Towards Improved and Interpretable Action Quality Assessment with Self-Supervised Alignment

Konstantinos Roditakis
Institute of Computer Science,
FORTH
Greece
Computer Science Department,
University of Crete
Greece
croditak@ics.forth.gr

Alexandros Makris
Institute of Computer Science,
FORTH
Greece
amakris@ics.forth.gr

Antonis Argyros
Institute of Computer Science,
FORTH
Greece
Computer Science Department,
University of Crete
Greece
argyros@ics.forth.gr

ABSTRACT

Action Quality Assessment (AQA) is a video understanding task aiming at the quantification of the execution quality of an action. One of the main challenges in relevant, deep learning-based approaches is the collection of training data annotated by experts. Current methods perform fine-tuning on pre-trained backbone models and aim to improve performance by modeling the subjects and the scene. In this work, we consider embeddings extracted using a self-supervised training method based on a differential cycle consistency loss between sequences of actions. These are shown to improve the state-of-the-art without the need for additional annotations or scene modeling. The same embeddings are also used to temporally align the sequences prior to quality assessment which further increases the accuracy, provides robustness to variance in execution speed and enables us to provide fine-grained interpretability of the assessment score. The experimental evaluation of the method on the MTL-AQA dataset demonstrates significant accuracy gain compared to the state-of-the-art baselines, which grows even more when the action execution sequences are not well aligned.

CCS CONCEPTS

• **Computing methodologies → Activity recognition and understanding**.

KEYWORDS

computer vision, video understanding, action quality assessment, video alignment

ACM Reference Format:
Konstantinos Roditakis, Alexandros Makris, and Antonis Argyros. 2021. Towards Improved and Interpretable Action Quality Assessment with Self-Supervised Alignment. In *The 14th PErvasive Technologies Related to Assistive Environments Conference (PETRA 2021), June 29-July 2, 2021, Corfu, Greece.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3453892.3461624

1 INTRODUCTION

Action Quality Assessment entails the evaluation of the execution of performed actions. It is a significant problem with applications in many domains, including healthcare (e.g., evaluate the progress of patients in performing certain rehabilitation tasks or train the personnel for performing medical procedures) and sports (e.g., to help athletes improve their performance).

The problem has recently attracted the interest of the computer vision community. Current approaches use either image or pose features or a combination of both to tackle the problem. The problem is treated as a regression or classification problem. In the former case, the action quality score is regressed, while in the latter, pairwise action execution comparisons are performed. In both cases, one significant challenge is the collection and utilization of data that domain experts annotate.

In this work, we propose a novel action quality assessment method that employs self-supervised features and video alignment. We consider a self-supervised training method that minimizes a differential cycle consistency loss between sequences of actions [5]. The resulting embeddings complement the commonly used appearance features (I3D) [2] and improve the performance without the need for additional annotations, scene modeling, or pose estimation. We also use the self-supervised embeddings to align the sequences before assessing the action execution quality. The goal of alignment is twofold: (a) to increase the accuracy of the quality assessment and (b) to allow for fine-grained interpretability at the frame level. To account for the inherent uncertainty of the action quality score (i.e., different judges may assign different scores), we regress the score distribution instead of a single value.

Video representations that rely on self-supervision enable the extraction of information-rich embeddings from raw video sequences without the need for annotations. This property allows the effortless transfer of methods in different domains. In this work, we employ the temporal cycle-consistency (TCC) learning method [5]. The method is based on the task of alignment between videos. Therefore, the resulting per-frame TCC embeddings can be directly used to align video sequences.

Temporal alignment is a crucial step in our approach. Typically, to perform the quality assessment, manually clipped sequences of a particular action are used. The limits of the clips that result

PETRA 2021, June 29-July 2, 2021, Corfu, Greece

ACM ISBN 978-1-4503-8792-7/21/06…$15.00
https://doi.org/10.1145/3453892.3461624

from this manual annotation process are typically not consistent among different executions of the same action. Furthermore, the action execution speed and rhythm between subjects vary. These inconsistencies deteriorate the accuracy of the quality assessment methods. We alleviate this problem by performing temporal alignment of the sequences to a reference one before performing the quality assessment.

Given that the assessment is performed on sequences that are aligned at the frame level, we are able to support fine-grained explainability which is not the case for methods that rely on misaligned sequences.

In summary, the contributions of this work are the following: (a) We introduce the self-supervised TCC embeddings as a complementary representation to the I3D features, (b) We perform video alignment before the quality assessment relying on the same TCC embeddings. Both contributions result in increased accuracy of the quality score estimation. Furthermore, the performed action alignment enables us to provide explainability of the assessment at frame level.

2 RELATED WORK

Previous action quality assessment approaches include task-specific [1, 27] and generic methods [3, 25]. The assessment is performed either by regressing an execution quality score [25] or by performing pairwise comparisons [3]. Recent methods exploit several deep learning architectures including Siamese networks [3, 10], sequence models [27], and incorporate temporal [4] and spatial attention mechanisms [12]. The training process is based on transfer learning and is typically supervised [4, 19]. Both appearance [3, 19, 25] and pose [15, 17, 22] based features have been exploited. Interpretability emerges by the exploitation of the properties of back-propagation and attention modules [3, 10].

Appearance-based methods: Appearance-based methods utilize RGB features to assess the performed actions [11, 20, 21, 25, 26]. Most approaches use either C3D [11, 20], or I3D features [2, 25]. Li et al. [11] split a video into clips to extract C3D features and concatenate them to predict action scores. Tang et al. [25] proposed an uncertainty-aware score distribution to model the ambiguity arising from score variation among multiple judges. Parmar and Morris [20] proposed a multi-task learning approach that improves performance compared to methods that only regress a score.

Pose-based methods: Human pose estimation can be particularly challenging in certain situations. For instance, in sports, several factors such as fast motion, occlusions, and extreme body poses hinder the performance of most current pose estimation methods. Nevertheless, several works exploit the pose as it is a significant cue for assessing the quality of the performed action to complement appearance information. Pirsiavash et al. [22] rely solely on pose data and train a linear SVR to regress the execution score of athletic performances. Sardari et al. [23] trained a two stage CNN. The first stage is a viewpoint invariant descriptor of the trajectories of the body joints and the second stage regresses the action quality. Pan et al. [17] perform graph-based modeling of joint relations. Gao et al. [7] use I3D features along with a proposed asymmetric interaction module to model actions that include agents with different roles. Nekoui et al. [14, 15], use both pose and appearance features and present a dataset with extreme poses to boost the performance of pose estimation in such scenarios. In our work, we avoid using pose altogether. Instead, we complement appearance information with the self-supervised TCC embeddings, which are shown to boost the action accuracy without the need for difficult to obtain pose annotations.

Self-supervision in videos: Recently, several self-supervised video representation learning methods have been proposed. Shuffle and Learn (SaL) [13] learns to predict the temporal order of shuffled triplets. Time-Contrastive Networks (TCN) [24] is based on multiview videos where utilization of a metric learning loss enforces simultaneous viewpoint observations to be near in the embedding space. A Single-view version of TCN enforces embedding similarity within a small temporal window of observations. Temporal cycle consistency (TCC) learning [5] is a method that learns spatiotemporal representations by aligning video sequences of the same action. This is done by utilizing a differential cycle consistency loss when comparing nearest neighbors between two sequences.

To the best of our knowledge, there are no previous works that employ self-supervised embeddings to solve the action quality assessment problem. We use the TCC embeddings both to align the video sequences and to complement the appearance features.

3 METHODOLOGY

The proposed method regresses the action execution quality score of video segments. It relies on two learning stages and a temporal alignment step. The first learning stage is **self-supervised** and performs representation learning to extract the TCC embeddings. The TCC embeddings are subsequently used to align the video segments temporally. The aligned segments are then fed to the Action Quality Assessment (AQA) learning stage, which is supervised and uses two backbone models (TCC and I3D) to encode segments of videos and learn the action quality score.

3.1 Self-supervised TCC embeddings

Segment encoding: A base network encodes the spatial information of each frame independently. Spatiotemporal information is computed by fusing frame-level convolutional features within a segment. More specifically, a *ResNet-50* [9] architecture extracts convolutional features from the *Conv4c* layer, with a size of 14×14×1024. Temporal encoding is performed with 3D convolutions and spatiotemporal pooling. Linear projections (FC-layers) are used to produce a 128-dimensional embedding vector. It is important to note that an embedding of a video segment can be viewed as a segment-aware frame that encodes spatiotemporal information of k context frames.

Cycle consistency in videos: Temporal Cycle-Consistency learning dictates two segment-aware frames u_i and v_j, in sequences U and V respectively, to establish cycle consistency if they express the same motion within their sequence context. To test cycle consistency, initially we select a point $u_i \in U$. Then we find the nearest neighbor of u_i, $v_j \in V$ and nearest neighbor of v_j, $u_k \in U$ as:

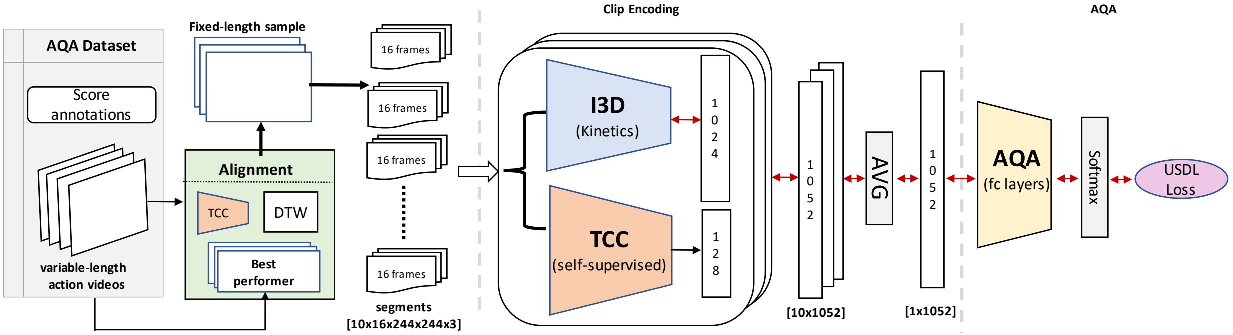


Figure 1: Overview of the proposed pipeline. Each video sample is aligned to a fixed-sized reference video V_{best} of the training set. The aligned sequence is split into M segments, which are passed to I3D and TCC backbones. Feature concatenation with average temporal pooling produces clip-level representation. AQA consists of FC-layers that aim to minimize USDL loss.

$$v_j = \underset{v \in \mathcal{V}}{\operatorname{argmin}} ||u_i - v||^2 \tag{1}$$

and

$$u_k = \underset{u \in \mathcal{U}}{\operatorname{argmin}} ||v_j - u||^2 \tag{2}$$

Cycle consistency exists, if $k = i$.

Temporal Cycle Consistency learning: In order for temporal cycle consistency formulation to be differentiable and ready to be integrated into a deep learning architecture, Soft-Nearest Neighbors (SSN) [6] with Cycle-back Regression is introduced. Learning is performed by randomly selecting pairs of videos and using segments to compute Cycle-back Regression which is the loss signal to optimize the encoder network.

Soft Nearest Neighbors: Given any frame with embedding x, we compute its soft nearest neighbor $\tilde{x}$, located in another video sequence U with N frames length, as:

$$\tilde{x} = \sum_{j}^{N} a_j v_j, \quad \text{where} \quad a_j = \frac{e^{-||x-v_j||^2}}{\sum_k^N e^{-||x-v_k||^2}}. \tag{3}$$

Cycle-back Regression loss: Initially, we select a frame $u_i \in U$ and keep the frame index i. First we compute the SNN of u_i, $\tilde{u} \in V$. Then we compute the similarity values β_k for each $u_k \in U$ using:

$$\beta_k = \frac{e^{-||\tilde{u}-u_k||^2}}{\sum_j^N e^{-||\tilde{u}-u_j||^2}}. \tag{4}$$

Each β_k and the frame index i are used to formulate the final objective:

$$L_{cbr} = \frac{|i-\mu|^2}{\sigma^2} + \lambda \log(\sigma), \tag{5}$$

where $\mu = \sum_k^N \beta_k \cdot k$, $\sigma = \sum_k^N \beta_k \cdot (k-\mu)^2$, and $\lambda = 0.001$ is a regularization weight.

3.2 AQA pipeline

We adopt the architecture from [20, 25] with the addition of a TCC based alignment pre-processing step and the incorporation of the TCC embeddings as complementary features to the I3D (see Fig. 1).

Video preparation: Prior to any other processing, the input videos are temporally aligned. We select a reference video V_{best} that corresponds to the best execution of an action, and we uniformly sample it to 103 frames. Each raw variable-length sample is aligned to the fixed-size reference video. The alignment is based on the *TCC* embeddings. We use the dynamic time warping (DTW) algorithm with the Euclidean distance metric.

Each fixed-length video of $N = 103$ frames is divided into $M = 10$ overlapping segments of size $k = 16$. Each RGB frame in the sequence is resized to $(w, h, c) = (224, 224, 3)$. The segment size k and spatial resolution (w, h, c) is dictated from the I3D backbone model we use in this work. The choice of input length N number utilized M segments is chosen according to be comparable with the baseline method [25].

Clip encoding: Each segment is encoded using two backbone models, the *TCC* [5] and the *I3D* [2]. The full feature vector is produced by concatenating the 1024-dimensional *I3D* feature vector and the 128-dimensional *TCC* embedding vector. The *I3D* model is pre-trained on the large-scale Kinetics dataset [2].

All segments with their accompanying segment-level features are passed from temporal pooling to produce a clip-level representation. Temporal pooling is performed with the AVG operator. Recent works [20, 25] demonstrate that applying temporal pooling after averaging features from the backbone part yields better performance.

Score prediction: The front-end transforms a clip-level representation to the prediction of the score annotation. It consists of 3 fully connected (FC) layers. The input dimension of the first FC-layer depends on the dimension of clip-level representation. The output dimension of the last layer depends on the input requirements of the loss function that is chosen. The *L1-L2* distance score loss function, used in [20], requires an output of a scalar value that corresponds to the score annotation. The *USDL* loss we use requires a quantized probability distribution. Thus, the output dimension of the last FC layers depends on the number of chosen bins. To predict probability ratios, the output of the end FC-layer is passed through a soft-max layer. The accepted score is the score that corresponds to the bin with the maximum probability value.

AQA loss function: We use the *USDL* loss function introduced in [25] due to its higher performance over *L1-L2* regression loss [20] and the ability to model uncertainty in single score labels. This is done by transforming each scoring label l to a Normal probability distribution $Q(x) = \mathcal{N}(x|\mu = l, \sigma^2)$, where σ is provided as hyperparameter. With USDL loss, training is based on minimizing the statistical discrepancy between predicted score distribution P and ground-truth score distribution Q. Kullback-Leibler (KL) Divergence is used in *USDL* loss where both score distributions are discretized into $b = 100$ bins.

4 EXPERIMENTS

4.1 Implementation details

Self-supervised training is performed, independently on the training set, for 150K iterations using *AdamOptimizer* with a fixed learning rate of 0.0001. The base network of the TCC encoding is pre-trained on ImageNet. We used the publicly available tensor-flow implementation of TCC [5].

Supervised AQA training is performed for 100 epochs using *AdamOptimizer* with a learning rate of 0.0001 and a weight decay of 0.00001. Our proposed AQA implementation is based on a publicly available implementation of [25]. During supervised training, the I3D backbone (pre-trained on kinetics datasets) is fine-tuned on the MTL-AQA dataset. The TCC encoding network does not consider the AQA loss and is not fine-tuned to the MTL-AQA dataset due to the incompatibility between the utilized deep-learning frameworks. Both encoding networks consider common spatial input of size of 224 × 224 and a temporal window of 16 frames.

4.2 Evaluated methods

We conducted a series of experiments to evaluate our method against state of the art in varying conditions. Specifically, we evaluate and compare the following methods and variants:

- **U-I3D**: Baseline method using I3D features [25].
- **A-I3D+TCC**: Proposed method using TCC based alignment and the TCC embeddings in the AQA pipeline.
- **A-I3D**: Our method using TCC based alignment only.
- **U-I3D+TCC**: Our method using the TCC embeddings in the AQA pipeline only.

4.3 Evaluation metrics

For compatibility with the evaluation process of previous works [8, 17, 20, 21, 25], we utilize the Spearman's rank correlation coefficient as an evaluation metric. To improve the quality of experiments, we repeat each model evaluation 5 times and report the median value as the final performance. Each model is trained for 100 epochs for the supervised AQA training, and its corresponding self-supervision session was performed in 150K iterations.

Table 1: Spearman's correlation scores of the evaluated methods on the selected dive.

Method	Spearman's rank correlation
U-I3D	0.62
A-I3D	0.70
U-I3D+TCC	0.66
A-I3D+TCC	**0.77**

4.4 Dataset

Evaluation is based on the MTL-AQA dataset that is introduced in [20]. This dataset focuses on diving performances and is the largest AQA dataset available with a total of 1412 samples. With respect to other introduced diving datasets [18, 21], it has higher variability in diving actions, performer gender (both males and females), and background variation. Each sample is labeled with a final score, calculated from 7 individual judges, and filtered through the decision process that considers action difficulty. Additionally, it is accompanied by fine-grained annotations such as action type attributes and commentary text. Regarding the action type attributes, each dive is annotated with the initial position of the athlete, such as a handstand (yes or no) and the type of dive flip that is performed (position, rotation type, number of somersaults, and number of twists). Moreover, each diving type is characterized by a difficulty score.

We test our method on a specific dive. We test the hypothesis that our proposed approach can improve existing methods which are designed to evaluate a variety of diving types by considering adequate samples. As in [16, 20], we pick action types by utilizing the action attributes (*arm stand*, *position*, *rotation_type*, *ss_no*, *tw_no)*. The training and test sets of each sub-action are generated using the *split0* dataset split, which is the standard evaluation split for MTL-AQA.

4.5 Quantitative results

Based on the selected dataset, we obtained quantitative results regarding the performance of all considered methods. The selected dive consist of 112 train and 33 test samples. The results are summarised on Table 1. To obtain these results, we used the original dataset annotations for the video limits. We didn't consider the preprocessing step that is used in the implementations of [20, 25] which considers only the end limit. We consider that this pre-processing step artificially creates an alignment that does not generalize in other datasets or action types. As it can be verified in Table 1, in this realistic setting (i.e., using the original annotations) our **A-I3D+TCC** method results in a clear performance improvement compared to the **U-I3D** baseline.

We also evaluated the influence of the alignment and of the TCC embeddings on the performance of the proposed method. The results of Table 1 show that the use of the TCC embeddings in the AQA pipeline improves the modeling capability of the network as well as the performance (**U-I3D+TCC**) compared to the

Table 2: Spearman's correlation scores of the baseline and the proposed method in the presence of annotation noise.

Method	Spearman's rank correlation
U-I3D	0.67
A-I3D+TCC	**0.77**

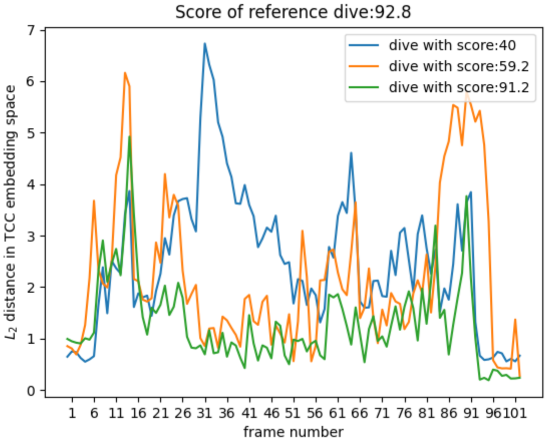


Figure 2: Per frame discrepancy of three diving samples in the TCC embedding space, when compared with the best performer. By aligning the action executions and comparing them in the embedding space, it is possible not only to perform AQA but also to localize appearance deviations temporally from the best performance.

baseline. Using only the alignment (**A-I3D**) also improves the performance compared to the baseline. This indicates that even though the baseline method provides robustness to small alignment errors, in practice, the actual alignment errors in the dataset deteriorate its performance.

To further investigate the relationship between alignment and the performance of the methods, we conducted another experiment where we added uniform noise with range 2 segments to the original annotation limits. The results are shown in Table 2. In this specific experiment, the baseline is favored by utilizing the heuristic pre-processing step that considers only the end limit [21, 25]. When annotation noise is applied to the baseline U-I3D, which uses heuristic prepossessing, performance drops from 0.79 to 0.67. Our proposed method considers all annotation noise in both self-supervised and supervised stages. Our method is shown to be robust to this noise, having the same performance as without the noise.

4.6 Interpretability and qualitative results

To the best of our knowledge, the interpretability of existing AQA methods is demonstrated coarsely by visualizing segments of single sequences. Practically, each visualized segment is not guaranteed to correspond to a specific part of the action. In Fig. 2, we demonstrate the effectiveness of the alignment techniques in interpretability. We select the best performer, and we perform a frame-by-frame comparison with other performers by utilizing the Euclidean distance of their representations in the TCC embedding space. It is worth noting that, in this work, TCC embeddings are not trained or fine-tuned with a loss function that is aware of score labels. At certain parts of the sequence, performers tend to deviate considerably from the best performer. As an example, the spike of the blue line that corresponds to the performer with a score of 40.0 indicates that there is a high-performance discrepancy of this performer when compared with the best performer, localized at that particular temporal point. This type of discrepancy visualization can assist anomaly detection and guide judges to focus their attention on that specific part of the sequence. What is mentioned above demonstrates that utilizing alignment-based representation learning has a strong potential towards interpretable AQA systems.

The influence of the alignment process on the input that is fed to the AQA pipeline is demonstrated qualitatively in Figure 3. The ending of a reference video is shown in the first row. In rows 2, 3, the three columns correspond to three different executions with a different score. The second row shows which frames are obtained when we uniformly normalize diving samples, i.e., without aligning them to the reference. The third row shows the obtained frames after alignment has been performed. The reference video stops a few frames before the athlete enters the water. From what is shown in the second row, we can observe that corresponding frames differ significantly and are not comparable since they are not aligned and, thus, they don't represent the same part of the action. In the third row, we can observe that frames are aligned closer to the frame of the reference videos; thus frame-by-frame comparison and evaluation is possible.

5 CONCLUSIONS

We proposed a novel action quality assessment method that employs self-supervised embeddings and video alignment. We choose the TCC embeddings since they don't have any special training requirements, and they are, by construction, able to perform video alignment. Integration of the embeddings is simple (feature concatenation only) and is not coupled to a specific network type. On a selected diving action of the MTL-AQA dataset, we have experimentally shown that the proposed method outperforms the previous approaches and can provide fine-grained interpretability of the action quality assessment scores. As future work, we plan to test our approach on datasets with performances from different sports and render the method able to generalize to multiple action types.

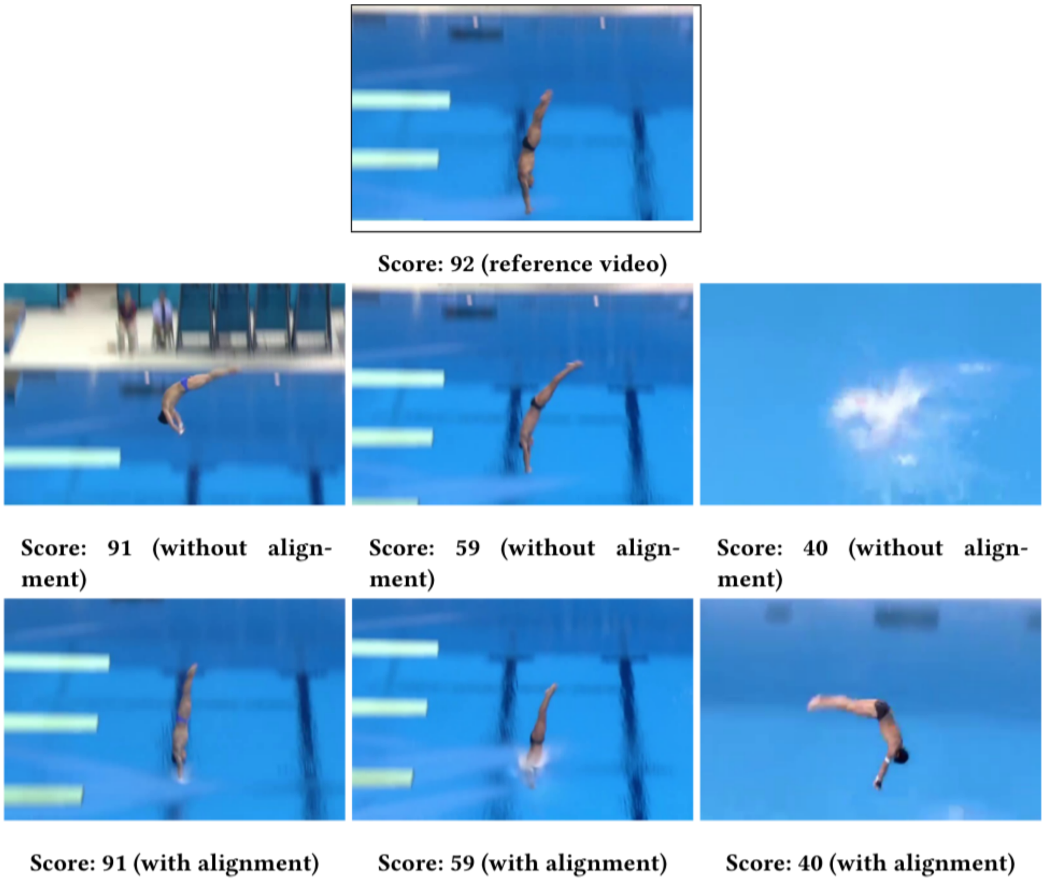
REFERENCES

[1] Gedas Bertasius, Hyun Soo Park, Stella X. Yu, and Jianbo Shi. 2017. Am I a Baller? Basketball Performance Assessment From First-Person Videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[2] João Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4724–4733. https://doi.org/10.1109/CVPR.2017.502

[3] Hazel Doughty, Dima Damen, and Walterio Mayol-Cuevas. 2018. Who's Better? Who's Best? Pairwise Deep Ranking for Skill Determination. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), 6057–6066. https://doi.org/10.1109/CVPR.2018.00634 arXiv:1703.09913

[4] Hazel Doughty, Walterio Mayol-Cuevas, and Dima Damen. 2019. The Pros and Cons: Rank-Aware Temporal Attention for Skill Determination in Long Videos.



Score: 92 (reference video)

Score: 91 (without alignment)

Score: 59 (without alignment)

Score: 40 (without alignment)

Score: 91 (with alignment)

Score: 59 (with alignment)

Score: 40 (with alignment)

Figure 3: Visualization of action alignment with a reference video. Top row: the ending of a reference diving performance with a score of 92. Second and third rows: Each column corresponds to a different dive with a different score. Second row: the obtained frames, without alignment to the reference video. Third row: the obtained frames when diving samples are aligned to the reference video, using TCC embeddings.

2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019), 7854–7863.

[5] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. 2019. Temporal Cycle-Consistency Learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[6] Nicholas Frosst, Nicolas Papernot, and Geoffrey Hinton. 2019. Analyzing and Improving Representations with the Soft Nearest Neighbor Loss. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 2012–2020. http://proceedings.mlr.press/v97/frosst19a.html

[7] Jibin Gao, Wei Shi Zheng, Jia Hui Pan, Chengying Gao, Yaowei Wang, Wei Zeng, and Jianhuang Lai. 2020. An Asymmetric Modeling for Action Assessment. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12375 LNCS (2020), 222–238. https://doi.org/10.1007/978-3-030-58577-8_14

[8] Yixin Gao, S. Swaroop Vedula, Carol E. Reiley, Narges Ahmidi, Balakrishnan Varadarajan, Henry C. Lin, Lingling Tao, Luca Zappella, Benjamín Béjar, David D. Yuh, Chi Chiung Grace Chen, René Vidal, Sanjeev Khudanpur, and Gregory D. Hager. 2014. *JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS): A Surgical Activity Dataset for Human Motion Modeling*. Technical Report. 1–10 pages.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] H. Jain, G. Harit, and A. Sharma. 2020. Action Quality Assessment using Siamese Network-Based Deep Metric Learning. *IEEE Transactions on Circuits and Systems for Video Technology* (2020), 1–1. https://doi.org/10.1109/TCSVT.2020.3017727

[11] Yongjun Li, Xiujuan Chai, and Xilin Chen. 2018. End-To-End Learning for Action Quality Assessment. In *Advances in Multimedia Information Processing – PCM 2018*, Richang Hong, Wen-Huang Cheng, Toshihiko Yamasaki, Meng Wang, and Chong-Wah Ngo (Eds.). Springer International Publishing, Cham, 125–134.

[12] Zhenqiang Li, Yifei Huang, Minjie Cai, and Yoichi Sato. 2019. Manipulation-skill assessment from videos with spatial attention network. *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019* (2019), 4385–4395. https://doi.org/10.1109/ICCVW.2019.00539 arXiv:1901.02579

[13] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. 2016. Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification. In *Computer Vision – ECCV 2016*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (Eds.). Springer International Publishing, Cham, 527–544.

[14] Mahdiar Nekoui, Fidel Omar Tito Cruz, and Li Cheng. 2021. EAGLE-Eye: Extreme-Pose Action Grader Using Detail Bird's-Eye View. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 394–402.

[15] Mahdiar Nekoui, Fidel Omar Tito Cruz, and Li Cheng. 2020. FALCONS: Fast learner-grader for contorted poses in sports. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* 2020-June (2020), 3941–3949. https://doi.org/10.1109/CVPRW50498.2020.00458

[16] Aiden Nibali, Zhen He, Stuart Morgan, and Daniel Greenwood. 2017. Extraction and Classification of Diving Clips From Continuous Video Footage. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

Workshops.

[17] Jia-hui Pan, Jibin Gao, and Wei-shi Zheng. 2019. Action Assessment by Joint Relation Graphs. *Iccv* (2019), 6331–6340.

[18] Paritosh Parmar and Brendan Tran Morris. 2019. Action quality assessment across multiple actions. *Proceedings - 2019 IEEE Winter Conference on Applications of Computer Vision, WACV 2019* abs/1812.0 (2019), 1468–1476. https://doi.org/10.1109/WACV.2019.00161 arXiv:1812.06367

[19] Paritosh Parmar and Brendan Tran Morris. 2019. What and how well you performed? a multitask learning approach to action quality assessment. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2019-June (2019), 304–313. https://doi.org/10.1109/CVPR.2019.00039 arXiv:1904.04346

[20] Paritosh Parmar and Brendan Tran Morris. 2019. What and How Well You Performed? A Multitask Learning Approach to Action Quality Assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[21] Paritosh Parmar and Brendan Tran Morris. 2017. Learning to Score Olympic Events. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[22] Hamed Pirsiavash, Carl Vondrick, and Antonio Torralba. 2014. Assessing the quality of actions. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 8694 LNCS. 556–571. https://doi.org/10.1007/978-3-319-10599-4_36

[23] Faegheh Sardari, Adeline Paiement, Sion Hannuna, and Majid Mirmehdi. 2020. VI-Net—View-Invariant Quality of Human Movement Assessment. *Sensors* 20, 18 (2020). https://doi.org/10.3390/s20185258

[24] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain. 2018. Time-Contrastive Networks: Self-Supervised Learning from Video. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 1134–1141. https://doi.org/10.1109/ICRA.2018.8462891

[25] Yansong Tang, Zanlin Ni, Jiahuan Zhou, Danyang Zhang, Jiwen Lu, Ying Wu, and Jie Zhou. 2020. Uncertainty-Aware Score Distribution Learning for Action Quality Assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[26] Xiang Xiang, Ye Tian, Austin Reiter, Gregory D. Hager, and Trac D. Tran. 2018. S3D: Stacking Segmental P3D for Action Quality Assessment. In *Proceedings - International Conference on Image Processing, ICIP*. IEEE Computer Society, 928–932. https://doi.org/10.1109/ICIP.2018.8451364

[27] Chengming Xu, Yanwei Fu, Bing Zhang, Zitian Chen, Yu-Gang Jiang, and Xiangyang Xue. 2019. Learning to Score Figure Skating Sport Videos. *IEEE Transactions on Circuits and Systems for Video Technology* (2019), 1–1. https://doi.org/10.1109/tcsvt.2019.2927118 arXiv:1802.02774