# A Trainable Similarity Measure for Image Classification

Pavel Paclík[1], Jana Novovičová[2], Robert P.W.Duin[1]

[1] Elect. Eng., Maths and Comp. Sc.
Delft University of Technology
Delft, The Netherlands
{p.paclik,r.p.w.duin}@ewi.tudelft.n

[2] Inst.of Information Theory and Automation
Academy of Science of Czech Republic
Prague, Czech Republic
novovic@utia.cas.cz

## Abstract

*In object recognition problems a two-stage system is usually adopted composed of a fast and simple detector and a more complex classifier. This paper studies a design of the second stage classifier based on the recently proposed trainable similarity measure which is specifically designed for supervised classification of images. Common global measures such as correlation suffer from uninformative pixels and occlusions. The proposed measure is based on local matches in a set of regions within an image which increases its robustness. The configuration of local regions is derived specifically for each prototype by a training procedure. The paper compares the classifiers built using the trainable similarity to the state-of-the-art AdaBoost classifiers on a real-world pedestrian recognition problem. The paper illustrates that for a given range of sample sizes the trainable similarity represents a better solution for second-stage classification than the AdaBoost algorithm which requires significantly larger training sets.*

## 1. Introduction

Recognition of objects in images is typically carried on in two stages. First, the objects in an image are localized by a detector [8, 7]. In the second stage, the identified candidate regions are processed by a more elaborate classifier performing the possible multi-class discrimination and rejecting the false alarms inevitably introduced by the detector [5]. This setup allows for a simpler/faster detector and more complex classifier. This study focuses on the development of the second-stage classifier for object recognition.

There are two basic methods for construction of a data representation, namely the feature-based and (dis)similarity-based approaches. While the feature-based approaches derive a a set of characteristics (features) directly from an input image patch, the similarity-based methods compare an input patch to the stored prototype patches.

A common similarity-based approach to image classification is based on the normalized cross-correlation. Although compelling due to its statistical interpretation, it suffers from presence of uninformative pixels and occlusions and is computationally expensive. In [5], we have proposed to compute the similarity to a prototype based on multiple matches in local sub-regions, rather that using the entire candidate region. This approach improves robustness of the similarity measure. Because our objective is the construction of a similarity measure facilitating supervised classification, we proposed to derive the appropriate set of local sub-regions optimizing the informativeness of the resulting similarity in terms of class separation. The proposed similarity measure is therefore *trained* specifically for each prototype so that the class of the prototype is separated as good as possible from all other classes in the labeled training set. Hence, we call this measure the *trainable similarity*.

The proposed measure combines advantages of the similarity-based and feature-based approaches. Because it derives a similarity-based data representation, it allows for easier rejection of unseen (dissimilar) examples. By using multiple prototypes the handling of multi-modal problems is also simplified. On the other hand, the proposed computation of similarity using local image matches resembles the feature-based approach extracting localized features from an image. However, extracting the local features specifically for each prototype might result in an overtrained system. In this paper, we therefore study the generalization behavior of classifiers based on the trainable similarity over a range of sample sizes on a pedestrian recognition dataset.

For the sake of comparison, we include the state-of-the-art AdaBoost classifier of Viola and Jones [8] which also leverages the local image information. Contrary to the trainable similarity where image content in local regions is matched to the stored prototype example, AdaBoost identifies a set of simple region features computed by summation and subtraction of image intensities. Each feature is turned into a possibly weak classifier by thresholding. Gradually focusing on more problematic examples, the AdaBoost algorithm derives a generalizing classifier ensemble.

0-7695-2521-0/06/$20.00 (c) 2006 IEEE

In the following section, we introduce the proposed trainable similarity measure. Section 3 describes the pedestrian recognition problem and the experimental setup. Section 4 provides discussion on results and short note on computational complexity. Conclusions are given in Section (5).

## 2. Trainable similarity measure

In order to formally derive the trainable similarity, let us first introduce a similarity measure $S_r(I,J)$ based on the correlation coefficient computed between equally-sized images $I$ and $J$:

$$S_r(I,J) = \frac{\sum_i (I^i - \bar{I})(J^i - \bar{J})}{\sqrt{\sum_i (I^i - \bar{I})^2 \sum_i (J^i - \bar{J})^2}}. \qquad (1)$$

The symbols $I^i$ and $J^i$ represent intensities of the $i$-th pixel of the corresponding images and $\bar{I}$ and $\bar{J}$ denote the corresponding means of image intensities.

Note that this measure is a global and symmetric. In order to avoid impact of uninformative pixels, we aim at the similarity based on local information which is specific to the prototype and thereby asymmetric.

The proposed similarity $S(I,J,R)$ is based on local image matches in a set of local image regions $R$. Each local region $r \in R$ is defined in the coordinate system of the image $I$. The local match $s(I,J,r)$ between the corresponding pixel values $\mathbf{r}(I)$ and $\mathbf{r}(J)$ may be measured, for example, by the correlation coefficient (1). The overall similarity $S(I,J,R)$ is a function of a set of local matches. We adopt here the arithmetic mean which increases robustness of the overall similarity measure to local disturbance:

$$S_{\text{mean}}(I,J,R) = \frac{1}{|R|} \sum_{j=1}^{|R|} s(I,J,r_j). \qquad (2)$$

A set of regions $R$ is determined specifically for each prototype during the training process considering a labeled training dataset $Tr = \{(I_1, \omega(I_1)), ..., (I_N, \omega(I_N))\}$ with $N$ images, where $\omega(I)$ denotes a class of the image $I$.

The process of training the similarity measure with respect to a prototype image $Pr$ proceeds as follows. Starting from an initial set of admissible regions $R_{\text{init}}$ a subset $R$ is selected optimizing the separability of the class of the prototype object $\omega(Pr)$ from all the remaining classes in the training set $Tr$. Formally, the Fisher separability criterion is adopted

$$\mathcal{C}(Tr, Pr, R) = \frac{(\hat{\mu}_T - \hat{\mu}_{NT})^2}{\hat{\sigma}_T^2 + \hat{\sigma}_{NT}^2}, \qquad (3)$$

expressing the separation between target and non-target training objects based on the similarity $S(I, Pr, R)$. The symbols $T$ and $NT$ denote the sets of target and non-target training objects, respectively:

$$\begin{aligned} T &= \{I_i \in Tr : \omega(I_i) = \omega(Pr)\} \\ NT &= \{I_j \in Tr : \omega(I_j) \neq \omega(Pr)\}. \end{aligned} \qquad (4)$$

The $\hat{\mu}_T$ ($\hat{\mu}_{NT}$) and $\hat{\sigma}_T^2$ ($\hat{\sigma}_{NT}^2$) denote mean and variance of the similarity values $S(I_i, Pr, R)$, $I_i \in T$ ($S(I_j, Pr, R)$, $I_j \in NT$).

A set of regions, maximizing the criterion (3) is derived using a search over a large set of randomly positioned regions $R_{\text{init}}$. Similarly to [5], we adopt here the sequential forward search. Alternatives, such as individual and backward search algorithms are discussed in [4]. Note that the number of local regions is not a user-specified parameter but is optimized automatically due to the multi-variate nature of the criterion (3).

In order to build a classifier based on the trainable similarity a set of prototype objects must be first selected. This may be achieved, for example, using a clustering procedure. For each prototype a separate similarity measure is derived by the training procedure. There exist different classifier-building strategies for similarity data representations. We adopt here *the similarity-space approach* [1, 6] where similarities to prototypes are considered dimensions of a new metric space in which all training objects may be represented. A general-purpose classifier such as Fisher linear discriminant (FLD), trained in the similarity space, therefore leverages the correlations between similarities to prototypes, unlike the frequently-used nearest-neighbor rule.

## 3. Experiments

In this section, we describe a set of experiments with a dataset originating from a pedestrian recognition problem. The dataset comprises of 7302 candidate regions identified in video sequences by a detector as pedestrians. Typical video sequence consists of 100 examples. The regions are scaled into fixed size and hand-labeled into two classes as true pedestrians (3502 examples) and non-pedestrians (3800 examples), respectively. Our goal is to design a second-stage classifier processing an incoming candidate region (image patch) and identifying it as a true pedestrian or rejecting it as a falsely-detected non-pedestrian.

Multiple images, originating from a single video sequence, often bear only minor differences (see Figure 1). In order to avoid that such almost identical images appear both in training and testing, we adopt the following evaluation scheme. The available dataset is split into two disjoint sets of image sequences used for algorithm design (4500 images) and evaluation (2802 images), respectively. The learning curves for the studied algorithms are constructed simulating the real-world scenario where images from a

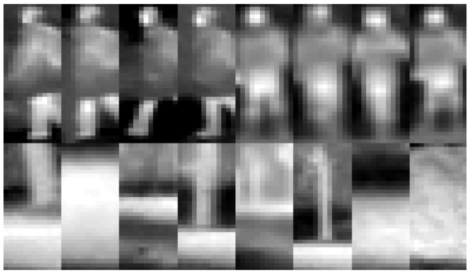

**Figure 1. Examples of candidate regions used in experiments. The pedestrian images in the upper row originate from two sequences.**

growing number of image sequences are utilized (from 2 to 25 sequences). The algorithm, trained on the desired number of sequences, is evaluated using the independent test set of image sequences. In order to assess the significance of estimated performances, the training is repeated 20 times varying the training sequences. The test set remains fixed.

The following similarity-based and feature-based algorithms are considered in this study:

**TrSim1** The trainable similarity using one prototype per class, selected by the $k$-center clustering algorithm [3]. Similarity measure $S_{\mathrm{mean}}$ uses local regions 6×6 pixels, selected by a sequential forward search from the initial pool of 300 regions randomly positioned over the candidate region. Similarities of all training objects to the two prototypes are computed. FLD is built in the resulting similarity space.

**TrSim10** The trainable similarity with 10 prototypes per class selected by the $k$-center algorithm. FLD classifier is built in the similarity space.

**CrossCorr10** The correlation coefficient $S_r(I, J)$ to 10 prototypes, selected using $k$-centers; FLD is trained in the similarity space.

**AdaBoost** The feature-based classifier of Viola and Jones using four types of region features based on an integral image representation [8]. The total considered set comprises 800 randomly sized and positioned region features. Analogously to [8], the base classifiers are constructed by identifying the optimal thresholds and classifier polarities over all training examples. The AdaBoost ensemble is trained until the weighted error used for the base classifier selection reaches zero.

**PCA** Principal Component Analysis (PCA) preserving 99% of overall variance is applied directly to the pixel intensities. The supervised PCA is used based on the pooled class covariance matrices [2]. FLD is trained in the resulting linear subspace.

Figure 2 depicts classifier performances varying the training sample sizes. We adopt the area under ROC measure (AUC) which summarizes the classifier performance over all possible operating points.

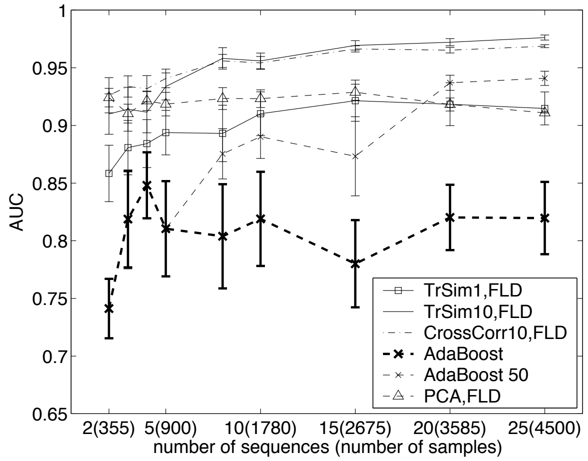

**Figure 2. Performance of classifiers (area under ROC curve) for the increasing number of training image sequences/examples. Each point is a result of 20-fold cross-validation.**

## 4. Discussion

It is interesting to compare the learning curves of the algorithm TrSim10 and CrossCorr10. While the CrossCorr10 is better for very small sample sizes, this difference gradually narrows. Eventually CrossCorr10 is significantly outperformed by TrSim10 for large sample sizes. This behavior is to be expected as the TrSim10 is the more complex classifier than CrossCorr10 due to training of individual similarity measures. More elaborate derivation of the data representation gradually becomes an advantage given sufficient training data. Figure 3 shows local regions trained with respect to several prototypes. The number of extracted local regions ranges between 5 and 11 over all training set sizes.

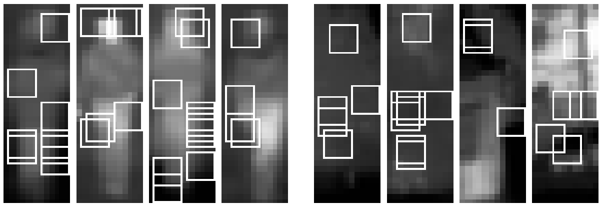

**Figure 3. Local regions derived by training of the similarity measure $S_{\mathrm{mean}}$ to eight prototypes (four pedestrians on the left and four non-pedestrians on the right)**

The PCA/FLD classifier provides better solution than the TrSim10 for small sample sizes but its does not improve with a growing number of samples. Interestingly, the AdaBoost algorithm yields the worst AUC performances with very large variances. It is outperformed even by the trainable similarity TrSim1 with a single prototype per class. Trying to understand the reasons for AdaBoost poor performance, we investigated the eventual ensemble sizes. It appears that for the smallest number of samples (two video sequences, 355 images) on average only 4.5 classifiers are used. For five sequences it is only 16 classifiers and even for the largest sample size of 25 sequences (4500 samples) only 130 classifiers are derived on average.

This suggests that the simple thresholded intensity summations and subtractions, used as weak learners, quickly find zero error solution on the training set. This effect is probably still emphasized by the low effective sample size of the datasets originating from image sequences (pedestrians occur in clusters with minor variations). In order to counter the effect of overtraining, we limit the number of classifiers to 50. This significantly improves performance – see the curve *AdaBoost 50* in Figure 2.

### 4.1. Computational complexity

The trainable similarity exhibits the highest training complexity amongst the studied algorithms. Its training time is in order of minutes on a common PC (Matlab implementation). In order to assess the execution complexity, we estimate the number of operations spent by an ideal algorithm implementation based on the parameters obtained in the experiments. For each algorithm, we assume that all quantities that may be precomputed during the training stage are precomputed. Estimated execution speeds for algorithms trained in three points on the learning curve are given in Table 1.

| *method* | *2 seq* | *10 seq* | *25 seq* |
|---|---|---|---|
| TrSim1 | 3 879 | 2 818 | 2 686 |
| TrSim10 | 37 020 | 35 163 | 33 904 |
| CrossCorr10 | 34 702 | 34 702 | 34 702 |
| AdaBoost | 467 | 952 | 1 788 |
| PCA/FLD | 46 483 | 73 044 | 81 594 |

**Table 1. Average number of operations required for processing of a single candidate region for three points on the learning curve.**

We observe that AdaBoost provides the fastest solutions. Because the CrossCorr10 algorithm uses a fixed number of prototypes its computational complexity in execution remains constant. The speed of both the TrSim1 and TrSim10 algorithms increases with the growing sample size as the number of automatically extracted regions decreases. PCA/FLD classifier exhibits the opposite effect where the larger training sets lead to increase of the problem complexity consequently to higher subspace dimensionality.

## 5. Conclusions

In this paper we to tried to understand possible benefits and shortcomings of employing the trainable similarity measure in building second-stage image classifiers. We focused on two points, namely its sensitivity to overtraining and the comparison with the AdaBoost classifier.

We have found out that the effect of overtraining of trainable similarity is apparent only for very small sample sizes. With a growing number of samples, the complexity of trained similarity measure starts to be paying off in terms of performance improvement. An interesting outcome of our study is realization that the AdaBoost algorithm overtrains very easily on the investigated dataset. A possible remedy might be employing even weaker type of learners. With a given set of available training examples, several alternative methods outperform the AdaBoost algorithm.

**Acknowledgments** Authors would like to thank Autoliv Electronics Europe for support and Carmen Lai and David Tax for fruitful discussions.

## References

[1] R. P. W. Duin, D. de Ridder, and D. M. J. Tax. Experiments with object based discriminant functions; a featureless approach to pattern recognition. *Pattern Recognition Letters*, 18(11-13):1159–1166, 1997.

[2] R. P. W. Duin, P. Juszczak, D. de Ridder, P. Paclík, E. Pekalska, and D. M. J. Tax. PR-Tools 4.0, a Matlab toolbox for pattern recognition. Technical report, ICT Group, TU Delft, The Netherlands, January 2004. http://www.prtools.org.

[3] D. Hochbaum and S. D. A best possible heuristic for the k-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985.

[4] P. Paclík. *Building Road Sign Classifiers*. PhD thesis, Czech Technical University Prague, 2004. http://prsysdesign.net/pavel/.

[5] P. Paclík, J. Novovičová, and R. P. W. Duin. Building road sign classifiers using a trainable similarity measure. *IEEE Trans. on Int. Transp. Systems, to appear*, 2006.

[6] E. Pekalska, P. Paclík, and R. P. W. Duin. A generalized kernel approach to dissimilarity based classification. *Journal of Machine Learning Research*, 1(2):175–211, 2001. Special Issue "New Perspectives on Kernel Based Learning Methods".

[7] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proc. of IEEE Int. Conference on Comp.Vision (ICCV)*, volume 2, pages 734–741, October 2003.

[8] P. Viola and M. J. Jones. Robust real-time face detection. *Int.Journal of Computer Vision*, 57(2):137–154, 2004.