Towards a Data-Driven Method for RGB Video-Based Hand Action Quality Assessment in Real Time

Tianyu Wang, Minhao Jin, Jingying Wang, Yijie Wang, Mian Li

UM-SJTU Joint Institute, Shanghai Jiao Tong University
800 Dongchuan Road, Minhang District, Shanghai, China
{gunnerwang27,jinminhao,wjymonica,yijiewang,mianli}@sjtu.edu.cn

1. Introduction

THE research community has begun to explore the area of Video-Based Action Quality Assessment on Human Body (**VB-AQA**) in recent years. VB-AQA resolves the absence of expert instructions. For VB-AQA, previous work fails to establish a consistent one-to-one matching between the features extracted and their corresponding physical part. Moreover, the computational efficiency is not seriously considered. The two issues become more obvious when it comes to human hands with a more complicated structure. To this end, the exploration on Video-Based Action Quality Assessment on Human Hand (**VH-AQA**) is still limited. To fill the gap between VB-AQA and VH-AQA, a novel data-driven method is proposed to rapidly assess the quality of continuous hand actions shown in the RGB video. The experiments are conducted on our introduced **Origami Video Dataset**. The contribution of our work is summarized as: (i) the first method, to the best of our knowledge, for VH-AQA that addresses the issues of mismatch and computational efficiency and (ii) a novel Origami dataset and two new metrics for validating methods relevant to VH-AQA.

2. Related Work

VB-AQA. Wang [3] employed a minor tuned KNN based model to classify the performance of moving bodies with templates. Pirsiavash *et al.* [1] trained a Linear Support Vector Regression (L-SVR) based model on both low-level and high-level spatio-temporal features.

Hand Pose Estimation. Deep learning is employed to estimate such hand poses with RGB camera instead of RGB-D one. The methods commonly follow the process of hand segmentation, pose estimation, and pose refinement. Zimmermann *et al.* [5] proposed a deep network that learns a network-implicit 3D articulation prior and yields a good estimate of the 3D hand pose.

3. Problem Definiton

For a regular RGB video, the features of jth joint in the tth frame are denoted as $\mathbf{p}^{(j)}(t) = [x^{(j)}(t), y^{(j)}(t), z^{(j)}(t)]$, where all the elements are normalized coordinates relative to the palm. The well-organized high-level features involving $2m$ joints for both hands (from left to right) in tth frame are $\boldsymbol{\phi}(t) = [\mathbf{p}^{(1)}(t), \mathbf{p}^{(2)}(t), \cdots, \mathbf{p}^{(2m)}(t)]$. Based on features of all valid frames, the quality assessment of hand actions consists of two tasks: performance evaluation and feedback indication. The former one generates a single score for each video. The latter one provides feedback that indicates how the performer should adjust each of his static poses to achieve the largest improvement in score.

4. Methodology

The overall framework is illustrated in Figure 1.

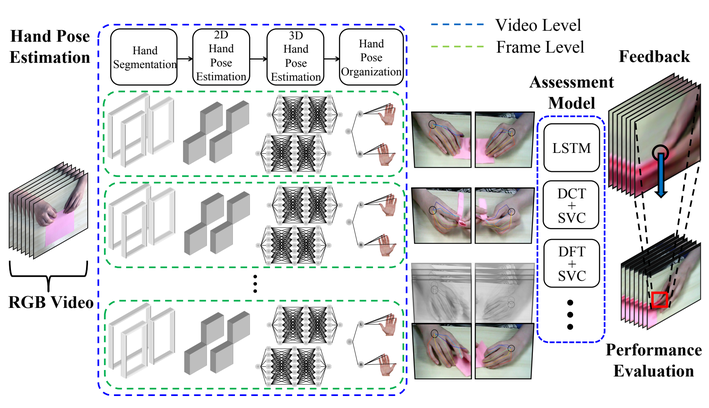

Figure 1: *The framework of our method.*

4.1 Hand Pose Estimation and Organization

Hand Segmentation. It is implemented referring to [2] based on *Egohands Dataset*. We notice that the bottleneck of computational efficiency usually lies in cropping the segmented part from the original image, since all pixels on the boundary need to be identified for this action. Therefore, we do not crop along the precise boundary of the detected boxes. Instead, flexible areas covering the detected boxes can be cropped and adjusted later.

2D Hand Pose Estimation. It is implemented with an encoder-decoder structure according to [4]. The 2D hand pose is estimated with keypoint score maps indicating the probability of each location being selected as a keypoint. An initial score map is predicted with an image feature representation produced by the encoder.

3D Hand Pose Estimation. It is conducted with *PosePrior* network in [5]. The network predicts relative and normalized 3D coordinates based on incomplete and noisy keypoint score maps obtained from the last step.

Hand Pose Organization. The forearms connected to both hands are detected if possible. If such regions extend to the lower image boundary, an egotistic view of the camera is concluded, where the left segment obtained is related to the left hand and right to right. Otherwise, an observer's view is concluded and the relationship is reversed. Four states are further defined and distinguished for each hand according to algorithm described in Figure 2 (the same for right hand). The states of open and fist front of the left hand are equivalent to the states of open and fist back of the right hand considering the desired outcome. Therefore, states of both hands boil down to distinct treatments towards left and right hands. Concretely, they are treated with the agnostic representation mentioned in [5].

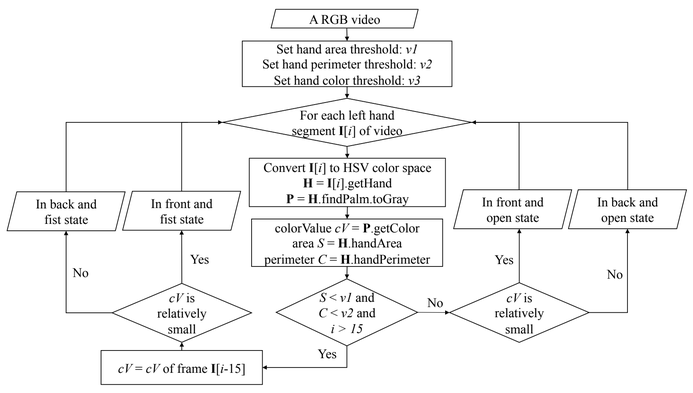

Figure 2: *The flow chart of the algorithm for recognizing geometric states of left hand segment.*

4.2 Quality Assessment of Hand Action

Performance Evaluation. Three models are employed for comparison. First, we leverage a 1-layer LSTM network. Second, DCT is applied to obtain synthesized features in frequency domain. Then SVC is manipulated upon the defined number of low frequency components to give a score. Third, DCT is replaced by DFT. The network is trained to minimize the Mean Absolute Error (MAE) between the target and predicted score:

$$\mathcal{L}(\cdot) = \sum_{n=1}^{N} \left| S_n - \underset{i \in \{1,2,3\}}{\arg\max}\, \hat{\mathbf{S}}_{n,i} \right|, \tag{1}$$

where $S_n \in \{1, 2, 3\}$ is the ground-truth performance score of nth video clip and $\hat{\mathbf{S}}_{n,i}, i = 1, 2, 3$ corresponds to the predicted probability for each of performance level.

Feedback for Improvement. We demonstrate our feedback with LSTM. It is accomplished by differentiating the output score with respect to the extracted features in each frame. The process begins with the last frame at time t_0. The output gate at t_0 is expressed as $\mathbf{O}(t_0) = \sigma\left(\mathbf{W}_o\boldsymbol{\phi}(t_0) + \mathbf{U}_o\mathbf{O}(t_0 - 1) + \mathbf{b}_o\right)$, where $\sigma(\cdot)$ is *sigmoid* function, $\mathbf{W}_o$ and $\mathbf{U}_o$ are shared parameter matrices, and $\mathbf{b}_o$ is biased term. For simplicity, we denote $\mathbf{Q}_k(t_0) = \sigma\left(\mathbf{W}_k\boldsymbol{\phi}(t_0) + \mathbf{U}_k\mathbf{O}(t_0 - 1) + \mathbf{b}_k\right)$ and $\mathbf{L}_k(t_0) = \tanh\left(\mathbf{W}_k\boldsymbol{\phi}(t_0) + \mathbf{U}_k\mathbf{O}(t_0 - 1) + \mathbf{b}_k\right)$. The corresponding gradient is calculated as

$$\begin{aligned}\frac{\partial \hat{\mathbf{S}}}{\partial \boldsymbol{\phi}(t_0)} &= H\left(\mathbf{W'}^{\mathrm{T}}\mathbf{O}(t_0) + \mathbf{b'}\right)\mathbf{W'}^{\mathrm{T}} \\ &\times \left[\mathbf{A'}\mathbf{Q}_o(t_0) + \mathbf{A}\mathbf{Q}_o(t_0)(1 - \mathbf{Q}_o(t_0))\mathbf{W}_o\right],\end{aligned} \tag{2}$$

where $H(\cdot)$ is derivative function of *softmax*, $\mathbf{W'}$ and $\mathbf{b'}$ are weight matrix and bias term for fully connected layer, respectively, and $\mathbf{A} = \tanh\left(\mathbf{L}_a(t_0)\mathbf{Q}_i(t_0) + \mathbf{Q}_f(t_0)\mathbf{h}(t_0 - 1)\right)$, $\mathbf{A'} = (1 - \mathbf{A}^2)[\mathbf{h}(t_0 - 1)\mathbf{Q}_f(t_0)(1 - \mathbf{Q}_f(t_0))\mathbf{W}_f + (1 - \mathbf{L}_a^2(t_0))\mathbf{W}_a\mathbf{Q}_i(t_0)\mathbf{L}_a(t_0)\mathbf{Q}_i(t_0)(1 - \mathbf{Q}_i(t_0))\mathbf{W}_i]$. The rendered gradient is represented as a $3 \times 6m$ matrix. Our objective is to select the maximum element of the row vector corresponding to the good performance $\max_{\boldsymbol{\phi}(t_0)}\left[\frac{\partial \hat{\mathbf{S}}}{\partial \boldsymbol{\phi}(t_0)}\right]_{\mathrm{g}}$. The outcome indicates the joint along with the corresponding direction that improves the score most. Furthermore, we can calculate for other t's by following the back propagation chain of the LSTM cell.

5. Experiments

Dataset. To the best of our knowledge, there is no public dataset relevant to VH-AQA. To this end, we establish our own **Origami Video Dataset** which consists of 144 video clips: 44 for good, 66 for medium and 34 for bad. The dataset involves a basic action in Origami: folding a piece of square paper into 8×8 small squares. The experts score one's performance with three levels based on the following rubrics: (1) whether the person ensures two edges overlap strictly when folding the paper lengthwise; (2) whether the creases are thin and clear. Ideally, the person should end up with clear and solid edges of the small squares.

Evaluation Accuracy. Performances of the three trained models on the test set are demonstrated and compared under different metrics and classes (Table 1). For LSTM, while it predicts quite well on good-level actions, the performance drops on the bad level. The reason is that LSTM is sensitive to the time domain variation where the good-level actions stand out from others. In the meantime, the latter two models feature a concentration of time-series data and thus achieve a more balanced outcome on the three levels. Therefore, LSTM is more suitable for time-sensitive actions while DCT+SVC shows more advantages in actions with standard rules to follow.

Table 1: *Model comparison.*

METRIC	LSTM	DCT+SVC	DFT+SVC
Accuracy	**89.91**	85.41	83.78
AUC	93.97	**95.95**	95.35
AUC (Bad)	89.41	**97.29**	97.23
AUC (Medium)	92.22	**93.44**	92.57
AUC (Good)	**97.19**	96.64	95.71

Computational Efficiency. We define two new time metrics: the Reconstruction Computational Cost (**RCC**), the computational time of hand pose estimation per frame and the Assessment Computational Ratio (**ACR**), the ratio of the computational time used for assessment to the video duration. The experiment is conducted on GPU 1080 Ti. The mean RCC fluctuates around **0.08 s**, indicating that for 12-fps videos, well-organized hand poses are nearly synchronous with the ongoing action. Besides, the mean ACR's over videos for three performance levels are **0.23**, **0.077** and **0.11**, respectively.

Feedback for Improvement. Two working examples are demonstrated in Figure 3. The joint that needs adjusting most is specified by the black circle, while the direction of the largest gradient is specified by the blue arrow.

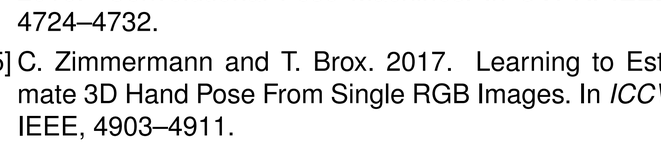

Figure 3: *Two examples of feedback for good hand actions.*

6. Conclusion

In this paper we proposed a novel method for rapidly assessing the quality of continuous hand actions shown in RGB videos. We demonstrated the method's evaluation accuracy and computational efficiency on our own Origami Video Dataset and two new metrics. In the future, we plan to contribute a larger and more diverse dataset as well as investigate more high-level features for VH-AQA.

References

[1] H. Pirsiavash, C. Vondrick, and A. Torralba. 2014. Assessing the Quality of Actions. In *ECCV*. Springer, 556–571.

[2] D. Victor. 2017. Real-time Hand Tracking Using SSD on Tensorflow.

[3] Y. Wang. 2018. Fitness Movement Recognition and Evaluation Based on Kinect. *Computer Science and Application* 8 (Jan. 2018), 1134–1145.

[4] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. 2016. Convolutional Pose Machines. In *CVPR*. IEEE, 4724–4732.

[5] C. Zimmermann and T. Brox. 2017. Learning to Estimate 3D Hand Pose From Single RGB Images. In *ICCV*. IEEE, 4903–4911.