On the importance of temporal features in Domain Adaptation methods for Action Recognition

Donatello Conte, Giuliano Giovanni Fioretti, Carlo Sansone

On the importance of temporal features in Domain Adaptation methods for Action Recognition

Donatello Conte[1][0000−0003−4642−4768], Giuliano Giovanni Fioretti[2][0000−0001−8233−2428], and Carlo Sansone[2][0000−0002−8176−6950]

[1] Université de Tours
Laboratoire d'Informatique Fondamentale et Appliquée de Tours (LIFAT - EA6300)
64 Avenue Jean Portalis, 37200 Tours, France
donatello.conte@univ-tours.fr
[2] Department of Electrical Engineering and Information Technology (DIETI)
University of Naples Federico II, via Claudio 21, 80125 Naples, Italy
giul.fioretti@studenti.unina.it, carlo.sansone@unina.it

Abstract. One of the most common vision problems is Video based Action Recognition. Many public datasets, public contests, and so on, boosted the development of new methods to face the challenges posed by this problem. Deep Learning is by far the most used technique to address Video-based Action Recognition problem. The common issue for these methods is the well-known dependency from training data. Methods are effective when training and test data are extracted from the same distribution. However, in real situations, this is not always the case. When test data has a different distribution than training one, methods result in considerable drop in performances. A solution to this issue is the so-called Domain Adaptation technique, whose goal is to construct methods that adapt test data to the original distribution used in training phase in order to perform well on a different but related target domain. Inspired by some existing approaches in the scientific literature, we proposed a modification of a Domain Adaptation architecture, that is more efficient than existing ones, because it improves the temporal dynamics alignment between source and target data. Experiments show this performance improvement on public standard benchmarks for Action Recognition.

Keywords: Domain Adaptation · Action Recognition · Temporal Shift Model

1 Introduction

Human action recognition has become a challenging topic during the last years due to the impact that it represents in the comprehension of human behaviour (see [19, 21, 27] for some examples). Unsurprisingly, the action recognition field is extremely more difficult task than object recognition, for various reasons. In addition to the typical difficulties related to image recognition, such as scale

variation, lighting or contrast, other obstacles must be taken into account to perform action recognition: the shooting point of view, the color scheme, the background clutter, and most importantly, the temporal dimension that further complicates the task.

Deep learning is by far the most used technique to address this problem (e.g. [16, 7, 24, 29], and [28] for a good survey of these methods). While very effective, these techniques suffer from the problem of being too dependent on training data. The vast majority of supervised learning methods share a common prerequisite: training data and testing data are extracted from the same distribution [26]. However this may not always be the case. When this constraint is violated, the classifier trained on a dataset, which will be referred to as the *source domain*, exhibits a considerable drop in performance when tested on a different dataset, called the *target domain*.

A solution to this issue can be the use of the so-called Domain Adaptation. Domain adaptation refers to the goal of learning a concept from labeled data in a source domain that performs well on a different but related target domain. There are many domain adaptation proposals in the scientific literature, also in the context of Action Recognition (see Section 2).

The main goal of this paper is to propose a new architecture of Domain Adaptation for Action Recognition. The contribution is to propose integrating a different temporal module within an existing architecture, in order to improve the temporal adaptability between source and target domains, and consequently, improving the performances.

The rest of the paper is organized as follows: Section 2 illustrates a brief survey of the main domain adaptation techniques for Action Recognition. In Section 3 we recall the basic principles of an existing architecture for Domain Adaptation which served as our basis for proposing our modification described in Secion 4. In Section 5, after describing the test protocol, we present some experimental results that prove the effectiveness of our approach. Some conclusions and perspectives are drawn in the last Section 6.

2 Related Works

Domain adaptation methods make the assumption that the tasks are the same and the differences are only caused by domain divergence. According to Wang et al. [5], considering the labeled data of the target domain, domain adaptation algorithms can be classified in three categories:

- Supervised (e.g. [17]): a small amount of labeled target data are present. The issue is that the labeled data are commonly not sufficient for tasks.
- Semi-supervised (e.g. [20]): there are limited labeled data and redundant unlabeled data in the target domain. This only allows the network to learn the structure information of the target domain.
- Unsupervised (e.g. [14]): there are labeled source data and only unlabeled target data available for training the network.

Supervised action recognition state-of-the-art methods all use deep learning algorithms, by leveraging CNNs for spatio-temporal information (see [2, 25] for some examples). However, as we said in the Section 1, these supervised action recognition approaches are still limited by the dependency on annotated labels for each clip. There is no guarantee of robust performances if the algorithms are directly transferred to another domain, due to the presence of domain shift.

In this context, many Domain Adaptation techniques are proposed and applied to the Action Recognition problem. First works in this direction was focused geometric transformations of videos in the context of Supervised Domain Adaptation. Some works utilise supervisory signals such as skeleton or pose [12] and corresponding frames from multiple viewpoints [22, 8].

On the contrary, Unsupervised Domain Adaptation (UDA) has been used for recovering and adapting changes, more general than only geometrical ones. At the first time, UDA for action recognition used shallow models to align source and target distributions of handcrafted features [1, 4]. By the advent of Deep Learning, more proposals have been done for Domain Adaptation, but especially for image-based tasks. Authors of [13, 15] proposed some Deep Networks to align the joint distributions by minimizing maximum mean discrepancy (MMD) or joint maximum mean discrepancy (JMMD) between source and target domains. Recently, some works deals with video domain adaptation. In [6] authors utilize an adversarial learning framework with 3D CNN to align source and target domains. TA3N [3] leverages a multi-level adversarial framework with temporal relation and attention mechanism to align the temporal dynamics of feature space for videos.

Inspired by this last work, and being convinced that temporal alignment remains the key feature in domain adaptation, our contribution can be stated as follow: we propose adapting an existing temporal alignment module to a Domain Adaptation Deep Network for improving the temporal dynamics correspondences between source and target videos.

3 Recalls basics of an architecture for Domain Adaptation

The architecture proposed by Chen et al. [3] consists of 3 main components: a spatial module, a temporal module, and a class predictor. The input data comes from frame-level feature vectors, which are extracted directly from the raw action videos via a ResNet convolutional network.

Spatial module. This component uses fully-connected layers to translate feature vectors into features useful for the action recognition task.

Temporal module. This module consists of a TRN that extracts the most representative frames of the videos and identifies the temporal relationships between them.

To capture temporal relations at different time scales, in the temporal relation module, multiple representations of relation features are generated, each of which leverages a different number of ordered frames.

In mathematical formula, considering a video input V composed by n ordered

frames, the temporal relation between k frames can be expressed as:

$$T_k(V) = h_\phi \left(\sum_{i<j<\ldots<k} g_\theta(f_i, f_j, \ldots, f_k) \right) \tag{1}$$

where f_i indicates the feature representation of the i^{th} frame and the h and g functions aim to fuse together the features of multiple frames.

This formula can then be used to accumulate frame relations at multiple time scales to obtain multiscale temporal relations:

$$MT_N(V) = T_2(V) + T_3(V) + \ldots + T_N(V) \tag{2}$$

Then, an attention mechanism focused on domain discrepancy is used. This is made up of a series of domain attention blocks with relation discriminators for each representation of relation features. Finally, the outputs of the blocks are added together.

Class predictor. Finally, this classifier is also a fully connected layer that converts features extracted from previous modules into the final class. For this purpose an attentive entropy loss is generated by domain entropy and class entropy.

In addition to these modules, there is a set of discriminators trained with the adversarial learning technique inspired by DANN [5]. The aim of these networks is to learn domain-invariant features and a symmetric mapping of source and target distributions in order to align them both spatially and temporally.

Spatial discriminator. It applies an image-based domain adaptation to learn the spatial parameters by maximizing the spatial domain discrimination loss.

Temporal discriminator. It applies a temporal-based domain adaptation to learn the features encoded in the temporal dynamics by maximizing the temporal domain discrimination loss.

Relation discriminator. It generates a domain attention value, used to attend local temporal features, maximizing the relation domain loss.

4 The new designed architecture

The architecture described above focuses on aligning the features that contribute more than others to the overall domain shift, i.e. those that have a greater discrepancy between domains.

A larger domain gap, however, can be caused by frames that are not relevant to the action recognition task. Therefore, aligning those irrelevant frames can lead to suboptimal results in some cases.

Moreover, this module performs temporal modeling only after feature extraction, identifying the most descriptive frames of actions and attending on those.

Therefore, several pieces of information may be lost during the feature extraction process.
These considerations led to the idea of designing a new architecture, with a temporal module that overcomes these disadvantages.

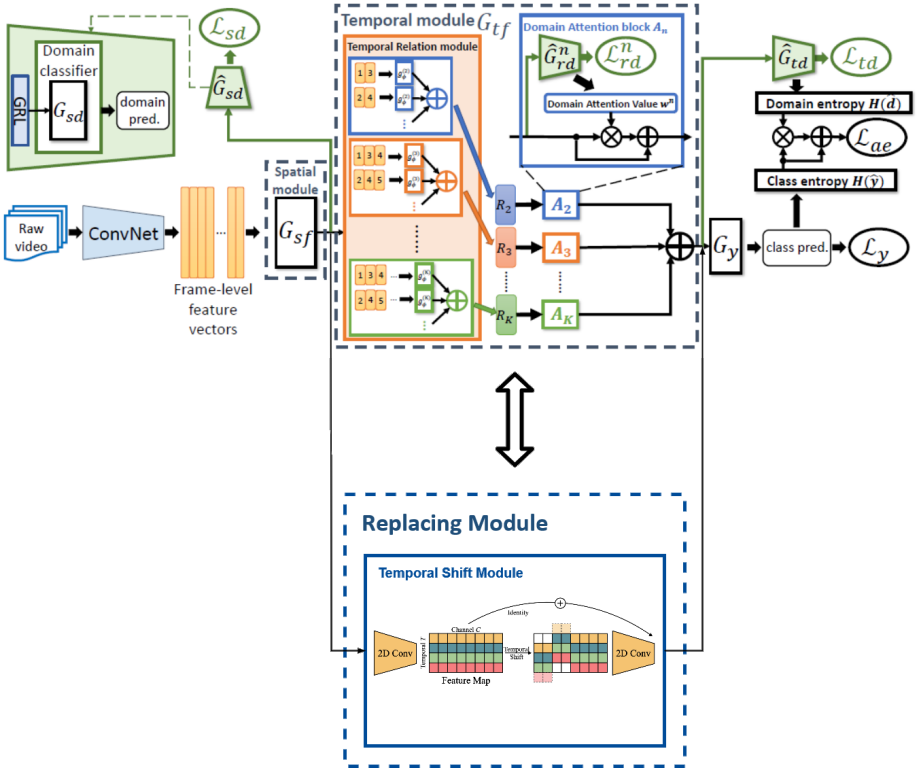

Fig. 1. The new designed architecture. Above we show the original architecture of domain adaptation. There are three main parts: on the left the feature extractor, in the middle the temporal module which aims to align the time instants of the sequences in the two domains, and on the right the classification part. The adaptation of the architecture that we propose, consists in replacing the temporal module (Temporal Relation Network, TRN) with the Temporal Shift Module (TSM) that is illustrated below. Note that, contrary to the original architecture, features of the video are directly fed in TSM, avoiding frames alignment as in TRN.

Specifically, the use of the Temporal Shift Module proposed by Lin et al. [11] in place of the Temporal Relation Network (TRN) was considered (as shown in Fig. 1). This module has the advantage of enabling all levels of temporal modeling, even during the feature extraction itself, just like methods based on 3D CNNs, but maintaining a reasonable computational time. Consequently, there

is no need to attend frame relations. Furthermore, this module avoids the risk of aligning unnecessary frames.
As explained by Lin et al. [11], an activation in a video model can be represented as $A \in \mathbb{R}^{N \times C \times T \times H \times W}$, where N is the batch size, C is the number of channels, T is the time, H is the height and W is the width of the image.
TSM performs temporal modelling shifting by ± 1 along the temporal dimension a fraction of the features extracted from the frame, about $\frac{1}{8}$, to merge them with the features extracted from the frames immediately preceding and immediately following.
Consequently, after the shifting operation, the information of the current frame is fused together with the neighboring frames. An example of a tensor with C channels and T frames can be seen in figure 2, where different colors for each row indicate the features extracted from different frames.

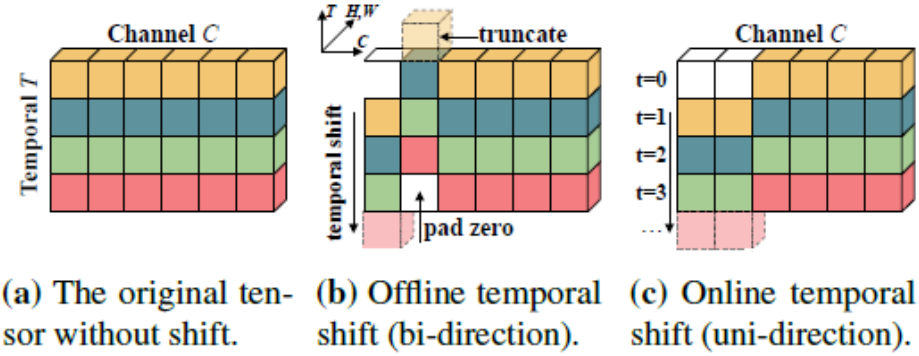

Fig. 2. Temporal modeling performed by TSM, obtained shifting a fraction of the features along the temporal dimension. The offline and online temporal shift are shown. (Figure from [11])

The new network formed by this temporal module was subsequently tested on the most common action datasets.

5 Experiments and results

In this section we describe the used experimental protocol (Datasets and metrics) and we present the results of some results together with some comments on them.

5.1 Datasets and metrics

The datasets considered for the experimental analysis are: UCF101 [23], Olympic Sports [18], and HMDB51 [10], all of which contain actions from real-world scenarios. UCF101 is an action recognition data set of realistic action videos, collected from YouTube, having 101 action categories. It contains 13,320 videos

from 101 action categories. The videos in 101 action categories are grouped into 25 groups, where each group can consist of 4-7 videos of an action. The videos from the same group may share some common features, such as similar background, similar viewpoint, etc. Olympic Sports dataset contains sport activities from YouTube sequences. It contains 16 sport classes, with 50 sequences per class. The HMDB51 dataset is a collection of realistic videos from various sources, including movies and web videos. The dataset is composed of $6,849$ video clips from 51 action categories (such as "jump", "kiss" and "laugh"), with each category containing at least 101 clips.

For the domain adaptation test, only the subsets of overlapping categories across UCF101 and Olympic Sports datasets and the overlapping categories across UCF101 and HMDB51 datasets are used. Therefore, the experimental tests are conducted over two action datasets, namely $UCF-Olympic$ and $UCF-HMDB_{full}$, that are the most commonly used to benchmark the performance of domain adaptation and action recognition algorithms [9].

The labeled target data are used in the learning phase to set an upper bound on how well the model can perform. This experiment will be indicated as *Target-only*, since it is conducted entirely on the target domain. The same can be done with a *Source-only* experiment to obtain a lower bound, which is a result of the absence of adaptation.

Furthermore, the architecture that uses the original TRN as temporal module will be referred to as *TRN-Model*, while the architecture that uses the TSM as temporal module, that is our proposition, will be referred to as *TSM-Model*.

The Accuracy, that is the percentage of correct classified videos, will be used as the performance metric for method comparison.

5.2 Parameters setting details

The data used for this experiments consists of frame-level feature vectors pre-extracted from a ResNet-101 model pre-trained on ImageNet. The number of frame-level feature vectors sampled for each video is fixed to 5. Only RGB inputs are considered for the temporal alignment operation. The stochastic gradient descent (SGD) is utilized as the optimizer with momentum fixed to 0.9. The initial learning rate is 3×10^{-2}, and then it is decreased following the common strategy shown in DANN [5]. The weight decay is 10^{-4} and the batch size is scaled proportionally to the ratio between source and target datasets.

Parameters settings for TRN-Model and TSM-Model are the follows. For TRN-Model, The optimization was conducted as described by Chen et al. [3]: the optimized values of λ^r, λ^t and λ^s, have been found using a coarse-to fine grid search approach. This means that firstly was used a coarse grid with the geometric sequence $[0, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}]$. Then, after finding the optimal range of values, being $[0, 1]$, a new fine-grid search was conducted in this range with the arithmetic sequence $[0, 0.25, 0.50, 0.75, 1]$. The final values found are: 1 for λ^r, 0.5 for λ^t and 0.75 for λ^s. A coarse search was also conducted on the γ value, whose best value is 0.3. For TSM-Model, it should be noted that in this case the λ^r representing the trade-off weighting for relational domain loss, has

no influence, since the domain attention blocks have been removed. The other values of λ^t, λ^s and γ have been found similarly to the other architecture. The final values found are: 0.5 for λ^t, 0.75 for λ^s and 0.7 for γ.

5.3 Results and comments

Source → Target	**U → O**	**Gain**	**O → U**	**Gain**
Source only	80.74		85.83	
TRN-Model	96.30	+5.56	90.42	+4.59
TSM-Model	**96.30**	**+5.56**	**92.08**	**+6.25**
Target only	97.74		94.45	

Table 1. Accuracy (%) for the $UCF - Olympics$ adaptation. Gain represents the absolute difference from the Source only accuracy.

Table 1 shows the accuracy for the $UCF - Olympic$ experiments. As we said before, we show the lower (training only on source data) and upper bound (training only on target data) and the performances of original architecture (TRN-Model) and the proposed one (TSM-Module). We tested the two configuration when UCF101 dataset was the source and target was the Olympic dataset and viceversa. It is clearly shown the gain when we perform a domain adaptation on data for classification and it is clear that our proposition outperform existing one.

Source → Target	**U → H**	**Gain**	**H → U**	**Gain**
Source only	72.5		85.83	
TRN-Model	75.28	+2.78	80.56	+7.71
TSM-Model	**77.22**	**+4.72**	**83.19**	**+10.34**
Target only	85.28		94.57	

Table 2. Accuracy (%) for the $UCF - HMDB_{full}$ adaptation. Gain represents the absolute difference from the Source only accuracy.

Similarly, Table 2 show the results of the $UCF - HMDB_{full}$ experiment. The gain with TSM-Model is even more pronounced.

The results show the importance of the spatial and temporal alignment across domains with high discrepancy. The domain shift is evident from the performance gap between baselines trained exclusively on the source domain and the upper bound defined by the networks trained on target domain data. The new designed architecture, TSM-Model, successfully improves the results of previous works on this application. Overall, the temporal shift allows to better understand the temporal dynamics of the videos.

6 Conclusions

In this paper we propose adapting an existing temporal alignment module to a Domain Adaptation Deep Network for improving the temporal dynamics correspondences between source and target videos. Experimental results show the importance of learning temporal consistency between source and target domains, in order to improve data adaptation within a deep learning context for Action Recognition.

In the future we plan to analyse what is the mutual contribution of spatial and temporal alignment in domain adaptation and we plan to design some new architecture that exploit this knowledge in order to boost current performances.

References

1. Cao, L., Liu, Z., Huang, T.S.: Cross-dataset action detection. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 1998–2005. IEEE (2010)
2. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
3. Chen, M.H., Kira, Z., AlRegib, G., Yoo, J., Chen, R., Zheng, J.: Temporal attentive alignment for large-scale video domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6321–6330 (2019)
4. Davar, N.F., de Campos, T., Windridge, D., Kittler, J., Christmas, W.: Domain adaptation in the context of sport video action recognition. In: Domain Adaptation Workshop, in conjunction with NIPS. University of Surrey (2011)
5. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. The journal of machine learning research **17**(1), 2096–2030 (2016)
6. Jamal, A., Namboodiri, V.P., Deodhare, D., Venkatesh, K.: Deep domain adaptation in action space. In: BMVC. vol. 2–3, p. 5 (2018)
7. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1725–1732 (2014)
8. Kong, Y., Ding, Z., Li, J., Fu, Y.: Deeply learned view-invariant features for cross-view action recognition. IEEE Transactions on Image Processing **26**(6), 3028–3037 (2017)
9. Kong, Y., Fu, Y.: Human action recognition and prediction: A survey. arXiv preprint arXiv:1806.11230 (2018)
10. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: 2011 International conference on computer vision. pp. 2556–2563. IEEE (2011)
11. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7083–7093 (2019)
12. Liu, M., Liu, H., Chen, C.: Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition **68**, 346–362 (2017)

13. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International conference on machine learning. pp. 97–105. PMLR (2015)
14. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Unsupervised domain adaptation with residual transfer networks. Advances in neural information processing systems **29** (2016)
15. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: International conference on machine learning. pp. 2208–2217. PMLR (2017)
16. Luvizon, D.C., Picard, D., Tabia, H.: 2d/3d pose estimation and action recognition using multitask deep learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5137–5146 (2018)
17. Motiian, S., Piccirilli, M., Adjeroh, D.A., Doretto, G.: Unified deep supervised domain adaptation and generalization. In: Proceedings of the IEEE international conference on computer vision. pp. 5715–5725 (2017)
18. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: European conference on computer vision. pp. 392–405. Springer (2010)
19. de Oliveira Silva, V., de Barros Vidal, F., Romariz, A.R.S.: Human action recognition based on a two-stream convolutional network classifier. In: 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA). pp. 774–778. IEEE (2017)
20. Saito, K., Kim, D., Sclaroff, S., Darrell, T., Saenko, K.: Semi-supervised domain adaptation via minimax entropy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8050–8058 (2019)
21. Shu, N., Tang, Q., Liu, H.: A bio-inspired approach modeling spiking neural networks of visual cortex for human action recognition. In: 2014 international joint conference on neural networks (IJCNN). pp. 3450–3457. IEEE (2014)
22. Sigurdsson, G.A., Gupta, A., Schmid, C., Farhadi, A., Alahari, K.: Actor and observer: Joint modeling of first and third-person videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7396–7404 (2018)
23. Soomro, K., Zamir, A.R., Shah, M.: A dataset of 101 human action classes from videos in the wild. Center for Research in Computer Vision **2**(11) (2012)
24. Wang, L., Qiao, Y., Tang, X.: Action recognition with trajectory-pooled deep-convolutional descriptors. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4305–4314 (2015)
25. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Gool, L.V.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016)
26. Wilson, G., Cook, D.J.: A survey of unsupervised deep domain adaptation. ACM Transactions on Intelligent Systems and Technology (TIST) **11**(5), 1–46 (2020)
27. Yeffet, L., Wolf, L.: Local trinary patterns for human action recognition. In: 2009 IEEE 12th international conference on computer vision. pp. 492–497. IEEE (2009)
28. Zhang, H.B., Zhang, Y.X., Zhong, B., Lei, Q., Yang, L., Du, J.X., Chen, D.S.: A comprehensive survey of vision-based human action recognition methods. Sensors **19**(5), 1005 (2019)
29. Zhao, Y., Xiong, Y., Lin, D.: Trajectory convolution for action recognition. Advances in neural information processing systems **31** (2018)