Zero-Shot Action Recognition in Videos: A Survey

Valter Luís Estevam Junior[a,c,*], Helio Pedrini[b] and David Menotti[c]

[a] Federal Institute of Paraná, Irati-PR, 84500-000, Brazil
[b] Universiy of Campinas, Institute of Computing, Campinas-SP, 13083-852, Brazil
[c] Federal University of Paraná, Department of Informatics, Curitiba-PR, 81531-970, Brazil

ARTICLE INFO

Keywords:
Zero-shot learning
Video action recognition
Visual embedding
Semantic label embedding
Experimental protocols
Deep features

ABSTRACT

Zero-Shot Action Recognition has attracted attention in the last years, and many approaches have been proposed for recognition of objects, events, and actions in images and videos. There is a demand for methods that can classify instances from classes that are not present in the training of models, especially in the complex task of automatic video understanding, since collecting, annotating, and labeling videos are difficult and laborious tasks. We identify that there are many methods available in the literature, however, it is difficult to categorize which techniques can be considered state of the art. Despite the existence of some surveys about zero-shot action recognition in still images and experimental protocol, there is no work focusing on videos. Hence, in this paper, we present a survey of the methods comprising techniques to perform visual feature extraction and semantic feature extraction as well to learn the mapping between these features considering specifically zero-shot action recognition in videos. We also provide a complete description of datasets, experiments, and protocols, presenting open issues and directions for future work essential for the development of the computer vision research field.

1. Introduction

In recent years, many works in the computer vision field have explored the human action or activity recognition problem using still images or videos. Several surveys [83, 65, 35, 26, 103, 40] show approaches addressing the human action recognition problem by proposing new visual or semantic features describing the actions more accurately. For example, the Dense Trajectory Features (DTF) [87] and its variant, the Improved Dense Trajectories (IDT) [86], are the most successful methods based on handcrafted visual features. They use Histogram of Gradient (HoG), Histogram of Optical Flow (HoF), and Motion Boundary Histogram (MBH) descriptors encoded with Bag-of-Features (BoF) or Fisher Vectors (FV). Another group of works explores semantic features such as poses, poselets, objects, scenes, and attributes. With these descriptors, it is necessary less computation to address the intra-class variability problem [103], for example. More recently, deep learning has been applied to human action recognition to extract visual features by exploring the convolution operation, temporal modeling, and multi-stream configuration [40].

All these approaches, visual or semantic-based, suffer inherent drawbacks, for example: (i) they do not generalize very well on large and complex datasets such as UCF101 [79], HMDB51 [43] or Kinetics [8]; (ii) the handcraft visual features, as well as manual-annotated semantic features, require heavy human labor or expert knowledge, which are not always available; and (iii) many labeled examples are required to reduce the generalization problem when deep learning is used. However, in a real-world scenario, there are many more actions than in the benchmark datasets used to learn the models. Moreover, the new examples may be unlabeled, which makes the supervised methods inappropriate. In this context, Zero-Shot Learning (ZSL), a particular case of transfer learning paradigm [60], emerges attempting to overcome these limitations. The human ability to recognize an action without ever having seen it before, that is, associating semantic information from several fonts to the visual appearance of actions, is the inspiration of ZSL approaches [39].

Since the seminal work of Lampert et al. [44], ZSL has been employed in recognition of objects, actions, activities, and events [22]. In Fig. 1, we provide an overview of ZSL approaches performed in videos. In this example, some videos from `Apply Eye Makeup` and `Ice Dancing` action classes are used to extract visual features in order to compose a visual space. Commonly, these visual features are obtained using the IDT method [87]; HoG, HoF, and MBH algorithms with Bag-of-Features approach [72]; or using deep features from C3D [81] or I3D [8] deep models.

Auxiliary semantic *side information* encoded as word vectors [39], hierarchical structures [2] or attributes manually annotated [69] are necessary to obtain a semantic space in which there are representations to the seen classes and also to many unseen possible classes, called prototypes. In ZSL, if we try to recognize a new video from the `New Action` class, which has never been seen before, in addition to extract visual features, it is necessary to associate them with a suitable prototype assigning a label. This is made by learning a $f(\cdot)$ *mapping function* between these spaces.

Fu et al. [22] and Xian et al. [93] provide an overview of ZSL action recognition problems, especially about still images and experimental protocols. More recently, Wang et al. [91] investigate the ZSL paradigm with focus on settings, methods, and applications. On the other hand, we investigate how can we recognize, using videos, what a human

*Corresponding author
valter.junior@ifpr.edu.br (V.L. Estevam)
ORCID(s): 0000-0001-9491-5882 (V.L. Estevam); 0000-0003-0125-630X (H. Pedrini); 0000-0003-2430-2030 (D. Menotti)

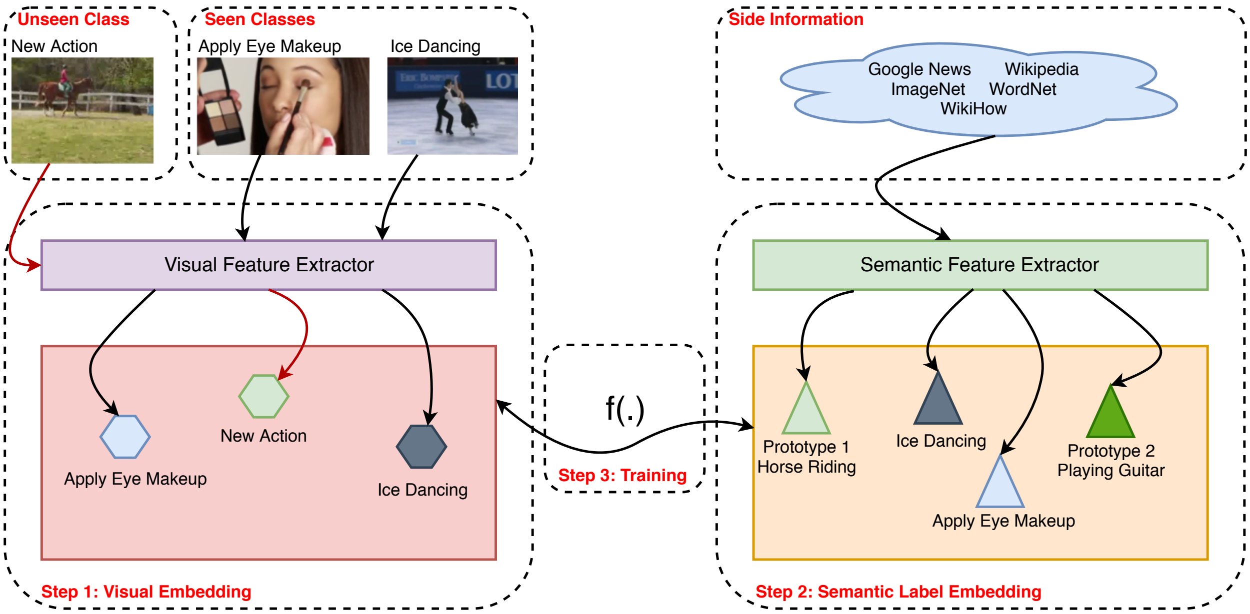


Figure 1: Schematic representation of a ZSL human action recognition framework.

is doing[1] without having seen it before. To the best of our knowledge, there is no survey about ZSL action recognition in videos and our main contributions are three-fold: (i) to provide a complete description of ZSL methods applied to human action recognition in videos detailing the methods used to extract visual features, semantic features, as well to perform the training; (ii) to present a discussion about the limitations of the benchmark datasets and evaluation protocols adopted in works in the literature; and (iii) to identify open issues pointing future research strategies to inspire different approaches in this knowledge domain.

The remainder of the text is organized as follows. We review the methods used to perform visual and semantic embedding in Section 2 and provide a complete description of ZSL approaches in Section 3. The benchmark datasets are presented in Section 4, whereas experimental protocols and performance are discussed in Section 5. We discuss open issues and directions for future work in Section 6. Finally, some concluding remarks are presented in Section 7.

2. Visual and Semantic Label Embedding Steps

Two crucial steps in any ZSL method are the visual and semantic label embedding. They are responsible for providing the features used to map the visual appearance to the semantic description of actions.

2.1. Visual Embedding Step

In Table 1, we present a set of methods used to perform visual embedding, organizing them into two groups: handcrafted (HF) and deep features (DF) based approaches. We also provide a brief review of these methods.

[1]Both actions and activities are investigated.

Handcrafted Visual Features Bag-of-Features methods are used in [72, 69, 51, 73, 21]. In the first ZSL work in videos [51], the visual words were obtained from a descriptor composed of spatio-temporal volumes and 1D Gabor detector. In later works, a well known combination of HoG, HoF, and MBH descriptors is used. More details about classical BoF descriptor can be found in [59].

An improved BoF descriptor, DTF, was proposed in [84] and used in [94, 25]. The DTF is able to characterize shape (point coordinates), appearance (Histogram of Gradient), motion (Histogram of Optical Flow) and variations on motion (Motion Boundary Histogram). The *dense* term refers to initial sampling in each frame with a grid of $W \times W$ points combined with spatio-temporal pyramid approach, as shown in Fig. 2 (on the left).

Since it is not possible to apply tracking in homogeneous regions of video frames, these points are removed from sampling. For each remaining point in each frame, the dense flow field is computed, and subsequent frames are concatenated to create a trajectory descriptor. Next, static trajectories of each sampled point are also removed (Fig. 2 (center)). Then, descriptors are computed from spatio-temporal volumes with $N \times N \times L$ dimension (e.g., 5 pixels × 5 pixels × 15 frames), subdivided in $n_\sigma \times n_\sigma \times n_\tau$ cells (e.g., 2 × 2 × 3), as shown in Fig. 2 (on the right). In the end, a codebook for each descriptor (trajectory, HoG, HoF, MBH) is created by fixing the number of visual words per descriptor to 4000 and performing k-means algorithm eight times, while keeping the results with the lowest error. The resulting histograms of visual words are used as video representation.

As shown in [85], the performance of the HoF descriptor degrades significantly in the presence of camera motion (e.g., pan, tilt and, zoom). Hence, the IDT method [87] provides a mechanism to canceling out the camera motion from

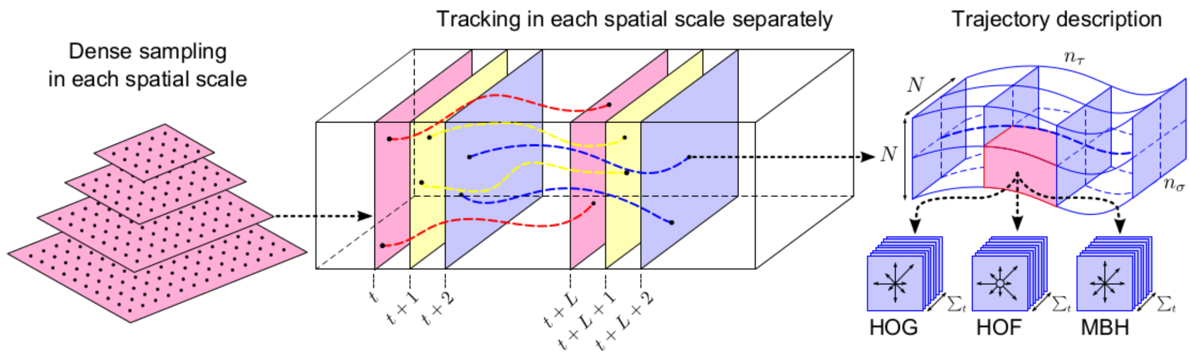


Figure 2: Dense Trajectory Method. Source: [84].

Table 1
Methods used to perform visual embedding for ZSL in videos. Handcrafted Features (HF) and Deep Features (DF).

	Method	Used in appoaches
HF	BoF	Liu et al. [51]
		Qiu et al. [69]
		Rohrbach et al. [73]
		Rohrbach et al. [72]
		Fu et al. [21]
	DTF	Xu et al. [94]
		Guadarrama et al. [25]
	IDT	Kodirov et al. [39]
		Gan et al. [23]
		Xu et al. [96]
		Xu et al. [95]
		Xu et al. [97]
		Gan et al. [24]
		Alexiou et al. [3]
		Wang and Chen [90]
		Zhang and Peng [100]
		Liu et al. [52]
		Fu et al. [19]
DF	From [41]	Jain et al. [32]
	VGG	Gan et al. [24]
		Zhang and Peng [100]
	ResNet-200	Zhu et al. [102]
	3D CNN	Mishra et al. [57]
	C3D	Wang and Chen [89]
		Zhang et al. [99]
		Liu et al. [52]
		Hahn et al. [27]
		Wang and Chen [88]
	I3D	Roitberg et al. [74]
		Piergiovanni and Ryoo [64]
	Other	Li et al. [48]

optical flow in the tracking phase. A human detector [67] is used to remove trajectories in regions where humans are not found. This method presents a promising performance and is used by [39, 23, 96, 95, 97, 24, 3, 90, 100, 52, 19]. However, it is computationally intensive and becomes impracticable on large-scale datasets [81, 52].

Deep Visual Features Deep learning has attracted much attention in recent years due to its advances in the image classification domain, where several convolutional neural network models (ConvNets or CNNs) were proposed with great performance [66]. ConvNets pre-trained in large-scale datasets can be easily used in multiple applications, and this ability is explored in ZSL. For example, a CNN pre-trained on the ImageNet dataset [13], called VGG 19 [78], is used in [24], providing a detector for 1000 different concepts. In their work, videos are represented in terms of detected visual concepts that are classified as relevant or irrelevant according to their similarity with a given textual query. Jain et al. [32] also propose an approach that relates objects and actions using the ImageNet dataset for training a CNN model from [41]. In [102], a ResNet-200 model is initially trained on ImageNet and fine-tuned on ActivityNet dataset [28]. However, such image-based deep models are not suitable for direct video representation due to the lack of motion modeling, as demonstrated in [81]. This problem can be overcome with deep models that consider spatio-temporal relations, providing features from their fully connected layers (`fc`). This strategy is applied in [57] using 3D CNN [33], in [89, 99, 52, 27, 88] using C3D [81], and in [74, 64] using I3D [8].

In the C3D network [81], full video frames are taken as input and do not require any preprocessing except to resize frames to 128×171 pixels. To propagate spatio-temporal information across all the layers, 3D convolutional filters ($3{\times}3{\times}3$ with stride $1{\times}1{\times}1$) and 3D polling layers ($2{\times}2{\times}2$ with stride $2{\times}2{\times}2$) are used. As shown in Fig. 3, the architecture also has two fully connected layers and a softmax output layer. This model is trained on Sports-1M Dataset [37] and the visual representation is extracted from `fc6` layer resulting in a vector with 4096 dimensions which are usually used without modifications. In [99], the dimensionality is reduced to 500 by applying PCA.

Training 3D ConvNets consists of learning many more parameters than 2D ConvNets. Therefore, the Two-Steam I3D architecture [8] uses a common pre-trained ImageNet Inception-V1 model [31] as base network, adding a batch normalization to each convolution layer. To properly ex-

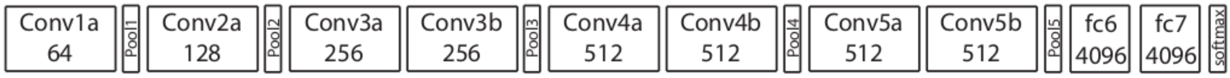


Figure 3: C3D architecture. Source: [81].

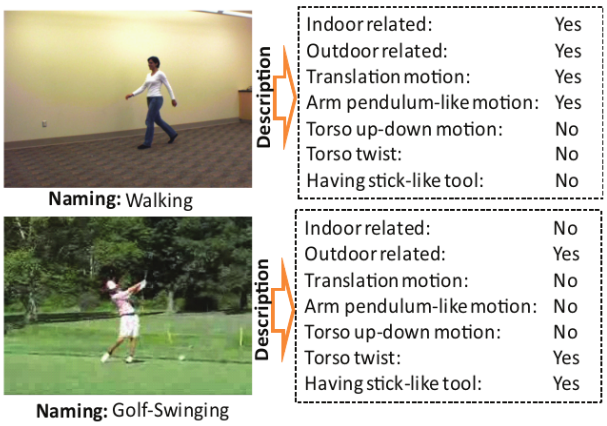


Figure 4: A set of attributes manually defined and annotated that can be directly associated with the visual characteristics of actions. Source: [51].

plore spatio-temporal ordering and long-range dependencies, it uses an LSTM layer after the last average pooling layer of the Inception-V1. Additionally, its performance can be improved by including an optical-flow stream [8]. The I3D model is trained on Kinetics dataset [8], and the visual representation is extracted from the last fully connected layer resulting in a representation of 256 dimensions in [74] and 1024 in [64]. It is likely that ZSL assumption (classes disjunction between the training and testing sets) has been violated since both C3D and I3D models are pre-trained on large-scale datasets [52]. Thus, a new problem emerges through the use of deep learning techniques. We present a detailed discussion on this topic in Section 5.

2.2. Semantic Label Embedding Step

Semantic label embedding is a step in which class labels are associated with a semantic representation using *side information*. As shown in Table 2, there are few strategies used at this moment.

Attribute-based Methods Using manually defined and annotated attributes are a common strategy [90, 57, 19, 51, 72, 23, 21]. As illustrated in Fig. 4, an expert needs to define all attributes and also their values. This approach has some inherent drawbacks, such as: (i) annotating videos is more difficult than annotating images; (ii) in a more complex or complete dataset, several attributes are necessary to alleviate the semantic intraclass variability; (iii) it is difficult to define what attributes are relevant, and (iv) this approach is not scalable.

Alternatively, dictionary learning techniques are proposed in [69, 39]. In these works, visual features are related to atoms in the automatically learned dictionary, alleviating the problem of manual definition of attributes.

Table 2
Methods used to perform semantic embedding for ZSL in videos. Attribute (A) and Word Embedding (WE)

	Method	Used in appoaches
A	Annotated	Wang and Chen [90]
		Mishra et al. [57]
		Fu et al. [19]
		Liu et al. [51]
		Rohrbach et al. [72]
		Gan et al. [23]
		Fu et al. [21]
	Dictionary learning	Qiu et al. [69]
		Kodirov et al. [39]
WE	Semantic hierarchies	Rohrbach et al. [73]
		Gan et al. [23]
	Word2Vec	Xu et al. [94]
		Xu et al. [96]
		Xu et al. [95]
		Jain et al. [32]
		Gan et al. [24]
		Alexiou et al. [3]
		Li et al. [48]
		Xu et al. [97]
		Wang and Chen [89]
		Qin et al. [68]
		Liu et al. [52]
		Wang and Chen [90]
		Mishra et al. [57]
		Mettes and Snoek [54]
		Zhu et al. [102]
		Roitberg et al. [74]
		Martinez et al. [53]
		Hahn et al. [27]
	GloVe	Zhang et al. [99]
		Piergiovanni and Ryoo [64]
		Zhang and Peng [100]
		Guadarrama et al. [25]

Word Embedding Methods Methods based on word embedding have gained attention due to their advantages in exploring side information acquired with less effort than handcrafted attributes or annotations. For example, in [25] semantic hierarchies are mined for subjects, verbs, and objects using the descriptions of videos from YouTube. On the other hand, Rohrbach et al. [73] use a well known WordNet [17] hierarchy to represent the action labels.

The most popular strategy for semantic label embedding in the last years is the skip-gram model [56], more specifically the Word2Vec implementation [55] used in [96, 95, 3, 97, 89, 52, 90, 57, 102, 74, 27]. This model is an efficient method for learning vector representations that captures a large number of syntactic and semantic word relationships [55]. The method consists of learning a neural network that calculates a similarity measure between words based on a softmax output. In ZSL, the semantic vector representation for the interest word (action label) is based on the activation of 300 neurons in a hidden layer of the skip-gram network when this word is provided as input.

Another approach to performing semantic label embedding is a count-based model called Global Vectors (GloVe) [61]. In this model, a large matrix of co-occurrence statistics is constructed by storing words in rows and contexts in columns. Intuitively, these statistics encode the meaning of words since semantically similar words occur together more frequently than semantically dissimilar words. Semantic vectors are learned such that their dot product equals the co-occurrence probability [1]. These learned vectors are used to represent semantic information of class labels in recent works [64, 99, 100] with promising performance, possibly because this approach outperforms Word2Vec on the word analogy prediction task [61], providing more semantically discriminative information.

Table 3
Overview of zero-shot human action recognition methods in videos. The following aspects are considered: Actions (AC) or Activities (AT); if the method explores few-shot learning (FSL); if the method uses the transductive setting (TS).

Group	Approach name	AC	AT	FSL	TS
Attributes	DDA [51]	✓		✓	
	SDL [69]	✓			
	CAC [73]		✓		
	PST [72]		✓		✓
	HSM [25]		✓		
	TMV [19]		✓	✓	✓
	M2LATM [21]		✓	✓	
Word label embedding	SES [94]	✓			
	UDA [39]	✓			✓
	MR+DA+ST+HC [96]	✓		✓	✓
	MTE [95]	✓			✓
	ConSE [48]	✓			
	MR+DA+ST+HC [97]	✓		✓	✓
	ASE [88]	✓		✓	✓
	JLEL [89]	✓		✓	
	BiDiLEL [90]	✓			✓
	ZSECOC [68]	✓			
Semantic relationship	SIR [23]	✓		✓	
	SAC [3]	✓			✓
Objects as attributes	Objects2action [32]	✓		✓	
	SAOE [54]	✓			
Multi-modal learning	HSE [99]		✓		
	SME [64]		✓		
	Action2vec [27]	✓			✓
Generative models	GZSL [57]	✓		✓	✓
	VDS [100]	✓			
Others	URM [102]	✓			
	ID [74]	✓		✓	

3. Training Step

The mapping between visual and semantic features is learned in the training step. In Table 3, we present the ZSL methods grouped into seven approach categories: attributes, word label embedding, semantic relationship, objects as attributes, multi-modal learning, generative models, among others. Furthermore, we categorize the works into actions or attributes. As shown, most works focus on action recognition instead of activity recognition, since the latter is a more complex task, requiring more discriminative semantic representations such as attributes or multi-modal based. In addition, we present if the work addresses the few-shot learning problem and if it uses any transductive setting.

Attribute-based Methods There are many methods that use attributes to perform ZSL such as: data-driven attributes (DDA) [51], sparce dictionary learning (SDL) [69], hierarchical semantic model (HSM) [25], composite activity classification (CAC) [73], direct and indirect attribute prediction (DAP) and (IAP) [45], and multi-modal latent attribute topic model (M2LATM) [21].

Liu et al. [51] proposed the first method to learn a mapping between visual and semantic attributes. In training, the attributes are manually defined, and functions are learned to project the instances in the same attribute space using SVM classifiers and raw visual features. These learned classifiers are used in the testing to project new instances in the built attribute space using only the raw features. The attributes obtained with SVM classifiers are latent and called data-driven attributes (DDA). With all instances projected in the attribute space, it is possible to classify new examples considering that an action instance is a point in this space and the positions of action instances will be close to the positions of their corresponding action classes. Therefore, a K-Nearest Neighbor (KNN) algorithm can be used to assign a proper label. This method is possible because, even for unknown classes, their attributes are known without previous visual examples.

Another attribute-based approach is presented in [69] and consists of a sparse dictionary learning technique (SDL). In their work, visual features (e.g., silhouette-based feature [49], optical flow [15], HoG [12], action bank [75], and space-time interesting points (STIP) [46]) are obtained and given as input signals to compute a dictionary and its sparse codes by minimizing the reconstruction error. Each human action is decomposed as sparse linear combinations of basic action units, the human action attributes or atoms in the dictionary.

Guadarrama et al. [25], on the other hand, proposed an approach based on hierarchical semantic models (HSM) as shown in Fig. 5. There are hierarchies learned to subjects, objects, and verbs. Thus, the training step consists of associating visual information with the corresponding leaf in the hierarchy. In their work, the DTF method is performed to extract handcrafted features and learn a codebook for the entire

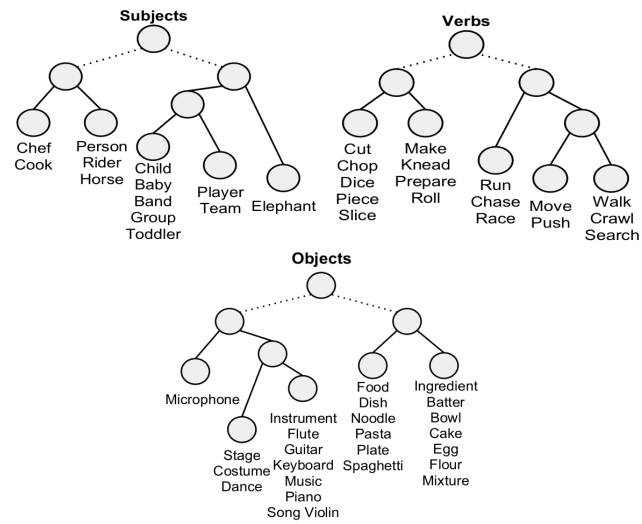


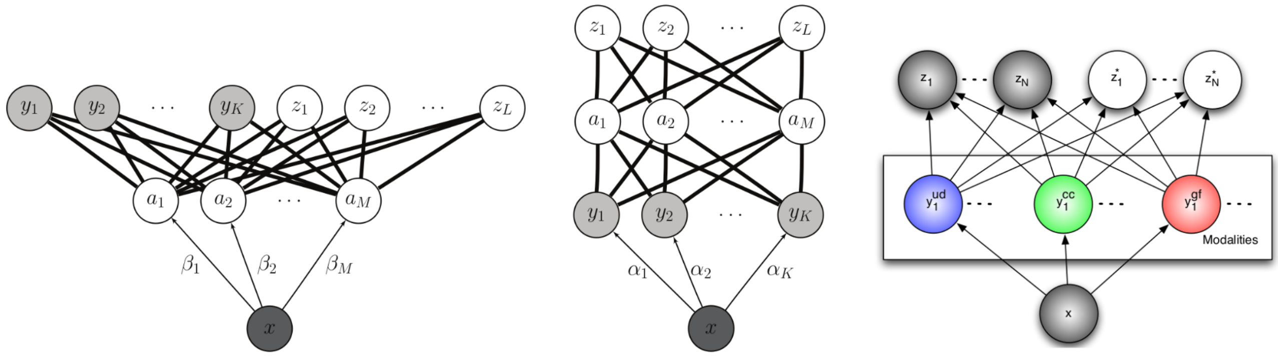
Figure 5: Small portion of semantic hierarchies, as presented in [25].

video representing the activity. Object detectors from [18] and [47] are used to select the maximum score assigned to each object in any frame representing subjects or objects. A multi-channel approach is applied to combine activities and subjects or objects descriptors passing this information to a non-linear SVM. Once the leaf classifiers are trained, it is possible to predict nodes by trading off specificity with semantic similarity in the sense of how semantically close the predicted triplet is to the true action. The posterior probabilities of internal nodes in the hierarchy are obtained by learning one-vs-all SVM classifiers for the leaf nodes and summing them. These values are used to compute the WUP similarity (from Wu & Palmer work [92]) and predict the better triplet (subject, object, verb) to each class label. However, other similarity functions defined over semantic hierarchies could also be used. Still dealing with activity recognition, but now on composite activities, the CAC method is proposed in [73], which considers three different features (video-based, context-based and co-occurrence) and computes confidence scores of attributes for each feature. Visual features are extracted using HoG, HoF, and MBH around densely sampled points [84] and a one-vs-all SVM is trained to predict these scores. External knowledge is used to identify which attributes are more relevant, and a modified KNN algorithm is used to replace the Euclidean distance with weights of class-attribute associations.

Lampert et al. [45] proposed two methods to recognize new classes using attributes manually defined or annotated, the Direct Attribute Prediction (DAP) and the Indirect Attribute Prediction (IAP). These methods were created to object classification, but are used as a training strategy or baseline in several ZSL action recognition works such as [94, 96, 39, 95, 97, 57, 68, 72, 3, 21, 19, 23, 20]. Fig. 6(a) shows a graphical representation of the DAP method. A classifier is learned to map each visual representation $\boldsymbol{x}$ to the attributes $a_1, \ldots, a_n$ of training examples, obtaining a set of parameters $\beta_1, \ldots, \beta_M$, which enable to predict the seen labels $y_1, \ldots, y_k$. At testing time, these set of parameters $\boldsymbol{\beta}$ are used to estimate the attribute values of each testing examples $z_1, \ldots, z_L$. Fig. 6(b) shows the IAP in which attribute prediction occur differently. First, a classifier learns a set of parameters $\boldsymbol{\alpha}$ to map the visual features $\boldsymbol{x}$ to the training classes $\boldsymbol{y}$. At testing time, the posterior distribution of the training class labels induces the distribution over the labels of unseen classes $\boldsymbol{z}$, employing a class-attribute relationship [45]. Another similar method called Multi-Modal Latent Attribute Topic Model (M2LATM) is proposed in [21] and shown in Fig. 6 (c). Differently to DAP, the M2LATM method works on three types of attributes: user-defined (UD), from any prior ontology; latent class-conditional (CC), discriminative for known classes; and generalized free (GF), which represents shared aspects not presented in the attribute ontology. In the ZSL scenario of [21], the testing set and the prototypes are built with user-defined attributes, and the testing instance attributes are inferred for each new example. Again, a KNN algorithm is applied to identify the neighbors of each prototype, in a similar way to DAP. Finally, the user-defined prototypes are projected onto the full attribute space (considering all types of attributes). This method is proposed in the context of the Unstructured Social Activity Attribute dataset (USAA) [20] and, due to the lack of adequate attribute annotations in other datasets, there is no extensive comparison of results in the literature. Hence, it is used only in [21, 19, 96, 97].

Word Label Embedding Many works in ZSL aim to classify new examples using semantic word embedding vectors by learning a properly mapping with the visual features. This approach takes advantage of a large amount of textual corpus, as well as efficient methods to construct semantic representations. However, the semantic gap problem emerges in this scenario, that is, the distribution of instances in the visual space is often distinct from that of their underlying semantics in the semantic space. This problem is the major issue addressed in the methods: Unsupervised Domain Adaptation (UDA) [39], Semantic Embedding Space (SES) [94], Manifold Regularization (MR) + Data Augmentation (DA) + Self-Training (ST) + Hubness Correction (HC) Xu et al. [96, 97], Multi-task Embedding (MTE) Xu et al. [95], Convex Combination of Similar Semantic Embedding Vectors (ConSE) [48], Joint Latent Embedding Learning (JLEL) [89], Bidirectional Latent Embedding (BiDiLEL) [90], Alternative Semantic Representations (ASR) [88], and Zero-Shot Action Recognition with Error-Correcting Output Codes (ZSECOC) [68].

A regularized sparse coding-based unsupervised domain adaptation method (UDA) is proposed to reduce the semantic gap by solving the domain shift problem [39]. This method is based on learning the projection of visual space onto the semantic space as a dictionary learning and sparse coding problem. The dictionary learned is regularized by two terms: (i) adaptation regularization constraint that maintains the dictionary learned from unlabeled tar-



(a) Direct Attribute Prediction (DAP) (b) Indirect Attribute Prediction (IAP) (c) Multi-Modal Latent Attribute Topic (M2LATM)

Figure 6: Graphical representation of DAP (a) and IAP (b) adapted from [45] and M2LATM (c) adapted from [21]. In graphics (a) and (b), x represents the set of visual features, always viewed, $y_1, \ldots y_K$ are training classes, $z_1, \ldots, z_L$ are testing classes, $\alpha_1, \ldots, \alpha_k$ are parameters learned by a classifier to each class, and $\beta_1, \ldots, \beta_M$ are parameters to attribute prediction. On the other hand, in (c) y can be multi-modal (e.g., video and audio).

get data similar to the dictionary learned from labeled data; and (ii) visual-semantic similarity constraint that regularizes the "closeness" of target data interpretations to their correct class labels in semantic space. Theoretically, multiple semantic spaces can be combined in this method allowing multi-modal learning.

In the SES method [94], the mapping between visual and semantic word embedding is made performing a non-linear SVR with RBF-χ^2 kernel. As actions are visually complex and ambiguous, the mapping models may not generalize well because the volume of training data is small compared to the complexity of available actions. Hence, it is added a transductive self-training strategy that aims to adjust the prototypes of unseen classes to be closer to the projected data points alleviating the domain shift problem [90]. This method is transductive because it utilizes the testing data without class labels. In fact, for each category prototype $g(y*)$ it is searched for the K nearest neighbors among the unlabelled testing instance projections, and re-define the adapted prototype $g'(y*)$ as the average of the K neighbors. The adapted prototypes are now more directly comparable with the testing data for matching. Additionally, a data augmentation procedure is performed with an auxiliary dataset that does not contain classes in common with the target dataset. For example, if a regressor is learned from the HMDB51 dataset, the data augmentation is made with the UCF101 dataset and vice-versa [94].

The hubness problem arises when self-training procedure is used [14]. This problem is characterized by a set of `hub` testing-class prototypes that become nearest to a large number of neighbors in the semantic space degrading the performance. There are two main approaches for performing hubness correction (HC): Normalized Nearest Neighbor (NRN) and Globally Corrected (GC) [14], both transductive. The first approach eliminates the bias by normalizing the distance of each prototype to all testing samples before performing Nearest Neighbor classification, and the second approach damps the effect of hub prototypes by using ranks rather than the original distance measures. Both approaches are used in [96, 94] along with manifold regularization (MR), data augmentation (DA), and self-training (ST).

Xu et al. [95] proposed a more robust method to reduce the domain shift. Their approach is based on multi-task regression that lies closely on a low-dimensional manifold and enables exploring the response variable relation. In their work, a new data augmentation method is proposed, the Prioritized Auxiliary Data Augmentation, formulated as a domain adaptation problem by minimizing the discrepancy between the marginal distributions of the auxiliary and target domains. Another approach for performing a self-adaptive domain shift is the convex combination of semantic embedding vectors (ConSE) [48]. It is based on two-output multilayer perceptron networks with two distinct loss functions: semantic soft label loss and hard binary label loss. The first function consists of a hing rank loss function to measure the similarity of the semantic output and semantic labels and the second one is a softmax loss function which is selected because of its robustness for multi-class classification.

There are methods that, instead of learning mapping functions between different spaces, aim to create a shared space for both visual and semantic data. For example, Joint latent embedding learning (JLEL) [89] is a multi-label ZSL action recognition framework based on the joint embedding of visual and semantic representations into a latent space using two models for parameter estimation: visual and semantic models. Another approach to explore joint latent embedding is the BiDiLEL [90]. The goal of this ZSL framework is to bridging the semantic gap and tackling the hubness problem. Two subsequent stages form the framework. In the bottom-up stage, a latent space is learned via supervised subspace learning that preserves intrinsic structures of visual data and provides a discriminative capability. Next, in the latent space, the mean of projected points of training data in the same class forms a `landmark` corresponding to a class label. The subsequent stage is top-down, and the semantic representations of all unseen-class labels in a given vocabulary are them embedding in the same latent space by retaining the semantic relatedness of all classes guided by the landmarks. The structured prediction method [101], a transductive self-training strategy, is then incorporated to address the domain shift problem [90]. Qin et al. [68] proposed a completely different approach to tackle the domain shift problem called ZSECOC. In their method, a discriminative Error-Correcting Output Code (ECOC) for seen categories from both category level semantics and intrinsic data structures deals with domain shift implicitly by transferring the well-established correlations among seen categories to unseen ones. The ZSECOC is partially derived from the text corpus that has well-defined class hierarchy relationships. In contrast to transductive methods, a simple semantic transfer strategy without using any unseen data is developed to generate effective codes for unseen categories [68].

Finally, it is important to present the work of Wang and Chen [88]. Although not proposing a new approach, it investigates different strategies for bridging the semantic gap. They use alternative semantic representations for ZSL based on textual descriptions of human actions and deep features extracted from still images relevant to human actions. For textual-based descriptions, a corpus obtained from Wikihow, Wikipedia and Online dictionary is preprocessed with natural language techniques (e.g., obtaining all words in documents and removing stoping words such as “is”, “you”, “of”). The word vectors are represented with average word vectors or Fisher word vectors. On the other hand, for image-based description, a dataset is created using action labels as keywords and relevant images collected with search engines. These images are used as input to a pre-trained CNN model, where deep image features are easily obtained. In the end, these features are coded with average feature vectors or Fisher feature vectors.

Semantic Relationships Some works explore relationships between class names and synonyms to obtain more discriminative semantics, such as semantic inter-class relationships (SIR) [23] and synonyms as context (SAC) [3]. In the SIR method, the semantic inter-class relationships between the known and unknown actions are leverage by label transfer learning. In this approach, classifiers of the known action classes are obtained, and various linguistic knowledge bases and semantic correlations approach (e.g., WordNet path length-based and Wikipedia vector based) are explored to obtain inter-class relationships also using the visual concepts detected. However, as previously discussed, word vector embedding often provides weaker discriminative power than manually labeled attributes. Hence, Alexiou et al. [3] proposed a SAC method that improves significantly the performance of several ZSL methods based on word vector embedding. In that work, a set of synonyms for each class name is used to supplement and extend its semantic dimension. A set of word vectors are used to represent an unseen class in a word vector embedding space, and a transductive algorithm is proposed to address the inherent misalignment between multiple domains by self-training. The synonyms are mined from multiple internet dictionaries, including Google, TheFreeDictionary, OxfordDictionary, and WordReference.

Objects as Attributes Two works proposed to use relationships between objects and actions: Objects2action [32] and Spatial-Aware Object Embedding (SAOE) [65]. In the first, the goal is to recognize actions using a semantic word embedding with a skip-gram model of object categories detected in videos. Action labels are associated with objects with a convex combination function of action and object affinities. Two embedding techniques explore these multi-word descriptions and bridge the semantic gap: average word vectors and Fisher word vectors that are also used in [88]. The second approach, SAOE, is based on detection of actors, objects, and their interaction. Action tubes from interactions between actors and objects are scored in three steps: (i) gather prior knowledge of actions, actors and objects, and their interactions; (ii) compute spatial-aware embedding scores for bounding boxes; and (iii) link boxes into action tubes.

Multi-Modal Learning In recent years, some approaches based on multi-modal learning (e.g., from object images, action images, auxiliary datasets, texts, and hierarchical ontologies) have been proposed, such as Hierarchical Sequence Embedding (HSE) [99], Shared Multimodal Embedding (SME) [64], and Action2Vec [27].

HSE [99] is a generic model for embedding sequential data of different modalities into hierarchically semantic spaces. The aim is to embed low-level (clips and sentences) and high-level (video and paragraph) data in the coherently semantic space and use gated recurrent neural networks (GRU) [11] to learn sequence models. In that work, the low-level correspondences are explored by deriving loss functions that ensure the embeddings for clips and sentences. In summary, this method is an encode-decode approach [80] composed of a layer of long short-term memory units (LSTM) [29] or GRU. Piergiovanni and Ryoo [64] proposed a method using multi-modal data (text and video unpaired) to learn a deep autoencoder architecture with 4 generative adversarial neural networks (GANs): video encoder-decoder and text encoder-decoder. Since both text and video are data sequences, they often have different lengths. Learning joint embedding spaces requires that the features from both modalities have the same dimensions. In their work, temporal attention filters proposed by Piergiovanni et al. [63] were used. With this approach, it is possible to learn a mapping from any input length to an N-dimensional vector (e.g., mapping both video and text to 1024-d). The joint representation space is obtained by minimizing the L_2 distance between the embeddings of a pair of text and video.

Hahn et al. [27] proposed another multi-modal embedding method, Action2Vec, inspired by the difficult to relate text and video. Hence, a hierarchical LSTM model is used to generate a joint embedding space for action videos and verbs, providing a novel semantic video representation. Each video is represented by 21 feature vectors each with 4096 dimensions. These vectors are passed through a 2-layer LSTM model [29]. Then, a 300 dimensional representation is used to compound a joint embedding space with vectors of 300 dimensions from Word2Vec. The word vectors of class names are used for pairwise ranking loss performing with cosine similarity.

Generative Methods There are two generative approaches: generative zero-shot learning (GZSL) [57] and visual data synthesis via GAN (VDS) [100]. GZSL is based on modeling each action class using a probability distribution whose parameters are functions of the attribute vector representing that action class. Any action class in the visual space can be expressed as a linear combination of a set of basis vectors. A loss function is formulated by evaluating if the weighted combination is close to the maximum likelihood estimate (MLE) or the maximum-a-posteriori estimate (MAP) for each seen action class. A transductive method is used to overcome the domain shift by fine-tuning the parameters using unseen classes and Expectation-Maximization algorithm (EM). In this case, since the probability distribution is Gaussian, the EM procedure is equivalent to a Gaussian Mixture Model (GMM). In the second work, the semantic gap is bridged by synthesizing video features of unseen categories, making ZSL a typical supervised problem with synthetic features. Two strategies are used, a multi-level semantic inference and a matching-aware, to boost the video feature synthesis and to overcome information degradation issue, respectively. Generative Adversarial Networks (GANs), more specifically InfoGAN [10] and Wasserstain GAN (W-GAN) [4], are used. The InfoGAN aims to learn more interpretable representation by maximizing the mutual information between latent code and output. On the other hand, W-GAN introduces a distance to measure the similarity between real and fake distribution, which makes the training stage more stable.

Other Approaches Zhu et al. [102] proposed an universal representation method (URM) for ZSL cross-dataset. In their work, the aim is to obtain a representation that can generalize to a more realistic cross-dataset unseen action recognition. It is proposed a pipeline that incorporates deep feature extraction, generalized multiple-instance learning, universal representation learning, and semantic domain adaptation to overcome the domain shift, which is more severe in the cross-dataset setting. Finally, Roitberg et al. [74] presented a method to identify if an input example is known or unknown based on a voting scheme, referred to as Informed Democracy (ID). This method leveraged the estimated uncertainty of the individual classifiers in their predictions to measure the novelty of a new input sample. They argue that, in an open world scenario, an action recognition model should be able to handle: (i) the standard classification of previously seen categories; (ii) knowledge transfer for generalization to new unseen classes; and (iii) knowing how to discriminate between the two cases automatically. A better way to asses the neural network confidence is to predict the probability distribution with Bayesian neural networks (BNN). Hence, they adapted zero-shot action recognition models to the open set scenario, where a testing sample may originate from either known or novel categories.

4. Benchmark Datasets

The first popular video benchmarks were small, with approximately 10k videos [8], as shown in Table 4. Larger and complex datasets are available since 2011, such as HMDB51 [43], UCF101 [79], ActivityNet [28] and, more recently, Kinetics [38].

KTH [76] is a dataset with six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed by 25 different people in four different scenarios (outdoors, outdoors with scale variation, outdoors with different clothes, and indoors). The database contains 2391 sequences taken over homogeneous backgrounds with a static camera with 25 frames per second (fps), i.e. the frame rate. This dataset is no longer challenging and has not been used to evaluate modern ZSL methods. Another simple dataset is the Weizmann [5] with nine types of actions (running, walking, jumping-jack, jumping-forward-on-two-legs, jumping-in-place-on-two-legs, galloping-sideways, waving-two-hands, waving-one-hand, and bending) performed by nine different people in low-resolution videos (180×155) with 25 fps.

KTH and Weizmann contain a single staged actor with no occlusion and low clutter. They present video clips with controlled illumination and camera position so that they are not quite representative of the complexity of the real-world scenario and are not used recently. To solve these limitations, Kuehne et al. [43] presented the HMDB51 dataset with videos from many sources such as digitized movies, Prelinger archive, YouTube, and Google videos. This dataset contains 51 actions grouped into 5 categories (general facial actions, facial actions with object manipulation, general body movements, body movements with object interaction, and body movements for human interaction). The height of all the frames is scaled to 240 pixels, and so the width is rescaled, keeping the original aspect ratio. The frame rate is converted to 30 fps in order to ensure consistency in the entire dataset. Due to the complexity of the videos and to the significant number of videos per class, this dataset is widely used for evaluation.

There are three datasets provided by the University of Central Florida (UCF) that are used in Zero-Shot Action Recognition: UCFSports [71], UCF50 [70] and UCF101 [79]. In these datasets, the complexity grows because the videos are taken from the Web and they contain random camera motion, poor lighting conditions, clutter, as well as changes in scale, appearance and viewpoints, and occasionally no focus on the actions of interest [70]. UCFSports, for example, contains 10 actions (diving, golf swing,

Table 4
Datasets used in experiments sorted by year of creation. The following aspects are considered: Actions (AC), Activities (AT), and M (actions, activities, events, and objects).

Datasets	AC/AT	Year	#Videos	#Classes	Used in papers
KTH [76]	AC	2004	2391	6	[51]
Weizmann [5]	AC	2005	81	9	[51], [69]
UCFSports [71]	AC	2008	150	10	[69], [32]
UIUC [82]	AT	2008	532	14	[51]
Olympic Sports [58]	AT	2010	800	16	[51], [96], [95], [97], [68], [57]
UCF50 [70]	AC	2010	6676	50	[69]
CCV [34]	M	2011	9317	20	[96], [97]
HMDB51 [43]	AC	2011	7000	51	[94], [96], [32], [95], [3], [97], [88], [90], [68] [57], [64], [102], [74], [27]
UCF101 [79]	AC	2012	13320	101	[94], [39], [23], [96], [32], [95], [3], [97], [88], [90], [68], [57], [64], [102], [74], [27]
MPII Cooking Composites [73, 72]	AT	2012	256	41	[73], [72]
Thumos14 [30]	AC	2014	1574	101	[32]
Breakfast [42]	AT	2014	1989	10	[89]
ActivityNet [28]	AT	2015	27801	203	[99], [64]
Charades [77]	AT	2016	9848	157	[89]
Kinetics [38]	AC	2017	306245	400	[27]
MLB-YouTube [62]	AT	2018	4290	8	[64]

kicking, lifting, riding horse, running, skateboarding, swing-bench, swing-side, and walking) distributed in 150 video sequences with a resolution of 720×480 and 10 fps. This dataset was collected from various sports featured on broadcast television channels, such as BBC and ESPN. On the other hand, UCF50, an extension of the UCF11 dataset [50], contains 50 categories with a minimum of 100 videos for action class and a total of 6676. Finally, UCF101 [79] has 101 action classes with a total of 13320 videos with frame resolution standardized to 25 fps and resolution to 320×240 and stored in `avi` format. The action categories are divided into five types (human-object interaction, body-motion only, human-human interaction, playing musical instruments, and sports) and grouped into 25 groups where each group consists of 4-7 videos of an action. This great variation of action types and the largest amount of examples make this dataset widely used for validations, as well as HMDB51.

Olympic Sports [58] is a complex dataset of activities collected from YouTube sequences. There are 16 activities with 50 sequences per class, and the complex motions go beyond simple punctual or repetitive actions in contrast to UCFSports [71], which contains periodic or simple actions such as walking, running, golf-swing or ball-kicking. Although proposed for activity recognition, this dataset was used in approaches that focus on action recognition [51, 96, 95, 97, 68, 57], demonstrating that the complexity of methods makes them able to work on simple activities. Columbia Consumer Videos (CCV) is a dataset introduced by Jiang et al. [34] and includes 9317 unconstrained videos from the Web, preserving the originality without post-editing. There are 20 semantic categories, including a broader set ranging from events, objects, to scenes annotated using the Amazon Mechanical Turk platform. The number of videos from each category varies from 200 to 800. This dataset is used only in few works [96, 97] because there are examples of actions, activities, objects, and events, being more indicated to video description or retrieval problems. Another limitation is the few number of actions. For example, if a standard protocol that divides in 50% as seen classes and the rest of unseen classes are performed, the result is a restrict visual space and poor global performance.

MPII Cooking Composites and Breakfast are datasets that contain only cooking activities. MPII Cooking Composites contains 41 basic cooking activities with varying length from 1 to 41 minutes distributed on 256 videos. However, this dataset was used only in the same work where it was introduced. Likewise, Breakfast [42] is a large dataset of daily cooking activities, including a total of 52 participants performing 10 activities in 18 real-life kitchens. The resolution is 320×240 pixels with 15 fps. This dataset was used in a work that explores multi-label zero-shot action recognition [89] because there are 49 action classes annotated[2] in the clips and more than one action per clip. Charades dataset [77] is also used in [89] and has activities composed of more than one action. Charades is a challenging dataset built with the collaboration of 267 persons from three continents by using the Amazon Mechanical Turk platform. The objective is to collect videos of common daily activities performed in their homes – especially, examples that are not easy to find on YouTube, Movies, or TV broadcasts. The dataset has 9848 annotated videos representing 157 actions with 30 seconds of duration each. However, as most works do not explore multi-label classification, these datasets are not suitable for evaluation.

The ActivityNet dataset was introduced by Heilbron et al. [28] and is a large-scale benchmark for human activity understanding. There is a range of complex human activities

[2] Actions that compound the ten cooking activities.

that are of interest to people in their daily living. More precisely, 203 activity classes with an average of 137 untrimmed videos per class and a total of 27 801 videos. These videos were collected from the Internet, exploring a large amount of video data on online repositories such as YouTube. Around 50% of the videos have a resolution of 1280×720, whereas the majority has 30 fps. This dataset is little explored, possibly due to its high complexity compared to their amount of videos per class (193 on average). It is used in recent works [99, 64] that explore multi-modal learning combining visual features with textual descriptions. In the Kinetics dataset [38], which is the most extensive collection of human actions available to benchmark, there are 400 complex human action classes from different YouTube videos with at least 400 video clips for each action. The clips are about 10 seconds long, variable resolutions and frame rates. This dataset can be considered the successor of HMDB51, UCF101, and ActivityNet (trimmed version) because it is more suitable for training deep networks from scratch. The HMDB51 and UCF101 datasets are not large enough or have sufficient variation to learn and evaluate the current generation of human action classification models based on deep learning, and this limitation is more evident in zero-shot action recognition.

As shown in Table 4, there is a group of datasets used only once, such as UIUC [82], Thumos14 [30], and MLB-YouTube [62]. The UIUC dataset is presented in [82]. It consists of 532 high-resolution sequences of 14 activities performed by 8 actors in a single view. The Thumos14 dataset was proposed in the Thumos Challenge context [16]. In this dataset, there are temporally untrimmed videos and background videos, that is, with a similar background but without actions in the scene. The 101 action classes are performed in realistic settings and distributed in 1574 video clips. MLB-YouTube [62] is a dataset with activities collected from broadcast baseball videos with a focus on fine-grained activity recognition. More precisely, it is composed of 20 baseball games (42 hours) from the 2017 MLB post-season available on YouTube. In this dataset, the structure of the scene is very similar among activities; often, the only difference is the motion of a single person. Additionally, there is a single camera viewpoint to determine the activity. Because of its objective, this dataset has limited potential in ZSL. A complete description of most of datasets can be found in [9] and [36].

5. Experimental Protocols and Performance Analysis

There are many experimental protocols to perform ZSL in videos. To identify the most relevant ones, we select only the works that use the same benchmark datasets. Thus, only works that use the HMBD51, UCF101, and Olympic Sports datasets were selected because, as shown in Table 4, there are more than three works for comparison. Table 5 reports the results of selected works, a complete discussion comprising the proportions and amount of runs used in experiments, a comparison of performance in inductive and inductive settings, and the identification of the most successful methods. It can be observed that the approaches usually are evaluated using a common strategy. Initially, the classes of the dataset are randomly split into two disjoint sets called seen (source) and unseen (target) with different proportions (90%/10%, 80%/20% and 50%/50%). This procedure is repeated many times (3, 5, 10, 30, 50) and, in none work, the chosen proportions and/or the number of runs are justified.

We present in Table 5 all combinations of splits and runs reported in each selected work with correspondent mean accuracy and standard deviation. Only in two works, the standard deviations are unavailable.

The motivations for the use of 90%/10%, 80%/20% or 50%/50% splits in each dataset is not clear. It is reasonable to think in terms of the size of the training split. That is, to evaluate if the method presents better results in the presence of more training information. However, at the same time that they use more information to learn, there are fewer examples to classify, and the results tend to be better. On the other hand, in a configuration of 50%/50%, the results tend to be worst because there are more examples to classify and less information available to learn the models. This behavior is clearly identified in Table 5. In large scale datasets such as HMDB51, UCF101 or Kinetics, with 50%/50% configuration, it is possible to obtain a relevant amount of videos for both, to learn and classify. Thus, this configuration is widely used.

Large scale datasets are necessary to learn more discriminative models, but they bring a problem, i.e., the amount of all possible combinations of splits of classes for each experiment is enormous, and it is impractical to perform experiments with all possible combinations.

Therefore, to use random splits is a valid strategy, but it is necessary to consider that the experiment is stochastic and that 5 or 10 random splits can be an insufficient sampling compared to all possible combinations. For example, considering the 90/10 split, how statistically significant is a result obtained with 3 or 5 random splits? At the same time, how feasible is to perform the experiments using much more random splits? Hence, we only compare the results of experiments performed at least 30 times because, with this amount of runs, it is possible to provide an interval estimation for mean accuracy if it is considered a normal distribution. In this case, the population standard deviation σ can be estimated by sample standard deviation s, and the mean accuracy of population is given as $\mu \approx \bar{x} \pm E$, where $E \approx z_{95\%} \frac{s}{\sqrt{n}}$ and represents the estimation error with 95% of confidence for n samples (the number of runs/splits). In Table 5, when it is impossible to estimate the mean accuracy with 1% of estimation error it is marked with * and, when it is impossible with 2% it is marked with †.

In Table 5, we identify the major results in each dataset and setting, highlighting them. For HMDB51 in inductive setting, the best estimated performance is 25.3 ± 2.0 from [100]. On the other hand, all works have statistically equivalent results in transductive setting, that is, there is

Table 5
ZSL performance on the HMDB51, UCF101, and Olympic Sports datasets. The results are presented with the same decimal places to mean accuracy ($\bar{x}$) and standard deviation (s) as their original paper: inductive setting (I) and transductive setting (T). * indicates that in this experiment is not possible to estimate $\mu = \bar{x} \pm 1.0$ with 95% of confidence. † indicates that in this experiment is not possible to estimate $\mu = \bar{x} \pm 2.0$ with 95% of confidence.

%	#	HMDB51		UCF101		Olympic Sports		Approach
		I	T	I	T	I	T	
90/10	3	51.85	–	49.44	–	–	–	Hahn et al. [27]
	5	–	–	77.52 ± 6.01	–	–	–	Gan et al. [23]
80/20	3	38.16	–	37.39	–	–	–	Hahn et al. [27]
	10	–	–	33.5 ± 3.5	–	–	–	Kodirov et al. [39]
	30	–	–	$\mathbf{51.1 \pm 1.2}$	$\mathbf{66.9 \pm 1.9}$	–	–	Wang and Chen [90]
50/50	3	24.10	–	21.96	–	–	–	Hahn et al. [27]
	5	19.7 ± 1.6	24.8 ± 2.2	18.3 ± 1.7	22.9 ± 3.3	44.3 ± 8.1	56.6 ± 7.7	Xu et al. [95]
		21.80 ± 0.87	26.13 ± 1.29	24.38 ± 1.00	32.00 ± 2.30	–	–	Wang and Chen [88]
	10	14.38	22.41	12.01	35.17	–	–	Alexiou et al. [3]
		21.03 ± 2.07	–	17.85 ± 1.95	–	–	–	Roitberg et al. [74]
		–	–	14.0 ± 1.8	–	–	–	Kodirov et al. [39]
		22.6 ± 1.2	–	15.1 ± 1.7	–	59.8 ± 5.6	–	Qin et al. [68]
	30	$18.0 \pm 3.0^{*}$	$21.2 \pm 3.0^{*}$	12.7 ± 1.6	18.6 ± 2.2	–	–	Xu et al. [94]
		19.28 ± 2.1	$20.67 \pm 3.1^{*}$	22.74 ± 1.2	$24.48 \pm 2.9^{*}$	$50.41 \pm 11.2^{*}$	$57.88 \pm 14.1^{\dagger}$	Mishra et al. [57]
		20.6 ± 0.8	22.3 ± 1.1	$\mathbf{26.4 \pm 0.6}$	$\mathbf{35.1 \pm 1.1}$	–	–	Wang and Chen [90]
	50	21.6 ± 2.6	23.7 ± 3.4	20.1 ± 1.8	23.9 ± 2.7	$47.7 \pm 8.4^{\dagger}$	$51.1 \pm 8.7^{\dagger}$	Xu et al. [96]
		14.5 ± 2.7	$24.1 \pm 3.8^{*}$	11.7 ± 1.7	22.1 ± 2.5	$51.7 \pm 11.3^{\dagger}$	$53.2 \pm 11.6^{\dagger}$	Xu et al. [97]
		$\mathbf{25.3 \pm 4.5}^{*}$	–	$\mathbf{28.8 \pm 5.7}^{*}$	–	–	–	Zhang and Peng [100]

no significant difference among the results at 95% of confidence.

Considering the UCF101, in inductive setting, there are two statistically equivalent results 26.4 ± 1.0 [90] and 28.8 ± 2.0 [100]. In transductive setting, the result achieved by the approach proposed in [90] 35.1 ± 1.0 is much better than the others in this dataset.

Finally, in Olympic Sports, the results present high standard variation so that the estimation error to obtain the same confidence level makes all estimated results to be equivalent. With this analysis, two approaches deserve attention, BiDiLEL [90] and VDS [100].

The BiDiLEL model is based on combinations of features. In the visual extraction step, C3D deep features are combined with IDT handcrafted features and in the semantic embedding step, it was used a combination of attributes and Word2Vec in UCF101, and only Word2Vec in HMDB51.

The C3D model is a ConvNet pre-trained in Sports-1M Dataset [81]. We believe that to use pre-trained deep models (e.g., C3D or I3D) in practice means intrinsically to use a cross-dataset approach and, if the same classes that are used to train the deep models were also used to test the ZSL methods, the disjunction between seen and unseen classes would not be respected because the deep model acquires the knowledge from classes that should be unseen. A similar analysis was presented in Martinez et al. [53], but in the context of cross-dataset studies. They argue that when external datasets are involved, one has to ensure that the terms of ZSL are still met and the seen and unseen categories are disjoint. It is not sufficient to remove only identical classes because there are similar classes such as `Basketball Shooting` (UCF101) × `Basketball` or `Basketball 3X3` or `wheelchair basketball` (Sports-1M).

A protocol to remove semantically similar classes from source category (seen) using the cosine similarity measure and a threshold parameter was defined by Martinez et al. [53]. However, we conjecture that, when pre-trained deep models are used, it is necessary to remove the similar classes from the target and not from the source. For example, in [90], we need to compare the classes between UCF101 and Sports-1M (used for training C3D model). It is observed that they share 23 identical classes and 17 similar classes[3]. Since that work uses the same 30 splits employed by Xu et al. [94], these shared classes were not removed from the target before the experiment and the restriction of ZSL is not preserved. To keep the ZSL disjunction between the training and testing sets, it is necessary to use only unknown classes in the testing time, excluding all classes that were used for training the deep model. In this case, UCF101 would have 61 possible classes for testing. This new restriction means that it may be impracticable to use UCF101 or HMDB51 dataset when pre-trained deep models such as C3D or I3D are used. Fortunately, there are new versions of Kinetics with 600 and 700 actions [7], allowing to evaluate new ZSL models based on I3D (Kinetics-400), for example. Xian et al. [93] also argued that it is necessary to preserve the disjunction between training and validation sets for parameter tuning, in their work about ZSL applied to object recognition context.

VDS approach utilizes GANs to generate more training data from the training set with the same statistic prop-

[3] Manual checks.

erties. This strategy brings high discriminative power and suffers much less information degradation than other methods. GANs suffer from instability in training because they are unrestricted and uncontrollable. Much effort has been made in recent years to tackle this problem. Therefore, these deep models have the potential to increase the power of representation and then leverage the results in the next years.

Evaluating the impact of transductive setting on performance, in Table 5, it is observed that this configuration presents better results than inductive setting in all works. This is due to the effectiveness of methods as self-training and hubness correction to alleviate the domain shift problem. Another important consideration is that the use of attributes generally results in better results than word vectors. Furthermore, as discussed earlier, this strategy is not scalable and became impracticable in real-world scenarios. Finally, it is necessary to explore more strategies to perform semantic embedding. As discussed earlier, there are few techniques explored being the Word2Vec the most used. However, GloVe emerges as a potential alternative having been used in the VDS approach with promising performance.

6. Open Issues and Future Work

Although much progress has been made in zero-shot action recognition in the last years, their performance is not often compared to conventional supervised learning. For example, while Carreira and Zisserman [8] obtained 98% and 80.9% of accuracy on UCF101 and HMDB51 datasets using the supervised learning paradigm, respectively, Hahn et al. [27] obtained 21.96% and 24.1% (50%/50% seem/unseen classes), respectively, using the ZSL paradigm and the same I3D model. Even if we compare with the bests results in ZSL, that is, Zhang and Peng [100] $\sim$ 28.8 $\pm$ 2.0 and $\sim$ 25.3 $\pm$ 2.0 using generative models or Wang and Chen [90] $\sim$ 35.1 $\pm$ 1.0 and $\sim$ 22.3 $\pm$ 1.0, using combinations for semantic and visual features in a transductive setting on the UCF101 and HMDB51 datasets, respectively, we can observe that still are a lot of room to achieve comparable or useful performance, and it requires to resolve or ameliorate the classical ZSL problem, the semantic gap.

Describing actions is much more challenging than describing nouns. Therefore, we believe that are few variation of strategies for semantic embedding. Most works explore only the Word2Vec algorithm without modifications or new techniques. A good example that explores this algorithm with a new approach was proposed by Alexiou et al. [3]. Although the result is not globally better than other approaches, their work demonstrated that to use synonyms leverage the performance of several ZSL methods. Even though still little explored, GloVe is a promising approach. We believe that it is necessary to incorporate more recent advances in language processing, for example, geometric deep learning with Graph Convolutional Networks [98], which presents better results than fastText [6] which, in turn, is more discriminative than GloVe and Word2Vec. On the visual extraction perspective, with the recent advances of deep learning methods, it seems to be imperative their use, mainly the pre-trained models, recurrent networks as well the generative models. We identify that multi-modal learning is a promising approach to solve the semantic gap. However, there are few studies with this perspective. It is intuitive that it is easier to recognize actions using the detection of objects in scene, or including more information from still images or texts because the features tend to be more descriptive. These possibilities need to be better explored such that we can create robust frameworks for zero-shot action recognition.

As previously discussed, it is necessary to establish a common protocol and mainly a straight definition of the use of seen classes to fine tuning the parameters of deep models. It lacks a work where several experimental protocols are applied to state-of-the-art approaches, such that the community is able to replicate and compare the experiments. For example, we believe that is more suitable for evaluation an experimental protocol where there is no need to randomly split the datasets. What criteria could be adopted to define which classes are used in the training and which ones are used in the testing? Is it possible to create a general split? If not, what standard should be adopted to create random splits and how many runs would be required? Answering these questions is critical to the progress of zero-shot action recognition in videos.

In this survey, only approaches for action recognition were selected. However, as shown in Tables 3 and 4, there are few approaches focusing on activity recognition and generally these approaches do not use the same datasets in their experimental evaluation. Thus, it is difficult to fairly compare the results.

Another important issue is the Few-Shot Learning (FSL) or generalized zero-shot learning. As ZSL suffers from a lack of common protocol to evaluate the methods, this problem is more dramatic in the FSL. The problem begins in the definition. What is understood as FSL? Which criteria define this problem? The approaches usually explore this problem as a special case, not focusing on the work. At the same time, FSL is much important since it is a problem closer to real-world situations.

We finish this section pointing an interesting and few explored problem that is to recognize if an example is known or unknown and, from this information, to decide which approach is more suitable for attempting to recognize it. Actually, we found only the Informed Democracy method [74] trying to solve this question.

7. Conclusions

We presented a survey of existing ZSL methods for action recognition in videos that describes several techniques used to perform visual and semantic extraction. We also described several methods that employ these features and bridge the semantic gap. A comprehensive description of databases and their main applications is presented.

An analysis of the results was presented jointly with a discussion about the experimental protocols, from where we can highlight the following conclusions: (i) it is very difficult to compare the experimental results because many of them use only one or two specific datasets (e.g., KTH, Weizmann, Charades, Breakfast, MPII Cooking Composites, UCF50) and do not follow the same protocol due to differences in split sizes or random runs, for example. To provide a comparison, we estimate the mean accuracy of each experiment using the available information, and we were able to analyze experiments with at least 30 runs, (ii) the best results use combinations of features and generative models [57, 96, 90, 100], (iii) by comparing the inductive against transductive setting, the results show that the latter always presented better performance, and (iv) it is necessary to investigate several configurations of protocols using the state-of-the-art methods to identify the better configurations and the criteria to generate the splits whether fixed or random.

Acknowledgments

This work was supported by the Federal Institute of Paraná, Federal University of Paraná and by grants from the National Council for Scientific and Technological Development (CNPq) (grant numbers 428333/2016-8, 313423/2017-2).

References

[1] Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B., 2015. Evaluation of output embeddings for fine-grained image classification, in: IEEE Computer Vision and Pattern Recognition, pp. 2927 – 2936.

[2] Al-Naser, M., Ohashi, H., Ahmed, S., Nakamura, K., Akiyama, T., Sato, T., Nguyen, P., Dengel, A., 2018. Hierarchical model for zero-shot activity recognition using wearable sensors, in: 10th International Conference on Agents and Artificial Intelligence, pp. 478–485.

[3] Alexiou, I., Xiang, T., Gong, S., 2016. Exploring synonyms as context in zero-shot action recognition, in: IEEE International Conference on Image Processing, pp. 4190–4194.

[4] Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks, in: Precup, D., Teh, Y.W. (Eds.), Proceedings of the 34th International Conference on Machine Learning, PMLR, International Convention Centre, Sydney, Australia. pp. 214–223.

[5] Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R., 2005. Actions as space-time shapes, in: Tenth IEEE International Conference on Computer Vision, pp. 1395–1402.

[6] Bojanowski, P., Grave, E., Joulin, A., Mikolov, T., 2016. Enriching word vectors with subword information. CoRR abs/1607.04606. arXiv:1607.04606.

[7] Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., Zisserman, A., 2018. A short note about kinetics-600. CoRR abs/1808.01340. arXiv:1808.01340.

[8] Carreira, J., Zisserman, A., 2017. Quo vadis, action recognition? a new model and the kinetics dataset, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4724–4733.

[9] Chaquet, J.M., Carmona, E.J., FernÃąndez-Caballero, A., 2013. A survey of video datasets for human action and activity recognition. Computer Vision and Image Understanding 117, 633 – 659.

[10] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P., 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, Curran Associates Inc., USA. pp. 2180–2188.

[11] Chung, J., Gülçehre, Ç., Cho, K., Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555. arXiv:1412.3555.

[12] Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 886–893 vol. 1.

[13] Deng, J., Dong, W., Socher, R., Li, L., and, 2009. Imagenet: A large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255.

[14] Dinu, G., Lazaridou, A., Baroni, M., 2014. Improving zero-shot learning by mitigating the hubness problem. International Conference on Learning Representations , 1–10.

[15] Efros, A., Berg, A.C., Mori, G., Malik, J., 2003. Recognizing action at a distance, in: Ninth IEEE International Conference on Computer Vision, pp. 726–733 vol.2.

[16] European Conference on Computer Vision, 2014. THUMOS Challenge 2014. https://www.crcv.ucf.edu/THUMOS14. Online; accessed 03 September 2019.

[17] Fellbaum, C. (Ed.), 1998. WordNet: an electronic lexical database. MIT Press.

[18] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D., 2010. Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 1627–1645.

[19] Fu, Y., Hospedales, T.M., Xiang, T., Fu, Z., Gong, S., 2014. Transductive multi-view embedding for zero-shot recognition and annotation, in: European Conference on Computer Vision, Springer International Publishing, Cham. pp. 584–599.

[20] Fu, Y., Hospedales, T.M., Xiang, T., Gong, S., 2012. Attribute learning for understanding unstructured social activity, in: European Conference on Computer Vision, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 530–543.

[21] Fu, Y., Hospedales, T.M., Xiang, T., Gong, S., 2014. Learning multimodal latent attributes. IEEE Transactions on Pattern Analysis and Machine Intelligence 36, 303–316.

[22] Fu, Y., Xiang, T., Jiang, Y., Xue, X., Sigal, L., Gong, S., 2018. Recent advances in zero-shot recognition: Toward data-efficient understanding of visual content. IEEE Signal Processing Magazine 35, 112–125.

[23] Gan, C., Lin, M., Yang, Y., Zhuang, Y., Hauptmann, A.G., 2015. Exploring semantic inter-class relationships (sir) for zero-shot action recognition, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI Press. pp. 3769–3775.

[24] Gan, C., Lin, M.C., Yang, Y., de Melo, G., Hauptmann, A.G., 2016. Concepts not alone: Exploring pairwise relationships for zero-shot video activity recognition, in: AAAI, pp. 3487 – 3493.

[25] Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., Saenko, K., 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition, in: IEEE International Conference on Computer Vision, pp. 2712–2719.

[26] Guo, G., Lai, A., 2014. A survey on still image based human action recognition. Pattern Recognition 47, 3343 – 3361.

[27] Hahn, M., meerahahn, da Silva, A.B., Silva, A.M.P., 2019. Action2vec: A crossmodal embedding approach to action learning. CoRR abs/1901.00484.

[28] Heilbron, F.C., Escorcia, V., Ghanem, B., Niebles, J.C., 2015. Activitynet: A large-scale video benchmark for human activity understanding, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 961–970.

[29] Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Computation 9, 1735–80.

[30] Idrees, H., Zamir, A.R., Jiang, Y.G., Gorban, A., Laptev, I., Sukthankar, R., Shah, M., 2017. The thumos challenge on action recognition for videos âĂIJin the wildâĂİ. Computer Vision and Image Understanding 155, 1 – 23.

[31] Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: Proceed-

ings of the 32Nd International Conference on Machine Learning - Volume 37, JMLR.org. pp. 448–456.

[32] Jain, M., v. Gemert, J.C., Mensink, T., Snoek, C.G.M., 2015. Objects2action: Classifying and localizing actions without any video example, in: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4588–4596.

[33] Ji, S., Xu, W., Yang, M., Yu, K., 2013. 3D Convolutional Neural Networks for Human Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 35, 221–231.

[34] Jiang, Y.G., Ye, G., Chang, S.F., Ellis, D.P.W., Loui, A.C., 2011. Consumer video understanding: a benchmark database and an evaluation of human and machine performance, in: Annual ACM International Conference on Multimedia Retrieval, pp. 1 – 8.

[35] K. Aggarwal, J., S. Ryoo, M., 2011. Human activity analysis: A review. ACM Computing Surveys 43, 16.

[36] Kang, S., Wildes, R.P., 2016. Review of action recognition and detection methods. CoRR abs/1610.06906.

[37] Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L., 2014. Large-scale video classification with convolutional neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732.

[38] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, A., Suleyman, M., Zisserman, A., 2017. The kinetics human action video dataset. CoRR abs/1705.06950, 1–22.

[39] Kodirov, E., Xiang, T., Fu, Z., Gong, S., 2015. Unsupervised Domain Adaptation for Zero-Shot Learning, in: IEEE International Conference on Computer Vision, pp. 2452–2460.

[40] Kong, Y., Fu, Y., 2018. Human action recognition and prediction: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) .

[41] Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: 25th International Conference on Neural Information Processing Systems - Volume 1, Curran Associates Inc., USA. pp. 1097–1105.

[42] Kuehne, H., Arslan, A.B., Serre, T., 2014. The language of actions: Recovering the syntax and semantics of goal-directed human activities. IEEE Conference on Computer Vision and Pattern Recognition , 780–787.

[43] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T., 2011. HMDB: A large video database for human motion recognition, in: International Conference on Computer Vision, pp. 2556–2563.

[44] Lampert, C.H., Nickisch, H., Harmeling, S., 2009. Learning to detect unseen object classes by between-class attribute transfer, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 951–958.

[45] Lampert, C.H., Nickisch, H., Harmeling, S., 2014. Attribute-based classification for zero-shot visual object categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence 36, 453–465.

[46] Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B., 2008. Learning realistic human actions from movies, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.

[47] Li, L.J., Su, H., Fei-Fei, L., Xing, E.P., 2010. Object bank: A high-level image representation for scene classification and semantic feature sparsification, in: Lafferty, J.D., Williams, C.K.I., Shawe-Taylor, J., Zemel, R.S., Culotta, A. (Eds.), Advances in Neural Information Processing Systems 23. Curran Associates, Inc., pp. 1378–1386.

[48] Li, Y., Hu, S., Li, B., 2016. Recognizing unseen actions in a domain-adapted embedding space, in: IEEE International Conference on Image Processing, pp. 4195–4199.

[49] Lin, Z., Jiang, Z., Davis, L.S., 2009. Recognizing actions by shape-motion prototype trees, in: IEEE 12th International Conference on Computer Vision, pp. 444–451.

[50] Liu, J., Jiebo Luo, Shah, M., 2009. Recognizing realistic actions from videos âĂIJin the wildâĂİ, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1996–2003.

[51] Liu, J., Kuipers, B., Savarese, S., 2011. Recognizing human actions by attributes, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Washington, DC, USA. pp. 3337–3344.

[52] Liu, K., Liu, W., Ma, H., Huang, W., Dong, X., 2018. Generalized zero-shot learning for action recognition with web-scale video data. World Wide Web 22, 807–824.

[53] Martinez, M., Haurilet, M., Stiefelhagen, R., 2018. Towards a fair evaluation of zero-shot action recognition using external data, in: European Conference on Computer Vision Workshops, pp. 1 – 9.

[54] Mettes, P.S.M., Snoek, C.G.M., 2017. Spatial-aware object embeddings for zero-shot localization and classification of actions, in: IEEE International Conference on Computer Vision, pp. 1 – 10.

[55] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., 2013a. Distributed representations of words and phrases and their compositionality, in: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Curran Associates Inc., USA. pp. 3111–3119.

[56] Mikolov, T., Yih, W.t., Zweig, G., 2013b. Linguistic regularities in continuous space word representations, in: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Atlanta, Georgia. pp. 746–751.

[57] Mishra, A., Verma, V.K., Reddy, M.S.K., Subramaniam, A., Rai, P., Mittal, A., 2018. A generative approach to zero-shot and few-shot action recognition. IEEE Winter Conference on Applications of Computer Vision , 372–380.

[58] Niebles, J.C., Chen, C.W., Fei-Fei, L., 2010. Modeling temporal structure of decomposable motion segments for activity classification, in: European Conference on Computer Vision,Computer Vision, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 392–405.

[59] O'Hara, S., Draper, B.A., 2011. Introduction to the bag of features paradigm for image classification and retrieval. CoRR abs/1101.3354. arXiv:1101.3354.

[60] Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359.

[61] Pennington, J., Socher, R., Manning, C.D., 2014. Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing, pp. 1532–1543.

[62] Piergiovanni, A., S. Ryoo, M., 2018. Fine-grained activity recognition in baseball videos. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops , 1853–1861.

[63] Piergiovanni, A.J., Fan, C., Ryoo, M.S., 2017. Learning latent subevents in activity videos using temporal attention filters, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI Press. pp. 4247–4254.

[64] Piergiovanni, A.J., Ryoo, M.S., 2018. Learning shared multimodal embeddings with unpaired data. CoRR abs/1806.08251.

[65] Poppe, R., 2010. A survey on vision-based human action recognition. Image and Vision Computing 28, 976 – 990.

[66] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M.P., Shyu, M.L., Chen, S.C., Iyengar, S.S., 2018. A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys 51, 92:1–92:36.

[67] Prest, A., Schmid, C., Ferrari, V., 2012. Weakly supervised learning of interactions between humans and objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 601–614.

[68] Qin, J., Liu, L., Shao, L., Shen, F., Ni, B., Chen, J., Wang, Y., 2017. Zero-shot action recognition with error-correcting output codes, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1042–1051.

[69] Qiu, Q., Jiang, Z., Chellappa, R., 2011. Sparse dictionary-based representation and recognition of action attributes, in: 2011 International Conference on Computer Vision, pp. 707–714.

[70] Reddy, K.K., Shah, M., 2013. Recognizing 50 human action categories of web videos. Machine Vision and Applications 24, 971–981.

[71] Rodriguez, M.D., Ahmed, J., Shah, M., 2008. Action mach a spatio-temporal maximum average correlation height filter for action recog-

nition, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.
[72] Rohrbach, M., Ebert, S., Schiele, B., 2013. Transfer learning in a transductive setting, in: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 26. Curran Associates, Inc., pp. 46–54.
[73] Rohrbach, M., Regneri, M., Andriluka, M., Amin, S., Pinkal, M., Schiele, B., 2012. Script Data for Attribute-Based Recognition of Composite Activities, in: 12th European Conference on Computer Vision, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 144–157.
[74] Roitberg, A., Al-Halah, Z., Stiefelhagen, R., 2018. Informed Democracy: Voting-based Novelty Detection for Action Recognition, in: British Machine Vision Conference, Newcastle upon Tyne, UK. pp. 1 – 14.
[75] Sadanand, S., Corso, J.J., 2012. Action bank: A high-level representation of activity in video, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1234–1241.
[76] SchÃijldt, C., Laptev, I., Caputo, B., 2004. Recognizing human actions: A local svm approach, in: International Conference on Pattern Recognition, pp. 32–36.
[77] Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A., 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding, in: Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham. pp. 510–526.
[78] Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv 1409.1556 , 1–14.
[79] Soomro, K., Roshan Zamir, A., Shah, M., 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. CoRR , 1–6.
[80] Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks, in: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, MIT Press, Cambridge, MA, USA. pp. 3104–3112.
[81] Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M., 2015. Learning spatiotemporal features with 3d convolutional networks, in: IEEE International Conference on Computer Vision, IEEE Computer Society, Washington, DC, USA. pp. 4489–4497.
[82] Tran, D., Sorokin, A., 2008. Human activity recognition with metric learning, in: European Conference on Computer Vision, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 548–561.
[83] Turaga, P., Chellappa, R., Subrahmanian, V.S., Udrea, O., 2008. Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video Technology 18, 1473–1488.
[84] Wang, H., KlÃďser, A., Schmid, C., Liu, C., 2011. Action recognition by dense trajectories, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3169–3176.
[85] Wang, H., KlÃďser, A., Schmid, C., Liu, C.L., 2013. Dense trajectories and motion boundary descriptors for action recognition. International Journal of Computer Vision 103, 60–79.
[86] Wang, H., Oneata, D., Verbeek, J., Schmid, C., 2016. A robust and efficient video representation for action recognition. Int. J. Comput. Vision 119, 219–238.
[87] Wang, H., Schmid, C., 2013. Action recognition with improved trajectories, in: 2013 IEEE International Conference on Computer Vision, pp. 3551–3558.
[88] Wang, Q., Chen, K., 2017a. Alternative semantic representations for zero-shot human action recognition, in: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part I, pp. 87–102.
[89] Wang, Q., Chen, K., 2017b. Multi-label zero-shot human action recognition via joint latent embedding. CoRR abs/1709.05107. arXiv:1709.05107.
[90] Wang, Q., Chen, K., 2017c. Zero-shot visual recognition via bidirectional latent embedding. Int. J. Comput. Vision 124, 356–383.
[91] Wang, W., Zheng, V.W., Yu, H., Miao, C., 2019. A survey of zero-shot learning: Settings, methods, and applications. ACM Trans. Intell. Syst. Technol. 10, 13:1–13:37.
[92] Wu, Z., Palmer, M., 1994. Verbs semantics and lexical selection, in: 32nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, Stroudsburg, PA, USA. pp. 133–138.
[93] Xian, Y., Schiele, B., Akata, Z., 2017. Zero-shot learning - the good, the bad and the ugly, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, pp. 3077–3086.
[94] Xu, X., Hospedales, T., Gong, S., 2015. Semantic embedding space for zero-shot action recognition, in: 2015 IEEE International Conference on Image Processing (ICIP), pp. 63–67.
[95] Xu, X., Hospedales, T., Gong, S., 2016. Multi-task zero-shot action recognition with prioritised data augmentation, in: 14th European Conference on Computer Vision, pp. 343–359.
[96] Xu, X., Hospedales, T., Gong, S., 2017a. Transductive zero-shot action recognition by word-vector embedding. International Journal of Computer Vision 123, 309–333.
[97] Xu, X., Hospedales, T., Gong, S., 2017b. Transductive zero-shot action recognition by word-vector embedding. International Journal of Computer Vision 123, 309–333.
[98] Yao, L., Mao, C., Luo, Y., 2018. Graph convolutional networks for text classification, in: AAAI, pp. 7370 – 7377.
[99] Zhang, B., Hu, H., Sha, F., 2018. Cross-modal and hierarchical modeling of video and text, in: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), Computer Vision – ECCV 2018, Springer International Publishing, Cham. pp. 385–401.
[100] Zhang, C., Peng, Y., 2018. Visual data synthesis via gan for zero-shot video classification, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization. pp. 1128–1134.
[101] Zhang, Z., Saligrama, V., 2016. Zero-shot recognition via structured prediction, in: European Conference on Computer Vision, Springer International Publishing, Cham. pp. 533–548.
[102] Zhu, Y., Long, Y., Guan, Y., Newsam, S.D., Shao, L., 2018. Towards universal representation for unseen action recognition, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pp. 9436–9445.
[103] Ziaeefard, M., Bergevin, R., 2015. Semantic human activity recognition: A literature review. Pattern Recognition 48, 2329 – 2345.