# Pose-guided Action Quality Assessment

PuShi
2021233307
pushi@shanghaitech.edu.cn

Yufan Feng
2019533141
fengyf2@shanghaitech.edu.cn

Bin Yang
2019533230
yangbin@shanghaitech.edu.cn

## Abstract

*The action quality assessment(AQA) technology has been widely used in athletic events in order to ensure the fairness of scoring. In traditional sports, such as diving, gymnastics and other sports, scoring usually contains a lot of subjective factors. Even calculating the average score of multiple judges is difficult to eliminate the artificial influence. The AI scoring system can reduce the controversies in many subjective scoring events. In this project, we are going to finish AQA task by a pose-guided pipeline, in which we split the task into two subtasks including pose estimation and pose-guided AQA.*

## 1. Introduction

### 1.1. Dataset

The *FineDiving*[6] dataset is the first fine-grained AQA dataset. The dataset they built, FineDiving (short for fine-grained grained Diving), focused on all kinds of Diving events and was the first fine-grained video data set for AQA. FineDiving contains the following features :(1) two layers of semantic structure. All videos are semantically annotated at two levels, action Type and sub-action Type. Among them, different action types are generated by the combination of different sub-action types. (2) Two-layer sequential structure. The action instances in each video are time-bounded and decomposed into successive steps according to a defined dictionary; (3) Official diving score, referee score and difficulty coefficient from FINa. Based on FineDiving, they further proposed a procedure-aware based AQA method to evaluate the quality of actions. The proposed framework constructs a new Temporal Segmentation Attention (TSA) perceptual embedding of learning process to achieve reliable scoring with better interpretability.

An example is shown in Figure1. The action types are shown on the top line, which are named with numbers and capital letters. Each action type is divided into several sub-action types, including somersaults pike, twist, entry and so on. There are 3000 video samples, covering 52 action types,

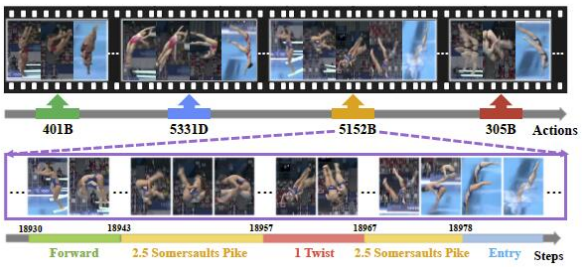



Figure 1. Example of FineDiving

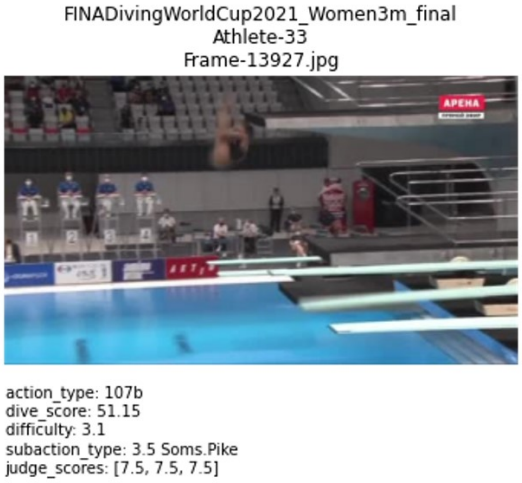
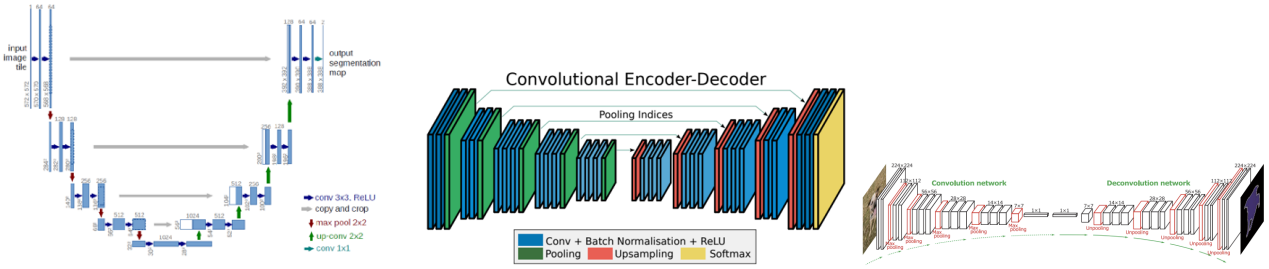



Figure 2. Our dataloader

29 sub-action types, and 23 difficulty degree types in total. One example of our dataloader is shown in figure2

### 1.2. Pose Estimation

Human Pose Estimation is an important task in computer vision. It is also an essential step for computers to understand Human actions and behaviors. In recent years, human pose estimation methods using deep learning have been proposed, and have achieved far better performance than traditional methods. In practice, the estimation of human body posture is often transformed into the prediction of human




(a) (b) (c)

Figure 3. (a)UNet (b)SegNet (c)DeconvNet

body key points, that is, the position coordinates of each key point of human body are predicted first, and then the spatial position relations between the key points are determined according to the prior knowledge, so as to get the predicted human skeleton.

For 2D pose estimation, most of the current research is multi-person pose estimation, that is, each image may contain more than one person. There are two approaches to solve this problem: top-down and bottom-up.

- Top-down: detect the target in the image and find out all the people, then crop out the person from the original image and input it into the network for attitude estimation after resize. In other words, top-down is to transform the problem of multi-person pose estimation into the problem of multi-person pose estimation.
- Bottom-up: find all the key points in the picture first and then group the key points to get an individual.

In general, top-down methods have higher accuracy and bottom-up methods have higher speed. In our project, we build a network for pose estimation based on HRNet[5], which belongs to top-down structure.

### 1.3. Pose-guided AQA

AQA evaluates the execution quality of an action by analyzing the performance of the action in the video. Unlike traditional motion recognition, AQA is more challenging: motion recognition can identify an action from one or several images, but AQA requires traversing the entire motion sequence to assess the quality of the action. Most existing AQA methods rely on the depth features of the video to regression the quality scores of different movements. However, it is difficult to assess the quality of different movements with little difference in a similar context. Diving competitions, for example, are usually filmed at an aquatic center, and all athletes perform the same routine: takeoff, flight, and entry into the water. The nuances of these routines are mainly in the number of twists and turns, the position in the air, and the water entry (e.g. splash size). Capturing these subtle differences requires an AQA approach that

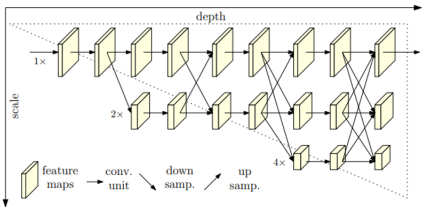



Figure 4. HRNet structure

can not only parse the steps of a dive, but also explicitly quantify the quality of execution of those steps.

In this part, we finished a AQA network based on the pose keypoints estimated in the previous task. However, we failed to get the results we wanted and the loss seems hard to converge. The details of models and experiments and the possible reasons will be discussed in the subsequent sections.

## 2. Methods

### 2.1. Pose Estimation

Most of the approaches proposed in previous articles are series architectures, where the network consists of one or more downsampling-upsampling architectures.

Three examples of the series architectures models are shown in figure3 including UNet[4], SegNet[1] and DeconvNet[2]. What these models have in common is that they first use convolution kernel to downsample the image to low resolution and then restore it to high resolution. The main problem with serial networks is that much of the information of high resolution images will be lost in the process of downsampling to low resolution images.

HRNet uses a parallel network architecture which is shown in figure4. It is able to maintain a high-resolution presentation throughout the process. We started from the high resolution subnet as the first stage, gradually added the high resolution subnets to the low resolution subnetworks, formed more stages, and connected the multi-resolution subnets in parallel. During the whole process, we repeatedly exchange information on parallel multi-resolution sub-

networks to perform multi-scale repeated fusion. Finally, We estimate the keypoints through high-resolution representation of the network output. The advantage of parallel network is that the image information can be better preserved by maintaining the resolution representation.

Moreover, we use DARK[8](the short of Distribution-Aware coordinate Representation of Keypoint) to reduce information loss of heat map coordinate transformation. Coordinate coding and decoding is a small but important part of pose estimation, AP of the HRNet model with DARK is increased by 3% - 4%.

### 2.2. Pose-guided AQA

We implemented an AQA model based on the keypoints of the previous problem. Inspired by the FineDiving[6] and CoRe[7], our model structure was designed as figure5. The main part of AQA based on a transformer decoder, which needs exemplar features as cross attention values. As for input feature tokens, We add an empty token after all frames then predict the final offset using the additional token. We assume that fine-grained labels can help model understand frame poses better. To this end, we apply a shared MLP on the rest tokens to predict frame label during training.

As we said in the previous sections, it's hard to do simple regression with source video and target score as they are always very similar. The reasonable method is to compare the tiny details between query and exemplar to predict the offset of the score.

In training part, we randomly choose an exemplar for each query and put them together into the Pose detector. The detected poses will be send into a transformer decoder, where the exemplar is used as *key* and *value*. The weights-shared MLP will classify the poses frame by frame in fine-grained level, and the offset score will be predicted by a new MLP. As what we get is the score difference between the query and the exemplar, we should add the offset score to the exemplar to get the predicted score. In testing part, we randomly choose 10 exemplars for each query and take the average score as the final result.

## 3. Results

### 3.1. Pose Estimation

Some examples of our pose estimation results are shown in figure6. While there might be multiple persons in one frame, we first adopt a detection net[3] to identify all bounding boxes with person. On choosing the largest bounding box, we predict human pose in this single box. We use DARK[8] pretrained on coco wholebody model for visualizing and will include another try in AQA task. We'll get 133 keypoints for each image and most of them are settled on the face. As we can see that the model worked well on these images and so as most of the others. However, there are some exceptions as shown in figure7. In image(a) the detected pose was wrong and in image(b) we only detected the people in the audience because the player's body is blurred. As for image(c) and (d), we only detected one pose for two players as they stand so close.

| Model | Spearmanr | R-L2 |
|---|---|---|
| Base model (coco wholebody) | 0.043 | 0.025 |
| Base model (coco) | 0.051 | 0.023 |
| Base model (efficientnet feature extractor) | 0.046 | 0.023 |
| w/o exemplar | 0.024 | 0.029 |
| with fine-grained classification | 0.053 | 0.023 |

Table 1. Results of AQA

### 3.2. Pose-guided AQA

We adopt the same transformer structure as [6]. The transformer layer has 8 multi-heads and the decoder is stacked by 6 transformer layers. We use Adam optimizer with learning rate set to $10^{-3}$, the first and second momentum set to 0.9 and 0.999, weight decay set to $10^{-4}$.

The results are shown in table1. We modified our model for several times against the following variants:

- Base model (coco wholebody): DARK[8] as the pose estimation net is pretrained on coco wholebody dataset and outputs 133 keypoints for one bbox per frame as results.
- Base model (coco): HRNet[5] as the pose estimation net is pretrained on coco dataset and outputs 17 keypoints for one bbox per frame as results. As we guess 133 keypoints could include much noise and more inaccuracies, we deduced the number of keypoints and trying to help model focus on key information.
- Base model (efficientnet feature extractor): the pose estimation net is replaced by a pretrained efficientnet feature extractor, which outputs a 2048 dimensional feature for each frame. This experiments was to explore that if rich image information could be absorb by our model. Additionally, we noticed that FineDiving[6] also used rich image feature encoding instead of pose keypoints as transformer input.
- w/o exemplar: a transformer encoder replace the original decoder to make query features as the only input. Instead of regressing offset based on exemplar, we directly regress the score as final output.
- with fine-grained classification: predicting frame labels using shared MLP.

As shown in the table, we failed to gain a desired result though strove hard. The values of Spearman's rank correlation are low, which means that our model didn't learn the

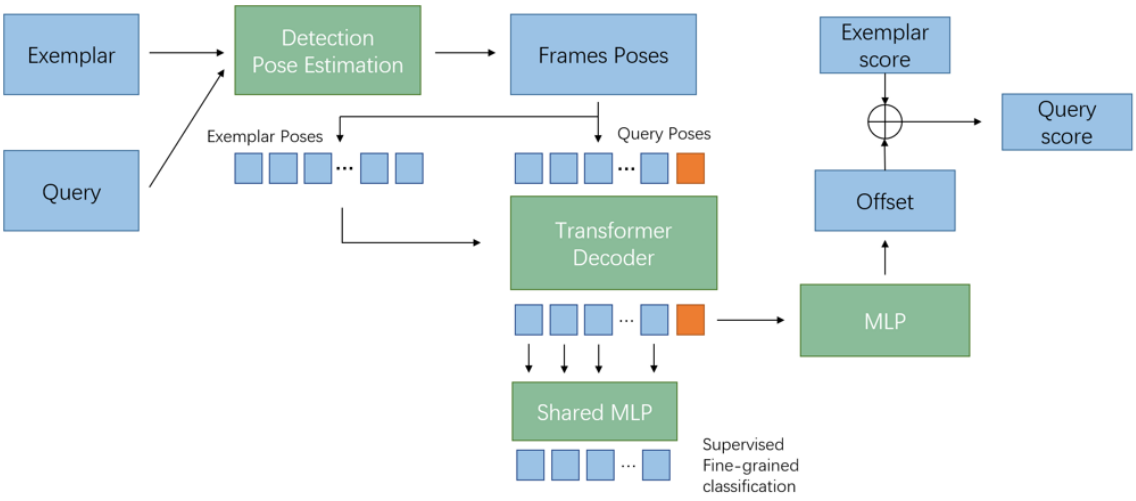
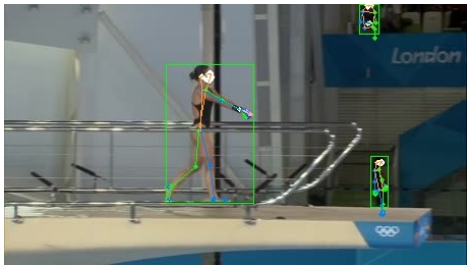
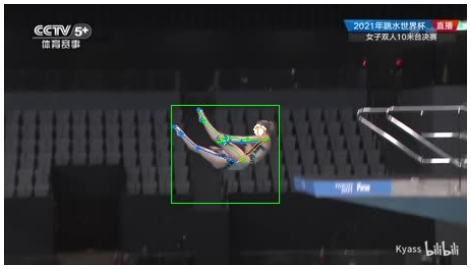



Figure 5. Our model



(a)



(b)

Figure 6. Pose estimation results

information of the videos. Also, as we randomly choose exemplar from training set, we find that exemplar would affect the test results significantly, which makes our results even more unreliable.

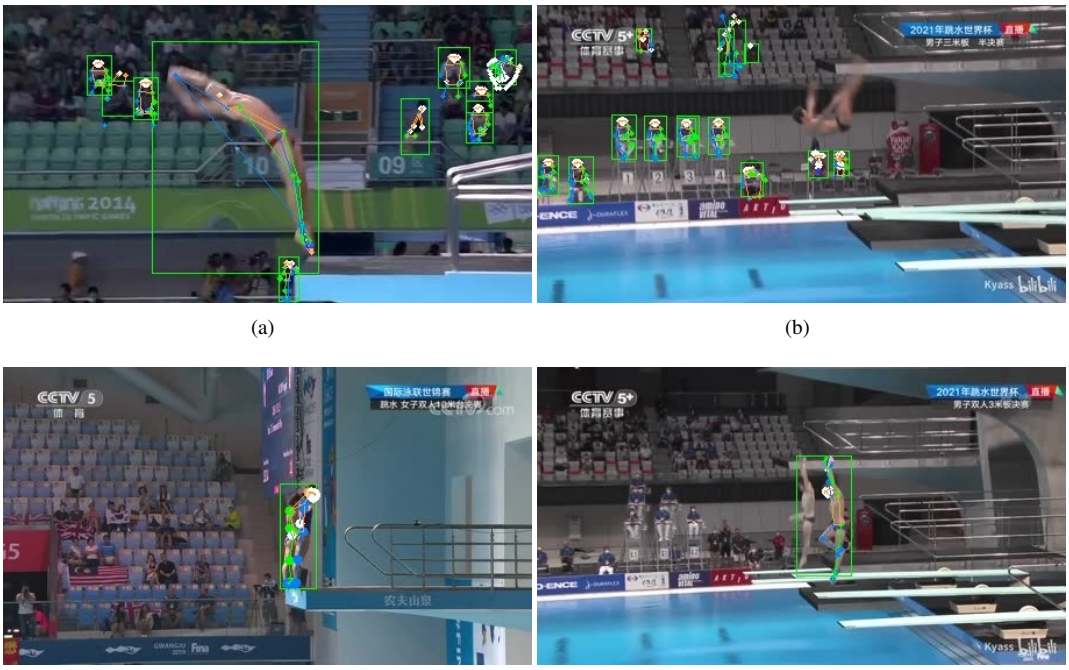
## 4. Conclusion and Analysis

We tracked our loss item during training(Figure 8). We can see that loss is not decreasing, which suggests our model does not converge. This could be our main problem. However, lots of reasons could cause this problem. The bad performance of our models may be caused by the following reasons:

1. As said before, there are some bad samples in the pose estimation results, which directly affects the effect of training. However, as the bad samples are in the minority, this may not be the main reason.

2. We used a single transformer decoder as our AQA backbone. This model could be too naive to fit our task as we did not tune any compute structure or include fancy training tricks. Comparing with FineDiving model structure[6], we abandoned I3D structure to deal with time stream, as we expect the transformer decoder could find out the attention weight between tokens, which can theoretically represent the relationship between frames. However, when this black-box transformer decoder failed to met our expectation, it could be hard to localize the problem. What's more, using last token as summarized feature of previous tokens is quite rough and simple.

3. The last possible reason is that we randomly choose the exemplar. Both the fluctuating losses during training and the test results highly depended on exemplar prove that our model does not understand the relationship between random exemplars and queries. But as we can see in the table1, we got the worst result without exemplar. It means that the exemplar does make a difference.



(a) (b)

(c) (d)

Figure 7. Bad examples

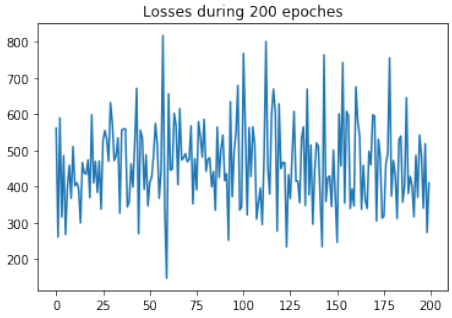



Figure 8. losses during training

## References


[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[2] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[3] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[5] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703, 2019.

[6] Jinglin Xu, Yongming Rao, Xumin Yu, Guangyi Chen, Jie Zhou, and Jiwen Lu. Finediving: A fine-grained dataset for procedure-aware action quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2949–2958, 2022.

[7] Xumin Yu, Yongming Rao, Wenliang Zhao, Jiwen Lu, and Jie Zhou. Group-aware contrastive regression for action quality assessment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7919–7928, 2021.

[8] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7093–7102, 2020.