

# LEARNING ACTION DESCRIPTORS FOR RECOGNITION

M.J. Marín-Jiménez, N. Pérez de la Blanca, M.A. Mendoza\*

M. Lucena, J.M. Fuertes

Dpt. Computer Science and A.I.  
University of Granada  
Granada, Spain

Dpt. Informatica  
University of Jaen  
Jaen, Spain

## ABSTRACT

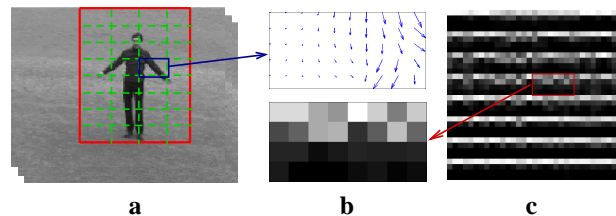
This paper evaluates different Restricted Boltzmann Machines models in unsupervised, semi-supervised and supervised frameworks using information from human actions. After feeding these multilayer models with low level features, we infer high-level discriminating features that highly improve the classification performance. This approach eliminates the difficult process of selecting good mid-level feature descriptors, changing the feature selection and extraction process by a learning stage. Two main contributions are presented. First, a new sequence-descriptor from accumulated histograms of optical flow (aHOF) is presented. Second, comparative results using unsupervised, supervised and semi-supervised classification experiments are shown. The results show that the RBM architectures provide very good results in our classification task and present very good properties for semi-supervised learning.

## 1. INTRODUCTION

Video-sequence classification of human motion is a challenging and open problem, at the root of which is the need of finding invariant characterizations of complex 3D human motions from 2D features [1]. In this context, searching for specific 2D features that code the highest possible discriminative information on 3D motion is a very relevant research problem.

Middle-level feature selection feeding a good classifier is the most popular approach in human action recognition (HAR). But this approach has the weakness of having to guess which functions of the raw data represent good features [2, 3, 4, 5, 6, 7]. In this respect, our approach, on one hand, is reminiscent of some of these ideas, since we use the low level information provided by optical flow, but, on the other hand, we process this information in a different way. In our approach, we learn our better feature vector from low level information using an abstraction process giving high level semantic features.

In [8, 9] Hinton introduced a new algorithm allowing to learn high level semantic features from raw data on Restricted



**Fig. 1. aHOF computation.** (a) BB enclosing person, with superimposed grid (8x4). (b) Top: optical flow inside the selected grid cell for the visible single frame. Bottom: in each aHOF cell, each column (one per orientation) is a histogram of OF magnitudes (i.e. 8 orientations  $\times$  4 magnitudes). (c) aHOF computed from 20 frames around the visible one. Note that in the areas with low motion (e.g. bottom half) most of the vectors vote in the lowest magnitude bins. (Intensity coding: white = 1, black = 0).

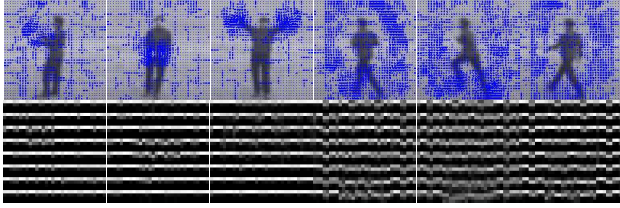
Boltzmann Machines (RBMs). In [10] the Discriminative Restricted Boltzmann Machine model (DBRM) is introduced as a discriminative alternative to the generative RBM. In [11] a distance measure is proposed on the feature space in order to get good features for non-parametric classifiers. Some of these algorithms have shown to be very successful in some image classification problems [12, 13, 14], where the raw data distributions are represented by the pixel gray level values. However, in our case, the motion describing the action is not explicitly represented in the raw image and a representation of it must be introduced. Here we evaluate the efficacy of these architectures to encode better features from the raw data descriptor in the different learning setups.

In section 2 the feature extraction process and the development of the new action descriptor are presented. Section 3 summarizes the main properties of the multilayer architectures. Section 4 shows the experimental set up with the obtained results. Finally, in section 5, conclusions and future work are presented.

## 2. ACTION DESCRIPTOR: AHOF

For each image, we focus our interest on the Bounding Box (BB) area enclosing the actor performing the action. On each image, we estimate the BB by using a simple thresholding method based on that given on [15], approximating size and

\*This work has been granted by TIC2005-01665 project and Consolider Ingenio MIPRCV project of the Spanish Minister of Science and Innovation.



**Fig. 2. Examples of aHOF for different actions.** Top row shows the optical flow estimated for the displayed frame. Bottom row represents the aHOF descriptor computed for the subsequence of 20 frames around that frame.

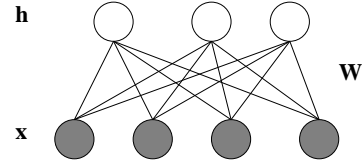
mass center, and smoothed along the sequence. BBs proportional to the relative size of the object in the image, and large enough to enclose the entire person, regardless of his pose, have been used (Fig. 1.a). All the frames are scaled to the same size  $40 \times 40$  pixels. Then the Farneback’s algorithm [16] is used to estimate the optical flow value on each pixel.

The idea of using optical flow features from the interior of the bounding box was firstly suggested in [2], although here we use it to propose a different image descriptor. The optical flow from each frame is represented by a set of *orientation*  $\times$  *magnitude* histograms (HOF) from non-overlapped regions (grid) of the cropped window. Each optical flow vector votes into the bin associated to its magnitude and orientation. The sequence-descriptor, named *aHOF* (accumulated Histogram of Optical Flow), is a normalized version of the image descriptor accumulated along the sequence. Therefore, a bin  $(i, j, k)$  of a aHOF  $H$  is computed as:  $H(l_i, o_j, m_k) = \sum_t H^t(l_i, o_j, m_k)$ , where  $l_i$ ,  $o_j$  and  $m_k$  are the spatial location, orientation and magnitude bins, respectively, and  $H^t$  is the HOF computed a time  $t$ . The normalization is given by each orientation independently on each histogram (see Fig. 1.b), what implies  $\sum_k H(l_i, o_j, m_k) = 1$ . Here each bin is considered a binary variable which value is the probability of taking value 1.

In practice, we associate multiple descriptors to each observed sequence, that is, one aHOF-descriptor for each subsequence of a fixed number of frames. Fig. 2 shows the aHOF representation for different actions in KTH database. The descriptor has been computed from a window of 20 frames around the displayed frame.

### 3. RBM AND ITS GENERALIZATIONS

A Restricted Boltzmann Machine (RBM) is a Boltzmann Machine having a bipartite connectivity graph between the visible variable  $\mathbf{x}$ -vector and the hidden variable  $\mathbf{h}$ -vector respectively (see 3). In RBM the probabilities between visible (hidden) variables are conditional independent given the hidden (visible) with expressions given by  $p(\mathbf{h}_i|\mathbf{x}) = \sigma(c_i + \mathbf{W}_i\mathbf{x})$  and  $p(\mathbf{x}_j|\mathbf{h}) = \sigma(b_j + \mathbf{W}_j\mathbf{h})$  where  $\sigma(x) = (1 + e^{-x})^{-1}$  is the logistic sigmoidal function and  $\mathbf{W}_i$  and  $\mathbf{W}_j$  represent



**Fig. 3. Restricted Boltzmann Machine.** Example of a RBM with 4 observed and 3 hidden units.

the  $i$ -row and  $j$ -column respectively of the model parameter  $\mathbf{W}$ -matrix. The vectors  $\mathbf{b}$  and  $\mathbf{c}$  represent the bias associated to the visible and hidden units respectively. In RBMs the equilibrium distribution  $p(\mathbf{x}, \mathbf{h})$  can be estimated using a very long Gibbs sampling alternating  $p(\mathbf{h}|\mathbf{x}, W)$  and  $p(\mathbf{x}|\mathbf{h}, W^T)$  starting from the visible variables clamped on the data.  $T$  denotes transpose.

Learning RBMs maximizing the gradient log-likelihood needs of averaging from the equilibrium distribution  $p(\mathbf{x}, \mathbf{h})$  what means a prohibitive cost. The Contrastive Divergence (CD) criteria proposed by Hinton [9], only needs to get samples from the data distribution  $p_0$ , and the one step Gibbs sampling distribution  $p_1$ , what implies an affordable cost. The parameter updating equations give updating values proportional to averages difference from these two distributions. That is,  $\Delta w_{ij} \propto \langle v_i h_j \rangle_{p_0} - \langle v_i h_j \rangle_{p_1}$  where  $\langle v_i h_j \rangle$  means average (using the subindex distribution) of the number of times that hidden unit  $j$  is on for the visible variable  $i$ . The equations for the bias  $b_i$  and  $c_j$  are similar.

A different strategy for training a two layers model, called autoencoder, is to learn the parameter vector  $(\mathbf{W}, \mathbf{b}, \mathbf{c})$  reconstructing each sample from the hidden layer. In this case the training criteria minimizes the negative log-likelihood of the reconstruction. In the binary case, it is given by

$$NLLR = - \sum_i p_i \log \tilde{p}_i - (1 - p_i) \log(1 - \tilde{p}_i)$$

where  $p_i$  denotes the sample and  $\tilde{p}_i$  its reconstruction .

#### 3.1. Other RBM models

**RBM with Nonlinear NCA.** Salakhutdinov and Hinton [11] proposed to estimate the weights  $W$  by minimizing the  $O_{NCA}$  criteria in order to define a good distance for non-parametric classifiers:

$$O_{NCA} = \sum_{a=1}^N \sum_{b:c^b=k} p_{ab} \quad (1)$$

$$p_{ab} = \frac{\exp(-\|f(\mathbf{x}^a|W) - f(\mathbf{x}^b|W)\|^2)}{\sum_{z \neq a} \exp(-\|f(\mathbf{x}^a|W) - f(\mathbf{x}^z|W)\|^2)} \quad (2)$$

where  $f(x|W)$  is a multi-layered network parametrized by the weight vector  $W$ ,  $N$  is the number of training samples, and  $c^b$  is the class label of sample  $b$ .



**Fig. 4. KTH database.** Examples of different actions, actors and scenarios.

**Discriminative RBM.** Larochelle and Bengio [10] propose the DRBM architecture to learn RBM using a discriminative approach. They add the label  $y$  to the visible data layer and models the following distribution:

$$p(y, \mathbf{x}, \mathbf{h}) \propto \exp \{E(y, \mathbf{x}, \mathbf{h})\} \quad (3)$$

where,  $E(y, \mathbf{x}, \mathbf{h}) = -\mathbf{h}^T \mathbf{W} \mathbf{x} - \mathbf{b}^T \mathbf{x} - \mathbf{c}^T \mathbf{h} - \mathbf{d}^T \vec{y} - \mathbf{h}^T \mathbf{U} \vec{y}$  with parameters  $\Theta = (\mathbf{W}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{U})$  and  $\vec{y} = (1_{y=i})_{i=1}^C$  for  $C$  classes. Two objective functions can be used with this model:

$$O_{gen} = -\sum_{i=1}^N \log p(y_i, \mathbf{x}_i); O_{disc} = -\sum_{i=1}^N \log p(y_i | \mathbf{x}_i) \quad (4)$$

where  $O_{gen}$  is the cost function for a generative model, and  $O_{disc}$  is the cost function for a discriminative model. Both cost functions can be combined in a single one (hybrid):

$$O_{hybrid} = O_{disc} + \alpha O_{gen} \quad (5)$$

Semisupervised training can be performed with DRBM models by using the following cost function:

$$O_{semi} = O_{disc} + \beta \left( -\sum_{i=1}^N \log p(\mathbf{x}_i) \right). \quad (6)$$

where  $O_{disc}$  is applied only to the labelled samples.

## 4. EXPERIMENTS AND RESULTS

We carry out experiments on KTH’s video sequence database. This database contains a total of 2391 sequences, where 25 actors performs 6 classes of actions (walking, jogging, boxing, hand clapping and hand waving). The sequences were taken in 4 different scenarios: outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3) and indoors (s4). Some examples are shown in Fig.4. As in [4], we split the database in 16 actors for training and 9 for test. In our experiments, we consider 5 different datasets: each one of the 4 scenario and the mixture of the 4 scenarios as the fifth one. In this way we make our results comparable with others appeared in the literature. In all the following experiments, we take subsequences of length 20 from the full length sequences. Each full sequence is classified by using majority voting on the independently classified subsequences. All the results reported in this paper are on the full length sequences.

	$L$	AE	DG	DRBM	NCA
S1	12	62.4	86.4	94.4	71.7
	128	94.1	94.4	94.5	95.4
	256	<b>94.6</b>	93.5	94.3	<b>95.7</b>
	512	94.4	94.4	94.6	95.5
S2	12	75.6	71.6	95.8	70.7
	128	<b>94.0</b>	92.8	95.6	<b>96.5</b>
	256	93.0	91.7	95.4	96.1
	512	93.8	90.8	95.6	96.0
S3	12	55.9	73.4	86.9	59.8
	128	89.3	89.7	87.0	92.2
	256	90.6	89.7	87.7	93.3
	512	<b>90.6</b>	90.5	87.9	<b>93.6</b>
S4	12	76.3	77.6	95.4	76.6
	128	73.2	92.0	96.3	94.1
	256	<b>93.7</b>	91.9	<b>96.4</b>	96.0
	512	<b>93.7</b>	92.0	<b>96.4</b>	95.3

**Table 1. Unsupervised vs Supervised.** This table shows a classification comparative between an unsupervised trained model (AE) y three different supervised models (DG,DRBM,NCA)(see text). In bold, the best results for all code vectors.

### 4.1. Evaluation of aHOF descriptor

For all our experiments, we compute the aHOF descriptors with 8 bins for orientation and 4 bins for magnitude<sup>1</sup>. We used a KNN classifier on subsequences of length 20, and three different grid configurations:  $2 \times 1$ ,  $4 \times 2$  and  $8 \times 4$  in order to define the best grid size for aHOF. The  $8 \times 4$  configuration provides the best results. With this experimental setup, the global classification performance on the four scenarios is 94.5%, when using 5NN on the subsequences. This result outperforms the state-of-the-art on KTH database (e.g. 91.8% Laptev *et al.* [6] or 90.5% Fathi&Mori [7]).

### 4.2. Experimental Results

#### Unsupervised and Supervised Code Learning.

In this experiment, we are interested in evaluate four different feature encoding architectures: (i) a RBM trained as an autoencoder (unsupervised) (denoted *AE*); (ii) a DRBM model trained in a supervised way using  $O_{gen}$  cost function (eq. 4) (denoted *DG*); (iii) a DRBM model trained by using  $O_{disc}$  cost function (eq. 4); and, (iv) a RBM trained with objective function NCA (eq. 2). We try different length codes (hidden units), from 12 up to 512 (half of the original vector dimensionality). In table 1, we show a comparative of the classification results using an 1NN classifier on the codes generated by the different models. It is remarkable that the maximum scores, in bold, for the four scenarios belong to only one code length. This points out the need of a minimum number of units to represent the data complexity.

<sup>1</sup>The magnitude intervals are  $(-\infty, 0.5]$ ,  $(0.5, 1.5]$ ,  $(1.5, 2.5]$ ,  $(2.5, \infty)$

	6	12	128	256	512	1024
4u+4l	68.1	84.8	<b>94.8</b>	94.1	94.4	95.0
8u+4l	61.3	67.4	<b>93.2</b>	92.5	92.0	92.9
12u+4l	49.0	60.5	<b>93.2</b>	89.6	91.2	92.0

**Table 2. Semi-supervised learning with DRBM.** Four labeled plus 4, 8 and 12 unlabeled actors, on scenario 1.

### Semisupervised Learning.

Table 2 shows how the classification performance changes with the proportion of unlabelled actors added during training. Results are averaged on 5 different training/test sets.  $\beta = 0.1$ . We have selected scenario 1 for this experiment.

As can be seen in the results, in order to use shorter codes we need a greater proportion of labelled data than the one needed for larger codes. Note that with 128-codes the model seems to reach a maximum in the performance.

	12	128	256	512
DRBM	68.7	93.9	93.3	92.1
Hyb	91.9	92.5	92.6	92.6

**Table 3. Hybrid DRBM vs Semisupervised DRBM.** First row is a semisupervised DRBM, and second row is an Hybrid DRBM.

### Hybrid DRBM VS Semisupervised DRBM

Table 3 shows a comparative between a supervised Hybrid-DRBM (eq. 5) and a semisupervised DRBM (eq. 6). In both cases, only 8 labelled actors are used for training. In the semisupervised case, 8 extra unlabelled actors are used. Test is done on the remaining 9 actors. Results are averaged on 5 different training/test sets. For the hybrid model  $\beta = 0.01$  is used, and  $\alpha = 0.1$  is used for the semisupervised model. We have selected scenario 1 for this experiment.

## 5. CONCLUSIONS

In this paper, RBM models trained with different cost functions are compared on the HAR task. As expected, table 1 shows that the supervised learning obtains the best global results, but the main goal of this paper is the unsupervised and semi-supervised cases. In fact, for NCA-256, the average classification performance on the four scenarios (95.3%) outperforms the state-of-the-art on KTH. The main conclusion from tables 3-2 is that the length of the hidden layer is very important when using unsupervised samples in the training set. In semi-supervised learning this length can be shorten according to the proportion of labeled samples in the learning set. In all the experiments, the scores associated with the length ( $L$ ) show one local maximum, although we do not have a clear explanation for this fact, we think that this length represents the shortest one that explains the data complexity. In this way, these results contrast with theoretical results where the longer the hidden layer the better the data is represented.

Clearly, the estimation of the shortest hidden layer needed to encode the data with unsupervised or semi-supervised learning techniques, remains an interesting open problem for these architectures.

## 6. REFERENCES

- [1] C. Rao and M. Shah, "View-invariance in action recognition," in *CVPR*, 2001, vol. 2, pp. 316–322. 1
- [2] A.A. Efros, A.C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *Int. Conf. Comp. Vision*, 2003, vol. 2, pp. 726–733. 1, 2
- [3] E. Shechtman and M. Irani, "Space-time behavior-based correlation or How to tell if two underlying motion fields are similar without computing them?," *IEEE PAMI*, vol. 29, no. 11, pp. 2045–2056, 2007. 1
- [4] Christian Schüldt, Ivan Laptev, and Barbara Caputo, "Recognizing human actions: A local SVM approach," in *Proc. ICPR*, Cambridge, U.K., 2004, vol. 3, pp. 32–36. 1, 3
- [5] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *2nd IEEE Workshop VS-PETS*, 2005, pp. 65–72. 1
- [6] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. CVPR'08*, 2008. 1, 3
- [7] A. Fathi and G. Mori, "Action recognition by learning mid-level motion features," in *CVPR*, 2008. 1, 3
- [8] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006. 1
- [9] G.E. Hinton, "Training product of experts by minimizing contrastive divergence.," *Neural Computation*, vol. 14(8), pp. 1711–1800, 2002. 1, 2
- [10] Hugo Larochelle and Yoshua Bengio, "Classification using discriminative restricted boltzmann machines," in *Proc. ICML*, 2008. 1, 3
- [11] R. Salakhutdinov and G. Hinton, "Learning a non-linear embedding by preserving class neighbourhood structure.," in *AI and Statistics*, 2007. 1, 2
- [12] G.E. Hinton, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313(5786), pp. 504–507, 2006. 1
- [13] A. Torralba, R. Fergus, and Y. Weiss, "Small codes and large database for recognition," in *CVPR*, 2008. 1
- [14] Jun Yang, Rong Yan, Yan Liu, and Eric P. Xing, "Harmonium models for video classification," *Stat. Anal. Data Min.*, vol. 1, no. 1, pp. 23–37, 2008. 1
- [15] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Systems, Man and Cybernetics*, vol. 9, pp. 62–66, 1979. 1
- [16] Gunnar Farneback, "Two-frame motion estimation based on polynomial expansion," in *Proc. of the 13th Scandinavian Conf. on Image Analysis*, June-July 2003, vol. 2749 of *LNCS*, pp. 363–370. 2