# Recognizing Activities via Bag of Words for Attribute Dynamics


Weixin Li[†] Qian Yu[§] Harpreet Sawhney[§] Nuno Vasconcelos[†]

[†]University of California, San Diego
La Jolla, CA 92093
{wel017, nvasconcelos}@ucsd.edu

[§]SRI International Sarnoff
Princeton, NJ 08540
{qian.yu, harpreet.sawhney}@sri.com



## Abstract

*In this work, we propose a novel video representation for activity recognition that models video dynamics with attributes of activities. A video sequence is decomposed into short-term segments, which are characterized by the dynamics of their attributes. These segments are modeled by a dictionary of attribute dynamics templates, which are implemented by a recently introduced generative model, the binary dynamic system (BDS). We propose methods for learning a dictionary of BDS's from a training corpus, and for quantizing attribute sequences extracted from videos into these BDS codewords. This procedure produces a representation of the video as a histogram of BDS codewords, which is denoted the bag-of-words for attribute dynamics (BoWAD). An extensive experimental evaluation reveals that this representation outperforms other state-of-the-art approaches in temporal structure modeling for complex activity recognition.*


## 1. Introduction

The recognition of human activities and events is an important problem for computer vision. Two lines of research have received substantial attention in this area. The first, motivated by the fact that an activity is naturally defined by an ordered set of short-term behaviors, aims to model the temporal composition of activities. This is usually done with low-level video representations. In fact, many methods have been proposed to model the temporal structure of low-level features extracted from video, *e.g.*, histograms of spatiotemporal filter responses. This includes

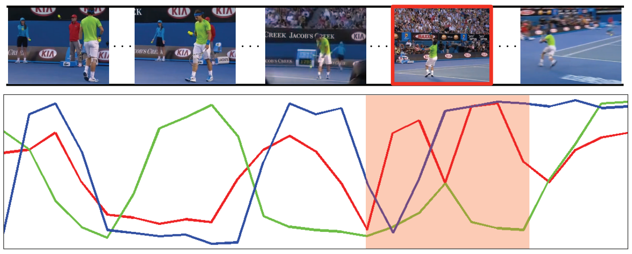

**Figure 1:** Challenges in modeling the dynamics of attributes of complex activities. (Top) YouTube video sequence annotated with "tennis-serve" activity. (Bottom) associated trajectory on a 3D attribute space (red for "arm-motion", green for "foot motion" and blue for "ball motion"). Note the complexity of the trajectory and the fact that only a short segment (red-shaded) is a staple of the action of interest.

both discriminative [11, 16, 7, 25] and generative models [12, 9, 4]. The second, inspired by recent advances in image analysis, is to represent activities as collections of semantic attributes [15, 23, 22, 6]. This entails an intermediate level of representation, where features are no longer visual, but identifiers of the occurrence of semantic concepts of interest, such as scene types, actions, objects, *etc*. This higher level of abstraction enables better generalization, facilitates semantic and contextual reasoning, and enables knowledge transfer from well-understood examples to unseen instances.

Advances along these two directions are complementary. While a detailed characterization of the temporal structure on top of low-level features is, in general, insufficient to characterize complex activities, the representation of video as an orderless set of attributes is incapable of fine-grained activity discrimination (*i.e.*, distinguishing between activities which express the same attributes in different orders). Recently, [14] has proposed to unify the two research directions, by modeling the temporal structure of the video projection in an attribute space. This was implemented by introducing a dynamic model, denoted binary dynamic system (BDS), which extends classical linear dynamic systems to binary observation spaces. While this model has been


This work was partially supported by the NSF award under Grant CCF-0830535 and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/NBC, or the U.S. Government.

shown to achieve state-of-the-art performance in standard benchmarks, it does not address two of the most significant challenges in the recognition of complex activities. The first is that such video rarely contains *only* the event of interest. In general, video sequences are only annotated with respect to a dominant event, or high-level subject, and not with respect to the footage that either precedes or trails it. The second is that a single model, such as the BDS, is unlikely to provide a good fit to the complex attribute space trajectories produced by the video. This is illustrated in Figure 1, which presents the trajectory of a video of the "tennis serve" activity in a space spanned by three closely-related attributes.

In this work, we propose to address these limitations with a new video representation, which is denoted the *bag-of-words for attribute dynamics* (BoWAD). This is an extension of the *bag-of-visual words* (BoVW), which has achieved great popularity for image classification [28]. Like the BoVW, the BoWAD is an histogram with respect to a dictionary of templates. However, rather than templates of visual appearance, it relies on *templates of attribute dynamics*. These templates are in fact generative models and, more precisely, *temporally localized* BDS's. In this way, an activity is represented as a collection of characteristic short-term behaviors, and no single BDS needs to model unduly complex attribute trajectories. We propose a procedure for learning a dictionary of BDS's, and for quantizing video with respect to this dictionary, and show that the representation achieves performance superior to that of state-of-the-art approaches of temporal structure modeling in challenging datasets.

## 2. Related Work

Over the last decade, the *bag-of-features* (BoF) has become a popular video representation for action recognition [27]. This consists of representing video as a collection of feature vectors. Several models exploiting the temporal structure of activities are based on this representation. For example, Laptev *et al.* [11] used a spatio-temporal binning pyramid to match vector-quantized histograms from different video regions. Niebles *et al.* [16] and Gaidon *et al.* [7] represented an activity with a small number of decomposable parts or atomic actions. Alternatives based on generative models have also been proposed. Laxton *et al.* [12] integrated confidences about objects and sub-actions over time, with dynamic Bayesian networks. Finally, dynamic systems have been used to represent the evolution of human activity, using different features (local binary patterns [9], tracked parts [13], or frame-wise motion histograms [4]).

Recently, image analysis research has shown that *semantics* or *attribute-based* representations can have substantial benefits over BoF, including better generalization and support for contextual reasoning [19, 10, 20]. This has motivated the application of these representations to action recognition. For example, Liu *et al.* [15] proposed the use of attributes as latent variables for support vector machines (SVMs) to recognize actions. Sadanand and Corso [23] have shown substantial improvements over standard benchmarks by using a bank of action detectors sampled broadly across semantic and viewpoint spaces. Rohrbach *et al.* [22] augmented video with text-script data and modeled activities as common sets of attributes, defined in terms of basic actions and objects. Finally, Li and Vasconcelos [14] introduced a model (BDS) of the temporal structure of attributes. This work suggests that the modeling of video trajectories in attribute space is crucial for the fine-grained understanding of human behavior .

In this work, we expand on the idea of [14], by learning dictionaries of models for attribute dynamics. This is related to the *bag-of-systems* framework of [21, 1], where a set of dynamic textures (DTs) [5] were used to characterize dynamic scenes. The main challenge of this dictionary leaning problem is the difficulty of identifying the "centroid" of a collection of *dynamic textures*, due to the non-Euclidean nature of the space of linear dynamic systems. [21] bypasses this problem with resort to a somewhat heuristic combination of multi-dimensional scaling and $k$-means (denoted MDS-$k$M); while [1] presents a procedure to directly average dynamic models in the parameter space, the approach only works for LDS's. We propose an alternative principled solution, which is specifically designed for clustering *attribute sequences*, and has a number of advantages over MDS-$k$M. These are shown to result in superior recognition accuracy.

## 3. The Bag of Words for Attribute Dynamics

In this section, we introduce a new representation for activity recognition, denoted the *bag-of-words for attribute dynamics* (BoWADs).

### 3.1. Words and Attributes

A popular representation for image classification is the *bag of visual words* (BoVW) [28], which has recently also become popular for action recognition [27]. This consists of representing an image as a BoF, learning a *dictionary* of representative feature vectors, which are denoted *visual words*, and using this dictionary to quantize the features extracted from an image to classify. The BoVW is the resulting histogram of visual word counts. This is frequently used as a feature vector for image or video classification. Despite the popularity of the BoVW, several works have demonstrated the benefits of alternative feature spaces, which encode higher-level semantics by representing images or video as collections of binary *attributes* [19, 10, 20, 15, 14].

Under this representation, activities are defined with respect to a set of $K$ attributes $\mathcal{C} = \{c_i\}_{i=1}^K$, inferred from video frames by a bank of attribute classifiers $\{\pi_i\}_{i=1}^K$. Pos-

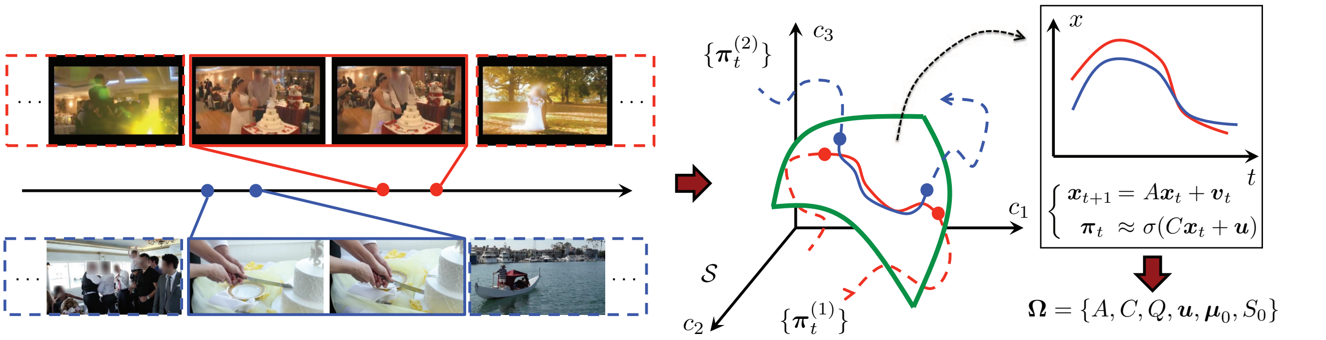


**Figure 2:** Learning a BDS. Video sequences (left) are encoded as trajectories in attribute space $\mathcal{S}$ (center). Sequences of similar semantics span similar trajectories. The BDS $\mathbf{\Omega}$ embeds a video trajectory into a low-dimensional space (shown in green), by binary PCA, and learns a Gauss-Markov process that describes the corresponding trajectory in the latent state space (right).

sible attributes include scene classes, objects, atomic actions, human-object interactions, *etc*. A video $\mathsf{v} \in \mathcal{X}$ is mapped into an *attribute space* $\mathcal{S}$ by a mapping

$$\boldsymbol{\pi} : \mathcal{X} \to \mathcal{S} = [0,1]^K, \tag{1}$$

where

$$\boldsymbol{\pi}(\mathsf{v}) = (\pi_1(\mathsf{v}), \cdots, \pi_K(\mathsf{v}))^T \tag{2}$$

is an attribute score vector. Component $\pi_i(\mathsf{v})$ is a confidence score quantifying the presence of the $i$-th attribute in $\mathsf{v}$. In this work, these scores are the *posterior probabilities* $\pi_c(\mathsf{v}) = p(c|\mathsf{v})$ of attribute $c$ given some low-level representation of video $\mathsf{v}$, *e.g.*, a BoF histogram of spatio-temporal descriptors.

### 3.2. Attribute-based Activity Recognition

In [15] a vector of attribute scores $\boldsymbol{\pi}(\mathsf{v})$ is computed for the whole video sequence $\mathsf{v}$. This *holistic* attribute representation disregards the temporal structure of the different attributes. While it can distinguish activities that lie on different regions of $\mathcal{S}$, it cannot disambiguate activities that contain similar attributes but with different temporal structure. This problem can be overcome by applying the attribute classifiers to video segments $\mathsf{v}_t$ extracted with a sliding window. As illustrated in Figure 2, this produces a sequence of attribute score vectors $\{\boldsymbol{\pi}_t\}_{t=1}^{\tau}$, where $\boldsymbol{\pi}_t = \boldsymbol{\pi}(\mathsf{v}_t)$. In summary, a video sequence is modeled as a trajectory in $\mathcal{S}$ and sequences of similar semantics span similar trajectories.

Li and Vasconcelos proposed to model a video trajectory in $\mathcal{S}$ with a binary dynamic system (BDS) [14], defined by

$$\begin{cases} \boldsymbol{x}_{t+1} &= A\boldsymbol{x}_t + \boldsymbol{v}_t, & \text{(3a)} \\ \boldsymbol{y}_t &\sim B(\boldsymbol{y}; \sigma(C\boldsymbol{x}_t + \boldsymbol{u})), & \text{(3b)} \end{cases}$$

where $\boldsymbol{x}_t \in \mathbb{R}^L$ ($L$ is the dimension of the latent space) and $\boldsymbol{y}_t \in [0,1]^K$ are state and observation variables; $\boldsymbol{u} \in \mathbb{R}^K$ a bias term; $A \in \mathbb{R}^{L\times L}$ a state transition matrix; $C \in \mathbb{R}^{K\times L}$ an observation matrix; $\boldsymbol{v}_t \sim \mathcal{N}(\mathbf{0}, Q)$ a state noise process; $\boldsymbol{x}_1 = \boldsymbol{\mu}_0 + \boldsymbol{v}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, S_0)$ an initial condition; $B(\boldsymbol{y}; \boldsymbol{p})$ a multivariate Bernoulli distribution of parameter $\boldsymbol{p} \in [0,1]^K$, and $\sigma(\boldsymbol{\theta})$ a component-wise logistic transformation, *i.e.*, $\sigma_i(\boldsymbol{\theta}) = (1 + e^{-\theta_i})^{-1}$. The observation model of (3b) can be interpreted as a *binary* principle component analysis (binary PCA) [24] of $\{\boldsymbol{y}_t\}$. Binary PCA is a dimensionality reduction technique for binary data. Given a matrix $Y = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_\tau] \in \{0,1\}^{K\times\tau}$, it determines a $L$-dimensional ($L \ll K$) embedding of the natural parameters $\Theta$ of the Bernoulli distribution, by maximizing the log-likelihood

$$\mathcal{L} = \log p(Y; \Theta) = \log \left[ \prod_{k,t} \sigma(\Theta_{kt})^{Y_{kt}} \sigma(-\Theta_{kt})^{1-Y_{kt}} \right] \tag{4}$$

subject to the constraint

$$\Theta = CX + \boldsymbol{u}\mathbf{1}^T, \tag{5}$$

where $C \in \mathbb{R}^{K\times L}$, $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_\tau] \in \mathbb{R}^{L\times\tau}$, $\boldsymbol{u} \in \mathbb{R}^K$ and $\mathbf{1} \in \mathbb{R}^\tau$ is the vector of all ones. Each column of $C$ is a basis vector of a latent subspace and the $t$-th column of $X$ contains the coordinates of the $\boldsymbol{y}_t$ in this basis (up to a translation by $\boldsymbol{u}$).

Since, in the context of attribute representations, only the the attribute scores $\boldsymbol{\pi}_t$ (and not the attribute variables themselves) are known, [14] replaced the log-likelihood of (4) by the expected log-likelihood

$$\mathbb{E}_Y[\mathcal{L}] = \sum_{k,t} \Big[ \pi_{kt} \log \sigma(\Theta_{kt}) + (1 - \pi_{kt}) \log \sigma(-\Theta_{kt}) \Big]. \tag{6}$$

The maximization of (6) under the constraint of (5) can be performed with an *expectation-maximization* (EM) -like iterative algorithm [24], which produces estimates of the parameters $C$, $\boldsymbol{u}$ and the latent sequence $X$. [14] exploited this to propose a BDS extension of the popular *dynamic texture* algorithm for learning linear dynamic systems [5, 2]. Given a sample $\mathcal{D}_b = \{\boldsymbol{y}_i\}_{i=1}^{\tau}$, this consists of learning the observation and state transition models in two steps. The first is a binary PCA analysis of $\mathcal{D}_b$, to determine $C$, $\boldsymbol{u}$, and the coefficients $\{\boldsymbol{x}_t\}$. As shown in Figure 2, $\{\boldsymbol{x}_t\}$ is a trajectory in the state space, which follows a Gauss-Markov process. The second step determines the matrix $A$ that provides the least squares fit to these coefficients. Note that this matrix characterizes the state space trajectory, which is mapped (given $C$ and $\boldsymbol{u}$) into the video trajectory in $\mathcal{S}$. Hence, $A$ depicts the *dynamics* of the attribute sequence.

### 3.3. Bag of Words for Attribute Dynamics

While substantially more descriptive than the holistic attribute model of [15], the BDS of [14] still has two serious limitations as a model of video dynamics. These are illustrated in Figure 1. First, there is, in general, no guarantee that the whole video sequence depicts the activity of interest. On the contrary, the segments that matter for event recognition (*e.g.*, a segment of "tennis-serve") are frequently surrounded by segments that are not informative for the recognition (*e.g.*, video of subsequent plays). Fitting a single dynamic model to long video sequences will lead to parameter estimates that are not representative of the event of interest. Second, since complex activities are composed of several atomic actions, sometimes disjoint in time, their state trajectories are unlikely to follow the Gauss-Markov process. Both of these limitations, however, are unlikely to hold if the BDS is fitted to a short-term video segment.

On the other hand, most activities can be effectively inferred by a characterization of the short-term segments that compose them. For example, the characterization of the activity "long-jump" by the attribute sequence "run-run", "run-jump" and "jump-land", is sufficient to discriminate it from the (very similar) activity "triple-jump", if the latter is characterized by the attribute sequence "run-jump", "jump-jump" and "jump-land". The presence (or absence) of a video segment with attributes "jump-jump" is sufficient to discriminate between the two activities. Based on these observations, we propose to model video with an extension of the BoVW that captures the *short-term dynamics* of the attribute representation of an action.

A video sequence is first split into a collection of temporal overlapping segments $\{\boldsymbol{s}^{(i)}\}_{i=1}^{N}$. Segment $\boldsymbol{s}^{(i)}$ has $\tau_i$ frames, which are fed to the attribute mapping of (7). This produces a set of attribute score vectors $\boldsymbol{\Pi}^{(i)} = \{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}$, which is denoted the *attribute sequence* of segment $\boldsymbol{s}^{(i)}$. The video sequence is finally represented by a *bag of attribute sequences* (BoAS), which plays the role, in the proposed framework, of the BoF in image classification. A dictionary of representative BDS's $\{\boldsymbol{\Omega}^{(i)}\}_{i=1}^{V}$, which are denoted *words for attributes dynamics* (WAD), learned from a set of training BoAS, is then used to quantize the BoAS extracted from the video sequence to classify. The resulting histogram of WAD counts, denoted a *bag of words for attribute dynamics* (BoWAD) is finally used as a feature vector for video classification. This representation is summarized in Figure 3.

## 4. Learning and Recognition with BoWADs

In section 5 we will show that, when combined with standard histogram-based classifiers *e.g.*, support vector machines (SVMs) with histogram intersection kernel (HIK), BoWADs are a very effective representation for the recognition of complex activities. For now, we address the problem of quantizing attribute sequences. We start with the problem of learning a WAD dictionary.

### 4.1. Clustering Samples in the Model Domain

Traditional clustering (*e.g.*, $k$-means) searches for prototypes in the space of training samples (*e.g.*, in $k$-means, a cluster prototype is the centroid of the samples in the cluster), using a metric suited for that space (*e.g.*, Euclidean distance). An extension to the clustering of BoAS is not straightforward because 1) attribute sequences can have different length; 2) the space of these sequences has non-Euclidean geometry; and 3) the search for optimal prototypes, under this geometry, may lead to intractable non-linear optimization. More importantly, because we are interested in characterizing the *appearance and dynamics of attribute sequences*, it is more desirable to find a set of prototype BDS's than a set of prototype sequences.

This becomes a problem of learning a *bag-of-models* (BoM) where, given a set of training samples $\mathcal{D} = \{\boldsymbol{z}_i\}_{i=1}^{N}$ ($\boldsymbol{z}_i \in \mathcal{Z}, \forall i$), the goal is to learn a dictionary of representative *models* $\{M_i\}_{i=1}^{N_C}$ in a *model space* $\mathcal{M}$. The proposed solution is based on two mappings. The first

$$f_{\mathcal{M}} : \mathcal{Z} \supseteq \{\boldsymbol{z}_i\} \mapsto M(\{\boldsymbol{z}_i\}) \in \mathcal{M} \tag{7}$$

maps a collection of examples $\{\boldsymbol{z}_i\} \subseteq \mathcal{D}$ into a model $M(\{\boldsymbol{z}_i\})$. The second,

$$\mathcal{M} \times \mathcal{M} \ni (M_1, M_2) \mapsto d_{\mathcal{M}}(M_1, M_2) \in \mathbb{R}_+ \tag{8}$$

is a measure of distance between models. The mapping of (7) is first used to produce a model $M(\boldsymbol{z}_i)$ per training example $\boldsymbol{z}_i$. Training samples are then clustered, at the model level, by alternating between two steps. In the *assignment step*, each $\boldsymbol{z}_i$ is assigned to the cluster whose model is closest to $M(\boldsymbol{z}_i)$, using the metric (8). In the *model refinement step*, the model associated with each cluster is relearned from the training samples assigned to it, via (7). This procedure is summarized in Algorithm 1 and denoted *bag-of-models clustering* (BMC).

BMC generalizes $k$-means, where $\boldsymbol{z}_i \in \mathbb{R}^d$ are feature vectors, $\mathcal{M}$ is the family of Gaussians of identity covariance

$$\mathcal{M} = \left\{ p(\boldsymbol{z}; \boldsymbol{\mu}) = \mathcal{G}(\boldsymbol{z}; \boldsymbol{\mu}, I_d) \mid \boldsymbol{\mu} \in \mathbb{R}^d \right\}, \tag{9}$$

(7) selects the model

$$M(\{\boldsymbol{z}_i\}) = \mathcal{G}(\boldsymbol{z}; \hat{\boldsymbol{\mu}}, I), \tag{10}$$

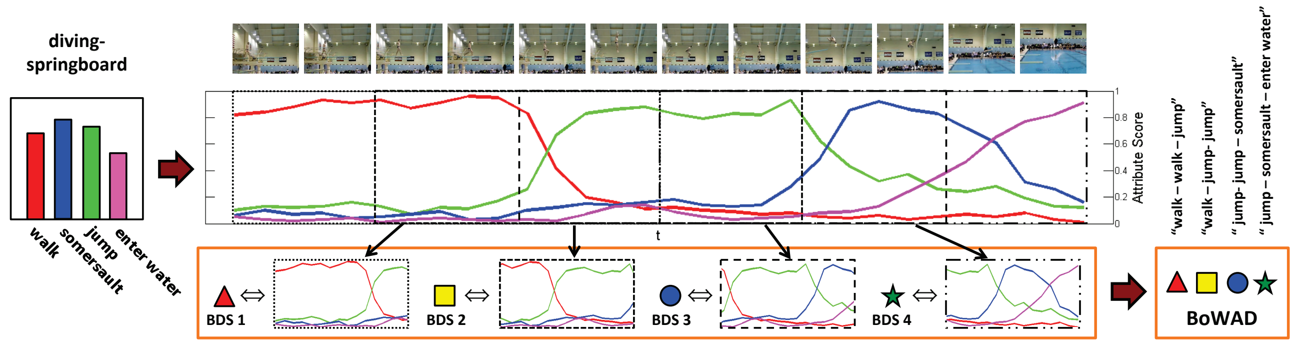


**Figure 3:** BoWAD representation of the activity "diving-springboard". (Top) video sequence. (Middle) the holistic vector of attribute scores is now represented as a trajectory in the attribute space (which is four dimensional, in this example, and represented as four colored functions). The trajectory is split into overlapping sort-term segments. (Bottom) each segment is assigned to the WAD associated with the BDS, in a learned BDS dictionary, that best explains it. Dictionary BDS's are models of short-term behaviors, such as "walk-walk-jump", "walk-jump-jump", "jump-jump-somersault" and "jump-somersault-enter water". The activity is represented by a BoWAD, which is a histogram of assignments of segments to WADs.

---

**Algorithm 1:** Bag-of-Models Clustering

**Input** : a set of samples $\mathcal{D} = \{\boldsymbol{z}_i\}_{i=1}^N$ $(\boldsymbol{z}_i \in \mathcal{Z}, \forall i)$, number of clusters $N_C$, an initial set of models $\{M_i^{(0)}\}_{i=1}^{N_C}$.

*set* $t = 0$ *and* $S_i^{(0)} = \varnothing, i = 1, \cdots, N_C$;

**repeat**

  $t = t + 1$;

  *Assignment-Step*: $\forall i, S_i^{(t)} = \{\boldsymbol{z} \in \mathcal{D} \mid \forall j \neq i, d_{\mathcal{M}}(M(\boldsymbol{z}), M_i^{(t-1)}) \leqslant d_{\mathcal{M}}(M(\boldsymbol{z}), M_j^{(t-1)})\}$

  *Refinement-Step*: $\forall i, M_i^{(t)} = M(\{S_i^{(t)}\})$

**until** $\forall i, S_i^{(t)} = S_i^{(t-1)}$;

**Output**: $\{M_i^{(t)}\}_{i=1}^{N_C}$ and $\{S_i^{(t)}\}_{i=1}^{N_C}$

---

where $\hat{\boldsymbol{\mu}}$ is the maximum likelihood estimate of the mean

$$\hat{\boldsymbol{\mu}} = \underset{\boldsymbol{\mu}}{\operatorname{argmax}}\, p(\{\boldsymbol{z}_i\}; \boldsymbol{\mu}) = \frac{1}{|\{\boldsymbol{z}_i\}|} \sum\nolimits_i \boldsymbol{z}_i, \tag{11}$$

and the measure of (8) is the (symmetric) Kullback-Leibler divergence

$$\mathrm{KL}(p_1||p_2) + \mathrm{KL}(p_2||p_1) = \ ||\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2||^2. \tag{12}$$

It should be noted that BMC (Algorithm 1) differs from the *bag-of-systems* method of [21, 1] in two ways. First, it clusters *attribute sequences* rather than the models themselves, as is done by [21, 1]. Note that, in the model refinement step of Algorithm 1, models are re-learned from examples $\{\boldsymbol{z}_i\}$. The refinement step of [21, 1] only considers the parameters of the models $M(\boldsymbol{z}_i)$ and not the examples $\boldsymbol{z}_i$ themselves. This usually entails loss of information. Second, Algorithm 1 *finds* the optimal representative for each cluster, according to the model fitting criterion of (7). In [21], the difficult geometry of the manifold defined by the LDS parameter tuple $(A, C) \in \mathbb{GL}(n) \times \mathbb{ST}(p, n)$, where $\mathbb{GL}(i)$ is the set of invertible matrices of size $n$ and $\mathbb{ST}(p, n)$ the Stiefel manifold of $p \times n$ orthonormal matrices $(p \geqslant n)$, precludes a simple estimate of the optimal representative. Instead, this is approximated by searching for the model $M(\boldsymbol{z}_i)$ closest to the optimal representative. Although [1] introduces an approach to directly cluster LDS's in their parameter space, its generalization to the BDS is still not quite clear. We will show, in Section 5, that these differences can lead to significantly improved performance by Algorithm 1.

## 4.2. Learning a Vocabulary of WADs

A WAD dictionary is learned by applying Algorithm 1 to a BoAS $\mathcal{P} = \{\boldsymbol{\Pi}^{(i)}\}_{i=1}^N$, as follows.

**Refinement-Step:** The mapping of (7) amounts to fitting a BDS to a BoAS $\mathcal{P}' = \{\boldsymbol{\Pi}^{(i)}\} \subseteq \mathcal{P}$. This is done with recourse to Algorithm 2, which extends the algorithm

---

**Algorithm 2:** Learning a Cluster for WADs Dictionary

**Input** : a set of $n$ sequences of attribute score vectors $\{\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}\}_{i=1}^n$, state space dimension $L$.

Binary PCA:

$\{C, X, \boldsymbol{u}\} = \text{B-PCA}(\{\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}\}_{i=1}^n, L)$ [24].

Estimate state parameters:

$A = \hat{X}_2^\tau (\hat{X}_1^{\tau-1})^\dagger, \quad V = \hat{X}_2^\tau - A\hat{X}_1^{\tau-1},$

(where $\hat{X}_2^\tau = [(X^{(1)})_2^{\tau_1}, \cdots, (X^{(n)})_2^{\tau_n}]$,

$\hat{X}_1^{\tau-1} = [(X^{(1)})_1^{\tau_1-1}, \cdots, (X^{(n)})_1^{\tau_n-1}]$,

and $X_{t_1}^{t_2} \equiv [\boldsymbol{x}_{t_1}, \cdots, \boldsymbol{x}_{t_2}]$).

$Q = \frac{1}{\sum_i (\tau_i - 1)} V(V)^T, \quad \boldsymbol{\mu}_0 = \frac{1}{n} \sum_{i=1}^n \boldsymbol{x}_1^{(i)},$

$S_0 = \frac{1}{n-1} \sum_{i=1}^n (\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_0)(\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_0)^T.$

**Output**: $\boldsymbol{\Omega} = \{A, C, Q, \boldsymbol{u}, \boldsymbol{\mu}_0, S_0\}$

---

of [14] for learning a BDS from a single attribute sequence. The extension follows the two-step decomposition of BDS learning discussed in Section 3.2. A binary PCA is first applied to *all* attribute score vectors in $\mathcal{P}'$. The parameters of the hidden Gauss-Markov process are then learned by solving a least squares problem involving *all* latent state sequences returned by binary PCA. In this way, the BDS learned per cluster jointly characterizes the appearance and dynamics of all attribute sequences in that cluster.

**Assignment-Step:** As a measure of distance between two BDS's, we use the Binet-Cauchy (BC) kernel. This was originally proposed in [26] as a measure of dissimilarity between infinite output sequences of two LDS's, and adapted to a measure of the dissimilarity between the outputs of two BDS's, $\boldsymbol{\Omega}_a$ and $\boldsymbol{\Omega}_b$, in [14]. It is defined as

$$\begin{aligned} & d_{BC}(\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b) \\ &= \mathbb{E}_{\boldsymbol{v}}\Big[\sum_{t=0}^{\infty} e^{-\lambda t}\Big(KL(B(\sigma(\boldsymbol{\theta}_t^{(a)}))||B(\sigma(\boldsymbol{\theta}_t^{(b)}))) \\ &\qquad + KL(B(\sigma(\boldsymbol{\theta}_t^{(b)}))||B(\sigma(\boldsymbol{\theta}_t^{(a)})))\Big)\Big] \\ &= \mathbb{E}_{\boldsymbol{v}}\Big[\sum_{t=0}^{\infty} e^{-\lambda t}\left(\sigma(\boldsymbol{\theta}_t^{(a)}) - \sigma(\boldsymbol{\theta}_t^{(b)})\right)^T \left(\boldsymbol{\theta}_t^{(a)} - \boldsymbol{\theta}_t^{(b)}\right)\Big], \end{aligned} \tag{13}$$

where $\{\sigma(\boldsymbol{\theta}_t^{(a)})\}$ and $\{\sigma(\boldsymbol{\theta}_t^{(b)})\}$ are the parameters of the multivariate Bernoulli distributions from which the binary attribute vectors are sampled, for the two BDS's. While the BC kernel between two LDS's can be computed in closed form, the evaluation of (13) is not trivial. Like the latent state sequence $\{\boldsymbol{x}_t\}$, its linear projection $\{\boldsymbol{\theta}_t\}$ is a sample from a high-dimensional Gaussian distribution. Hence, (13) amounts to computing the expectation of a nonlinear function with respect to a multivariate Gaussian distribution, and is intractable in general. Following [14], we resort to a numeric solution which approximates the summation by a finite number of terms. This has been empirically shown to produce good results.

### 4.3. Quantization

Given a WAD dictionary $\{\boldsymbol{\Omega}^{(i)}\}_{i=1}^{V}$, a BoAS $\{\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}\}_{i=1}^{N}$ is quantized by assigning the $i$-th attribute sequence to the $k^*$-th cluster according to

$$k^* = \operatorname{argmin}_j \ d_{BC}\big(\boldsymbol{\Omega}(\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}), \boldsymbol{\Omega}^{(j)}\big), \tag{14}$$

where $\boldsymbol{\Omega}(\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i})$ is the BDS learnt from $\{\boldsymbol{\pi}_t^{(i)}\}_{t=1}^{\tau_i}$ using (7).

## 5. Experiments

A number of experiments were performed to compare the BoWAD representation to previous models of temporal activity structure. The low-level representation used in all experiments was the BoF of [11]. A set of spatio-temporal interest points (STIPs) were first detected, a feature vector was extracted from the support of each interest point, and quantized into a vocabulary learnt from the training set. Binary SVMs using histogram intersection kernel (HIK) with probability outputs [3] were used as attribute models, learned from annotated training video clips (see supplementary material for attribue definitions). In all experiments, both BDS's and BoWADs used a 5-dimensional state space.

**Table 1:** Classification Accuracy on Weizmann Activity

| **Sets** | BoF | BoF-TP [11] | Attribute [15] | BDS [14] | BoWAD: MDS-$k$M [21] | BoWAD: BMC |
|---|---|---|---|---|---|---|
| Syn20×1 | 23.3% | 36.7% | 17.8% | 64.4% | **100%** | **100%** |
| Syn10×2 | 28.9% | 31.1% | 16.7% | 65.6% | 98.9% | **100%** |

### 5.1. Weizmann Activity

The first set of experiments was based on composite sequences synthesized from the Weizmann dataset [8], which contains 10 atomic action classes, performed by 9 people, for a total of 90 samples. BoWAD was compared to the vanilla BoF, BoF with $t3$ temporal pyramids [11] (denoted "BoF-TP"), holistic attributes [15] (denoted "Attribute") and BDS [14]. Attribute sequences were computed over 30-frame sliding video windows of 10-frame step. As in [14], 30 low-level attributes were defined for the original 10 actions. To compute BoWADs, each short-term attribute sequence consisted of the attribute vectors from 12 consecutive windows, extracted with a step of 3 windows. WAD dictionaries were learned with both BMC and the MDS-$k$M algorithm of [21] . One-*v.s.*-all SVMs with HIK were used in all histogram-based methods (BoF, BoF-TP, BoWAD, attribute models), where STIP features used a 1000-word vocabulary. For the BDS, we used the kernel $K(\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b) = \exp(-\frac{1}{\gamma} d_{BC}^2(\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b))$ (same for the rest of experiments).

Two datasets were created. The first, "Syn20×1", aimed to test the ability of the different approaches to detect activity classes of large variability. An activity was defined as a sequence of 20 *consecutive* atomic actions from Weizmann. This sequence was inserted at a random temporal location of a larger sequence of 40 atomic actions. The remaining 20 actions in the larger sequence were randomly selected from Weizmann. The second, "Syn10×2", tested the ability of the different approaches to detect *discontinuous* activities. In this case, each activity was defined by two subsequences, each with 10 consecutive atomic actions. The two subsequences were randomly inserted at non-overlapping locations of the larger (40 atomic actions) sequence.

Table 1 summarizes the performance of the different methods. The very weak performance of BoF, BoF-TP, and Attribute, show that modeling of activity dynamics is criti-

**Table 2:** Average Precisions for Activity Recognition on Olympic Sports Dataset.

| **Activity** | Laptev *et al.* [11] (BoF-TP) | Niebles *et al.* [16] | Tang *et al.* [25] | Attri-bute [15] | BDS [14] | BoWAD | |
|---|---|---|---|---|---|---|---|
| | | | | | | MDS-$k$M [21] | BMC |
| high-jump | 52.4% | 68.9% | 18.4% | **93.2%** | 82.2% | 86.8% | 83.9% |
| long-jump | 66.8% | 74.8% | 81.8% | 82.6% | **92.5%** | 83.9% | 91.9% |
| triple-jump | 36.1% | 52.3% | 16.1% | 48.3% | 52.1% | 64.2% | **75.7%** |
| pole-vault | 47.8% | 82.0% | **84.9%** | 74.4% | 79.4% | 68.0% | 76.5% |
| gym. vault | 88.6% | 86.1% | 85.7% | 86.7% | 83.4% | 86.7% | **91.4%** |
| shot-put | 56.2% | 62.1% | 43.3% | 76.2% | 70.3% | 58.0% | **79.4%** |
| snatch | 41.8% | 69.2% | **88.6%** | 71.6% | 72.7% | 56.4% | 73.4% |
| clean-jerk | 83.2% | 84.1% | 78.2% | 79.4% | 85.1% | 78.2% | **85.4%** |
| javelin throw | 61.1% | 74.6% | 79.5% | 62.1% | **87.5%** | 56.6% | 76.7% |
| ham. throw | 65.1% | 77.5% | 70.5% | 65.5% | 74.0% | 71.3% | **79.2%** |
| discus throw | 37.4% | 58.5% | 48.9% | **68.9%** | 57.0% | 62.6% | 66.9% |
| diving-plat. | 91.5% | 87.2% | **93.7%** | 77.5% | 86.0% | 85.2% | 82.0% |
| diving-sp. bd. | 80.7% | 77.2% | 79.3% | 65.2% | 78.3% | 75.2% | **82.3%** |
| bask. layup | 75.8% | 77.9% | **85.5%** | 66.7% | 78.1% | 66.6% | 60.8% |
| bowling | 66.7% | 72.7% | 64.3% | 72.0% | 52.5% | 64.4% | **73.0%** |
| tennis-serve | 39.6% | 49.1% | 49.6% | 55.2% | 38.7% | 68.1% | **73.2%** |
| mean AP | 62.0% | 72.1% | 66.8% | 71.6% | 73.2% | 70.8% | **78.2%** |

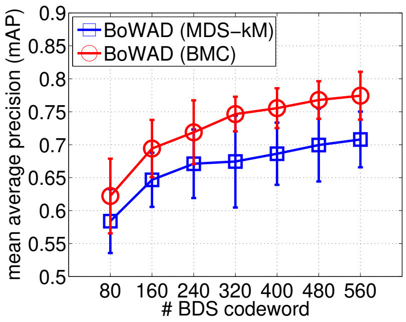


**Figure 4:** Mean average precision (mAP) *v.s.* size of BDS dictionary on Olympic Sports. Vertical bars indicate standard deviation of mAP in cross-validation.

cal for success in these datasets. While the BDS has substantially improved performance, the underlying assumption of a single dynamic process is a limitation for these sequences, where the activities of interest are not temporally aligned and are surrounded by irrelevant video. Substantially better performance is achieved with the BoWAD representation, which has perfect performance on these datasets. Both clustering strategies achieve good results, although BMC outperforms MDS-$k$M slightly.

### 5.2. Olympic Sports

The second set of experiment was conducted on the Olympic Sports dataset [16]. The performance of BoWADs, learned with BMC and MDS-$k$M, was compared to BoF-TP [11], activity models with decomposable segments [16], the hidden Markov model with latent states of variable duration of [25], the holistic attribute representation of [15], and the BDS [14]. In all cases, a 3000-word STIP vocabulary was used to quantize low-level features. BDS's and BoWAD used the 40 attributes defined by [15]. A 30 frame sliding video window, with a step of 4 frames, was used to compute attribute scores. For the BoWAD, attribute sequences consisted of 12 consecutive attribute vectors, with a 75% overlap between consecutive sequences. Performance was measured with per-category average precisions (AP) and mean AP, using 5-fold cross-validation.

As shown in Table 2, the BoWAD again achieves the best results. In fact, it achieves the best results reported in the literature with the similar low-level features (STIP) on this dataset. This includes methods based on much more sophisticated classifiers, such as the 74.4% of [15] or the 76.5% of [14], which use latent SVMs or multiple kernel classifiers to combine supervised, unsupervised attributes (dynamics), and low-level features. The BoWAD achieves 78.2% by simply quantizing attribute dynamics. It works particularly well for categories, such as "tennis-serve", which have large variability and tend to include video irrelevant for activity detection, or category pairs, such as "triple-jump" and "long-jump", that differ in subtle ways. The robustness inherent to a vocabulary of dynamics is critical for the former (compare the 73.2% of BoWAD-BMC with the 38.7% of the BDS on "tennis serve"), while the detailed characterization of attribute dynamics is critical for the latter (75.7% *v.s.* 48.3% of Attribute on "triple-jump"). With regards to clustering algorithms, there is now a substantial gap between MDS-$k$M (70.8%) and BMC (78.2%). Figure 4 shows that this difference holds across a large range of WAD dictionary sizes. The robustness of the proposed representation is reinforced by the fact that a 320-word BoWAD has mAP (75%) superior to all other representations of Table 2.

### 5.3. TRECVID-MED11

The third set of experiments used the 2011 TRECVID multimedia event detection (MED) open source dataset [17]. The event collection (EC) set was used for training and the development set (DEVT) for testing (events 1-5). EC contains 2,062 training samples of 5 high-level events, with 100-200 positive examples per event. DEVT has around 11,000 samples. We manually defined 93 attributes and used a 10,000-word low-level feature dictionary. Attribute scores were computed with a 180-frame sliding window with steps of 30 frames, and attribute sub-sequences ($\tau = 10$) were extracted every window. BoWAD used a dictionary of size 1000.

The performance of the different methods is summarized in Table 3. On this highly challenging dataset, the gap between BoWAD and the other representations is enor-

**Table 3:** Average Precision for Event Detection on TRECVID MED11 DEVT Dataset.

| **Event** (E001-E005) | Random Guess | Laptev *et al.* [11] (BoF-TP) | Niebles *et al.* [16] | Tang *et al.* [25] ($d = 1$ / $d \leqslant d_{max}$) | Attribute [15] | BDS [14] | BoWAD | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | MDS-$k$M [21] | BMC |
| attempting a board trick | 1.18% | 8.22% | 5.84% | 6.24% / 15.44% | 18.91% | 8.41% | 26.62% | **29.99%** |
| feeding an animal | 1.06% | 2.54% | 2.28% | 5.28% / 3.55% | 4.95% | 1.78% | 4.61% | **7.36%** |
| landing a fish | 0.89% | 9.77% | 9.18% | 7.30% / 14.02% | 24.17% | 6.20% | 24.97% | **28.10%** |
| wedding ceremony | 0.86% | 5.52% | 7.26% | 9.48% / 15.09% | 16.68% | 12.24% | 22.15% | **22.39%** |
| working on a wood project | 0.93% | 4.09% | 4.05% | 3.42% / 8.17% | 5.11% | 5.08% | 12.39% | **18.32%** |
| mean AP | 0.98% | 6.01% | 5.72% | 6.34% / 11.25% | 13.96% | 6.74% | 18.15% | **21.23%** |

mous. In fact, the BoWAD learned by BMC (21.23%) almost doubles the best previous results in the literature that model temporal structure of complex events (*i.e.*, 11.25% of [25]). The fact that the BoWAD substantially outperforms the BDS also confirms the observation that the robustness of a vocabulary of local attribute dynamics is critical for accurate detection of complex activities. For example, events in the class "attempting a board trick" include a repetition of local actions, *e.g.*, "slide-jump-(somersault)-land-slide". While it is difficult to model this sequence as a whole, due the large variability of cutting in different videos, it is much easier to capture short-term signature actions, such as "slide-jump", which are usually not broken during video editing. Finally, with respect to clustering algorithms, BMC agains substantially outperforms MDS-$k$M.

## 6. Conclusion

In this work, we proposed a novel solution to the problem of modeling attribute and dynamics for activity recognition. The method combines the advantages, in terms of robustness, of histogram-based representations, with the power of BDS's to model the dynamics of video attributes. We developed new algorithms for learning BDS dictionaries and quantizing video with them. The proposed representation significantly outperforms other state-of-the-art attribute-based or temporal-structure-modeling approaches in complex activity recognition.

## References


[1] B. Afsari, R. Chaudhry, A. Ravichandran, and R. Vidal. Group action induced distances for averaging and clustering linear dynamical systems with applications to the analysis of dynamic scenes. *CVPR*, 2012. 2, 5

[2] A. Chan and N. Vasconcelos. Classifying video with kernel dynamic textures. *CVPR*, 2007. 3

[3] C. Chang and C. Lin. Libsvm: a library for support vector machines. *ACM Trans. Intelligent Systems and Technology*, 2(3):27, 2011. 6

[4] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. *CVPR*, 2009. 1, 2

[5] G. Doretto, A. Chiuso, Y. Wu, and S. Soatto. Dynamic textures. *IJCV*, 51(2):91–109, 2003. 2, 3

[6] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong. Attribute learning for understanding unstructured social activity. *ECCV*, 2012. 1

[7] A. Gaidon, Z. Harchaoui, and C. Schmid. Actom sequence models for efficient action detection. *CVPR*, 2011. 1, 2

[8] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *IEEE TPAMI*, 29(12):2247–2253, 2007. 6

[9] V. Kellokumpu, G. Zhao, and M. Pietikäinen. Human activity recognition using a dynamic texture based method. *BMVC*, 2008. 1, 2

[10] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. *CVPR*, 2009. 2

[11] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. *CVPR*, 2008. 1, 2, 6, 7, 8

[12] B. Laxton, J. Lim, and D. Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. *CVPR*, 2007. 1, 2

[13] B. Li, M. Ayazoglu, T. Mao, O. Camps, and M. Sznaier. Activity recognition using dynamic subspace angles. *CVPR*, 2011. 2

[14] W. Li and N. Vasconcelos. Recognizing activities by attribute dynamics. *NIPS*, 2012. 1, 2, 3, 4, 6, 7, 8

[15] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. *CVPR*, 2011. 1, 2, 3, 4, 6, 7, 8

[16] J. Niebles, C. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. *ECCV*, 2010. 1, 2, 7, 8

[17] P. Over, G. Awad, J. Fiscus, B. Antonishek, M. Michel, A. F. Smeaton, W. Kraaij, and G. Quenot. Trecvid 2011 – an overview of the goals, tasks, data, evaluation mechanisms, and metrics. *Proceedings of TRECVID 2011*, 2011. 7

[18] N. Rasiwasi, P. J. Moreno, and N. Vasconcelos. Bridging the gap: Query by semantic example. *IEEE Trans. Multimedia*, 9(5):923–938, 2007. 2

[19] N. Rasiwasia and N. Vasconcelos. Holistic context models for visual recognition. *IEEE TPAMI*, 34(5):902–917, 2012. 2

[20] A. Ravichandran, R. Chaudhry, and R. Vidal. Categorizing dynamic textures using a bag of dynamical systems. *IEEE TPAMI*, 35(2):342–353, 2012. 2, 5, 6, 7, 8

[21] M. Rohrbach, M. Regneri, M. Andriluka, S. Amin, M. Pinkal, and B. Schiele. Script data for attribute-based recognition of composite activities. *ECCV*, 2012. 1, 2

[22] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. *CVPR*, 2012. 1, 2

[23] A. I. Schein, L. K. Saul, and L. H. Ungar. A generalized linear model for principal component analysis of binary data. *AISTATS*, 2003. 3, 5

[24] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. *CVPR*, 2012. 1, 7, 8

[25] S. Vishwanathan, A. J. Smola, and R. Vidal. Binet-cauchy kernels on dynamical systems and its application to the analysis of dynamic scenes. *IJCV*, 2006. 6

[26] H. Wang, M. Ullah, A. Kläser, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. *BMVC*, 2009. 2

[27] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213 – 238, 2007. 2