Robust estimation of flight parameters for ski jumpers

Katja Ludwig, Moritz Einfalt, Rainer Lienhart

Angaben zur Veröffentlichung / Publication details:

Ludwig, Katja, Moritz Einfalt, and Rainer Lienhart. 2020. “Robust estimation of flight parameters for ski jumpers.” In *IEEE International Conference on Multimedia and Expo 2020 (ICME), London, UK, 6-10 July 2020*, edited by Shuai Wan, Ce Zhu, Jian Zhang, Jianquan Liu, Shiwen Mao, Wen-Huang Cheng, and Jiwen Lu, 1–6. Piscataway, NJ: IEEE. https://doi.org/10.1109/ICMEW46912.2020.9105973.

ROBUST ESTIMATION OF FLIGHT PARAMETERS FOR SKI JUMPERS

Katja Ludwig, Moritz Einfalt, Rainer Lienhart

Multimedia Computing and Computer Vision Lab
University of Augsburg
{katja.ludwig, moritz.einfalt, rainer.lienhart}@informatik.uni-augsburg.de

ABSTRACT

This paper presents a model that robustly estimates important flight parameters for ski jumpers during their flight phase based on several camera views from the side along the jumpers' typical flight trajectories. A convolutional neural network for pose estimation, but also trained to detect skis, serves as a base model. It identifies 98.0% of the relevant flight parameters correctly within an angle threshold of 5 degrees, improving by 11.6% over previous work. In postprocessing, a pose checker first removes all wrong poses by using comparisons of distances and relative positions of the detected keypoints. A second step executes two RANSAC variants. One robustly estimates the average pose and another one the average pose angles. This model lifts the detection performance to 99.3% of the relevant flight parameters within a threshold of 5 degrees.

Index Terms— computer vision, sports, human pose estimation, robust estimation

1. INTRODUCTION

Ski jumping is an Olympic discipline in which the success of athletes highly depends on the body posture during the jump. A ski jump can be divided into four phases. In the first phase, athletes slide down the in-run and gain speed. While approaching the take-off table, ski jumpers lift their body and take off with the help of their speed and their own leap (phase 2). During launch and the following flight phase (phase 3) it is important for the athletes to position their body perfectly in order to increase lift, which is necessary to achieve a long flight distance. In the fourth phase, the athletes land on the ground. The landing point determines the final jumping distance.

Athletes work hard to achieve the perfect body posture at take-off and during the flight phase in every jump. Therefore, many ski jumping hills are lined with cameras along the flight trajectories of the ski jumpers. Coaches evaluate the recorded jumps by selecting frames manually, annotating relevant keypoints by hand and calculating the flight parameters using these hand-annotated keypoints. The system proposed in this paper fully automates this process. Given the videos from all cameras along the hill that belong to a single jump as input, our model (1) detects keypoints of the athlete as well as ski tips and ski tails in each video frame of each camera if present, (2) executes a robust estimation in each camera view based on the single-frame results and (3) outputs the flight parameters for each camera. The relevant flight parameters for the coaches are shown in Figure 1. Based on these parameters and a Principal Components Analysis, it is possible to predict if a jump has a long, medium or short distance.

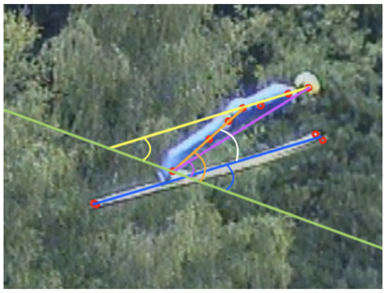


Fig. 1. Relevant angles of a ski jumping pose: upper body angle (yellow), lower body angle (orange), total body angle (purple), average ski angle (blue) and angle difference between lower body and skis (white). The green line represents the tangent to the flight trajectory of the athlete.

2. RELATED WORK

Human pose estimation is an active research field in computer vision. Most recent methods use deep convolutional neural networks (CNNs) for that task, like the currently best approaches [1, 2] on human pose estimation benchmarks like MPII Human Pose [3] and COCO [4]. Regarding the architecture, recent human pose estimation approaches are dividable in single-stage [5, 6] and multi-stage [1, 2, 7, 8] methods. The basis of single-stage approaches are mainly networks that perform well on image classification tasks, like ResNet [9] or VGG [10]. Mask R-CNN [5], which we use as a base model, firstly determines regions of interest and then executes single-person pose estimation on these regions. Multi-stage approaches [1, 2, 7, 8] try to refine the pose estimates in every

978-1-7281-1485-9/20/$31.00 ©2020 IEEE

stage.

Computer vision has become quite popular in analyzing athletes of different sport disciplines. [11] propose a user-assisted method for estimating and tracking athlete poses from monocular TV sports footage. Their model is evaluated on hurdles and triple jump videos. [12] use a multi-step architecture to estimate the poses of ski jumpers. With a convolutional sequence to sequence model, they predict the jump forces of ski jumpers directly from the pose estimates. With the usage of a dilated convolutional network [13] automatically detect events like ground contact in pose sequences of triple jump recordings. [14] predict the location of the basketball from a monocular view, even if it's occluded, based on the trajectories of the players. For a performance analysis of swimmers, [15] use a convolutional neural network with frame sequences of the swimmer and the swimming style as inputs. The knowledge of the swimming style and the usage of a pose refinement over time based on a pose sequence of fixed length improve the results per frame.

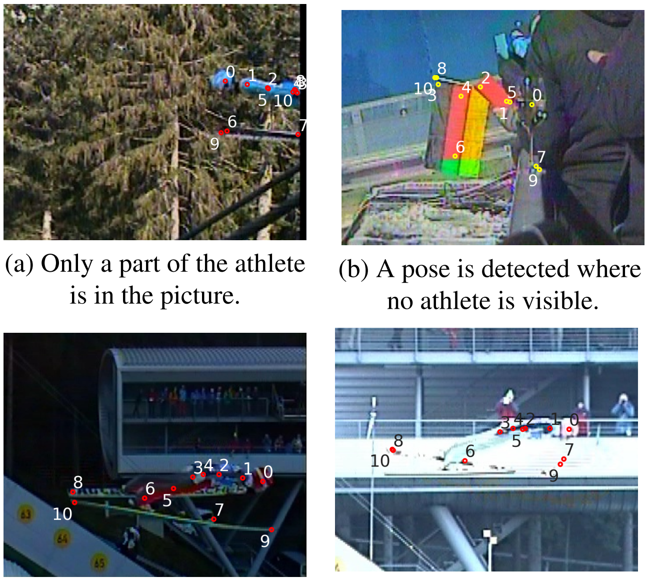
For many computer vision applications, robust estimation is an important step as results are often computed from noisy data with some outliers. A popular strategy for robust estimation is Random Sample Consensus (RANSAC) [16], which uses some samples of the whole data set to compute the model parameters and then calculates how many data points from the whole data set are in conformity with this model. After some iterations, the model with the highest number of matching data points is chosen. [17] improve this method by adding local optimization after choosing the best model. [18] propose a differentiable version in order to make it includable in end-to-end trained deep learning pipelines.

3. MODEL ARCHITECTURE

In a previous system [12], the keypoint detection was split into separate steps. At first, the position of the ski jumper was located within the frame using MobileNet [19]. Next, at this location a convolutional pose machine [8] detected the athlete's joints. Third, a Hough transformation was used to identify the skis. For each camera view, the mean pose was calculated afterwards and the flight parameters were computed based on this mean pose. A careful evaluation by the coaches and performance diagnosticians showed that this multi-step model generates mostly reasonable results regarding the single-frame results, but the usage of a mean pose often impairs the final result due to outliers. This happens especially often for the ski detections, as the Hough transformation produces more than sporadically false results.

Hence, we have developed a new model that performs all detections in one single step. It is based on Mask R-CNN [5], but uses a branch to detect keypoints instead of generating segmentation masks. This model is also able to learn non-body keypoints like ski tips and ski tails, which is more reliable than the previously used Hough transformation.

In the hand-annotation process, the flight parameters were derived as the mean of the angles of all annotated poses per camera view (usually 2-4). However, there are many more frames per camera showing the complete athlete. Thus, the new model can use all images from each camera view. On average, it detects a ski jumper in 14 images per camera. These pose detections are passed on to the postprocessing, where they first undergo a plausibility check to identify gross mistakes. These checks are based on the keypoints of the pose itself: The system checks (1) if the length of both skis is nearly equal, (2) if the length of the body (the distance head to hip plus hip to ankle) is not shorter than half of and not longer than the ski length, (3) if the head is above the ski tips, (4) if the hand is far enough from the ski tips or tails and the head, (5) if the length of the lower leg and thigh are nearly the same, (6) if the size of the upper body is similar to the size of the lower body, (7) if all keypoints are not too close to the image boundaries and (8) if all joints (except for the ankle) are on one side of the skis. Poses that do not pass these checks are removed. Examples for invalid poses can be found in Figure 2b)-d). Furthermore, this step sorts out poses where only a part of the athlete is shown in the picture. An example can be found in Figure 2a). The model already detects the ski jumper, but the poses are not precise enough to contribute to the final result. Therefore they are discarded. The pose checking process removes around 42.8% of all poses, so that on average 8 images remain per camera. Figure 3 visualizes this effect.



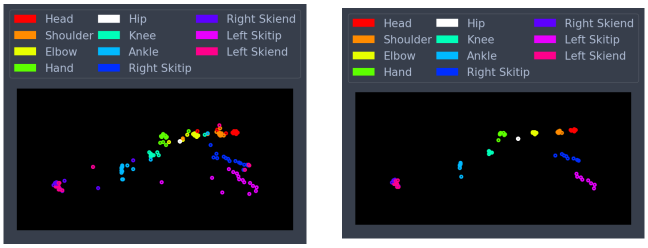
(a) Only a part of the athlete is in the picture.

(b) A pose is detected where no athlete is visible.

(c) The length of the skis does not match.

(d) Ankle and knee are at wrong positions.

Fig. 2. Invalid poses identified by pose checking. The detected keypoints are visualized by red circles and marked with numbers, whereby number 0 marks the head, 1 the shoulder, 2 the elbow, 3 the hand, 4 the hip, 5 the knee, 6 the ankle, 7/9 the right/left ski tip, 8/10 the right/left ski tail.



(a) All 15 poses of one camera view.

(b) Remaining 8 poses after plausibility check.

Fig. 3. Effect of pose filtering: On the left side, all poses of one camera view are displayed, centered at the hip joint. The right side shows the remaining poses after filtering.

The second postprocessing step takes all plausible poses and uses a robust estimation to output the final flight parameters. We use this technique as the pose of the athlete barely changes within one camera view. In the hand-annotation process, the mean is used, but using the mean is too sensitive to detection outliers. Hence, we use locally optimized RANSAC [17] in two variants for that purpose. The first variant calculates the relevant angles from the keypoint locations and applies a constant RANSAC model to each set of angles. In a second variant, the poses are normalized by translating the hip of an athlete to the origin of the coordinate system. A constant RANSAC model is applied to the normalized keypoints, which results in a robust mean pose. The flight parameters per camera are in turn derived from this mean pose.

Summarizing, the model consists of three main steps: (1) detecting keypoints of ski jumpers in all frames of one camera view, (2) checking the detected poses and removing the invalid ones and (3) robustly estimating the flight parameters based on RANSAC.

4. EVALUATION

4.1. Dataset

The dataset used in this paper was collected and provided by the Institute for Applied Training Science (IAT) in Leipzig. The training dataset contains 10,070 annotated images from 290 jumps. The videos were recorded at different ski jumping hills, during multiple events and with different athletes, so their statures and dressings vary. The footage also covers a wide variety of weather and light conditions, e.g. snow, rain, fog, summer, winter, day and night. Only few images from every video are annotated, usually 2-4 frames per camera. Annotated keypoints are both ski tips and tails, head, shoulder, elbow, hand, hip, knee and ankle. The annotations of the joints are only available of one side of the body (the one facing the camera). The dataset contains images of the flight phase as well as images of the athlete during in-run, where the skis are not visible and not annotated.

The results presented in this section are computed using an independent test set with 3,388 images from 101 different jumps. Figure 4 shows an exemplary pose estimate on the test set. The numbers mark the same keypoints as described in Fig. 2.

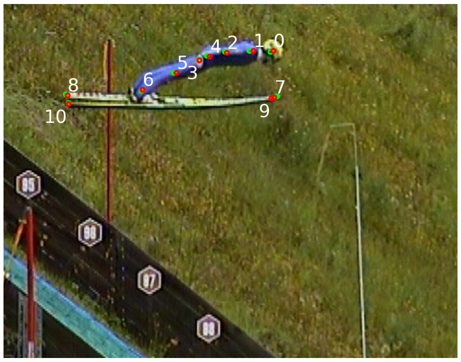


Fig. 4. Example of a pose estimate (red) with ground truth (green).

4.2. PCK and PCA

We use two different evaluation protocols. The first protocol compares the estimated flight parameters per camera view with the mean of the angles from the annotated poses of one camera view. Our evaluation focuses on this protocol, since it complies with the evaluation by coaches. The second protocol evaluates the pose estimation results image-wise on the video images with ground truth annotations available. We will refer to this technique as the evaluation on annotated images. We only apply it if mentioned explicitly. Using the second protocol allows to calculate the Percentage of Correct Keypoints (*PCK*). PCK considers a keypoint as correct at a certain threshold t if the distance of the detected keypoint to the ground truth keypoint is less or equal than t times the distance between shoulder and hip joint. The recall at a certain PCK threshold tells us the percentage of the keypoints that is considered correct at that threshold. The PCK curves for varying thresholds are visualized in Figure 5 with solid lines.

As the interest of this paper is not the raw joint positions but the body angles, we also define the Percentage of Correct Angles (*PCA*). Analogous to PCK, PCA considers an angle as correct at a threshold t if the difference between the angle computed from the detected joints and the angle calculated from the ground truth joints is below or equal t. The recall at a PCA threshold t measures the percentage of the angles that are considered correct at threshold t. Table 1 shows the recall values at PCA thresholds of $5°$ and $3°$ for all five relevant angles for the ski jump coaches (upper body angle, lower body angle, total body angle, ski angle and difference between lower body and ski angle).

If the system detects a ski jumper, it is nearly always a cor-

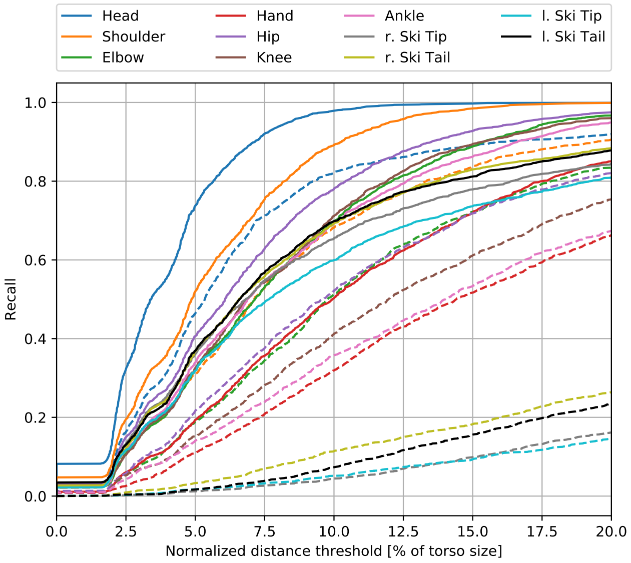


Fig. 5. PCK curves for varying thresholds on the test set. The results of the proposed model are displayed with solid lines, the results from the previous system with dashed lines.

rect detection. The remaining false detections are removed by the pose checker. Hence, we do not use precision as a metric. The main focus is the compliance of the detected keypoints with the ground truth. Therefore, the distance between a detected keypoint and its corresponding ground truth position is the metric of interest.

4.3. Comparison with Previous Model

Regarding the previous system [12] with the multi-step pose estimation, the current model achieves greater accuracy for all keypoints. The PCK values for the previous model are visualized in Figure 5 with dashed lines. Huge differences are encountered in the detection of the skis. The recall of the previous model at a PCK threshold of 20% is only 14.9% for the left ski tip and 16.4% for the right ski tip (Figure 5, dashed lines), while the current model achieves 81.2% and 84.3%, respectively (Figure 5, solid lines). The reason for this huge difference is that the Hough transformation used for the ski detection is far less precise. The previous system could not include it in the neural network training as the annotations at that time included only the averaged position of left and right ski tips and tails, meaning that a ski tip annotation was located in the middle between left and right ski tip.

Although the positions of the ski tips and tails are not accurately estimated with the Hough transformation, the results for the derived angles of the skis are good. 88.4% of the detected ski angles are within ±5°. In general, as Table 1 shows, the new model improves the recall values at a PCA threshold of 5° for all five relevant angles at least by 4.0% and at most by 22.9%, and by 13.0% to 30.0% at a PCA threshold of 3°.

Table 1. Recall values in % at PCA thresholds of 5° and 3° on annotated test set images. C stands for the current system, P for the previous one.

System	Lower Body	Upper Body	Total Body	Ski	Diff. L.B./S.	Total
C@3°	**84.6**	**92.3**	**97.4**	**97.4**	**81.3**	**90.6**
P@3°	64.0	75.6	84.4	80.9	51.3	71.3
C@5°	**97.4**	**99.0**	**99.8**	**98.6**	**94.9**	**98.0**
P@5°	84.3	91.0	95.8	88.4	72.0	86.4

4.4. RANSAC on Angles

The first variant of robustly estimating the flight parameters starts with the calculation of the five angles of interest for every pose. As a second step, for each camera view, a constant model is estimated with RANSAC for each angle. We use 100 iterations with a sample size of 4. Hence, for each type of angle, 4 computed angle values are randomly chosen, and the average is calculated. Then, the number of inliers for this model among all calculated angle values is computed. Other angles are defined as inliers if they deviate at most by 4° from the average angle of the samples. For the model with the most inliers, the final result is calculated as the average angle of all inliers. The results for the PCA metric are shown in Figure 6. The results for all flight parameters, except the ski angle, are improved. Ski keypoints are the keypoints with the lowest PCK (see Figure 5) and have therefore the highest estimation errors. For angle RANSAC, the angles are calculated in advance. Some were calculated based on wrong poses, but ended up as inliers, if they fit the threshold. Hence, they might cause the final result to differ too much from the true value.

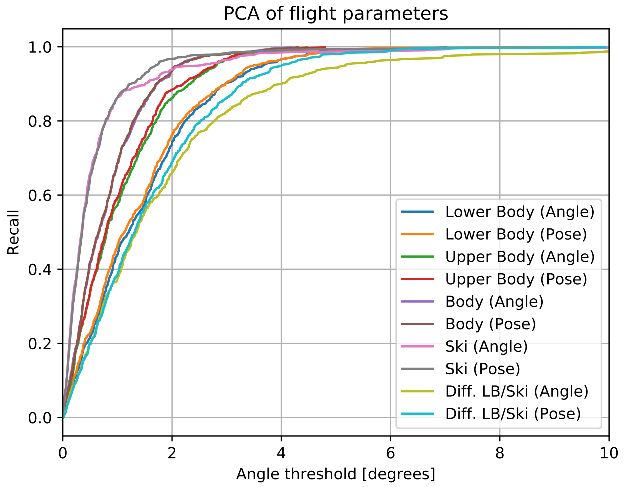


Fig. 6. PCA curves for varying thresholds on test set for RANSAC on angles and RANSAC on poses.

4.5. RANSAC on Poses

The second approach translates the poses such that the detected hip joint lies in the origin of the coordinate system. All poses are now relative to the hip. RANSAC is executed on the translated joint coordinates with 100 iterations and 4 or 5 samples. Experiments on the validation set show that using 5 samples for the ski keypoints and 4 samples for the body joints achieves the best results. Hence, for each keypoint, 4 or 5 random samples are chosen and the average keypoint coordinates are calculated. The number of inliers among all values for this keypoint is computed afterwards. We define other keypoints as inliers if their distance to the average point of the samples is at most 35% of the currently estimated torso size. The result of this computation is a robustly estimated mean pose for each camera view. The angles of the flight parameters are now calculated by using the keypoints of the mean pose. Figure 6 shows the PCA metric for this approach. RANSAC based on poses improves the results for all flight parameters (see Table 2).

4.6. Comparison of RANSAC on Angles and Poses

Both RANSAC methods generate good results and improve the PCA values achieved on the annotated images, except RANSAC on angles for the ski angle. Table 2 displays the recall values at PCA thresholds of 3° and 5°. RANSAC based on poses also improves the results for ski angles. The reason is that the calculation of an average pose is more precise for the skis, as only keypoints that are close together are included in the angle calculation. Table 2 shows that RANSAC on poses generates the best results for all angles.

Table 2. Recall values in % at PCA thresholds of 3° and 5°: results on annotated images are marked with A, results with RANSAC on angles with B and results with RANSAC on poses with C.

	Lower Body	Upper Body	Total Body	Ski	Diff. L.B./S.	Total
A@3°	84.6	92.3	97.4	97.4	81.3	90.6
B@3°	90.1	96.4	**97.9**	96.3	82.3	92.8
C@3°	**90.3**	**96.7**	**97.9**	**98.2**	**86.1**	**94.0**
A@5°	97.4	99.0	99.8	98.6	94.9	98.0
B@5°	**98.9**	**100**	**100**	98.6	94.3	98.4
C@5°	**98.9**	**100**	**100**	**99.2**	**97.9**	**99.3**

4.7. Jumping Distance Prediction Based on Flight Angles

The dataset provides 84 videos with the information of the jumping distance. All videos are from the same ski jumping hill and recorded with the same camera settings. We use a Principal Component Analysis to map the five estimated flight parameters to their two principal components. Figure 7 depicts the flight parameters for two camera views in this two-dimensional sub space with respect to the color-coded jumping distance. An exact prediction of the jumping distance is not feasible, as important information like the wind speed and direction is not available, but it is possible to see clusters of jumps with high distances in both visualizations (colored yellow in the left figure and yellow and orange in the right figure). Hence, if a new jump is recorded and the flight parameters extracted, it is possible to perform the same Principal Component Analysis as before and predict if the jump has a long, medium or short distance.

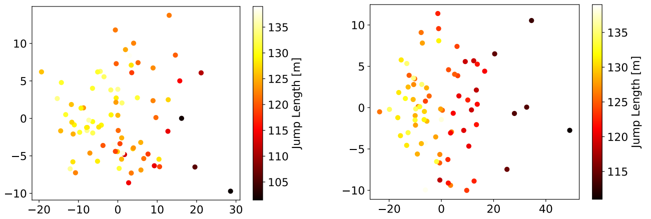


Fig. 7. Results of Principal Component Analysis based on all flight parameters for two cameras.

5. CONCLUSION

This paper proposes a technique for robustly estimating flight parameters for ski jumpers. It uses a Mask R-CNN [5] based model to detect the joints of the athlete and the ski tips and tails. As the coaches are interested in the flight parameters per camera, we can use a robust estimation based on all detections of one camera view to determine the values. This robust estimation includes a pose checker that removes wrong poses. It executes checks based on the distances and relative positions of the keypoints, e.g. it compares the body and the ski length or the length of upper and lower body. In a second step, RANSAC calculates the robust estimation of the poses considered valid by the pose checker. We examined two versions of RANSAC. The first operates already on calculated angles based on single poses, the second operates on single poses and calculates the final flight parameters based on the estimated mean pose.

Our evaluations showed that the new model performs notably better than the previous one which used a multi-step pipeline to detect the body joints and a Hough transformation for the skis. Especially for the results of the ski angle and the difference between lower body and ski angle, the proposed model improves the recall at a PCA threshold of 5° by absolute +10% on the test set images. The percentage of correct angles can be improved even further with the usage of the robust estimation. The recall at a PCA threshold of 5° is over 97% for all flight parameters, using the best performing version of RANSAC, and over 86% at a PCA threshold of 3°.

6. ACKNOWLEDGEMENTS

This work was funded by the Federal Institute for Sports Science (BISp) based on a resolution of the German Bundestag. We would like to express our gratitude to the Institute for Applied Training Science (IAT) Leipzig, especially Dr. Sören Müller, for the collection and provision of the data used in this paper.

7. REFERENCES

[1] Zhihui Su, Ming Ye, Guohui Zhang, Lei Dai, and Jianda Sheng, "Cascade feature aggregation for human pose estimation," *arXiv preprint arXiv:1902.07837*, 2019.

[2] Yuanhao Cai, Zhicheng Wang, Binyi Yin, Ruihao Yin, Angang Du, Zhengxiong Luo, Zeming Li, Xinyu Zhou, Gang Yu, Erjin Zhou, Xiangyu Zhang, Yichen Wei, and Jian Sun, "Res-steps-net for multi-person pose estimation," *Joint COCO and Mapillary Workshop at ICCV 2019: COCO Keypoint Challenge Track*, 2019.

[3] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele, "2d human pose estimation: New benchmark and state of the art analysis," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[4] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[6] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun, "Cascaded pyramid network for multi-person pose estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7103–7112.

[7] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun, "Rethinking on multi-stage networks for human pose estimation," *arXiv preprint arXiv:1901.00148*, 2019.

[8] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[11] Mykyta Fastovets, Jean-Yves Guillemaut, and Adrian Hilton, "Athlete pose estimation from monocular tv sports footage," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 1048–1054.

[12] Dan Zecha, Christian Eggert, Moritz Einfalt, Stephan Brehm, and Rainer Lienhart, "A convolutional sequence to sequence model for multimodal dynamics prediction in ski jumps," in *Proceedings of the 1st International Workshop on Multimedia Content Analysis in Sports*, 2018, pp. 11–19.

[13] Moritz Einfalt, Charles Dampeyrou, Dan Zecha, and Rainer Lienhart, "Frame-level event detection in athletics videos with pose-based convolutional sequence networks," in *Proceedings Proceedings of the 2nd International Workshop on Multimedia Content Analysis in Sports*, 2019, pp. 42–50.

[14] Xinyu Wei, Long Sha, Patrick Lucey, Peter Carr, Sridha Sridharan, and Iain Matthews, "Predicting ball ownership in basketball from a monocular view using only player trajectories," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 63–70.

[15] Moritz Einfalt, Dan Zecha, and Rainer Lienhart, "Activity-conditioned continuous human pose estimation for performance analysis of athletes using the example of swimming," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 446–455.

[16] Martin A Fischler and Robert C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[17] Ondřej Chum, Jiří Matas, and Josef Kittler, "Locally optimized ransac," in *Joint Pattern Recognition Symposium*. Springer, 2003, pp. 236–243.

[18] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother, "Dsac-differentiable ransac for camera localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6684–6692.

[19] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.