HUMAN ACTION RECOGNITION USING DIRECTION AND MAGNITUDE MODELS OF MOTION

Yassine Benabbas, Samir Amir, Adel Lablack, Chabane Djeraba
LIFL UMR CNRS 8022, University of Lille1, TELECOM Lille1
IRCICA, Parc de la Haute Borne, 56950 Villeneuve dAscq
{*yassine.benabbas,samir.amir,adel.lablack,chabane.djeraba*}*@lifl.fr*

Keywords: Human action recognition, Motion analysis, Video understanding

Abstract: This paper proposes an approach that uses direction and magnitude models to perform human action recognition from videos captured using monocular cameras. A mixture distribution is computed over the motion orientations and magnitudes of optical flow vectors at each spatial location of the video sequence. This mixture is estimated using an online k-means clustering algorithm. Thus, a sequence model which is composed of a direction model and a magnitude model is created by circular and non-circular clustering. Human actions are recognized via a metric based on the Bhattacharyya distance that compares the model of a query sequence with the models created from the training sequences. The proposed approach is validated using two public datasets in both indoor and outdoor environments with low and high resolution videos.

1 Introduction

Human action recognition and understanding is a challenging topic in computer vision. It consists in the automatic labeling of actions or activities performed by a human being in a video sequence. Human action recognition is important in a lot of domains. It is widely used for video-surveillance in public areas such as malls, metro stations or airports in order to detect abnormal events. In addition, disabled and aged people can be more efficiently aided by monitoring their actions using an event detection system. Automatic labeling of actions is used to improve human-computer interactions and video retrieval applications such as searching for fight scenes in action movies or goals in soccer videos.

This paper presents an approach for human action recognition from videos. The goal is to recognize simple daily life actions (e.g. walking, answering a phone, etc.) in a video sequence. These actions consist of motion patterns performed by a single person over a short period of time. Some approaches detect actions from still images, while other approaches use as input stereoscopic videos or 3D motion data (Ganesh and Bajcsy, 2008) . In this work, we use video sequences to detect actions by combining spatial and temporal information (Johansson et al., 1994). We focus on monocular videos since they are widespread and challenging.

The common approaches extract a set of image features from the video sequence. Then, these features are used classify the actions using training data. The selection of the image representation and the classification algorithms is influenced by the number and type of actions, as well as the environment and recording settings. In our approach we extract major motion orientations and magnitudes at each location of the scene using Gaussian mixtures and mixtures von Mises distributions. The von Mises distribution was recently applied for trajectory shape analysis (Prati et al., 2008) and event detection in video surveillance (Djeraba et al., 2010). These distributions form the direction and magnitude models. We define then a distance metric between models to recognize the actions from training videos.

This paper is organized as follows: we highlight in Section 2 the relevant works for human action recognition. In Section 3, we describe our approach which is composed of *models creation* and *action recognition* stages. We present and discuss the experimental results of our approach in Section 4. Finally, we conclude and outline potential future work in Section 5.

2 Related Work

Over the recent years, many techniques have been proposed for human action recognition and understanding that are described in comprehensive surveys (Poppe, 2010; Turaga et al., 2008). We classify these techniques according to the image representation and the action classification algorithms that have been used.

Image representation: it describes how the features extracted from the video sequences are represented. These features consist generally in optical flow vectors (Ali and Shah, 2010), hollistic features (Kosmopoulos and Chatzis, 2010; Sun et al., 2009), local spatio-temporal features such as the cuboid features (Dollar et al., 2005) or the Hessian features (Willems et al., 2008). A descriptor is then constructed to represent the video sequence. It can be done by training Ada-boost classifiers over low-level features (Fathi and Mori, 2008), using motion-sensitive responses to model motion contrasts (Escobar et al., 2009), analyzing trajectories of moving points (Messing et al., 2009) or spatio-temporal descriptors such as HOG/HOF (Laptev et al., 2008), HOG3D (Klser et al., 2008) or the extended SURF (ESURF) (Willems et al., 2008).

Action classification: it consists in finding the correct action associated to a query video. The classification can be performed using a classifier such as SVM (Mauthner et al., 2009), Hidden Markov Models (HMM) (Ivanov and Bobick, 2000; Kosmopoulos and Chatzis, 2010), Self Organizing Map (SOM) (Huang and Wu, 2009) or Gaussian Process (Wang et al., 2009b), a distance function such as a transferable distance function (Yang et al., 2009) or a discriminative model such as a Hidden Conditional Random Field (HCRF) (Zhang and Gong, 2010) to label the sequences as a whole.

Several datasets are available such as KTH (Laptev and Lindeberg, 2004) and Activities of Daily Living (ADL) (Messing et al., 2009), in order to train a classifier or to compare different approaches.

Local spatio-temporal features have recently become popular and have been shown successful for human action recognition (Wang et al., 2009a). Our models are inspired by the HOG/HOF features (Laptev et al., 2008) that extract only the major motion orientations/magnitudes and attributes them a variance and a weight instead of coarse histograms which are frequencies of the observations over intervals. Our approach has the originality of using the direction and magnitude models to represent the actions without human body-part detection. Indeed, it relies on the optical flow vectors as a feature to construct the sequence model that is estimated and updated in real time using an online algorithm. The model extracts major motion orientations and magnitudes at each block of the scene. We choose this dense representation for the models since this kind of sampling generally outperforms other sampling methods (Wang et al., 2009a). The actions are then recognized using a distance metric between the model associated to the template sequences and the model of a query sequence.

3 Approach Description

We propose in the following an approach that recognizes actions performed by a single person. Figure 1 shows the main steps, divided into two major stages:

- Models creation: quantifies the motion using the optical flow vectors in order to estimate a direction model and a magnitude model over motion orientations and magnitudes for the whole video sequence.
- Action recognition: recognizes the action associated to a query video by comparing its sequence model with the sequence models of template videos using a distance metric.

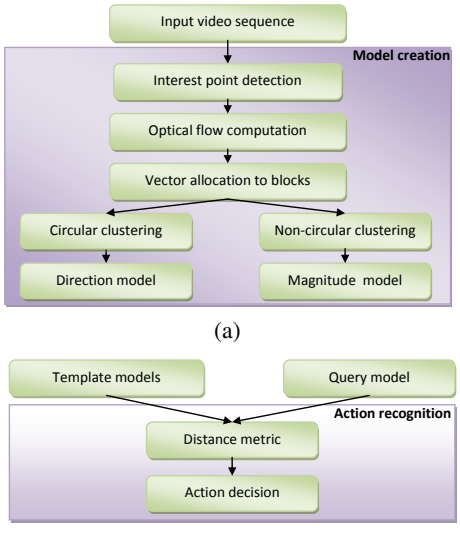


Figure 1: Approach steps. (a)Models creation stage, (b)Action recognition stage.

3.1 Models Creation

In order to create the model of a video sequence, we start by extracting a set of interest points from each input frame. We consider Shi and Tomasi feature detector (Shi and Tomasi, 1994) which finds corners with high eigenvalues in the frame. We also consider that, in our targeted video scenes, camera positions and lighting conditions allow a large number of interest points to be captured and tracked easily.

Once we define the set of points of interest, we track them over the next frames using optical flow vectors. We have used Bouguet's (Bouguet, 2000) implementation of the KLT tracker (Lucas and Kanade, 1981) which is efficient in handling features near the image border and is computationally efficient. The result of the operation of matching features between frames is a set of four-dimensional vectors V:

$$V = \{V_1 ... V_N | V_i = (X_i, Y_i, A_i, M_i)\}$$

where

- X_i and Y_i are the image location coordinates of the feature i,
- A_i : is the motion direction of the feature i.
- M_i : is the motion magnitude of the feature i and it corresponds to the distance between the position of feature i in the frame f and its corresponding posistion in the frame $f+1$,

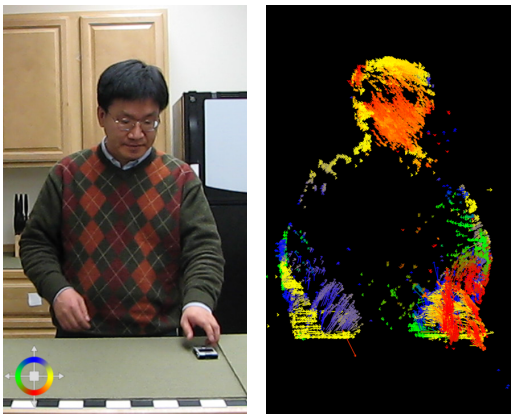
This step also allows the removal of static and noise features. Static features are the features that move less than a minimum magnitude. By contrast, noise features are features whose magnitudes exceed the threshold. In our experiments, we set the minimum motion magnitude to 1 pixel per frame and the maximum to 20 pixels per frame.

The next step consists in dividing the scene into a grid of $W \times H$ blocks. Then, each vector is allocated to its corresponding block depending on its origin. The size of the block affects the precision of the system and will be discussed in Section 4.3.

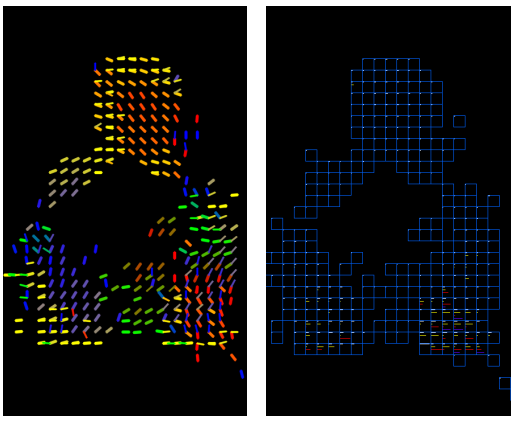
Then, we apply a circular clustering algorithm to the orientations of the optical flow vectors for each block. The set of $W \times H$ estimated circular distributions is called the direction model. Figure 2 illustrates the construction of the direction model associated to an 'answerPhone' action.

In this work, we cluster circular data using a mixture of von Mises distribution. Thus, the probability of an orientation θ with respect to the block $B_{x,y}$ is defined by:

$$p_{x,y}(\theta) = \sum_{i=1}^{K} \psi_{i_{x,y}} \cdot V\left(\theta; \phi_{i_{x,y}}, \gamma_{i_{x,y}}\right)$$



(a) (b)



(c) (d)

Figure 2: Direction model of an 'answerPhone' action. (a) current frame, (b) Optical flow vectors, (c) Direction model of the video sequence, (d) Magnitude model of the video sequence.

where K is the number of distributions and represents the maximum number of major orientations to consider (we choose empirically $K = 4$ which corresponds to the 4 cardinal directions). $\psi_{i_{x,y}}$, $\phi_{i_{x,y}}$, $\gamma_{i_{x,y}}$ denote respectively the weight, mean angle and dispersion of the i^{th} distribution for the block $B_{x,y}$. The von Mises distribution $V(\theta; \phi, \gamma)$ with mean orientation ϕ and dispersion parameter γ, over the angle θ, has the following probability density function:

$$V(\theta; \phi, \gamma) = \frac{1}{2\pi I_0(\gamma)} \exp\left[\gamma \cos(\theta - \phi)\right]$$

where $I_0(\gamma)$ is the modified Bessel function of the first kind and order 0 defined by the following equation:

$$I_0(\gamma) = \sum_{r=0}^{\infty} \left(\frac{1}{r!}\right)^2 \left(\frac{1}{2}\gamma\right)^{2r}$$

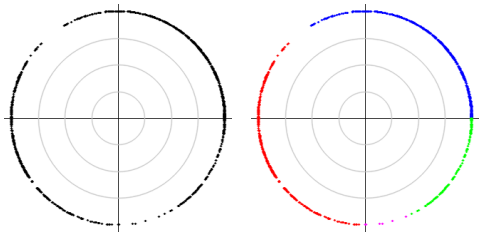
By analogy, we cluster the magnitudes of the optical flow vectors for each block using Gaussian mixtures. The set of estimated Gaussian mixtures constitutes the magnitude model as illustrated in Figure 2(d). Thus, the probability of a magnitude v with respect to the block $B_{x,y}$ is defined by:

$$p_{x,y}(v) = \sum_{i=1}^{J} \omega_{i_{x,y}} G(v; \mu_{i_{x,y}}, \sigma^2_{i_{x,y}})$$

where $\omega_{i_{x,y}}, \mu_{i_{x,y}}, \sigma^2_{i_{x,y}}$ are respectively the weight, mean and variance of the i^{th} Gaussian distribution.

For each frame, we update the Gaussian mixture parameters using a K-means approximation algorithm described in (Kaewtrakulpong and Bowden, 2001). We use it also to update the parameters of the mixture of von Mises distributions by adapting the algorithm in order to deal with circular data and considering the inverse of the variance as the dispersion parameter; $\gamma = 1/\sigma^2$. The circular clustering algorithm is given below and works as follows. It gets as input a data point x which is an orientation in our case. Then this orientation is matched against the first distribution in the mixture. If no distribution satisfies the matching condition, then the last distribution is replaced by a distribution with a mean equal to the new orientation. After that, the parameters of all the distributions in the mixture are updated. Finally, the distribution are sorted according to a fitness. This last action allows to define the order of the distributions for the next matching.

Figure 3 shows the detected clusters in block using our algorithm. We remind that our circular clustering algorithm did not process the hole data illustrated in Figure 3(a) in a single run. In fact, the algorithm is run at each frame and clusters are updated as data is added to the block. Thus, the temporal dimension impacts the final results. This explains the different clusters even if the hole data may be assimilated to a single cluster if we do not consider the temporal dimension.



(a) (b)

Figure 3: Representation of the estimated clusters using our circular clustering algorithm (a) accumulated raw data, (b) estimated clusters.

Algorithm 1 online-mVM

input a data point x on $\mathbb{R}$
a mixture of K vM distributions
return an updated clustering over a mixture of K von Mises distributions
initialize learning rate α = 1/400
initialize matching threshold β = 2.5
$c \leftarrow 0$
for $i = 1$ to K **do**
  {Getting the first match for the input value}
  if $c = 0$ and $x - \theta_i \leq \beta^2/\gamma_i$ **then**
    $c \leftarrow i$
  end if
end for
if $c \neq 0$ **then**
  {a match is found, update the parameters}
  for $i = 1$ to K **do**
    $\psi_i \leftarrow \psi_i(1-\alpha)$
  end for
  $\psi_c \leftarrow \psi_c + \alpha$
  $\rho \leftarrow \alpha\psi_c(x - \theta_c)$
  $\theta_c \leftarrow \theta_c + \rho$
  $\gamma_c \leftarrow (\gamma_c^{-1} + \rho^2 - \gamma_c^{-1})^{-1}$
  $n_c \leftarrow n_c + 1$
else
  {no match found, discard the last distribution}
  $n_k \leftarrow 1$
  $\theta_k \leftarrow x$
  $\gamma_k \leftarrow \gamma_0$
  for $i = 1$ to K **do**
    $\psi_i \leftarrow \frac{n_i}{\sum_{j=1}^{k} n_j}$
  end for
end if
sort distributions by $weight \times dispersion$

In the following, we note the model of the sequence s by $Sm(s) = (Dm(s), Mm(s))$, where $Dm(s)$ and $Mm(s)$ are respectively the direction model and the magnitude model associated to the sequence s. Figure 7 shows the direction and magnitude models of some video sequences from the KTH dataset.

3.2 Action Recognition

Once the model of a video sequence has been computed, we detect the action that corresponds to this 'query' video by comparing its model with the models of the template sequences using a distance metric. The action associated to the model that has the shortest distance with the model of the query sequence is then selected.

Let $T = \{t_1, t_2, ..., t_n\}$ be a set of n template sequences and their respective models are $\{Sm(t_1), Sm(t_2), ..., Sm(t_n)\}$. Given a query sequence q with its model $Sm(q)$, the distance between $Sm(q)$ and a template sequence model $Sm(t_l)$ is defined by:

$$D(Sm(q),Sm(t_l)) = Norm(A_{Dm(q),Dm(t_l)}) + Norm(B_{Mm(q),Mm(t_l)})$$

where $Norm$ corresponds to the L2-Norm. The $W \times H$ matrices $A_{Dm(q),Dm(t_l)}$ and $B_{Mm(q),Mm(t_l)}$ contain the distances between each element of the two direction models $Dm(q)$ and $Dm(t_l)$ and the two magnitude models $Mm(q)$ and $Mm(t_l)$ respectively. Each element $A_{M,M'}(x,y)$ is defined by the following formula:

$$A_{M,M'}(x,y) = \sum_{i=1}^{K} \left(\psi_{i_{x,y}} \psi'_{i_{x,y}} Dist_d(V_{i_{x,y}}, V'_{i_{x,y}}) \right)$$

where $\psi_{i_{x,y}}$ (resp. $\psi'_{i_{x,y}}$) and $V_{i_{x,y}}$ (resp. $V'_{i_{x,y}}$) are the i^{th} weight and i^{th} von Mises distribution associated to the direction model M (resp. M') at the block $B_{x,y}$. $Dist_d(V,V')$ is the Bhattacharyya distance between two von Mises distributions V and V' defined by the following equation:

$$Dist_d(V,V') = \sqrt{1 - \int_{-\infty}^{+\infty} \sqrt{V(\theta)V'(\theta)} d\theta}$$

where $0 \leq Dist_d(V,V') \leq 1$. This equation can be computed using the closed form expression:

$$Dist_d(V,V') = \sqrt{1 - \sqrt{\frac{1}{I_0(\gamma)I_0(\gamma')}} I_0\left(\frac{\sqrt{\gamma^2 + \gamma'^2 + 2\gamma\gamma' cos(\phi - \phi')}}{2} \right)}$$

where ϕ (resp. ϕ') and γ (resp. γ') are respectively the mean angle and the dispersion parameter of the distribution V (resp. V').

By analogy, we define each element $B_{N,N'}(x,y)$ by the following equation:

$$B_{N,N'}(x,y) = \sum_{i=1}^{K} \left(\omega_{i_{x,y}} \omega'_{i_{x,y}} Dist_m(G_{i_{x,y}}, G'_{i_{x,y}}) \right)$$

where $\omega_{i_{x,y}}$ (resp. $\omega'_{i_{x,y}}$) and $G_{i_{x,y}}$ (resp. $G'_{i_{x,y}}$) are the i^{th} weight and Gaussian distribution associated to the magnitude model N (resp. N') at the block $B_{x,y}$. $Dist_m(G,G')$ is the Bhattacharyya distance between two Gaussian distributions G and G' defined by the following closed form expression:

$$Dist_m(G,G') = \frac{(\mu - \mu')^2}{4(\sigma^2 + \sigma'^2)} + \frac{1}{2} \ln \left(\frac{\sigma^2 + \sigma'^2}{2\sigma\sigma'} \right)$$

where μ (resp. μ') and σ^2 (resp. σ'^2) are respectively the mean and the variance of the distribution G (resp. G'). We note that this step allows parallelization out of the box since computing $D(Sm(q),Sm(t_i))$ for any $i < n$ does require to compute $D(Sm(q),Sm(t_j)); j < n, j \neq i$.

We believe that our method can perform in real time because the *models creation* step is performed online and the *action recognition* step can be parallelized. However, our current implementation does yet not parallelize the processing.

4 Experiments and Results

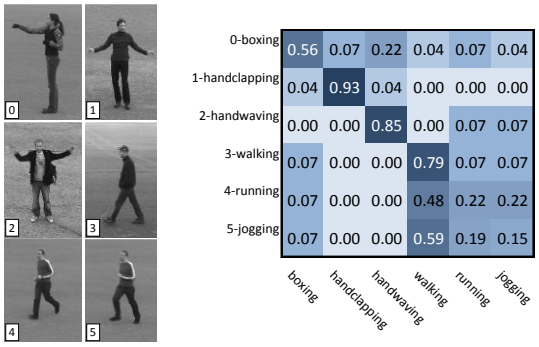
We demonstrate the performance of our approach using two standard datasets containing a variety of daylife actions. In addition to the confusion matrices, we also report the effect of different number of action classes and different block sizes to the efficiency and effectiveness of the system.

4.1 Action recogntion performance

KTH dataset (Laptev and Lindeberg, 2004): is a dataset that contains low resolution videos (gray-scale images with a resolution of 160×120 pixels) of 6 actions performed several times by 25 different subjects. This dataset is challenging because the sequences are recorded in different indoor and outdoor scenarios with scale variations and different clothes. We divide the dataset as suggested by Schuldt et al. (Schuldt et al., 2004) into two sets, a training set (16 people) and a test set (9 people). We include 'person01' to 'person16' in the training set and 'person17' to 'person25' in the validation set. We use a block size of 5×5 pixels.

Action samples and the confusion matrix are reported in Figure 4. Our approach is able to achieve satisfying results on the first three actions of this dataset where the human is motionless. However, our system considers the 'running' and 'jogging' actions as 'walking'; this is due to the fact that these actions differ slightly in speed and stride length and have similar orientations.

Activities of Daily Living (ADL) dataset (Messing et al., 2009): is a dataset that contains high resolution videos (1280×720 pixels) of 10 daily life actions (such as peelBanana, useSilverware, answerPhone) performed by 5 different subjects. We follow the leave-one-out experimentation protocol in our evaluation. It is performed by considering a sequence as



(a) (b)

Figure 4: Results on KTH dataset. (a) Action samples, (b) Confusion matrix using a block size of 5×5.

a query sequence, and all the remaining ones as template sequences for the recognition of an action. This procedure is performed for all sequences and the results are averaged for each action category.

In Figure 5, we present the confusion matrix obtained with our approach on this dataset. The approach achieved an average accuracy of 0.84 with a block size of 5×5 pixels. It is a very satisfying performance, however, the peelBanana action is confused with the actions eatSack and useSilverware.

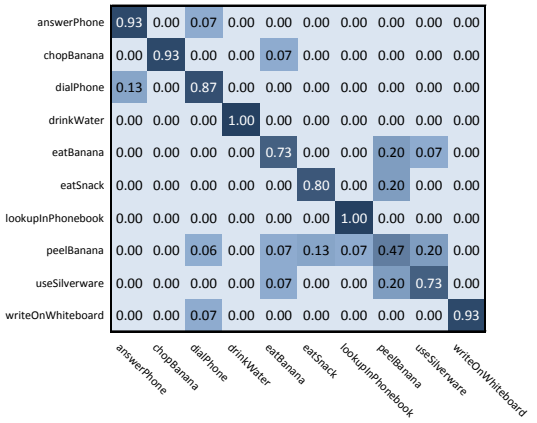
4.2 Comparative study

We compare our approach with other systems using KTH and Activities of Daily Living datasets and present their precision in Table 1. It shows that the approaches which are based on local spatio-temporal features (Dollar et al., 2005; Laptev et al., 2008) and velocity histories (Messing et al., 2009) outperform our system on the KTH dataset. The latter uses the velocities of tracked key-points as a low-level features. However, our system gets a better precision on the Activities of Daily Living (ADL) dataset because it combines both motion magnitude and orientation information.

Method	ADL	KTH
Our proposed approach	0.84	0.58
Velocity histories (Messing et al., 2009)	0.63	0.74
Space-time interest Points (Laptev et al., 2008)	0.59	0.80
Spatio-temporal Cuboids (Dollar et al., 2005)	0.36	0.66

Table 1: Comparison of classifying precision on 2 different datasets.

(a)



(b)

Figure 5: Results on ADL dataset. (a) Action samples, (b) Confusion matrix using a block size of 5×5.

Compared to the HOG/HOF features (Laptev et al., 2008), our scene model learns major motion orientations/magnitudes and does not consider noisy motion. In addition each mixture distribution returns exact mean orientations with their variances and weight. While the HOG/HOF features compute coarse histograms of oriented gradients (HOG) and optical flow (HOF). In addition, a histogram computes frequencies over intervals, which is less precise than our method since it attributes mean values as in our scene model. Our approach has better results on the high resolution ADL dataset since the motion information is more precise. However, our approach suffered from the the lack of precision and the frequent noise in low resolution videos.

Some other approaches achieve good results on the KTH dataset but we cannot compare to them because of their different setup. They have either used more training data or subdivided the problem into simpler tasks.

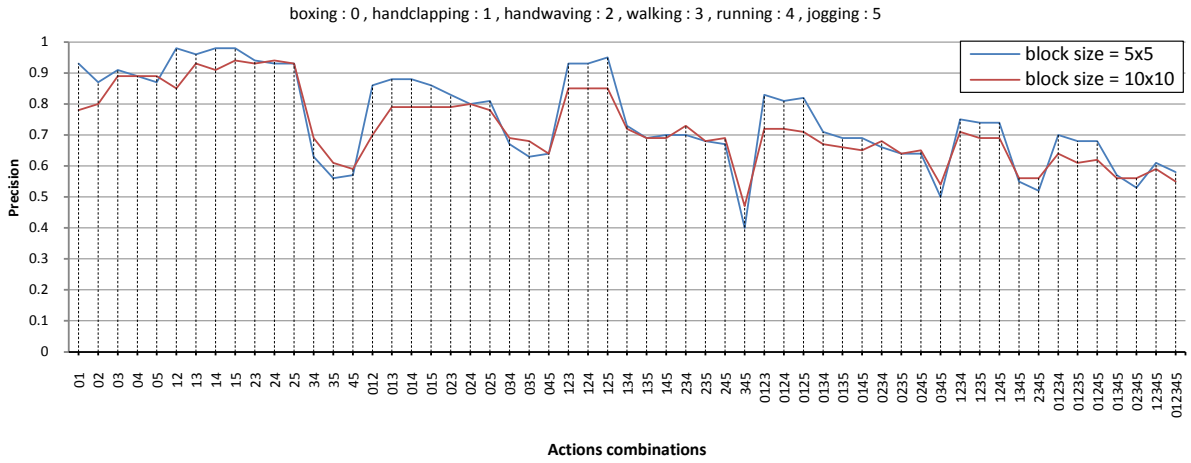


Figure 6: Influence of the block size and actions combinations on the precision.

4.3 Performance on action powerset

We study the influence of the block size and the number of action classes using the KTH dataset. Thus, we have repeated the experimentation on each element of the powerset of the KTH set of actions : handwaving, boxing, handclapping, walking, running and jogging, which we note $A = \{0,1,2,3,4,5\}$ respectively. The graphs in Figure 6 show the precision of our system for each subset of A. The blue graph is obtained using a block size of 5×5 pixels while the red graph is obtained using a block size of 10×10 pixels.

The lowest precision rate $(\sim 40\%)$ is reached with the combination 345 which corresponds to the actions jogging, running and walking. It highlights the difficulty to distinguish between the speed of related actions in low resolution videos.

Our experiments show also that increasing the block size reduces the precision of the action recognition system but decreases exponentially the processing time. In addition, increasing the number of template sequences increases the processing. However this does not necessarily imply increasing performance, because we reached 0.51 precision rate on the KTH dataset using the leave-one-out experimentation protocol as in the Activities of Daily Living (ADL) dataset.

5 Conclusion

We have presented an effective action recognition system that relies on direction and magnitude models. We have extracted optical flow vectors from video sequences in order to learn statistical models over motion orientations and magnitudes. The result is a sequence model that estimates major orientations and magnitudes in each spatial location of the scene. We have used a distance metric to recognize an action by comparing the model of a query sequence with the models of template sequences. Relying on motion orientations and magnitudes, our approach has shown promising results compared to other state-of-the art approaches in particular using high resolution videos. Our future work will focus on two directions: improving the flexibility of the classifier with respect to adding or removing action classes, and performing action detection for online applications.

ACKNOWLEDGEMENTS

This work has been supported by the Multi-modal Interfaces for Disabled and Ageing Society (MIDAS) ITEA 2-07008 European project and the French ANR project Comportemennts Anormaux Analyse Detection Alerte (CAnADA).

REFERENCES

Ali, S. and Shah, M. (2010). Human action recognition in videos using kinematic features and multipleinstance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(2):288–303.

Bouguet, J.-Y. (2000). Pyramidal implementation of the lucas kanade feature tracker description of the algorithm. Intel Corporation Microprocessor Research Labs.

Djeraba, C., Lablack, A., and Benabbas, Y. (2010). *Multi-*

Modal User Interactions in Controlled Environments. Springer-Verlag.

Dollar, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *2nd International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (PETS)*, pages 65–72.

Escobar, M.-J., Masson, G. S., Vieville, T., and Kornprobst, P. (2009). Action recognition using a bio-inspired feedforward spiking network. *International Journal of Computer Vision (IJCV)*, 82(3):284–301.

Fathi, A. and Mori, G. (2008). Action recognition by learning mid-level motion features. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ganesh, S. and Bajcsy, R. (2008). Recognition of human actions using an optimal control based motor model. In *Workshop on Applications of Computer Vision (WACV)*.

Huang, W. and Wu, J. (2009). Human action recognition using recursive self organizing map and longest common subsequence matching. In *Workshop on Applications of Computer Vision (WACV)*.

Ivanov, Y. and Bobick, A. (2000). Recognition of visual activities and interactions by stochastic parsing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):852 –872.

Johansson, G., Bergstrm, S. S., Epstein, W., and Jansson, G. (1994). *Perceiving Events and Objects*. Lawrence Erlbaum Associates.

Kaewtrakulpong, P. and Bowden, R. (2001). An improved adaptive background mixture model for real-time tracking with shadow detection. In *2nd European Workshop on Advanced Video Based Surveillance Systems*.

Klser, A., Marszaek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference (BMVC)*.

Kosmopoulos, D. and Chatzis, S. (2010). Robust visual behavior recognition, a framework based on holistic representations and multicamera information fusion. *IEEE Signal Processing Magazine*, 27(5):34 –45.

Laptev, I. and Lindeberg, T. (2004). Velocity adaptation of space-time interest points. In *International Conference on Pattern Recognition (ICPR)*, pages 52–56.

Laptev, I., Marszałek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lucas, B. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 674–679.

Mauthner, T., Roth, P. M., and Bischof, H. (2009). Instant action recognition. In *16th Scandinavian Conference on Image Analysis (SCIA)*.

Messing, R., Pal, C., and Kautz, H. (2009). Activity recognition using the velocity histories of tracked keypoints. In *International Conference on Computer Vision (ICCV)*.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing (IVC)*, 28(6):976 – 990.

Prati, A., Calderara, S., and Cucchiara, R. (2008). Using circular statistics for trajectory shape analysis. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1 –8.

Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions: A local svm approach. In *International Conference on Pattern Recognition (ICPR)*.

Shi, J. and Tomasi, C. (1994). Good features to track. In *Internatioal Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600.

Sun, X., Chen, M., and Hauptmann, A. (2009). Action recognition via local descriptors and holistic features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2009 (CVPR Workshops)*, pages 58 –65.

Turaga, P., Chellappa, R., Subrahmanian, V. S., and Udrea, O. (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488.

Wang, H., Ullah, M. M., Kläser, A., Laptev, I., and Schmid, C. (2009a). Evaluation of local spatio-temporal features for action recognition. In *British Machine Vision Conference*.

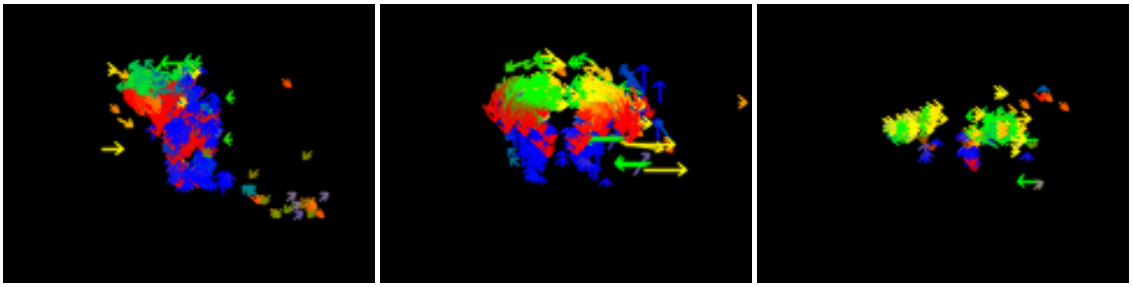
Wang, L., Zhou, H., Low, S.-C., and Leckie, C. (2009b). Action recognition via multi-feature fusion and gaussian process classification. In *Workshop on Applications of Computer Vision (WACV)*, pages 1 –6.

Willems, G., Tuytelaars, T., , and Gool, L. V. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. In *European Conference on Computer Vision (ECCV)*.

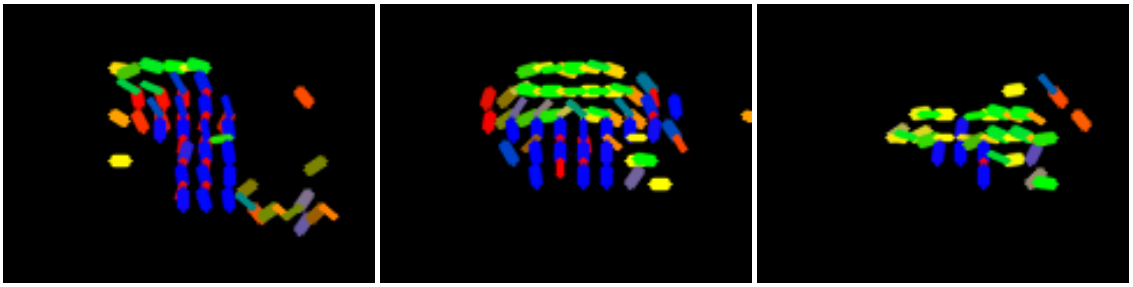
Yang, W., Wang, Y., and Mori, G. (2009). Efficient human action detection using a transferable distance function. In *Asian Conference on Computer Vision (ACCV)*.

Zhang, J. and Gong, S. (2010). Action categorization with modified hidden conditional random field. *Pattern Recognition (PR)*, 43(1):197–203.

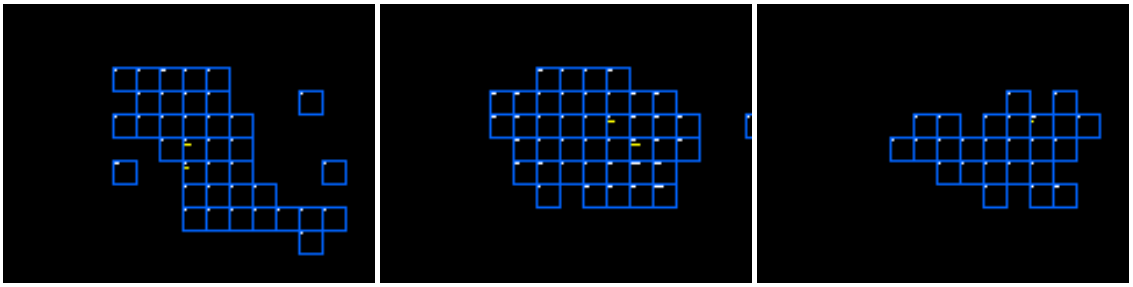
(a) Sample frames



(b) Optical flow vectors



(c) Direction model



(d) Magnitude model

Figure 7: Sample frames with associated direction models and magnitude models