**Swim Stroke Analytics: Front Crawl Pulling Pose Classification**

Hossein Fani, Amin Mirlohi

Faculty of Computer Science, University of New Brunswick, NB, Canada

{hossein.fani, sam.mirlohi}@gmail.com

Hawre Hosseini

Department of Electrical & Computer Engineering, Ryerson University, ON, Canada

Rainer Herpers

Bonn-Rhein-Sieg University of Applied Sciences, Sankt Augustin, Germany

[rainer.herpers@h-brs.de](mailto:rainer.herpers@h-brs.de)

**Abstract**

In this work, we automatically distinguish the efficient *high elbow* pose from *dropping* one in pulling phase of front crawl stroke from two dimensional front view swimming videos. We predict the pull's efficiency given arms key positions and angles with respect to the water surface. We evaluate our approach over a dataset of swimmers with different level of expertise and physiological characteristics through classification metrics. Our results show that our proposed method performs on par with human judgments.

1. **Introduction**

Majority of people, 65\% to be precise, are visual learners who need to see what they are learning [[Reaching the Visual Learner: Teaching Property Through Art](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=587201)]. In sport, in particular, athletes attain their maximum potential not only by listening to the coaches but also by reviewing their own workouts in practice. In this respect and in order to improve understanding, training, and performance, computer vision algorithms find huge potential in automating video analysis of athletes and providing immediate feedback on poses' efficiency to the coaches as well as athletes themselves. For instance, action recognition provides swim coaches and swimmers with automatic performance assessment through statistics such as stroke rate, based on visual perception of periodic body kinematic. Our ambition is to employ computer vision algorithms to analyze swim stroke of a novice swimmer in compare to the professionals', captured by her own off-the-shelf camera, for training purposes. This task is challenging due to the aquatic environment (e.g. water splashes and reflections) and missing information such as depth dimension in the low quality amatuar-recorded videos.

We focus our study to front crawl swimming style and predict the efficiency of the arm stroke under water, called pulling, from the front view angle as seen in Figure 1. Pulling’s objective is to generate moving drive forward through the water. Efficient pulling is initiated by maintaining the elbow close to the surface, so-called *high elbow* as shown in Figure 1 (left), followed by dropping the forearm below the elbow and a sweeping motion of the upper arm. This is in contrast with inefficient *dropping elbow*, as shown in Figure 1 (right), where the whole arm is down deep to the water. In this work, our goal is to predict the quality of pulling phase as of being high elbow versus dropping elbow given an underwater video of front crawl strokes.

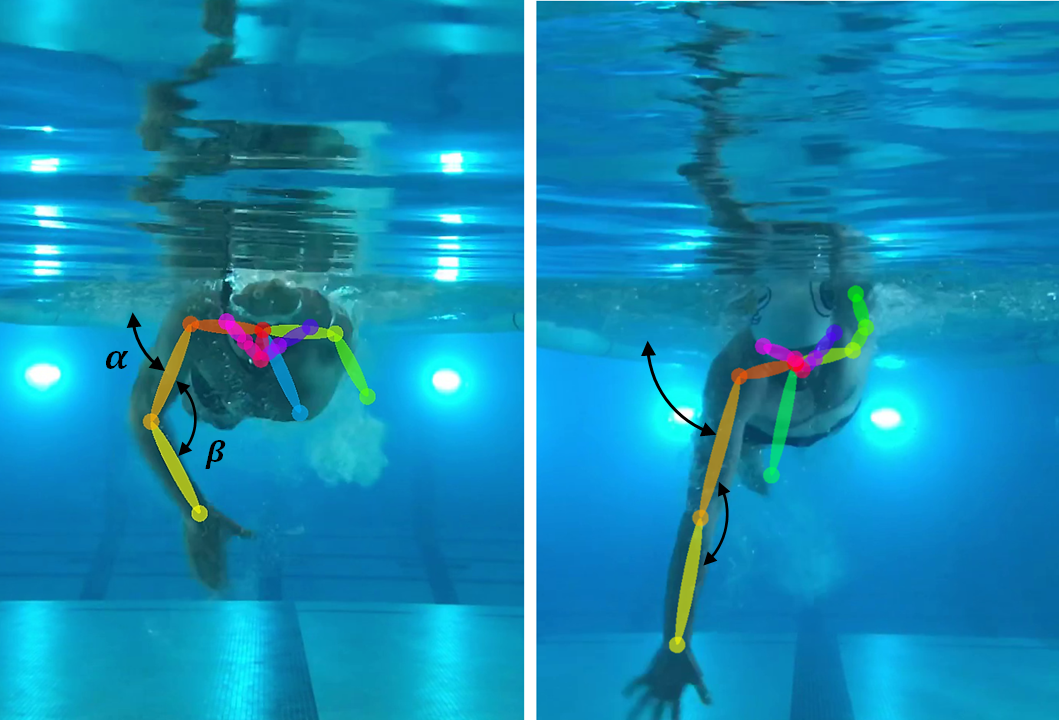


Figure 1. High elbow (left) vs. dropping elbow (right) in pulling phase of front crawl stroke.

Action recognition in sports, swimming in particular, have been already investigated in the literature [[High Performance Moves Recognition and Sequence Segmentation Based on Key Poses Filtering](http://homepages.dcc.ufmg.br/~erickson/publications/vicente_wacv2016.pdf), [Combat sports analytics: Boxing punch classification using overhead depthimagery](https://eprints.qut.edu.au/93143/1/ICIP2015_2.pdf), [Understanding and analyzing a large collection of archived swimming videos](http://ieeexplore.ieee.org/document/6836037/), [Continuous Video to Simple Signals for Swimming Stroke Detection with Convolutional Neural Networks](http://openaccess.thecvf.com/content_cvpr_2017_workshops/w2/papers/Victor_Continuous_Video_to_CVPR_2017_paper.pdf), [Pose Estimation for Deriving Kinematic Parameters of Competitive Swimmers](http://www.multimedia-computing.de/mediawiki//images/f/f8/Pose_estimation_for_kinematics_with_copyright_notice.pdf)]. Sha et al. [[Understanding and analyzing a large collection of archived swimming videos](http://ieeexplore.ieee.org/document/6836037/)] and Victor et al. [[Continuous Video to Simple Signals for Swimming Stroke Detection with Convolutional Neural Networks](http://openaccess.thecvf.com/content_cvpr_2017_workshops/w2/papers/Victor_Continuous_Video_to_CVPR_2017_paper.pdf)] estimate the front crawl stroke rate. While the approaches are robust for the task of stroke rate estimation in natural videos, i.e., broadcast videos taken at races, they are not extensible to obtain performance indicators regarding to efficiency of body movements in swim strokes such as pulling pose. Zecha et al. [[Pose Estimation for Deriving Kinematic Parameters of Competitive Swimmers](http://www.multimedia-computing.de/mediawiki//images/f/f8/Pose_estimation_for_kinematics_with_copyright_notice.pdf)], however, constrained their work to a lab environment, i.e., swimming glass channel, where the swimmer’s whole body, under and above water, is visible from side view at all times such that almost all kinematic parameters are able to be quantitatively extracted. Our work is inspired by Zecha et al. but distinguishes itself as we are going to analyse the swimmer kinematic in a real aquatic setting, i.e., swimming pool, from her own amataur-recorded video with no extra prior knowledge about the environment. Needless to say that the swim glass channel is not accessible to all type of swimmers and the side view recording in a real swim pool needs the camera to constantly follow the swimmer alongside the pool which adds not only complexity (active vision), but also requires sophisticated apparatus. In our work, however, we need front view recording from a fixed position which can be easily setup.

We detect the swimmer body parts using the state-of-the-art pose detection method OpenPose [[Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields](https://arxiv.org/abs/1611.08050)]. The choice of OpenPose is motivated for its functionality on image or video taken from webcam and IP cameras. This provides huge benefit with compare to the skeletal tracking capability of Microsoft Kinect or the likes which depends on depth information, i.e. three dimensional camera. We then measure the angles between upper arm, forearm, and water surface. Given the arm joints position and the respective angles, we train a classifier on a manually labeled dataset of swimmers of different expertise and physiological properties[[1]](#footnote-0). Our results show that we are able to successfully infer that a swimmer is doing either high or dropping elbow in pulling phase of her front crawl stroke with a precision which is on par with the human judgment. Our work imply a general process to analyze other sports in which body movements plays an important role in performance achievement like cycling, sprinting, and other track and field athletics, to name a few.

The main contribution of this paper is not a new pose estimation algorithm, but rather a novel, yet effective application of existing pose estimation method, i.e., OpenPose, in swim stroke analysis. Hence our concrete contributions are:

1. We propose a novel application of computer vision in swimming performance feedback taking advantage of existing infrastructure.
2. Our framework performs on par with a human annotator on amateur-recorded video of swimmers in a real aquatic environment of swimming pools, allowing performance assessments accessible for almost all end users.
3. We build a manually labeled dataset of swimmers with different expertise, i.e., beginner, novice, and professionals, from wide variety of physiological properties, i.e., gender, age and size, in order to train a model generalized enough to all type of swimmers.

The rest of the paper is organized as follows: we first present the related works in Section 2, then we continue with the problem definition and our proposed approach details in Section 3. The experimental setup and evaluation is described in Section 4, followed by a discussion. Finally, Section 5 concludes the paper.

1. **Related Work**

Researchers have already investigated action recognition in swimming analysis in the literature. Sha et al. [[Swimmer Localization from a Moving Camera](http://ieeexplore.ieee.org/document/6691533/)] propose a technique to track the swimmer and her location automatically in the aquatic environment. The authors distinguish different swim-states as start, dive, underwater, swimming, turn, and end. Then, they extract state-specific features to localize and track the swimmer. Once the swimmer's position has been determined, they estimate the stroke rate in their next work by focusing on body parts (the elbow and the back) and the angle between the parts in front crawl [[Understanding and analyzing a large collection of archived swimming videos](http://ieeexplore.ieee.org/document/6836037/)]. Tong et al. [[Local Motion Analysis and Its Application in Video based Swimming Style Recognition](http://ieeexplore.ieee.org/document/1699438/)] have done swim stroke detection as an application to local motion recognition in videos containing periodic actions. Victor et al. [[Continuous Video to Simple Signals for Swimming Stroke Detection with Convolutional Neural Networks](http://openaccess.thecvf.com/content_cvpr_2017_workshops/w2/papers/Victor_Continuous_Video_to_CVPR_2017_paper.pdf)] improves the task of stroke detection and stroke rate estimation by employing convolutional neural network. All works have done based on broadcast videos taken at races. While the approaches are robust for the swimmer localization, stroke detection, and stroke rate estimation in natural videos, they are not extensible to obtain performance indicators related to poses happening under water such as kick rate or pulling. Zecha et al. [[Pose Estimation for Deriving Kinematic Parameters of Competitive Swimmers](http://www.multimedia-computing.de/mediawiki//images/f/f8/Pose_estimation_for_kinematics_with_copyright_notice.pdf)], however, constrained their work to a lab environment, i.e., swimming glass channel, where the swimmer’s whole body, under and above water, is visible at all times such that almost all kinematic parameters are able to be quantitatively extracted. Greif and Lienhart [[A kinematic model for Bayesian tracking of cyclic human motion](https://opus.bibliothek.uni-augsburg.de/opus4/frontdoor/deliver/index/docId/1313/file/TR_2009_16.pdf)] similarly propose a kinematic model from video recorded from the side either with one camera in a swimming canal or two cameras in open-air pool; one camera recording above and one below the water surface. While Zecha et al. as well as Greif and Lienhart do pose detection from a supervised learning over manually labeled dataset of body joints’ location in frames, our work skips this step and relies on the-state-of-the-art pose estimation api, i.e., OpenPose. Plus, our work does not require such highly developed and sophisticated experimental environments.

Closely related to our work but in other sports are the works by Vicente et al. [[High Performance Moves Recognition and Sequence Segmentation Based on Key Poses Filtering](http://homepages.dcc.ufmg.br/~erickson/publications/vicente_wacv2016.pdf)] and Bidhendi et al. [[Combat sports analytics: Boxing punch classification using overhead depthimagery](https://eprints.qut.edu.au/93143/1/ICIP2015_2.pdf)]. Vicente et al. [[High Performance Moves Recognition and Sequence Segmentation Based on Key Poses Filtering](http://homepages.dcc.ufmg.br/~erickson/publications/vicente_wacv2016.pdf)**]** trains a model on the athlete key poses, extracted from frames, to identify kick and punch in Taekwondo. Bidhendi et al. [[Combat sports analytics: Boxing punch classification using overhead depthimagery](https://eprints.qut.edu.au/93143/1/ICIP2015_2.pdf)] do pose detection on overhead images of boxers to classify their punches to uppercut, straight, and hook. Like our proposed approach, these works rely on a pose detection api but with depth information in the frames as features. Vicente et al. extract pose features from frames by using RGB-D sensor (Microsoft Kinect) and Bidhendi et al. employ low resolution depth images. However, in our work we use OpenPose whose pose estimation does not need depth information.

1. **Swim Stroke Analysis**
   1. **Problem Definition**

Stroke cycle of the front crawl swimming style consists of four phases in order: i) catch, ii) pull, iii) push, which happen under the water, and iv) recovery which happens above the water as shown from the front view in the Figure 2. We focus on the pulling phase in our work for three reasons. Firstly, pulling generates almost all the moving drive forward and its efficient pose dramatically improves swimming performance in front crawl. Secondly, pulling pose in front crawl is shared among other swimming styles, i.e., butterfly and breaststroke. So, our work is easily extensible to more swimming styles. Thirdly, pulling analysis needs simple-to-record video frames in our framework. It only requires front view frames captured by a camera which is fixed in just under 10 centimeter of water (aligned with swimmer’s head on the z-axis) for up to 30 seconds. A typical swimmer is able to do so by a water resistant mobile phones such as iPhone 7 or Samsung S7 and there is no need for highly developed waterproof camera and sophisticated under-water recording skills.

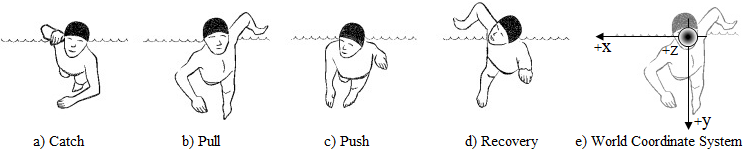


Figure 2. Front crawl stroke phases [[The Handbook of Swimming](https://www.amazon.com/Handbook-Swimming-Pelham-practical-sports/dp/0720720613)] and respective world coordinate system.

Now, given a amatuer-recorded video frames of a swimmer who swims towards the camera on the optical axis, our task is to predict her pulling pose as of being either high or dropping elbow. Simply, this implies pulling classification problem given the frames as our observation. We formally define our swim stroke analysis as follows:

**Definition 1. (Pulling Classification)** Let the camera and world coordinate systems are totally aligned with no translation but their z-axis are in opposite directions as shown in Figure 2 (e). Given 𝕍={f1:N} the recorded video 𝕍 consists of N frames, pulling classifier c: 𝕍→ {-1, 0, 1} is a function that maps a frame of a video f𝕍 to either 1; high elbow, or 0; dropping elbow, or -1; if the no recognizable pulling pose has been identified.

Our swim stroke analysis seeks to learn the pulling classifier c given a manually labeled frames from a set of 𝕍.

* 1. **Approach**

To learn the pulling classifier c, we perform a supervised learning method on a set of frames {f1:N} and their respective pulling labels {y1:N} where y{1; high elbow, 0; dropping elbow, -1; no recognizable pulling pose}. Each frame f is represented by a d-dimensional feature vector Φ(f) ∈ Rd; 1, where the function Φ extracts features by analyzing the swimmer pose. Our proposed approach can be briefly describes in two steps: i) pose extraction, and ii) model training, each of which is explained in the following.

**Pose Extraction.** To learn the pulling classifier c, we map each frame to feature space in which the arms’ poses and angles in the frame are the features. Formally, given a frame f ∈ 𝕍, we map it to d-dimensional feature vector by the function Φ: 𝕍→ Rd. The feature vector is concatenated from two types of arm’s skeletal information as follows:

* Joints: the 2-dimensional position of the right and left shoulders, elbows, and wrists
* Joint angles: the angle between upper arms and the water surface, namely ⍺, and the angle between upper arms and forearms, namely β

We show these features for a sample frame in Figure 1. We use OpenPose [[Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields](http://openaccess.thecvf.com/content_cvpr_2017/papers/Cao_Realtime_Multi-Person_2D_CVPR_2017_paper.pdf)][[2]](#footnote-1) to extract the joints. OpenPose is a real-time multi-person two dimensional (2D) pose detection. It is able to detect human body parts, including hand and facial keypoints (in total 130 key points) on single frame (image) in real-time. The choice of OpenPose is motivated for its functionality on image and video captured by webcam or IP camera which does not have extra depth information. OpenPose takes an image and identifies the locations of anatomical body points for each person happen to be in the image through a multistage feedforward convolutional network. At each stage, it predicts a set of body points; 17 from the body, 69 from the face, and 20 from hand, in the image plus a confidence score. We feed in OpenPose with each frame to detect only the joints, particularly the shoulder, elbow, and the wrist joints.

We calculate ⍺ and β based on where is the angle between two lines of slope and . As we assume that the recording settings in each frame follows the Definition 1, we only need to calculate the slope of line linking shoulder and elbow for ⍺ since the x-axis slope is zero. We compute the β based on the slopes of upper arm and the forearm as well.

**Model Training.** To extract the arm’s joints, we rely on the already trained model of OpenPose to detect the swimmers’ arm joints under water. We learn the pulling classifier function c through the classification methods, namely random forest [[Random Forests](https://link.springer.com/article/10.1023/A:1010933404324)] and svm [[Support-vector networks](https://link.springer.com/article/10.1007/BF00994018)] over feature vectors. Random forest is an ensemble method which averages the predictions of several decision trees as base classifiers; yielding overall better model over a single decision tree. More, random forest due to its base classifier is inherently able to learn multiclass classifications such as our pulling classifier c. In our random forest classifier, the number of decision trees in the forest is 100 and the criterion to measure the quality of a node split is Gini impurity for the information gain. As an alternative, we opt for svm with one-versus-rest (ovr) multiclass strategie as well.

1. **Experimentation**

In this section, we explain our test bed in terms of experimental setup, data acquisition, evaluation methodology, and results.

* 1. **Setup**

We used iPhone 7 and LG G6 both with 1920×1080×30fps to record the swimmers front crawl strokes under water. We fixed the camera 10 cm under water on the one end of swim pool wall and parallel to the water surface. In order to identify the body parts and poses, we extracted the video’s frames and applied OpenPose library with its default settings, i.e., COCO model to identify 17 body parts with neural net resolution 656368. We build and run OpenPose library on Intel Core i7 3770 with 4 cores, NVIDIA 1070 graphic card of 1920 GPU cores and 8GB frame buffer, and 16GB DDR3 of system memory. We used scikit-learn[[3]](#footnote-2) python library to train and evaluate our baseline classifiers and all the running parameters were left to the default values.

* 1. **Data Acquisition**

Our experiments include 25 swimmers of master swim club at Ryerson University[[4]](#footnote-3). The swimmers are from different levels of expertise. Fast (beginner), faster (novice), and fastest (professional) who swim in lines \{1,2\}, \{5,6\}, and \{3,4\} respectively. Also, they are sampled from wide range of swimmers with different genders, ages, and body sizes as shown in Figure 3. As a result, our study is independent of swimmer's physiological characteristics. We record swimmers' front crawl pulling strokes under water from the front view as they approach from one side of the pool to the wall where our camera is installed (one lap swim of 25 yd = 22.86 m). We then extract frames and filter out those in which no arm joint has been detected by OpenPose. This way we filter out the starting frames where no trace of swimmer has been detected either by camera or OpenPose. This leads us from 15,384 frames to a dataset of 2,633 frames. We manually labeled all frames by a swimming expert. Table 1 shows the distribution of different labels in our dataset. As shown, the distribution is skewed toward the ‘no pulling pose’ which could be due to the fact that either OpenPose falls short to correctly detect the swimmer’s arm joints in a frame or the pulling pose is neither high nor dropping.

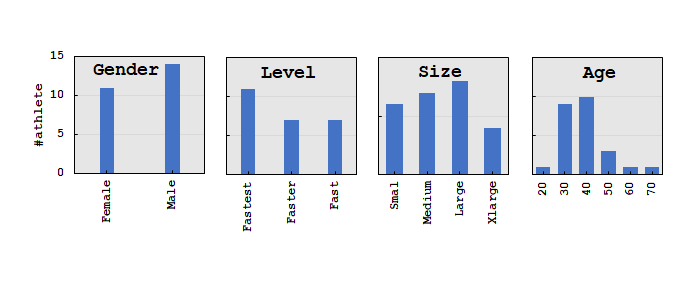


Figure 3. Distribution of the swimmers over level of expertise, gender, size, and age in our dataset .

|  |  |  |
| --- | --- | --- |
| **label** | **#** | **%** |
| no pulling pose | 1274 | 48.38587 |
| dropping elbow | 1001 | 38.01747 |
| high elbow | 358 | 13.59666 |
| total | 2633 |  |

Table 1. Distribution of different pulling poses in our dataset.

* 1. **Evaluation and Results**

We evaluate our multiclass classifiers, random forest and svm, on our dataset with three different feature subsets each of which includes only i) joints ii) joint angles (alpha and beta) iii) and both joints and angles. Due to label imbalance distribution, we did stratified 10-fold cross-validation and report the performance of each baseline by *weighted* average of all three classes for precision, recall, f1-measure, as well as accuracy in Table 2.

As evident in Table 2, all random forest baselines unanimously outperform the svm counterparts. We attribute this to the fact that random forest is an ensemble classifier whose prediction is based on the average over several independent decision trees. This is better than a single base classifier due to the lower variance. Among the random forest and svm baselines, we observe that joints contribute more than angles to learn a better classifier in all metrics. For instance, while random forest with angles as feature set could not perform better than 56\% in accuracy, joints excels and reach to 66\%. Nonetheless, joints plus the angles leads to the best performance with minor 1% improvement in the random forest baseline.

Table 2. The performance of our baselines.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **model** | **feature set** | **precision** | **recall** | **f-measure** | **accuracy** |
| random forest | joints+angles | **0.652** | **0.666** | **0.649** | **0.666** |
| joints | 0.643 | 0.656 | 0.640 | 0.656 |
| angles | 0.546 | 0.557 | 0.547 | 0.557 |
| svm | joints+angles | 0.482 | 0.498 | 0.442 | 0.498 |
| joints | 0.457 | 0.439 | 0.383 | 0.439 |
| angles | 0.422 | 0.437 | 0.396 | 0.437 |

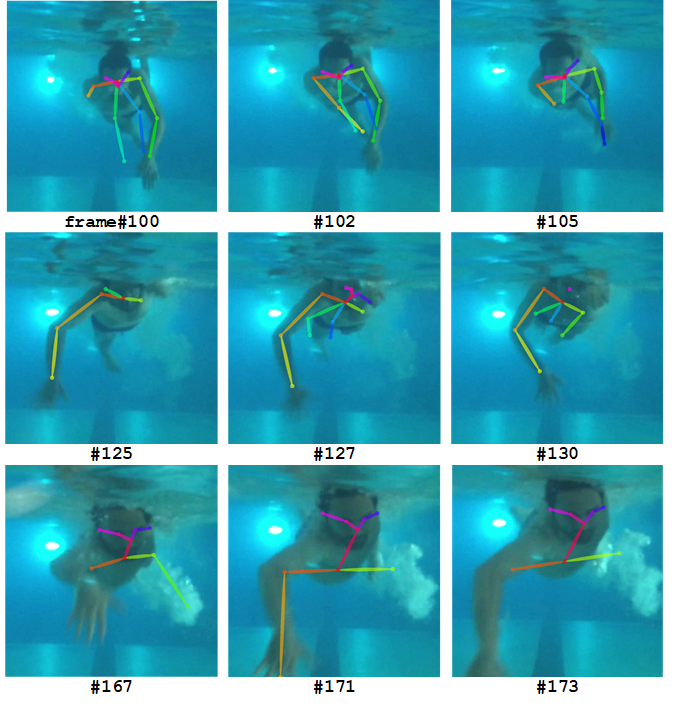


Figure 4. Sample user and his respective identified arm joints for ‘dropping elbow’ (first row), ‘high elbow’ (second row), and ‘no pulling pose’ (last row).

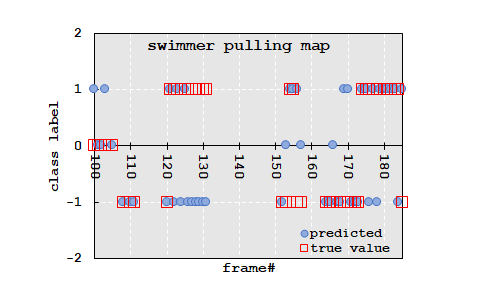


Figure 5. Pulling prediction map for a swimmer shown in Figure 4 by random forest classifier with arm joints and angles as the feature subset.

1. **Discussion and Future Direction**

The goal of this study is to give feedbacks to the swimmer about her efficiency of her pulling under water. We introduce the concept of *swimmer pulling map* which reveals a swimmer’s pulling pose class prediction within the frames (times) in her amateur-recorded video. Pulling map is able to give feedbacks on the swimmer’s pulling efficiency over the course of different frames. For instance and for the swimmer in Figure 4 we show the respective map in Figure 5. As seen in the frames and the pulling map in Figure 4 and 5 respectively, while the swimmer is doing dropping elbow on his left arm, he is doing high elbow on the right arm. Presumably, the swimmer is right handed and has some frailty in his left and needs to practice more on his left arm pulling.

Two possible future directions to our work are: (1) Our pose detection library, OpenPose, is not perfect and shows some false detection as in Figure 4 (last row). As we said in Section 3.2, we assume the pose detection model of OpenPose *a priori*, which is not specifically trained for under-water environment. Moreover, we have not include any prepossessing step on the video frames in our approach. These were intentional since we wanted to show the performance of our work with bare minimum configuration. An improvement to our work would be re-training the OpenPose on the datasets of preprocessed frames in each of which the positions of the swimmers’ joints are labeled as well already. (2) At a higher application level, we aim to extend our work to breaststroke and butterfly swim styles as the pulling phase analysis is very similar to front crawl.

1. **Conclusion**

In this work, we could automatically predict the efficiency of swimmers’ pulling stroke, which happens under water. Specifically, we applied the state-of-the art pose detection method, i.e., OpenPose, to underwater amateur recorded front crawl swimming stroke. Then, we extract the angles made by the swimmers arm during swimming plus the arm joints’ locations on each frame to train a classifier. We evaluate our proposed approach on a manually labeled dataset of swimming frames. Our results show that features including arm joints (shoulder, elbow, wrist) and the angles between upper arm and water surface and between upper arm and forearm, can achieve up to 67% accuracy with random forest which is on par with human judgment.

1. **Acknowledgement**

This work would not have been possible without the support of the swimmers at Ryerson’s master swim club. We are especially indebted to Danica Vidotto, the head coach, who have been supportive of our project and worked actively to provide us with meet ups to videotape the swimmers. We are also grateful to computer vision and image processing laboratory at Ryerson University. We would especially like to thank Dr. Javad Alirezaie[[5]](#footnote-4), the director of the lab, as he provides us with a gpu workstation to run our experiments. The last, not the least, we are immensely thankful to our labmate, Negar Arabzade, who provided us with her iPhone 7 to do underwater recording.

---- Comments from the Reviewers: ----

Importance/Relevance: **Of sufficient interest**

Novelty/Originality/Contribution: **Very original**

Comment on Novelty/Originality/Contribution: **This paper studies an interesting and challenging problem of analyzing swimming efficiency using computer vision. It uses classifiers to distinguish high elbow and dropping elbow. It defines and predicts pull efficiency.**

Technical Correctness: **Definitely correct**

Experimental Validation: **Limited but convincing**

Clarity of Presentation: **Very clear**

Reference to Prior Work: **References adequate**

-----

Importance/Relevance: **Of sufficient interest**

Comment on Importance/Relevance: **Quite interesting approach to swim-stroke recognition**

Novelty/Originality/Contribution: **Moderately original**

Comment on Novelty/Originality/Contribution: **Each of the methods combined is well known, but the combination and application are new to me**

Technical Correctness: **Probably correct**

Experimental Validation: **Limited but convincing**

Comment on Experimental Validation: **The authors analyze a fairly large number of video frames, and discuss the results fairly , I feel. They properly consider class imbalances**

Clarity of Presentation: **Very clear**

Comment on Clarity of Presentation: **Well written paper**

Reference to Prior Work: **References adequate**

-----

Importance/Relevance: **Of sufficient interest**

Novelty/Originality/Contribution: **Moderately original**

Comment on Novelty/Originality/Contribution: **The authors use two classifiers on top of OpenPose applied to frontal videos of swimmers.**

Technical Correctness: **Contains minor errors**

Comment on Technical Correctness: **I do not see how the efficiency, which is supposed to be estimated, is really measured. What are the output classes?**

Experimental Validation: **Limited but convincing**

Comment on Experimental Validation: **The authors hand labeled 2600 frames for the crawling phase , which is a good size of data set. The results are not overwhelming in terms of accuracy.**

Clarity of Presentation: **Clear enough**

Comment on Clarity of Presentation: **The English could be improved (I think "relieved" is used wrongly, and "amateurly" does not really exist.**

Reference to Prior Work: **References adequate**

1. We publicly release the dataset as well as all other artifacts such as codes, features, and the results at <https://goo.gl/mxz6tG>. We excludes the raw recording videos of swimmers due to the privacy. [↑](#footnote-ref-0)
2. https://github.com/CMU-Perceptual-Computing-Lab/openpose [↑](#footnote-ref-1)
3. http://scikit-learn.org [↑](#footnote-ref-2)
4. http://www.rec.ryersonrams.ca/ViewArticle.dbml?DB\_OEM\_ID=22310&ATCLID=204954666 [↑](#footnote-ref-3)
5. https://www.ee.ryerson.ca/people/Alirezaie.html [↑](#footnote-ref-4)