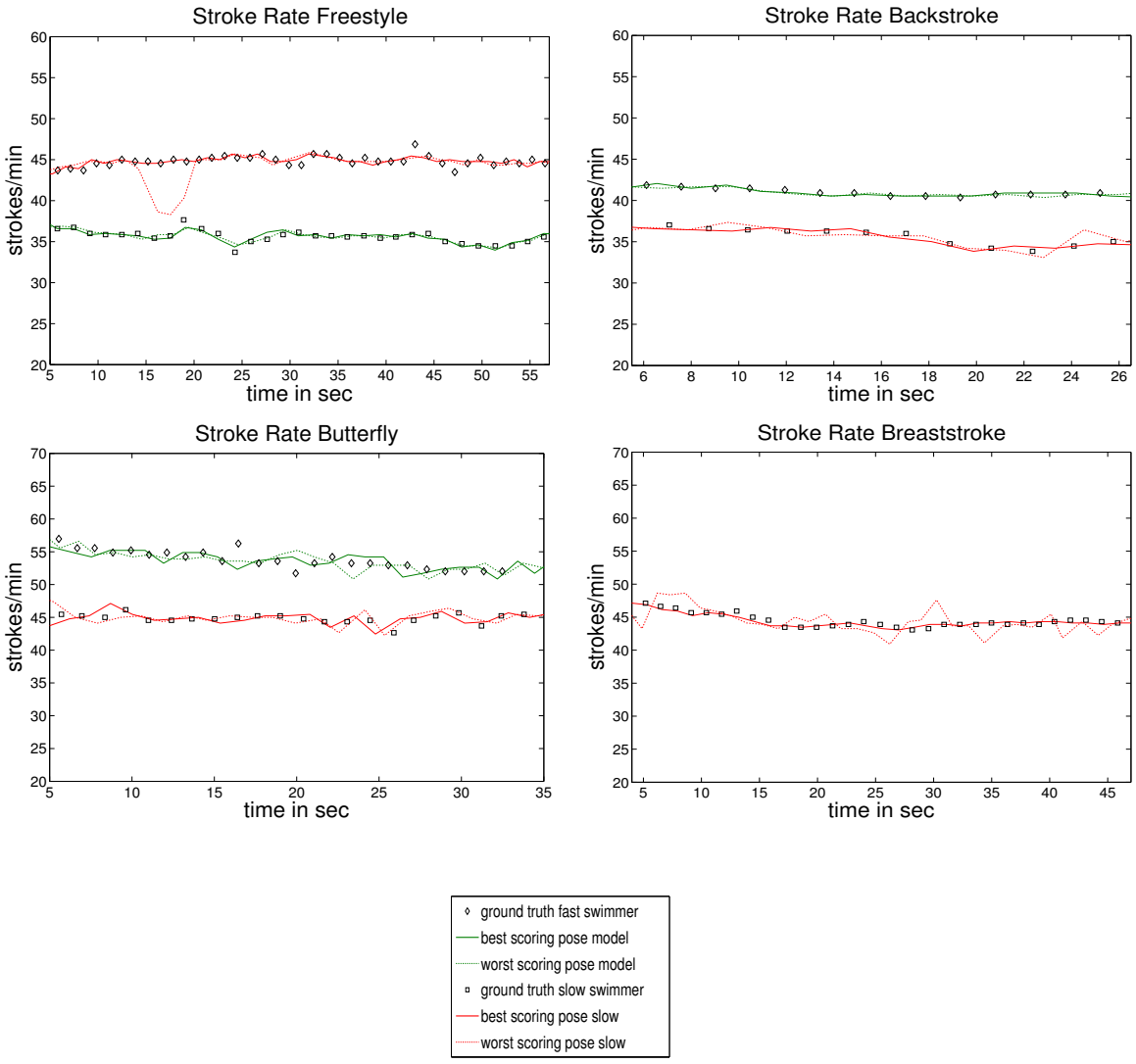


Figure 7. Stroke rate traces for all four major swimming styles and different stream velocities. Black diamonds and squares depict the ground truth. It can be seen that both the best scoring pose models (straight lines) and the worst scoring pose models (dotted lines) reproduce the stroke rate very well. Pose models with a quality measure smaller than 0.02 are unusable and therefore rejected.

[4] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D., "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **32**(9), 1627 –1645 (2010).

[5] Beetz, M., Hoyningen-Huene, N., Bandouch, J., Kirchlechner, B., Gedikli, S., and Maldonado, A., "Camera-based observation of football games for analyzing multi-agent activities," in [*Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*], 42–49, ACM (2006).

[6] Eng, H.-L., Wang, J., Kam, A. H., and Yau, W.-Y., "Novel region-based modeling for human detection within highly dynamic aquatic environment," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* **2**, 390–397 (2004).

[7] Eng, H.-L., Wang, J., Wah, A., and Yau, W.-Y., "Robust human detection within a highly dynamic aquatic environment in real time," *Image Processing, IEEE Transactions on* **15**, 1583 –1600 (june 2006).

[8] Eng, H., Toh, K., Yau, W., and Wang, J., "Dews: A live visual surveillance system for early drowning

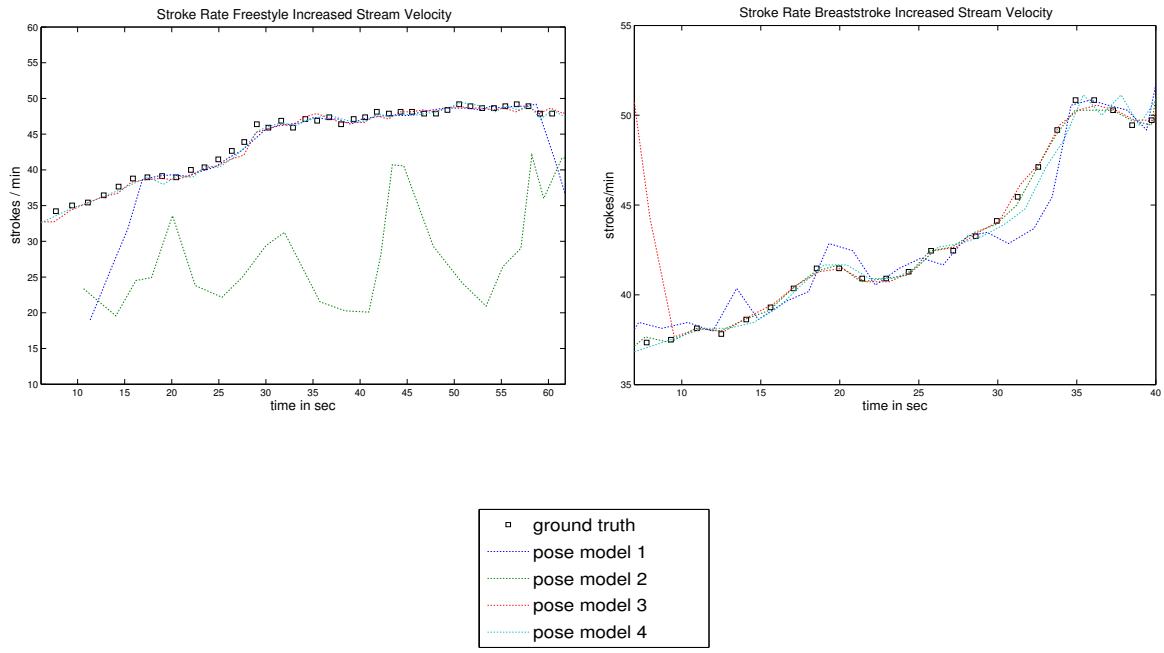


Figure 8. Stroke rate traces for freestyle and breaststroke. In both test videos, the stream velocity has been increased slowly. Most depicted pose models (dotted lines) capture the increase very well.

detection at pool," *Circuits and Systems for Video Technology, IEEE Transactions on* **18**(2), 196–210 (2008).

[9] Kam, A., Lu, W., and Yau, W., "A video-based drowning detection system," in [*Proceedings of the 7th European Conference on Computer Vision-Part IV*], 297–311, Springer-Verlag (2002).

[10] Meniere, J., "System for monitoring a swimming pool to prevent drowning accidents," (Oct. 17 2000). US Patent 6,133,838.

[11] Bachlin, M., Forster, K., Schumm, J., Breu, D., Germann, J., and Troster, G., "An automatic parameter extraction method for the 7x50m stroke efficiency test," in [*Pervasive Computing and Applications, 2008. ICPCA 2008. Third International Conference on*], **1**, 442–447, IEEE (2008).

[12] Davey, N., Anderson, M., and James, D., "An accelerometer-based system for elite athlete swimming performance analysis," in [*Proceedings of SPIE*], **5649**, 409 (2005).

[13] Ries, C. X. and Lienhart, R., "Automatic pose initialization of swimmers in videos," *Visual Information Processing and Communication* **7543**(1), 75430J, SPIE (2010).

[14] Greif, T. and Lienhart, R., "A kinematic model for bayesian tracking of cyclic human motion," *Visual Information Processing and Communication* **7543**(1), 75430K, SPIE (2010).

[15] "Institut für Angewandte Trainingswissenschaft, Marschnerstrasse 29, 04109 Leipzig, Germany."

[16] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A., "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results." http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.