Tangent Distance Kernels for Support Vector Machines

Bernard Haasdonk
Computer Science Department
Albert-Ludwigs-University Freiburg
79110 Freiburg, Germany
haasdonk@informatik.uni-freiburg.de

Daniel Keysers
Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen – University of Technology
52056 Aachen, Germany
keysers@informatik.rwth-aachen.de

Abstract

When dealing with pattern recognition problems one encounters different types of a-priori knowledge. It is important to incorporate such knowledge into the classification method at hand. A very common type of a-priori knowledge is transformation invariance of the input data, e.g. geometric transformations of image-data like shifts, scaling etc. Distance based classification methods can make use of this by a modified distance measure called tangent distance [13, 14]. We introduce a new class of kernels for support vector machines which incorporate tangent distance and therefore are applicable in cases where such transformation invariances are known. We report experimental results which show that the performance of our method is comparable to other state-of-the-art methods, while problems of existing ones are avoided.

1. Introduction

An important factor for the choice of a classification method for a given problem is the available a-priori knowledge. During the last few years support vector machines (SVM) [15] have shown to be widely applicable and successful particular in cases where a-priori knowledge consists of labeled learning data.

If more knowledge is available, it is reasonable to incorporate and model this knowledge within the classification algorithm and to expect either to obtain better classification results or to require less training data. Therefore, much active research is dealing with adapting the general SVM methodology to cases where additional a-priori knowledge is available. This is the case e.g. in optical character recognition (OCR). Here it is known that the data is subject to e.g. affine transformations and this knowledge can be exploited to improve classification accuracy.

We want to focus on the very common case where variability of the data can be modeled by transformations which leave the class membership unchanged.

If these transformations can be modeled by mathematical groups of transformations one can incorporate this knowledge independently of the classifier during the feature-extraction stage by group-integration, normalization etc. [4]. This leads to invariant features, on which any classification algorithm can be applied.

A possibility particularly designed to kernel methods, is to build invariant kernels by integrating systems of differential equations [3].

Such transformations however often cannot be described by global transformation groups or this is not desired. In case of OCR, for instance, small rotations of a letter are accepted, but large rotations change class memberships like Z → N, M → W, 6 → 9 etc.

Several methods are known to incorporate such local transformation knowledge in special classifiers. In the next two sections we review a method for distance-based classifiers called *tangent distance* and some existing methods for SVM. Then we are able to combine SVM with tangent distance, which results in a new class of SVM-kernels in Section 4. This is a new method of dealing with invariances in SVM, which circumvents certain problems of existing approaches. Experimental results in Section 5 confirm the applicability of our approach.

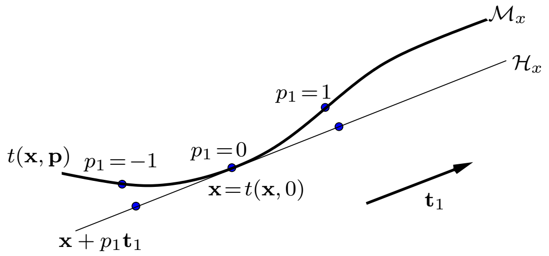
$\mathcal{M}_x$
$\mathcal{H}_x$
$p_1=1$
$t(\mathbf{x},\mathbf{p})$
$p_1=-1$
$p_1=0$
$\mathbf{x}=t(\mathbf{x},0)$
$\mathbf{t}_1$
$\mathbf{x}+p_1\mathbf{t}_1$

Figure 1. Notation for $\mathbf{p} \in I\!R^1$ and $\mathbf{x} \in I\!R^2$.

2. Tangent distance

We will formalize the a-priori-knowledge about local invariances as a differentiable transformation $t(\mathbf{x},\mathbf{p})$, which maps a vector $\mathbf{x} \in I\!R^d$ to $I\!R^d$ depending on some parameter vector $\mathbf{p} = (p_1,\ldots,p_l)^T \in I\!R^l$. We assume that $t(\mathbf{x},\mathbf{0}) = \mathbf{x}$ and that t does not change the class membership of $\mathbf{x}$ for small p_i. This induces a manifold $\mathcal{M}_\mathbf{x} := \left\{t(\mathbf{x},\mathbf{p}) | \mathbf{p} \in I\!R^l\right\} \subset I\!R^d$ of transformed patterns.

For computational reasons we approximate the manifold $\mathcal{M}_\mathbf{x}$ by its tangent hyperplane at point $\mathbf{x}$

$$\mathcal{H}_\mathbf{x} := \left\{ \mathbf{x} + \sum_{i=1}^{l} p_i \mathbf{t}_i \,\middle|\, p_i \in I\!R \right\}.$$

Here $\mathbf{t}_i := \frac{\partial}{\partial p_i} t(\mathbf{x},\mathbf{p})\Big|_{\mathbf{p}=\mathbf{0}}$. denote the tangents that span the plane $\mathcal{H}_\mathbf{x}$, cf. Figure 1.

An approach for dealing with these local invariances in distance-based classifiers is the use of *tangent distance (TD)* introduced in [13, 14]. The idea behind this method is that an adequate dissimilarity measure for two feature vectors $\mathbf{x}$ and $\mathbf{x}'$ is the distance of their manifolds $\mathcal{M}_\mathbf{x}$ and $\mathcal{M}_{\mathbf{x}'}$ or the corresponding linear approximations $\mathcal{H}_\mathbf{x}$ and $\mathcal{H}_{\mathbf{x}'}$, respectively. This is exactly the definition of the so called *two sided TD*

$$d_{2S}(\mathbf{x},\mathbf{x}') := \min_{\mathbf{p},\mathbf{p}'} \left\| \mathbf{x} + \sum_{i=1}^{l} p_i \mathbf{t}_i - \mathbf{x}' - \sum_{i=1}^{l} p'_i \mathbf{t}'_i \right\|.$$

Impressing results on the USPS and NIST handwritten digit datasets have been presented. A computationally cheaper approximation is the *one sided TD*

$$d_{1S}(\mathbf{x},\mathbf{x}') := \min_{\mathbf{p}} \left\| \mathbf{x} + \sum_{i=1}^{l} p_i \mathbf{t}_i - \mathbf{x}' \right\|.$$

The best recognition results on the USPS dataset have been achieved recently by application of one sided TD in a statistical pattern recognition framework [7, 8].

3. Invariance in SVM

Based on learning data $\{(\mathbf{x}_i, y_i)\}_{i=1}^N \subset I\!R^d \times \{-1,+1\}$ of feature vectors $\mathbf{x}_i$ and their labels y_i, a SVM implements a binary decision function $f(\mathbf{x}) := \mathrm{sgn}(g(\mathbf{x}))$ with $g(\mathbf{x}) = \sum_{i=1}^N y_i \alpha_i K(\mathbf{x},\mathbf{x}_i) + b$, where K is a positive definite (p.d.) kernel and $b \in I\!R$ is an offset value. Training of the SVM consists of determining the values α_i such that

$$W(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)$$

is maximized under the constraints $\sum_{i=1}^N \alpha_i y_i = 0$ and $0 \le \alpha_i \le C$ for $i = 1,...,N$. Here C is a regularizing parameter and $\boldsymbol{\alpha}$ denotes the vector with components α_i. The offset value b is calculated based on $\boldsymbol{\alpha}$ and the training set.

Some striking properties of the resulting representation $g(\mathbf{x})$ are: Firstly, most α_i turn out to be zero, such that only $\mathbf{x}_i$ with $\alpha_i \neq 0$ contribute to the sum. These vectors are called support vectors (SV). Secondly, it can be interpreted as a simple linear function in a high-dimensional space induced by the kernel K. For further details and theoretical foundation refer to [15].

In [5] an extensive survey of current techniques for combining the information of transformation invariances with SV-learning is presented. We give some basic ideas here, for details refer to [5] and the references therein.

The *virtual support vector (VSV) method* is based on the idea of generating virtual training data by transformations $t(\mathbf{x}_i,\mathbf{p})$ of training points for small parameters $\mathbf{p}$, and training on this extended set. As the size of this extended set is a multiple of the original one, training is computationally demanding. To circumvent this, the VSV-method is a two step method: first an ordinary SVM-training is performed, then only the set of resulting SVs is extended by small transformations, finally a second SVM-training is performed on this set.

Further simplification of computation is obtained by using the linear approximation $\mathcal{H}_\mathbf{x}$ instead of $\mathcal{M}_\mathbf{x}$. By doing so the necessary transformations of a SV $\mathbf{x}$ reduce to adding multiples of the tangents $\mathbf{t}_i$.

The advantages of the VSV method are the applicability to arbitrary standard kernels and a clear increase of recognition performance. Problems are the two training stages and an increased number of SV after the second stage, which lead to longer training and classification times [12].

Other methods like *invariant hyperplanes* try to modify the kernel function in a very simple way, such that it globally fits all local invariant directions best. This turns out to be equivalent to a prewhitening of the data along these directions which fit best to all local invariance directions simultaneously. Obviously, this method does not respect local invariance in each training point, furthermore it appears to be computationally very hard in the nonlinear case. The advantage is the use of the original SVM-training procedure after prewhitening the training data.

The so called *kernel jittering* method is also based on the idea of small transformations of the training points. Instead of performing these shifts before training, they are performed during kernel-evaluation.

4. Tangent distance kernels

In constructing kernels which incorporate TD we face some problems concerning interpretation of the resulting kernels as scalar products or equivalently their positive definiteness.

A basic property of TD-measures is that they are not metrics as the triangle inequality is invalid, which can be shown easily by counterexamples. This implies that TD-measures cannot be induced norms of any scalar product in any Hilbert space and discards several further possible relations to any scalar product.

A second problem is that some TD variants, e.g. d_{1S}, are not symmetric. This problem, however, is solvable by defining symmetric modifications of TD, which have the same computational efficiency as d_{1S}. We define the square of the *mean TD* to be the mean of the two squared one sided distances:

$$d_{MN}(\mathbf{x},\mathbf{x}') := \sqrt{\frac{1}{2}\left(d^2_{1S}(\mathbf{x},\mathbf{x}') + d^2_{1S}(\mathbf{x}',\mathbf{x})\right)}.$$

Of course a simple mean of the distances would also be a possible modification. During calculation we rather deal with squared distances than with real distances, therefore this definition is more practical.

We further introduce the *midpoint TD* to be the sum of the two one sided distances to the midpoint $\bar{\mathbf{x}} := \frac{1}{2}(\mathbf{x}+\mathbf{x}')$:

$$d_{MP}(\mathbf{x},\mathbf{x}') := d_{1S}(\bar{\mathbf{x}},\mathbf{x}) + d_{1S}(\bar{\mathbf{x}},\mathbf{x}').$$

Other TD-measures are possible, e.g. combinations of TD-measures with the Euclidean distance in order to prevent the situation (in high dimensional spaces very unlikely), that points which are very distant have accidentally small TD-distance, cf. [14].

We now are able to define our TD-kernels. Given an arbitrary distance-based kernel, i.e. $K(\mathbf{x},\mathbf{x}') := k(\|\mathbf{x}-\mathbf{x}'\|)$ we simply replace the Euclidean distance by any TD-measure and obtain the corresponding TD-kernels $K_{1S}(\mathbf{x},\mathbf{x}') := k(d_{1S}(\mathbf{x},\mathbf{x}'))$ and analogously K_{2S}, K_{MN} and K_{MP}.

We denote two particular distance based kernels: the radial basis function kernel $K^{RBF}(\mathbf{x},\mathbf{x}') := e^{-\gamma\|\mathbf{x}-\mathbf{x}'\|^2}$ and the negative distance kernel $K^{ND}(\mathbf{x},\mathbf{x}') := -\|\mathbf{x}-\mathbf{x}'\|^\gamma$. The latter one is not p.d. but for $\gamma \in (0,2]$ still *conditional positive definite (c.p.d.)*, which is completely sufficient for application in SVM, cf. [2, 10].

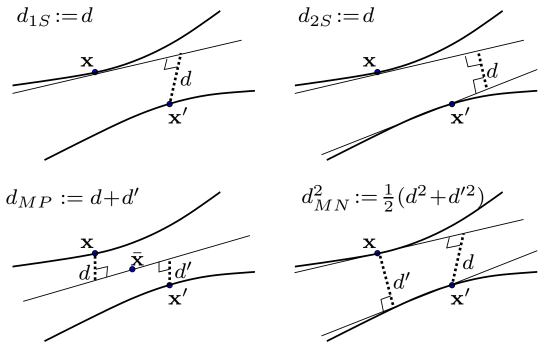

Figure 2. Illustration of TD-measures.

In Figure 2 we illustrate the four TD-variants.

At this point we can already state some properties of these TD-kernels. They share the obvious disadvantages with the kernel-jittering method compared to VSV. First there is the non-symmetry of the kernels which use one sided transformations and the problem of not being c.p.d. This is a serious problem from a theoretical point of view, as the global optimum of the SVM-solution cannot be guaranteed anymore. Nevertheless such kernels often prove to be applicable, e.g. Gaussian dynamic time warping kernel [1], kernel-jittering [5], or sigmoid-kernel [11], which is not p.d. for large ranges of its parameters. Similar to kernel-jittering, our approach is only applicable to distance based kernels. A disadvantage from the computational point of view is the necessity of calculating the tangents and transformations during training and classification which results in a slowdown proportional to l^2, where l denotes the number of tangent directions.

Advantages of our approach are the following: The set of SVs remains small, we only require one training stage, no generation of virtual data or prewhitening of the data is necessary. Furthermore our approach effectively respects local invariances instead of a global integration of these local directions as in the *invariant hyperplane* approach. In contrast to the VSV-method, we also respect these local invariances during classification. In contrast to VSV or kernel-jittering approach, we do not have to decide about the fixed values for the p_i.

5. Experiments

We tested our approach on the US-Postal-Service digit dataset, as there exists a lot of reference results in the literature, in particular results from the VSV-method and TD-approaches. Table 1 lists some of them. The data consists of 7291 training and 2007 test images of 16×16 greyvalue images of handwritten digits. Some results in the literature

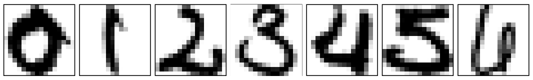

Figure 3. Examples of USPS digits.

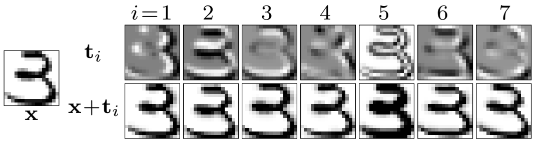

Figure 4. Tangents $\mathbf{t}_i$ and shifts $\mathbf{x}+\mathbf{t}_i \in \mathcal{H}_{\mathbf{x}}$.

use a training set extended by about 2400 machine printed digits. These are marked with *, as their error rates are not directly comparable. Figure 3 shows some example images.

Method	Error rate [%]
Human Performance [13]	2.5
Neural Net (LeNet1) [14]	4.2
SVM, no invariance [11]	4.0
SVM, VSV-method [12]	3.2
k-Nearest Neighbour [14]	*5.7
k-NN + TD [14]	*2.5
TD + kernel densities [7]	2.4
* := extended training set.	

Table 1. Selection of USPS results.

We transformed the original USPS digits to values in $[0, 1]$ and scaled them to norm ≤ 1. We chose the two kernels K^{RBF}, K^{ND} and their corresponding TD-kernels for our experiments. We used the seven tangent directions of Simard [13]: x,y-translation, scaling, rotation, line thickening and two hyperbolic transformations. Figure 4 shows some tangents and points on the hyperplane $\mathcal{H}_{\mathbf{x}}$ of an example $\mathbf{x}$ by shifting $\mathbf{x}$ along these seven tangent directions.

The multiclass-problem was solved by the *decision directed acyclic graph (DDAG)* combination of pairwise SVM [9]. We applied no nodewise model-selection, but simply used fixed kernel-parameter γ and factor C for all SVM in the nodes of a DDAG. SVM-Light was used for the nodewise optimization [6]. We performed 5 to 18 training passes for each of the ten kernels with different parameter sets. For each of these ten model-sequences we selected the model with minimal test-error.

For each of the ten best SVM-DDAGs, we report detailed statistics. In Table 2 we list the kernel-parameter γ, regularization factor C, and number of parameter sets, i.e. number of trained DDAGs, sorted by test-error rate.

Obviously the use of tangent information improves the classification performance remarkably for both basic kernel types compared to ignoring the tangent information as K^{RBF} and K^{ND}. Comparison with Table 1 shows, that the decrease in error rate is comparable to the decrease obtained by using the VSV-method. The gain is not as large as in using TD instead of Euclidean distance in nearest neighbour classification.

Kernel	Error rate [%]	γ	C	# param. sets
K^{RBF}	4.6	8	10	14
K^{RBF}_{1S}	4.1	20	10	11
K^{RBF}_{2S}	3.8	20	10	11
K^{RBF}_{MP}	3.5	20	10	9
K^{RBF}_{MN}	3.4	10	10	9
K^{ND}	5.1	1.0	10	12
K^{ND}_{1S}	5.0	0.3	1	18
K^{ND}_{MN}	4.2	0.7	10	6
K^{ND}_{MP}	3.9	0.3	70	11
K^{ND}_{2S}	3.6	0.3	10	5

Table 2. USPS results with TD-kernels.

Kernel	Training-time [s]	Test-time [s]	Average # SVs
K^{RBF}	199	228	175
K^{RBF}_{1S}	2814	3878	267
K^{RBF}_{2S}	6057	8172	190
K^{RBF}_{MP}	2437	3394	232
K^{RBF}_{MN}	3159	3950	133
K^{ND}	224	291	177
K^{ND}_{1S}	2947	4364	298
K^{ND}_{MN}	3737	5023	176
K^{ND}_{MP}	2805	4122	282
K^{ND}_{2S}	7873	11433	269

Table 3. Details of the models.

Among all TD-kernels, the one sided kernels seem to imply the smallest gain. This might be due to the fact that they are not symmetric, which is a basic property of ordinary kernels.

We performed no excessive parameter-optimization and therefore obtained slightly worse absolute values in the case of the standard rbf-kernel than the 4.4% presented in [9]. Similar to the values mentioned there, the absolute result cannot compare with 4.0% obtained by the nodewise optimized SVM-graph in Table 1, as we did not perform nodewise parameter-optimization, but used fixed global parameters.

In Table 3 we list details of the resulting SVM-DDAGs, i.e. training time, test time (on a standard 1 GHz-PC) and average number of SVs per DDAG-node.

The major problem of our approach seems to consist in the slowdown of factor $\geq$ 12 in both training and classification caused by the calculations of the tangents and the tangent distances. Quantitative comparisons to other methods are not possible, as we did not implement them and no runtime results are available in the literature. In [12] a factor of 2 is reported for the VSV method using two tangent directions. The definitely larger slowdown factor for the case of using all seven tangents is however not available.

Among the symmetric TD-kernels there seems to be no preferable choice with respect to the error rate. Additionally regarding the time complexity, the kernels based on the midpoint-TD seem to be the best choice.

6. Conclusion and perspectives

We successfully demonstrated the generation of new SVM-kernels by substituting distance-measures in distance-based kernels. We defined modifications of TD, which combine the advantages of existing formulations: the symmetry of the two sided TD and the computational ease of the one sided TD. We presented a new method for incorporating a-priori knowledge consisting of transformation invariance into the SVM methodology, by introducing TD-kernels. The recognition performance is comparable to other methods, furthermore disadvantages from existing methods are circumvented.

The presented results can definitely be refined by more parameter sets and performing parameter-optimization for each SVM node in the multiclass decision graph.

After our initial experiments it seems promising to perform further comparisons, in particular with regard to runtimes, with the VSV and kernel-jittering approach.

The experiments can be extended in various ways. First to larger databases, e.g. the MNIST digit database, which is also widely used in literature. Application to other areas than OCR would also be interesting in order to confirm the usability of our approach. Further distance based kernels can be implemented, e.g. $K(\mathbf{x}, \mathbf{x}') := -\log(1 + \|\mathbf{x} - \mathbf{x}'\|^{\gamma})$, cf. [2, 10].

Our result is a confirmation that the class of applicable kernels is not restricted to c.p.d. kernels, where *applicable* means producing accurate results. Although the theoretical property of kernels being c.p.d. is necessary for global optimality statements, in practice this is not always the case. In fact this might be seen as a general strategy for real problems: When designing suitable problem-dependent kernels, giving up the property of (conditional) positive definiteness leads to increased flexibility in incorporating a-priori-knowledge while it can preserve or even increase accuracy or speed.

References

[1] C. Bahlmann, B. Haasdonk, and H. Burkhardt. Handwriting recognition with support vector machines – a kernel approach. In *Proceedings 8th International Workshop on Frontiers in Handwriting Recognition*, 2002, in press.

[2] C. Berg, J. Christensen, and P. Ressel. *Harmonic analysis on semigroups: theory of positive definite and related functions*. Springer, 1984.

[3] C. J. C. Burges. Geometry and invariance in kernel based methods. In *Advances in Kernel Methods — Support Vector Learning*, pages 89–116. MIT Press, 1999.

[4] H. Burkhardt and S. Siggelkow. Invariant features in pattern recognition – fundamentals and applications. In *Nonlinear Model-Based Image/Video Processing and Analysis*, pages 269–307. John Wiley & Sons, 2001.

[5] D. DeCoste and B. Schölkopf. Training invariant support vector machines. *Machine Learning*, 46(1):161–190, 2002.

[6] T. Joachims. Making large–scale SVM learning practical. In *Advances in Kernel Methods — Support Vector Learning*, pages 169–184. MIT Press, Cambridge MA, 1999.

[7] D. Keysers, J. Dahmen, T. Theiner, and H. Ney. Experiments with an extended tangent distance. In *Proceedings 15th International Conference on Pattern Recognition, vol. 2*, pages 38–42. IEEE Computer Society, 2000.

[8] D. Keysers, W. Macherey, J. Dahmen, and H. Ney. Learning of variability for invariant statistical pattern recognition. In *ECML 2001, 12th European Conference on Machine Learning, LNCS, 2167*, pages 263–275. Springer, 2001.

[9] J. Platt, N. Cristianini, and J. Shawe-Taylor. Large margin dags for multiclass classification. In *Advances in Neural Information Processing Systems, 12*, pages 547–553. MIT Press, 2000.

[10] B. Schölkopf. The kernel trick for distances. TR MSR 2000-51, Microsoft Research, Redmond, WA, 2000.

[11] B. Schölkopf, C. Burges, and V. Vapnik. Extracting support data for a given task. In *Proceedings First International Conference on Knowledge Discovery & Data Mining*, pages 252–257. AAAI Press, 1995.

[12] B. Schölkopf, C. Burges, and V. Vapnik. Incorporating invariances in support vector learning machines. In *Artificial Neural Networks — ICANN'96, LNCS, 1112*, pages 47–52. Springer, 1996.

[13] P. Simard, Y. Le Cun, and J. Denker. Efficient pattern recognition using a new transformation distance. In *Advances in Neural Information Processing Systems, 5*, pages 50–58. Morgan Kaufmann, San Mateo, CA, 1993.

[14] P. Y. Simard, Y. A. LeCun, J. S. Denker, and B. Victorri. Transformation invariance in pattern recognition — tangent distance and tangent propagation. In *LNCS, 1524*, pages 239–274. Springer, 1998.

[15] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1996.