Effective Codebooks for Human Action Recognition

Lorenzo Seidenari, Lamberto Ballan, Marco Bertini, Alberto Del Bimbo and Giuseppe Serra
Media Integration and Communication Center (MICC), University of Florence, Italy

1. Introduction

Goal: **learn a model to represent and detect several human actions.**
Automatic recognition of human actions is an important task which recently has received large attention from the scientific community. Several applications could benefit from it, for example:

- In a visual surveillance scenario such a system could detect possibly dangerous situations, thus minimizing human effort and errors.
- A lot of videos we watch and download (news, movies, music clips, sports ...) contain people; their behaviour is often a very discriminant feature for automatic annotation and retrieval.
- An immersive HCI environment could unintrusively modify the application state or gather user feedback.

2. Classification Framework

The proposed method is based on the Bag-of-Words (BoW) approach; this technique aims at representing a video as an unordered collection of (visual) words. Local **spatio-temporal features** are extracted in correspondence of informative points. A "visual dictionary" is created by quantizing visual features. By assigning descriptors in videos to the nearest visual prototype in the dictionary the BoW is computed as a word count.

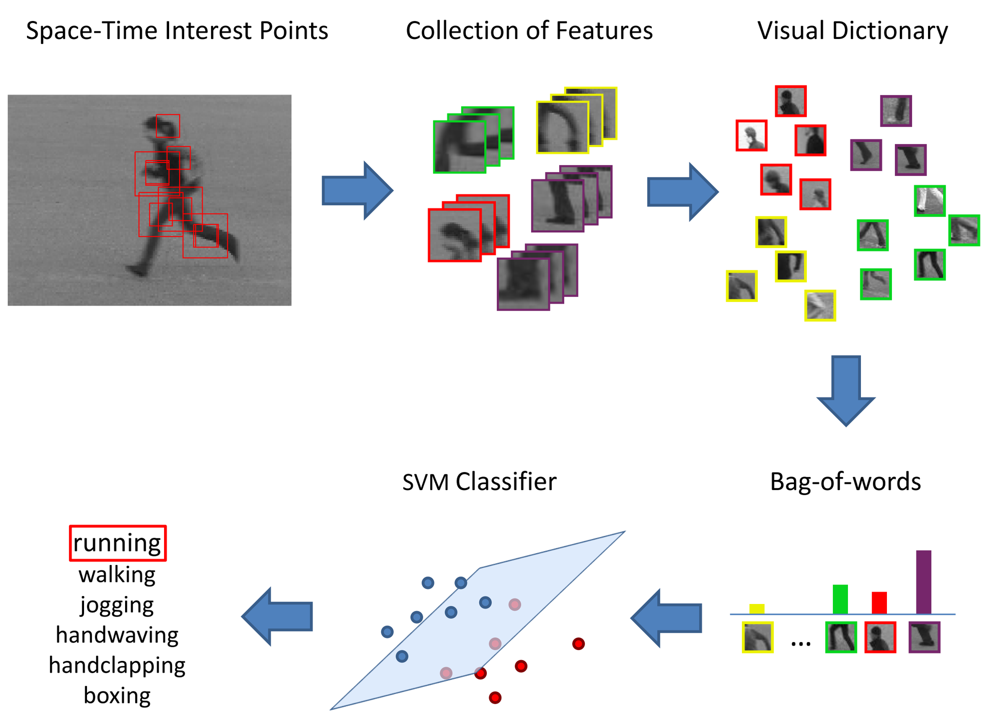

Figure 1: *Action Categorization Framework.*

3. Approach

Modelling human actions is a challenging task. Even simple actions may exhibit large **intra-class variability** due to:

- Actor appearance variation (clothing, posture and scale).
- Environment change (cluttered or dynamic background, illumination change).

Other issues are actors' limbs **self occlusions** and confusion between visually similar but semantically different actions (i.e. jogging and running). For these reasons we propose an approach which aims at modelling locally informative space-time patches at **multiple spatial and temporal scales**, using **motion** and dynamic **appearance** information.

4. Feature detection and description

Detector. The detector applies two separate linear filters to spatial and temporal dimensions, respectively. The response function is computed as follows:

$$R = (I(x,y,t) * g_\sigma(x,y) * h_{ev}(t))^2 + (I(x,y,t) * g_\sigma(x,y) * h_{od}(t))^2 \quad (1)$$

where $I(x,y,t)$ is a sequence of gray-level images over time, $g_\sigma(x,y)$ is the spatial Gaussian filter with kernel σ, h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters applied along the time dimension. The detector is applied at **multiple spatial and temporal scales**: $\sigma = \{2,4\}$ and $\tau = \{2,4\}$.

Descriptors. The space-time volume is divided in 18 sub-regions (3 along the spatial directions and 2 along the temporal) to compute **position dependent statistics**. We use 3D gradient and optical flow to build two descriptors.

- The gradient descriptor is computed by quantizing the angles θ (8 bins) and ϕ (4 bins).
- Either the optic flow descriptor is computed from orientation but by adding a "no-motion" bin.

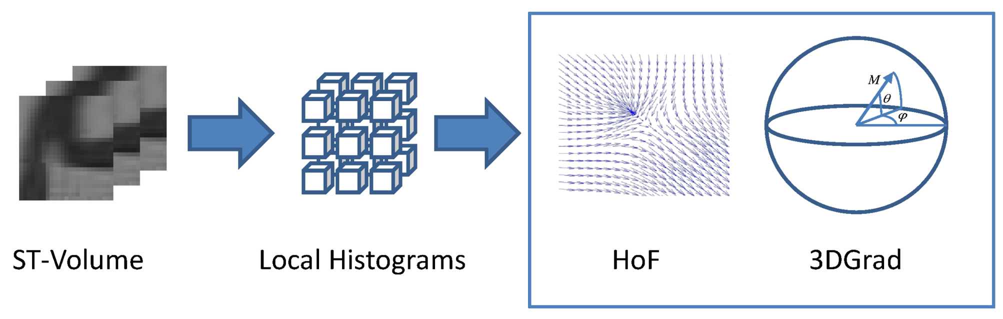

Figure 2: *Descriptor computation flow.*

5. Action representation and classification

We propose three improvements in the action representation model:

- We propose two different **descriptor combination** strategies: at the feature level, and at the BoW level. This allow to exploit their **complementarity**.
- Due to the **high dimensionality** of our descriptor and the **dense sampling** of the multiscale detector the feature space is highly non-uniformly populated. **Radius-based** clustering provides a better encoding of intermediate frequencies visual words.
- In high dimensional feature spaces, finding the best visual word prototype can be difficult; using Gaussian kernel density estimation, we **smooth the hard assignment** computing the uncertainty frequency distribution with:

$$UFD(w) = \frac{1}{n}\sum_{i=1}^{n} \frac{K_\sigma(D(w,p_i))}{\sum_{j=1}^{|V|} K_\sigma(D(v_j,p_i))} \quad (2)$$

where D is the Euclidean distance and K_σ is the Gaussian kernel:

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}\frac{x^2}{\sigma^2}\right) \quad (3)$$

where σ is the scale parameter of the Gaussian kernel;

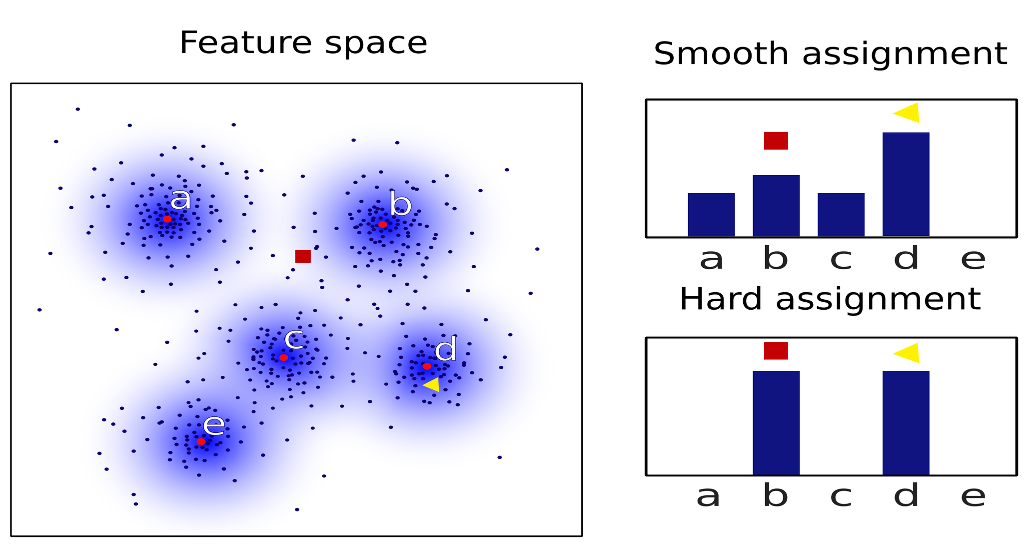

Figure 3: *Hard and soft feature-word assignment.*

Classification is performed using non-linear SVMs with the χ^2 kernel. To perform multi-class classification we use the *one-vs-one* approach.

6. Results

Datasets. We test our approach on two state-of-the-art datasets: **KTH** and **Weizmann**.

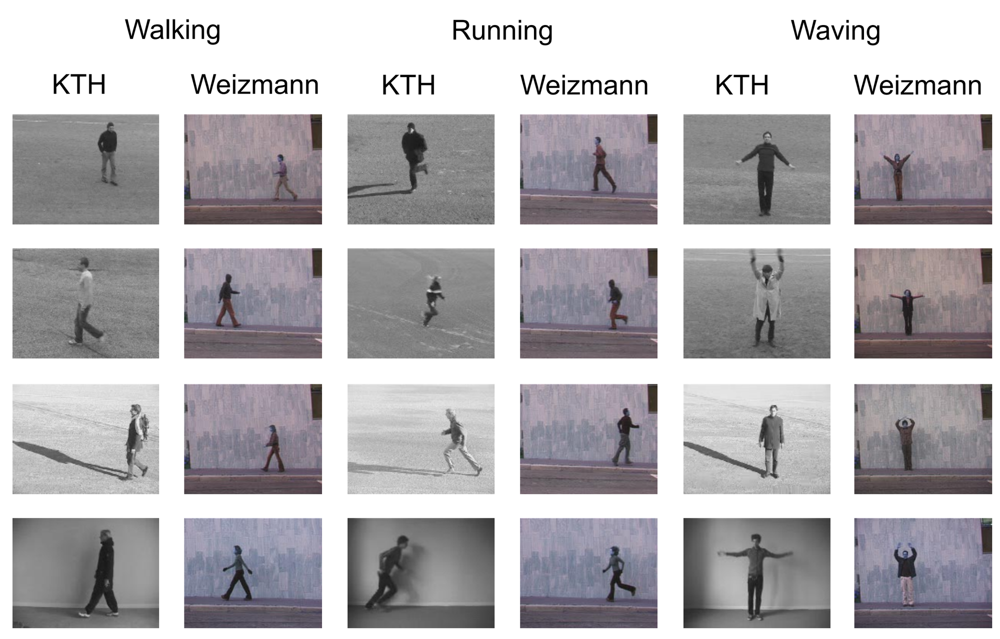

Figure 4: *Sample frames from the datasets used.*

Descriptor Evaluation. The best descriptor combination is obtained by concatenating the two BoW vectors.

Descriptor	KTH	Weizmann
3DGrad	90.38 $\pm$ 0.8	92.30$\pm$1.6
HoF	88.04 $\pm$ 0.7	89.74$\pm$1.8
3DGrad_HoF comb.	91.09 $\pm$ 0.4	92.38$\pm$1.9
3DGrad+HoF comb.	**92.10 $\pm$ 0.4**	**92.41 $\pm$1.9**

Table 1: *Comparison of our descriptors.*

Comparison to the state-of-the-art. We show a comparison of our method to the most recent results on both datasets.

Method		KTH	Weizmann
Our method		**92.57**	**95.41**
Laptev *et al.*	[CVPR08]	91.8	-
Dollár *et al.*	[PETS05]	81.2	-
Wong and Cipolla	[ICCV07]	86.62	-
Scovanner *et al.*	[MM07]	-	82.6
Niebles *et al.*	[IJCV08]	83.33	90
Liu *et al.*	[CVPR08]	-	90.4
Kläser *et al.*	[BMVC08]	91.4	84.3
Willems *et al.*	[ECCV08]	84.26	-
Schüldt *et al.*	[ICPR04]	71.7	-

Table 2: *Comparison to the state-of-the-art.*

Our method begin to perform better than k-means clustering as soon as **middle-low** frequency words are included in the codebook (500 for Weizmann, 1500 for KTH); with an appropriate choice of the codebook size this allows to **outperform the state-of-the-art**.

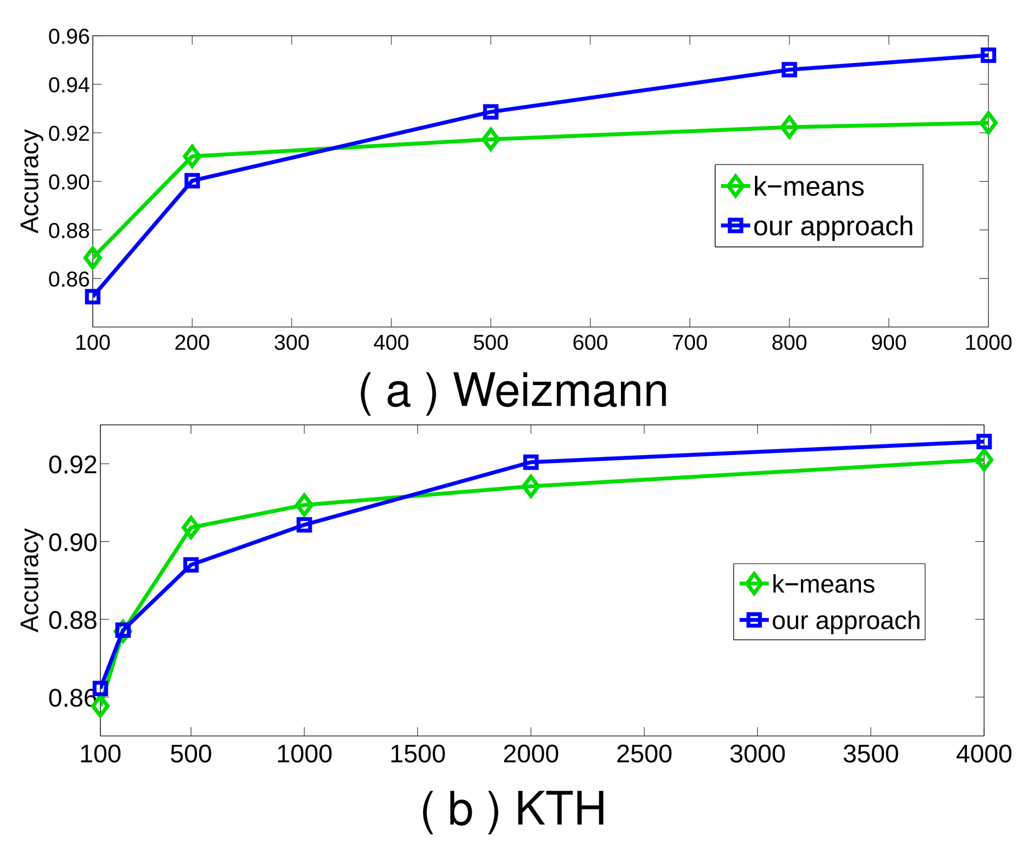

Figure 5: *Accuracy varying the codebook size.*