Graph Distillation for Action Detection with Privileged Modalities

Zelun Luo[1,2⋆][0000−0003−3597−5046], Jun-Ting Hsieh[1],
Lu Jiang[2], Juan Carlos Niebles[1,2], and Li Fei-Fei[1,2]

[1] Stanford University [2] Google Inc.

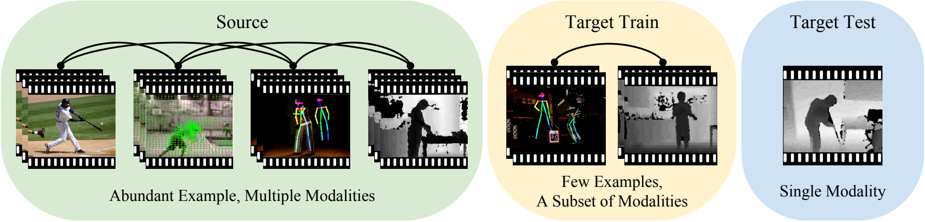

Fig. 1. Our problem statement. In the source domain, we have abundant data from multiple modalities. In the target domain, we have limited data and a subset of the modalities during training, and only one modality during testing. The curved connectors between modalities represent our proposed graph distillation.

Abstract. We propose a technique that tackles action detection in multimodal videos under a realistic and challenging condition in which only limited training data and partially observed modalities are available. Common methods in transfer learning do not take advantage of the extra modalities potentially available in the source domain. On the other hand, previous work on multimodal learning only focuses on a single domain or task and does not handle the modality discrepancy between training and testing. In this work, we propose a method termed graph distillation that incorporates rich privileged information from a large-scale multimodal dataset in the source domain, and improves the learning in the target domain where training data and modalities are scarce. We evaluate our approach on action classification and detection tasks in multimodal videos, and show that our model outperforms the state-of-the-art by a large margin on the NTU RGB+D and PKU-MMD benchmarks. The code is released at http://alan.vision/eccv18_graph/.

1 Introduction

Recent advancements in deep convolutional neural networks (CNN) have been successful in various vision tasks such as image recognition [7,17,23] and object

[⋆] Work done during an internship at Google Cloud AI.

detection [13,44,45]. A notable bottleneck for deep learning, when applied to multimodal videos, is the lack of massive, clean, and task-specific annotations, as collecting annotations for videos is much more time-consuming and expensive. Furthermore, restrictions such as privacy or runtime may limit the access to only a subset of the video modalities during test time.

The scarcity of training data and modalities is encountered in many real-world applications including self-driving cars, surveillance, and health care. A representative example is activity understanding on health care data that contain Personally Identifiable Information (PII) [16,34]. On the one hand, the number of labeled videos is usually limited because either important events such as falls [40,64] are extremely rare or the annotation process requires a high level of medical expertise. On the other hand, RGB violates individual privacy and optical flow requires non-real-time computations, both of which are known to be important for activity understanding but are often unavailable at test time. Therefore, detection can only be performed on real-time and privacy-preserving modalities such as depth or thermal videos.

Inspired by these problems, we study action detection in the setting of limited training data and partially observed modalities. To do so, we make use of a large action classification dataset that contains various *heterogeneous* modalities as the source domain to assist the training of the action detection model in the target domain, as illustrated in Fig. 1. Following the standard assumption in transfer learning [60], we assume that the source and target domain are similar to each other. We define a modality as a privileged modality if (1) it is available in the source domain but not in the target domain; (2) it is available during training but not during testing.

We identify two technical challenges in this problem. First of all, due to modality discrepancy in types and quantities, traditional domain adaption or transfer learning methods [12,41] cannot be directly applied. Recent work on knowledge and cross-modal distillation [18,26,33,49] provides a promising way of transferring knowledge between two models. Given two models, we can specify the distillation as the direction from the strong model to the weak model. With some adaptations, these methods can be used to distill knowledge between modalities. However, these adapted methods fail to address the second challenge: how to leverage the privileged modalities effectively. More specifically, given multiple privileged modalities, the distillation directions and weights are difficult to be pre-specified. Instead, the model should learn to dynamically adjust the distillation based on different actions or examples. For instance, some actions are easier to detect by optical flow whereas others are easier by skeleton features, and therefore the model should adjust its training accordingly. However, this dynamic distillation paradigm has not yet been explored by existing methods.

To this end, we propose the novel *graph distillation* method to learn a dynamic distillation across multiple modalities for action detection in multimodal videos. The graph distillation is designed as a layer attachable to the original model and is end-to-end learnable with the rest of the network. The graph can dynamically learn the example-specific distillation to better utilize the complementary information in multimodal data. As illustrated in Fig. 1, by effectively leveraging the privileged modalities from both the source domain and the training stage of the target domain, graph distillation significantly improves the test-time performance on a single modality. Note that graph distillation can be applied to both single-domain (from training to testing) and cross-domain (from one task to another) tasks. For our cross-domain experiment (from action classification to detection), we utilized the most basic transfer learning approach, *i.e.* pre-train and fine-tune, as this is orthogonal to our contributions. We can potentially achieve even better results with advanced transfer learning and domain adaptation techniques and we leave it for future study.

We validate our method on two public multimodal video benchmarks: PKU-MMD [28] and NTU RGB+D [46]. The datasets represent one of the largest public multimodal video benchmarks for action detection and classification. The experimental results show that our method outperforms the state-of-the-art approaches. Notably, it improves the state-of-the-art by 9.0% on PKU-MMD [28] (at 0.5 tIoU threshold) and by 6.6% on NTU RGB+D [46]. The remarkable improvement on the two benchmarks is a convincing validation of our method.

To summarize, our contribution is threefold. (1) We study a realistic and challenging condition for multimodal action detection with limited training data and modalities. To the best of our knowledge, we are first to effectively transfer multimodal privileged information across domains for action detection and classification. (2) We propose the novel graph distillation layer that can dynamically learn to distill knowledge across multiple privileged modalities and can be attached to existing models and learned in an end-to-end manner. (3) Our method outperforms the state-of-the-art by a large margin on two popular benchmarks, including action classification task on the challenging NTU RGB+D [46] and action detection task on PKU-MMD [28].

2 Related Work

Multimodal Action Classification and Detection. The field of action classification [3,50,52] and action detection [2,11,14,65] in RGB videos has been studied by the computer vision community for decades. The success in RGB videos has given rise to a series of studies on action recognition in multimodal videos [10,20,22,25,51,55]. Specifically, with the availability of depth sensors and joint tracking algorithms, extensive research has been done on action classification and detection in RGB-D videos [39,47,48,61] as well as skeleton sequences [24,30,31,32,46,63]. Different from previous work, our model focuses on leveraging privileged modalities on a source dataset with abundant training examples. We show that it benefits action detection when the target training dataset is small in size, and when only one modality is available at test time.

Video Understanding Under Limited Data. Our work is largely motivated by real-world situations where data and modalities are limited. For example, surveillance systems for fall detection [40,64] often face the challenge that annotated videos of fall incidents are hard to obtain, and more importantly, yhr recording of RGB videos is prohibited due to privacy concerns. Existing approaches to tackling this challenge include using transfer learning [36,42] and leveraging noisy data from web queries [5,27,59]. Specifically to our problem, it is common to transfer models trained on action classification to action detection.

The transfer learning methods are proved to be effective. However, it requires the source and target domains to have the same modalities. In reality, the source domain often contains richer modalities. For instance, suppose the depth video is the only available modality in the target domain, it remains nontrivial to transfer the other modalities (*e.g.* RGB, optical flow) even though they are readily available in the source domain and could make the model more accurate. Our method provides a practical approach to leveraging the rich multimodal information in the source domain, benefiting the target domain of limited modalities.

Learning Using Privileged Information. Vapnik and Vashist [53] introduced a *Student-Teacher* analogy: in real-world human learning, the role of a teacher is crucial to the student's learning process since the teacher can provide explanations, comments, comparisons, metaphors, etc. They proposed a new learning paradigm called Learning Using Privileged Information (LUPI), where at training time, additional information about the training example is provided to the learning model. At test time, the privileged information is not available, and the student operates without the supervision of the teacher [53].

Several work employed privileged information (PI) on SVM classifiers [53,56]. Ding et al. [8] handled missing modality transfer learning using latent low-rank constraint. Recently, the use of privileged information has been combined with deep learning in various settings such as PI reconstruction [49,57], information bottleneck [38], and Multi-Instance Multi-Label (MIML) learning [58]. The idea more related to our work is the combination of distillation and privileged information, which will be discussed next.

Knowledge Distillation. Hinton et al. [18] introduced the idea of knowledge distillation, where knowledge from a large model is distilled to a small model, improving the performance of the small model at test time. This is done by adding a loss function that matches the outputs of the small network to the high-temperature soft outputs of the large network [18]. Lopez-Paz et al. [33] later proposed a generalized distillation that combined distillation and privileged information. This approach was adopted by [19] and [15] in cross-modality knowledge transfer. Our graph distillation method is different from prior work [18,26,33,49] in that the privileged information contains multiple modalities and that the distillation directions and weights are dynamically learned rather than being predefined by human experts.

3 Method

Our goal is to assist the training in the target domain with limited labeled data and modalities by leveraging the source domain dataset with abundant examples and multiple modalities. We address the problem by distilling the knowledge from the privileged modalities. Formally, we model action classification and detection as an L-way classification problem, where a "background class" is added in action detection.

Let $\mathcal{D}_t = \{(x_i, y_i)\}_{i=1}^{|\mathcal{D}_t|}$ denote the training set in the target domain, where $x_i \in \mathbb{R}^d$ is the input and $y_i \in \mathbb{R}$ is an integer denoting the class label. Since training data in the target domain is limited, we are interested in transferring knowledge from a source dataset $\mathcal{D}_s = \{(x_i, \mathcal{S}_i, y_i)\}_{i=1}^{|\mathcal{D}_s|}$, where $|\mathcal{D}_s| \gg |\mathcal{D}_t|$, and the source and target data may have different classes. The new element $\mathcal{S}_i = \{x_i^{(1)}, ..., x_i^{(|\mathcal{S}|)}\}$ is a set of privileged information about the i-th sample, where the superscript indexes the modality in $\mathcal{S}_i$. As an example, x_i could be the depth image of the i-th frame in a video and $x_i^{(1)}, x_i^{(2)}, x_i^{(3)} \in \mathcal{S}_i$ might be RGB, optical flow and skeleton features about the same frame, respectively. For action classification, we employ the standard softmax cross entropy loss:

$$\ell_c(f(x_i), y_i) = -\sum_{j=1}^{L} \mathbb{1}(y_i = j) \log \sigma(f(x_i)), \tag{1}$$

where $\mathbb{1}$ is the indicator function and σ is the softmax function. The class prediction function $f : \mathbb{R}^d \to [1, L]$ computes the probability for each action class.

In the rest of this section, Section 3.1 discusses the overall objective of privileged knowledge distillation. Section 3.2 details the proposed graph distillation over multiple modalities.

3.1 Knowledge Distillation with Privileged Modalities

To leverage the privileged information in the source domain data, we follow the standard transfer learning paradigm. We first train a model with graph distillation using all modalities in the source domain, and then transfer only the visual encoders (detailed in Sec 4.1) of the target domain modalities. Finally, the visual encoder is finetuned with the rest of the target model on the target task. The visual feature encoding step is shared between the tasks in the source and target data and is therefore intuitive to use the same visual encoder architecture (as shown in Fig. 2) for both tasks.

To train a graph distillation model on the source data, we minimize:

$$\min \frac{1}{|\mathcal{D}_s|} \sum_{(x_i, y_i) \in \mathcal{D}_s} \ell_c(f(x_i), y_i) + \ell_m(x_i, \mathcal{S}_i). \tag{2}$$

The loss consists of two parts: the first term is the standard classification loss in Eq. (1) and the latter is the imitation loss [18]. The imitation loss is often defined as the cross-entropy loss on the *soft logits* [18]. In existing literatures, the imitation loss is computed using a pre-specified distillation direction. For example, Hinton et al. [18] computed the soft logits by $\sigma(f_{\mathcal{S}}(x_i)/T)$, where T is the temperature, and $f_{\mathcal{S}}$ is the class prediction function of the cumbersome model. Gupta et al. [15] employed the "soft logits" obtained from different layers of the labeled modality. In both cases, the distillation is pre-specified, *i.e.*, from a cumbersome model to a small model in [18] or from a labeled modality to an unlabeled modality in [15]. In our problem, the privileged information comes

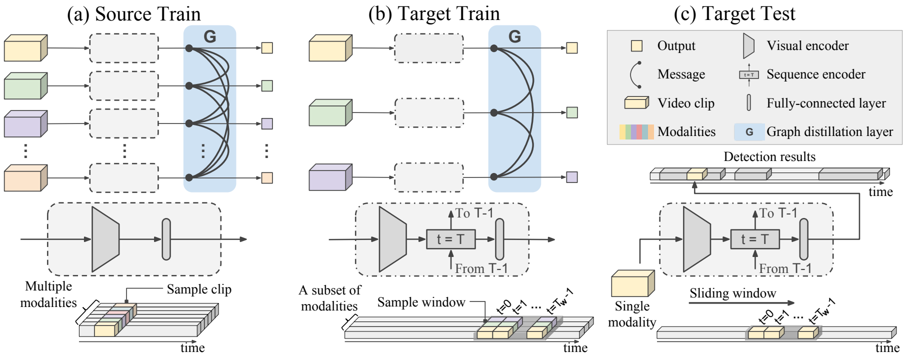

Fig. 2. An overview of our network architectures. (a) Action classification with graph distillation (attached as a layer) in the source domain. The visual encoders for each modality are trained. (b) Action detection with graph distillation in the target domain at training time. In our setting, the target training modalities is a subset of the source modalities (one or more). Note that the visual encoder trained in the source is transferred and finetuned in the target. (c) Action detection in the target domain at test time, with a single modality.

from multiple heterogeneous modalities and it is difficult to pre-specify the distillation directions and weights. To this end, our the imitation loss in Eq. (2) is derived from a dynamic distillation graph.

3.2 Graph Distillation

First, consider a special case of graph distillation where only two modalities are involved. We employ an imitation loss that combines the logits and feature representation. For notation convenience, we denote x_i as $x_i^{(0)}$ and fold it into $\mathcal{S}_i = \{x_i^{(0)}, \cdots, x_i^{(|\mathcal{S}|)}\}$. Given two modalities $a, b \in [0, |\mathcal{S}|]$ $(a \neq b)$, we use the network architectures discussed in Section 4 to obtain the logits and the output of the last convolution layer as the visual feature representation.

The proposed imitation loss between two modalities consists of the loss on the logits l_{logits} and the representation l_{rep}. The cosine distance is used on both logits and representations as we found the angle of the prediction to be more indicative and better than KL divergence or L1 distance for our problem.

The imitation loss ℓ_m from modality b to a is computed by the weighted sum of the logits loss and the representation loss. We encapsulate the loss between two modalities into a message $m_{a \leftarrow b}$ passing from b to a, calculated from:

$$m_{a \leftarrow b}(x_i) = \ell_m(x_i^{(a)}, x_i^{(b)}) = \lambda_1 l_{logits} + \lambda_2 l_{rep}, \tag{3}$$

where λ_1 and λ_2 are hyperparameters. Note that the message is directional, and $m_{a \leftarrow b}(x_i) \neq m_{b \leftarrow a}(x_i)$.

For multiple modalities, we introduce a directed graph of $|\mathcal{S}|$ vertices, named *distillation graph*, where each vertex v_k represents a modality and an edge $e_{k \leftarrow j} \geq 0$ is a real number indicating the strength of the connection from v_j to v_k. For a fixed graph, the total imitation loss for the modality k is:

$$\ell_m(x_i^{(k)}, \mathcal{S}_i) = \sum_{v_j \in \mathcal{N}(v_k)} e_{k \leftarrow j} \cdot m_{k \leftarrow j}(x_i), \tag{4}$$

where $\mathcal{N}(v_k)$ is the set of vertices pointing to v_k.

To exploit the dynamic interactions between modalities, we propose to learn the distillation graph along with the original network in an end-to-end manner. Denote the graph by an adjacency matrix $\mathbf{G}$ where $\mathbf{G}_{jk} = e_{k \leftarrow j}$. Let ϕ_k^l be the logits and ϕ_k^{l-1} be the representation for modality k, where l indicates the number of layers in the network. Given an example x_i, the graph is learned by:

$$z_i^{(k)}(x_i) = W_{11}\phi_k^{l-1}(x_i^{(k)}) + W_{12}\phi_k^l(x_i^{(k)}), \tag{5}$$

$$\mathbf{G}_{jk}(x_i) = e_{k \leftarrow j} = W_{21}[z_i^{(j)}(x_i) \| z_i^{(k)}(x_i)] \tag{6}$$

where W_{11}, W_{12} and W_{21} are parameters to learn and $\cdot \| \cdot$ indicates the vector concatenation. W_{21} maps a pair of inputs to an entry in $\mathbf{G}$. The entire graph is learned by repetitively applying Eq. (6) over all pairs of modalities in $\mathcal{S}$.

As a distillation graph is expected to be sparse, we normalize $\mathbf{G}$ such that the nonzero weights are dispersed over a small number of vertices. Let $\mathbf{G}_{j:} \in \mathbb{R}^{1 \times |\mathcal{S}|}$ be the vector of its j-th row. The graph is normalized:

$$\mathbf{G}_{j:}(x_i) = \sigma(\alpha[\mathbf{G}_{j1}(x_i), ..., \mathbf{G}_{j|\mathcal{S}|}(x_i)]), \tag{7}$$

where α is used to scale the input to the softmax operator.

The message passing on distillation graph can be conveniently implemented by attaching a new layer to the original network. As shown in Fig. 2(a), each vertex represents a modality and the messages are propagated on the graph layer. In the forward pass, we learn a $\mathbf{G} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$ by Eq. (6) and (7) and compute the message matrix $\mathbf{M} \in \mathbb{R}^{|\mathcal{S}| \times |\mathcal{S}|}$ by Eq. (3) such that $\mathbf{M}_{jk}(x_i) = m_{k \leftarrow j}(x_i)$. The imitation loss to all modalities is calculated by:

$$\ell_m = (\mathbf{G}(x_i) \odot \mathbf{M}(x_i))^T \mathbf{1}, \tag{8}$$

where $\mathbf{1} \in \mathbb{R}^{|\mathcal{S}| \times 1}$ is a column vector of ones; $\odot$ is the element-wise product between two matrices; $\boldsymbol{\ell}_\mathbf{m} \in \mathbb{R}^{|\mathcal{S}| \times 1}$ contains imitation loss for every modality in $\mathcal{S}$. In the backward propagation, the imitation loss ℓ_m is incorporated in Eq. (2) to compute the gradient of the total training loss. This graph distillation layer is end-to-end trained with the rest of the network. As shown, the distillation graph is an important and essential structure which not only provides a base for learning dynamic message passing through modalities but also models the distillation as a few matrix operations which can be conveniently implemented as a new layer in the network.

For a modality, its performance on the cross-validation set often turns out to be a reasonable estimator to its contribution in distillation. Therefore, we add a constant bias term $\mathbf{c}$ in Eq. (7), where $\mathbf{c} \in \mathbb{R}^{|\mathcal{S}| \times 1}$ and c_j is set w.r.t. the cross-validation performance of the modality j and $\sum_{k=1}^{|\mathcal{S}|} c_k = 1$. Therefore, Eq. (8) can be rewritten as:

$$\ell_m = ((\mathbf{G}(x_i) + \mathbf{1}\mathbf{c}^T) \odot \mathbf{M}(x_i))^T \mathbf{1} \tag{9}$$

$$= (\mathbf{G}(x_i) \odot \mathbf{M}(x_i))^T \mathbf{1} + (\mathbf{G}_{prior} \odot \mathbf{M}(x_i))^T \mathbf{1} \tag{10}$$

where $\mathbf{G}_{prior} = \mathbf{1}\mathbf{c}^T$ is a constant matrix. Interestingly, by adding a bias term in Eq. (7), we decompose the distillation graph into two graphs: a learned example-specific graph $\mathbf{G}$ and a prior modality-specific graph $\mathbf{G}_{prior}$ that is independent to specific examples. The messages are propagated on both graphs and the sum of the message is used to compute the total imitation loss. There exists a physical interpretation of the learning process. Our model learns a graph based on the likelihood of observed examples to exploit complementary information in $\mathcal{S}$. Meanwhile, it imposes a prior to encouraging accurate modalities to provide more contribution. By adding a constant bias, we use a more computationally efficient approach than actually performing message passing on two graphs.

So far, we have only discussed the distillation on the source domain. In practice, our method may also be applied to the target domain on which privileged modality is available. In this case, we apply the same method to minimize Eq. (2) on the target training data. As illustrated in Fig. 2(b), a graph distillation layer is added during the training of the target model. At the test time, as shown in Fig. 2(c), only a single modality is used.

4 Action Classification and Detection Models

In this section, we discuss our network architectures as well as the training and testing procedures for action classification and detection. The objective of action classification is to classify a trimmed video into one of the predefined categories. The objective of action detection is to predict the start time, the end time, and the class of an action in an untrimmed video.

4.1 Network Architecture

For action classification, we encode a short clip of video into a feature vector using the visual encoder. For action detection, we first encode all clips in a window of video (a window consists of multiple clips) into initial feature vectors using the visual encoder, then feed these initial feature vectors into a sequence encoder to generate the final feature vectors. For either task, each feature vector is fed into a task-specific linear layer and a softmax layer to get the probability distribution across classes for each clip. Note that a background class is added for action detection. Our action classification and detection models are inspired by [50] and [37], respectively. We design two types of visual encoders depending on the input modalities.

Visual Encoder for Images. Let $X = \{x_t\}_{t=1}^{T_c}$ denote a video clip of image modalities (*e.g.* RGB, depth, flow), where $x_t \in \mathbb{R}^{H \times W \times C}$, T_c is the number of frames in a clip, and $H \times W \times C$ is the image dimension. Similar to the temporal stream in [50], we stack the frames into a $H \times W \times (T_c \cdot C)$ tensor and encode the video clip with a modified ResNet-18 [17] with $T_c \cdot C$ input channels and without the last fully-connected layer. Note that we do not use the Convolutional 3D (C3D) network [3,52] because it is hard to train with limited amount of data [3].

Visual Encoder for Vectors. Let $X = \{x_t\}_{t=1}^{T_c}$ denote a video clip of vector modalities (*e.g.* skeleton), where $x_t \in \mathbb{R}^D$ and D is the vector dimension. Similar to [24], we encode the input with a 3-layer GRU network [6] with T_c timesteps. The encoded feature is computed as the average of the outputs of the highest layer across time. The hidden size of the GRU is chosen to be the same as the output dimension of the visual encoder for images.

Sequence Encoder. Let $X = \{x_t\}_{t=1}^{T_c \cdot T_w}$ denote a window of video with T_w clips, where each clip contains T_c frames. The visual encoder first encodes each clip individually into a single feature vector. These T_w feature vectors are then passed into the sequence encoder, which is a 1-layer GRU network, to obtain the class distributions of these T_w clips. Note that the sequence encoder is only used in action detection.

4.2 Training and Testing

Our proposed graph distillation can be applied to both action detection and classification. For action detection, we show that our method can optionally pre-train the action detection model on action classification tasks, and graph distillation can be applied in both pre-training and training stages. Both models are trained to minimize the loss in Eq. (2) on per-clip classification, and the imitation loss is calculated based on the representations and the logits.

Action Classification. Fig. 2(a) shows how graph distillation is applied in training. During training, we randomly sample a video clip of T_c frames from the video, and the network outputs a single class distribution. During testing, we uniformly sample multiple clips spanning the entire video and average the outputs to obtain the final class distribution.

Action Detection. Fig. 2(b) and Fig. 2(b) show how graph distillation is applied in training and testing, respectively. As discussed earlier, graph distillation can be applied to both the source domain and the target domain. During training, we randomly sample a window of T_w clips from the video, where each clip is of length T_c and is sampled with step size s_c. As the data is imbalanced, we set a class-specific weight based on its inverse frequency in the training set. During testing, we uniformly sample multiple windows spanning the entire video with step size s_w, where each window is sampled in the same way as training. The outputs of the model are the class distributions on all clips in all windows (potentially with overlaps depending on s_w). These outputs are then post-processed using the method in [37] to generate the detection results, where the activity threshold γ is introduced as a hyperparameter.

5 Experiments

In this section, we evaluate our method on two large-scale multimodal video benchmarks. The results show that our method outperforms representative baseline methods and achieves the state-of-the-art performance on both benchmarks.

5.1 Datasets and Setups

We evaluate our method on two large-scale multimodal video benchmarks: NTU RGB+D [46] (classification) and PKU-MMD [28] (detection). These datasets are selected for the following reasons. (1) They are (one of the) largest RGB-D video benchmarks in each category. (2) The privileged information transfer is reasonable because the domains of the two datasets are similar. (3) They contain abundant modalities, which are required for graph distillation.

We use NTU RGB+D as our dataset in the source domain, and PKU-MMD in the target domain. In our experiments, unless stated otherwise, we apply graph distillation whenever applicable. Specifically, the visual encoders of all modalities are jointly trained on NTU RGB+D by graph distillation. On PKU-MMD, after initializing the visual encoder with the pre-trained weights obtained from NTU RGB+D, we also learn all available modalities by graph distillation on the target domain. By default, only a single modality is used at test time.

NTU RGB+D [46]. It contains 56,880 videos from 60 action classes. Each video has exactly one action class and comes with four modalities: RGB, depth, 3D joints, and infrared. The training and testing sets have 40,320 and 16,560 videos, respectively. All results are reported with cross-subject evaluation.

PKU-MMD [28]. It contains 1,076 long videos from 51 action classes. Each video contains approximately 20 action instances of various lengths and consists of four modalities: RGB, depth, 3D joints, and infrared. All results are evaluated based on the Average Precision (mAP) at different temporal Intersection over Union (tIoU) thresholds between the predicted and the ground truth intervals.

Modalities. We use a total of six modalities in our experiments: RGB, depth (D), optical flow (F), and three skeleton features (S) named Joint-Joint Distances (JJD), Joint-Joint Vector (JJV), and Joint-Line Distances (JLD) [9,24], respectively. The RGB and depth videos are provided in the datasets. The optical flow is calculated on the RGB videos using the dual TV-L1 method [62]. The three spatial skeleton features are extracted from 3D joints using the method in [9] and [24]. Note that we select a subset of the ten skeleton features in [9,24] to ensure the simplicity and reproducibility of our method, and our approach can potentially perform better with the complete set of features.

Baselines. In addition to comparing with the state-of-the-art, we implement three representative baselines that could be used to leverage multimodal privileged information: *multi-task learning* [4], *knowledge distillation* [18], and *cross-modal distillation* [15]. For the multi-task model, we predict the raw pixels of the other modalities from the representation of a single modality, and use the L_2 distance as the multi-task loss. For the distillation methods, the imitation loss is calculated as the high-temperature cross-entropy loss on the soft logits [18], and

Table 1. Comparison with state-of-the-art on NTU RGB+D. Our models are trained on all modalities and tested on the single modality specified in the table. The available modalities are RGB, depth (D), optical flow (F), and skeleton (S).

Method	Test Modality	mAP	Method	Test Modality	mAP
Shahroudy [47]	RGB+D	0.749	Ours	RGB	**0.895**
Liu [29]	RGB+D	0.775	Ours	D	0.875
Liu [32]	S	0.800	Ours	F	0.857
Ding [9]	S	0.823	Ours	S	0.837
Li [24]	S	0.829			

Table 2. Comparison of action detection methods on PKU-MMD with state-of-the-art models. Our models are trained with graph distillation using all privileged modalities and tested on the modalities specified in the table. “Transfer” refers to pre-training on NTU RGB+D on action classification. The available modalities are RGB, depth (D), optical flow (F), and skeleton (S).

		mAP @ tIoU thresholds (θ)		
Method	Test Modality	0.1	0.3	0.5
Deep RGB (DR) [28]	RGB	0.507	0.323	0.147
Qin and Shelton [43]	RGB	0.650	0.510	0.294
Deep Optical Flow (DOF) [28]	F	0.626	0.402	0.168
Raw Skeleton (RS) [28]	S	0.479	0.325	0.130
Convolution Skeleton (CS) [28]	S	0.493	0.318	0.121
Wang and Wang [54]	S	0.842	-	0.743
RS+DR+DOF [28]	RGB+F+S	0.647	0.476	0.199
CS+DR+DOF [28]	RGB+F+S	0.649	0.471	0.199
Ours (w/o | w/ transfer)	RGB	0.824 | 0.880	0.813 | 0.868	0.743 | 0.801
Ours (w/o | w/ transfer)	D	0.823 | 0.872	0.817 | 0.860	0.752 | 0.792
Ours (w/o | w/ transfer)	F	0.790 | 0.826	0.783 | 0.814	0.708 | 0.747
Ours (w/o | w/ transfer)	S	0.836 | 0.857	0.823 | 0.846	0.764 | 0.784
Ours (w/ transfer)	RGB+D+F+S	**0.903**	**0.895**	**0.833**

L_2 loss on both representations and soft logits in cross-modal distillation [15]. These distillation methods originally only support two modalities, and therefore we average the pairwise losses to get the final loss.

Implementation Details. For action classification, we train the visual encoder from scratch for 200 epochs using SGD with momentum with learning rate 10^{-2} and decay to 10^{-1} at epoch 125 and 175. λ_1 and λ_2 are set to $10, 5$ respectively in Eq. (3). At test time we sample 5 clips for inference. For action detection, the visual and sequence encoder are trained for 400 epochs. The visual encoder is trained using SGD with momentum with learning rate 10^{-3}, and the sequence encoder is trained with the Adam optimizer [21] with learning rate 10^{-3}. The activity threshold γ is set to 0.4. For both tasks, we down-sample the frame rates of the datasets by a factor of 3. The clip length and detection window T_c and T_w are both set to 10. For the graph distillation, α is set to 10 in Eq. (7). The output dimensions of the visual and sequence encoder are both set to 512. Since it is nontrivial to jointly train on multiple modalities from scratch, we employ curriculum learning [1] to train the distillation graph. To do so, we first fix the distillation graph as an identity matrix (uniform graph) in the first 200 epochs.

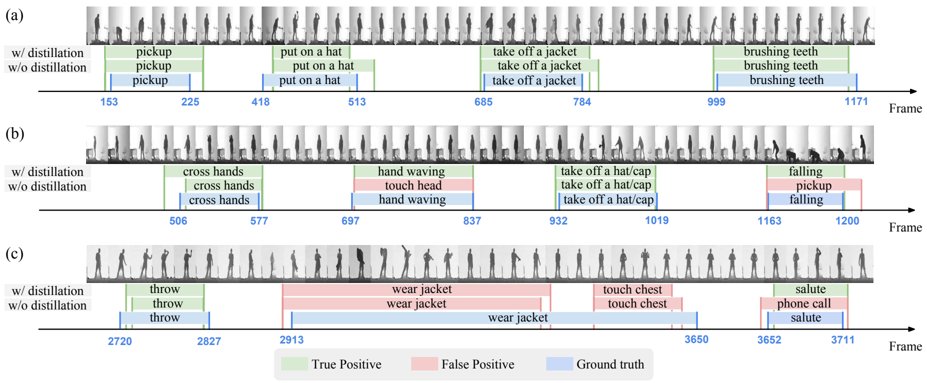

Fig. 3. A comparison of the prediction results on PKU-MMD. (a) Both models make correct predictions. (b) The model without distillation in the source makes errors. Our model learns motion and skeleton information from the privileged modalities in the source domain, which helps the prediction for classes such as "hand waving" and "falling". (c) Both models make reasonable errors.

In the second stage, we compute the constant vector $\mathbf{c}$ in Eq. (9) according to the cross-validation results, and then learn the graph in an end-to-end manner.

5.2 Comparison with State-of-the-Art

Action Classification. Table 1 shows the comparison of action classification with state-of-the-art models on NTU RGB+D dataset. Our graph distillation models are trained and tested on the same dataset in the source domain. NTU RGB+D is a very challenging dataset and has been recently studied in numerous studies [24,29,32,35,47]. Nevertheless, as we see, our model achieves the state-of-the-art results on NTU RGB+D. It yields a 4.5% improvement, over the previous best result, using the depth video and a remarkable 6.6% using the RGB video. After inspecting the results, we found the improvement mainly attributes to the learned graph capturing complementary information across multiple modalities. Fig. 4 shows example distillation graphs learned on NTU RGB+D. The results show that our method, without transfer learning, is effective for action classification in the source domain.

Action Detection. Table 2 compares our method on PKU-MMD with previous work. Our model outperforms existing methods across all modalities. The results substantiate that our method can effectively leverage the privileged knowledge from multiple modalities. Fig. 3 illustrates detection results on the depth modality with and without the proposed distillation.

5.3 Ablation Studies on Limited Training Data

Section 5.2 has shown that our method achieves the state-of-the-art results on two public benchmarks. However, in practice, the training data are often lim-

Table 3. The comparison with (a) baseline methods using Privileged Information (PIs) on mini-NTU RGB+D, (b) distillation graphs on mini-NTU RGB+D and mini-PKU-MMD. Empty graph trains each modality independently. Uniform graph uses a uniform weight in distillation. Prior graph is built according to the cross-validation accuracy of each modality. Learned graph is learned by our method. "D" refers to the depth modality.

(a) Baseline methods using PIs.

Method	mAP / RGB
Empty graph	0.464
Multi-task [4]	0.456
Cross-distillation [15]	0.503
Knowledge distillation [18]	0.524
Learned graph	**0.619**

(b) Different distillation graphs.

	mini-NTU	mini-PKU
Graph	mAP / RGB	mAP @ 0.5 / D
Empty graph	0.464	0.501
Uniform graph	0.537	0.513
Prior graph	0.571	0.515
Learned graph	**0.619**	**0.559**

Table 4. The mAP comparison on mini-PKU-MMD at different tIoU threshold θ. The depth modality is chosen for testing. "src", "trg", and "PI" stand for source, target, and privileged information, respectively.

		mAP @ tIoU thresholds (θ)		
	Method	0.1	0.3	0.5
1	trg only	0.248	0.235	0.200
2	src + trg	0.583	0.567	0.501
3	src w/ PIs + trg	0.625	0.610	0.533
4	src + trg w/ PIs	0.626	0.615	0.559
5	src w/ PIs + trg w/ PIs	0.642	0.629	0.562
6	src w/ PIs + trg	0.625	0.610	0.533
7	src w/ PIs + trg w/ 1 PI	0.632	0.615	0.549
8	src w/ PIs + trg w/ 2 PIs	0.636	0.624	0.557
9	src w/ PIs + trg w/ all PIs	0.642	0.629	0.562

ited in size. To systematically evaluate our method on limited training data, as proposed in the introduction, we construct mini-NTU RGB+D and mini-PKU-MMD by randomly sub-sampling 5% of the training data from their full datasets and use them for training. For evaluation, we test the model on the full test set.

Comparison with Baseline Methods. Table 3(a) shows the comparison with the baseline models that uses privileged information (see Section 5.1). The fact that our method outperforms the representative baseline methods validates the efficacy of the graph distillation method.

Efficacy of Distillation Graph. Table 3(b) compares the performance of predefined and learned distillation graphs. The proposed learned graph is compared with an empty graph (no distillation), a uniform graph of equal weights, and a prior graph computed using the cross-validation accuracy of each modality. Results show that the learned graph structure with modality-specific prior and example-specific information obtains the best results on both datasets.

Efficacy of Privileged Information. Table 4 compares our distillation and transfer under different training settings. The input at test time is a single depth modality. By comparing row 2 and 3 in Table 4, we see that when transferring the

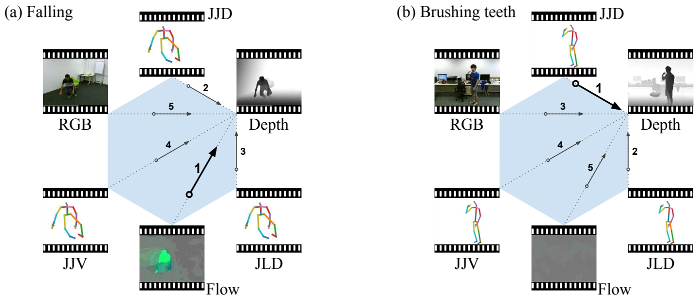

Fig. 4. The visualization of graph distillation on NTU RGB+D. The numbers indicate the ranks of the distillation weights, with 1 being the largest and 5 being the smallest. (a) Class "falling": Our graph assigns more weight to optical flow because optical flow captures the motion information. (b) Class "brushing teeth": In this case, motion is negligible, and our graph assigns the smallest weight to it. Instead, it assigns the largest weight to skeleton data.

visual encoder to the target domain, the one pre-trained with privileged information in the source domain performs better than its counterpart. As discussed in Section 3.2, graph distillation can also be applied to the target domain. By comparing row 3 and 5 (or row 2 and 4) of Table 4, we see that performance gain is achieved by applying the graph distillation in the target domain. The results show that our graph distillation can capture useful information from multiple modalities in both the source and target domain.

Efficacy of Having More Modalities. The last three rows of Table 4 show that performance gain is achieved by increasing the number of modalities used as the privileged information. Note that the test modality is depth, the first privileged modality is RGB, and the second privileged modality is the skeleton feature JJD. The results also suggest that these modalities provide each other complementary information during the graph distillation.

6 Conclusion

This paper tackles the problem of action classification and detection in multimodal video with limited training data and partially observed modalities. We propose the novel graph distillation method to assist the training of the model by leveraging privileged modalities dynamically. Our model outperforms representative baseline methods and achieves the state-of-the-art for action classification on NTU RGB+D dataset and action detection on the PKU-MMD. A direction for future work is to combine graph distillation with advanced transfer learning and domain adaptation techniques.

Acknowledgement. This work was supported in part by Stanford Computer Science Department and Clinical Excellence Research Center. We specially thank Li-Jia Li, De-An Huang, Yuliang Zou, and all the anonymous reviewers for their valuable comments.

References

1. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: International Conference on Machine Learning (ICML) (2009)
2. Buch, S., Escorcia, V., Shen, C., Ghanem, B., Niebles, J.C.: SST: Single-stream temporal action proposals. In: CVPR (2017)
3. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Computer Vision and Pattern Recognition (CVPR) (2017)
4. Caruana, R.: Multitask learning. In: Learning to learn, pp. 95–133. Springer (1998)
5. Chen, X., Gupta, A.: Webly supervised learning of convolutional networks. In: International Conference on Computer Vision (ICCV) (2015)
6. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling (2014)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition (CVPR) (2009)
8. Ding, Z., Shao, M., Fu, Y.: Missing modality transfer learning via latent low-rank constraint. IEEE Transactions on Image Processing **24**(11), 4322–4334 (Nov 2015). https://doi.org/10.1109/TIP.2015.2462023
9. Ding, Z., Wang, P., Ogunbona, P.O., Li, W.: Investigation of different skeleton features for cnn-based 3d action recognition. arXiv preprint arXiv:1705.00835 (2017)
10. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2015)
11. Escorcia, V., Heilbron, F.C., Niebles, J.C., Ghanem, B.: Daps: Deep action proposals for action understanding. In: European Conference on Computer Vision (ECCV) (2016)
12. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: International Conference on Computer Vision (ICCV). pp. 2960–2967 (2013)
13. Girshick, R.: Fast r-cnn. In: International Conference on Computer Vision (ICCV) (2015)
14. Gorban, A., Idrees, H., Jiang, Y., Zamir, A.R., Laptev, I., Shah, M., Sukthankar, R.: Thumos challenge: Action recognition with a large number of classes. In: Computer Vision and Pattern Recognition (CVPR) Workshop (2015)
15. Gupta, S., Hoffman, J., Malik, J.: Cross modal distillation for supervision transfer. In: Computer Vision and Pattern Recognition (CVPR) (2016)
16. Haque, A., Guo, M., Alahi, A., Yeung, S., Luo, Z., Rege, A., Jopling, J., Downing, N.L., Beninati, W., Singh, A., Platchek, T., Milstein, A., Fei-Fei, L.: Towards vision-based smart hospitals: A system for tracking and monitoring hand hygiene compliance. Proceedings of Machine Learning for Healthcare 2017 (2017)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (CVPR) (2016)
18. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. In: NIPS workshop (2015)
19. Hoffman, J., Gupta, S., Darrell, T.: Learning with side information through modality hallucination. In: Computer Vision and Pattern Recognition (CVPR) (2016)
20. Jiang, L., Meng, D., Mitamura, T., Hauptmann, A.G.: Easy samples first: Self-paced reranking for zero-example multimedia search. In: MM (2014)
21. Kingma, P.K., Ba, J.: Adam: A method for stochastic optimization (2015)

22. Koppula, H.S., Gupta, R., Saxena, A.: Learning human activities and object affordances from rgb-d videos. The International Journal of Robotics Research **32**(8), 951–970 (2013)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems (NIPS) (2012)
24. Li, C., Zhong, Q., Xie, D., Pu, S.: Skeleton-based action recognition with convolutional neural networks. arXiv preprint arXiv:1704.07595 (2017)
25. Li, W., Chen, L., Xu, D., Gool, L.V.: Visual recognition in rgb images and videos by learning from rgb-d data. IEEE Transactions on Pattern Analysis and Machine Intelligence **40**(8), 2030–2036 (Aug 2018). https://doi.org/10.1109/TPAMI.2017.2734890
26. Li, Y., Yang, J., Song, Y., Cao, L., Luo, J., Li, J.: Learning from noisy labels with distillation. In: International Conference on Computer Vision (ICCV) (2017)
27. Liang, J., Jiang, L., Meng, D., Hauptmann, A.G.: Learning to detect concepts from webly-labeled video data. In: IJCAI (2016)
28. Liu, C., Hu, Y., Li, Y., Song, S., Liu, J.: Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding. arXiv preprint arXiv:1703.07475 (2017)
29. Liu, J., Akhtar, N., Mian, A.: Viewpoint invariant action recognition using rgb-d videos. arXiv preprint arXiv:1709.05087 (2017)
30. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal lstm with trust gates for 3d human action recognition. In: European Conference on Computer Vision (ECCV) (2016)
31. Liu, J., Wang, G., Hu, P., Duan, L.Y., Kot, A.C.: Global context-aware attention lstm networks for 3d action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2017)
32. Liu, M., Liu, H., Chen, C.: Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition **68**, 346–362 (2017)
33. Lopez-Paz, D., Bottou, L., Schölkopf, B., Vapnik, V.: Unifying distillation and privileged information. In: International Conference on Learning Representations (ICLR) (2016)
34. Luo*, Z., Hsieh*, J.T., Balachandar, N., Yeung, S., Pusiol, G., Luxenberg, J., Li, G., Li, L.J., Downing, N.L., Milstein, A., Fei-Fei, L.: Computer vision-based descriptive analytics of seniors' daily activities for long-term health monitoring. Machine Learning for Healthcare (MLHC) (2018)
35. Luo, Z., Peng, B., Huang, D.A., Alahi, A., Fei-Fei, L.: Unsupervised learning of long-term motion dynamics for videos. In: Computer Vision and Pattern Recognition (CVPR) (2017)
36. Luo, Z., Zou, Y., Hoffman, J., Fei-Fei, L.: Label efficient learning of transferable representations across domains and tasks. In: Advances in neural information processing systems (NIPS) (2017)
37. Montes, A., Salvador, A., Giro-i Nieto, X.: Temporal activity detection in untrimmed videos with recurrent neural networks. arXiv preprint arXiv:1608.08128 (2016)
38. Motiian, S., Piccirilli, M., Adjeroh, D.A., Doretto, G.: Information bottleneck learning using privileged information for visual recognition. In: Computer Vision and Pattern Recognition (CVPR) (2016)
39. Ni, B., Wang, G., Moulin, P.: Rgbd-hudaact: A color-depth video database for human daily activity recognition. In: Consumer Depth Cameras for Computer Vision (2013)

40. Noury, N., Fleury, A., Rumeau, P., Bourke, A., Laighin, G., Rialle, V., Lundy, J.: Fall detection-principles and methods. In: Engineering in Medicine and Biology Society (2007)
41. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**(10), 1345–1359 (2010). https://doi.org/10.1109/TKDE.2009.191
42. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on knowledge and data engineering **22**(10), 1345–1359 (2010)
43. Qin, Z., Shelton, C.R.: Event detection in continuous video: An inference in point process approach. IEEE Transactions on Image Processing **26**(12), 5680–5691 (2017)
44. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Computer Vision and Pattern Recognition (CVPR) (2016)
45. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Neural Information Processing Systems (NIPS) (2015)
46. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In: Computer Vision and Pattern Recognition (CVPR) (2016)
47. Shahroudy, A., Ng, T.T., Gong, Y., Wang, G.: Deep multimodal feature analysis for action recognition in rgb+ d videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2017)
48. Shao, L., Cai, Z., Liu, L., Lu, K.: Performance evaluation of deep feature learning for rgb-d image/video classification. Information Sciences **385**, 266–283 (2017)
49. Shi, Z., Kim, T.K.: Learning and refining of privileged information-based rnns for action recognition from depth sequences. In: Computer Vision and Pattern Recognition (CVPR) (2017)
50. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems (NIPS) (2014)
51. Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from rgbd images. In: AAAI workshop on Pattern, Activity and Intent Recognition (2011)
52. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: International Conference on Computer Vision (ICCV) (2015)
53. Vapnik, V., Vashist, A.: A new learning paradigm: Learning using privileged information. Neural networks **22**(5), 544–557 (2009)
54. Wang, H., Wang, L.: Learning robust representations using recurrent neural networks for skeleton based action classification and detection. In: International Conference on Multimedia & Expo Workshops (ICMEW) (2017)
55. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: Computer Vision and Pattern Recognition (CVPR) (2012)
56. Wang, Z., Ji, Q.: Classifier learning with hidden information. In: Computer Vision and Pattern Recognition (CVPR) (2015)
57. Xu, D., Ouyang, W., Ricci, E., Wang, X., Sebe, N.: Learning cross-modal deep representations for robust pedestrian detection. In: Computer Vision and Pattern Recognition (CVPR) (2017)

58. Yang, H., Zhou, J.T., Cai, J., Ong, Y.S.: Miml-fcn+: Multi-instance multi-label learning via fully convolutional networks with privileged information. In: Computer Vision and Pattern Recognition (CVPR) (2017)
59. Yeung, S., Ramanathan, V., Russakovsky, O., Shen, L., Mori, G., Fei-Fei, L.: Learning to learn from noisy web videos (2017)
60. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Advances in neural information processing systems (NIPS) (2014)
61. Yu, M., Liu, L., Shao, L.: Structure-preserving binary representations for rgb-d action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **38**(8), 1651–1664 (2016)
62. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime tv-l 1 optical flow. Pattern Recognition pp. 214–223 (2007)
63. Zhang, S., Liu, X., Xiao, J.: On geometric features for skeleton-based action recognition using multilayer lstm networks. In: IEEE Winter Conference on Applications of Computer Vision (WACV) (2017)
64. Zhang, Z., Conly, C., Athitsos, V.: A survey on vision-based fall detection. In: Conference on PErvasive Technologies Related to Assistive Environments (PETRA) (2015)
65. Zhao, Y., Xiong, Y., Wang, L., Wu, Z., Tang, X., Lin, D.: Temporal action detection with structured segment networks. In: International Conference on Computer Vision (ICCV) (2017)