3rd International Symposium on Visual Computing, Lake Tahoe, Nevada, USA , 2007

Boosting with temporal consistent learners: an application to human activity recognition.*

Pedro Canotilho Ribeiro, Plinio Moreno, and José Santos-Victor
{pribeiro, plinio, jasv}@isr.ist.utl.pt

Instituto Superior Técnico & Instituto de Sistemas e Robótica
1049-001 Lisboa - Portugal

Abstract. We present a novel boosting algorithm where temporal consistency is addressed in a short-term way. Although temporal correlation of observed data may be an important cue for classification (e.g. of human activities) it is seldom used in boosting techniques. The recently proposed Temporal AdaBoost addresses the same problem but in a heuristic manner, first optimizing the weak learners without temporal integration. The classifier responses for past frames are then averaged together, as long as the total classification error decreases.
We extend the GentleBoost algorithm by modeling time in an explicit form, as a new parameter during the weak learner training and in each optimization round. The time consistency model induces a fuzzy decision function, dependent on the temporal support of a feature or data point, with added robustness to noise. Our temporal boost algorithm is further extended to cope with multi class problems, following the JointBoost approach introduced by Torralba *et. al.* We can thus (i) learn the parameters for all classes at once, and (ii) share features among classes and groups of classes, both in a temporal and fully consistent manner.
Finally, the superiority of our proposed framework is demonstrated comparing it to state of the art, temporal and non-temporal boosting algorithms. Tests are performed both on synthetic and 2 real challenging datasets used to recognize a total of 12 different human activities.

1 Introduction

Although short-term temporal information may convey critical information for several classification problems, it is ignored by many classifiers. Nowadays, data for video applications are acquired at high frame rates in such a way that information changes smoothly in most of the cases. Human motion follows this behavior, thus generate similar motion data during several consecutive frames. An adequate model of this type of data consistency inside the classifier will improve the performance of any non-sequential classifier, that can be used, for instance, to recognize human activities in several different applications, e.g. surveillance, intelligent environments, human/robot interaction or interface.

* This work has been partially supported by Fundação para a Ciência e Tecnologia (Proj. 61911 - GESTINTERACT; plurianual funds through the POS_Conhecimento Program that includes FEDER funds) and EU Project URUS (IST-045062). Pedro Ribeiro also acknowledges the PhD grant SFRH/BD/18936/2004

When the temporal evolution of the features is essential like in human activity recognition, there are usually two types of classifiers used: (i) non-sequential, and (ii) sequential. Non-sequential classifiers aim to maximize the number of individual labels predicted correctly, encompassing the temporal dynamics in a short-term manner, usually in the feature computation step. On the other hand, sequential classifiers predict jointly the entire sequence of labels with the highest probability. An example of a non-sequential classifier is the Adaboost cascade for pedestrian detection using pairs of images to compute motion features [1]. In the case of sequential classifiers, a recent work [2] proposes a Conditional Random Field classifier trained with gradient tree boosting. Another approach is the correlation of data volumes to match activities in video, such as the spatio-temporal descriptor based on optical flow, proposed by Efros et.al. [3].

Opposed to most of the works, including the ones referred above, that use a temporal window fixed by hand, we derive a non-sequential classifier that learns the optimal temporal window. We rely on the GentleBoost algorithm [4], that can be defined as a forward stagewise approximate optimization of the exponential loss. We propose to include explicitly short-time consistency in GentleBoost, considering the non-sequential weak classifiers. The recently proposed TemporalBoost [5] also introduced temporal consistency in a boosting procedure, by averaging previous AdaBoost weak classifiers sequentially, while the classification error decreases. However, temporal support is considered in a heuristic procedure only after training the weak classifiers, and the TemporalBoost averaged output is mapped to a binary value. This allows to use the standard AdaBoost procedure [6], but discards the advantages of a fuzzy output.

In this paper we propose to model time directly during the weak classifiers training. As the basis of our work, we consider the gentleBoost algorithm using regression stumps as weak learners. A regression stump is similar to a single node binary tree, that selects a branch for a given feature according to a threshold using a binary decision function. Alternatively, we propose to compute a new decision function, by averaging the decision value in the learned temporal window. This procedure transforms the binary decision into a fuzzy one, according to the temporal window size. This new weak classifier based on a fuzzy decision function provides two main advantageous properties to boosting algorithms: (i) added noise robustness, and (ii) better performance.

In order to extend the binary classification framework to multiclass problems, we adapt JointBoosting [7] algorithm to fit our framework. JointBoost proposes to share weak classifiers among classes by selecting the group of classes which are going to share a feature, allowing to: (i) learn the strong classifiers of each class jointly, and (ii) reduce the number of weak learners needed to attain good performance.

2 GentleBoost with temporal consistent learners

The Boosting algorithm provides a framework to sequentially fit additive models in order to build a final strong classifier, $H(x_i)$. This is done minimizing, at each round, the weighted squared error, $J = \sum_{i=1}^{N} w_i(y_i - h_m(x_i))^2$, where $w_i = e^{-y_i h_m(x_i)}$ are the weights and N the number of training samples. At each round, the optimal weak classifier is then added to the strong classifier and the data weights adapted, increasing the weight of the misclassified samples and decreasing correctly classified ones [7].

In the case of GentleBoost it is common to use simple functions such as regression stumps. They have the form $h_m(x_i) = a\delta\left[x_i^f > \theta\right] + b\delta\left[x_i^f \leq \theta\right]$, where f is the number of the feature and δ is an indicator function (i.e. δ[*condition*] is one if *condition* is *true* and zero otherwise). Regression stumps can be viewed as decision trees with only one node, where the indicator function sharply chooses branch a or b depending on threshold θ and feature x_i^f. To optimize the stump one must find the set of parameters $\{a, b, f, \theta\}$ that minimizes J w.r.t. h_m. A closed form for the optimal a and b are obtained and the value of pair $\{f, \theta\}$ is found using an exhaustive search [7]. Next section shows how to include temporal consistency in the regression stumps.

2.1 Weak learners with temporal consistency

In this work we add temporal consistency to the weak classifier response, h_m, using a set of T consecutive data points to perform the classification. The rationale is to use as much as possible of the information available in order to improve classifier output.

We propose to include the temporal window size T as an additional parameter of the regression stump. In this way the regression stump will only use advantageous information at each round, by choosing how many points to use depending on the feature values, opposed to the common approach of constant window length. Thus, consistency is included by defining the new Temporal Stumps as the mean classification output of the regression stump, in a temporal window of size T,

$$h_m^*(x_i) = \frac{1}{T}\sum_{t=0}^{T-1}\left(a\delta\left[x_{i-t}^f > \theta\right] + b\delta\left[x_{i-t}^f \leq \theta\right]\right). \tag{1}$$

The particular information extracted within the temporal window become clear if we put a and b in evidence,

$$h_m^*(x_i) = a\left(\frac{1}{T}\sum_{t=0}^{T-1}\delta\left[x_{i-t}^f > \theta\right]\right) + b\left(\frac{1}{T}\sum_{t=0}^{T-1}\delta\left[x_{i-t}^f \leq \theta\right]\right). \tag{2}$$

The new temporal weak classifier of Eq. 2 can be viewed as the classic regression stump with a different "indicator function". If $T = 1$ it becomes the original regression stump, and for $T > 1$ the indicator function changes. The new indicator functions

$$\Delta_+^T(f, \theta, T) = \frac{1}{T}\sum_{t}^{T-1}\delta\left[x_{i-t}^f > \theta\right], \;\; \Delta_-^T(f, \theta, T) = \frac{1}{T}\sum_{t}^{T-1}\delta\left[x_{i-t}^f \leq \theta\right], \tag{3}$$

compute the percentage of points above and below the threshold θ, in the temporal window T and for the feature number f. The indicator functions with temporal consistency in Eq. 3, can take any value in the interval $[0\ 1]$, depending on the length of the temporal window used. For example, if $T = 2$ the functions can take 3 different values, $\Delta_+^T \in \{0,\ 1/2,\ 1\}$, if $T = 3$ can take four values, $\Delta_+^T \in \{0,\ 1/3,\ 2/3,\ 1\}$ and so on.

The fuzzy output of the new "indicator function", Δ, represents the confidence of threshold choice to use the data with temporal support T. Thus, at each boosting round, we use a weighted confidence of both branches, instead of choosing only one branch. We present experimental results that show that optimizing this fuzzy regression stump brings additional resistance to noise, thus increasing the generalization capabilities.

During classification of unseen data, the algorithm has the possibility to decrease the confidence measure, Δ, for instance if the new data is noisy when compared to the training data. This differs from the usual boosting binary decision and can be compared to what fuzzy trees brought to decision trees.

Replacing the weak classifier with temporal consistency of Eq. 2 in the cost function, we compute the optimal temporal stump parameters a and b,

$$a = \frac{\bar{\nu}_+\bar{\omega}_- - \bar{\nu}_-\bar{\omega}_\pm}{\bar{\omega}_+\bar{\omega}_- - (\bar{\omega}_\pm)^2}, \quad b = \frac{\bar{\nu}_-\bar{\omega}_+ - \bar{\nu}_+\bar{\omega}_\pm}{\bar{\omega}_+\bar{\omega}_- - (\bar{\omega}_\pm)^2}, \tag{4}$$

with $\bar{\nu}_+ = \sum_i^N w_i y_i \Delta_+^T$, $\bar{\nu}_- = \sum_i^N w_i y_i \Delta_-^T$,
$\bar{\omega}_+ = \sum_i^N w_i \Delta_+^T$, $\bar{\omega}_- = \sum_i^N w_i \Delta_-^T$, $\bar{\omega}_\pm = \sum_i^N w_i \Delta_-^T \Delta_+^T$.

Note that all the above variables are functions of $\{f, \theta, T\}$ that we dropped for notation simplicity. To find the optimal f,θ and T we use exhaustive search.

Comparing the temporal weak learner with the original GentleBoost weak learner, we have an additional parameter T to optimize. The algorithm is similar to Gentleboost, now optimizing the presented temporal stump, h_m^*.

It is important to remark that the proposed framework is, as the classic boosting approaches, a non-sequential single-frame classifier. It should be used to classify data at one time instant with the internal difference that it "looks" back a few points in time, adding consistency to the decision. The data used does not need to have any special characteristic despite the fact of having some temporal sequence.

2.2 Results on Synthetic data

We perform tests with synthetic data in order to illustrate the advantages of our algorithm over other boosting approaches: (i) improved noise robustness, and (ii) improved performance in data with large overlapping in the feature space. We create synthetic data and apply three boosting algorithms: (i) GentleBoost [4], (ii) TemporalBoost [5] and (iii) our optimal temporal boosting.

The aim is to learn two elliptic trajectories using point location as features, and then classify new points as belonging to one of the trajectories. The input features (x_i^1, x_i^2) are noisy observations of the actual ellipses, generated accordingly to: $x_i^1 = a\cos t + \mathcal{N}(0,\sigma)$ and $x_i^2 = b\sin t + \mathcal{N}(0,\sigma)$, where a and b are the major and minor axis, t represents time, and $\mathcal{N}(\mu,\sigma)$ is Gaussian noise. In Figure 1(a) we observe examples of trajectories with $\sigma = 0$, and $\sigma = 0.15$.

Figure 1(b) plots the evolution of the recognition rate along rounds for the test set. The experiment corresponds to trajectories corrupted by noise with $\sigma = 0.15$. Our boosting proposal clearly outperforms TemporalBoost and GentleBoost.

Noise robustness: We perform 15 experiments for each boosting algorithm, changing the noise variance linearly from 0 to 0.3. For each experiment, we compute recognition rate along 100 rounds and then pick the maximum value. In Figure 1(d) we plot the recognition performance maxima vs noise variance, showing experimentally that our boosting algorithm is more robust to noise than GentleBoost and TemporalBoost. The reason for this behavior is the fuzzy output of the classifiers, that takes into account uncertainty in decisions at each boosting round.

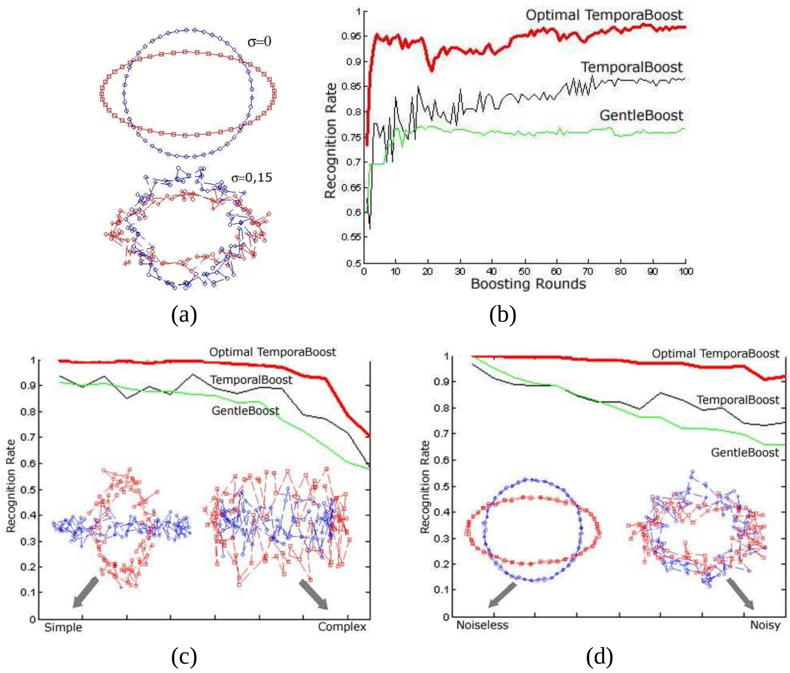


Fig. 1. Comparison of the recognition rate evolution for: GentleBoost, TemporalBoost and our OptimalTemporalBoost. The two class elliptical trajectories in (a) are used to compute the evolution along rounds, in (b), for the three algorithms. Picking the maximum of the evolution along rounds for each experiment we vary: the problem complexity presented in (c), and the varying the features noise in (d).

Class overlap: In this experiment we increase gradually the amount of overlap among classes, keeping noise constant. Like in the previous experiment, we pick the best recognition rate in all rounds. In Figure 1(c) we see that our algorithm surpasses GentleBoost and TemporalBoost. The explicit inclusion of temporal parameters in the regression stump, and consequently joint optimization with the remaining ones can explain the improvement of our algorithm over TemporalBoost.

2.3 Comments

We show experimentally that our temporal GentleBoost algorithm increases noise robustness up to 20%, and is able to handle class overlapping in average with 10% better results than TemporalBoost. These results clearly indicate the advantage of using this new framework instead of other boosting approaches when working with temporal data. Additionally, the explicit time formulation in the regression stumps allow us to extend, in a straightforward manner, the temporal GentleBoost to multi-class problems.

3 Going to the multi class problem

A Multi-class categorization problem is usually solved as a set of multi-binary problems where separate classifiers are trained and applied independently. As pointed by Torralba *et al* [7], for the object detection problem, this is a waste of resources because many of the features used can be shared among several classes. In the case of boosting, sharing helps to: i) reduce computational complexity by sharing weak classifiers among classes and ii) reduce the amount of training data needed in order to attain the same classification performance. We generalize the temporal GentleBoost to the multi-class problem likewise Torralba *et al* extended GentleBoost to multi-class problems.

3.1 The Temporal-JointBoosting algorithm

The idea behind JointBoosting [7] is to share weak classifiers (and features) across classes. At each round the algorithm chooses a weak classifier that shares a feature among the subset of classes. The optimal subset of classes is chosen by minimizing the error cost function for all possible combinations.

The optimization to be solved has now one more variable, the classes, thus one must solve, $J = \sum_{c=1}^{C} \sum_{i=1}^{N} w_i^c (y_i^c - h_m(c, x_i))^2$, the new weighted least squares problem in each iteration, where $w_i^c = e^{-y_i^c h_m(c,x_i)}$ are the new class-specific weights. Shared stumps use the data that belongs to the optimal subset of classes as positive samples, and the remaining data as negative samples. For classes in the optimal subset $S(n)$, the stump function is similar to the binary case. For classes outside the optimal subset, the stump function is a class-specific constant, k^c. (see [7] for details). The shared temporal stump has the following form:

$$h_m(c, x) = \begin{cases} a_S \Delta_+^T + b_S \Delta_-^T & \text{if } c \in S(n) \\ k_S^c & \text{if } c \notin S(n), \end{cases} \tag{5}$$

where $\Delta_{+/-}^T$ are the temporal consistency function defined in Eq. 3. The optimal parameters of the shared stump are:

$$a_S = \frac{\sum_{c\in S(n)} \bar{\nu}_+^c \sum_{c\in S(n)} \bar{\omega}_-^c - \sum_{c\in S(n)} \bar{\nu}_-^c \sum_{c\in S(n)} \bar{\omega}_\pm^c}{\sum_{c\in S(n)} \bar{\omega}_+^c \sum_{c\in S(n)} \bar{\omega}_-^c - \left(\sum_{c\in S(n)} \bar{\omega}_\pm^c\right)^2}, \tag{6}$$

$$b_S = \frac{\sum_{c\in S(n)} \bar{\nu}_-^c \sum_{c\in S(n)} \bar{\omega}_+^c - \sum_{c\in S(n)} \bar{\nu}_+^c \sum_{c\in S(n)} \bar{\omega}_\pm^c}{\sum_{c\in S(n)} \bar{\omega}_+^c \sum_{c\in S(n)} \bar{\omega}_-^c - \left(\sum_{c\in S(n)} \bar{\omega}_\pm^c\right)^2}, \tag{7}$$

$$k^c = \frac{\sum_i w_i^c y_i^c}{\sum_i w_i^c}, \;\; c \notin S(n), \tag{8}$$

and obtain θ, f, T and $S(n)$, exhaustive search is performed [7].

3.2 Results on Synthetic data

We apply multiclass versions of the previously used three boosting algorithms: (i) One against all version of the TemporalBoost, (ii) JointBoost, and (iii) JointBoost version of optimal temporal boost. In this case we aim to model several classes, and perform similar tests to the class overlap tests in the binary problem.

Five elliptical trajectories were generated for 10 levels of overlapping between classes, each one corresponding to 5 class classification problem. We vary the problem complexity, starting from the simplest case (easily separable), and increasing the proximity among classes toward a more complex problem. In Figure 2 we see that

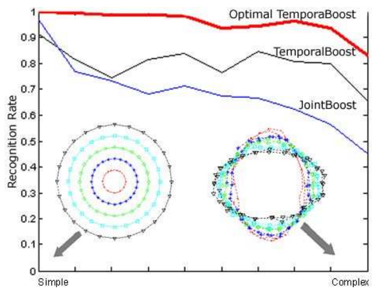


Fig. 2. Recognition rate in multi-class synthetic datasets, using 10 levels of increasing overlap among classes.

multi-class boosting algorithms have a very similar behavior to the two-class boosting. In this case the classification rate improvement is larger than the binary counterpart, with our temporal version of JointBoost performing almost 20% better than TemporalBoost, and 30% better than the original JointBoost. Our JointBoost version of optimal temporal boost further improves the advantages over current state of the art methods when working with the intuitively more complex multiclass problem. The following step is to test our temporal JointBoost in real and very challenging datasets.

4 Human activity recognition

For the real datasets tests we consider the problem of human activity recognition using 2 different scenarios. Firstly we present results on the CAVIAR [8] scenario, a very challenging dataset due to: i) perspective distortion, ii) radial distortion and iii) the presence of a vanishing point in the image that makes human images varying from top view to side view. Using the CAVIAR dataset we aim to classify five general and basic human activities, {Active, Inactive, Walking, Running, Fighting}. The Active class considers movement of body parts that do not originate translation in the image.

The second dataset contains body specific movements that can be viewed as a detailed interpretation of the Active class. The movements considered are related to 2 body parts: i) the trunk and ii) the arms. For the trunk we recognize the movements of bending down/stand up and turning right/left. The arms movements comprise rising/putting down both right and left arms. We consider a total of 8 movements.

4.1 General activities recognition

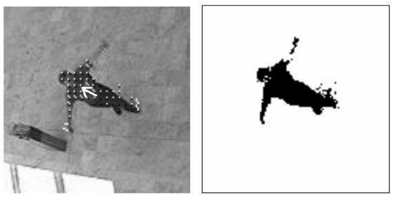
The problem is to recognize five human activities from video sequences, {Active, Inactive, Walking, Running, Fighting}. A total of about 16,000 images with ground truth were used and are distributed according to table 3(a). Figure 3(b) shows tree examples of each considered activity. Figure 3(c) shows an image from the fighting scenario

id	# Frames	Activity
1	3,211	Inactive
2	1,974	Active
3	9,831	Walking
4	297	Running
5	594	Fighting

(a)

(b)

(c)



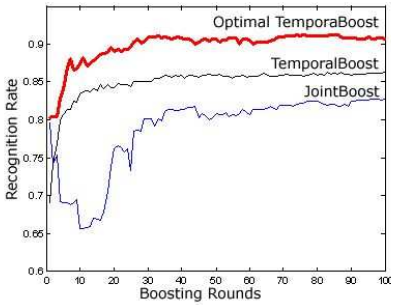
(d)

Fig. 3. Considered activities and data distribution for the CAVIAR dataset (a) and example images for each class (b). The test scenario is exemplified in (c) and (d) presents the two types of information used to compute all the 29 features: the target velocity and the optical flow vectors (the optical flow represented in the image was sampled from the all the computed vectors).

used to extract the examples from figure 3(b). Note the wide variability of this dataset, for example the third column (in figure 3(b)) correspond to the walking activity and in approximately one second the person changes from top to side view.

The features used to perform classification were obtained from the detected moving blobs in the scene that correspond to people. Once the information regarding the position of the target over time is provided, we compute 29 features based on 2 characteristics: i) the instantaneous position and velocity of the tracked subject and ii) the *optic flow* or instantaneous pixel motion inside the target's bounding box. An example of subject's velocity and optic flow is plotted in Figure 3(d). The rationale behind the 29 features is to model some important characteristics of people movements: i) speed, ii) regularity of the trajectory, iii) motion energy and iv) regularity of motion. We also use both instantaneous, averaged and second order moments of these quantities. A detailed description of the features can be found in [9].

We perform a leave one out subset process to compute the recognition rate, dividing the dataset into four different subsets (each one with similar number of frames). The definitive recognition rate is the average of the four tests. We compare three algorithms: (i) One against all TemporalBoost, (ii) JointBoost and (iii) optimal temporal JointBoost. We show the average recognition rate for each boosting round in Figure 4(a) and the best recognition rate and correspondent standard deviation in table 4(b).



(a)

Algorithm	Mean	Std
OptBoost	91%	±3.2%
TempBoost	86%	±2.8%
JointBoost	83%	±3.3%

(b)

Fig. 4. Recognition rate in multi-class synthetic datasets, using 10 levels of increasing overlap among classes (a). Algorithms recognition rate comparison for the CAVIAR scenario(b) with best recognition rate and standard deviation (c).

In this test, optimal temporal JointBoost outperforms one against all TemporalBoost by 5%, and JointBoost by 8%.

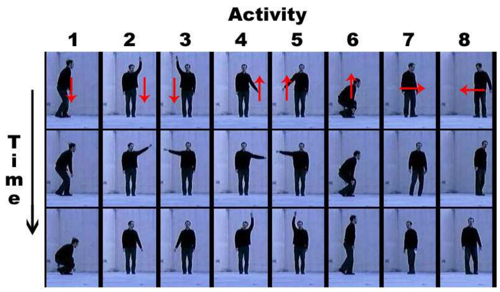
4.2 Body parts movements recognition

This group of tests aim to recognize body movements that do not originate translation of the target in the image. In Table 5(a) we see the data distribution for every type of body movement, and in Figure 5(b) examples of them are plotted.

The activities considered here are based on the movement of 2 body parts, the trunk and the arms, and for each movement several sequences were recorded. Each movement was performed in 2 different locations and/or performed in 2 different ways. For example, turning is always performed in the same manner but in 2 different locations, one near the camera and one far from it. The opposite happens with the arms movement, they are performed in the same location but in two different sights. The bending movements are performed in both locations and sights (e.g. side and front sight). See figure 6(b) for example images illustrating this differences and figure 6(a) for the working scenario. In each case 4 different sequences where recorded, and the dataset contains a total of 3147 frames.

id	# Frames	Movement
1	429	bend trunk down
2	243	right arm down
3	248	left arm down
4	224	right arm up
5	248	left arm up
6	459	bend trunk up
7	610	rotate trunk right
8	686	rotate trunk left

(a)



(b)

Fig. 5. Considered body part movements and data distribution in (a) and example images for the eight movements in (b).

As features we consider optic flow referred to the person centroid in order to obtain only the relative motion of the different parts of the body. The idea is to have a qualitative measure of movements, without segmenting or identifying parts of the body, that would be a difficult and sometimes an impossible problem due to recurrent occlusions.

We assume that the body parts are arranged around the body centroid, thus the optical flow vectors are averaged along angular directions with the origin at the centroid. A detailed explanation of the feature extraction method can be found in [10]. The resultant features are vectors along a discrete number of angular direction that aim to represent the radial and normal movement of the person with respect to the correspondent angular direction. In figure 6(c) are plotted the features used, with 10 angular direction, for one of the rising left arm example.

To compare the performance of all the boosting algorithms we present in figure 6(d) the evolution of the recognition rate where the classifiers are trained with half of the sequences and tested with the remaining ones (performed twice by exchanging the test and training sets). In this experiment examples from the 2 locations and sights are present in the training set.

In figure 6(e) we evaluate the ability of our algorithm to recognize activities performed in different sights and image locations (scales). For this experiment we train the algorithms with sequences recorded at one location and one sight and in the test set we use sequences recorded at a different location and/or sight. Combining the two locations and sights we perform a total of four tests.

The results clearly show the advantage of using our algorithm, performing more than 11% better than TemporalBoost and 20% than JointBoost. In the more complicated test (figure 6(e)) the differences between the methods are even greather (15% in relation to TemporalBoost), with a 3% decreasing in the recognition rate of our method, when compared with the previous and much simpler test. These results indicate that we are able to recognise types of activities, even when performed differently, rather than exact types of movements.

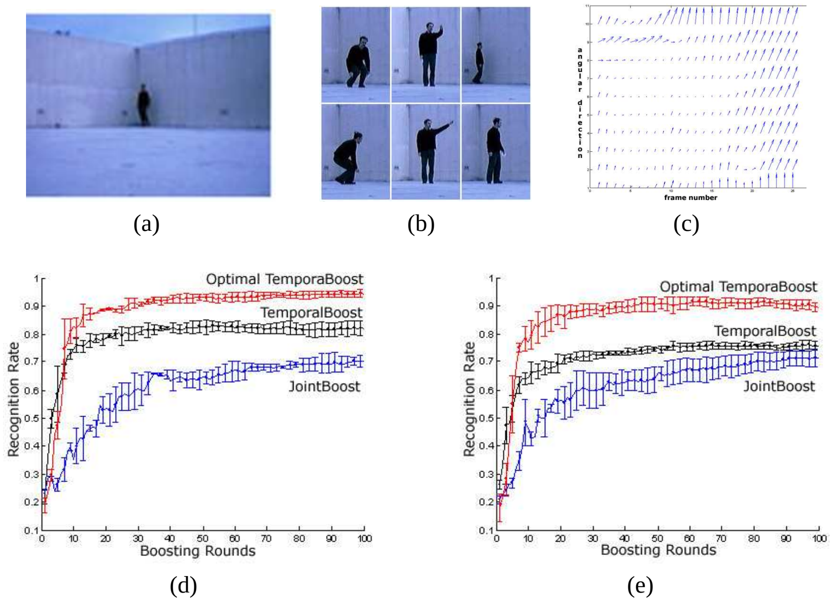


Fig. 6. Examples of: the global scenario (a), bending, rotating and rising right arm movements, illustrating the location and sigh differences (b) and the features computed for one rising left arm example (c). The algorithms recognition rate are compared when: the classifiers are trained with half of the sequences and tested with the remaining ones (d) and when train and test are done in different locations and sights (e).

4.3 Discussion

The single-frame overall recognition rate of 91%, in the general human activity, and 94%, in the specific body parts movements recognition tests are very good result, taking into acount the wide type of distortions present in the first scenario, that makes it one of the most challenging scenarios for this kind of task, and that we do not used any dynamic model of activity transitions. The inclusion of transitional dynamic models depends on the desired application, but the results presented here can be straightforward used as a lower level for several applications, e.g. urban surveillance, intelligent environments or human/robot interfaces and interaction. Furthermore, the inclusion of such higher level models (restrictions) should increase the recognition rate making the system very robust on real world aplpications.

5 Conclusions

We have proposed a methodology to handle temporal data consistency in non-sequential single-frame classification problems. Although the temporal evolution of the data might be crucial for certain classification problems (e.g. human activity), even when performing single-frame classification, it is rarely addressed at the level of the classifier.

We have adopted the boosting framework and propose a method whereby time is taken into account in the boosting optimization steps. In our work, temporal consistency is treated as part of the overall optimization procedure, contrasting with the heuristic approach adopted in the recently proposed Temporal AdaBoost.

More specifically, we extend the GentleBoost algorithm by modeling time as an explicit parameter to optimize. As a consequence of this time consistency model, we obtain a fuzzy decision function, depending on the temporal support of the data, which brings additional robustness to noise. Finally, we allow our temporal boosting algorithm to cope with multi class problems, following the JointBoosting approach.

We have conducted extensive tests to demonstrate the superiority of our approach when compared to the Temporal Boost and the GentleBoost algorithms. We use synthetic datasets with increasing complexity as well as real video data recognizing human activities. The results show that our method clearly outperform the Temporal Boosting algorithm by $5-10\%$ and standard GentleBoost algorithm by as much as $10-20\%$.

Using the real datasets we achieve performance always greater than 90% for a very challenging scenario, due to large image distortions. In the classification of more specific body parts movements the recognition rate is superior to 94%. This are very good results taking into acount that we perform single-frame classification without modeling the activities dynamics.

We present results that clearly show the importance of temporal data consistency in single-frame classification problems as well as the importance of as handling time in an optimal, fully consistent manner. Additionally, this new framework can be used with any type of sequential data, thus being applicable to a wide range of other problems, rather than the ones discussed here.

References

1. Viola, P., Jones, M., Snow, D.: Detecting pedestrians using patterns of motion and appearance. International Journal of Computer Vision (2) (2005) 153–161
2. Dietterich, T.G., Ashenfelter, A., Bulatov, Y.: Training conditional random fields via gradient tree boosting. In: Proceedings of the ICML '04, NY, USA (2004) 28
3. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: Proceedings of the ICCV'03, IEEE Computer Society (2003) 726
4. J. Friedman, T.H., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Annals of Statistics **28**(2) (2000) 337407
5. Smith, P., da Vitoria Lobo, N., Shah, M.: Temporalboost for event recognition. In: Proceedings ICCV'05. Volume 1. (2005) 733– 740
6. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: International Conference on Machine Learning. (1996) 148–156
7. Torralba, A., Murphy, K., Freeman, W.: Sharing visual features for multiclass and multiview object detection. IEEE Transactions on PAMI **29**(5) (2007) 854–869
8. (http://homepages.inf.ed.ac.uk/rbf/CAVIAR/)
9. Ribeiro, P., Santos-Victor, J.: Human activities recognition from video: modeling, feature selection and classification architecture. In: BMVC Workshop on HAREM. (2005)
10. Pla, F., Ribeiro, P.C., Santos-Victor, J., Bernardino, A.: Extracting motion features for visual human activity representation. In: Proceedings of the IbPRIA'05. (2005)