Harnessing Object and Scene Semantics for Large-Scale Video Understanding

Zuxuan Wu[†], Yanwei Fu[§], Yu-Gang Jiang[†], Leonid Sigal[§]
[†]Shanghai Key Lab of Intel. Info. Processing, School of Computer Science, Fudan University
[§]Disney Research
{zxwu, ygj}@fudan.edu.cn, {yanwei.fu, lsigal}@disneyresearch.com

Abstract

Large-scale action recognition and video categorization are important problems in computer vision. To address these problems, we propose a novel object- and scene-based semantic fusion network and representation. Our semantic fusion network combines three streams of information using a three-layer neural network: (i) frame-based low-level CNN features, (ii) object features from a state-of-the-art large-scale CNN object-detector trained to recognize 20K classes, and (iii) scene features from a state-of-the-art CNN scene-detector trained to recognize 205 *scenes. The trained network achieves improvements in supervised activity and video categorization in two complex large-scale datasets - ActivityNet and FCVID, respectively. Further, by examining and back propagating information through the fusion network, semantic relationships (correlations) between video classes and objects/scenes can be discovered. These video class-object/video class-scene relationships can in turn be used as semantic representation for the video classes themselves. We illustrate effectiveness of this semantic representation through experiments on zero-shot action/video classification and clustering.*

1. Introduction

The ubiquitous availability and use of devices that can capture and share videos on social platforms is astounding; an estimated $1-5$ *hours* of videos are being uploaded to YouTube per *second* by the users. Such growth in visual media requires robust and scalable approaches for video indexing, search and summarization. However, general video understanding in unconstrained and, often, user-generated videos is extremely challenging. Videos vary greatly in terms of both the semantic content (*e.g.*, concert) and appearance of that content (*e.g.*, as observed from audience or backstage). The same or similar content can be recorded from a variety of views (*e.g.*, front-row or obstructed-view seat in the back), under a breadth of viewing conditions (*e.g.*, natural or stage lighting), and can be of nearly arbitrary length (*e.g.*, an hour long professional recoding, egocentric snippet, or iPhone highlight). Hence appearance variability within a given topic is often greater than variability across topics making recognition difficult.

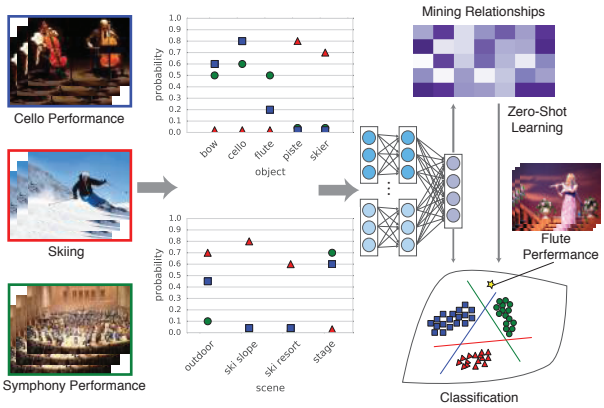

Figure 1. Illustration of the proposed Object-Scene semantic Fusion (OSF) network and its application on several tasks.

In computer vision, video understanding is often addressed in the form of action/activity recognition or localization (this limits the scope to human-centric events and video content); generic video categorization [12] has been much less thoroughly explored. In both domains the focus, over the years, has been largely on learning video-based representations (*e.g.* HoG, HoF or MBH [35]), combined with supervised (or weakly supervised [5, 25, 29]) classifiers for recognition/categorization. Recent successes in deep learning, particularly Convolutional Neural Networks (CNNs), opened opportunities for learning discriminative hierarchical frame-based [13] or spatio-temporal representations [31, 34] jointly with the classifiers in an end-to-end fashion. Recent CNN approaches have shown remarkable improvements in performance on datasets where large amount of labeled data is available [31]. However, ability to learn from limited labeled data or scale such approaches up from at most few hundred classes to thousands, if not tens of thousands, of classes, presents significant challenges for the community; the latter, in particular, due to insurmountable

efforts to scale up annotation to tens of millions of videos and practical inability to find and label rare events.

Semantic representations provide one way of bridging the current challenges. Semantic representations, in the form of attributes [15] or objects [16], are becoming increasingly popular in object categorization [42] and scene understanding [16]. Such representations typically improve generalization by representing classes that may potentially have only few, or even no, training instances in terms of semantic entities that are much easier to train classifiers for. One challenge is that relationship between semantic entities and classes often needs to be defined by hand, which is costly and non-trivial (*e.g.*, there may not be a general agreement on whether *horse* is *furry*[1]); techniques have been proposed to solve for correlations algorithmically (*e.g.*, sparse-coding [8]) at the cost of model expressiveness[2]. In video categorization the use of such semantic representations has been much more limited, with few recent works focusing on attribute-based event recognition [17], joint actor-action based reasoning [37], object-action relationships [10], and at a very limited scale object + scene features to improve action classification [9]. However, these works focus largely on the improved video classification performance, with semantic entities as black-box features, and not finding robust semantic decompositions of actions for other tasks (*e.g.*, zero-shot prediction and clustering).

To address these issues we introduce a novel Object-Scene semantic Fusion (OSF) network for large-scale video categorization. OSF combines three streams of information using a three-layer fusion neural network: (i) frame-based low-level CNN features, (ii) object features from a state-of-the-art large-scale CNN object-detector with 20K classes and (iii) scene features from a state-of-the-art CNN scene-detector trained to recognize 205 scenes. This framework (see Figure 1) has a number of appealing properties. First, it is defined as an end-to-end network and hence joint training of all streams and the fusion layers is possible. Second, complex non-linear relationships between semantic entities (objects and scenes) and the video class labels can be learned and need not be specified by hand. Third, by examining and back propagating information through the fusion layers, semantic relationships between video classes and objects/scenes can be discovered. This Object and Scene semantic Representation (OSR), in the form of video class-object/video class-scene relationships, can be used for a variety of tasks, including zero-shot recognition of novel categories and measuring similarity (clustering). In addition to appealing conceptual properties OSF/OSR improves both supervised and zero-shot classification on two challenging and large-scale datasets for activity (ActivityNet [7]) and generic video categorization (FCVID [12]).

[1] See http://www.1freewallpapers.com/furry-black-white-horse.

[2] Linear model needs to be assumed.

2. Related Works

The fields of action recognition and video classification are too broad to review completely; we focus only on the most relevant literature.

Traditional action/video classification: There is a variety of works in the field of video classification, with most focusing on developing more discriminative features and better classifiers [11]. A typical video classification pipeline in recent literatures usually relies on the state-of-the-art dense trajectory features [35], which are local descriptors (*e.g.*, HoG, HoF and MBH) computed around densely extracted frame patch trajectories. Bag-of-words and more advanced feature encoding strategies such as Fisher Vector [28] have been adopted to quantize the local descriptors for classification (normally by an SVM classifier).

Deep models (CNNs/LSTMs): More recently, driven by the great success of Convolutional Neural Networks (CNN) on image analysis tasks [6, 31, 32], a few works attempted to leverage CNN models to learn feature representations for video classification. For instance, Karparthy *et al*. extended CNN models into the time domain by stacking frames [13]. To better explore the motion information, Simonyan *et al*. [31] recently proposed to train two CNNs on still images and optical flow fields separately to capture appearance and motion information. Final predictions were generated by averaging scores from the two corresponding CNN streams. In order to model the temporal dynamics in videos, there are also a few works utilizing recurrent networks like the long-short term memory (LSTM) for recognition [3, 21, 36]. All these works, however, focused on extracting and encoding videos directly, using neural networks, and result in the representations that are neither semantic nor inherently interpretable. None of these works investigate or utilize object and/or scene semantics.

CNN model visualization: Our work is also partly inspired by the techniques for visualization and understanding of CNN networks. Zeiler *et al*. [39] proposed Deconvolutional Network (DeconvNet) to approximately reconstruct the input of each layer from the corresponding output. More advanced recent visualization techniques are discussed in [30, 40]; Deep Dream[3] has also been influential. We use visualization-inspired technique for discovering object-video class and scene-video class relationships from the proposed OSF network.

Semantic (Object/Scene) Context: Complex video semantics like activities and events have been shown to strongly correlate with their involved objects and scenes, which provide strong semantic context prior for video classification [4, 18]. For example, in Figure 1, *symphony per-*

[3] http://googleresearch.blogspot.fr/2015/06/inceptionism-going-deeper-into-neural.html

formance often takes place in a *concert hall*, whilst *skiing* commonly happens *outdoors*. Prest *et al*. [25] used a weakly supervised method to model human actions as interactions between humans and objects. Ikizler-Cinbis *et al*. [9] proposed an approach for understanding human activities by integrating multiple feature clues from objects, scenes and people. Li *et al*. [16] developed a large number of pre-trained generic object detectors named ObjectBank to generate high-level visual representations. ActionBank was proposed in [27] as a semantic feature for video classification. These semantic representations have largely been explored on smaller datasets and almost exclusively as context for improving supervised classification.

Perhaps the closest to ours is a more recent work of [10], where relations between 15,000 object categories and high-level video categories like complex events are systematically studied. The authors conclude that objects are important for action and event recognition, and object-action/event relations are generic. However, the relations were discovered using relatively simple generative learning method (sum of averaged object response vectors per action class), and hence the obtained relations tend to be noisy. In contrast, we learn the relations using a more advanced and robust discriminative neural network classifier with a special architecture tailored for the task. Discriminative nature of the learning allows us to focus on relationships that tend to improve classification performance. In addition, we also look at importance of scenes.

3. Approach

Our goal is to learn a semantic model for video classification that enables effective supervised classification and, at the same time, allows us to discover semantic representations of our classes that are useful for other (unsupervised) tasks, like zero-shot learning or clustering. To this end, we first introduce an Object-Scene semantic Fusion (OSF) network (Sec. 3.1). OSF consists of a three-layer neural network that fuses information from three CNN streams: (i) a generic image feature stream, designed to capture low-level features of video frames, such as texture and color, (ii) an object stream that captures confidences among 20K object categories it is pre-trained to detect, and (iii) a scene stream, that similarly captures confidences among 205 scene categories it is pre-trained to detect.

Given the learned OSF model, we analyze it to *discover*, in Sec. 3.2, Object and Scene semantic Representation (OSR) which captures relationships (correlations) between video class labels and semantic entities (objects and scenes). This procedure is not as trivial as it may sound, as unlike with linear models, discovering such relations in our non-linear fusion architecture requires optimization. Finally, we show how discovered OSR can be utilized for zero-shot video classification (Sec. 3.3).

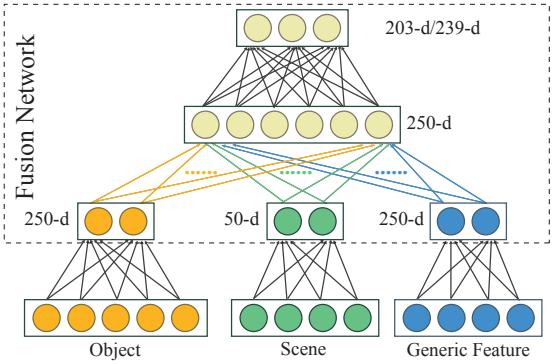

Figure 2. Object-Scene semantic Fusion (OSF) network.

Notation: Suppose we have a large-scale video dataset, $\mathcal{D}$, where each video V_i is associated with a class label $z_i \in Z_{Tr}$ from the training label set Z_{Tr}:

$$\mathcal{D} = \{(X_i, z_i)\}_{i=1,\cdots,n_{Tr}},$$

where n_{Tr} is the total number of training videos; each video is represented by a set of frames: $V_i = \{\mathbf{f}_{i,1}, \cdots, \mathbf{f}_{i,n_i}\}$, where n_i is the total number of frames in video Vi.

3.1. Object-Scene Semantic Fusion (OSF) Network

As stated above, OSF network has four components: object stream, scene stream, generic feature stream, and a three-layer neural network that fuses information from the three streams. The overall structure of the network is illustrated in Figure 2.

Object stream ($\mathcal{O}-stream$) extracts the object-related information for video classification. We use a VGG-19 CNN model, proposed in [32], which consists of 16 convolutional and 3 fully connected layers. VGG-19 for this stream is pre-trained by using all ImageNet 20,574 object classes [2]. We note that since humans can distinguish 30,000 basic object categories [1], our 20,574-class model is a good proxy for generic object cognition (covering roughly $2/3$ of human distinguishable objects). We use output of the last fully connected layer ($FC8$) as the input for the fusion network; in other words, for the j-th frame of video i, $\mathbf{f}_{i,j}$, this stream outputs $\mathbf{f}_{i,j} \mapsto \mathbf{x}^{\mathcal{O}}_{i,j} \in \mathbb{R}^{20574}$.

Scene stream ($\mathcal{S}-stream$) extracts the scene-related information to help video classification. Here we use VGG-16 CNN model provided by [41]. VGG-16 consists of 13 convolutional and 3 fully connected layers. The model is pre-trained using Places205 dataset [41] (205 scene classes and 2.5 million images). We again use the output of the last fully connected layer ($FC8$) as the input for the fusion network; in other words, for the j-th frame of video i, $\mathbf{f}_{i,j}$, this stream outputs $\mathbf{f}_{i,j} \mapsto \mathbf{x}^{\mathcal{S}}_{i,j} \in \mathbb{R}^{205}$.

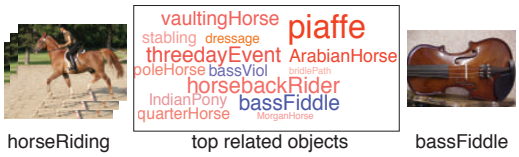

Figure 3. Objects with highest responses for the class *horse riding*; size indicates importance. See text for discussion.

Generic feature stream ($\mathcal{F}-$stream) extracts more generic visual information that maybe directly relevant for video class prediction (*e.g.*, texture, color) that the other two streams may overlook by suppressing object/scene irrelevant feature information. Once again we use a VGG-19 CNN model pre-trained on all of ImageNet. However, for this stream we take features of the first (not last) fully connected layer as input to the fusion network. In other words, for the j-th frame of video i, $\mathbf{f}_{i,j}$, this stream outputs $\mathbf{f}_{i,j} \mapsto \mathbf{x}^{\mathcal{F}}_{i,j} \in \mathbb{R}^{4096}$. Note that this stream could easily adopt more advanced networks (*e.g.*, motion network) to better account for the temporal structures in videos.

Fusion network is composed of three layers neural network (two hidden layers and one output layer) designed to fuse $\mathcal{O}-$stream, $\mathcal{S}-$stream and $\mathcal{F}-$stream features defined above. Specifically, video-level feature representation is first generated by averaging the frames of each video for each stream. Since we do not fine-tune the network end-to-end, this is done explicitly; but can be equivalently implemented by a pooling operation inserted between each stream and the first layer of the fusion network. For example, video V_i we represent as $\bar{\mathbf{x}}^{\mathcal{O}}_i = \sum_{k=1}^{n_i} \mathbf{x}^{\mathcal{O}}_{i,k}$, $\bar{\mathbf{x}}^{\mathcal{S}}_i = \sum_{k=1}^{n_i} \mathbf{x}^{\mathcal{S}}_{i,k}$, $\bar{\mathbf{x}}^{\mathcal{F}}_i = \sum_{k=1}^{n_i} \mathbf{x}^{\mathcal{F}}_{i,k}$.

The averaged representations $\bar{\mathbf{x}}^{\mathcal{O}}_i$, $\bar{\mathbf{x}}^{\mathcal{S}}_i$, $\bar{\mathbf{x}}^{\mathcal{F}}_i$ are fed into a first hidden layer of the fusion network, consisting of 250, 50, and 250 neurons respectively for each stream (550 neurons total). We use fewer neurons for $\mathcal{S}-$stream because it has fewer dimensions. The output of the first hidden layer is fused by the second (250 neurons) fully connected layer across all streams. Then a softmax classifier layer is added for video classification. Note that we normalize the ground truth labels with L_1 norm when a sample has multiple labels. We denote $f(\cdot)$ as the non-linear function approximated by semantic fusion network and $f_z(\bar{\mathbf{x}}_i)$ as the score of video instance V_i belong to the class z. The most likely class label $\hat{z}_i$ of V_i is hence inferred as:

$$\hat{z}_i = \text{argmax}_{z \in Z_{Tr}} f_z(\bar{\mathbf{x}}_i). \quad (1)$$

3.2. Object and Scene Semantic Representation

Once the OSF network is trained, the correlation between objects/scenes and video classes can be mined using the "visualization" of the network [30]. The goal is to find from object and scene streams a pseudo video representation that maximizes the neuron activity of each of the classes. Such object/scene representation identifies the most discriminative objects for the specified video class[4].

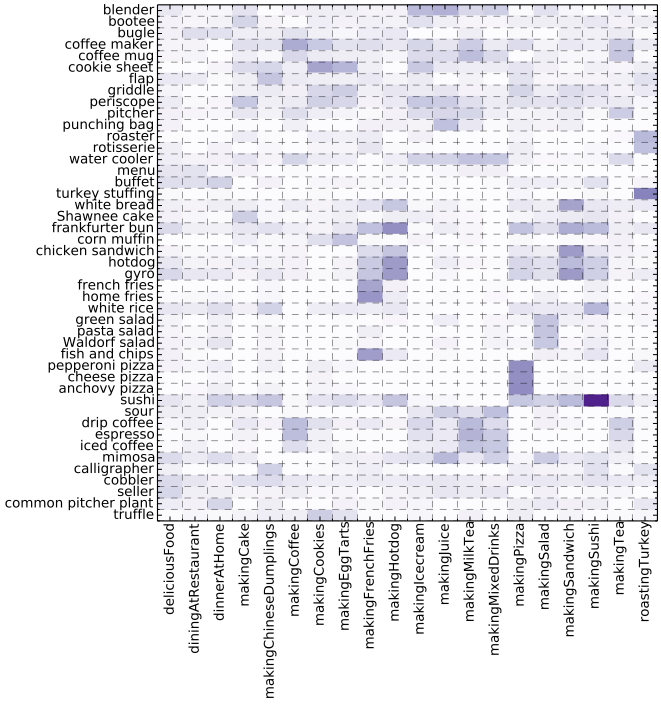

Figure 4. The visualization of a part from the $\Pi^{\mathcal{O}}$ learned on FCVID, where each entry denotes the response score between each object and video class pair.

More formally, let $f_z(\bar{\mathbf{x}}_i)$ be the score of the class z computed by the fusion network for video V_i. We need to find an L_2-regularized feature representation, such that the score $f_z(\bar{\mathbf{x}}_i)$ is maximized with respect to object or scene:

$$\hat{\mathbf{x}}^k_z = \arg\max_{\bar{\mathbf{x}}^k_i} f_z(\bar{\mathbf{x}}_i) - \lambda \left\| \bar{\mathbf{x}}^k_i \right\|_2 \quad (2)$$

where λ is the regularization parameter and $k \in \{\mathcal{O}, \mathcal{S}\}$. The locally-optimal representation $\bar{\mathbf{x}}^k_i$ can be obtained by back-propagation with randomly initialized $\bar{\mathbf{x}}_i$. We set $\lambda = 1e{-}3$, learning rate to 0.8 and also fix the maximum iterations to 1000 to maximize the classification score of each class, aiming to find the representative objects/scenes associated with those classes. This way we can obtain object-video class / scene-video class semantic representation (OSR) matrices:

$$\Pi^k = \left[\left(\hat{\mathbf{x}}^k_z \right)^T \right]_z; \quad k \in \{\mathcal{O}, \mathcal{S}\}. \quad (3)$$

Interpretability of OSR: Given the object scene representation matrix, we try to answer the following question: *what*

[4]Note that the method in Eq.(2) is also applicable to generic feature stream which, however, has no semantic meaning and hence is less useful as confirmed in Sec 4.4 (Table 3).

are intrinsic semantic properties of a video concept? Figure 3 presents a depiction of the objects with the highest value for the category *horse riding*, including *Piaffe*, *Arabian Horse* and *horseback Rider*. It is appealing that these objects are semantically meaningful for *horse riding* and can help discriminate the class. Interestingly, we also find *horse riding* is related to *bassFiddle*, which is largely due to similar visual (*e.g.*, shape and color) appearance.

To further validate the effectiveness of the learned representation, we illustrate part of the $\Pi^{\mathcal{O}}$ matrix in Figure 4, where each entry indicates the response score between the object and the video class. Objects with high scores tend to be semantically meaningful for corresponding classes.

3.3. Zero-shot Learning via OSR Correlations

One of the perhaps most interesting applications of the discovered OSR correlation matrices $\Pi = \left[\Pi^{\mathcal{O}}, \Pi^{\mathcal{S}}\right]$, computed using Eq.(2), is zero-shot learning. Different from the the supervised learning, this task is defined as transferring knowledge from known (source) classes to a disjoint set of unknown (target) classes in order to improve recognition. Specifically, zero-shot learning attempts to do this without having any labeled instances of the unknown classes available. We denote Z_{Te} as the label set of testing instances, under assumption that $Z_{Tr} \cap Z_{Te} = \varnothing$.

One of the key assumptions we make for zero-shot recognition is that object-scene semantic space is a good proxy for measuring semantic distance of video content. In other words, video samples that contain similar objects and scenes are likely to belong to the same video class. In this sense, if we are able to represent a video sample by a vector containing probability (or confidences) of it containing objects and scene we can do classification by a simple nearest neighbor approach; comparing this object-scene vector representation to video class prototypes represented in the same object-scene semantic space. Matrix Π implicitly defines prototypes for all training video classes. For testing zero-shot classes, however, we have no data and hence cannot learn prototypes directly, but can synthesize them using some knowledge of similarity between zero-shot and training video categories.

Testing-class Prototype: The prototype of testing classes can be defined using the OSR matrix,

$$\Pi_{\tilde{z}} = \sum_{z=1}^{|Z_{Tr}|} sim\left(\tilde{z},\, z\right) \cdot \Pi_z \tag{4}$$

where $z \in Z_{Tr}$ and $\tilde{z} \in Z_{Te}$ (remember $Z_{Tr} \bigcap Z_{Te} = \varnothing$); $sim\left(\tilde{z},\, z\right)$ is the semantic similarity between the testing class $\tilde{z}$ and the training class z; Π_z is the the column of Π corresponding to class z. Similarity function can be defined manually [38], using WordNet [26], or by semantic word vectors [19, 23]. Here we use word2vec [19] to define the similarity function $sim\left(\tilde{z},\, z\right)$ between the known (source) and unknown (target) classes.

Zero-shot Recognition: Given the synthesized testing prototypes and the representation of the test sample $g(V)$, the class label can be inferred using a simple nearest neighbor lookup:

$$\hat{z} = \text{argmax}_{\tilde{z} \in Z_{Te}}\, cos\left(g\left(V\right), \Pi_{\tilde{z}}\right) \tag{5}$$

$cos\left(\cdot\right)$ indicates the cosine similarity, which takes scale into account and works better than a dot product (as we illustrate in Table 3). The only missing part we have not discussed is how to obtain representation of the test video $g(V)$.

The simplest approach is to define $g(V)$ using the $\mathcal{O}-$stream and $\mathcal{S}-$stream directly. This approach would correspond to $g(V) = \left[\bar{\mathbf{x}}^{\mathcal{O}}, \bar{\mathbf{x}}^{\mathcal{S}}\right]$. However, as we show in the experiments this tends to produce poor performance. One of the reasons is that contextual information among all three streams and the fusion network are not utilized and hence the individual predictions obtained using the object and scene streams tend to be noisier. The alternative which works much better in practice, is to define $g(V)$ with respect to the training class prototypes, by forming a "pseudo-instance" prototype. This approach is inspired by ConSE [22] and we describe it in details below.

Probability Calibration: We first employ Platt Scaling [24] to calibrate the output of the fusion network $f\left(\cdot\right)$ into a probability distribution $p\left(\cdot\right)$, defined for each of the training classes. Hence the probability of video V_i belonging to a class label $z \in Z_{Tr}$ is denoted $p\left(z|\bar{\mathbf{x}}_i\right)$, such that the sum across all training classes is $\sum_{z=1}^{|Z_{Tr}|} p\left(z|\bar{\mathbf{x}}_i\right) = 1$.

Pseudo-Instance Prototype: For a testing instance, we synthesize a "pseudo" prototypes as in ConSE [22]. We use z_t to denote the t^{th} most likely training label for video V according to $p\left(\cdot\right)$ function; and $p\left(z_t|\bar{\mathbf{x}}\right)$ is the probablity of video V belonging to training label z_t, which is also the t^{th} largest probability for the posterior of video V over all training classes. Thus given the top T predictions, the pseudo prototype of the testing instance V can be synthesized by using our OSR matrix Π, formally we have

$$g(V) = \frac{1}{\Delta} \sum_{t=1}^{T} p\left(z_t|\bar{\mathbf{x}}\right) \cdot \Pi_{z_t} \tag{6}$$

where $\Delta = \sum_{t=1}^{T} p\left(z_t|\bar{\mathbf{x}}\right)$ is a normalization factor; and Π_{z_t} indicates the z_t-th column of Π.

4. Experiment

We conduct a number of experiments to explore the benefits of our semantic formulation. We start by showing that our object-scene semantic fusion (OSF) network is effective for supervised action and video categorization (Sec. 4.2).

We then show effectiveness of object and scene semantic representation (OSR) that OSF allows us to discover from data. Specifically, we show that OSR is effective for (i) computing semantic distance among video classes by using it to discover class group structure through clustering (Sec. 4.3) and (ii) that it can be utilized for effective zero-shot classification (Sec. 4.4) .

4.1. Experimental Setup

Datasets: We adopt two challenging large-scale video benchmark datasets to evaluate our approach.

ActivityNet [7] is a recently released large-scale video dataset for human activity recognition and understanding. ActivityNet consists of $27,801$ video clips annotated into 203 activity classes, totaling 849 hours of video. Compared with existing action recognition benchmarks (*e.g.*, UCF101 [33] or HMDB51 [14]), ActivityNet is more challenging, since it contains fine-grained action categories that require subtle details to differentiate among (*e.g.*, *drinking beer* and *drinking coffee*). ActivityNet provides both *trimmed* and *untrimmed* videos for its classes. Trimmed videos consist of hand annotated segments that contain frames corresponding to given actions; untrimmed videos have much longer videos which contain frames irrelevant to the dominant action or multiple actions. We use the more challenging *untrimmed* setting for our experiments. ActivityNet consists of training, validation and test splits, however, test split is not made available by the authors. To this end we use validation split as our test set.

Fudan-Columbia Video Dataset (FCVID) [12] contains $91,223$ web videos annotated manually into 239 categories. Categories cover a wide range of topics (not only activities), such as social events (*e.g.*, *tailgate party*), procedural events (*e.g.*, *making cake*), object appearances (*e.g.*, *panda*) and scenic videos (*e.g.*, *beach*). We use standard split of $45,611$ videos for training and $45,612$ videos for testing.

Evaluation Metrics: To evaluate our OSF network for supervised classification, we adopt the standard training and testing splits and compute average precision for each class as suggested in [7, 12]. Mean average precision (mAP) is used to measure the overall performance on both datasets. For zero-shot learning, since there is no off-the-shelf splits defined on these datasets, we split the video datasets into source and target categories. More precisely, we split ActivityNet into 140 source and 63 target classes; FCVID into 160 source and 79 target classes. Mean accuracy (the mean of the diagonal of the confusion matrix) is used to measure the zero-shot learning performance.

Word2Vec Embedding: To generate semantic word representations, we compute $1,000$-dimensional embedding vector by training word2vec [19] on a large text corpus, including UMBC WebBase (3 billion words) and the latest Wikipedia articles (3 billion words).

4.2. OSF Network for Supervised Recognition

In this section we focus on exploring the effectiveness of our object-scene fusion network.

Baselines: We compare with a number of alternative methods to combine multiple features in supervised classification. Among them, early and late fusions are two straightforward ways to integrate multiple features and are two variants of our model.

1. *Early Fusion-NN*, concatenates all three streams into a long vector and then uses it as the input to train a neural network for categorization;
2. *Late Fusion-NN*, trains a neural network classifier using each of three streams independently and then the outputs from all the networks are averaged to obtain the final prediction scores;
3. *Early Fusion-SVM*, utilizes the χ^2-kernel SVM for classification, where kernel matrices are first computed for each stream and then averaged for classification;
4. *Late Fusion-SVM*, learns a χ^2-kernel SVM classifier for each stream and then combines prediction results;
5. *SVM-MKL* [20], combines the multiple stream features using multiple kernel learning with χ^2-kernel.

We note that most of these baselines are very strong as they are using exactly the same features as our fusion network and complex state-of-the-art non-linear classifiers.

Results: Table 1 summarizes the comparisons of our approach and alternative methods. As can be seen from the table, our OSF network achieves 56.8% and 76.5% mAP on ActivityNet and FCVID respectively, outperforming other fusion baselines by clear margins. For early fusion, direct concatenation of features is the most straightforward way of combining representations; this, however, suffers from high dimensionality ($> 25k$ dimensions) which leads to overfitting. Late fusion suffers from the "heterogeneous" classification scores coming from each stream; each stream has varying discriminative capacity and (may) results in incomparable classification scores. In contrast to the alternative fusion methods, our OSF network can implicitly explore the correlations among the streams to derive a fused representation, which is more semantically discriminative for recognition. In addition, compared with the state-of-the-art published results 42.5% [12] and 73.0% [7], our OSF framework achieves 3.5% and 14.3% (percentage points) improvement on FCVID and ActivityNet respectively. Note, results in [12] and [7] are obtained by combining multiple state-of-the-art handcrafted visual features (*e.g.*, improved dense trajectories) and deep features. Our network achieves superior performance by jointly modeling semantic representations (objects and scenes) with low-level deep features. Further improvement can be obtained by considering motion features, which we currently omit.

	ActivityNet	FCVID
Early Fusion-NN	55.9	75.2
Late Fusion-NN	54.4	73.3
Early Fusion-SVM	55.8	75.5
Late Fusion-SVM	54.6	73.4
SVM-MKL [20]	56.3	74.9
Heilbron *et al*. [7]	42.5	–
Jiang *et al*. [12]	–	73.0
OSF Network	**56.8**	**76.5**

Table 1. Comparisons with alternative baselines on ActivityNet and FCVID datasets.

	ActivityNet	FCVID
$\mathcal{F}-\text{stream}$	47.4	67.7
$\mathcal{O}-stream$	44.8	55.5
$\mathcal{S}-stream$	18.8	41.3
$\mathcal{F}-stream+\mathcal{O}-stream$	56.2	75.6
$\mathcal{F}-stream+\mathcal{S}-stream$	52.6	72.3
$\mathcal{O}-stream+\mathcal{S}-stream$	55.4	72.8
OSF Network	**56.8**	**76.5**

Table 2. Results of variants of our network. "+" denotes two streams used in network fusion.

To further evaluate the contribution of each stream in our network, we break down our network with different combinations of the three streams. The results are reported in Table 2. We adopt a 3-layer NN classifier for each single stream (similar in structure to our fusion network); and a variant network with two streams. We can see that the performance of $\mathcal{F}-stream > \mathcal{O}-stream > \mathcal{S}-stream$, which indicates that though the high-level semantic information expressed in objects and scene is important, the generic feature stream ($\mathcal{F}-stream$) still has significant low-level discriminative information which is very useful for classification. In addition, since scene detectors are usually prone to noise, especially in complex long videos with cluttered background, the performance of $\mathcal{S}-stream$ is lower. Notice that $\mathcal{S}-stream$ achieves significantly better results on FCVID than ActivityNet, since categories in ActivityNet are all actions while FCVID contains more generic video classes, where scene clues are more important. In addition, the three streams are complementary. Combining arbitrary two streams offers better performance than single stream. Our OSF network result is further improved performance over pair-wise combinations.

4.3. Object and Scene Semantic Representation

We now investigate the object and scene semantic representation derived from the trained OSF network on FCVID. For each class of interest, we obtain a pseudo video representation that maximizes the neuron activity, identifying the most discriminative objects for the specified video class. Since related classes share certain objects and scenes, we expect their pseudo representations to be similar. To validate the effectiveness of the derived video representation, we compute the cosine similarity between each pair of video classes using the pseudo representations and then obtain the similarity matrix of all the categories. We perform Normalized Cut method to group and order the categories of the similarity matrix for visualization in Figure 5.

As we can see from the figure, the pseudo video representation can indeed discover some group structures of the video classes. We also compare with the groups discovered using word vectors (blue dashed lines). Comparing all rows in the figure, the group structure generated by the object and scene semantic representations can identify more fine-grained categories, while similarity computed by word vectors seems to group too many classes together. Since word vectors are trained on large text corpus, they fail to distinguish categories with similar class names that are visually and semantically different (*e.g*., *make juice* and *make paper plane*). In the first two groups of the second row, clearly the scene information played an important role to separate classes *baseball*, *sportsTrack* and *soccer Professional* (outdoor sports) into a separate group from classes of *barbell workout*, *fencing*, and *pull ups* (indoor sports). This result validates the superiority of our object and scene representation. Interestingly, our grouping results are even better than the manually labeled hierarchy provided by the dataset. For example, our method can group the following classes together: *rafting*, *fishing*, *mountain*, since these classes have similar objects appearing (*e.g*., *mountain*, *raft* and *water*) and highly coherent scene information (*e.g*., *outdoor*, *sky*). Nevertheless, manual defined ontology categorizes them as *extreme sports*, *sceneries* and *leisure sports* respectively.

4.4. Zero-shot Learning

Baselines: We compare the following methods for large-scale zero-shot recognition:

1. *DAP-word*. Support vector regressors are used to learn to regress from concatenated features from three streams to each dimension of 1000-d word vector representing a class. For zero-shot learning, the predicted word vectors of testing instances are matched against the 1000-d word vector prototypes of unknown classes, obtained using word2vec, with nearest neighbor approach. This is a generalization of DAP [15].

2. *ConSE* [22] uses the same $p(\cdot)$ function to predict the posterior of one testing instance belonging to each known class. Eq.(6) is utilized to synthesize the pseudo-instance prototypes from known classes by replacing the semantic representations (Π_{z_t}) with 1000-d word vectors for each class; the testing class prototypes, Eq.(5), are also replaced by 1000-d word vectors.

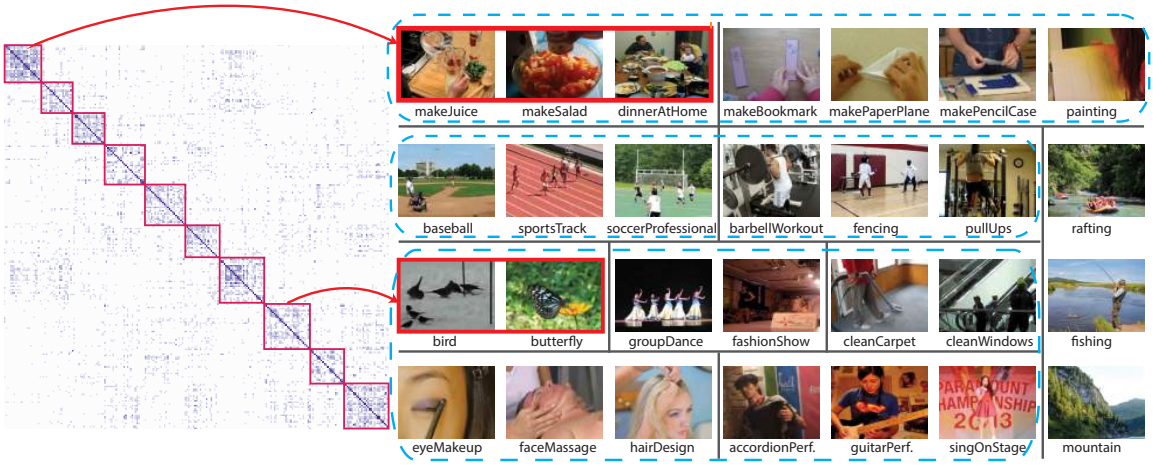

Figure 5. **Left**: Similarity matrix of categories in FCVID computed with the derived OSR, where the red boxes indicate the automatically generated category groups. **Right**: Visual examples of the groups indicated on the similarity matrix, with the red arrows indicating the correspondence of the similarity matrix with the class examples. Groups we discovered are separated by black lines; groups discovered by word vectors are circled by blue dashed lines.

3. *ConSE-pseudo* is a variant of ConSE which is more comparable to our method. The main steps of ConSE-pseudo are the same as ours, while the difference is that ConSE-pseudo replaces our semantic representation of known and unknown classes with 1000-d word vectors for zero-shot recognition in both Eq.(6) and Eq.(5).

4. Nearest neighbor (*NN*): uses Eq.(4) to synthesize the prototypes of testing classes and $\hat{z}_i = \text{argmax}_{\tilde{z} \in Z_{Te}} cos\left(\left[\bar{\mathbf{x}}_i^O, \bar{\mathbf{x}}_i^S\right], \Pi_{\tilde{z}}\right)$ to infer the class label for V_i. This alternative is discussed in Sec. 3.3.

We compare these methods to our proposed model (Ours) that uses OSR and the two variants discussed in Section 3.3.

Results: The zero-shot learning results are summarized in Table 3. Our method is better than all the other baselines on both datasets. We can see that using the semantic representation derived by Eq.(2) can offer better zero-shot recognition performance. This is validated by two observations: (1) Our results improve by 1.4% and 1.3% percentage points (or 13% and 12% respectively) over ConSE, which is a state-of-the-art approach for zero-shot learning. The improvements are largely due to our semantic representation obtained by mining visual video class-object/video class-scene correlations, which therefore is more semantically discriminative than word vectors trained with text corpus. (2) This improvement does not come from the way we generate testing-class prototype by Eq.(5): our results are 3.5% and 2.6% higher than those of ConSE-pseudo; the only difference between our method and ConSE-pseudo is that ConSE-pseudo replaces our semantic representation with semantic word vectors. (3) The results of all methods are better than those of *NN*. This is in part due to the contextual information shared across streams, which can be discovered in the OSF network, is not fully utilized.

	ActivityNet	FCVID
Chance	1.6	1.3
DAP-word [15]	11.3	9.0
ConSE [22]	10.7	10.6
ConSE-pseudo	8.6	9.3
NN	8.5	8.8
Ours (Dot Product)	11.8	11.4
Ours (+$\mathcal{F}$-Stream)	11.6	11.8
Ours	**12.1**	**11.9**

Table 3. Zero-shot Learning Accuracy(%).

5. Conclusion

We present a novel Object-Scene semantic Fusion (OSF) framework for large-scale video understanding, which has a number of appealing properties. Our fusion network combines three streams (*i.e.*, object, scene and generic feature) of information using a three-layer neural network to model object and scene dependencies. This results in supervised video classification improvements in two large-scale benchmark datasets. Further, by examining and back propagating information through the fusion layers, semantic relationships (correlations) between video classes (or activities) and objects/scenes can be identified. These relationships can, in turn, be utilized as semantic representation for the video classes themselves. We empirically evaluate the learned representations in the task of zero-shot learning and clustering, and the results corroborate the effectiveness of the discovered relationships.

Acknowledgment: Z. Wu and Y.-G. Jiang were supported in part by a China's National 863 Program (#2014AA015101) and a grant from NSFC (#61572134).

References

[1] I. Biederman. Recognition by components - a theory of human image understanding. *Psychological Review*, 1987. 3.1

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 3.1

[3] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 2

[4] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong. Learning multi-modal latent attributes. *TPAMI*, 2013. 2

[5] Y. Fu and L. Sigal. Semi-supervised vocabulary-informed learning. In *CVPR*, 2016. 1

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[7] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 1, 4.1, 4.2

[8] S. J. Hwang and L. Sigal. A unified semantic embedding: relating taxonomies and attributes. In *NIPS*, 2014. 1

[9] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *ECCV*, 2010. 1, 2

[10] M. Jain, J. C. van Gemert, and C. G. Snoek. What do 15,000 object categories tell us about classifying and localizing actions? In *CVPR*, 2015. 1, 2

[11] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. High-level event recognition in unconstrained videos. *IJMIR*, 2013. 2

[12] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *CoRR*, 2015. 1, 4.1, 4.2

[13] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 1, 2

[14] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV*, 2011. 4.1

[15] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE TPAMI*, 2013. 1, 1, 4.4

[16] L.-J. Li, H. Su, E. P. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*, 2010. 1, 2

[17] J. Liu, Q. Yu, O. Javed, S. Ali, A. Tamrakar, A. Divakaran, H. Cheng, and H. Sawhney. Video event recognition using concept attributes. In *IEEE Workshop on WACV*, 2013. 1

[18] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009. 2

[19] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 3.3, 4.1

[20] P. Natarajan, S. Wu, S. Vitaladevuni, X. Zhuang, S. Tsakalidis, U. Park, R. Prasad, and P. Natarajan. Multimodal feature fusion for robust event detection in web videos. In *CVPR*, 2012. 5, 4.2

[21] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, 2015. 2

[22] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014. 3.3, 2, 4.4

[23] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014. 3.3

[24] J. Platt. Probabilities for SV machines. In *Advances in Large Margin Classifiers*, 2000. 3.3

[25] A. Prest, C. Schmid, and V. Ferrari. Weakly supervised learning of interactions between humans and objects. *IEEE TPAMI*, 2012. 1, 2

[26] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where – and why? semantic relatedness for knowledge transfer. In *CVPR*, 2010. 3.3

[27] S. Sadanand and J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012. 2

[28] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *IJCV*, 2013. 2

[29] N. Shapovalova, A. Vahdat, K. Cannons, T. Lan, and G. Mori. Similarity constrained latent support vector machine: An application to weakly supervised action classification. In *ECCV*, 2012. 1

[30] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR*. 2014. 2, 3.2

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

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2, 3.1

[33] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CRCV-TR-12-01*, 2012. 4.1

[34] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 1

[35] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013. 1, 2

[36] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *ACM MM*, 2015. 2

[37] C. Xu, S.-H. Hsieh, C. Xiong, and J. J. Corso. Can humans fly? action understanding with multiple classes of actors. In *CVPR*, 2015. 1

[38] F. X. Yu, L. Cao, R. S. Feris, J. R. Smith, and S.-F. Chang. Designing category-level attributes for discriminative visual recognition. In *CVPR*, 2013. 3.3

[39] M. D. Zeiler and R. Fergus:. Visualizing and understanding convolutional networks. In *ECCV*. 2014. 2

[40] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. In *ICLR*, 2015. 2

[41] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*. 2014. 3.1

[42] Y. Zhu, A. Fathi, and L. Fei-Fei. Reasoning about object affordances in a knowledge base representation. In *ECCV*, 2014. 1