Action Recognition Based on Discriminative Embedding of Actions Using Siamese Networks

Conference Paper · October 2018

DOI: 10.1109/ICIP.2018.8451226

3 authors:

Debaditya Roy
Nihon University
14 PUBLICATIONS **47** CITATIONS
SEE PROFILE

Krishna Mohan Chalavadi
Indian Institute of Technology Hyderabad
57 PUBLICATIONS **351** CITATIONS
SEE PROFILE

Sri Rama Murty Kodukula
Indian Institute of Technology Hyderabad
34 PUBLICATIONS **1,132** CITATIONS
SEE PROFILE

ACTION RECOGNITION BASED ON DISCRIMINATIVE EMBEDDING OF ACTIONS USING SIAMESE NETWORKS

Debaditya Roy, C. Krishna Mohan

VIGIL, Department of Computer Science and Engineering
Indian Institute of Technology Hyderabad
{cs13p1001,ckm}@iith.ac.in

K. Sri Rama Murty

Department of Electrical Engineering
Indian Institute of Technology Hyderabad
ksrm@iith.ac.in

ABSTRACT

Actions can be recognized effectively when the various atomic attributes forming the action are identified and combined in the form of a representation. In this paper, a low-dimensional representation is extracted from a pool of attributes learned in a universal Gaussian mixture model using factor analysis. However, such a representation cannot adequately discriminate between actions with similar attributes. Hence, we propose to classify such actions by leveraging the corresponding class labels. We train a Siamese deep neural network with a contrastive loss on the low-dimensional representation. We show that Siamese networks allow effective discrimination even between similar actions. The efficacy of the proposed approach is demonstrated on two benchmark action datasets, HMDB51 and MPII Cooking Activities. On both the datasets, the proposed method improves the state-of-the-art performance considerably.

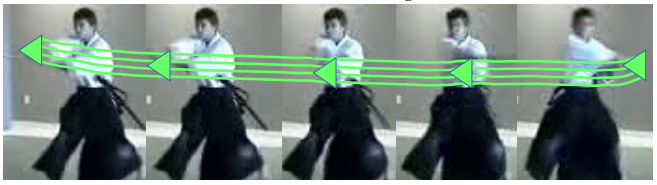
Index Terms— action recognition, Siamese networks, factor analysis

1. INTRODUCTION

Action recognition is the process of discovering similarity across different instances of the same action and realizing the differences across instances of different actions. One of the ways this can be attained is by modelling human actions as a combination of attributes. For example, consider three actions: (a) *running* - composed of attributes like the alternative movement of legs, (b) *cricket bowling* - contains attributes of *running* and *throwing* the ball, and (c) *baseball pitch* - involves only the attributes *throwing* the ball. In order to classify these actions correctly, both shared and unique attributes need to be identified. However, such attributes cannot be identified explicitly for all actions and implicit attribute extraction is required. This paper proposes implicit attribute modeling using a universal attribute model and a factor analysis technique for extracting only necessary attributes for an action. Further, many actions may have very similar attributes that makes it difficult to distinguish them. For example, two different actions like *baseball swing* and *sword exercise* appear visually similar based on the attributes i.e. motion trajectories as shown in Figure 1. In such cases, class labels need to be used to discriminate the actions. This motivates the use of discriminative embedding of attribute based representations using Siamese networks.

Attributes are generally short duration events and in approaches like [16] and [32], sequential modelling of attributes has been proposed. However, manual annotation of events is subjective and building sequential models is not feasible for many fluid actions. In order to alleviate the problem of manual event marking, short-term

(a) *baseball swing*



(b) *sword exercise*

Fig. 1. Inter-action similarity shown with motion trajectory. Best viewed in color.

features are considered such as, improved dense trajectory (iDT) [27] which describes a set of points being tracked across several frames, 3D convolutional neural network (CNN) features which encapsulate a spatio-temporal volume [23], or a set of 2D CNN features from frames and their corresponding optical flow maps [22]. For a single representation of an entire video, aggregation based frameworks like bag-of-words (BoW) [18], Fisher vector [26], and vector of locally aggregated descriptors (VLAD) [17] which do not consider temporal dependency have gained prominence. Fisher vectors have also been extensively used as a feature for standard classifiers such as SVM [27] and feed-forward neural networks [5] to perform action recognition. Especially, Fisher vectors calculated with iDT features have shown good classification performance on large action datasets[26, 25]. An improved VLAD representation [17] has been shown to perform better than Fisher vector on action datasets.

Apart from using classifiers like SVM, low-dimensional discriminative embedding has also shown improvement in classification performance [2, 4]. Especially, Siamese neural networks have been used successfully for the verification of persons [3], objects [31], and gestures [1]. Especially, it shows promise where less number of examples are available as in one-shot recognition of image categories [8, 13]. More recently, in [30], actions were recognized using Siamese networks. Each action was divided into *precondition (cause)* and *effect* parts which are fed as input to separate CNN stream. For the *precondition* part, a 4096-D feature vector is obtained for each frame and average pooled to obtain a single 512-D

feature representation for the entire part. Then, this representation is transformed for each action separately, and the output is compared to the 512-D output of the *effect* part. This approach requires the division of an action video into *precondition* and *effect* which is not only subjective but requires laborious manual annotation.

Given the approaches presented above, we propose a framework which provides a low-dimensional representation of an action video and a discriminative embedding scheme for effective action classification. Short-term features like histogram of oriented gradients (HOG), histogram of optical flow (HOF), and motion boundary histogram (MBH) are used for implicit attribute modelling of all the actions through a Gaussian mixture model. For every clip, a fixed-dimensional super vector is obtained by concatenating the adapted means of the GMM. The super-vector is then decomposed using factor analysis to get a low-dimensional representation for each clip. Then, a Siamese deep neural network is trained using contrastive loss for discriminating actions using the low-dimensional representation. Figure 2 presents a block diagram of the proposed approach.

2. PROPOSED METHOD

We can consider each clip to be a sample function which realizes the random process generating the action. To match the similarity of two action clips, the probability density function (*pdf*) of the clips needs to be matched. If we assume that the underlying *pdf* of the random process can be estimated using a GMM, then the number of mixtures must be sufficiently large to accommodate the intra-action variances in unconstrained videos. Such a GMM called universal attribute model (UAM) can be represented as follows

$$p(\mathbf{x}_l) = \sum_{c=1}^{C} w_c \mathscr{N}(\mathbf{x}_l | \boldsymbol{\mu}_c, \boldsymbol{\sigma}_c), \tag{1}$$

where the mixture weights w_c satisfy the constraint $\sum_{c=1}^{C} w_c = 1$ and $\boldsymbol{\mu}_c, \boldsymbol{\sigma}_c$ are the mean and covariance for mixture c of the UAM, respectively. A feature $\mathbf{x}_l$ is part of a clip $\mathbf{x}$ represented as a set of feature vectors $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_L$. This feature can be either a HOF or an MBH descriptor and we train a separate UAM for each during evaluation using standard EM estimation.

As the goal is to find the *pdf* of the action that generates a clip, we need to adapt the UAM parameters using the features extracted from the clip. We perform a maximum *aposteriori* (MAP) adaptation similar to [19, 10] for obtaining the requisite *pdf* which describes the clip. The adapted means for each mixture are then concatenated to compute a $(Cd \times 1)$-dimensional super action-vector for each clip represented as

$$\mathbf{s}(\mathbf{x}) = [\hat{\boldsymbol{\mu}}_1 \hat{\boldsymbol{\mu}}_2 \cdots \hat{\boldsymbol{\mu}}_C]^t. \tag{2}$$

2.1. Low-dimensional representation using factor analysis

The super action-vector (SAV) still contains unmodified means of those attributes which do not contribute to the action. In order to remove those attributes and arrive at a low-dimensional representation, the SAV $\mathbf{s}$ is decomposed as

$$\mathbf{s} = \mathbf{m} + \mathbf{T}\mathbf{w}, \tag{3}$$

where $\mathbf{m}$ is the actor and viewpoint independent component of SAV that can be initialized using the UAM supervector as the UAM is trained using large number of actors and viewpoints resulting in a distribution that is marginalized over views and actors. The part $\mathbf{T}\mathbf{w}$ denotes the actor and view dependent component of the SAV. The matrix $\mathbf{T}$ is a low-rank rectangular matrix known as the total variability matrix of size $Cd \times r$ and a $\mathbf{w}$ is the r-dimensional low-dimensional representation having a standard normal prior distribution $\mathscr{N}(\mathbf{0}, \mathbf{I})$ where $\mathbf{I}$ is $d \times d$-dimensional identity matrix [6]. For a given clip, $\mathbf{w}$ can be defined as a total factor of the adapted SAV and it can be obtained using factor analysis.

To find the low-dimensional vector for a given clip, the posterior mean and covariance of $\mathbf{w}$ given a video clip $\mathbf{x}$ are calculated using the EM algorithm as in [12]. The posterior distribution of $\mathbf{w}$ given a video clip is conditioned on the means of the UAM. Hence, the centered first-order Baum-Welch statistics are calculated as

$$\tilde{\mathbf{F}}_c(\mathbf{x}) = \sum_{l=1}^{L} p(c|\mathbf{x}_l)(\mathbf{x}_l - \boldsymbol{\mu}_c), c = 1, 2, \cdots, C \tag{4}$$

After the final M-step in the EM algorithm, the low-dimensional vector for a given clip is represented using the mean of its posterior distribution as follows

$$\mathbf{w}(\mathbf{x}) = (I + \mathbf{T}^t \boldsymbol{\Sigma}^{-1} \mathbf{N}(\mathbf{x}) \mathbf{T})^{-1} \mathbf{T}^t \boldsymbol{\Sigma}^{-1} \tilde{\mathbf{s}}(\mathbf{x}). \tag{5}$$

where $\boldsymbol{\Sigma}$ is a diagonal covariance matrix of dimension estimated during factor analysis training (see [12]) and models the residual variability not captured by the total variability matrix $\mathbf{T}$. The matrix $\mathbf{N}(\mathbf{x})$ is a diagonal matrix of dimension $Cd \times Cd$ whose diagonal blocks are $n_c(\mathbf{x})\mathbf{I}$, for $c = 1, ..., C$, and the supervector $\tilde{\mathbf{s}}(\mathbf{x}) = [\tilde{\mathbf{F}}_1(\mathbf{x}) \tilde{\mathbf{F}}_2(\mathbf{x}) \cdots \tilde{\mathbf{F}}_C(\mathbf{x})]^t$.

The $\mathbf{T}$-matrix contains the eigenvectors of the largest r factors (eigenvalues) of the total variability covariance matrix [6]. Though these low-dimensional vectors can be used directly for comparing the similarity between actions using cosine scoring without the use of labels, it cannot account for inter-action similarity. Inter-action similarity is caused when two different actions appear visually similar. We hypothesize that using action labels can mitigate these issues and propose to use Siamese networks which employ contrastive loss to increase inter-action variance.

2.2. Discriminative embedding using Siamese network

Siamese networks are trained to differentiate between inputs rather than classify inputs. In this work, the low-dimensional representation of a video is used for differentiating between videos of different classes. The Siamese network consists of two identical sister neural networks with the same weights, each of which takes in the representation for a clip. The last layers of the two networks culminate in a contrastive loss function which evaluates the similarity between the two videos.

The Siamese network is optimized using a contrastive loss computed as

$$(1 - y)\frac{1}{2} e_w^2 + (y)\frac{1}{2}\{max(0, a - e_w)\}^2, \tag{6}$$

where $e_w = \|\mathbf{g}_w(\mathbf{w}_1) - \mathbf{g}_w(\mathbf{w}_2)\|_2$ is the Euclidean distance between the two outputs of the sister networks and is a measure of the semantic similarity between the inputs. The term $\mathbf{g}_w$ is the output of either one of the sister networks and $\mathbf{w}_1$ and $\mathbf{w}_2$ are the inputs. The label y is either 1 if the inputs are from the same class and 0 otherwise and the margin a is set to be greater than 0 so that dissimilar pairs beyond this margin do not contribute to the loss. This ensures that the network is optimized for dissimilar pairs that the networks consider as fairly similar. For training the network, the contrastive

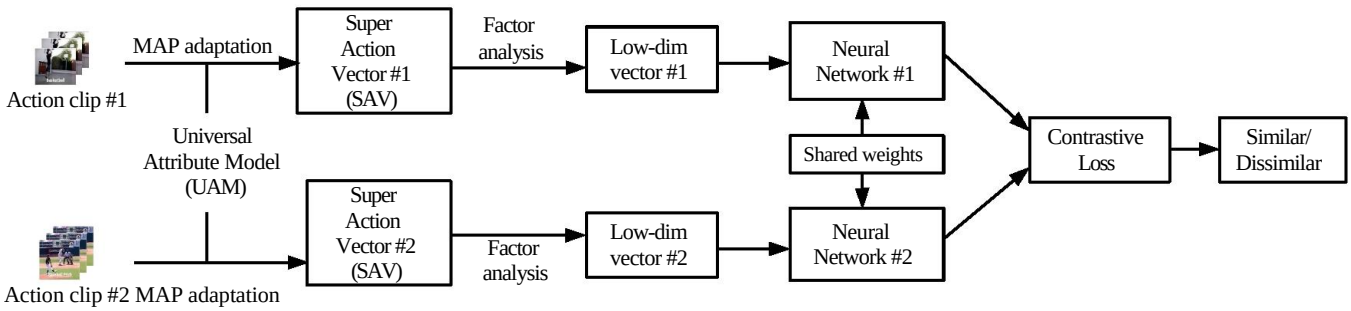


Fig. 2. Proposed architecture for action recognition

loss value is calculated using both the inputs, and then back propagated to both the networks.

3. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of Siamese networks for action recognition on the following datasets.

- *HMDB51* dataset [14] consists of 6766 video clips from 51 actions such as eating, smiling, clapping, bike riding, shaking hands, etc. Each clip is approximately 3 seconds long and recorded at 30 frames per second.
- *MPII Cooking Activities* dataset [21] contains 5,609 videos of 65 fine-grained cooking activities with low inter-class variability. Each clip is 5 seconds long and recorded at 30 frames per second and all the activities are performed by 12 different subjects.

The results reported here for both the datasets are based on the training and testing splits provided on the official websites.

3.1. Experimental settings

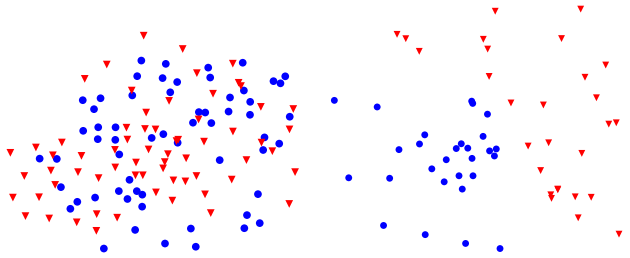
The three configurations used for each sister neural network in the Siamese architecture are NN1: $200 - 100R$, NN2: $200 - 100R - 50R$, and NN3: $200 - 100R - 100R$ where R represents ReLU activation. The input action-vector is calculated using a UAM with 256 mixtures trained on the training split of HMDB51. The NN3 configuration achieves the best results among the three overall feature descriptors as shown in Table 1 and it is used for reporting the results in subsequent experiments. We use a dropout of 0.1 between the layers except for the output layer, and the learning rate was set to 0.0001 with *RMSprop* used as an optimizer. It is found empirically that increasing the number of layers or neurons caused significant deterioration of classification performance. We hypothesize that as the number of training instances per action in both HMDB51 and is around 70, networks with more number of parameters cannot be trained effectively.

The HOG, HOF and MBH features are extracted using the same spatio-temporal volumes used in improved dense trajectory (iDT)[26]. It is found that changing the feature descriptor and the number of mixtures produces a significant change in the classification performance. However, any change in the action-vector dimension does not result in any significant difference in accuracy, and the value of 200 was chosen as the dimension as it performed marginally better empirically.

In Figure 3, we present a case where Siamese networks are able to differentiate between two visually similar actions *baseball swing* and *sword exercise* (as shown in Figure 1). It can be observed that action-vectors (200 dimensional) for both actions are interspersed. On the other hand, the Siamese network 1^{st} hidden layer output (100 dimensions - NN3 configuration) is clearly separable for these actions.

Table 1. Performance (in %) of Siamese network with different configurations on HMDB51 dataset

Configuration	HOG	HOF	MBH
NN1	42.1	45.9	49.6
NN2	57.8	59.6	62.4
NN3	72.1	78.5	**83.1**



(a) action-vector (b) Siamese 1^{st} hidden layer

Fig. 3. t-SNE (stochastic neighbour embedding) plots for *baseball swing* (▼) and *sword exercise* (●). Best viewed in color.

3.2. Comparison with embedding techniques

In Tables 2 and 3, a comparison of Siamese networks with linear embedding techniques such as linear discriminant analysis (LDA) and probabilistic LDA (PLDA) [20] and non-linear embedding techniques like kernel discriminant analysis (kDA) is presented. It can be observed that on both HMDB51 and MPII cooking dataset, Siamese networks based embedding of action-vectors performs better than other linear and non-linear embedding techniques. This is also true across the different feature vectors and different UAMs with a varying number of mixtures. The results are reported for 256, and 512 Gaussian mixtures as any increase in the number of mixtures did not improve classification performance but incurred significant training time.

In Table 4, we present the performance of Siamese networks with state-of-the-art techniques on the HMDB51 dataset. Temporal segment networks (TSN) [29], max pooled deep features (CNN +C3D) and iDT features [4], two-stage temporal segment net-

Table 2. Accuracy (%) of various discriminative embedding techniques on HMDB51

Embedding Technique	# UAM mixtures					
	256			512		
	HOG	HOF	MBH	HOG	HOF	MBH
LDA	74.55	77.32	78.56	73.98	75.95	77.52
PLDA	76.24	78.17	79.54	74.21	76.67	78.69
kDA	61.45	65.16	66.12	69.51	70.53	72.75
Siamese (NN3)	72.14	78.53	**83.10**	76.41	79.35	83.04

Table 3. Accuracy (%) for various discriminative embedding techniques on MPII Cooking

Embedding Technique	# UAM mixtures					
	256			512		
	HOG	HOF	MBH	HOG	HOF	MBH
LDA	74.14	77.48	78.48	76.45	79.48	78.61
PLDA	65.34	67.56	66.85	69.12	67.35	67.63
kDA	63.45	64.16	65.21	64.51	65.53	67.75
Siamese (NN3)	75.48	76.95	78.46	77.81	79.24	**80.27**

works [15], and iDT features with deep neural networks (DNN) [5] all provide a single feature representation for a video by utilizing spatial, temporal and spatio-temporal CNNs, and further augment it with iDT features. It can be observed that discriminative embedding of action-vectors using a single feature descriptor MBH which is part of the iDT features is more effective in action recognition than these methods. This shows that factor analysis seems a better choice for discriminative action representation and following it with contrastive loss training aids in classification even further.

In Table 5, a comparison with state-of-the-art approaches is presented for MPII cooking activities dataset. Approaches like interaction part mining [33], localized semantic features [34], etc. concentrate on object detection, and its manipulation to recognize the actions. Instead in [2], frame-based CNN features from a video are embedded on Grassmannian manifold. A support vector regressor is learned on the embedded features augmented with iDT features for classification. It can be observed that the proposed Siamese network based embedding scheme performs better than all these embedding and object interaction based approaches.

Table 4. Comparison with state-of-the-art on HMDB51

Method	**Accuracy** (in %)
3DCNN features + Siamese [30]	63.4
Traj-pooled deep CNN + iDT(fisher) [28]	65.9
Long-term CNNs +iDT [24]	67.2
Temporal segment networks (TSN) [29]	69.4
iDT-FV + DNN Hybrid [5]	70.4
Temporal linear embedding (TLE) [7]	71.1
Max-pool (CNN+C3D) + iDT[4]	73.1
Deep local video feature [15]	75.0
Action-vector (MBH) + Siamese	**83.1**

Table 5. Comparison with state-of-the-art on MPII Cooking

Method	**Accuracy (in %)**
CoHOG +iDT [11]	46.6
Localized Semantic Features [34]	70.5
VideoDarwin [9]	72.0
Interaction Part Mining [33]	72.4
GRP-CNN + iDT(Fisher) [2]	75.7
Action-vector (MBH) + Siamese	**80.3**

4. CONCLUSION

In this paper, we explored discriminative embedding of low-dimensional action representations to distinguish between visually similar actions. A Siamese deep neural network was trained with action-vectors as inputs, and it was shown that actions with similar attributes were distinguishable. As compared to other linear and non-linear discriminative embedding techniques, Siamese networks were shown to produce better classification performance. Finally, on benchmark datasets like HMDB51 and MPII Cooking Activities, the proposed method produces state-of-the-art recognition performance as compared to existing approaches.

5. REFERENCES

[1] S. Berlemont, G. Lefebvre, S. Duffner, and C. Garcia. Siamese neural network based similarity metric for inertial gesture classification and rejection. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–6, May 2015. 1

[2] A. Cherian, B. Fernando, M. Harandi, and S. Gould. Generalized rank pooling for activity recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2017. 1, 4

[3] D. Chung, K. Tahboub, and E. J. Delp. A two stream siamese convolutional neural network for person re-identification. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1

[4] I. Cosmin Duta, B. Ionescu, K. Aizawa, and N. Sebe. Spatio-temporal vector of locally max pooled features for action recognition in videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 3, 4

[5] C. R. de Souza, A. Gaidon, E. Vig, and A. M. López. *Sympathy for the Details: Dense Trajectories and Hybrid Classification Architectures for Action Recognition*, pages 697–716. Springer International Publishing, Cham, 2016. 1, 3, 4

[6] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, May 2011. 2

[7] A. Diba, V. Sharma, and L. Van Gool. Deep temporal linear encoding networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 4

[8] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, April 2006. 1

[9] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars. Modeling video evolution for action recogni-

tion. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 4

[10] N. Inoue and K. Shinoda. A fast and accurate video semantic-indexing system using fast map adaptation and gmm supervectors. *IEEE Transactions on Multimedia*, 14(4):1196–1205, Aug 2012. 2

[11] H. Kataoka, K. Hashimoto, K. Iwata, Y. Satoh, N. Navab, S. Ilic, and Y. Aoki. Extended co-occurrence hog with dense trajectories for fine-grained activity recognition. In D. Cremers, I. Reid, H. Saito, and M.-H. Yang, editors, *Computer Vision – ACCV 2014*, pages 336–349, Cham, 2015. Springer International Publishing. 4

[12] P. Kenny, G. Boulianne, and P. Dumouchel. Eigenvoice modeling with sparse training data. *IEEE transactions on speech and audio processing*, 13(3):345–354, 2005. 2

[13] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015. 1

[14] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011. 3

[15] Z. Lan, Y. Zhu, A. G. Hauptmann, and S. Newsam. Deep local video feature for action recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1219–1225. IEEE, 2017. 3, 4

[16] K. Li, J. Hu, and Y. Fu. Modeling complex temporal composition of actionlets for activity prediction. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part I*, ECCV'12, pages 286–299, Berlin, Heidelberg, 2012. Springer-Verlag. 1

[17] Y. Li, W. Li, V. Mahadevan, and N. Vasconcelos. Vlad3: Encoding dynamics of deep features for action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1951 – 1960, June 2016. 1

[18] X. Peng, L. Wang, X. Wang, and Y. Qiao. Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *Computer Vision and Image Understanding*, 150:109 – 125, 2016. 1

[19] F. Perronnin. Universal and adapted vocabularies for generic visual categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1243–1256, July 2008. 2

[20] S. J. Prince and J. H. Elder. Probabilistic linear discriminant analysis for inferences about identity. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007. 3

[21] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1194–1201. IEEE, 2012. 3

[22] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 568–576. Curran Associates, Inc., 2014. 1

[23] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014. 1

[24] G. Varol, I. Laptev, and C. Schmid. Long-term Temporal Convolutions for Action Recognition. *arXiv:1604.04494*, 2016. 4

[25] G. Varol and A. A. Salah. Extreme learning machine for large-scale action recognition. In *ECCV workshop*, 2014. 1

[26] H. Wang, D. Oneata, J. Verbeek, and C. Schmid. A robust and efficient video representation for action recognition. *International Journal of Computer Vision*, 119(3):219–238, July 2015. 1, 3

[27] H. Wang and C. Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3551–3558, 2013. 1

[28] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4305–4314, 2015. 4

[29] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. *Temporal Segment Networks: Towards Good Practices for Deep Action Recognition*, pages 20–36. Springer International Publishing, Cham, 2016. 3, 4

[30] X. Wang, A. Farhadi, and A. Gupta. Actions transformations. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1, 4

[31] X. Wang and A. Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2015. 1

[32] J. Wu and D. Hu. Learning effective event models to recognize a large number of human actions. *IEEE Transactions on Multimedia*, 16(1):147–158, Jan 2014. 1

[33] Y. Zhou, B. Ni, R. Hong, M. Wang, and Q. Tian. Interaction part mining: A mid-level approach for fine-grained action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 4

[34] Y. Zhou, B. Ni, S. Yan, P. Moulin, and Q. Tian. Pipelining localized semantic features for fine-grained action recognition. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 481–496, Cham, 2014. Springer International Publishing. 4