

# NEAREST CONVEX HULL CLASSIFIERS FOR REMOTE SENSING CLASSIFICATION

Jianjun Qing\*, Hong Huo, Tao Fang

Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, Shanghai, China –  
qjj\_email@yahoo.com.cn

Commission VII, WG VII/4

**KEY WORDS:** Pattern recognition, Classification, Image understanding, Accuracy analysis, Land cover

## ABSTRACT:

This paper introduces a new approach, nearest convex hull (NCH), for remote sensing classification. NCH is an intuitive classification method which labels the test point as the training class whose convex hull is closest to it. Some attractive advantages of this learning algorithm are the robustness to noises and the scale of training samples, the straightforward way to handle multi-class tasks, and most of all the capability of processing high dimensional and nonlinear data. In our work, we deduce the NCH algorithm again basing on theories of the computational geometry, from which a simpler implementation of it is presented. Then we apply it to real-world remote problems and compare it with two other state-of-arts classifiers:  $K$ -NN and SVM. Experiments in this paper confirm the promising performance of NCH for remote sensing classification.

## 1. INTRODUCTION

For the reason of simpleness and effectiveness,  $K$ -nearest neighbor ( $K$ -NN) algorithms have been successfully used for a lot of remote sensing classification tasks (McRoberts et al., 2002). The rule of this kind of classifiers is that they label a test object as the most common class among its  $K$  nearest neighbors (Cover and Hart, 1967), where the smallest  $K$  is one for the nearest neighbor classifier (simplest version of  $K$ -NN). In spite of the big success achieved in the past,  $K$ -NN is challenged by more and more remote applications with data of hyperspectral and high spatial resolution. For instance, it is likely to be subjected to noise affection or problems of small training samples (Muñoz-Marí and Bruzzone, 2007). And the  $K$ -NN approach was suggested that it should be cautiously used for high dimension data too (Beyer et al., 1999).

To deal with challenges met by  $K$ -NN, a number of supervised and unsupervised classifiers have been proposed subsequently. One of the most famous tools is the support vector machine (SVM) which shows powerful abilities in almost all kinds of pattern recognition problems (Burges, 1998). Unlike  $K$ -NN, SVM was originally designed for classification of two categories and was built upon a complex statistical theory. Preferable advantages of SVM are listed below. Firstly, the principle of structural risk minimization equips SVM with a high ability to generalize. Secondly, since SVM essentially solves a convex optimization problem, its optimal solution is unique and global. Furthermore, SVM processes high-dimensional data easily through the kernel trick. Applications of SVM in remote sensing fields can be found in (Brown et al., 2000; Melgani and Bruzzone, 2004; PAL and MATHER, 2005). Nevertheless, one apparent defect of SVM is the extension for multi-classification, which is not direct and is still an ongoing research. For example, to choose a multi-class strategy among approaches one-versus-one, one-versus-all, all-versus-all, decision directed acyclic graph (Platt et al., 2000), pairwise coupling (Hastie and Tibshirani, 1998), error correcting output code (Dietterich and Bakiri, 1995) etc., which criteria should

we prefer: accuracy or efficiency? How to handle unclassifiable regions effectively without loss of accuracy?

In this paper, we introduce the nearest convex hull (NCH) (Nalbantov et al., 2007), a new classifier sharing some ideas with both  $K$ -NN and SVM, for remote sensing classification. NCH is an intuitive geometric classification method. According to Nalbantov's state, NCH assigns the test point to the class whose convex hull is closest to it. Several attractive properties of NCH are: 1. Like  $K$ -NN approach, NCH determines the proximity of the test object to a given class without considering samples from other classes. 2. NCH also classifies multi-class problems in a straightforward way. 3. With the benefit from the kernel trick, dealing with high dimension data or nonlinear data is also relatively easy for NCH. 4. Because eliminating one member point of a convex set doesn't or only locally affects the whole convex hull, NCH is robust to issues of small training samples and noise. The major work of this paper is dedicated to evaluating the performance of NCH for remote sensing applications. In addition, we improve Nalbantov's method on implementation, from which NCH can be calculated using the optimization formulation of SVM for both separable and inseparable cases.

The remainder of this paper is structured as follows. Section 2 briefly reviews some related theories of computation geometry and gives the idea of NCH algorithm. Section 3 describes the implementation of the new learning algorithm. In section 4, experiments are carried out on two real-world remote sensing data sets including one benchmark data set from UCI repository of machine learning databases (Blake and Merz, 1998) and one SPOT5 image of Shanghai. For comparison, algorithms  $K$ -NN and SVM are also evaluated. Finally, conclusions are given in Section 5.

## 2. THEORIES OF THE NEAREST CONVEX HULL

### 2.1 The convex hull and its properties

The convex hull of a set, denoted  $conv(S)$ , is the smallest convex set containing  $S$  (Bertsekas et al., 2003). For a convex

set  $S$  with a finite number of samples  $x_0, \dots, x_{m-1}$ , the convex hull can be calculated by all convex combinations of these samples,

$$\begin{aligned} \text{conv}(S) &= \{x \mid x = \sum a_i x_i\}, \\ \text{s.t. } \sum a_i &= 1, \quad a_i \geq 0, \quad i = 0, \dots, m-1. \end{aligned} \quad (1)$$

Convex hull has many properties, of which the following three are the most useful for pattern recognition tasks (Bertsekas et al., 2003; Yang and Cohen, 1999):

1. According to Krein-Milman theorem, a compact set  $S$  is equal to the convex hull of its extreme points. In this sense, not the whole but the extreme points are enough to compute the convex hull of a training set.
2. Convex hulls are affine invariant, i.e., the convex hull undergoes the same affine transformation as its sample set does.
3. Convex hulls have local controllability. By this property, adding or eliminating an extreme (or non-extreme) point from a training set will partly (or never) modify the convex hull of this set, which also means that convex hulls are not sensitive to noises.

### 2.2 Convex hulls for classification

Very lately, Nalbantov (Nalbantov et al., 2007) proposed the nearest convex hull (NCH) algorithm. The main idea of this classifier comes from the intuition for classification tasks. As we know, to label a new object, the most intuitive way is to assign it to the class having the minimal distance. Here a feasible choice of evaluating the distance from the test object to one class is using the distance from it to that class's convex hull. Therefore the rule of NCH is to assign the test to the class whose convex hull is nearest (Nalbantov et al., 2007). In the following we'll give an example to illustrate this learning algorithm in more detail.

A simple synthetic problem of multi-category is presented in Fig. 1. Look at this figure, three classes ( $C_0$ ,  $C_1$  and  $C_2$ ) of training samples can be found, where points of each category are enclosed by their convex hulls (bordered by gray curves). For the vision comparison, we discuss two other state-of-arts classifiers  $K$ -NN and SVM at first. As illustrated in Fig. 2 (a), set  $K = 3$ ,  $K$ -NN approach will draw a sphere (or circle on a plane) around the test point  $p$  with only three training samples inside. Obviously the final label of point  $p$  in this figure will be class  $C_0$ , since two of three training points in the enclosure are from this class. Though  $K$ -NN is always thought as a simple and powerful nonparametric technique of pattern recognition, it

is not robust to noises or high dimensional data (Muñoz-Mari and Bruzzone, 2007, Beyer et al., 1999).

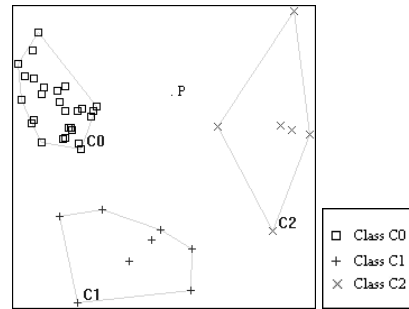


Figure 1. Three-classification task and a test point  $p$

In contrast to the simpleness of  $K$ -NN, SVM is famous for its high generalization performance. By SVM's rule, the optimization hyperplane maximizing the "margin" between two classes is the decision plane to classify a test sample. From Bennett's geometric interpretation (Bennett and Bredensteiner, 2000), in the separable case, finding the maximum margin between the two sets is equivalent to finding the closest points on two convex hulls of these sets. Take the Fig. 2 (b) as an example, the hyperplane  $H_{12}$  will be the separating plane between class  $C_1$  and class  $C_2$ , for it rightly cuts the shortest line segment connecting the two convex hulls into two equal parts at  $90^\circ$ . Using the one-versus-one strategy, the test point  $p$  will be assigned as class  $C_2$ . One drawback of SVM for multi-class applications can also be demonstrated in this figure, where an unclassifiable region (the shade region in this figure) exists.

With respect to the NCH algorithm, the class of minimum distance is chosen to predict a test point. For the above  $n$ -class ( $n=3$ ) classification task, NCH has the following decision function:

$$\text{class}(x) = \arg \min_{k=0, \dots, n-1} d_k(x, \text{conv}(C_k)) \quad (2)$$

where  $d_k(x, \text{conv}(C_k))$  is the distance between the unclassified sample  $x$  and the convex hull of class  $C_k$ .

Note, to be distinguished with the parameter  $K$  in  $K$ -NN,  $k$  is used to represent the  $k$ -th class of a classification task in this work. Fig. 2 (c) demonstrates the idea of NCH algorithm to estimate the label of the test object  $p$ . In this demo, distance  $d_{pC2}$  is the shortest one, thus object  $p$  belongs to class  $C_2$  by NCH.

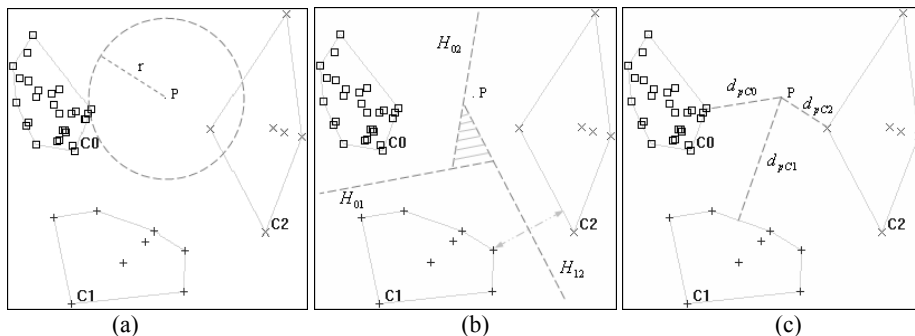


Figure 2 Classify point  $p$  by different algorithms  
(a)  $K$ -NN with  $K=3$ , (b) SVM with the linear kernel, and (c) NCH with the linear kernel.

### 3. THE IMPLEMENTATION OF NCH

Consider a more common n-class classification for the formula (2), where the k-th class has  $m_k$  training samples. From the above analysis, one of the key questions of NCH's approach is the calculation of the distance  $dk(x, \text{conv}(C_k))$ . Assume that all training samples of the k-th class forms a new class A and the test point  $x$  forms a new class B. With the definition (1), calculation of the k-th distance  $dk(x, \text{conv}(C_k))$  can be transformed into solving the following quadratic optimization problem:

$$\begin{aligned} \min_{\mu, \nu} \quad & \frac{1}{2} \|\mathbf{x}'_{kA} \boldsymbol{\mu} - \mathbf{x}'_{kB} \boldsymbol{\nu}\|^2 \\ \text{s.t.} \quad & \mathbf{e}' \boldsymbol{\mu} = 1, \mathbf{e}' \boldsymbol{\nu} = 1, \mathbf{0} \leq \boldsymbol{\mu} \leq \sigma_A \mathbf{e}, \mathbf{0} \leq \boldsymbol{\nu} \leq \sigma_B \mathbf{e} . \end{aligned} \quad (3)$$

where  $\mathbf{x}_{kA} = (x_0, \dots, x_{m_k-1})'$ ,  $\mathbf{x}_{kB} = x$ . In formula (3), vector  $\boldsymbol{\mu}$  is the convex coefficient vector of class A, and vector  $\boldsymbol{\nu}$  of class B. Parameters  $\sigma_A$  and  $\sigma_B$  are the reduced factors for convex hulls of class A and B respectively.

According to Bennett's similar conclusion in (Bennett and Bredensteiner, 2000), optimization problem (3) can be proved to be the dual problem of the C-Margin problem with the following formulation:

$$\begin{aligned} \min_{\hat{\boldsymbol{w}}, \hat{\boldsymbol{\xi}}, \hat{\boldsymbol{\eta}}, \alpha, \beta} \quad & \frac{1}{2} \|\hat{\boldsymbol{w}}\|^2 + (\sigma_A \hat{\boldsymbol{\xi}} + \sigma_B \hat{\boldsymbol{\eta}}) - (\alpha - \beta) \\ \text{s.t.} \quad & \mathbf{x}_{kA} \hat{\boldsymbol{w}} - \alpha \mathbf{e} + \hat{\boldsymbol{\xi}} \geq \mathbf{0} \\ & -\mathbf{x}_{kB} \hat{\boldsymbol{w}} + \beta \mathbf{e} + \hat{\boldsymbol{\eta}} \geq \mathbf{0} \end{aligned} \quad (4)$$

In formula (4),  $\hat{\boldsymbol{w}}$  is the normal of the parallel hyperplanes ( $\mathbf{x}' \hat{\boldsymbol{w}} = \alpha$  and  $\mathbf{x}' \hat{\boldsymbol{w}} = \beta$ ) separating class A and B with a margin  $(\alpha - \beta) / \|\hat{\boldsymbol{w}}\|$ . Parameters  $\hat{\boldsymbol{\xi}}$  and  $\hat{\boldsymbol{\eta}}$  are soft margin errors to make all training samples be separable. Furthermore, C-Margin problem (4) is the equivalent problem of the inseparable SVM problem as follows:

$$\begin{aligned} \min_{\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\eta}} \quad & \frac{1}{2} \|\boldsymbol{w}\|^2 + (\Gamma_A \boldsymbol{\xi} + \Gamma_B \boldsymbol{\eta}) \\ \text{s.t.} \quad & \mathbf{x}_{kA} \boldsymbol{w} + b \mathbf{e} + \boldsymbol{\xi} \geq \mathbf{e} \\ & -\mathbf{x}_{kB} \boldsymbol{w} - b \mathbf{e} + \boldsymbol{\eta} \geq \mathbf{e} \end{aligned} \quad (5)$$

In formula (5),  $\mathbf{x}' \boldsymbol{w} + b = 0$  is the separating hyperplane to classify class A and B according to the maximum margin rules. In order to address inseparable cases, slack variables  $\boldsymbol{\xi}$  and  $\boldsymbol{\eta}$  are employed.  $\Gamma_A$  and  $\Gamma_B$  are weight vectors for the unbalanced training data.

A special relationship here is that if  $\hat{\boldsymbol{w}}^*$  is the KKT point of C-Margin problem (4) and  $\boldsymbol{w}^*$  is the KKT point of SVM problem (5), then the formulation below exists:

$$\begin{cases} \hat{\boldsymbol{w}}^* = \boldsymbol{w}^* / \rho, \\ \rho = \mathbf{e}' \boldsymbol{a}^* / 2, \end{cases} \quad (6)$$

where  $\boldsymbol{a}^*$  is the optimal solution of the following dual problem of SVM problem (5):

$$\begin{aligned} \min_a \quad & \frac{1}{2} \sum_{i=0}^{m_k} \sum_{j=0}^{m_k} a_i a_j y_{ki} y_{kj} (\mathbf{x}_{ki} \cdot \mathbf{x}_{kj}) - \sum_{i=0}^{m_k} a_i \\ \text{s.t.} \quad & \sum_{i=0}^{m_k} a_i y_{ki} = 0 \\ & \mathbf{0} \leq \mathbf{a} \leq \Gamma' \mathbf{e} \end{aligned} \quad (7)$$

with  $\boldsymbol{\Gamma} = (\boldsymbol{\Gamma}_A', \boldsymbol{\Gamma}_B')$ . To solve the problem (7) in practice, a Mercer kernel function  $k(x_i, x_j)$  is usually introduced to replace the inner product  $(\mathbf{x}_{ki} \cdot \mathbf{x}_{kj})$  (Burgess, 1998).

However, it's not true that the two separating hyperplanes produced by methods (4) and (5) are the same (Bennett and Bredensteiner, 2000). Despite this pity, distance  $d_k(x, \text{conv}(C_k))$  can be computed through the following representation:

$$\begin{aligned} d_k(x, \text{conv}(C_k)) &= \|\mathbf{x}'_{kA} \boldsymbol{\mu} - \mathbf{x}'_{kB} \boldsymbol{\nu}\| \\ &= \|\hat{\boldsymbol{w}}^*\| \\ &= 2 \|\boldsymbol{w}^*\| / \mathbf{e}' \boldsymbol{a}^* \end{aligned} \quad (8)$$

Based on deductions above, optimization problem (3) can finally be resolved through Eq. (8). In this process, separable classification problems are the special cases for all formulas discussed, i.e., set the reduced factor  $\sigma_A$  to one or set the weight coefficient  $\Gamma_A$  to the infinite. That is, for all classification tasks, NCH can be implemented through the optimization formulation of SVM. At this point, the proposed method is superior to Nalbantov's which only handles separable cases easily.

For the three-class problem in Fig. 1, the output of algorithms K-NN, SVM and NCH are given in Fig. 3. As can be seen from this figure, both SVM and NCH have a smoother decision boundary than K-NN, in other words, the former two algorithms are less sensitive to noises than the latter. It may be due to the fact that algorithms SVM and NCH classify a new object using information of the global training samples, while K-NN does this basing on local neighbors of the test sample. Look at the center part of each figure, all three algorithms are observed to leave no unclassifiable region. However, with the multi-class method of the LIBSVM tool (Chang and Lin, 2001), SVM assigns all the primal unclassifiable points (the dashed lines region) the same class  $C_0$ .

### 4. EXPERIMENTS AND FURTHER COMMENTS

In this section, we present an extensive evaluation of the NCH technique and two other methods of K-NN and SVM for remote sensing data classification. Two groups of data sets are provided in this work: one benchmark data satimage from UCI repository of machine learning databases (Blake and Merz, 1998) and one remote sensing image of SPOT5. All features of both two data sets are linearly scaled to [-1, 1]. In these two experiments, some default settings are given below. The one-versus-one method is chosen as the SVM's multi-class strategy. To simplify the parameter sets in training stages, both NCH and SVM adopt RBF kernel only. Confusion matrixes are built to evaluate the differences of different classification results. The statistical accuracies in terms of the kappa coefficient (KC) and the overall accuracy (OA) are also recorded. In all experiments, only parameters to make each classifier achieve the top accuracy remain. In addition, it's worth noting that most of experiments in this work are finished basing on the freely available LIBSVM software packages (Chang and Lin, 2001).

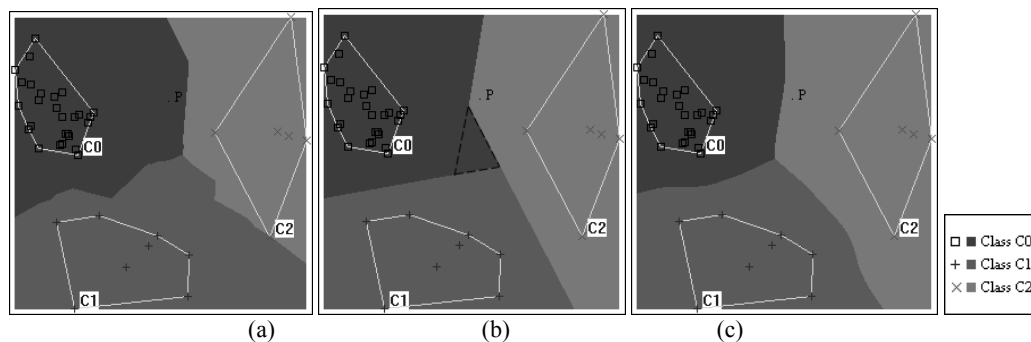


Figure 3. Results of different algorithms to classify the data in Fig.1.  
 (a) K-NN with K =3, (b) SVM with the linear kernel, and (c) NCH with linear the kernel.

#### 4.1 Classification of a benchmark data set: satimage

The preliminary experiment is on data satimage which is also one of the common benchmark databases in pattern recognition fields. This database actually is a small section of Landsat MSS imagery that consists of four digital images of the same scene in four spectral bands with a spatial resolution of about 80m × 80m. There are 4435 training samples and 2000 test samples in this database. Each sample pixel is represented by 36 features (all spectral values of its and its 3x3 neighbors’). Six different classes are used to label all samples, which are red soil (RS), cotton crop (CC), grey soil (GS), damp grey soil (DGS), soil with vegetation stubble (SVS) and very damp grey soil (VDGS).

In this experiment, optimization parameters of the SVM are  $\Gamma_A = 5e$ ,  $\Gamma_B = 5e$ , and  $\gamma = 1$ . For NCH, these parameters change as  $\Gamma_A = 1e$ ,  $\Gamma_B = \infty$ , and  $\gamma = 1$ . The only parameter  $K$  of the  $K$ -NN here is set to 3. Statistical results of different classifiers are shown in table 1, table 2 and table 3.

From table I to table III, all three classifiers show a good performance for the satimage database with the OA bigger than 90%. In detail, the best OA is 92.30% with  $KC = 0.9052$  achieved by NCH, which yields a gain of 1.7% and 0.45% with respect to  $K$ -NN and SVM. However, none of them can classify all six classes well in this database. For example, though all classifiers have a high classification accurate on the red soil class (the top is even 99.35% by NCH), they are challenged by the damp grey soil class, where the worst score observed is only 66.35% by SVM. Furthermore, as far as the statistical votes been concerned,  $K$ -NN attains one vote for the best accuracy rate on class damp grey soil, SVM gets two on class cotton crop and class soil with vegetation stubble, and NCH holds three for the class red soil, grey soil and very damp grey soil. It’s clear that in this database NCH performs better than the other two algorithms.

Real value Prediction	RS	CC	GS	DGS	SVS	VDGS	user accuracy [%]
RS	456	0	3	0	1	0	99.13
CC	0	217	1	1	2	1	97.75
GS	2	0	365	32	2	14	87.95
DGS	1	0	20	150	5	38	70.09
SVS	2	5	0	1	215	8	93.07
VDGS	0	2	8	27	12	409	89.30
producer accuracy [%]	98.92	96.88	91.94	71.09	90.71	87.02	

Overall accuracy: 90.60%  
 Kappa: 0.8846

Table 1 Classification of K-NN for satimage

Real value Prediction	RS	CC	GS	DGS	SVS	VDGS	user accuracy [%]
RS	457	0	4	0	0	0	99.13
CC	0	219	1	3	3	0	96.90
GS	2	0	372	32	1	14	88.36
DGS	0	0	13	140	2	18	80.92
SVS	2	3	1	1	222	11	92.50
VDGS	0	2	6	35	9	427	89.14
producer accuracy [%]	99.13	97.77	93.70	66.35	93.67	90.85	

Overall accuracy: 91.85%  
 Kappa: 0.8997

Table 2 Classification of SVM for satimage

Real value Prediction	RS	CC	GS	DGS	SVS	VDGS	user accuracy [%]
RS	458	0	3	0	1	0	99.13
CC	0	216	1	2	3	0	97.30
GS	1	2	380	32	0	12	88.99
DGS	0	0	6	141	2	18	84.43
SVS	2	4	1	2	219	8	92.80
VDGS	0	2	6	34	12	432	88.89
producer accuracy [%]	99.35	96.43	95.72	66.82	92.41	91.91	

Overall accuracy: 92.30%  
 Kappa: 0.9052

Table 3 Classification of NCH for satimage

#### 4.2 Classification of a high spatial resolution image

The study data shown in Fig. 4 is a SPOT5 image of Shanghai containing 1171 × 910 pixels. The image is acquired with a high spatial resolution of approximately 2.5 m. In this experiment, all samples are expected to be classified into six classes: road, barren, building, vegetation, pool, and river. Before classification, the remote sensing image can be segmented by some segmentation algorithms such as multi-resolution segmentation (Baatz and Schape, 2000). Then classification can be carried on samples of segmented regions, which is also called object-oriented classification. Therefore, besides pixels’ spectrum attributes, more information, such as shape features and texture features contained in the high spatial resolution image, can be employed to determine one test object’s class index. In this work, total 1035 image regions are generated after segmentation. Among these regions, nearly 20

samples per class are randomly selected as the training set. To get the statistical evaluation of each classifier, total 256 random regions are also obtained for testing. In addition, fourteen features of spectrum, shape and texture are extracted from each segmented object to participate in this experiment.



Fig. 4. SPOT5 Image of Shanghai

Real value Prediction	road	barren	building	vegetation	pool	river	user accuracy [%]
road	<b>13</b>	3	9	1	0	1	48.15
barren	3	<b>34</b>	12	8	1	0	58.62
building	7	9	<b>22</b>	3	0	0	53.66
vegetation	3	8	0	<b>43</b>	4	0	74.14
pool	0	2	0	6	<b>48</b>	2	82.76
river	0	0	0	0	0	<b>14</b>	100.00
producer accuracy [%]	50.00	60.71	51.16	70.49	90.57	82.35	

Overall accuracy: 67.97%  
Kappa: 0.6037

Table 4 Statistical Result Of K-Nn For Spot5 Image

Real value Prediction	road	barren	building	vegetation	pool	river	user accuracy [%]
road	<b>12</b>	2	9	1	0	1	48.00
barren	3	<b>34</b>	12	6	1	0	60.71
building	8	11	<b>22</b>	4	0	0	48.89
vegetation	3	8	0	<b>46</b>	3	0	76.67
pool	0	1	0	4	<b>49</b>	2	87.50
river	0	0	0	0	0	<b>14</b>	100.00
producer accuracy [%]	46.15	60.71	51.16	75.41	92.45	82.35	

Overall accuracy: 69.14%  
Kappa: 0.6180

Table 5 Statistical result of SVM for SPOT5 image

Real value Prediction	road	barren	building	vegetation	pool	river	user accuracy [%]
road	<b>14</b>	4	9	1	0	1	48.28
barren	2	<b>35</b>	12	7	1	0	61.40
building	7	9	<b>22</b>	4	0	0	52.38
vegetation	3	8	0	<b>46</b>	4	0	75.41
pool	0	0	0	3	<b>48</b>	2	90.57
river	0	0	0	0	0	<b>14</b>	100.00
producer accuracy [%]	53.85	62.50	51.16	75.41	90.57	82.35	

Overall accuracy: 69.92%  
Kappa: 0.6282

Table 6 Statistical result of NCH for SPOT5 image

Parameter configurations for the SPOT5 image are set as follows:  $K=1$  for  $K$ -NN,  $(\Gamma_A, \Gamma_B, \gamma) = (200e, 200e, 0.077)$  for SVM, and  $(\Gamma_A, \Gamma_B, \gamma) = (200e, 200e, 0.077)$  for NCH. Fig. 5 shows the results for methods tested in this section, and table IV to table VI give the related statistical analysis.

From Fig. 5 (a) to Fig. 5 (c), algorithms  $K$ -NN, SVM and NCH perform similarly well on classes of river, pool and most of vegetation. However, for the remainder three classes, wrong outputs or distinct differences occur often with these three classifiers. This conclusion can be further confirmed by table IV, tables V, and table VI. In the prediction of all the 256 test samples, the pool class is ideally classified by each algorithm with accuracy bigger than 90%. But for class road, barren and building, all classifiers have a bad accuracy. And the road class is the most critical. For this class, NCH exhibits the best accuracy (53.85%), whereas the worst accuracy (46.15%) is obtained by the SVM classifier. Global evaluation of these three algorithms is acquired using the OA and the KC. In terms of these two statistics, NCH again has the top OA of 69.92% with the KC = 0.6282, where OA and KC are (69.14%, 0.6180) for SVM and (67.97%, 0.6037) for KNN.

### 5. CONCLUSIONS

This paper has addressed the problem of remote sensing classification with a new algorithm NCH, which has been reported to have an excellent generalization to deal with ordinary pattern recognition problems (Nalbantov et al., 2007). The main analysis of our work aims two different objectives: 1) easier implementation of the NCH algorithm comparing with the original version and 2) evaluation of NCH's performance for common remote sensing tasks.

Basing on theories of computation geometry, we deduced NCH's algorithms in another way different from the original method. And then an easier implemental approach for it was proposed in this work, which can be simply programmed with a little modification to the famous tool LIBSVM.

The classification ability of NCH was assessed on three kinds of data sets: one synthetic data for demo example, one benchmark set of satimage, and one SPOT5 image. In our experiments, comparison was carried out with the other two state-of-arts classifier: KNN and SVM. For the synthetic data, NCH was observed to have a smooth decision boundary like SVM, where this algorithm also resolved the unclassifiable regions left by the latter. Experiments on real remote sensing datas further showed the promising performance of NCH. On the database satimage, NCH had a high OA of 92.30% which is the top rank among all three classifiers. Even for the hard task to deal with the spot5 image, NCH was also found to perform slightly better than  $K$ -NN and SVM. With all the obtained results, NCH is proved to have a good potentiality for remote sensing classifications. Two points for our future work will concentrate on the automatic choice of the parameters for the learning algorithm and on the optimization of faster prediction.

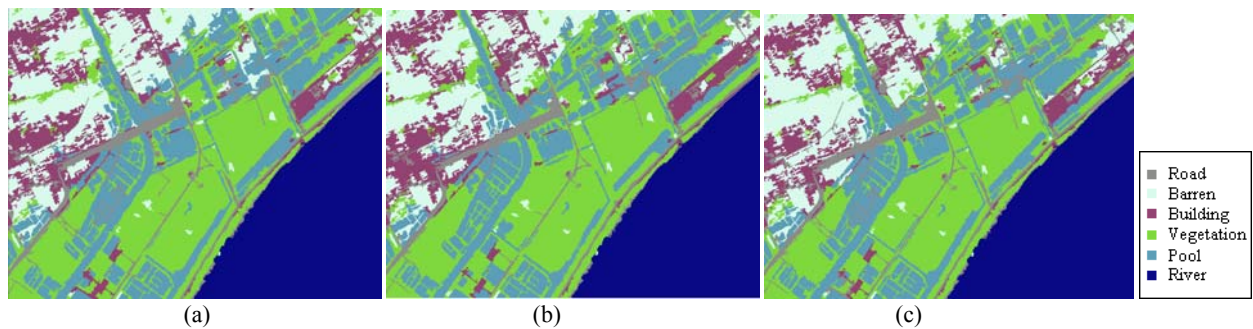


Figure 5. Results of the SPOT5 image with different algorithms  
(a) K-NN, (b) SVM, and (c) NCH

#### ACKNOWLEDGEMENTS

This research has been supported by the National Key Basic Research and Development Program of China (Grant No. 2006CB701303) and the National High Technology Research and Development Program of China (Grant No. 2006AA12Z105).

#### REFERENCE

- Baatz, M., and Schape, A., 2000, Multiresolution Segmentation: an optimization approach for high quality multi-scale image segmentation. [http://www.agit.at/papers/2000/baatz\\_FP\\_12.pdf](http://www.agit.at/papers/2000/baatz_FP_12.pdf)
- Bennett, K. P. and Bredensteiner, E. J., 2000. Duality and Geometry in SVM classifiers. *Proc. ICML*, pp. 57-64.
- Bertsekas, D. P., Nedic, A. and Ozdaglar, A. E., 2003. *Convex Analysis and Optimization*. MA, Athena Scientific.
- Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft U., 1999. When is nearest neighbor meaningful. *Lecture Notes in Computer Science*, 1540, pp. 217-235.
- Blake, C. L. and Merz, C. J., 1998. UCI Repository of Machine Learning Databases. Univ. California, Dept. Inform. Comput. Sci., Irvine, CA. [Online]. Available: <http://www.ics.uci.edu/~mlern/MLRepository.html>.
- Brown, M., Lewis, H. G. and Gunn, S. R., 2000. Linear spectral mixture models and support vector machines for remote sensing. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 38(5), pp. 2346-2360.
- Burges, C. J. C., 1998. A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*, 2, pp. 121-167.
- Chang, C. C. and Lin, C. J., 2001. *LIBSVM: A library for support vector machines*. <http://www.csie.ntu.edu.tw/~cjlin>.
- Cover, T., Hart, P., 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13, pp. 21-27.
- Dietterich, T. G. and Bakiri, G., 1995. Solving multiclass learning problems via error correcting output codes. *Journal of Artificial Intelligence Research*, 2, pp. 263-286.
- Hastie, T., Tibshirani, R., 1998. Classification by Pairwise Coupling. *The Annals of Statistics*. 26, p. 451-471.
- McRoberts, R. E., Nelson, M. D., & Wendt, D. G., 2002. Stratified estimation of forest area using satellite imagery, inventory data, and the k-nearest neighbors technique. *Remote Sensing of Environment*, 82, pp. 457-468.
- Melgani, F., Bruzzone, L., 2004. Classification of hyperspectral remote sensing images with support vector machines. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 42(8), pp. 1778-1790.
- Muñoz-Marí, J., Bruzzone, L., 2007. A support vector domain description approach to supervised classification of remote sensing images. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*. 45(8), p. 2683-2692.
- Nalbantov, G. I., Groenen, P. J. F. and Bioch, J. C., 2007. nearest convex hull classification. [http://repub.eur.nl/publications/eco\\_man/ese/ese1/127417203/](http://repub.eur.nl/publications/eco_man/ese/ese1/127417203/).
- PAL, M. and MATHER, P. M., 2005. Support vector machines for classification in remote sensing. *International Journal of Remote Sensing*, 26(5), pp. 1007-1011.
- Platt, J. C., Cristianini, N., and Shawe-Taylor, J., 2000. Large margin DAG's for multiclass classification. *Adv. Neural Inf. Process. Syst*, 12, pp. 547-553.
- Yang, Z. W. and Cohen, F. S., 1999. Image registration and object recognition using affine invariants and convex hulls. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 8(7), pp. 934-46.