In: Stilla U, Gamba P, Juergens C, Maktav D (Eds) JURSE 2011 - Joint Urban Remote Sensing Event --- Munich, Germany, April 11-13, 2011

SVM-Based Road Verification with partly Non-Representative Training Data

Marcel Ziems, Christian Heipke, Franz Rottensteiner
Institute of Photogrammetry and GeoInformation
Leibniz Universität Hannover
Hannover, Germany
{ziems, heipke, rottensteiner}@ipi.uni-hannover.de

***Abstract*—In this paper we present a SVM-based method for automatic quality control of a road database in urban areas. The road verification is carried out by comparing the database objects to high-resolution aerial imagery. The method is trimmed to produce reliable results even if the training data selection is partly non-epresentative. A reliability metric is assigned to the SVM decision that is based on the distance of a test object to the training data. This metric can be applied to any SVM-based classification task. Our experiments show that the classifier is very reliable in only accepting road objects that are actually correct.**

I. Introduction

For many machine learning approaches a representative training dataset is a pre-condition. However, for classification problems on the basis of remote sensing data this is quite hard to realize, especially if rather application-oriented class definitions are directly used, or if only a few training data are available.

For the purpose of automated road verification a modular system was presented in [1] that combines several state-of-the-art road detection modules to form a more general solution. The data fusion concept behind the approach requires a confidence or reliability metric for each module. One of the modules carries out an object-wise classification based on Support Vector Machines (SVM) [2] to decide if a road segment from the database actually belongs to a road or not. We require a reliability metric for the integration of this module into the combined system. In order to be able to deal also with non-representative training data we propose a new reliability metric for SVM that is based on the distance of a test point from the training data.

The SVM classifier has proven to be well-suited for many applications in pattern recognition [2]. Compared to Bayes methods it has been noted as a drawback of SVM that it does not assign a probability to the classification result [3], so that there is no straight-forward definition of a reliability metric. There have been approaches to derive a probabilistic output from the distance of a test point to the hyperplane separating the classes. An example is [4], where a logistic function is applied to define class posterior probabilities depending on the test point's distance to the hyperplane. In [5] an interpretation based on analytical geometry has been used to link the SVM results to a posterior probability. In [6], probability estimates are derived for a multiclass SVM by a pair-wise coupling strategy using the basic idea of [4]. In [7] two metrics for the quality of the classification are presented. The static reliability measure takes the size of the margin as a quality measure, whereas the dynamic reliability measure analyses the classes of the k-nearest neighbours of a test sample in the training set. These methods focus on the overlap of the trained classes as the limiting factor for the reliability of the SVM result, assuming the reliability to improve with the distance from the hyperplane.

A test sample to be classified may be far away from the training clusters (Fig. 1), and as a consequence, the classification result may become very uncertain without indication by any of the reliability measures cited so far. With the concept of Support Vector Domain Description (SVDD) [8], also known as one-class SVM, outliers from a data distribution, e.g. the training data, can be detected effectively. There have been geometrical interpretations of the SVDD concept in [9]–[11], where it was applied to pattern denoising problems. In this paper, we combine the SVDD concept from [8] and its geometric interpretation from [9]–[11] to derive a new reliability metric for SVM classifiers.

II. Method

The goal of the method is the verification of a road database. In a first step the image is segmented based on information from the database: for each road object a region of interest is defined. An image region corresponds to a road object in the image if the database information is correct; otherwise, it belongs to the background (Fig. 2). The verification problem is thus turned into a classification problem for the two classes *road* and *background* which we solve by SVM. We use the SVM classifier in the implementation of the open-source library LIBSVM [12]. For the tests presented in this paper, we used an RGB feature space. The features were selected from each band such that changes in surface properties of the road object, shadow effects on the road and road markings are reflected in the feature space. For more comprehensive information concerning the feature selection, refer to [1].

The training data for the SVM are selected manually and include examples for road and background regions. The background regions are realised by shifting the road region

978-1-4244-8657-1/11/$26.00

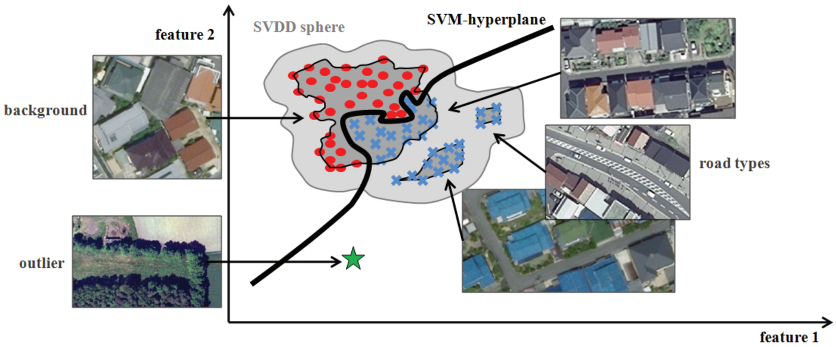

Fig. 1. In a two-dimensional feature space the black line is the SVM hyper-plane separating roads (blue) from non-roads (red). The grey regions enclose the training data. During the testing phase points outside the grey regions, such as the green star corresponding to a row of trees, are considered to be outliers. Note that the two-class SVM will classify it as a road.

Fig. 2. Segments generated from the database information. Yellow: correct roads; red: incorrect road.

in a direction orthogonal to the road axis. Thus, the non-road regions have a similar shape and size as the road regions.

Due to the high variety of objects in urban scenes there is obviously no guarantee that every possible appearance of a road is represented well by the trained model, and the situation is even worse for the *background* objects. In order to detect critical situations, we combine the two class SVM with the SVDD to distinguish not only between the classes *road* and *background*, but also a third class, *outlier*. Furthermore, we propose a reliability metric that is realised as a distance measure from each test sample to the training data as represented by the SVDD (Fig. 1). In the following section we give a short overview of the SVDD concept and show how the distance is defined for our task.

A. The SVDD Concept

The SVDD approach constructs a hypersphere (a sphere in a multi-dimensional space) around the training data. This hypersphere encloses most of the training data by minimal volume. In [8] the problem of finding this minimum hypersphere, represented by its center a and radius R, is formulated as

$$R^2 + C \cdot \sum_i \xi_i \rightarrow min \tag{1}$$
$$s.t.\ (||x_i - a||^2 \leq R^2 + \xi_i) \wedge (\xi_i \geq 0) \quad \forall i$$

where x_i represents a training sample in the input feature space. The slack variables ξ_i allow a number of training samples to lie outside the hypersphere and thus make the approach robust against isolated samples. The parameter C is a trade-off constant controlling the relative importance of each term. Analogously to the conventional SVM the optimisation problem is solved by introducing Lagrangian multipliers α_i. The problem can be tackled by maximising a function L with respect to α_i [8]:

$$L = \sum_i \alpha_i \langle x_i, x_i \rangle - \sum_{i,j} \alpha_i \alpha_j \langle x_i, x_j \rangle \tag{2}$$
$$s.t.\ 0 \leq \alpha_i \leq C \quad \forall i \quad , \quad \sum_i \alpha_i = 1$$

After training, only the feature vectors with non-zero coefficients α_i are used for testing. They are called the support vectors of the hypersphere SV_h. As a simple shaped form like a hypersphere in the original feature space is not necessarily a good description, the inner products $\langle \cdot, \cdot \rangle$ are replaced by a kernel function $K(x, y) = \langle \phi(x), \phi(y) \rangle$, where ϕ is a mapping of the data into a new feature space. Following [8], the new feature space is denoted as *feature space* in differentiation to the *input space*, which is the original feature space. We use the Gaussian radial basis function as the kernel function:

$$K(x, y) = e^{-\frac{d(x,y)^2}{2\sigma^2}} \tag{3}$$

$$where \qquad \sigma^2 = \frac{1}{N(N-1)} \sum_{i,j} ||x_i - x_j||^2 \tag{4}$$

In Eq. 3, $d(x, y)$ is a distance between x and y, e.g. $d(x, y) = ||x - y||$ in case of the Euclidean distance. The width σ of the Gaussian Kernel influences the generalization of the SVDD boundary. Simelar to [10] we select a relatively small kernel width, namely the average square distance of all training samples (Eq. 4). A test sample z lies outside the SVDD if $R - d_a(z) < 0$, where $d_a(z)$ is the distance of $\phi(z)$ to the sphere centre $a = \sum_i \alpha_i x_i$ in feature space. Using $K(z, z)$=1 results in [8]:

$$d_a(z)^2 = 1 - 2\sum_j \alpha_j K(x_j, z) + \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) \tag{5}$$
$$\forall x \in SV_h$$

The radius R can be expressed by the distance between a and any support vectors that reside on the boundary (SV_b) [8]:

$$R^2 = 1 - 2\sum_j \alpha_j K(x_j, x) + \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j) \tag{6}$$
$$\forall x \in SV_b$$

B. Distances Based on the SVDD

Unlike [8] we do not want to classify all the samples outside the SVDD as outliers because they may reside very close to the decision boundary in input space. In such a case we assume them to be still covered by the trained model and thus, we want to use the distance $d_\nu(z) = ||z - \nu||$ of a point z to its nearest point ν on the SVDD boundary in input space as a measure for the reliability of the classification. For that purpose, the input space distance $d_\nu(z)$ has to be derived from the feature space distances $d_a(z)$ and R. Through the support vector approach the input data are mapped to a manifold in feature space [13]. For the Gaussian radial basis function kernel the mapped feature space is of infinite dimension and the manifold looks like a sphere around the origin, because $||\phi(x)||^2 = 1, \forall x \in \Re^d$ [13], where d is the dimension of the input space. For the Euclidian distance in feature space the following relationship with the input space distance is given in [14]:

$$||\phi(z) - \phi(y)||^2 = 2\left(1 - e^{-\frac{||z-y||^2}{2\sigma^2}}\right) \tag{7}$$

According to Eq. 7 it is obvious that finding the shortest distance in feature space is equivalent to finding the shortest distance in input space. In feature space, the nearest boundary point $\phi(\nu)$ is on the shortest path from the SVDD center a to a test sample $\phi(z)$. Thus, the position vectors of $\phi(\nu)$, $\phi(z)$, and a are coplanar and the geometrical interpretation, displayed in Fig. 3 holds [11]. As the feature space distances $d_a(z)$ and R are available from Eq. 6 and 7 and as $||\phi(z)|| = ||\phi(\nu)|| = 1$ is true for the Gaussian kernel, the angles α, β, and γ in Fig. 3 can be determined:

$$\cos\gamma = \frac{1 + ||a||^2 - d_a(z)^2}{2||a||} \quad , \quad \cos\beta = \frac{1 + ||a||^2 - R^2}{2||a||} \tag{8}$$

$$||\phi(z) - \phi(\nu)||^2 = 2 - 2\cos\alpha \tag{9}$$

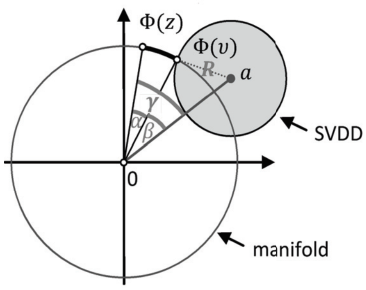

Fig. 3. 2D projection of the SVDD with center a and radius R. Both the test sample $\phi(z)$ and its nearest boundary point $\phi(\nu)$ lie on the manifold.

Replacing the feature space distance $||\phi(z)-\phi(\nu)||$ in Eq. 7 by the result of Eq. 9 gives the required distance $d_\nu(z)$ (see also [11]):

$$d_\nu(z) = ||z - \nu|| = \sigma\sqrt{-2\ln[\cos(\gamma - \beta)]} \tag{10}$$

C. Outlier Detection

After determining R and a by the SVDD method, a test sample z will be considered as reliable if it is enclosed by the hypersphere, or if it is outside the hypersphere and

$$d_\nu(z) \leq P \cdot d_{avg} \tag{11}$$

The SVM classification result is only accepted if the test sample z is reliable, otherwise (i.e., if z is outside the hypersphere and $d_\nu(z) > P \cdot d_{avg}$) z is classified as an *outlier*. The distance d_{avg} in Eq. 11 is calculated as the average of all the distances between training samples and their nearest boundary points on the hypersphere. Here only training samples that are inside the hypersphere are considered, e.g. those which are not support vectors of the hypersphere. The weight parameter P controls the actual decision boundary. Its effect on the results will be investigated in section III.

The scheme for database verification described in [1] is based on decision level fusion of the individual modules for road verification. In this context, the distance $d_\nu(z)$ is not used to distinguish outliers from inliers, but it is used to derive a reliability measure for the decision of the SVM. This reliability measure is a strictly monotonic decreasing function of $d_\nu(z)$. For details of the fusion concept, refer to [1].

III. Results and Evaluation

In order to evaluate the proposed verification strategy, we used a road dataset of a suburban test site in Uraga (Japan; Fig. 4). The reference was created manually. There is a significant difference between the database and the image due to a large redevelopment zone that stretches from the north-west to the south-east of the scene. For training the SVM and the SVDD we used the regions marked by blue lines in Fig. 4. As the road segments used for training are all placed in the centre of the scene where the roads and their surroundings have a fairly homogeneous appearance, the characteristics of the redevelopment area and forest areas in the periphery are not well represented in the training data. The kernel width σ was determined according to Eq. 4. The constant C (Eq. 1) was set to $C = 0.98$. Such a definition will result in a low degree of generalisation, i.e., the SVM hyperplane and the SVDD hypersphere will be determined so that there will be few training samples on the wrong side of the decision boundary or outside the hypersphere, respectively. Furthermore, SVM and SVDD are applied with the same input for the training.

TABLE I
Confusion matrices for three scenarios defined in the text.

system / reference	road	non-road	outlier
- SVM			
road	154	73	-
non-road	14	35	-
- SVM + SVDD (P=0)			
road	113	12	102
non-road	0	2	47
- SVM + SVDD (P=2)			
road	138	17	72
non-road	1	7	41

Table I shows the confusion matrices for the classification results achieved in three different scenarios. In the first scenario, only the two-class SVM is used and no *outlier* class is considered. In this scenario, $73 + 14 = 87$ of 276 test objects (32%) are misclassified. In the second scenario, we combined the SVM with the SVDD approach, but only considering test points inside the hypersphere, i.e. using $P = 0$ (Eq. 11). This resulted in a reduced error rate of about 5%, but only 49% of the regions were classified into road and background objects. In the third scenario, we again combined the SVM with the SVDD approach, this time also accepting points outside the

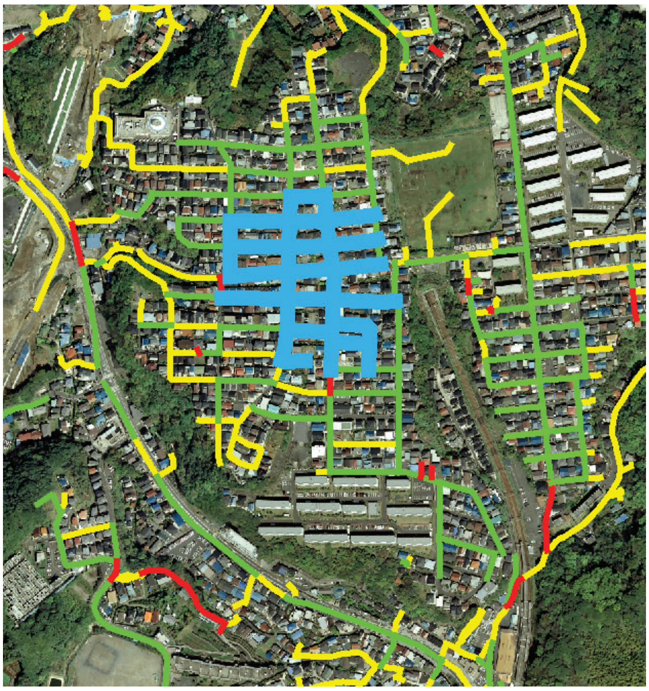

Fig. 4. The Uraga test site (size: 0.76 km x 0.92 km; GSD: 0.2 m), superimposed with the classifiation results for $P = 2$ (316 road database objects). Blue: training regions (40 road, 80 background); green: correctly classified objects; red: incorrectly classified objects; yellow: outliers.

hypersphere with $P = 2$ (Eq. 11). The percentage of the wrongly classified objects is still low (7%), whereas 60% of the test samples were accepted. This scenario is shown in Fig. 4. It becomes clear that the roads that have a similar appearance and surrounding as the training objects in the centre of the scene are considered to be reliable, and they are largely correct. The outliers are largely found where due to different surroundings the roads have a different appearance from the training data (e.g. in the forest) or where incorrect data base objects have no correspondence in the *background* training samples (e.g. in the redevelopment zone). Comparing the first and the third scenario, the two-class SVM classifies an additional 40% of the test samples, but of these samples, about 50% are incorrect. One can assume that the training data are only representative for about 60% of the test data. Fig. 5 shows the percentages of test samples considered reliable (*coverage*) and misclassifications (*error ratio*) as functions of P. It turns out that only for smaller values of P ($P < 4$), the distance to the hypersphere contributes a functional relation for the reliability metric. Given the fact that the detection of errors in the database is the most important goal of quality control, a small value such as the one used in our third scenario seems to be a reasonably good choice.

IV. Conclusions and Outlook

The results presented in this paper show that outlier detection helps to avoid wrong decisions of a two-class SVM if the training data are non-representative. They also show that in such a case the distance from the hypersphere can be used as a basis for detecting outliers. The experiment showed that the percentage of objects, for which a decision has been taken, might be low if non-representative training data are

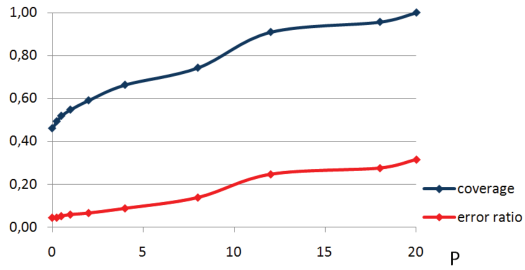

Fig. 5. Influence of the parameter P on the coverage and the error rate.

used. However, the significant reduction of wrong decisions allows the combination with other classification approaches without compromising the system reliability. It is the rationale of our decision level fusion concept to integrate several modules having complementary strength in road classification [1]. Furthermore, the outputs of the method can be utilized in any SVM-based scenario to advice the operator during the testing phase if additional training data should be used to classify more objects. Additionally, an active learning strategy, which automatically selects training samples with maximal distance to the SVDD boundary could be intergrated. In future, more comprehensive tests are required to define a suitable function of the distance to the hypersphere for the integration of the module into the overall road database verification system. It is further planned to introduce the posterior probabilities from [4] for the points that lie in areas near the separating hyperplane.

References

[1] Ziems, M., Fujimura, H., Heipke, C., Rottensteiner, F.: Multiple-model based verification of japanese road data. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Volume XXXVII-4-8-2/W9. (2010) 13–19 (on CD-ROM).

[2] Vapnik, V.N.: Statistical Learning Theory. Wiley, New-York,USA (1998)

[3] Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York, USA (2006)

[4] Platt, J.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In Smola, A., Bartlett, P.L., Schölkopf, B., Schuurmons, D., eds.: Advances in Large Margin Classifiers. MIT Press, Cambridge, MA, USA (2000)

[5] Madevska-Bogdanova, A. Nikolik, D., Curfs, L.: Probabilistic svm outputs for pattern recognition using analytical geometry. Neurocomputing **62** (2004) 293–303

[6] Wu, T.F., Lin, C.J., Weng, R.C.: Probability estimates for multiclass classification by pairwise coupling. Journal of Machine Learning Research **5** (2004) 975–1005

[7] Liu, Y., Zheng, Y.F.: One-against-all multi-class svm classification using reliability measures. In: International Joint Conference on Neural Networks. Volume 2. (2005) 849–854

[8] Tax, D.M.J., Duin, R.P.W.: Support vector data description. Machine Learning **54**(1) (2004) 45–66

[9] Kwok, J.T.Y., Tsang, I.W.H.: The pre-image problem in kernel methods. IEEE Transactions on Neural Networks **15**(6) (2004) 1517–1525

[10] Park, J., Kang, D., Kwok, J., Lee, S.W., Hwang, B.W., Lee, S.W.: SVDD -based pattern denoising. Neural Computation **19**(7) (2007) 1919–1938

[11] Guo, S.M., Chen, L.C., H., T.S.: A boundary method for outlier detection based on support vector domain description. Pattern Recognition **42**(1) (2009) 77–83

[12] Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines (2001) Software available at http://www.csie.ntu.edu.tw/~ cjlin/libsvm/(accessed 5/5/4010).

[13] Burges, C.J.C.: Geometry and invariance in kernel based methods. In Smola, A., Bartlett, P.L., Schölkopf, B., Schuurmons, D., eds.: Advances in kernel methods: support vector learning. MIT Press, Cambridge, MA, USA (1999) 89–116

[14] Williams, C.K.I.: On a connection between kernel pca and metric multidimensional scaling. Machine Learning **46** (2002) 11–19