AI Coach: Deep Human Pose Estimation and Analysis for Personalized Athletic Training Assistance

Jianbo Wang[*]
Zhejiang University
wjbkimberly@zju.edu.cn

Kai Qiu
Microsoft Research Asia
Kai.Qiu@microsoft.com

Houwen Peng
Microsoft Research Asia
Houwen.Peng@microsoft.com

Jianlong Fu[†]
Microsoft Research Asia
jianf@microsoft.com

Jianke Zhu
Zhejiang University
jkzhu@zju.edu.cn

ABSTRACT

Recent years have witnessed an unprecedented growing of sport videos, as different types of sports activities can be widely-observed (i.e., from professional athletics to personal fitness). Existing approaches by computer vision have predominantly focused on creating experiences of content browsing and searching by video tagging and summarization. These techniques have already enabled a wide-range of applications for sports enthusiasts, such as text-based video search, highlight generation, and so on. In this paper, we take one step further to create an AI coach system to provide personalized athletic training experiences. Especially for sports activities which the training quality largely depends on the correctness of human poses in a video sequence. As sports videos often involve grand challenges of fast movement (e.g., skiing, skating) and complex actions (e.g., gymnastics), we propose to design the system with several distinct features: (1) trajectory extraction for a single human instance by leveraging deep visual tracking, (2) human pose estimation by proposing a novel human joints relation model in spatial and temporal domains, (3) pose correction by abnormal detection and exemplar-based visual suggestions. We have collected sports training videos from 30 sports enthusiasts, namely Freestyle Skiing Aerials dataset (63 clips). We show that the proposed system can lead to a remarkably better user training experience by extensive user studies.

CCS CONCEPTS

• **Computing methodologies → Activity recognition and understanding**; *Tracking*; Object detection.

KEYWORDS

sports videos, object tracking, human pose estimation

ACM Reference Format:
Jianbo Wang, Kai Qiu, Houwen Peng, Jianlong Fu, and Jianke Zhu. 2019. AI Coach: Deep Human Pose Estimation and Analysis for Personalized Athletic Training Assistance. In *Proceedings of the 27th ACM International Conference on Multimedia (MM'19), Oct. 21–25, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3343031.3350910

1 INTRODUCTION

Sports events (e.g., Olympic Games, NBA, etc.) have attracted increasing interests from all over the world. With the increasing of different types of digital devices (e.g., digital cameras, smart phones), the number of sports videos have been witnessed an unprecedented growth. Existing algorithms on sports videos mainly focus on creating user experiences of content browsing and searching by video tagging and summarization technologies. For example, Tong et al. [40] use replay detection to efficiently localize the highlights in sports videos. Liu et al. [28] construct an infrastructure for extracting and delivering the highlights and design an UI for the mobile clients to effectively browse and interact with the video highlights. Kitani et al. [21] propose an unsupervised approach to discover first-person action categories, which can be useful for video indexing and retrieval.

A novel application is recently proposed to provide personalized athletic training experiences, as a plenty of records in various sports fields can be usually refreshed by improving training performance. It is reported that sports videos have been extensively used by athletes to analyze performances and improve skills [43]. For example, in the game of Freestyle Skiing Aerials, professional athletes often replay videos again and again to analyze detailed movements of opponents and themselves. The final score is relevant to many detailed parts of athlete pose (e.g., a slight bend at the waist, angle of the knees, split of skis, etc.). Accurate analysis of athletes pose are crucial cues to determine the overall score. In some other sports (e.g., dancing, gymnastic competition, etc.), the performance also grounded heavily on athlete pose.

However, at the present stage, sports videos are mainly processed by humans to analyze performances of players, which is neither efficient nor scalable, compared with fully automatic algorithms. Since deep neural networks dominate the field of computer vision, high-performance models for detection, tracking and pose estimation show great potential to process sports videos. For example, Zecha et al. [47] predict the jump forces of ski jumpers directly from pose

[*] This work was conducted when Jianbo Wang was a research intern at MSRA.
[†] Corresponding author.

MM '19, October 21–25, 2019, Nice, France

ACM ISBN 978-1-4503-6889-6/19/10…$15.00
https://doi.org/10.1145/3343031.3350910

estimates. Despite the great improvement of pose estimation models, the difficulties of fast movement and complex actions in sports videos still remain huge challenge to existing approaches.

In this paper, we design an AI Coach system to create personal athletic training experiences and help athletes to improve skills. This system consists of three closely-related components. (1) trajectory extraction for a single human instance by leveraging deep visual tracking, (2) human pose estimation by proposing a novel human joints relation model in spatial and temporal domains, (3) pose correction by abnormal detection and exemplar-based visual suggestions. Extensive experiments are conducted to demonstrate the effectiveness of this AI Coach system. First, to evaluate the ability of our proposed video pose estimation model, we show the state-of-the-art performance on two widely-used single-person pose estimation video datasets, i.e., sub-JHMDB [16] and Penn Action [48]. Second, we have collected Freestyle Skiing Aerials dataset (63 clips, 51 clips for training, 12 clips for validation) as training videos from 30 sports enthusiasts. We show that the proposed system can lead to a remarkably better user training experience by extensive user studies, compared with existing athletic training assistants on the consumer side.

2 RELATED WORK

2.1 Tracking and Detection

The state-of-the-art visual tracking methods typically use a one-stage regression framework or a two-stage classification framework. The representative one-stage regression framework is based on discriminative correlation filters [15, 24, 25, 27, 41] , which regress all the circular-shifted versions of the input features into soft labels generated by a Gaussian function. The two-stage tracking-by-detection framework [3, 23, 26] mainly consists of two steps. The first step draws a sparse set of samples around the previously predicted location, and the second step classifies each sample as the target or the background.

There are two established classes of methods for object detection in images, one based on sliding windows and the other based on region proposal classification. The former one is a one-stage detector, which directly uses backbone for object instance prediction, such as YOLO [36] and SSD [29]. The later one is a two-stage detector, including a region proposal generation branch and a regional classification branch, such as R-CNN [13] and faster-RCNN [37].

2.2 Pose Estimation

Traditional methods for image-based pose estimation has been focused on graphical structure modeling [1, 8, 17, 34, 35, 39, 46]. Recently convolutional neural networks approaches [4, 5, 14, 20, 32, 33, 42, 45] have achieved dominant results. The natural extension of single-image pose estimation, i.e, video-based pose estimation, has therefore gained much importance recently being able of representing activities and body motions.

Accuracy of pose estimation in videos can be improved by taking temporal context into account. In previous works, researchers devoted themselves to temporal information learning. These dominant approaches can be divided into two categories: using short-time information come from adjacent frames and using long-range temporal dependency. Here short-time methods means we only consider consistency constraints in adjacent frames. In later works [38], the authors proposed using optical flow cue to captures the temporal consistency of human poses in video sequences based on a designed spatial-temporal graph. Girdhar et al. [12] tried to extend Mask-RCNN [14] with spatial-temporal operations by inflating the 2D convolutions into 3D convolution. In [44], the authors proposed to use optical flow cue to warp estimated pose from adjacent frames, then align heatmaps to the current frame. While long-range means the receptive field along time in these methods are global. In [31], the author constructed an RNN style model to capture temporal correlation among video frames.

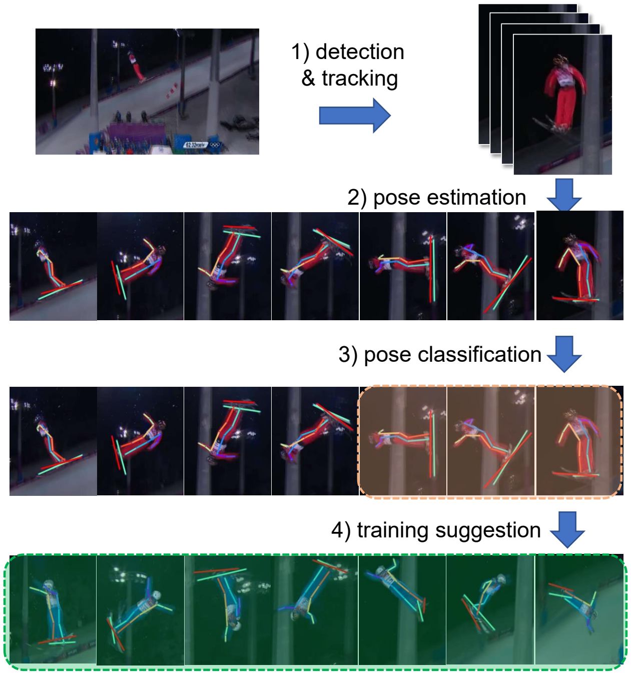


Figure 1: Framework of our proposed AI Coach system. It consists of four steps: (1) Detecting and tracking the athlete in video. (2) Based on the tracking trajectory, a pose estimation model is used to extract poses of this athlete. (3) Given the extracted poses, a classification model is used to recognize "bad poses". (4) Our system can present some related good examples for the athlete.

Since long-term temporal information is more useful for video pose estimation. However, in the proposed memory-based (eg. LSTM) architectures [31], messages need to be delivered between distant positions. This may causes gradients vanishing, loss and errors in temporal information. Thus, it is worthwhile to explore how to balance looking correctly and looking long-term dependencies.

2.3 Sport Video Applications

Significant efforts have been devoted to sport video analysis in recent years. Traditional approaches mainly focus on video tagging and retrieval to ease the review for coaches. Inspired by the success of deep learning in various vision tasks [9, 10], deep convolutional neural networks have been applied to improve the analysis process. There have been sport video analysis systems from both industry and academia, such as IBM's golf highlights system [18], Kitani et al. [21] and Liu et al. [28]. However, little effort has been made towards utilizing instance-level features, such as keypoints which are crucial for human-centric analysis. Also, valuable suggestions are required for coach to know what to do after they see the problems.

3 OVERVIEW OF PERSONALIZED ATHLETIC TRAINING ASSISTANCE SYSTEM

The proposed AI Coach system aims to analyze poses and highlight "bad poses" for athletes in sports videos. Specifically, for each athlete in a video, we utilize a human detector and tracking model to build a tubelet which focus on this athlete. Then, our proposed single-person video pose estimation model extract the pose in each frame. Based on the extracted poses of an athlete, a pose classification model is designed to recognize "bad poses". These "bad poses" are highlighted to athletes, and thus help athletes revise their poses. The pipeline consists of three steps: (1) Trajectory extraction. Given a sport video, a human detector is utilized to localize all humans in the first frame. For each human bounding box extracted by the human detector in the first step, a tracking model is used to track this human from the second frame to the last frame of the video. When tracking is finished, each human in this video is surrounded by a tubelet along the video. (2) For each human tubelet, our proposed single-person video pose estimation model is used to extract human pose in each frame. (3) Based on the extracted pose, we design a classification model to recognize "bad poses" and highlight these "bad poses" to warn the athlete and show standard poses to the user.

3.1 Detection and tracking

Since sports videos mainly focus on athletes, the athletes are usually very obvious in the videos, especially for the games that need scoring. Similar to R-CNN serial models [13, 37], we design a binary human detection module to detect human in the first frame. Given the bounding box of human in the first frame of a video, a real time online tracking model is used to extract trajectory for a single human instance.

3.2 Pose estimation

Sports videos usually suffer from blur due to the fast movement of athletes. To relieve this issue, we propose a pose estimation model to take advantage of spatial and temporal relation of human keypoints in videos. Human keypoints in videos have strong spatial and temporal relation between each other. Here the spatial relation means the structural information of human body (e.g., elbow is close to shoulder and wrist), and the temporal relation means the the smooth movement of a keypoint along time dimension. According to spatial relation between human keypoints, it is possible to estimate the location of one keypoint (e.g., elbow) given the locations of other keypoints (e.g., shoulder and wrist) at the same timestamp. As for temporal relation, it is also possible to estimate the location of one keypoint (e.g., elbow) given the locations of the same keypoint in a sequential timestamps (e.g., a sequence of elbows). Inspired by this, we design a video pose estimation module to refine coarse heatmaps generated by a basic image pose estimation model, by using the spatial and temporal relations among coarse heatmaps. This module consists of two branches, one for modeling spatial relation of different keypoints at the same timestamp, and another one for modeling temporal relation of the same keypoint at a sequential timestamps. Futhermore, this module can be formulated into multi-head and multi-layer manner to improve performance.

3.3 Pose classification

Many sports have clear criterion to judge whether a pose is good or bad, e.g., skiing, diving, gymnastics, and so on. Given the label of poses annotated by sports experts, a classification model is used to recognize "bad poses". These bad poses are highlighted to warn the user.

3.4 User Interface

We design a simple yet efficient user interface for users. To analyze a sport video, it only takes three steps: (1) Upload a sport video or shoot a video. (2) After uploading a video, this system will launch the detection, tracking, pose estimation and pose classification module, and then present the classification results for each frame to the user. (3) Detected "bad poses" and estimated score will be presented for the user. User can slide mouse to view all the "bad poses" recognized by this system. Also the suggested videos with standard poses will be presented to the user.

4 TRACKING APPROACH

AI coach system demands fast and accurate tracking of athletes' position to provide selected regions for subsequent pose estimation and scoring. However, existing open-source tracking algorithms fail to fulfill our following requirements:

- Human-centric algorithm. Existing tracking algorithms are designed for arbitrary objects [22], without constraints on object classes. This kind of algorithms is general, but not qualified in sport video analysis, since that we only pay attention to human's position and motion in sport videos.
- Fast speed and low error. For an online system, low latency and high speed are crucial for user experience. Moreover, different from other video types, sport videos include heavily motion blur due to fast motion of athletes or cameras. Therefore, tracking algorithms need to modeling fast motions, and guarantee high accuracy.

To address these issues, in this section, we propose a human-centric tracking algorithm designed for sport videos. It mainly consists of three modules: detection, tracking and verification. The detection module identifies the position of target athletes in the first frame. Then, the tracking module tracks the athlete's motion and position in the subsequent frames. Meanwhile, if the tracking module is not confident, the verification module is triggered. It will selects proposals from the whole image, and verifies each proposal is the target or not. As a whole, these three parts share the same backbone network to guarantee fast speed. Moreover, both the detection and tracking modules are fine tuned on human-centric data, and make the system to be more robust on human detection and tracking.

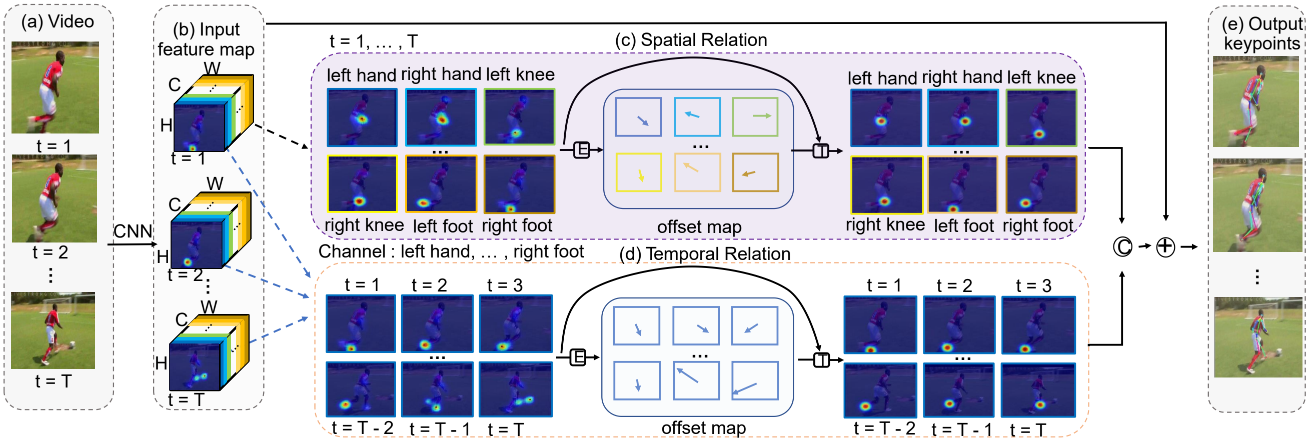

(a) Video
t = 1
t = 2
⋮
t = T
CNN
(b) Input feature map
C
W
H
t = 1
t = 2
t = T
t = 1, ... , T
(c) Spatial Relation
left hand right hand left knee
right knee left foot right foot
offset map
left hand right hand left knee
right knee left foot right foot
Channel : left hand, ... , right foot
(d) Temporal Relation
t = 1 t = 2 t = 3
t = T - 2 t = T - 1 t = T
offset map
t = 1 t = 2 t = 3
t = T - 2 t = T - 1 t = T
E
T
©
⊕
(e) Output keypoints
⋮

Figure 2: Spatial-Temporal relation module. Given a sport video, a CNN model (e.g., ResNet-50) is used to extract features from each frame. This feature has the size of $[T, C, H, W]$ and is fed into the Spatial-Temporal Relation Module (STRM). Spatial Relation Module is utilized to extract spatial relation among different keypoints, which is conducted in each time frame separately. Temporal Relation Module is used to extract temporal relation of a specific keypoint among time dimension, which is conducted for each keypoint separately. The spatial and temporal relation module are realized with residual connection. Features output from spatial relation module and temporal relation module are concatenated and go through a convolution layer to reduce dimension.

Detection. Similar to R-CNN serial models [13, 37], we design a binary human detection module. It contains a region proposal network to extract candidate regions, and a classification network to identify a region is human or not. Both of these two networks are implemented by two fully-collected layers. The region proposal network is performed on candidate anchors [37]. An anchor is centered at the sliding window, and associated with a scale and aspect ratio. By default, we use 3 scales and 3 aspect ratios, yielding $k = 9$ anchors at each sliding position. Since that, for sport videos, the target athlete commonly presents at/near the center of the image, we remove the candidate regions near the image boarder.

The detection module outputs one or several bounding boxes localizing the athletes in the first frame, and then feed the box coordinates to the tracking module.

Tracking. The tracking module is built on the prevailing Siamese tracker [23, 49]. Siamese tracking architecture takes an image pair as input, comprising an exemplar image and a candidate search image. The exemplar image represents the object of interest (e.g., an image patch centered on the detected target human in the first video frame), while the search image is typically larger and represents the search area in subsequent video frames. Both inputs are processed by a backbone network with parameters. This yields two feature maps, which are cross-correlated to generate a score map. The cross correlation amounts to performing an exhaustive search of the exemplar pattern over the search image. The maximum response in the score map indicates the target position in search images.

The tracking module is fast, since it only searches the exemplar target in a local image region. However, this may cause the tracker to be failed, if the previous predictions are not precise. Therefore, we introduce a verification module to rectify the tracker when its confidence becomes low.

Verification. The verification module is a binary classification model. Its architecture is similar to [19], which takes a 107x107 patch as input and outputs two neurons indicating the probabilities of foreground target and background respectively. We update the last three convolutional layers of the network online to train a strong softmax-based classifier which can distinguish the foreground from the background effectively. Through online updating, the verification module helps the tracker tackle with various cluttered background during tracking.

Given a tracking result, if its tracking score is below a setting threshold, then it will be evaluated by the verification module. If the tracking result cannot pass through the verification, then the system will trigger detection module to re-detection the human target in the whole image. This verification and re-detection guarantee the robustness of the system.

5 POSE ESTIMATION AND ABNORMAL DETECTION

In this section, we propose a new pose estimation algorithm for sport videos. Also, we present our pose classification method. As for pose estimation, we first introduce a structural-aware convolution module, which takes account of both spatial and temporal structures of human keypoints in video sequences. Then, we present the network architecture and pipeline of our pose estimation approach.

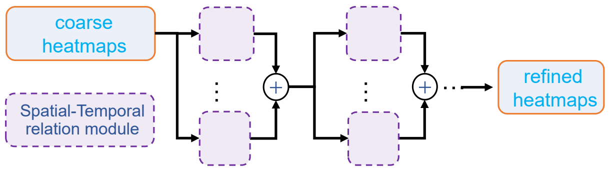


Figure 3: Fusing spatiotemporal refining results.

5.1 Spatial-temporal Relation Module

Different from previous methods [44] that utilize only one frame or two consecutive frames as a basic processing unit, our method considers a short video sequence containing T frames. To learn the spatial and temporal context relations among these T frames, we propose a new relation module and utilize it for pose feature extraction and keypoint position refinement.

Relation module. The proposed relation module is visualized in Fig. 2. It contains two branches, one branch learns a structure encoder to capture the relationship among features, the other applies a localization transform. Let Δp_x and Δp_y denote the transformation parameters. For a position (x, y) on the input feature map, it will be translated to a new position (u, v) on the output feature map according to the learned transformation. This translation is represented as $(u, v) = (x, y) + (\Delta p_x, \Delta p_y)$. Here, we set (u, v) as a grid location, thus (x, y) may be a fractional location. We employ a bilinear interpolation strategy to calculate the approximated value of the fractional location (x, y). Note that this translation is performed on feature channels, each channel has a unique encoded transformation parameter Δp. We apply this module to learn spatial and temporal context relations, and then fuse the information through feature concatenation, as shown in Fig. 2.

Spatial relation. For spatial relation, we learn a position transform on feature maps. Specifically, for a T-frame video clip, the feature map $\mathbf{f}$ has a size of $[T, C, H, W]$. Here, C, H, W represent the size of channel, height and weight, respectively. The spatial transformation is conducted on each temporal frame and repeated T time along the temporal dimension. To save computational cost in learning, we first reduce the dimension of the feature map. Then, two fully connected layers are applied to learn the position transform. The transformation parameter has a size of $[T, C, 2]$ and is applied to translate each channel. The size of output feature map is the same as the input feature map, i.e. $[T, C, H, W]$. This module learns a spatial relation over feature channels, and transfers information along feature channels, and thus enhance the feature representation

Temporal relation. For temporal relation, we learn a information transform over the feature maps of T frames. Specifically, for the input feature map with the shape of $[T, C, H, W]$, we first transpose it to the shape of $[C, T, H, W]$. Then, they are passed through group convolution operations to reduce the size of channels. Finally, a fully connected feed-forward network is used to predict the transformation weights and position offset.

Inspired by [11], we stack this module in multiple heads and layers. As shown in Fig. 3, in the same layer, outputs from multiple heads will be summed up as the input for next layer.

Table 1: The extracted features from body parts.

P$_j$	P$_i$	P$_l$	P$_k$
left hip	left shoulder	left hip	left knee
right hip	right shoulder	right hip	right knee
left hip	left knee	right hip	right knee
left knee	left ankle	right knee	right ankle
left hip	left ankle	right hip	right ankle
left knee	left hip	left knee	left ankle
right knee	right hip	right knee	right ankle
left front of snowboard	left bottom of snowboard	right front of snowboard	right bottom of snowboard

5.2 Pose Classification and Training Suggestion

Pose Classification. Here, our aim is to identify some specific gestures which we call them “bad pose”. After obtaining human keypoints, pose classification is conducted by a simple support vector machine (SVM).

In Freestyle Skiing Aerials dataset, our pose estimation algorithm results in 19 body parts in every frame that can be listed as 1) nose, 2) bottom of head, 3) top of head, 4) left shoulder, 5) left elbow, 6) left wrist, 7) left hip, 8) left knee, 9) left ankle, 10) right shoulder, 11) right elbow, 12) right wrist, 13) right hip, 14) right knee, 15) right ankle, 16) left front of snowboard, 17) left bottom of snowboard, 18) right front of snowboard, 19) right bottom of snowboard. Specifically, only 12 keypoints (including shoulders, hips, knees, ankles both in the left and right sides, and keypoints on snowboards) coordinates are used for classification. Some of the parts (such as nose, head) were discarded since they would not provide valuable information in siing tasks.

We calculated 8 angles given in Table 1 which are composed of 12 body parts. The angles between body parts were calculated by taking the cosine value of the two vectors V_1 and V_2, here V_1 and V_2 are defined as: :

$$V1 = (x_i - x_j, y_i - y_j), \quad V2 = (x_k - x_l, y_k - y_l)$$

such that (x_i, y_i), (x_j, y_j), (x_k, y_k) and (x_l, y_l) belong to 4 different body parts: P_i, P_j, P_k and P_l (as shown in Table 1).

In skiing contest, some gestures will be considered as bad pose, like bending the hips, crossed snowboards and bending the knees, etc. In order to obtain the ability of distinguishing bad pose and good pose in skiing sport. We train and test our model in skiing dataset. Qualitative results are shown in Fig. 4.

Training Suggestion. Once finished pose classification, we got some bad pose frames in a video. In order to offer some good advice for athletes, one video clip with correct poses will be provided by our system. These clips are sampled from some standard sports videos with the same action of bad frames. As Fig. 1 shown, a training suggestion will be provided based the bad pose recognized by classification model.

6 EXPERIMENTS AND APPLICATIONS

Experiments has been conducted on sport video dataset to evaluate our tracking method. To demonstrate the effectiveness of our proposed pose estimation model, we conduct extensive experiments on two open video-based pose estimation benchmarks. Furthermore,

Table 2: Evaluation on VOT2018-LT dataset.

Tracker	F-score	Precision	Recall
SiamFC [3]	0.433	0.636	0.328
PTAV [7]	0.481	0.595	0.404
SiamRPN [23]	0.546	0.574	0.521
Ours	**0.610**	**0.646**	**0.578**

Table 3: Evaluation on sport video dataset.

Tracker	F-score	Precision	Recall
SiamRPN [23]	0.627	0.635	0.620
Ours	**0.710**	**0.733**	**0.695**

we also collect sport video datasets, namely Freestyle Skiing Aerials, to show the function of our AI Coach system to help athletes improve performance.

6.1 Experiments on Tracking

In this subsection, we evaluate the tracking component of our system on two long-term tracking datasets. One is the widely-used public VOT2018-LT [22] dataset, the other is a sport video dataset. The sport video dataset is collected from LaSOT [6], which is one of current largest tracking benchmarks. We collect 20 sport videos, including playing basketball, uneven bars, etc. We compare our proposed tracker with existing state-of-the-art tracking methods, including SiamFC [3], SiamRPN [23] and PTAV [7]. Here, SiamRPN can be regarded as the baseline, since it is one component used in our tracking system.

Table 2 shows the results evaluated on VOT-LT benchmark. We adopt the standard Precision, Recall and F-score, which are defined as that the prediction matches the ground truth if the overlap exceeds a threshold. The metrics are proposed by Alan Lukezic et al. [30], and used in VOT-18 challenge.

We can observe that our method achieves the best performance among the compared trackers. The underlying reason is that the introduced verification and re-detection modules reduce the tracking failures, while enhancing the prediction precision of target position. Moreover, the proposed tracking component runs at a real-time speed, i.e. ∼ 30 frames per second on a GeForce GTX 1080 GPU. Table 3 presents the results evaluated on sport videos. We can see that our method achieves the best performance, because that our tracker is specifically designed for human-centric sport videos.

6.2 Experiments on Pose Estimation

To fairly compare with existing pose estimation model, we conduct comparison experiments with state-of-the-art image-based and video-based pose estimation model [31, 44] on two video-based pose estimation benchmarks, Penn Action [48] and sub-JHMDB [16]. We also present extensive ablation studies of our proposed model to analyze the effects of each component.

Datasets. Two video pose estimation benchmarks are used in our experiments. Penn Action Dataset is a widely used dataset for single-person pose estimation in video. It contains 2326 video clips with 163841 frames in total. The annotations consist of 13 keypoints and the occlusion label (visible or not) of each keypoint. To fairly compare with existing work, evaluation is only conducted on visible keypoints. We use the standard training and testing split, in which 87543 frames are used for training and 76298 frames are used for testing. Sub-JHMDB dataset contains 316 video clips with 11200 frames in total. Each person are annotated with 15 keypoints. We use the standard 3-fold cross validation setting provided by the dataset to do all evaluations.

Table 4: Comparisons of results on sub-JHMDB dataset using PCK@0.2, which have resnet-50 (*) as backbone.

Method	Head	Sho	Elb	Wri	Hip	Knee	Ank	Mean
CPM [42]	98.4	94.7	85.5	81.7	97.9	94.9	90.3	91.9
RPM [2]	98.0	95.5	86.9	82.9	97.9	94.9	89.7	92.2
LSTM PM [31]	98.2	96.5	89.6	86.0	**98.7**	95.6	90.9	93.6
Resnet-50* [44]	97.8	96.1	88.8	84.7	98.0	95.6	91.5	93.2
LSTM-PM*[31]	98.0	97.2	89.9	86.0	98.2	96.2	92.8	94.0
STRN* (Ours)	**98.8**	**98.0**	**91.4**	**86.5**	98.6	**97.4**	**93.3**	**94.9**

Implementation Details. For the data augmentation, all images are randomly scaled by a factor. For Penn, this factor is in the range of [0.8, 1.4], while for sub-JHMDB it is in the range of [1.2, 1.8]. Rotation with degree in the range of $[-40°, 40°]$ and random flipping are also utilized. At last, all the images are resized to a fixed size (384×288) with bodies located at center. Initial weights of model are loaded from pre-trained backbone. During training, the initial learning rate is 8e-5. We train base model and our model with both 150 epochs. Batch size is 2, the network ingests 2 video clips with temporal resolution of 10. Intermediate supervision is implemented such that at the end of each head or layer, we have loss to be computed instead of just at the end. It is widely used to prevent vanishing gradients.

Evaluation Metrics. For consistent comparison with previous work [31, 38], the metric PCK@0.2 is used, which means a prediction is considered correct if it lies within $(\alpha = 0.2) \times max(h, w)$, where h and w presents the height and width of bounding box. Following previous works, we use 13 human keypoints to train and evaluate our methods.

Comparison Experiments. Table 5 and Table 4 show the performance of our models and previous works on Penn dataset as well as sub-JHMDB dataset. For sub-JHMDB, T is set as 10. The best result is achieved by stacking our modules with 3 layers and 2 heads. While 2 layers and 2 heads are used for experiments on Penn dataset, T is set to 5.

We first conducted experiments on widely used simple baseline model[44] in recent pose estimation work. LSTM pose machine [31] achieved state-of-art results on video pose estimation benchmarks. Since its backbone network is shallower, the same backbone (Resnet-50) and training setting are used for fair comparison.

Ablation study. Here we will discuss the contribution of spatial refinement and temporal refinement. How to choose T and the number of head and layer. All of the experiments are conducted on sub-JHMDB split1 dataset. The experiments in Table 6 and Table 7 are conducted on 3 layers, 2 heads model.

Contribution of ST Module. In Table 6 we study the effect of spatial blocks, temporal blocks. For example, in the space-only version, the temporal blocks will not be trained, the output of each ST-block only consists of spatial refined heatmaps. That means

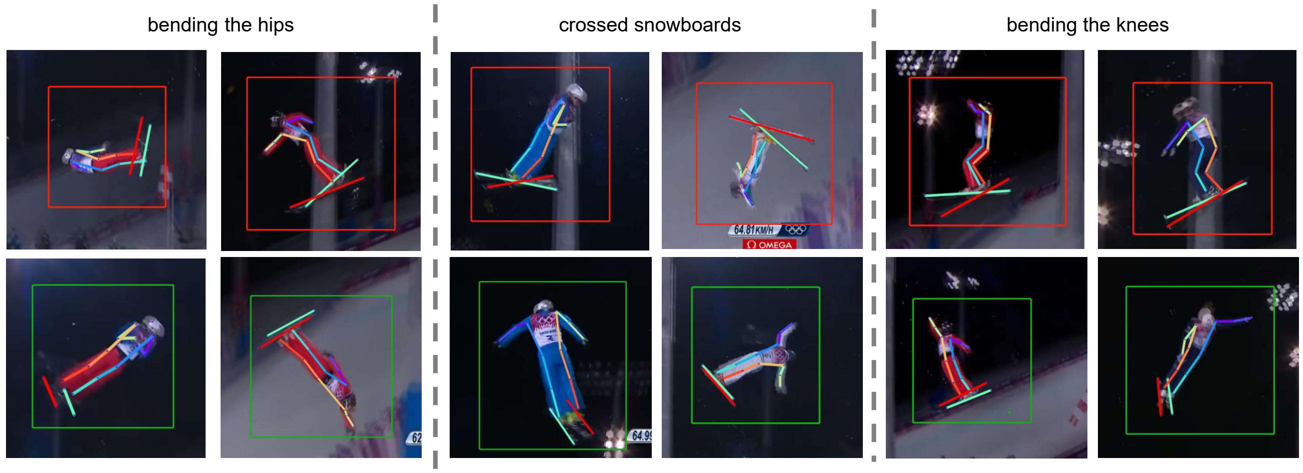


Figure 4: Examples for detected good and bad pose based on keypoints-based feature. Here we show three categories of so called "bad poses" defined in ski technique: bending the hips, crossed snowboards and bending the knees. The detected bad poses are marked in red while correct pose are marked in green.

Table 5: Comparisons of results on Penn Action dataset using PCK@0.2, which have resnet-50 (*) as backbone.

Method	Head	Sho	Elb	Wri	Hip	Knee	Ank	Mean
CPM [42]	98.6	97.9	95.9	95.8	98.1	97.3	96.6	97.1
RPM [2]	98.5	98.2	95.6	95.1	97.4	97.5	96.8	97.0
LSTM PM [31]	98.9	98.6	96.6	96.6	98.2	98.2	97.5	97.7
Resnet-50* [44]	99.1	99.3	97.6	97.1	99.3	99.1	98.5	98.6
LSTM-PM*[31]	99.2	99.4	98.0	97.7	99.4	99.1	98.3	98.7
STRN* (Ours)	**99.5**	**99.5**	**98.6**	**98.4**	**99.5**	**99.4**	**98.6**	**99.1**

Table 6: Contribution of ST Module

Method	Head	Sho	Elb	Wri	Hip	Knee	Ank	Mean
Resnet-50 [44]	98.0	95.6	89.1	85.0	98.6	95.6	91.1	93.3
STRN-S (Ours)	97.9	97.3	91.2	86.2	**99.2**	97.4	92.6	94.6
STRN-T (Ours)	98.4	**97.6**	91.1	86.1	**99.2**	97.3	92.5	94.6
STRN-ST (Ours)	**98.6**	**97.6**	**92.5**	**87.8**	98.8	**97.5**	**93.0**	**95.1**

we only consider influences come from coarse heatmaps in the same frame. The time-only version can be set up similarly. Table 6 shows that both the space-only and time-only versions improve our Resnet-50 baseline, but are inferior to the ST combined version.

Analysis of different T. As we present in Sec. 2.2, in our model, features in distant frame can directly contributing to each other. Table 7 shows that with the growth of T, we can gain improvement. However, results tend to be saturated with the increase of T (from 10 to 15). It makes sense since appearances in far away frames may very different from current timestamp, their weights of impact will reduce with increase of T.

Contribution of Multi-stage. Stacking multiple blocks can significantly improve mAP as shown in Table 8 since under iterative refinements, we can gain better results.

Table 7: Effects of the size of video clips *T*

Method	T	Head	Sho	Elb	Wri	Hip	Knee	Ank	Mean
Resnet-50 [44]	-	98.0	95.6	89.1	85.0	98.6	95.6	91.1	93.3
STRN (Ours)	5	97.9	97.1	91.6	86.9	99.0	97.3	93.5	94.8
STRN (Ours)	10	**98.6**	**97.6**	**92.5**	**87.8**	98.8	**97.5**	93.0	**95.1**
STRN (Ours)	15	98.5	97.4	92.0	87.1	**99.1**	97.2	**93.7**	95.0

Table 8: Contribution of multi-heads and multi-layers

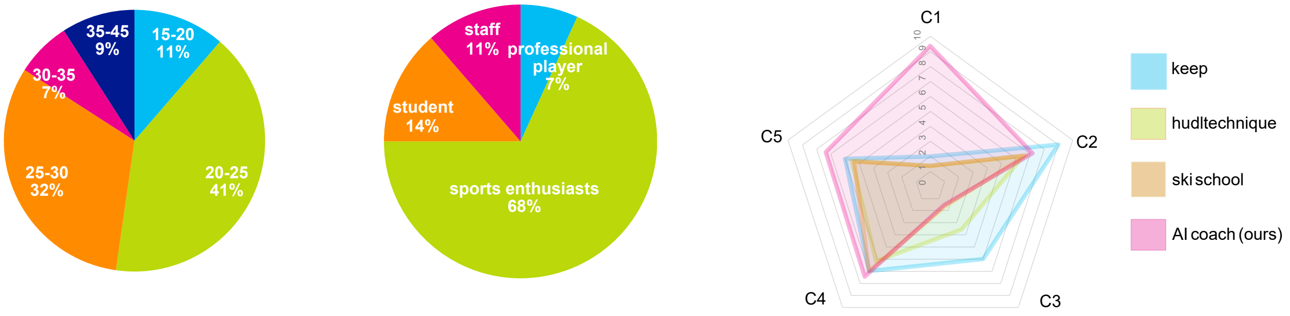
Method	L	H	Head	Sho	Elb	Wri	Hip	Knee	Ank	Mean
Resnet-50 [44]	-	-	98.0	95.6	89.1	85.0	98.6	95.6	91.1	93.3
STRN (Ours)	1	1	98.2	**97.7**	91.1	87.3	**99.1**	97.4	92.7	94.8
STRN (Ours)	3	1	97.8	97.6	92.3	86.8	98.8	**97.7**	**93.0**	94.9
STRN (Ours)	3	2	**98.6**	97.6	**92.5**	**87.8**	98.8	97.5	**93.0**	**95.1**

6.3 Experiments on Pose Classification

To evaluate the function of our proposed AI Coach system, we collected Freestyle Skiing Aerials dataset to evaluate accuracy of pose classification.

Dataset. We invited 30 skiing enthusiasts to provide skiing training videos as well as feedbacks. These 30 enthusiasts comes from different colleges, with the age in the range of (20, 30). Half of them are males and half are females. With the help of them, we build dataset namely Freestyle Skiing Aerials dataset (63 clips, 51 clips for training, 12 clips for validation) to conduct experiments. Every frame in this dataset is annotated with 3 binary labels: whether the athlete bend the hips, crossed snowboards or bend the knees.

Paramater Setting. For the analysis using SVM, the radial basis kernel function (RBF) was used. As kernel parameters C was set as 2. For every error types, binary classification is applied on it independently.



(a) User age distribution

(b) User career distribution

(c) Comparative analysis with other sports apps
(the mean of the results)

Figure 5: User distribution and results of user study.

Table 9: Results for Pose Classification

Metrics	bending the hips	crossed snowboards	bending the knees
Precision	83.7	36.6	77.7
Recall	83.2	18.8	75.2
F1	83.4	24.8	76.4

Evaluation Metrics. As the evaluation metric, the standard precision and recall are calculated for every class.

$$Precision = (\frac{TP}{TP + FP}), \quad Recall = (\frac{TP}{TP + FN})$$

$$F1 = (\frac{2 \times Precision \times Recall}{Precision + Recall})$$

where TP, FP and FN indicate true positives, false positives and false negatives, respectively.

As shown in Fig. 4, detected keypoints provide strong cues to distinct bad (a) or good (b) poses to improve training efficiency since the body structure bring fine-grained distinguishing feature for abnormal pose detection. The results (Table 9) showed that applying our pose estimation method on Freestyle Skiing Aerials dataset obtain good performance on detecting bad poses. However, it is hard to detect errors with crossed snowboards since they always mixed with the background which results in failed to detect keypoints on snowboards.

6.4 Evaluation of Usability

We invited 44 volunteers. All of them are in the age of (15,45). The age distribution and occupational distribution are shown in Fig. 5 (a) and (b), respectively. We conducted experiments on three famous sports apps:

- Keep[1]. Keep arranges training plans according to the user's scenario, fitness purpose, etc.
- Hudltechnique[2]. Hudltechnique replays sport videos in multiple slow motion speeds and provides video analysis.
- Ski school[3]. Ski school provides high quality video tips taken from their full series of ski school lessons.

During the process, we conducted interviews and shared findings by radar chart to evaluate our system. Participants were asked to use a 10-point scale (10 means the most satisfied, higher is better) to evaluate our system according to the following 5 criteria:

- C1: Whether the software can automatically detect athletes' errors and provide suggestions conditioned on these errors?
- C2: Interface friendliness.
- C3: Does it support multiple sports?
- C4: The quality of provided corresponding video lessons.
- C5: Overall score of the app.

Results can be found in Fig. 5 (c), here we calculate the mean score for each app in each aspect. The superior results show the effectiveness of our system in evaluating and improving ski technique.

7 CONCLUSION

In this paper, we create an AI Coach system to provide personalized athletic training experiences for posture-wise sports activities. To relieve the challenges of fast movement and complex actions, we propose to design the system with several distinct features:(1) trajectory extraction for a single human instance by leveraging deep visual tracking, (2) human pose estimation by proposing spatial-temporal relation network, (3) bad pose detection and exemplar-based visual suggestions. Extensive experiments demonstrate the effectiveness of our models and our proposed AI Coach system. In the future, beyond just the humans in one scene, we will explore pose estimation with 3D information in the wild.

8 ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China under Grants (61831015) and also by the National Key Research and Development Program of China (No. 2016YFB1001501). The authors would like to thank Yiran Li, Peijun Xia, Bo Wang, Youwen Hu, Summer Wang and Yuanchun Xu for promoting the technology transfer of this research to Microsoft XiaoIce.

[1] https://www.gotokeep.com/

[2] https://www.25pp.com/ios/detail$_5$20812/

[3] https://apps.apple.com/gb/app/ski-school-lite/id396879036

REFERENCES

[1] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. 2009. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*.

[2] Vasileios Belagiannis and Andrew Zisserman. 2016. Recurrent Human Pose Estimation. In *IEEE International Conference on Automatic Face Gesture Recognition*.

[3] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. 2016. Fully-convolutional siamese networks for object tracking. In *ECCV*.

[4] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*.

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. 2018. Cascaded pyramid network for multi-person pose estimation. In *CVPR*.

[6] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. 2019. Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*.

[7] Heng Fan and Haibin Ling. 2017. Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking. In *ICCV*.

[8] Pedro F Felzenszwalb and Daniel P Huttenlocher. 2005. Pictorial structures for object recognition. *IJCV* 61, 1 (2005), 55–79.

[9] Jianlong Fu, Yue Wu, Tao Mei, Jinqiao Wang, Hanqing Lu, and Yong Rui. 2015. Relaxing from vocabulary: Robust weakly-supervised deep learning for vocabulary-free image tagging. In *ICCV*.

[10] Jianlong Fu, Heliang Zheng, and Tao Mei. 2017. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *CVPR*.

[11] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. 2019. Video action transformer network. In *CVPR*.

[12] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. 2018. Detect-and-Track: Efficient Pose Estimation in Videos. In *CVPR*.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *ICCV*.

[15] J. F. Henriques, R Caseiro, P Martins, and J Batista. 2015. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Transactions on Pattern Analysis Machine Intelligence* 37, 3 (2015), 583–596.

[16] Hueihan Jhuang, Juergen Gall, Silvia Zuffi, Cordelia Schmid, and Michael J Black. 2013. Towards understanding action recognition. In *ICCV*.

[17] Sam Johnson and Mark Everingham. 2010. Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation.. In *BMVC*.

[18] Dhiraj Joshi, Michele Merler, Quoc-Bao Nguyen, Stephen Hammer, John Kent, John R Smith, and Rogerio S Feris. 2017. Ibm high-five: Highlights from intelligent video engine. In *Proceedings of the 25th ACM international conference on Multimedia*. ACM, 1249–1250.

[19] Ilchae Jung, Jeany Son, Mooyeol Baek, and Bohyung Han. 2018. Real-time mdnet. In *ECCV*.

[20] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. 2018. Multi-scale structure-aware network for human pose estimation. In *ECCV*.

[21] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. 2011. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR*.

[22] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, and Zhiqun He. 2017. The Visual Object Tracking VOT2017 Challenge Results. In *IEEE International Conference on Computer Vision Workshop*.

[23] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. 2018. High performance visual tracking with siamese region proposal network. In *CVPR*.

[24] Yang Li, Zhan Xu, and Jianke Zhu. 2017. CFNN: Correlation Filter Neural Network for Visual Object Tracking. In *IJCAI*.

[25] Yang Li and Jianke Zhu. 2014. A scale adaptive kernel correlation filter tracker with feature integration. In *ECCV Workshops*.

[26] Yang Li, Jianke Zhu, and Steven CH Hoi. 2015. Reliable patch trackers: Robust visual tracking by exploiting reliable patches. In *CVPR*.

[27] Yang Li, Jianke Zhu, Steven C.H. Hoi, Wenjie Song, Zhefeng Wang, and Hantang Liu. 2019. Robust Estimation of Similarity Transformation for Visual Object Tracking. In *AAAI*.

[28] Qingshan Liu, Zhigang Hua, Cunxun Zang, Xiaofeng Tong, and Hanqing Lu. 2005. Providing on-demand sports video to mobile devices. In *Proceedings of the 13th annual ACM international conference on Multimedia*. ACM, 347–350.

[29] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. 2016. Ssd: Single shot multibox detector. In *ECCV*.

[30] Alan Lukežič, Luka Čehovin Zajc, Tomáš Vojíř, Jiří Matas, and Matej Kristan. 2018. Now you see me: evaluating performance in long-term visual tracking. *arXiv preprint arXiv:1804.07056* (2018).

[31] Yue Luo, Jimmy Ren, Zhouxia Wang, Wenxiu Sun, Jinshan Pan, Jianbo Liu, Jiahao Pang, and Liang Lin. 2018. LSTM pose machines. In *CVPR*.

[32] Alejandro Newell, Zhiao Huang, and Jia Deng. 2017. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*. 2277–2287.

[33] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. 2017. Towards accurate multi-person pose estimation in the wild. In *CVPR*.

[34] Leonid Pishchulin, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. 2013. Poselet conditioned pictorial structures. In *CVPR*.

[35] Deva Ramanan. 2007. Learning to parse images of articulated bodies. In *Advances in neural information processing systems*. 1129–1136.

[36] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *CVPR*.

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NeurIPS*.

[38] Jie Song, Limin Wang, Luc Van Gool, and Otmar Hilliges. 2017. Thin-slicing network: A deep structured model for pose estimation in videos. In *CVPR*.

[39] Yuandong Tian, C Lawrence Zitnick, and Srinivasa G Narasimhan. 2012. Exploring the spatial hierarchy of mixture models for human pose estimation. In *ECCV*.

[40] Xiaofeng Tong, Qingshan Liu, Yifan Zhang, and Hanqing Lu. 2005. Highlight ranking for sports video browsing. In *Proceedings of the 13th annual ACM international conference on Multimedia*. ACM, 519–522.

[41] Jack Valmadre, Luca Bertinetto, João Henriques, Andrea Vedaldi, and Philip HS Torr. 2017. End-to-end representation learning for correlation filter based tracking. In *CVPR*.

[42] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. 2016. Convolutional pose machines. In *CVPR*.

[43] Barry D Wilson. 2008. Development in video technology for coaching. *Sports Technology* 1, 1 (2008), 34–40.

[44] Bin Xiao, Haiping Wu, and Yichen Wei. 2018. Simple baselines for human pose estimation and tracking. In *ECCV*.

[45] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. 2017. Learning feature pyramids for human pose estimation. In *ICCV*.

[46] Yi Yang and Deva Ramanan. 2013. Articulated human detection with flexible mixtures of parts. *IEEE transactions on pattern analysis and machine intelligence* 35, 12 (2013), 2878–2890.

[47] Dan Zecha, Christian Eggert, Moritz Einfalt, Stephan Brehm, and Rainer Lienhart. 2018. 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*. ACM, 11–19.

[48] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. 2013. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*.

[49] Zhipeng Zhang and Houwen Peng. 2019. Deeper and Wider Siamese Networks for Real-Time Visual Tracking. In *CVPR*.