

Flow Dynamics Correction for Action Recognition

Author

Wang, L, Koniusz, P

Published

2024

Conference Title

ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

Version

Accepted Manuscript (AM)

DOI

[10.1109/ICASSP48485.2024.10446223](https://doi.org/10.1109/ICASSP48485.2024.10446223)

Rights statement

This work is covered by copyright. You must assume that re-use is limited to personal use and that permission from the copyright owner must be obtained for all other uses. If the document is available under a specified licence, refer to the licence for details of permitted re-use. If you believe that this work infringes copyright please make a copyright takedown request using the form at <https://www.griffith.edu.au/copyright-matters>.

Downloaded from

<https://hdl.handle.net/10072/435601>

Griffith Research Online

<https://research-repository.griffith.edu.au>

FLOW DYNAMICS CORRECTION FOR ACTION RECOGNITION

Lei Wang^{*,†,§} Piotr Koniusz^{§,†}

[†]Australian National University, [§]Data61/CSIRO

ABSTRACT

Various research studies indicate that action recognition performance highly depends on the types of motions being extracted and how accurate the human actions are represented. In this paper, we investigate different optical flow, and features extracted from these optical flow that capturing both short-term and long-term motion dynamics. We perform power normalization on the magnitude component of optical flow for flow dynamics correction to boost subtle or dampen sudden motions. We show that existing action recognition models which rely on optical flow are able to get performance boosted with our corrected optical flow. To further improve performance, we integrate our corrected flow dynamics into popular models through a simple hallucination step by selecting only the best performing optical flow features, and we show that by ‘translating’ the CNN feature maps into these optical flow features with different scales of motions leads to the new state-of-the-art performance on several benchmarks including HMDB-51, YUP++, fine-grained action recognition on MPII Cooking Activities, and large-scale Charades.

Index Terms— optical flow, power normalization, flow correction, hallucination, action recognition

1. INTRODUCTION

The motion cues for Action Recognition (AR) [26, 10, 14, 29, 28, 30, 32, 23] can be extracted from multiple resources, *e.g.*, RGB videos, depth videos, 3D point clouds, skeleton sequences, and optical flow videos. A thorough comparison of using different kinds of features for AR can be found in review papers [22, 25, 23]. Many state-of-the-art (SOTA) AR methods, apart from the use of RGB frames, rely on some form of optical flow which comes in many flavours. The introduction of optical flow estimation [36, 1, 34, 15] led to a dramatic boost of performance in many areas of AR. TV-L1 [36] preserves the discontinuities in the flow field, and provides an increased robustness in terms of occlusions, illumination changes and noise. LDOF [1] integrates rich descriptors into the variational optical flow setting to cope with large displacements. DeepFlow [34] boosts the performance w.r.t. fast motions by employing LDOF with a descriptor matching within a multi-stage architecture. EpicFlow [15] targets large



(a) *Marathon*: stride = 1, 4, 8 and 12 respectively (from left to right).



(b) *Kick ball*: stride = 1, 2 and 4.

(c) *Situp*: stride = 1, 2 and 4.

Fig. 1. Multi-stride optical flow (LDOF) on (a) YUP++ and (b) – (c) HMDB-51. Different strides (temporal scales) can capture different granularity levels of motions, and the visual appearance varies between different temporal scales.



(a) DeepFlow: $\gamma = 0.1, 0.5$ and 5 .

(b) TV-L1: $\gamma = 0.1, 0.5$ and 5 .



(c) EpicFlow: $\gamma = 0.1$. (d) EpicFlow: $\gamma = 0.5$. (e) EpicFlow: $\gamma = 5$.

Fig. 2. (a)–(b) show the strength of PN (γ) for optical flow correction on action *Kick ball*. Small γ preserves the dominant motions and large γ boosts some weak motions and maintains more rich motion dynamics. Each pair of figures in (c) – (e) shows with (left) and without (right) dominant motions on action *dribble*. All actions are from HMDB-51.

displacements with significant occlusions through a dense matching by edge-preserving interpolation from a sparse set of matches. These optical flow computation methods are quite mature and widely used in practice, hence they are of our interest for further investigations.

However, in general, articulated motion and human motion in particular are problematic. Some human body parts, *i.e.*, hands can move very fast, whereas other parts may follow a slower motion pattern. Indeed, different motion speeds likely introduce nuisance variations that contribute to poorer recognition results. We perform power normalization on the magnitude component of optical flow for the correction of flow dynamics to boost subtle or dampen sudden motions. We show that with our corrected flow dynamics, handcrafted IDT descriptors [20], popular two-stream network [7], I3D [2] and even AssembleNet/AssembleNet++ [18, 17] are able to im-

* This paper has been accepted for IEEE ICASSP 2024.

prove the AR performance by 3 – 5% on average. Since these SOTA AR approaches rely on optical flow estimation methods to pre-compute motion information for CNNs and such a two-stage method is computationally expensive, storage demanding, and not end-to-end trainable, we also propose simple trainable CNN streams on top of a CNN network (*e.g.*, I3D [2] and AssembleNet++ [17]) that learn to ‘translate’ the RGB output into several OFFs that are extracted at different scales to form the short-term and long-term motion dynamics. Different OFFs are not only synthesized but they also provide self-supervisory signals. Our main contributions are:

- i. We show that correcting optical flow maps by so-called power normalization (PN) produces various motion dynamics that dampen sudden motions or noise and magnify tiny motions.
- ii. We investigate various aspects of our model, *e.g.*, different kinds of optical flow or scales of motion (short-term and long-term motions). With the corrected flow dynamics, our model outperforms previous approaches on 4 benchmarks including dynamic scenes classification and fine-grained AR by a large margin.
- iii. We introduce a Selector for selecting the best corrected motion dynamics to learn the feature streams. We also show that different optical flow features (OFFs) extracted from either short-term or long-term motion dynamics can be synthesized implicitly to handle various speeds and dynamics of actions.

2. APPROACH

2.1. Flow Dynamics Correction

Multi-stride optical flow computation. We choose TV-L1 [36], LDOF [1], DeepFlow [34] and EpicFlow [15] because (i) they are often used in video classification tasks and (ii) they cope with large displacements, occlusions, and small motions. Setting stride = 1 is the most common setting that is widely used in the optical flow computations to capture the temporal information. In this work, we explore the effects of using different strides for optical flow computations.

We let the stride step take values between one and the average number of frames in each dataset to form different scales of motion dynamics. On HMDB-51 and YUP++, we use stride = 1, 2, 4, 6, 8, 12, 15, 30, 45 for all 4 optical flow computations. If the selected stride value is greater than the total number of frames in a given video, we drop this stream as the temporal information can be captured later in the shorter-term streams with smaller strides. Fig. 1 shows some visualizations of LDOF with different strides to form different motion dynamics on YUP++ and HMDB-51. As shown in these figures, the multi-stride flow dynamics are very different from the original optical flow computed by the use of common setting (stride = 1). Optical flow computed at different temporal scales are able to capture motion dynamics

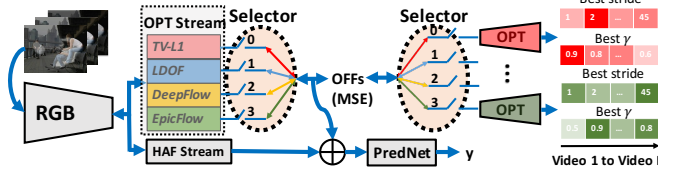


Fig. 3. We use optical flow (OPT) streams and a Selector to learn to hallucinate the best optical flow features (OFFs). The OFFs and features from the High Abstraction Features (HAF) stream are concatenated by \oplus , and then feed into the PredNet (a simple MLP) for classification. The Selector is used to choose the optical flow types we learn to hallucinate, given the best OFFs. MSE represents the mean square error loss, while y denotes the output class label from PredNet.

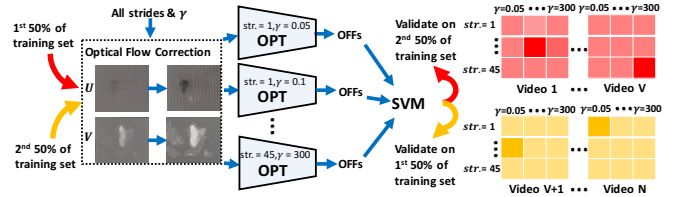


Fig. 4. Our stride and γ selector. Pre-selection of best (stride, γ) per optical flow type per video.

at different granularity levels, and the visual appearances in these optical flow are different; hence, our multi-stride optical flow can capture more rich motion and related appearance information for downstream video processing tasks.

Optical flow correction. Let \mathbf{U} and \mathbf{V} be two maps with the displacement components (along x and y axis, respectively) of the computed multi-stride optical flow. The magnitude and angle of the optical flow (\mathbf{U}, \mathbf{V}) are computed (by element-wise operations) as

$$M = \sqrt{\mathbf{U}^2 + \mathbf{V}^2}, \quad (1)$$

$$\Phi = \arctan(\mathbf{U}/\mathbf{V}). \quad (2)$$

As videos are highly affected by many issues like noise, camera shaking, dynamic background environments and a mixture of fast and slow motions, *e.g.*, human actions, we apply the element-wise power normalization (PN), on the magnitude component M for the flow correction to get the power normalized magnitude matrix M'

$$M' = \text{sign}(M) \cdot (1 - (1 - \text{abs}(M))^\gamma), \quad (3)$$

where $\gamma > 0$ decides the strength of PN, and all operations are element-wise. The PN here is used for the flow correction that is performed on each optical flow frame. The normalization is done on the magnitude component of the optical flow so as to boost or dampen subtle or sudden motions. We then compute optical flow features (OFFs) from such mended motion clips. The use of abs and sign in Eq. (3) is for maintaining the motion direction. We use $\gamma > 1$ to boost weak and dampen dominant motions (*c.f.* $0 < \gamma < 1$ to preserve only dominant motions) which gives us selective focus on various motion dynamics. Note that if $\gamma = 1$, PN is not performed.

Finally, we recover two optical flow maps (\mathbf{U}' , \mathbf{V}') based on the corrected M' and Φ as

$$\mathbf{U}' = M' \cdot \sin(\Phi), \quad (4)$$

$$\mathbf{V}' = M' \cdot \cos(\Phi). \quad (5)$$

Fig. 2 shows some visualizations of corrected flow dynamics on HMDB-51. The color intensity shows the effects of power normalization with different γ values. We notice that smaller γ preserves mainly the dominant motions whereas large γ boosts some weak motions and keeps more rich and fine-grained information.

2.2. Stride and γ selector

We introduce a lightweight hallucination¹ model (HAL) inspired by [31, 27]; however, our HAL only has the optical flow streams built on top of a backbone network. The input to our HAL is the RGB video, and it learns (during training) to translate latent features from RGB into OFFs, which represent various motion dynamics based on optical flow. Our pipeline uses the corrected flow dynamics illustrated in Fig. 3. There are 4 switches that activate the corresponding optical flow streams based on the selection made by the Selector.

Figure 4 shows our stride and γ selector. Given the corrected optical flow, we split train data into two halves. We train scoring optical flow networks (e.g., I3D or AssembleNet/AssembleNet++ optical flow stream pre-trained on Kinetics-400), one per optical flow type, stride choice and γ choice. We train on one half of train data, and score via SVM each video on the second half of train data in terms of which (stride, γ) recognises video correctly (or is the closest to correct decision). Then, we train networks on the second half of the train data and score videos on the first half. With such scoring, we can train four optical flow networks by directing to them best (stride, γ) per video. We choose (i) the best performing optical flow feature for optimal (stride, γ) per optical flow type or (ii) only one best performing optical flow type to hallucinate thus preventing the overparametrization (Selector uses pre-scores to choose also the best optical flow type, so we pre-select (type, stride, γ) per video). As a result, our proposed method is able to generate the better OFFs without the need of optical flow computation during the test stage. Due to the optical flow type and best (stride, γ) selectors, the network generates features with the best motion dynamics per video rather than static features from one kind of optical flow and fixed stride.

3. EXPERIMENTS

3.1. Datasets and Protocols

We evaluate the use of flow dynamics correction in popular action recognition models on 4 benchmarks: HMDB-51 [11],

¹‘Hallucination’ conveys the model’s ability to generate video representations during the test stage, making them available without the original, time-consuming computation and processing steps.

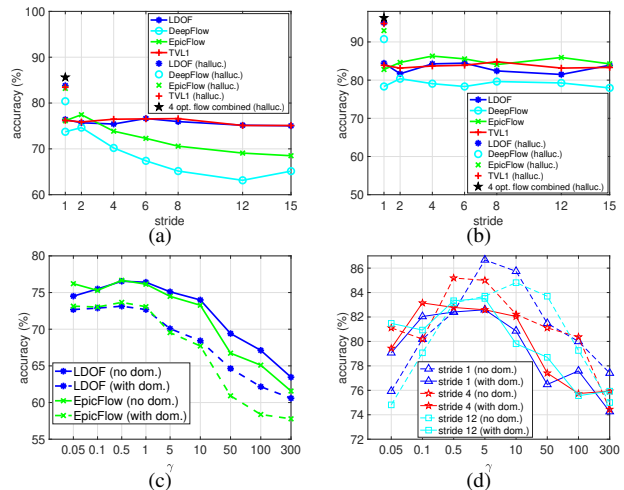


Fig. 5. (Top row) Evaluations of OFFs w.r.t. different scales of motions on (a) HMDB-51 (split 1) and (b) YUP++ (*static camera*). (Bottom row) Evaluations of PN with/without the use of dominant motions w.r.t. (c) different optical flow on HMDB-51 and (d) different scales of motions on YUP++.

YUP++ [6], MPII Cooking Activities [16] and Charades [19]. Using standard protocols, we report recognition/classification accuracy (%) for HMDB-51 and YUP++, mean average precision (mAP) for MPII and Charades. First, we use our HAL with flow dynamics correction for ablation studies, and then we compare our method versus the SOTA methods.

3.2. Ablation Study

Flow estimation quality. Fig. 5 (top) shows the comparison of using different OFFs on HMDB-51 and YUP++. As shown in Fig. 5(a), the accuracies of using OFFs extracted from DeepFlow and EpicFlow decrease when long-term motion is used, whereas the TV-L1 and LDOF perform better in terms of both short-term and long-term motions. For natural dynamic scenes classification (see Fig. 5(b)), all OFFs perform almost equally well. This is mainly because the motions in natural dynamic scenes are generally periodic, whereas human actions are far more complicated in terms of the dynamics in different body parts. After integrating each optical flow feature into our hallucination pipeline, the performance increases by $\sim 9\%$ on HMDB-51 and $\sim 10\%$ on YUP++, respectively. Hallucinating all 4 OFFs further improves the performance by 2 – 3% on both datasets.

With dominant motions. Fig. 5 (bottom) shows the performance comparison between with and without the use of dominant motions, i.e., dominant magnitude of M is subtracted from it before PN. For AR on HMDB51 (see Fig. 5(c)), using the dominant motions (with dom.) does not improve the overall recognition accuracies, and the performance for the use of LDOF and EpicFlow drops by 2-5%. HMDB-51 is a challenging dataset with videos captured by moving cameras with noise, especially in sports-related actions, making environment-dependent actions a significant factor. Relying

	sp1	sp2	sp3	mean acc.
all 4 opt. flow (stride=1)	83.5	83.5	83.5	83.5
all 4 opt. flow (best stride)	85.6	85.2	85.5	85.4
all 4 opt. flow (corrected)	87.5	86.7	87.5	87.3
best opt. flow only (corrected)	86.9	86.8	86.8	86.8

Table 1. Evaluations of our HAL variants on HMDB-51. sp1, sp2, and sp3 denote three splits.

	IDT-FV [20]	Two-stream net. [7]	I3D [2]	DEEP-HAL [31]	ODF+SDF [27]	HAL (ours)
Original	57.2	69.2	80.9	82.5	87.6	83.5
Flow Corr. (ours)	60.7	77.8	83.0	85.0	89.0	87.3
Improvement	↑3.5	↑8.6	↑2.1	↑2.5	↑1.4	↑3.8

EvaNet [12] 82.3 BIKE [35] 84.3 SCK+DCK [10] 86.1 VideoMAE V2 [24] 88.1

Table 2. Evaluations of various methods (*top*) w/w/o our flow dynamics correction and (*bottom*) comparisons to the state of the art on HMDB-51.

solely on dominant motions is insufficient for reliable action classification. For natural dynamic scenes classification on the YUP++ dataset (see Fig. 5(d)), using dominant motions (with dom.) when $\gamma \leq 0.5$ performs slightly better, as it filters out static camera motion.

With Selector. Tab. 1 shows the evaluations of our HAL variants. We first choose stride = 1 for all 4 optical flow types and hallucinate all 4 OFFs on the HMDB-51 dataset. Note that this is the default setting which is widely used where the optical flow computation is done on two consecutive frames (we set it as baseline for comparison). As in Tab. 1, choosing the best stride per optical flow (*all 4 opt. flow (best stride)*) performs better than using the common setting (stride = 1) for all 4 OFFs by $\sim 2\%$. Hallucinating the best stride and γ per optical flow (*all 4 opt. flow (corrected)*) performs better than (*all 4 opt. flow (best stride)*) by $\sim 1.8\%$. We also hallucinate the top performing optical flow feature choosing from all 4 OFFs by using our stride and γ selector (*best opt. flow only (corrected)*), and the overall accuracy on HMDB51 is quite close to (*all 4 opt. flow (corrected)*). Note that (*best opt. flow only (corrected)*) only hallucinate one best OFF, whereas (*all 4 opt. flow (corrected)*) hallucinate 4 best OFFs and each best optical flow feature is chosen from each optical flow.

For the rest experiments, by default, we choose to hallucinate the top performing optical flow feature selected from all 4 OFFs by using our selector (*best opt. flow only (corrected)*).

3.3. Comparisons With the SOTA Methods

Tab. 2 shows the results on HMDB-51. With corrected flow dynamics (denoted as *Flow Corr.*), IDT-FV outperforms the use of original optical flow by 3.5%. The use of optical flow correction on two-stream network boosts the performance by more than 8%. Although our HAL uses only the optical flow streams, it still achieves very competitive results compared to its similar competitors, e.g., DEEP-HAL and ODF+SDF.

Tab. 3 shows the results on YUP++. We notice that our HAL performs equally well compared to DEEP-HAL even without the use of flow dynamics correction, and with our corrected optical flow, it outperforms the baseline method (DEEP-HAL) by $\sim 2\%$. Our method also outperforms more complex T-ResNet and MSOE (two-stream) by $> 2\%$.

	Two-stream net. [7]	I3D [2]	ADL I3D [21]	DEEP-HAL [31]	HAL (ours)
Original	92.0 / 91.9	89.9 / -	91.7 / -	92.2 / 92.6	92.4 / 92.6
Flow Corr. (ours)	92.4 / 92.8	90.3 / -	92.2 / -	92.4 / 92.8	94.3 / 94.2
Improvement	↑0.9	↑0.4	↑0.5	↑0.2	↑1.9

T-ResNet [6] 87.0 / 87.6 MSOE(two-stream) [8] 92.0 / 91.9

Table 3. Evaluations of various methods (*top*) w/w/o our flow dynamics correction and (*bottom*) comparisons to the state of the art on YUP++. We report *mean over stat.&dyn.* / *mean over all (stat.&dyn.&mixed)*.

	IDT-FV [20]	I3D [2]	DEEP-HAL [31]	ODF+SDF [27]	HAL (ours)
Original	67.6	74.8	81.8	84.8	82.8
Flow Corr. (ours)	74.0	80.4	83.5	86.2	86.2
Improvement	↑6.4	↑5.6	↑1.7	↑1.4	↑3.4

KRP-FS [4] 70.0 KRP-FS+IDT [4] 76.1 GRP [3] 68.4 GRP+IDT [3] 75.5

Table 4. Evaluations of various methods (*top*) w/w/o our flow dynamics correction and (*bottom*) comparisons to the state of the art on MPII.

	I3D [2]	DEEP-HAL [31]	AssembleNet [18]	AssembleNet++ [17]	HAL (ours, (with I3D))	HAL (ours, AssembleNet++)
Original	40.0	43.1	56.6	59.8	45.3	62.0
Flow Corr. (ours)	42.1	45.7	59.7	62.0	48.7	64.9
Improvement	↑2.1	↑2.6	↑3.1	↑2.2	↑3.4	↑2.9

ActionCLIP [33] 44.3 SlowFast [5] 45.2 En-VidTr-L [37] 47.3 MoViNet-A6 [9] 63.2 TubeViT-L [13] 66.2

Table 5. Evaluations of various methods (*top*) w/w/o flow dynamics correction and (*bottom*) comparisons to the state of the art on Charades.

Tab. 4 shows that our HAL achieves on par mAP performance compared to more complicated ODF+SDF when we activate the use of flow correction in its optical flow stream, and it performs better than DEEP-HAL by $\sim 3\%$. Flow dynamics correction boosts IDT-FV, I3D, DEEP-HAL and ODF+SDF for AR by $\sim 6\%$, 6% , 2% and 2% respectively on MPII. Note that MPII contains some fine-grained actions where different motion dynamics are of great importance to the recognition tasks, and our model with optical flow correction achieves the new state-of-the-art performance.

Tab. 5 shows our simple HAL achieves the best results on Charades. Our model is based on self-supervision, which learns to hallucinate the best motion dynamic features, makes our pipeline lightweight in comparison to competitors such as Contrastive Language-Image Pre-Training (CLIP) model, e.g., ActionCLIP, and video transformer-based model, e.g., En-VidTr-L. With AssembleNet++ backbone and our flow dynamics correction, we outperform AssembleNet++ by $\sim 3\%$.

4. CONCLUSIONS

In this paper, we address the challenge of selecting and enhancing motion dynamics through power normalizing optical flow speed components, achieving state-of-the-art results in action recognition benchmarks. Our approach allows tailored modeling of actions based on their significance (e.g., distinguishing subtle hand waves from robust walks or jogs). We show that leading action recognition methods benefit from our flow dynamics correction, and our low-computational-cost pipeline is advantageous for tasks like clustering and captioning in video processing.

5. REFERENCES

- [1] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *TPAMI*, 33(3):500–513, March 2011. 1, 2
- [2] J. Carreira and A. Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. *CVPR*, pages 1–10, 2018. 1, 2, 4
- [3] A. Cherian, B. Fernando, M. Harandi, and S. Gould. Generalized rank pooling for action recognition. In *CVPR*, 2017. 4
- [4] A. Cherian, S. Sra, S. Gould, and R. Hartley. Non-linear temporal subspace representations for activity recognition. In *CVPR*, pages 2197–2206, 2018. 4
- [5] C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, Seoul, Korea, 2019. IEEE. 4
- [6] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Temporal residual networks for dynamic scene recognition. In *CVPR*, 2017. 3, 4
- [7] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional Two-Stream Network Fusion for Video Action Recognition. In *CVPR*, pages 1–9, 2016. 1, 4
- [8] I. Hadji and R. P. Wildes. A new large scale dynamic texture dataset with application to ConvNet understanding. In *ECCV*, September 2018. 4
- [9] D. Kondratyuk, L. Yuan, Y. Li, L. Zhang, M. Brown, and B. Gong. Movinets: Mobile video networks for efficient video recognition. *arXiv*, 2021. 4
- [10] P. Koniusz, L. Wang, and A. Cherian. Tensor representations for action recognition. *TPAMI*, 2020. 1, 4
- [11] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV*, pages 2556–2563, 2011. 3
- [12] A. Piergiovanni, A. Angelova, A. Toshev, and M. S. Ryoo. Evolving space-time neural architectures for videos. In *ICCV*, 2019. 4
- [13] A. Piergiovanni, W. Kuo, and A. Angelova. Rethinking video vits: Sparse video tubes for joint image and video learning. In *CVPR*, pages 2214–2224, 2023. 4
- [14] Z. Qin, Y. Liu, P. Ji, D. Kim, L. Wang, R. McKay, S. Anwar, and T. Gedeon. Fusing higher-order features in graph neural networks for skeleton-based action recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 1
- [15] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, pages 1164–1172, 2015. 1, 2
- [16] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, 2012. 3
- [17] M. S. Ryoo, A. Piergiovanni, J. Kangasputa, and A. Angelova. Assemblenet++: Assembling modality representations via attention connections. In *ECCV*, pages 1–19, 2020. 1, 2, 4
- [18] M. S. Ryoo, A. Piergiovanni, M. Tan, and A. Angelova. Assemblenet: Searching for multi-stream neural connectivity in video architectures. In *ICLR*, pages 1–15, 2020. 1, 4
- [19] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *ECCV*, 2016. 3
- [20] H. Wang and C. Schmid. Action Recognition with Improved Trajectories. *ICCV*, pages 3551–3558, 2013. 1, 4
- [21] J. Wang and A. Cherian. Learning discriminative video representations using adversarial perturbations. In *ECCV*, pages 716–733, 2018. 4
- [22] L. Wang. Analysis and evaluation of Kinect-based action recognition algorithms. Master’s thesis, The University of Western Australia, Nov 2017. 1
- [23] L. Wang. *Robust Human Action Modelling*. PhD thesis, The Australian National University, Nov 2023. 1
- [24] L. Wang, B. Huang, Z. Zhao, Z. Tong, Y. He, Y. Wang, Y. Wang, and Y. Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *CVPR*, pages 14549–14560, June 2023. 4
- [25] L. Wang, D. Q. Huynh, and P. Koniusz. A comparative review of recent kinect-based action recognition algorithms. *TIP*, 2019. 1
- [26] L. Wang, D. Q. Huynh, and M. R. Mansour. Loss switching fusion with similarity search for video classification. *ICIP*, 2019. 1
- [27] L. Wang and P. Koniusz. Self-supervising action recognition by statistical moment and subspace descriptors. In *ACMMM*, 2021. 3, 4
- [28] L. Wang and P. Koniusz. Temporal-viewpoint transportation plan for skeletal few-shot action recognition. In *ACCV*, pages 4176–4193, 2022. 1
- [29] L. Wang and P. Koniusz. Uncertainty-dtw for time series and sequences. In *ECCV*, pages 176–195. Springer, 2022. 1
- [30] L. Wang and P. Koniusz. 3mformer: Multi-order multi-mode transformer for skeletal action recognition. In *CVPR*, pages 5620–5631, 2023. 1
- [31] L. Wang, P. Koniusz, and D. Q. Huynh. Hallucinating IDT descriptors and I3D optical flow features for action recognition with cnns. In *ICCV*, 2019. 3, 4
- [32] L. Wang, K. Sun, and P. Koniusz. High-order tensor pooling with attention for action recognition. *ICASSP*, 2024. 1
- [33] M. Wang, J. Xing, and Y. Liu. Actionclip: A new paradigm for video action recognition. *CoRR*, abs/2109.08472, 2021. 4
- [34] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. Deepflow: Large displacement optical flow with deep matching. In *ICCV*, pages 1385–1392, 2013. 1, 2
- [35] W. Wu, X. Wang, H. Luo, J. Wang, Y. Yang, and W. Ouyang. Bidirectional cross-modal knowledge exploration for video recognition with pre-trained vision-language models. In *CVPR*, 2023. 4
- [36] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *Pattern Recognition*, pages 214–223, 2007. 1, 2
- [37] Y. Zhang, X. Li, C. Liu, B. Shuai, Y. Zhu, B. Brattoli, H. Chen, I. Marsic, and J. Tighe. Vidtr: Video transformer without convolutions. In *ICCV*, pages 13577–13587, October 2021. 4