Attention Transfer from Web Images for Video Recognition

Junnan Li
NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore
lijunnan@u.nus.edu

Yongkang Wong
Interactive & Digital Media Institute
National University of Singapore
yongkang.wong@nus.edu.sg

Qi Zhao
Department of Computer Science and Engineering
University of Minnesota
qzhao@cs.umn.edu

Mohan S. Kankanhalli
School of Computing
National University of Singapore
mohan@comp.nus.edu.sg

ABSTRACT

Training deep learning based video classifiers for action recognition requires a large amount of labeled videos. The labeling process is labor-intensive and time-consuming. On the other hand, large amount of weakly-labeled images are uploaded to the Internet by users everyday. To harness the rich and highly diverse set of Web images, a scalable approach is to crawl these images to train deep learning based classifier, such as Convolutional Neural Networks (CNN). However, due to the domain shift problem, the performance of Web images trained deep classifiers tend to degrade when directly deployed to videos. One way to address this problem is to fine-tune the trained models on videos, but sufficient amount of annotated videos are still required. In this work, we propose a novel approach to transfer knowledge from image domain to video domain. The proposed method can adapt to the target domain (i.e. video data) with limited amount of training data. Our method maps the video frames into a low-dimensional feature space using the class-discriminative spatial attention map for CNNs. We design a novel Siamese EnergyNet structure to learn energy functions on the attention maps by jointly optimizing two loss functions, such that the attention map corresponding to a ground truth concept would have higher energy. We conduct extensive experiments on two challenging video recognition datasets (i.e. TVHI and UCF101), and demonstrate the efficacy of our proposed method.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**; **Transfer learning**; *Neural networks*;

KEYWORDS

Domain Adaptation; Action Recognition; Attention Map

MM '17, October 23–27, 2017, Mountain View, CA, USA

ACM ISBN 978-1-4503-4906-2/17/10…$15.00
https://doi.org/10.1145/3123266.3123432

Figure 1: Weakly-labeled Web images collected from commercial search engines provide rich and diverse training data for model training. However, there exist domain shift between Web images and videos, where the data distributions of the two domains differ. (Left: Web images retrieved by keywords. Right: Video frames from TVHI dataset.)

1 INTRODUCTION

Recent advancements in deep Convolutional Neural Network (CNN) have led to promising results in large-scale video classification [8, 19, 32, 38, 42]. A prerequisite of deep model training is the availability of large-scale labeled training data. However, the acquisition and annotation of such datasets (e.g. UCF101 [35], ActivityNet [14], Sport1M [19]) is often labor-intensive. On the other hand, Web images are easier to collect by querying widely available commercial search engines. Unlike videos, Web images usually capture the representative moments of events or actions and provide more diverse examples for each concept. This makes them a good auxiliary source to enhance video concept recognition.

A naive approach to harness information from Web images is to directly apply the Web images trained classifier to video data. However, learning video concepts from Web images introduces the domain shift problem [29], where the variation in data between the source and target domain jeopardize the performance of the trained classifier. As shown in Figure 1, the Web images (left) differ from video frames (right) in background, color, lighting, actors, and so on. Several recent works [10, 11, 23, 37] address this by jointly utilizing images and videos to train shared CNNs, and use the shared CNNs to map both images and videos into the same feature space. However, in order to learn domain-invariant feature representations for the shared CNNs, sufficient amount of annotated video data are required for training. It is expensive and time-consuming to collect labeled training data for various video domains (e.g. movies, consumer videos, egocentric videos, *etc.*).

(a) Original Frame (b) Handshake (c) High Five (d) Hug (e) Kiss

Figure 2: Examples of heatmap overlayed on a video frame. The respective spatial attention maps are generated with a Web image trained CNN.

There exist two common scenarios to adapt the classifier trained on a source domain to the target domain: (1) *unsupervised* scenario: no labeled data in the target domain is available; and (2) *supervised* scenario: labeled training data is available in the target domain. In this work, we propose a new approach to transfer knowledge from Web images to videos. Comparing with using the features from the last layers of CNNs for video recognition [10, 11, 23, 37], our method can achieve better performance on target domain under both unsupervised and supervised scenarios.

The proposed approach is based upon class-discriminative spatial attention maps for CNNs [31, 47, 48], which are initially proposed to visualize the discriminative regions on images that are 'important' for each class prediction. The spatial attention maps are later applied for weakly-supervised object localization [31, 48], and to transfer knowledge from deeper networks to shallower networks [47].

Our method roots from this observation: Given a video frame and a Web images trained concept classifier (i.e. CNN model), if a concept appears in that frame, certain regions in the spatial attention map w.r.t. that concept would have high energy heatmap (or activation). On the other hand, if a concept does not appear in the frame, the corresponding spatial attention map would have low and sparse activations (see Figure 2). Therefore, the spatial attention maps are informative of the concept. Furthermore, since the spatial attention map is computed from the features of the convolutional layer of CNNs (see Section 3.1 for details), we presume that it is more domain-invariant compared with the features of the last fully-connected (fc) layer. In this study, we propose approaches to exploit the spatial attention map, so that the video classifier trained on Web images would suffer less from the domain shift.

Our contributions are as follows:

- We propose to use class-discriminative spatial attention maps for cross-domain knowledge transfer from Web images to videos. Experiments on action/interaction recognition with two challenging datasets (i.e. TVHI and UCF101) demonstrate the efficacy of the proposed methods.
- We propose an energy-based method on the spatial attention maps, with the aim of assigning the highest energy to the ground truth concept. We design an Energy Network to learn class-specific energy functions, and construct a Siamese structure that jointly optimizes over two loss functions, energy loss and triplet loss. We show that our method can achieve superior performance over several baselines.
- We collected a new Human Interaction Image (HII) dataset to facilitate research in interaction recognition, which contains images for four types of interactions (please refer to Section 4.1 for more details).

The rest of the paper is organized as follows: Section 2 reviews the related work. Section 3 delineates the details of the proposed method. Section 4 elaborates on the experiments and discusses the results. Section 5 concludes the paper.

2 RELATED WORK

2.1 Action Recognition on Unconstrained Data

Action recognition is a very active research area and has been widely studied. A detailed survey can be found in [6]. Most existing works take a two-step approach: feature-extraction and classifier training. Many hand-crafted features are designed for video appearance and motion representations, where Improved Dense Trajectories (IDT) [40] combined with Fisher vector coding [27] achieve state-of-the-art performance. Recent approaches use deep networks (particularly CNNs) to jointly learn feature extractors and classifiers for action recognition. Tran *et al.* [38] use 3D CNNs to learn spatial-temporal features. Simonyan and Zisserman [32] propose two-stream networks: one stream captures spatial information from video frames and the other stream captures motion information from stacked optical flows. Recurrent Neural Networks (RNNs), with the ability to model sequential information, have also been utilized for action recognition [8, 25]. However, these deep learning based approaches all require large amount of well-labeled videos to avoid overfitting.

2.2 Learning from Web Data

To harness the information from large-scale Web images, several works use Web images as auxiliary training data for video recognition [10, 11, 23, 37]. Ma *et al.* [23] collect a large web action image dataset, and achieve performance gain by combining web images with video frames to train CNNs. Sun *et al.* [37] propose a domain transfer approach for action localization, where they iteratively train a shared CNN on video frames and Web images. Gan *et al.* [10, 11] jointly exploit Web images and Web videos, and propose a mutually voting approach to filter noisy Web images and video frames [10].

Another common usage for Web images is to learn semantic concept detectors and apply them for video retrieval [3, 7, 34, 44].

Chen *et al.* [3] discover concepts from tags of Web images, whereas Singh *et al.* [34] construct pairs of concepts to crawl web images for training concept detectors. However, due to the domain shift between Web images and videos, their detectors are not suitable for zero-shot video recognition and required to be retrained on videos.

Web images are inherently noisy. Several solutions are proposed to train CNNs on noisily labeled data [5, 36, 45]. However, recent studies show that state-of-the-art CNNs trained with large-scale noisily labeled images are surprisingly effective in a range of vision problems [18, 20]. This suggests that learning from weakly-labeled Web images is a scalable solution to train deep networks.

2.3 Domain Adaptation

Domain shift refers to the situation where data distribution differs between source domain and target domain, causing the classifier learned from source domain to perform poorly on target domain. A large number of domain adaptation approaches have been proposed to address this problem, where the key focus is to learn domain-invariant feature representations. There are two common strategies: one approach is to reweight the instances from the source domain [4, 22], and the other approach is to find a mapping function that would align the source distribution with the target domain [1, 9, 12, 26].

Deep networks can learn nonlinear feature representations that manifest the underlying invariant factors and are transferable across domains and tasks [46]. Therefore, deep networks have been recently exploited for domain adaptation. Tzeng *et al.* [39] introduce an adaptation layer and domain confusion loss to learn domain-invariant features across tasks. Bousmalis *et al.* [2] propose domain separation networks that explicitly model the unique characteristics for each domain, so that the invariance of the shared feature representation is improved. Long *et al.* [21] build a deep adaptation network (DAN) that explores multiple kernel variant of maximum mean discrepancies (MK-MMD) to learn transferable features. Yosinski *et al.* [46] explore feature transferability of deep CNNs. They show that while the first layers of a CNN can learn general features, the features from the last layers are more specific and less transferable. Therefore the CNN needs to be fine-tuned on sufficient labeled target data to achieve domain adaptation. Very recently, attention map has been studied as a mechanism to transfer knowledge [47]. Different from the above problem, their work studies knowledge transfer from a deeper network to a shallower network within the same domain.

In this work, we explore the use of attention for cross-domain knowledge transfer from Web images to videos. We show that attention is a more transferable feature compared with the features from the last layers of CNN. Different from previous works that utilize Web images for video recognition [10, 11, 23, 37], our method can better address the domain shift problem with significantly less training data in the target domain.

3 PROPOSED METHOD

In this section, we first briefly overview the Gradient-weighted Class Activation Mapping (Grad-CAM) [31], which is the fundamental component that allows effective domain adaptation from Web image to video. Then, we state the problem statement and detail the proposed domain adaptation approaches.

3.1 Spatial Attention Map

In this work, we adopt Grad-CAM to generate class-discriminative spatial attention maps. Grad-CAM improves upon CAM [48] and required no re-training of the CNN. It has been shown to be a robust method for visualizing deep CNNs, and achieves state-of-the art results for weakly-supervised concept localization in images.

Grad-CAM works as following. Assume that we have a probe image (or image frame from a video sequence) $\boldsymbol{F}$, a target concept c, and a trained CNN model, which the last convolutional layer produce K feature maps $\boldsymbol{A}^k$. The image $\boldsymbol{F}$ is first forwardly propagated through the trained CNN model, then Grad-CAM generates the spatial attention map $L(\boldsymbol{F}, c)$ by a weighted combination of the convolutional feature maps,

$$L(\boldsymbol{F}, c) = ReLU\Big(\sum\nolimits_k \alpha_k^c \boldsymbol{A}^k\Big). \quad (1)$$

The weight α_k^c captures the importance of the k-th feature map for the concept c, and is calculated by backpropagating gradients to the convolutional feature map $\boldsymbol{A}^k$. Prior the backpropagation operation, vector quantization is performed on the gradients for the penultimate layer of the CNN model (the layer before softmax that outputs raw scores) where the dimension of concept c is set to 1 and the remaining as 0. The gradients flowing back to $\boldsymbol{A}^k$ are global-average-pooled to obtain α_k^c. More details of Grad-Cam can be found in [31].

3.2 Problem Statement

Given a set of weakly-labeled Web images and a set of videos that share the same set of concepts $C = \{c_1, c_2, \ldots, c_n\}$, we first adopt state-of-the-art CNN models to pre-train the image-based classifier on the Web images. In this work, instead of using the given videos to train a video-based classifier, our goal is to study how to exploit the Web images trained CNN model to classify the videos. Specifically, we aim to explore domain adaptation mechanism so that the pre-trained model can better adapt to the domain shift when being deployed to videos. We propose to address this problem by using the spatial attention maps $L(\boldsymbol{F}, c_i)$ where $i = 1, 2, \ldots, n$ under two scenarios. Briefly, the first scenario, namely *unsupervised domain adaptation*, directly uses the Web images trained CNN model to the video frames without further training on any videos (Section 3.3), whereas the second scenario, namely *supervised domain adaptation*, utilizes the available training videos to improve the domain adaptation (Section 3.4).

3.3 Unsupervised Domain Adaptation

Directly applying the Web images trained CNN to classify videos would lead to poor performance. This is because the score generated from the last fc layer of a CNN is domain-specific [46]. Therefore, we propose to exploit features from the convolutional layer, using spatial attention map. The attention map incorporates the more general convolutional feature with class information, hence is more transferable across domains.

For a frame $\boldsymbol{F}$ in a given video, let $L(\boldsymbol{F}, c_i)$ be the spatial attention map generated by a pre-trained CNN model for concept c_i. Denote $c_{\text{gt}}^{\boldsymbol{F}}$ as the ground truth concept that exists in $\boldsymbol{F}$, we define an energy function E on the spatial attention map, such that for each video, $\sum_F E(L(\boldsymbol{F}, c_i))$ is the largest when $c_i = c_{\text{gt}}^{\boldsymbol{F}}$, and smaller otherwise.

We define E based on a simple yet effective observation: Assuming that the CNN model has been pre-trained on Web images to detect certain concepts, given a frame and its spatial attention map corresponding to a concept, certain region of the frame would have higher activations in the attention map if the concept is present in the corresponding region (See Figure 2). Therefore, we apply a sliding window over $L(\boldsymbol{F}, c_i)$ with window size of $s \times s$ ($s = 3$) and step size of 1. We then compute the sum of the value of $L(\boldsymbol{F}, c_i)$ within each sliding window as the local activation. The maximum of all local activations is taken as the energy E. For a video with N frames, the output score over each concept is calculated as the mean energy across all frames,

$$score(\boldsymbol{F}, c_i) = \frac{1}{N} \sum\nolimits_{F} E(L(\boldsymbol{F}, c_i)), \tag{2}$$

and the predicted video-level concept is inferred as the one with the highest score,

$$c_v = \arg\max_{c_i} \ score(\boldsymbol{F}, c_i). \tag{3}$$

3.4 Supervised Domain Adaptation

Since each concept may have different activation patterns in the corresponding attention maps, a universal energy function proposed in Section 3.3 could not optimally fit to all concepts. Given training data from the video domain, we design an Energy Network (EnergyNet) to learn an energy function $E_{\text{net}}(\boldsymbol{F}, c)$ that explicitly encodes class information. In other words, the network takes a concept c and the generated attention map $L(\boldsymbol{F}, c)$ as input, and outputs how confident it feels that the concept exists in the frame.

To achieve this, we first flatten the attention map $L(\boldsymbol{F}, c)$ of size $w \times h$ into a vector $\boldsymbol{V}_L \in \mathbb{R}^{w \cdot h}$, and we employ the skip-gram model of word2vec [24] to convert a concept c into a word embedding vector $\boldsymbol{V}_c \in \mathbb{R}^c$. We use the word vectors provided by [17], where words within a multi-words concept (e.g. *walking, with, dog*) are aggregated using Fisher Vectors. This word embedding can capture the semantic relatedness and compositionality between concepts, and lead to better performance compared with a one-hot embedding.

We then embed $\boldsymbol{V}_L$ and $\boldsymbol{V}_c$ into a d-dimensional space using a two-layer fully-connected network with *ReLU* nonlinearity between the layers. The embedding is represented by $f(\boldsymbol{V}_L, \boldsymbol{V}_c) \in \mathbb{R}^d$.

The energy function is defined as,

$$E_{\text{net}}(\boldsymbol{F}, c) = \boldsymbol{W}_f f(\boldsymbol{V}_L, \boldsymbol{V}_c), \tag{4}$$

where $\boldsymbol{W}_f \in \mathbb{R}^{1 \times d}$ is the weight of the last fc layer for the Energy Network. Similarly as Eq. 2, the score for a video over each concept c_i can be computed as,

$$score(\boldsymbol{F}, c_i) = \frac{1}{N} \sum\nolimits_{F} E_{\text{net}}(\boldsymbol{F}, c_i), \tag{5}$$

and the predicted concept is the one with the highest score.

The proposed EnergyNet jointly optimizes two loss functions, energy loss and triplet loss. The conceptual example of the optimization structure is shown in Figure 3.

3.4.1 Energy Loss. For each frame $\boldsymbol{F}$ in the training videos, we denote the attention map generated by the ground truth concept $L(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}})$ as the *true map*, and the attention map generated by any other concept $L(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{fa}})$ as the *false map*. Intuitively, we want the true map to have higher energy than the false maps. Therefore, we design the energy loss to be the hinge loss between the true map and false maps as,

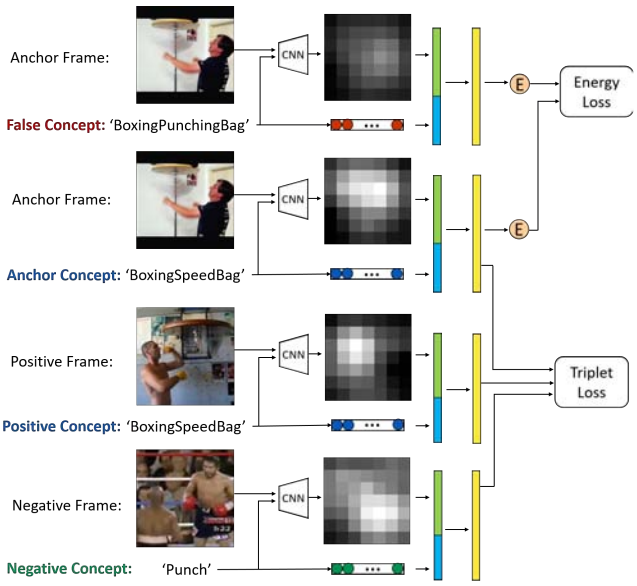


Figure 3: Illustration of Siamese EnergyNet with shared parameters. From top to bottom, the inputs are $(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{fa}})$, $(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}})$, $(\boldsymbol{F}^+, c^+)$ and $(\boldsymbol{F}^-, c^-)$. The energy loss and triplet loss are jointly optimized.

$$\begin{aligned} ELoss(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}}, c^{\boldsymbol{F}}_{\text{fa}}) = \max\{0,\ & E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{fa}}) - E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}}) + m\}, \\ & \forall\ c^{\boldsymbol{F}}_{\text{fa}} \in C,\ \ c^{\boldsymbol{F}}_{\text{fa}} \neq c^{\boldsymbol{F}}_{\text{gt}}, \end{aligned} \tag{6}$$

where m is a margin enforced between true and false pairs. Based on preliminary experiments, we set m to 1 in all of our experiments.

3.4.2 Hard Negative Mining. Generating all true-false concept pairs $(c^{\boldsymbol{F}}_{\text{gt}}, c^{\boldsymbol{F}}_{\text{fa}})$ with brute force approach would result in many pairs that easily satisfy $E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}}) - E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{fa}}) \geq m$, thus having minimum contribution to the training process and can lead to slow convergence. Therefore, it is important to select the hard false concepts $c^{\boldsymbol{F}}_{\text{fa}}$ such that the energy loss is larger for those concepts that give $\arg\min_{c^{\boldsymbol{F}}_{\text{fa}}} (E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{gt}}) - E_{\text{net}}(\boldsymbol{F}, c^{\boldsymbol{F}}_{\text{fa}}))$.

To achieve this, we mine the hard negative samples using an online approach. We first generate large mini-batches where the false concepts are chosen randomly, then we forward the mini-batches through the EnergyNet and select the top K samples with the highest energy loss and apply back-propagation on the selected K samples. To prevent early convergence to local minima in training stage, we generate smaller mini-batches at the start so that the initial K samples are *semi-hard*. After the training loss decreases below a threshold, we increase the mini-batch size to raise the probability to generate stronger negative samples. In addition to the K hard samples, we also insert a few random samples into each training batch.

3.4.3 Triplet Loss on Embedding. Inspired by [16, 30, 43], we construct a triplet loss on the embedding space to learn a more representative feature embedding $f(\boldsymbol{V}_L, \boldsymbol{V}_c)$ that capture intra-class

similarity and inter-class difference of the *true maps*. We use $f(\boldsymbol{F}, c)$ to denote $f(\boldsymbol{V}_L, \boldsymbol{V}_c)$ in this subsection for simplicity purposes.

Given two frames $\boldsymbol{F}_1$, $\boldsymbol{F}_2$, and their respective true concepts $c_{\text{gt}}^{\boldsymbol{F}_1}$, $c_{\text{gt}}^{\boldsymbol{F}_2}$, we define the distance in the embedding space $f(\boldsymbol{F}, c)$ based on the Cosine distance,

$$D\Big(f(\boldsymbol{F}_1, c_{\text{gt}}^{\boldsymbol{F}_1}), f(\boldsymbol{F}_2, c_{\text{gt}}^{\boldsymbol{F}_2})\Big) = 1 - \frac{f(\boldsymbol{F}_1, c_{\text{gt}}^{\boldsymbol{F}_1}) \cdot f(\boldsymbol{F}_2, c_{\text{gt}}^{\boldsymbol{F}_2})}{\|f(\boldsymbol{F}_1, c_{\text{gt}}^{\boldsymbol{F}_1})\| \|f(\boldsymbol{F}_2, c_{\text{gt}}^{\boldsymbol{F}_2})\|}. \quad (7)$$

We design our triplet loss on $f(\cdot)$, such that the distance between embeddings of the same class is small, and the distance between embeddings of different classes is larger. We select an anchor frame $\boldsymbol{F}$, a positive frame $\boldsymbol{F}^+$ and a negative frame $\boldsymbol{F}^-$ from three videos V, V^+, V^-, where V and V^+ are of the same class, and V^- is from a different class. We generate the *anchor map*, the *positive map* and the *negative map* as the *true maps* of the three frames,

$$\begin{aligned} L &= L(\boldsymbol{F}, c), \quad \forall\, c = c_{\text{gt}}^{\boldsymbol{F}}, \\ L^+ &= L(\boldsymbol{F}^+, c^+), \ \forall\, c^+ = c, c^+ = c_{\text{gt}}^{\boldsymbol{F}^+}, \\ L^- &= L(\boldsymbol{F}^-, c^-), \ \forall\, c^- \neq c, c^- = c_{\text{gt}}^{\boldsymbol{F}^-}. \end{aligned} \quad (8)$$

The embeddings $f(\boldsymbol{F}, c)$ for the three frames are then computed by forwarding L and c through the embedding layers.

Here, we enforce the following condition

$$D\Big(f(\boldsymbol{F}, c), f(\boldsymbol{F}^-, c^-)\Big) > D\Big(f(\boldsymbol{F}, c), f(\boldsymbol{F}^+, c^+)\Big), \quad (9)$$

by defining the triplet loss as,

$$\begin{aligned} &TLoss(\boldsymbol{F}, \boldsymbol{F}^+, \boldsymbol{F}^-, c, c^+, c^-) = \\ &\quad \max\Big\{0,\ D\Big(f(\boldsymbol{F}, c), f(\boldsymbol{F}^+, c^+)\Big) - D\Big(f(\boldsymbol{F}, c), f(\boldsymbol{F}^-, c^-)\Big) + \beta\Big\}, \end{aligned} \quad (10)$$

where β represents the margin, and is set to 0.5 in the experiment based on preliminary experiments.

3.4.4 Joint Learning. During the training phase, we want to jointly optimize the energy loss and the triplet loss. To achieve this, we create a Siamese EnergyNet (see Figure 3) comprised of 4 instances of the same EnergyNet (with shared parameters). The input to the Siamese network is $\Big\{(\boldsymbol{F}, c_{fa}^{\boldsymbol{F}}), (\boldsymbol{F}, c_{gt}^{\boldsymbol{F}}), (\boldsymbol{F}^+, c^+), (\boldsymbol{F}^-, c^-)\Big\}$. In practice, we pre-compute the attention maps $L(F, c)$ for all F and c, so that training can be much faster.

Since the input consists of both *false anchor map* $L(\boldsymbol{F}, c_{fa}^{\boldsymbol{F}})$ and *negative true map* $L(\boldsymbol{F}^-, c^-)$, we want to select the hard negatives $\Big\{(\boldsymbol{F}, c_{fa}^{\boldsymbol{F}}), (\boldsymbol{F}^-, c^-)\Big\}$ that maximize both the energy loss and the triplet loss. We mine for the hard negatives using the same online approach as Section 3.4.2. As recommended in practice by [30, 43], we do not select hard positives. We apply Stochastic Gradient Descent (SGD) to train the network. We let $d = 64$ for the embedding $f(\cdot) \in \mathbb{R}^d$. The learning rate is set as 0.0001, and we use a weight decay of 0.0005.

4 EXPERIMENTS

4.1 Datasets

We consider two labeled video datasets for two tasks: human interaction recognition and action recognition. Both datasets contain videos in unconstrained environments and have been widely used for benchmarking action/interaction recognition methods. For each video dataset, we have a corresponding Web image dataset.

4.1.1 Human Interaction Recognition. We use the TV Human Interaction (TVHI) [28] dataset for video recognition. It consists of 300 video clips compiled from 23 different TV shows. It contains four types of human interactions: Handshake, Highfive, Hug and Kiss. Each interaction has 50 videos, and the remaining 100 videos are negative examples. We use the 200 positive videos, and keep the train/test split as [28]. The dataset is generally considered challenging due to occlusion and viewpoint changes.

We collected an image dataset corresponding to the four types of interactions, namely the Human Interaction Image (HII) dataset[1]. Given an action name, we crawled Web images from Commercial Search Engines (Google, Bing and Flickr) using keyword search. Duplicate images were removed by comparing their color histogram. In total, we collected 17.5K images. Since the images are noisy, we manually filtered out the irrelevant ones that do not contain a concept. The filtered dataset contains 2410 images with at least 550 images per class. For experiments, we evaluate with both the noisy data and the filtered one.

4.1.2 Action Recognition. We use UCF101 [35], a large-scale video dataset for action recognition. It consists of 101 action classes, over 13k clips and 27 hours of video collected from YouTube. The videos are captured under various lighting conditions with camera motion, occlusion and low frame quality, making the task challenging. We use the three provided train/test split, and report the classification accuracy for evaluation.

We use BU101 [23] as the Web image dataset, which has class-to-class correspondence with the UCF101 dataset. It was collected from the Web using key phrases, and then manually filtered to remove irrelevant frames. It comprises 23.8K images with a minimum of 100 images per class.

4.2 Training CNNs with Web Images

CNNs pre-trained from ImageNet have been widely used for action recognition [11, 32]. We choose the state-of-the-art model (i.e. 101-layer ResNet [13]) and fine-tune it on the Web image dataset. To show that our method can generalize to other CNN architectures, we also evaluate VGGNet16 [33] for human interaction recognition.

To pre-process the images, we first resize the shorter side to 224 pixels while keeping the aspect ratio. Then we apply center crop to obtain the 224×224 input compatible with the CNN architecture. We augment the training images with random horizontal flipping. We apply SGD with a mini-batch size of 32. For both HII dataset and BU101 dataset, we randomly split 30% of the images for validation. The result on the validation set is shown in Table 1. The CNN models trained on HII-noisy obtain better performance than HII-filter. This finding is consistent with [18, 20], where large-scale noisily labeled images can be effective for image classification task. In addition, model trained on HII has higher accuracy than BU101. This is because it contains fewer classes and more images per class.

The Web-image trained CNN is then used to generate spatial attention maps for video frames. The size of the attention map for ResNet and VGGNet is 7×7 and 14×14, respectively.

[1] available via https://doi.org/10.5281/zenodo.832380

Table 1: Classification accuracy (%) on validation set for training CNNs with Web images.

CNN Model	HII-noisy	HII-filtered	BU101
ResNet101	95.2	94.1	88.3
VGGNet16	90.1	89.4	-

4.3 Experiment Setup

Unsupervised Domain Adaptation. In this scenario, we directly apply the Web images trained CNNs to classify the videos. We examine two types of classifiers:

- **CNN**: We use the output score from the last fc layer of the CNN to classify each frame. The class of the video is determined by voting of the frames' classes. We select the majority frames that has the same class, and apply late fusion (average) on the frame-level scores to calculate video-level score. We also experiment with averaging all frame-level scores, where the performance are slightly degrade.
- **Unsupervised Attention (UnAtt)**: Proposed method delineated in Section 3.3, where a sliding-window approach is used to compute energy from the attention maps. The energy score is computed with Eq. 2.

Supervised Domain Adaptation. In this setting, we further train the image-trained classifiers on the labeled training videos with four methods:

- **SVM**: We extract features from the penultimate layer of the image-trained CNN (pool5 for ResNet and fc7 for VGG), and train one-versus-rest linear SVM classifiers with the soft margin cost set as 1. The video class is predicted with majority voting.
- **finetune+CNN**: We fine-tune the CNNs on frames from the training videos. The training process is described in Section 4.2. Then we use the output score from the last layer of the CNN to classify given videos, where voting is applied.
- **finetune+UnAtt**: We apply the Unsupervised Attention method with the fine-tuned CNN.
- **finetune+EnergyNet**: We train an EnergyNet (Section 3.4) using spatial attention maps generated by the fine-tuned CNN, and calculate the score with Eq. 5.

4.4 Human Interaction Recognition Task

We use the CNNs pre-trained on Human Interaction Image (HII) dataset, and evaluated the methods on the test set of TVHI dataset, which consists a total of 100 videos with 25 videos per class. We report the mean average precision (mAP) for evaluation purpose. For supervised methods, we train on all frames from the 100 training videos of TVHI. To test the scalability of our methods, we investigate with using both noisy and filtered images to pre-train the CNNs.

Results. Table 2 shows the results for both unsupervised and supervised domain adaptation, The proposed methods (UnAtt and EnergyNet) outperforms corresponding baselines using either ResNet101 or VGG16. The proposed unsupervised method (UnAtt) can achieve better performance compared with the supervised baseline method (i.e. finetune+CNN with ResNet101), demonstrating its efficacy to transfer knowledge across domains.

Table 2: Mean Average Precision (mAP) (%) for both Unsupervised (shaded in blue) and Supervised Domain Adaptation on TVHI dataset. CNNs pre-trained on both the filtered images and noisy images are evaluated.

CNN Model	Method	HII-filtered	HII-noisy
ResNet101	CNN	89.7	91.6
	UnAtt	94.3	**96.0**
	finetune+CNN	92.6	92.9
	finetune+UnAtt	94.7	96.3
	finetune+EnergyNet	96.8	**98.7**
VGG16	CNN	86.7	85.3
	UnAtt	**89.0**	86.9
	finetune+CNN	89.5	88.9
	finetune+UnAtt	90.5	89.4
	finetune+EnergyNet	**91.7**	90.7

Table 3: Comparison with state-of-the-art methods on TVHI dataset.

Methods	mAP
Patron *et al.* [28]	42.4
Hoai *et al.* [15]	71.1
Wang *et al.* [41]	78.2
NoImage+ResNet	44.9
ResNet+UnAtt	96.0
ResNet+finetune+EnergyNet	**98.7**

By fine-tuning on the video frames, both the baseline methods and the proposed method achieve performance improvement. For ResNet101, using finetune+EnergyNet leads to +5.8 in mAP when compared with finetune+CNN, and +2.4 against finetune+UnAtt. This shows that the EnergyNet can further adapt to the video domain by training on the attention maps.

A surprising result is that ResNet101 can benefit from training on large amount of noisy images as compared with using filtered images. This demonstrates the robustness of ResNet in learning good feature representations from noisy data. VGG16 only degrades by a small margin using noisy images instead of filtered ones. This indicates that our method can directly make use of weakly-labeled Web images and does not require manual labeling.

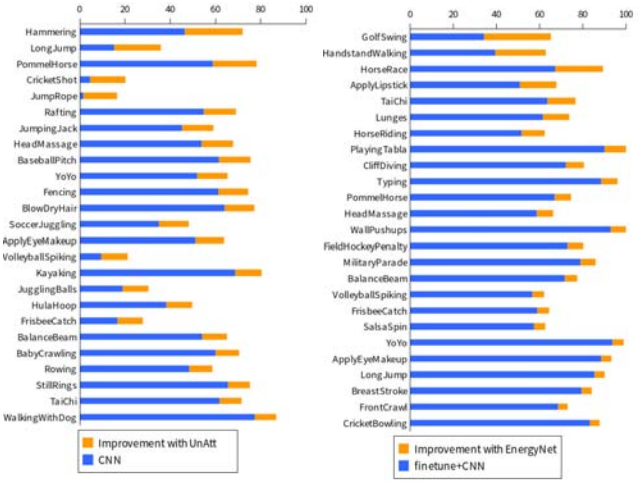
Comparison with state-of-the-art. We compare our method with the state-of-the-art approaches for interaction recognition. Since all previous approaches do not use deep learning, we create another baseline NoImage+ResNet, where we directly fine-tune a ResNet101 model pre-trained from ImageNet on the video frames (without Web images). As shown in Table 3, we can see that using Web images as auxiliary training data significantly improves the performance, especially when the amount of the available training video is low.

4.5 Action Recognition Task

We use the CNN model pre-trained on BU101 dataset, and evaluate the methods on the test set of UCF101 dataset. We report the classification accuracy averaged over the three test splits. To reduce

Table 4: Mean classification accuracy of Unsupervised Domain Adaptation on UCF101 (averaged over three test splits).

Method	Top-1	Top-3
CNN	62.5	78.5
UnAtt	**66.4**	**82.4**



(a) Unsupervised Domain Adaption **(b) Supervised Domain Adaption**

Figure 4: The 25 action classes with the largest accuracy improvement on UCF101 dataset. Supervised domain adaptation is performed with 20% training videos.

redundant frames and training time, we sample one frame for every five from the videos for both training and test. Based on the experimental results on TVHI, we choose ResNet101 as the CNN model for this task.

Results. Table 4 shows the result for unsupervised domain adaptation, where the efficacy of the proposed method (UnAtt) is proved. Figure 4 shows the 25 action classes with the largest accuracy improvement, for both unsupervised and supervised scenario.

For supervised domain adaptation, we study how the amount of training data in the target domain influence the performance of the proposed method. Each training split in the UCF101 dataset comprises around 95 clips per action class. We randomly sample 5%, 10%, 20%, 33%, 50% and 100% videos from the training set, and report the performance for each sampled set. The experiments are separately done on the three train/test splits, and the result is averaged across three test splits and reported in Figure 5.

There are several findings from the results. First, fine-tuning the CNN on the training videos significantly improves the performance, and the improvement increases as more videos are used. Second, the proposed method (EnergyNet) outperforms the baselines using different number of training videos. The improvement is largest (+3.1%) with 20% of videos used. When using all training videos, the improvement is +0.7%. This suggests that the proposed method is most effective when the amount of training data in the target domain is limited, which is the general scenario for domain adaptation problems. Furthermore, the proposed EnergyNet can achieve better performance using fewer videos compared with the baseline CNN method using more videos. (EnergyNet-10% outperforms CNN-20%, EnergyNet-50% outperforms CNN-100%)

Table 5: Comparison with state-of-the-art methods on UCF101. * refers to methods that use Web data.

	Method	Accuracy (%)
W/O Motion	Spatial stream network [32]	73.0
	LRCN [8]	71.1
	Karpathy *et al.* [19]	65.4
	* Webly-supervised [10]	69.3
	* Ma *et al.* spatial [23]	83.5
With Motion	Two-stream network [32]	88.0
	IDT+FV [40]	87.9
	C3D [38]	82.3
	TDDs [42]	90.3
	TDDs+IDT-FV [42]	91.5
	* Ma *et al.* spatial+IDT-FV [23]	91.1
Ours	* UnAtt (no training video)	66.4
	* EnergyNet (20% training video)	85.0
	* EnergyNet (all training video)	88.0

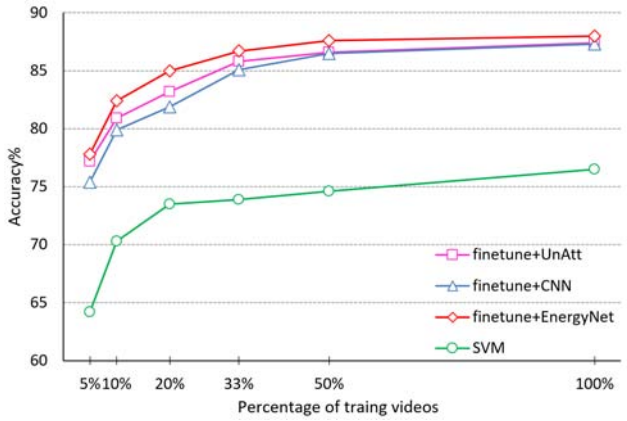


Figure 5: Mean classification accuracy of Supervised Domain Adaptation with different number of training videos used.

Comparison with state-of-the-art. In Table 5 we compare our method with state-of-the-art results. We directly quote the results from published papers. Among those approaches, two-stream network [32], IDT+FV [40], C3D [38], TDDs [42] and Ma *et al.* spatial+IDT-FV [23] incorporate motion features from videos. Other approaches only utilize appearance features from static frames.

Ma *et al.* spatial [23] and Webly-supervised [10] are the two methods that most relate to ours. Ma *et al.* spatial [23] use images from the same BU101 dataset and all videos from UCF101 to train a shared CNN. Webly-supervised [10] use images and videos from the Web to train a LSTM classifier.

With only 20% of the training videos used, our method outperforms previous methods that only use spatial features, which again shows the efficacy of the proposed method when training data in the video domain is limited. Using all training videos, our method achieves comparable performance with state-of-the-art methods that utilize both spatial and motion features.

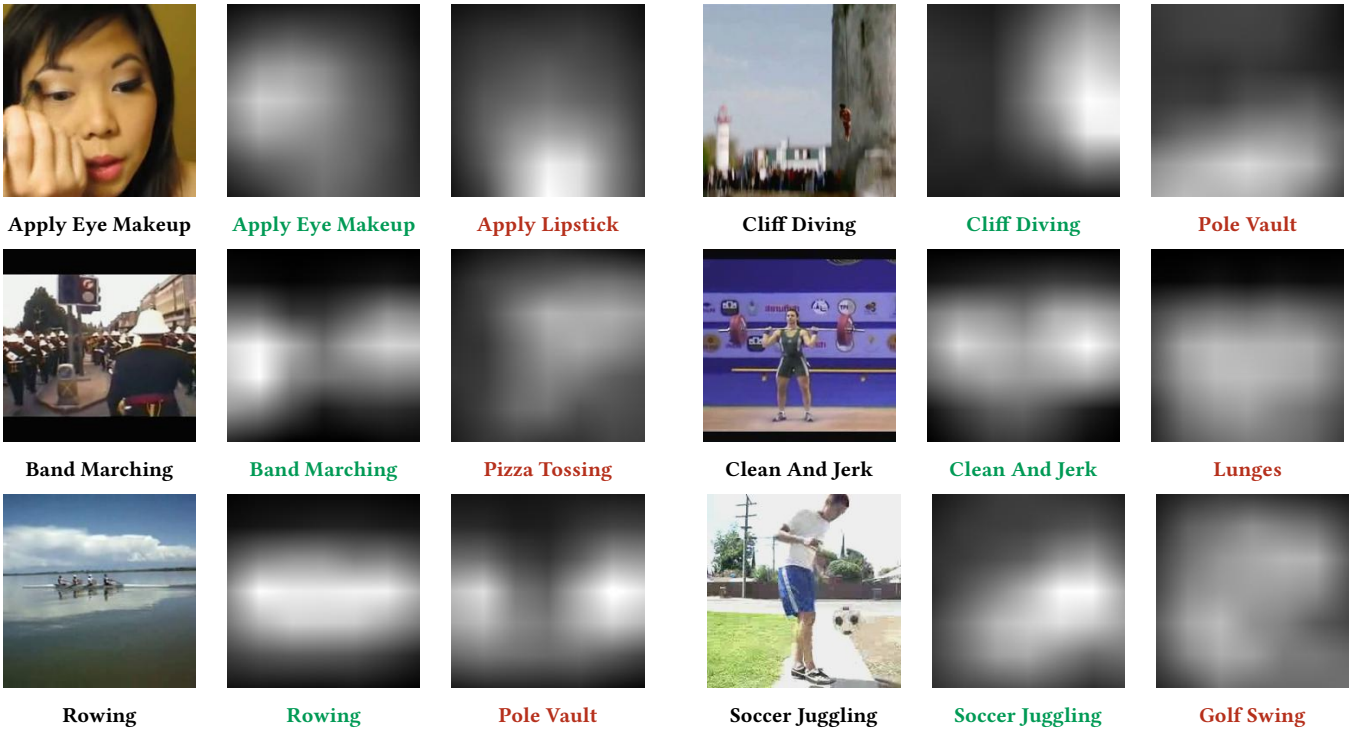


Figure 6: Example frames from UCF101 and their corresponding spatial attention maps (resized to image size) for two concepts generated by the fine-tuned CNN. The green concept is the one correctly predicted by the proposed EnergyNet, while the red concept is the one wrongly predicted by the baseline CNN method. The EnergyNet correctly assigns a higher energy for the ground truth concept.

4.6 Visualization of Examples

To provide more intuitions of how transferring attention contribute to video recognition, we show in Figure 6 some example frames from UCF101 where the baseline CNN method predicts wrongly and the proposed EnergyNet predicts correctly. We show the spatial attention maps generated by the fine-tuned CNN for both the ground truth concept and the concept wrongly predicted by the baseline. For each example, the EnergyNet correctly assigns a higher energy to the spatial attention map corresponding to the ground truth concept.

5 CONCLUSION

In this work, we propose a new attention-based method to adapt a Web image trained CNN to video recognition. The proposed method utilizes class-discriminative spatial attention map, which is a low-dimensional feature space that incorporates the convolutional features with class information. We study unsupervised and supervised domain adaptation, and construct a Siamese EnergyNet that jointly optimizes two loss functions to learn class-specific energy functions for the attention maps. We conduct experiments on human interaction recognition and action recognition, and show the efficacy of the proposed method to adapt the domain shift problem, especially when the amount of training data in the target domain is limited.

Since the proposed method focuses on static visual knowledge transfer from Web images to videos, we do not consider motion features. For future work, we intend to incorporate motion features into our framework, so that the performance can be further improved. In addition, we believe that attention-based cross-domain knowledge transfer has other potentials beyond video recognition. We aim to explore using Web images for action localization in videos, both spatially and temporally.

ACKNOWLEDGMENT

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centre in Singapore Funding Initiative.

REFERENCES

[1] Mahsa Baktashmotlagh, Mehrtash Tafazzoli Harandi, Brian C. Lovell, and Mathieu Salzmann. 2013. Unsupervised Domain Adaptation by Domain Invariant Projection. In *ICCV*. 769–776.
[2] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain Separation Networks. In *NIPS*. 343–351.
[3] Jiawei Chen, Yin Cui, Guangnan Ye, Dong Liu, and Shih-Fu Chang. 2014. Event-Driven Semantic Concept Discovery by Exploiting Weakly Tagged Internet Images. In *ICMR*. 1.
[4] Minmin Chen, Kilian Q. Weinberger, and John Blitzer. 2011. Co-Training for Domain Adaptation. In *NIPS*. 2456–2464.
[5] Xinlei Chen and Abhinav Gupta. 2015. Webly Supervised Learning of Convolutional Networks. In *ICCV*. 1431–1439.
[6] Guangchun Cheng, Yiwen Wan, Abdullah N. Saudagar, Kamesh Namuduri, and Bill P. Buckles. 2015. Advances in Human Action Recognition: A Survey. *CoRR* (2015).
[7] Santosh Kumar Divvala, Ali Farhadi, and Carlos Guestrin. 2014. Learning Everything about Anything: Webly-Supervised Visual Concept Learning. In *CVPR*. 3270–3277.
[8] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. 2015. Long-term recurrent

convolutional networks for visual recognition and description. In *CVPR*. 2625–2634.

[9] Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. 2013. Unsupervised Visual Domain Adaptation Using Subspace Alignment. In *ICCV*. 2960–2967.

[10] Chuang Gan, Chen Sun, Lixin Duan, and Boqing Gong. 2016. Webly-Supervised Video Recognition by Mutually Voting for Relevant Web Images and Web Video Frames. In *ECCV*. 849–866.

[11] Chuang Gan, Ting Yao, Kuiyuan Yang, Yi Yang, and Tao Mei. 2016. You Lead, We Exceed: Labor-Free Video Concept Learning by Jointly Exploiting Web Videos and Images. In *CVPR*. 923–932.

[12] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*. 2066–2073.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR*. 770–778.

[14] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. ActivityNet: A large-scale video benchmark for human activity understanding. In *CVPR*. 961–970.

[15] Minh Hoai and Andrew Zisserman. 2014. Improving Human Action Recognition Using Score Distribution and Ranking. In *ACCV*. 3–20.

[16] Elad Hoffer and Nir Ailon. 2015. Deep Metric Learning Using Triplet Network. In *SIMBAD workshop*. 84–92.

[17] Mihir Jain, Jan C. van Gemert, Thomas Mensink, and Cees G. M. Snoek. 2015. Objects2action: Classifying and Localizing Actions without Any Video Example. In *ICCV*. 4588–4596.

[18] Armand Joulin, Laurens van der Maaten, Allan Jabri, and Nicolas Vasilache. 2016. Learning Visual Features from Large Weakly Supervised Data. In *ECCV*. 67–84.

[19] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Fei-Fei Li. 2014. Large-Scale Video Classification with Convolutional Neural Networks. In *CVPR*. 1725–1732.

[20] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Li Fei-Fei. 2016. The Unreasonable Effectiveness of Noisy Data for Fine-Grained Recognition. In *ECCV*. 301–320.

[21] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. 2015. Learning Transferable Features with Deep Adaptation Networks. In *ICML*. 97–105.

[22] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S. Yu. 2014. Transfer Joint Matching for Unsupervised Domain Adaptation. In *CVPR*. 1410–1417.

[23] Shugao Ma, Sarah Adel Bargal, Jianming Zhang, Leonid Sigal, and Stan Sclaroff. 2017. Do Less and Achieve More: Training CNNs for Action Recognition Utilizing Action Images from the Web. *Pattern Recognition* (2017). http://dx.doi.org/10.1016/j.patcog.2017.01.027

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*. 3111–3119.

[25] Joe Yue-Hei Ng, Matthew J. Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *CVPR*. 4694–4702.

[26] Jie Ni, Qiang Qiu, and Rama Chellappa. 2013. Subspace Interpolation via Dictionary Learning for Unsupervised Domain Adaptation. In *CVPR*. 692–699.

[27] Dan Oneata, Jakob J. Verbeek, and Cordelia Schmid. 2013. Action and Event Recognition with Fisher Vectors on a Compact Feature Set. In *ICCV*. 1817–1824.

[28] Alonso Patron-Perez, Marcin Marszalek, Ian D. Reid, and Andrew Zisserman. 2012. Structured Learning of Human Interactions in TV Shows. *TPAMI* 34, 12 (2012), 2441–2453.

[29] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. 2010. Adapting Visual Category Models to New Domains. In *ECCV*. 213–226.

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *CVPR*. 815–823.

[31] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. 2016. Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization. *CoRR* (2016).

[32] Karen Simonyan and Andrew Zisserman. 2014. Two-Stream Convolutional Networks for Action Recognition in Videos. In *NIPS*. 568–576.

[33] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*.

[34] Bharat Singh, Xintong Han, Zhe Wu, Vlad I. Morariu, and Larry S. Davis. 2015. Selecting Relevant Web Trained Concepts for Automated Event Retrieval. In *CVPR*. 4561–4569.

[35] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. *CoRR* (2012).

[36] Sainbayar Sukhbaatar and Rob Fergus. 2014. Learning from Noisy Labels with Deep Neural Networks. *CoRR* (2014).

[37] Chen Sun, Sanketh Shetty, Rahul Sukthankar, and Ram Nevatia. 2015. Temporal Localization of Fine-Grained Actions in Videos by Domain Transfer from Web Images. In *ACMMM*. 371–380.

[38] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. C3D: Generic Features for Video Analysis. In *ICCV*.

[39] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. 2015. Simultaneous Deep Transfer Across Domains and Tasks. In *ICCV*.

[40] Heng Wang and Cordelia Schmid. 2013. Action Recognition with Improved Trajectories. In *ICCV*. 3551–3558.

[41] Hanli Wang, Yun Yi, and Jun Wu. 2015. Human Action Recognition With Trajectory Based Covariance Descriptor In Unconstrained Videos. In *ACMMM*. 1175–1178.

[42] Limin Wang, Yu Qiao, and Xiaoou Tang. 2015. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*. 4305–4314.

[43] Xiaolong Wang and Abhinav Gupta. 2015. Unsupervised Learning of Visual Representations Using Videos. In *ICCV*. 2794–2802.

[44] Shuang Wu, Sravanthi Bondugula, Florian Luisier, Xiaodan Zhuang, and Pradeep Natarajan. 2014. Zero-Shot Event Detection Using Multi-modal Fusion of Weakly Supervised Concepts. In *CVPR*. 2665–2672.

[45] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. 2015. Learning from massive noisy labeled data for image classification. In *CVPR*. 2691–2699.

[46] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks?. In *NIPS*. 3320–3328.

[47] Sergey Zagoruyko and Nikos Komodakis. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *ICLR*.

[48] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning Deep Features for Discriminative Localization. In *CVPR*. 2921–2929.