DMC-Net: Generating Discriminative Motion Cues for Fast Compressed Video Action Recognition

Zheng Shou[1,2] Xudong Lin[2] Yannis Kalantidis[1] Laura Sevilla-Lara[1,3]
Marcus Rohrbach[1] Shih-Fu Chang[2] Zhicheng Yan[1]

[1]Facebook AI [2]Columbia University [3]Univesity of Edinburgh

Abstract

*Motion has shown to be useful for video understanding, where motion is typically represented by optical flow. However, computing flow from video frames is very time-consuming. Recent works directly leverage the motion vectors and residuals readily available in the compressed video to represent motion at no cost. While this avoids flow computation, it also hurts accuracy since the motion vector is noisy and has substantially reduced resolution, which makes it a less discriminative motion representation. To remedy these issues, we propose a lightweight generator network, which reduces noises in motion vectors and captures fine motion details, achieving a more Discriminative Motion Cue (**DMC**) representation. Since optical flow is a more accurate motion representation, we train the DMC generator to approximate flow using a reconstruction loss and an adversarial loss, jointly with the downstream action classification task. Extensive evaluations on three action recognition benchmarks (HMDB-51, UCF-101, and a subset of Kinetics) confirm the effectiveness of our method. Our full system, consisting of the generator and the classifier, is coined as **DMC-Net** which obtains high accuracy close to that of using flow and runs two orders of magnitude faster than using optical flow at inference time.*

1. Introduction

Video is a rich source of visual content as it not only contains appearance information in individual frames, but also temporal motion information across consecutive frames. Previous work has shown that modeling motion is important to various video analysis tasks, such as action recognition [39, 47, 22], action localization [35, 34, 38, 5, 37, 24, 25] and video summarization [43, 28]. Currently, methods achieving state-of-the-art results usually follow the two-stream network framework [39, 4, 46], which consists of two Convolutional Neural Networks (CNNs), one for the decoded RGB images and one for optical flow, as shown in Figure 2a. These networks can operate on either single frames (2D inputs) or clips (3D inputs) and may utilize 3D spatiotemporal convolutions [44, 46].

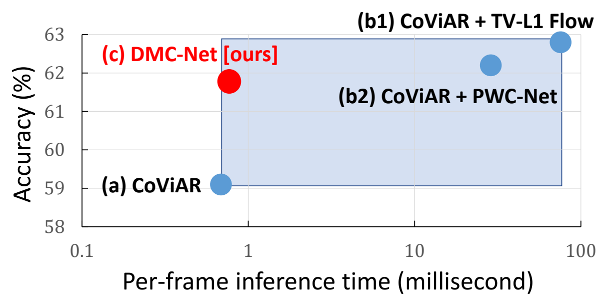
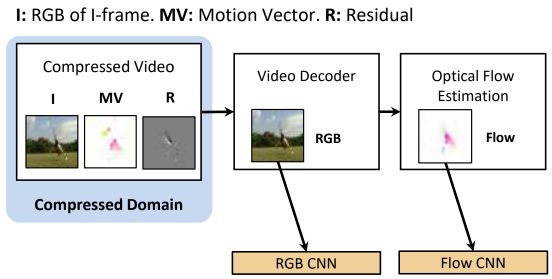


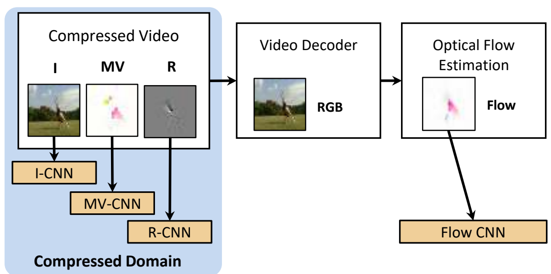
Figure 1: Comparing inference time and accuracy for different methods on HMDB-51. **(a)** Compressed video based method *CoViAR* [52] is very fast. **(b)** But in order to reach high accuracy, *CoViAR* has to follow two-stream networks to add the costly optical flow computation, either using TV-L1 [55] or PWC-Net [42]. **(c)** The proposed *DMC-Net* not only operates exclusively in the compressed domain, but also is able to achieve high accuracy while being two orders of magnitude faster than methods that use optical flow. The blue box denotes the improvement room from *CoViAR* to *CoViAR + TV-L1 Flow*; x-axis is in logarithmic scale.

Extracting optical flow, however, is very slow and often dominates the overall processing time of video analysis tasks. Recent work [52, 57, 56] avoids optical flow computation by exploiting the motion information from compressed videos encoded by standards like MPEG-4 [23]. Such methods utilize the motion vectors and residuals already present in the compressed video to model motion. The recently proposed CoViAR [52] method, for example, contains three independent CNNs operating over three modalities in the compressed video, *i.e.* RGB image of I-frame (**I**), low-resolution Motion Vector (**MV**) and Residual

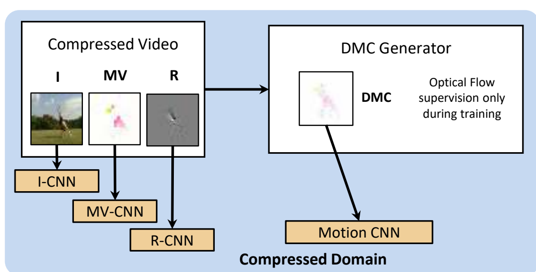
This work was partially done when Zheng Shou interned at Facebook.



(a) Two-stream network (TSN) [39]



(b) CoViAR [52] + Flow



(c) DMC-Net (ours)

Figure 2: Illustrations of (a) the two-stream network [39], (b) the recent CoViAR [52] method that achieves high accuracy via fusing compressed video data and optical flow, and (c) our proposed DMC-Net. Unlike *CoViAR+Flow* that requires video decoding of RGB images and flow estimation, our DMC-Net operates exclusively in the compressed domain at inference time while using optical flow to learn to capture discriminative motion cues at training time.

(**R**). The predictions from individual CNNs are combined by late fusion. CoViAR runs extremely fast while modeling motion features (see Figure 2b). However, in order to achieve state-of-the-art accuracy, late fusion with optical flow is further needed (see Figure 1).

This performance gap is due to the motion vector being less informative and discriminative than flow. First, the spatial resolution of the motion vector is substantially reduced (*i.e*. 16x) during video encoding, and fine motion details, which are important to discriminate actions, are permanently lost. Second, employing two CNNs to process motion vectors and residuals separately ignores the strong interaction between them. Because the residual is computed as the difference between the raw RGB image and its reference frame warped by the motion vector. The residual is often well-aligned with the boundary of moving object, which is more important than the motion at other locations for action recognition according to [32]. Jointly modeling motion vectors and residuals, which can be viewed as coarse-scale and fine-scale motion feature respectively, can exploit the encoded motion information more effectively.

To address those issues, we propose a novel approach to learn to generate a **Discriminative Motion Cue (DMC)** representation by refining the noisy and coarse motion vectors. We develop a lightweight DMC generator network that operates on stacked motion vectors and residuals. This generator requires training signals from different sources to capture discriminative motion cues and incorporate high-level recognition knowledge. In particular, since flow contains high resolution and accurate motion information, we encourage the generated DMC to resemble optical flow by using a pixel-level reconstruction loss. We also use an adversarial loss [13] to approximate the distribution of optical flow. Finally, the DMC generator is also supervised by the downstream action recognition classifier in an end-to-end manner, allowing it to learn motion cues that are discriminative for recognition.

During inference, the DMC generator is extremely efficient with merely *0.23* GFLOPs, and takes only *0.106 ms* per frame which is negligible compared with the time cost of using flow. In Figure 2c, we call our full model *DMC-Net*. Although optical flow is required during training, our method operates *exclusively in the compressed domain* at inference time and runs two orders of magnitude faster than methods using optical flow, as shown in Figure 1. Our contributions are summarized as follows:

- We propose DMC-Net, a novel and highly efficient framework that operates exclusively in the compressed video domain and is able to achieve high accuracy without requiring optical flow estimation.
- We design a lightweight generator network that can learn to predict discriminative motion cues by using optical flow as supervision and being trained jointly with action classifier. During inference, it runs two orders of magnitude faster than estimating flow.
- We extensively evaluate DMC-Net on 3 action recognition benchmarks, namely HMDB-51 [21], UCF-101 [40] and a subset of Kinetics [20], and demonstrate that it can significantly shorten the performance gap between state-of-the-art compressed video based methods with and without optical flow.

2. Related Work

Video Action Recognition. Advances in action recognition are largely driven by the success of 2D ConvNets in image recognition. The original Two-Stream Network [39] employs separate 2D ConvNets to process RGB frames and optical flow, and merges their predictions by late fusion. Distinct from image, video possesses temporal structure and motion information which are important for video analysis. This motivates researchers to model them more effectively, such as 3D ConvNets [44, 4], Temporal Segment Network (TSN) [49], dynamic image networks [1], and Non-Local Network [50]. Despite the enormous amount of effort on modeling motion via temporal convolution, 3D ConvNets can still achieve higher accuracy when fused with optical flow [4, 46], which is unfortunately expensive to compute.

Compressed Video Action Recognition. Recently, a number of approaches that utilize the information present in the compressed video domain have been proposed. In the pioneering works [56, 57], Zhang *et al*. replaced the optical flow stream in two-stream methods by a motion vector stream, but it still needed to decode RGB image for P-frame and ignored other motion-encoding modalities in compressed videos such as the residual maps. More recently, the CoViAR method [52] proposed to exploit all data modalities in compressed videos, *i.e*. RGB I-frames, motion vectors and residuals to bypass RGB frame decoding. However, CoViAR fails to achieve performance comparable to that of two-stream methods, mainly due to the low-resolution of the motion vectors and the fact that motion vectors and residuals, although highly related, are processed by independent networks. We argue that, when properly exploited, the compressed video modalities have enough signals to allow us to capture more discriminative motion representation. We therefore explicitly learn such representation as opposed to relying on optical flow during inference.

Motion Representation and Optical Flow Estimation. Traditional optical flow estimation methods explicitly model the displacement at each pixel between successive frames [15, 54, 7, 2]. In the last few years CNNs have successfully been trained to estimate the optical flow, including FlowNet [8, 17], SpyNet [31] and PWC-Net [42], and achieve low End-Point Error (EPE) on challenging benchmarks, such as MPI Sintel [3] and KITTI 2015 [29]. Im2Flow work [12] also shows optical flow can be hallucinated from still images. Recent work however, shows that accuracy of optical flow is not strongly correlated with accuracy of video recognition [33]. Thus, motion representation learning methods focus more on generating discriminative motion cues. Fan *et al*. [9] proposed to transform TV-L1 optical flow algorithm into a trainable sub-network, which can be jointly trained with downstream recognition network. Ng *et al*. [30] employs fully convolutional ResNet model to generate pixel-wise prediction of optical flow, and can be jointly trained with recognition network. Unlike optical flow estimation methods, our method does not aim to reduce EPE error. Also different from all above methods of motion representation learning which take decoded RGB frames as input, our method refines motion vectors in the compressed domain, and requires much less model capacity to generate discriminative motion cues.

3. Approach

In this section, we present our approach for generating *Discriminative Motion Cues (**DMC**)* from compressed video. The overall framework of our proposed **DMC-Net** is illustrated in Figure 3. In Section 3.1, we introduce the basics of compressed video and the notations we use. Then we design the DMC generator network in Section 3.2. Finally we present the training objectives in Section 3.3 and discuss inference in Section 3.4.

3.1. Basics and Notations of Compressed Video

We follow CoViAR [52] and use MPEG-4 Part2 [23] encoded videos where every I-frame is followed by 11 consecutive P-frames. Three data modalities are readily available in MPEG-4 compressed video: (1) RGB image of I-frame (**I**); (2) Motion Vector (**MV**) records the displacement of each macroblock in a P-frame to its reference frame and typically a frame is divided into 16x16 macroblocks during video compression; (3) Residual (**R**) stores the RGB difference between a P-frame and its reference I-frame after motion compensation based on MV. For a frame of height H and width W, I and R have shape (3, H, W) and MV has shape (2, H, W). But note that MV has much lower resolution in effect because its values within the same macroblock are identical.

3.2. The Discriminative Motion Cue Generator

Input of the generator. Existing compressed video based methods directly feed motion vectors into a classifier to model motion information. This strategy is not effective in modeling motion due to the characteristics of MV: (1) MV is computed based on simple block matching, making MV noisy and (2) MV has substantially lower resolution, making MV lack fine motion details. In order to specifically handle these characteristics of MV, we aim to design a lightweight generation network to reduce noise in MV and capture more fine motion details, outputting DMC as a more discriminative motion representation.

To accomplish this goal, MV alone may not be sufficient. According to [32], the motion nearby object boundary is more important than the motion at other locations for action recognition. We also notice R is often well-aligned with the boundary of moving objects. Moreover, R is strongly correlated with MV as it is computed as the difference between the original frame and its reference I-frame compensated

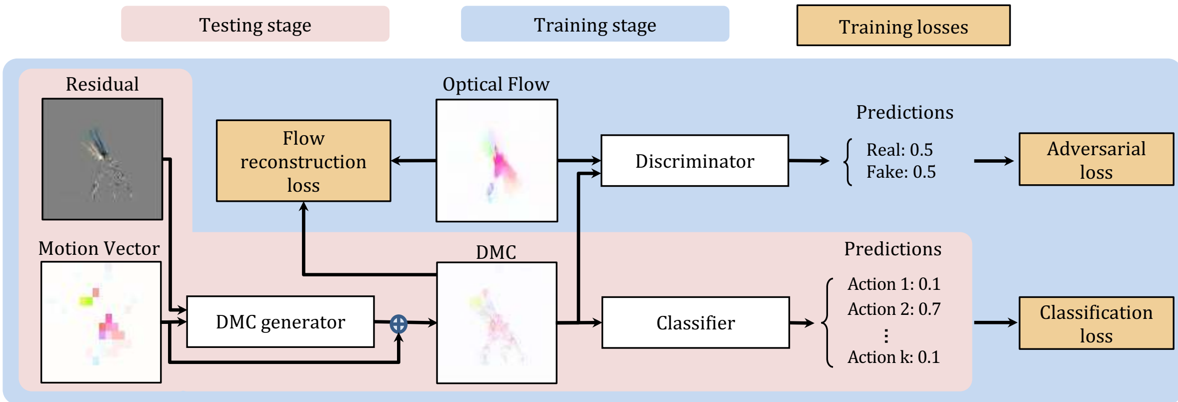


Figure 3: The framework of our Discriminative Motion Cue Network (DMC-Net). Given the stacked residual and motion vector as input, the DMC generator reduces noise in the motion vector and captures more fine motion details, outputting a more discriminative motion cue representation which is used by a small classification network to classify actions. In the training stage, we train the DMC generator and the action classifier jointly using three losses. In the test stage, only the modules highlighted in pink are used.

Network Architecture	GFLOPs
C3D [44]	38.5
Res3D-18 [45]	19.3
ResNet-152 [14]	11.3
ResNet-18 [14]	1.78
DMC generator (PWC-Net [42])	36.15
DMC generator [ours]	0.23

Table 1: Computational complexity of different networks. Input has height 224 and width 224.

Layer	Input size	Output size	Filter config
`conv0`	5, 224, 224	8, 224, 224	8, 3x3, 1, 1
`conv1`	13, 224, 224	8, 224, 224	8, 3x3, 1, 1
`conv2`	21, 224, 224	6, 224, 224	6, 3x3, 1, 1
`conv3`	27, 224, 224	4, 224, 224	4, 3x3, 1, 1
`conv4`	31, 224, 224	2, 224, 224	2, 3x3, 1, 1
`conv5`	33, 224, 224	2, 224, 224	2, 3x3, 1, 1

Table 2: The architecture of our Discriminative Motion Cue (DMC) generator network which takes stacked motion vector and residual as input. Input/output size follows the format of #channels, height, width. Filter configuration follows the format of #filters, kernel size, stride, padding.

using MV. Therefore, we propose to stack MV and R as input into the DMC generator, as shown in Figure 3. This allows utilizing the motion information in MV and R as well as the correlation between them, which cannot be modeled by separate CNNs as in the current compressed video works [52, 57, 56].

Generator network architecture. Quite a few deep generation networks have been proposed for optical flow estimation from RGB images. One of these works is PWC-Net [42], which achieves SoTA performance in terms of both End Point Error (EPE) and inference speed. We therefore choose to base our generator design principles on the ones used by PWC-Net. It is worth noting that PWC-Net takes decoded RGB frames as input unlike our proposed method operating only in the compressed domain.

Directly adopting the network architecture of the flow estimator network in PWC-Net for our DMC generator leads to high GFLOPs as indicated in Table 1. To achieve high efficiency, we have conducted detailed architecture search experimentally to reduce the number of filters in each convolutional layer of the flow estimator network in PWC-Net, achieving the balance between accuracy and complexity. Furthermore, since our goal is to refine MV, we propose to add a shortcut connection between the input MV and the output DMC, making the generator to directly predict the refinements which are added on MV to obtain DMC.

Table 2 shows the network architecture of our DMC generator: 6 convolutional layers are stacked sequentially with all convolutional layers densely connected [16]. Every convolutional filter has a 3x3 kernel with stride 1 and padding 1. Each convolutional layer except `conv5` is followed by a Leaky ReLU [26] layer, where the negative slope is 0.1.

As shown in Table 1, our DMC generator only requires 0.63% GFLOPs used by the flow estimator in PWC-Net if it were adopted to implement our DMC generator. Also, Table 1 compares our DMC generator with other popular network architectures for video analysis including frame-level models (ResNet-18 and ResNet-152 [14]) and clip-level models (C3D [44] and Res3D [45]). We observe that the complexity of DMC generator is orders of magnitude smaller compared to that of other architectures, which makes it running much faster. In the supplementary material, we explored a strategy of using two consecutive networks to respectively rectify errors in MV and capture fine motion details while this did not achieve better accuracy.

3.3. Flow-guided, Discriminative Motion Cues

Compared to MV, optical flow exhibits more discriminative motion information because: (1) Unlike MV is computed using simple block matching, nowadays dense flow estimation is computed progressively from coarse scales to fine scales [55]. (2) Unlike MV is blocky and thus misses fine details, flow keeps the full resolution of the corresponding frame. Therefore we propose to guide the training of our DMC generator using optical flow. To this end, we have explored different ways and identify three effective training losses as shown in Figure 3 to be presented in the following: a flow reconstruction loss, an adversarial loss, and a downstream classification loss.

3.3.1 Optical Flow Reconstruction Loss

First, we minimize the per-pixel difference between the generated DMC and its corresponding optical flow. Following Im2Flow [12] which approximates flow from a single RGB image, we use the Mean Square Error (MSE) reconstruction loss $\mathcal{L}_{\text{mse}}$ defined as:

$$\mathcal{L}_{\text{mse}} = \mathbb{E}_{\mathbf{x} \sim p} \left\| \mathcal{G}_{\text{DMC}}(\mathbf{x}) - \mathcal{G}_{\text{OF}}(\mathbf{x}) \right\|_2^2, \quad (1)$$

where p denotes the set of P-frames in the training videos, $\mathbb{E}$ stands for computing expectation, $\mathcal{G}_{\text{DMC}}(\mathbf{x})$ and $\mathcal{G}_{\text{OF}}(\mathbf{x})$ respectively denote the DMC and optical flow for the corresponding input frame $\mathbf{x}$ sampled from p. Since only some regions of flow contain discriminative motion cues that are important for action recognition, in the supplementary material we have explored weighting the flow reconstruction loss to encourage attending to the salient regions of flow. But this strategy does not achieve better accuracy.

3.3.2 Adversarial Loss

As pointed out by previous works [27], the MSE loss implicitly assumes that the target data is drawn from a Gaussian distribution and therefore tends to generate smooth and blurry outputs. This in effect results in less sharp motion representations especially around boundaries, making the generated DMC less discriminative. Generative Adversarial Networks (GAN) [13] has been proposed to minimize the Jensen$-$Shannon divergence between the generative model and the true data distribution, making these two similar. Thus in order to help our DMC generator learn to approximate the distribution of optical flow data, we further introduce an adversarial loss. Note that unlike GAN which samples from random noise, adversarial loss samples from the input dataset, which already has large variability [27].

We relax the notational rigor and use $\mathcal{G}_{OF}(\mathbf{x})$ to refer to the optical flow corresponding to the frame $\mathbf{x}$, although for many optical flow algorithms the input would be a pair of frames.

Let our DMC generator $\mathcal{G}_{\text{DMC}}$ be the **Generator** in the adversarial learning process. As shown in Figure 3, a **Discriminator** $\mathcal{D}$ is introduced to compete with $\mathcal{G}_{\text{DMC}}$. $\mathcal{D}$ is instantiated by a binary classification network that takes as input either **real** optical flow or **fake** samples generated via our DMC generator. Then $\mathcal{D}$ outputs a two-dimensional vector that is passed through a softmax operation to obtain the probability $P_{\mathcal{D}}$ of the input being *Real*, *i.e*. flow versus *Fake*, *i.e*. DMC. $\mathcal{G}_{\text{DMC}}$ and $\mathcal{D}$ are trained in an alternating manner: $\mathcal{G}_{\text{DMC}}$ is fixed when $\mathcal{D}$ is being optimized, and vice versa.

During training $\mathcal{D}$, $\mathcal{G}_{\text{DMC}}$ is fixed and is only used for inference. $\mathcal{D}$ aims to classify the generated DMC as Fake and classify flow as Real. Thus the adversarial loss for training $\mathcal{D}$ is:

$$\begin{aligned} \mathcal{L}_{\text{adv}}^{D} = &\mathbb{E}_{\mathbf{x} \sim p}[-\log P_{\mathcal{D}}(\text{Fake}|\mathcal{G}_{\text{DMC}}(\mathbf{x})) \\ &- \log P_{\mathcal{D}}(\text{Real}|\mathcal{G}_{\text{OF}}(\mathbf{x}))], \end{aligned} \quad (2)$$

where p denotes the set of P-frames in the training set and $\mathcal{G}_{\text{DMC}}(\mathbf{x})$ and $\mathcal{G}_{\text{OF}}(\mathbf{x})$ respectively represent the DMC and optical flow for each input P-frame $\mathbf{x}$.

During training $\mathcal{G}_{\text{DMC}}$, $\mathcal{D}$ is fixed. $\mathcal{G}_{\text{DMC}}$ is encouraged to generate DMC that is similar and indistinguishable with flow. Thus the adversarial loss for training $\mathcal{G}_{\text{DMC}}$ is:

$$\mathcal{L}_{\text{adv}}^{G} = \mathbb{E}_{\mathbf{x} \sim p}[-\log P_{\mathcal{D}}(\text{Real}|\mathcal{G}_{\text{DMC}}(\mathbf{x}))], \quad (3)$$

which can be trained jointly with the other losses designed for training the DMC generator in an end-to-end fashion, as presented in Section 3.3.3.

Through the adversarial training process, $\mathcal{G}_{\text{DMC}}$ learns to approximate the distribution of flow data, generating DMC with more fine details and thus being more similar to flow. Those fine details usually capture discriminative motion cues and are thus important for action recognition. We present details of the discriminator network architecture in the supplementary material.

3.3.3 The Full Training Objective Function

Semantic classification loss. As our final goal is to create motion representation that is discriminative with respect to the downstream action recognition task, it is important to train the generator jointly with the follow-up action classifier. We employ the softmax loss as our action classification loss, denoted as $\mathcal{L}_{\text{cls}}$.

The full training objective. Our whole model is trained with the aforementioned losses putting together in an end-to-end manner. The training process follows the alternating training procedure stated in Section 3.3.2. During training the discriminator, $\mathcal{D}$ is trained while the DMC generator $\mathcal{G}_{\text{DMC}}$ and the downstream action classifier are fixed. The full training objective is to minimize the adversarial loss

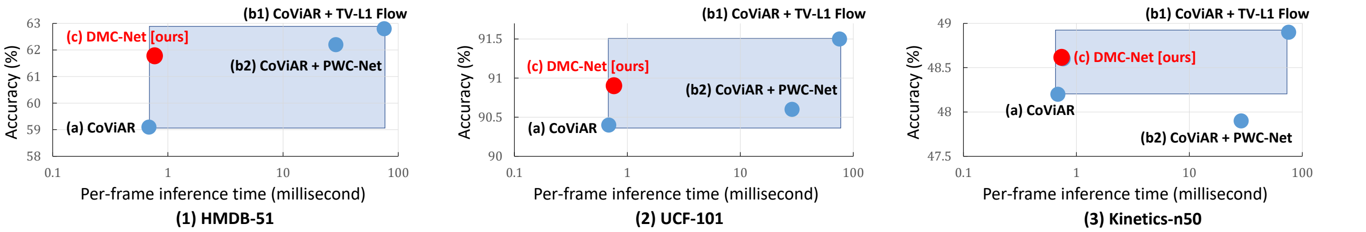


Figure 4: Accuracy vs. speed on 3 benchmarks. Results on UCF-101 and HMDB-51 are averaged over 3 splits. (b1) and (b2) use ResNet-18 to classify flow and (c) also uses ResNet-18 to classify DMC. The proposed *DMC-Net* not only operates exclusively in the compressed domain, but also is able to achieve higher accuracy than (a) while being two orders of magnitude faster than methods that use optical flow. The blue area indicates the improvement room from (a) to (b1).

$\mathcal{L}^{D}_{\text{adv}}$ in Equation 2. During training the generator $\mathcal{G}_{\text{DMC}}$, $\mathcal{D}$ is fixed while the DMC generator $\mathcal{G}_{\text{DMC}}$ and the downstream action classifier are trained jointly with the following full training objective to be minimized:

$$\mathcal{L}_{\text{cls}} + \alpha \cdot \mathcal{L}_{\text{mse}} + \lambda \cdot \mathcal{L}^{G}_{\text{adv}}, \tag{4}$$

where $\mathcal{L}_{\text{mse}}$ is given by Equation 1, $\mathcal{L}^{G}_{\text{adv}}$ is given by Equation 3, and α, λ are balancing weights.

3.4. Inference

As shown in Figure 3, despite having three losses jointly trained end-to-end, our DMC-Net is actually quite efficient during inference: basically first the generator outputs DMC and then the generated DMC is fed into the classification network to make action class prediction. We compare our inference speed with other methods in Section 4.4.

4. Experiments

In this section, we first detail our experimental setup, present quantitative analysis of our model, and finally compare with state-of-the-art methods.

4.1. Datasets and Evaluation

UCF-101 [41]. This dataset contains 13,320 videos from 101 action categories, along with 3 public train/test splits.
HMDB-51 [21]. This dataset contains 6,766 videos from 51 action categories, along with 3 public train/test splits.
Kinetics-n50. From the original Kinetics-400 dataset [4], we construct a subset referred as **Kinetics-n50** in this paper. We keep all 400 categories. For each class, we randomly sample 30 videos from the original training set as our training videos and randomly sample 20 videos from the original validation set as our testing videos. We evaluate on the full set in the supplementary material.
Evaluation protocol. All videos in the above datasets have single action label out of multiple classes. Thus we evaluate top-1 video-level class prediction accuracy.

4.2. Implementation Details

Training. For I, MV, and R, we follow the exactly same setting as used in CoViAR [52]. Note that I employs ResNet-152 classifier; MV and R use ResNet-18 classifier. To ensure efficiency, DMC-Net also uses ResNet-18 to classify DMC in the whole paper unless we explicitly point out. To allow apple-to-apple comparisons between DMC and flow, we also choose frame-level ResNet-18 classifier as the flow CNN shown in Figure 2b. TV-L1 [54] is used for extracting optical flow to guide the training of our DMC-Net. All videos are resized to 340×256. Random cropping of 224×224 and random flipping are used for data augmentation. More details are in the supplementary material.
Testing. For I, MV, and R, we follow the exactly same setting as in CoViAR [52]: 25 frames are uniformly sampled for each video; each sampled frame has 5 crops augmented with flipping; all 250 (25×2×5) score predictions are averaged to obtain one video-level prediction. For DMC, we following the same setting except that we do not use cropping and flipping, which shows comparable accuracy but requires less computations. Finally, we follow CoViAR [52] to obtain the final prediction via fusing prediction scores from all modalities (*i.e.* I, MV, R, and DMC).

4.3. Model Analysis

How much gain DMC-Net can improve over CoViAR? Figure 4 reports accuracy on all three datasets. **CoViAR + TV-L1** and **CoViAR + PWC-Net** follow two-stream methods to include an optical flow stream computed by TV-L1 [55] and PWC-Net [42] respectively. **CoViAR + TV-L1** can be regard as our upper bound for improving accuracy because TV-L1 flow is used to guide the training of **DMC-Net**. By only introducing a lightweight DMC generator, our **DMC-Net** significantly improves the accuracy of **CoViAR** to approach **CoViAR + Flow**. Figure 5 shows that the generated DMC has less noisy signals such as those in the background area and DMC captures fine and sharp details of motion boundary, leading to the accuracy gain over **CoViAR**.
How effectiveness is each proposed loss? On HMDB-51, when only using the classification loss, the accuracy of DMC-Net is 60.5%; when using the classification loss and the flow reconstruction loss, the accuracy is improved to 61.5%; when further including the adversarial training loss, DMC-Net eventually achieves 61.8% accuracy. As in-

		Two-Stream Method (RGB+Flow)		Compressed Video Based Methods	
		BN-Inception	ResNet152	CoViAR	DMC-Net [ours]
Time (ms)	Preprocess	75.0	75.0	0.46	0.46
	CNN (**S**)	1.6	7.5	0.59	0.89
	Total (**S**)	76.6	82.5	1.05	1.35
	CNN (**C**)	0.9	4.0	0.22	0.30
	Total (**C**)	75.9	79.0	0.68	0.76
FPS	CNN (**C**)	1111.1	250.0	4545.4	3333.3
	Total (**C**)	13.1	12.6	1470.5	1315.7

(a) DMC-Net vs. Two-stream methods and CoViAR

	Generator	Generator + Cls.
	Time (ms) / FPS	Time (ms) / FPS
Deepflow [51]	1449.2 / 0.7	1449.5 / 0.7
Flownet2.0 [17]	220.8 / 4.5	221.0 / 4.5
TVNet [9]	83.3 / 12.0	83.5 / 12.0
PWC-Net [42]	28.6 / 35.0	28.8 / 34.8
DMC-Net [ours]	**0.1 / 9433.9**	**0.3 / 3333.3**

(b) DMC-Net vs. flow estimation methods

Table 3: Comparisons of per-frame inference speed. **(a)** Comparing our DMC-Net to the two-stream methods [18, 14] and the CoViAR method [52]. We consider two scenarios of forwarding multiple CNNs sequentially and concurrently, denoted by **S** and **C** respectively. We measure CoViAR's CNN forwarding time using our own implementation as mentioned in Section 4.4 and numbers are comparable to those reported in [52]. **(b)** Comparing our DMC-Net to deep network based optical flow estimation and motion representation learning methods, whose numbers are quoted from [9]. CNNs in DMC-Net are forwarded concurrently. All networks have batch size set to 1. For the classifier (denoted as Cls.), all methods use ResNet-18.

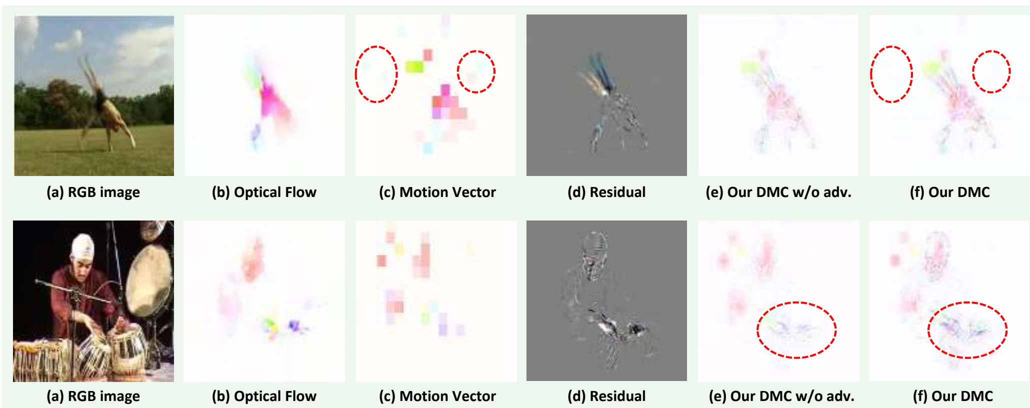


Figure 5: A Cartwheel example (top) and a PlayingTabla (bottom) example. All images in one row correspond to the same frame. For the Cartwheel example, these noisy blocks in the background (highlighted by two red circles) are reduced in our DMC. For the PlayingTabla example, our DMC exhibits sharper and more discriminative motion cues around hands (highlighted by the red circle) than our DMC w/o the adversarial loss during training. Better viewed in color.

dicated by previous literature [19], using an adversarial loss without a reconstruction loss often introduces artifacts.

4.4. Inference Speed

Following [52], we measure the average per-frame running time, which consists of the time for data pre-processing and the time for CNN forward pass. For the CNN forward pass, both the scenarios of forwarding multiple CNNs sequentially and concurrently are considered. Detailed results can be found in Table 3 (a). Results of **two-stream methods** are quoted from [52]. Due to the need of decoding compressed video into RGB frames and then computing optical flow, its pre-process takes much longer time than compressed video based methods. **DMC-Net** accepts the same inputs as **CoViAR** and thus **CoViAR** and **DMC-Net** have the same pre-processing time. As for the CNN forwarding time of compressed video based methods, we measure **CoViAR** and **DMC-Net** using the exactly same implementation as stated in Section 4.2 and the same experimental setup: we use one NVIDIA GeForce GTX 1080 Ti and set the batch size of each CNN to 1 while in practice the speed can be further improved to utilize larger batch size. Despite adding little computational overhead on CoViAR, DMC-Net is still significantly faster than the conventional **two-stream methods**.

Deepflow [51], **Flownet** [17] and **PWC-Net** [42] have been proposed to accelerate optical flow estimation by using deep networks. **TVNet** [9] was proposed to generate even better motion representation than flow with fast speed. Those estimated flow or generated motion representation can replace optical flow used in two-stream methods to go through a CNN for classification. We combine these meth-

	HMDB-51	UCF-101
Compressed video based methods		
EMV-CNN [56]	51.2 (split1)	86.4
DTMV-CNN [57]	55.3	87.5
CoViAR [52]	59.1	90.4
DMC-Net (ResNet-18) [ours]	62.8	90.9
DMC-Net (I3D) [ours]	71.8	92.3
Decoded video based methods *(RGB only)*		
Frame-level classification		
ResNet-50 [14]	48.9	82.3
ResNet-152 [14]	46.7	83.4
Motion representation learning		
ActionFlowNet (2-frames) [30]	42.6	71.0
ActionFlowNet [30]	56.4	83.9
PWC-Net (ResNet-18) + CoViAR [42]	62.2	90.6
TVNet [9]	71.0	94.5
Spatio-temporal modeling		
C3D [44]	51.6	82.3
Res3D [45]	54.9	85.8
ARTNet [48]	70.9	94.3
MF-Net [6]	74.6	96.0
S3D [53]	75.9	96.8
I3D RGB [4]	74.8	95.6
I3D RGB + DMC-Net (I3D) [ours]	77.8	96.5
Decoded video based methods *(RGB + Flow)*		
Two-stream [39]	59.4	88.0
Two-Stream fusion [11]	65.4	92.5
I3D [4]	80.7	98.0
R(2+1)D [46]	78.7	97.3

Table 4: Accuracy averaged over all three splits on HMDB-51 and UCF-101 for both state-of-the-art compressed video based methods and decoded video based methods.

ods with a ResNet-18 classifier in Table 3 (b). We can see that our DMC generator runs much faster than these state-of-the-art motion representation learning methods.

4.5. Comparisons with Compressed Video Methods

As shown in the top section of Table 4, **DMC-Net** outperforms all other methods that operate in the compressed video domain, *i.e.* **CoViAR** [52], **EMV-CNN** [56] and **DTMV-CNN** [57]. Our method outperforms methods like [56, 57] that the output of the MV classifier is trained to approximate the output of the optical flow classifier. We believe this is because of the fact that approximating the classification output directly is not ideal, as it does not explicitly address the issues that MV is noisy and low-resolutional. By generating a more discriminative motion representation DMC, we are able to get features that are highly discriminative for the downstream recognition task. Furthermore, our DMC-Net can be combined with these classification networks of high capacity and trained in an end-to-end manner. **DMC-Net (I3D)** replaces the classifier from ResNet-18 to I3D, achieving significantly higher accuracy and outperforming a number of methods that require video decoding. Our supplementary material discusses the speed of I3D.

4.6. Comparisons with Decoded Video Methods

In this section we compare DMC-Net to approaches that require decoding all RGB images from compressed video. Some only use the RGB images, while others adopt the two-stream method [39] and further require computing flow.

RGB only. As shown in Table 4, decoded video methods only based on RGB images can be further divided into three categories. **(1) Frame-level classification**: 2D CNNs like ResNet-50 and ResNet-152 [14] have been experimented in [10] to classify each frame individually and then employ simple averaging to obtain the video-level prediction. Due to lacking motion information, frame-level classification underperforms **DMC-Net**. **(2) Motion representation learning**: In Table 4, we evaluate **PWC-Net (ResNet-18) + CoViAR** which feeds estimated optical flow into a ResNet-18 classifier and then fuses the prediction with **CoViAR**. The accuracy of **PWC-Net (ResNet-18) + CoViAR** is not as good as DMC-Net because our generated DMC contains more discriminative motion cues that are complementary to MV. For TVNet [9], the authors used BN-Inception [18] to classify the generated motion representation and then fuse the prediction with a RGB CNN. The accuracy of TVNet is better **DMC-Net (ResNet-18)** thanks to using a strong classifier but is worse than our **DMC-Net (I3D)**. **(3) Spatio-temporal modeling**: There are also a lot of works using CNN to model the spatio-temporal patterns across multiple RGB frames to implicitly capture motion patterns. It turns out that our **DMC-Net** discovers motion cues that are complementary to such spatio-temporal patterns: **I3D RGB + DMC-Net (I3D)** improves **I3D RGB** via incorporating predictions from our **DMC-Net (I3D)**.

RGB + Flow. As shown in Table 4, the state-of-the-art accuracy is belonging to the two-stream methods [20, 46], which combine predictions made from a RGB CNN and an optical flow CNN. But as discussed in Section 4.4, extracting optical flow is quite time-consuming and thus these two-stream methods are much slower than our **DMC-Net**.

5. Conclusion

In this paper, we introduce **DMC-Net**, a highly efficient deep model for video action recognition in the compressed video domain. Evaluations on 3 action recognition benchmarks lead to substantial gains in accuracy over prior work, without the assistance of computationally expensive flow. The supplementary materials can be found in [36].

6. Acknowledgment

Zheng Shou would like to thank the support from Wei Family Private Foundation when Zheng was at Columbia.

References

[1] Hakan Bilen, Basura Fernando, Efstratios Gavves, and Andrea Vedaldi. Action recognition with dynamic image networks. *IEEE transactions on pattern analysis and machine intelligence*, 2018. 3

[2] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International journal of computer vision*, 61(3):211–231, 2005. 3

[3] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 3

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 1, 3, 6, 8

[5] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1130–1139, 2018. 1

[6] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-fiber networks for video recognition. In *ECCV*, 2018. 8

[7] J Lewis M Black D Sun, S Roth. Learning optical flow. In *ECCV*, 2008. 3

[8] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766, 2015. 3

[9] Lijie Fan, Wenbing Huang, Stefano Ermon Chuang Gan, Boqing Gong, and Junzhou Huang. End-to-end learning of motion representation for video understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6016–6025, 2018. 3, 7, 8

[10] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8

[11] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 8

[12] Ruohan Gao, Bo Xiong, and Kristen Grauman. Im2flow: Motion hallucination from static images for action recognition. In *CVPR*, 2018. 3, 5

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 2, 5

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4, 7, 8

[15] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. 3

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 4

[17] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE conference on computer vision and pattern recognition (CVPR)*, volume 2, page 6, 2017. 3, 7

[18] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 7, 8

[19] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017. 7

[20] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2, 8

[21] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2556–2563. IEEE, 2011. 2, 6

[22] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 1

[23] Didier Le Gall. Mpeg: A video compression standard for multimedia applications. *Communications of the ACM*, 1991. 1, 3

[24] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 988–996. ACM, 2017. 1

[25] Tianwei Lin, Xu Zhao, and Zheng Shou. Temporal convolution based action proposal: Submission to activitynet 2017. *arXiv preprint arXiv:1707.06750*, 2017. 1

[26] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3, 2013. 4

[27] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *ICLR*, 2016. 5

[28] Engin Mendi, Hélio B Clemente, and Coskun Bayrak. Sports video summarization based on motion analysis. *Computers & Electrical Engineering*, 39(3):790–796, 2013. 1

[29] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015. 3

[30] Joe Yue-Hei Ng, Jonghyun Choi, Jan Neumann, and Larry S Davis. Actionflownet: Learning motion representation for action recognition. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1616–1624. IEEE, 2018. 3, 8

[31] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, page 2. IEEE, 2017. 3

[32] Laura Sevilla-Lara, Yiyi Liao, Fatma Guney, Varun Jampani, Andreas Geiger, and Michael J Black. On the integration of optical flow and action recognition. *arXiv preprint arXiv:1712.08416*, 2017. 2, 3

[33] Laura Sevilla-Lara, Yiyi Liao, Fatma Guney, Varun Jampani, Andreas Geiger, and Michael J. Black. On the integration of optical flow and action recognition. In *German Conference on Pattern Recognition (GCPR)*, Oct. 2018. 3

[34] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1417–1426. IEEE, 2017. 1

[35] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weaklysupervised temporal action localization in untrimmed videos. In *ECCV*, 2018. 1

[36] Zheng Shou, Xudong Lin, Yannis Kalantidis, Laura Sevilla-Lara, Marcus Rohrbach, Shih-Fu Chang, and Zhicheng Yan. Dmc-net: Generating discriminative motion cues for fast compressed video action recognition. *arXiv preprint arXiv:1901.03460*, 2019. 8

[37] Zheng Shou, Junting Pan, Jonathan Chan, Kazuyuki Miyazawa, Hassan Mansour, Anthony Vetro, Xavier Giro-i Nieto, and Shih-Fu Chang. Online action detection in untrimmed, streaming videos-modeling and evaluation. In *ECCV*, 2018. 1

[38] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016. 1

[39] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 2014. 1, 2, 3, 8

[40] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 2

[41] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. In *CRCV-TR-12-01*, 2012. 6

[42] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3, 4, 6, 7, 8

[43] Antonio Tejero-de Pablos, Yuta Nakashima, Tomokazu Sato, Naokazu Yokoya, Marko Linna, and Esa Rahtu. Summarization of user-generated sports video by using deep action recognition features. *IEEE Transactions on Multimedia*, 20(8):2000–2011, 2018. 1

[44] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1, 3, 4, 8

[45] Du Tran, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri. Convnet architecture search for spatiotemporal feature learning. *arXiv preprint arXiv:1708.05038*, 2017. 4, 8

[46] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 1, 3, 8

[47] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013. 1

[48] Limin Wang, Wei Li, Wen Li, and Luc Van Gool. Appearance-and-relation networks for video classification. In *CVPR*, 2018. 8

[49] 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. In *ECCV*, 2016. 3

[50] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 3

[51] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013. 7

[52] Chao-Yuan Wu, Manzil Zaheer, Hexiang Hu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Compressed video action recognition. In *CVPR*, 2018. 1, 2, 3, 4, 6, 7, 8

[53] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. In *ECCV*, 2018. 8

[54] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint Pattern Recognition Symposium*, pages 214–223. Springer, 2007. 3, 6

[55] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l1 optical flow. In *Joint Pattern Recognition Symposium*, 2007. 1, 5, 6

[56] Bowen Zhang, Limin Wang, Zhe Wang, Yu Qiao, and Hanli Wang. Real-time action recognition with enhanced motion vector cnns. In *CVPR*, 2016. 1, 3, 4, 8

[57] Bowen Zhang, Limin Wang, Zhe Wang, Yu Qiao, and Hanli Wang. Real-time action recognition with deeply transferred motion vector cnns. *IEEE Transactions on Image Processing*, 2018. 1, 3, 4, 8