Tensor-based learning machine for remotely sensed image target detection

ZHANG Lefei[1], **ZHANG Liangpei**[1], **TAO Dacheng**[2]

1. *State Key Lab of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Hubei Wuhan* 430079, *China*;
2. *School of Computer Engineering, Nanyang Technological University*, 639798, *Singapore*

Abstract: This paper proposes a new way to detect the targets in remote sensing images based on the tensor learning machine (TLM). This method is based on tensor and tensor algebra. To utilize the multidimensional data of the remote sensing image, the vector-based learning machine is generalized to the tensor-based learning machine which accepts tensors as input, then the convex optimization theory and the alternating projection procedure are used to get the solution of the TLM. TLM is tested to target detection using the hyperspectral remote sensing data and high resolution remote sensing data. The experiments demonstrate the effectiveness of the proposed method, by comparing TLM with support vector machine, the tensor learning machine can keep a high probability of successful detection and reduce the false alarm.

Key words: tensor, convex optimization, supervised learning, target detection

CLC number: TP751.1 **Document code:** A

Citation format: Zhang L F, Zhang L P and Tao D C. 2010. Tensor-based learning machine for remotely sensed image target detection. *Journal of Remote Sensing*. **14**(3): 519—533

1 INTRODUCTION

Target detection is a focused issue in remotely sensed image processing. The support vector machine (SVM) has been widely used in remotely sensed image target detection (Tan & Du, 2008). As a vector-based learning machine, it aims to find an optimal criterion to distinguish the target and background by learning the training samples of spectral vector, and use this criterion to classify all the pixels in the image. SVM for target detection considers the spectral information of the target and background. However, with the development of the spatial resolution, spectral resolution and temporal resolution of remotely sensed data, many targets must be represented as a two-dimensional or multidimensional array instead of a vector. In this situation, the efficiency of SVM decreases, which is presented by the false alarm increasing. Consequently, a new data model which could represent the multidimensional array must be introduced instead of the vector for hyperspectral and high resolution remotely sensed images target detection.

Tensor, as a data model which could represent the multidimensional array, is considered to represent the target and background objects with several or more pixels better than the vector. Tensor learning is a new research direction in machine learning and data mining. Some popular learning machines such as the support vector (SVM) (Vapnik, 1995), the minimax probability machine (MPM) (Strohmann *et al*., 1993), and Fisher discriminant analysis (FDL) (Duda *et al*., 2001), only use the one-dimensional vectors as the input samples for training. However, the data is existed as the model of tensor in reality (Li *et al*., 2008), so that the structure information would be inevitable lost if we represent the data by vector. Therefore, the tensor must be introduced as the training samples of the learning machine and the vector-based learning machine must be generalized to tensor learning machine (TLM). In this paper, the principle of tensor learning machine is presented and TLM for target detection using hyperspectral and high resolution remotely sensed images is discussed.

2 TENSOR AND TENSOR ALGEBRA

2.1 Tensor

Tensor $\boldsymbol{X} \in R^{L_1 \times L_2 \times \cdots \times L_M}$ represents a multidimensional array (Tao *et al*., 2007). M is the order of tensor $\boldsymbol{X}$ and the i^{th} dimension of the tensor is of size L_i. An element of $\boldsymbol{X}$ is denoted as $\boldsymbol{X}_{l_1,l_2,\cdots,l_M}$, where $1 \leqslant l_i \leqslant L_i$ and $1 \leqslant i \leqslant M$, l_i denotes the location of this element in the dimension or mode i.

For examples of the low-order tensors, a 0-order tensor $\boldsymbol{X} \in R$ is a scalar, a 1-order tensor $\boldsymbol{X} \in R^{L_1}$ is a vector, and a 2-order

---

Received: 2009-04-27; **Accepted**: 2009-09-02

Foundation: The 973 National Basic Research Program of China (No. 2009CB723905), the National Natural Science Foundation of China (No. 40930532 and No. 40771139), and the 863 National High Technology Research and Development Program of China (No. 2009AA12Z148).

First author biography: ZHANG Lefei (1986—), male, master student. He received the B.S. degree in sciences and techniques of remote sensing from Wuhan University, China, in 2008. His research interests include hyperspectral data analysis, high-resolution image processing, and pattern recognition. E-mail: zhanglefei.wh@gmail.com tensor $\boldsymbol{X}\in R^{L_1\times L_2}$ is a matrix, in which $\boldsymbol{X}_{l_1,l_2}$ ($1\leqslant l_i\leqslant L_i$) denotes the element of the matrix at row l_1 and column l_2. The remotely sensed image can be represented as a 3-order tensor $\boldsymbol{X}\in R^{L_1\times L_2}\times L_3$, where L_1 and L_2 denote the height and the width of the image and L_3 denotes the number of the bands, and the element $\boldsymbol{X}_{l_1,l_2,l_3}$ stands for the digital number (DN) of pixel at band l_3, row l_1 and column l_2(Fig. 1).

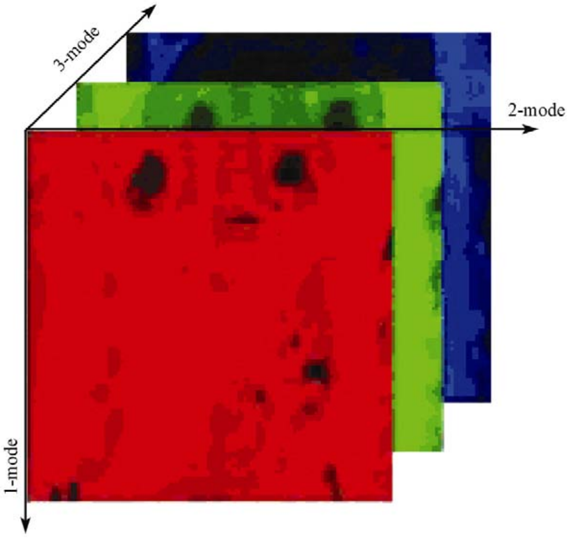


Fig. 1 3-order tensor represents a remotely sensed image

2.2 Tensor algebra

We have following definitions of the basic tensor algebra (Lathauwer, 1997):

2.2.1 Tensor outer product (or outer product)

The outer product of tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ and another tensor $\boldsymbol{Y}\in R^{L'_1\times L'_2\times\cdots\times L'_M}$ is defined by:

$$(\boldsymbol{X}\otimes\boldsymbol{Y})_{l_1,l_2,\cdots,l_M,l'_1,l'_2,\cdots,l'_{M'}} = \boldsymbol{X}_{l_1,l_2,\cdots,l_M}\boldsymbol{Y}_{l'_1,l'_2,\cdots,l'_{M'}} \quad (1)$$

2.2.2 Tensor contraction

The contraction of a tensor is obtained by equating two indices and summing over all values of the repeated indices. The contraction on tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M\times L'_1\times L'_2\times\cdots\times L'_{M'}}$ and $\boldsymbol{Y}\in R^{L_1\times L_2\times\cdots\times L_M\times L''_1\times L''_2\times\cdots\times L''_{M''}}$ is defined by:

$$\begin{aligned}&[\boldsymbol{X}\otimes\boldsymbol{Y};(1:M)(1:M)]\\&=\sum_{l_1=1}^{L_1}\cdots\sum_{l_M=1}^{L_M}(\boldsymbol{X})_{l_1,l_2,\cdots,l_M,l'_1,l'_2,\cdots,l'_{M'}}(\boldsymbol{Y})_{l_1,l_2,\cdots,l_M,l''_1,l''_2,\cdots,l''_{M''}}\end{aligned} \quad (2)$$

The condition of contraction is that tensor $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the same size at the specifically mode, in other words, the numbers of the elements in tensor $\boldsymbol{X}$ and $\boldsymbol{Y}$ are equally at specifically dimension. A contraction reduces the tensor order by 2. For example, for a $M+M'$ order tensor $\boldsymbol{X}$ and a $M+M''$ order tensor $\boldsymbol{Y}$, the result of $\boldsymbol{M}$ times contraction on $\boldsymbol{X}\otimes\boldsymbol{Y}$ is a $M+M''$ order tensor.

2.2.3 Mode-d product ($_d\boldsymbol{U}$)

It's a special type of contraction because it happens on a tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ and a matrix $\boldsymbol{U}\in R^{L'_d\times L_d}$. If the size of the d mode of tensor $\boldsymbol{X}$ is L_d, then the mode-d product $\boldsymbol{X}\times_d\boldsymbol{U}$ of a tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ and a matrix $\boldsymbol{U}\in R^{L'_d\times L_d}$ is an tensor of size $L_1\times L_2\cdots\times L_{d-1}\times L'_d\times L_{d+1}\times\cdots\times L_M$ defined by:

$$\begin{aligned}&(\boldsymbol{X}\times{}_d\boldsymbol{U})_{l_1,l_2,\cdots,l_{d-1},l'_d,l_{d+1},\cdots,l_M}\\&=\sum_{l'_d}(\boldsymbol{X}_{l_1,l_2,\cdots,l_{d-1},l_d,l_{d+1},\cdots,l_M}\boldsymbol{U}_{l'_d,l_d})\end{aligned} \quad (3)$$

The essence of this computation is a tensor contraction on a M order tensor $\boldsymbol{X}$ and a 2 order tensor $\boldsymbol{U}$. And the result of mode-d product on $\boldsymbol{X}\times_d\boldsymbol{U}$ is a M order tensor.

The mode-d product also happens on a tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ and a vector $\boldsymbol{\omega}\in R^{1\times n}$, because a vector is a special type of matrix. The mode-d product $\boldsymbol{X}\times_d\boldsymbol{\omega}_d$ of tensor $\boldsymbol{X}$ and vector $\boldsymbol{\omega}$ is an $M-1$ order tensor; and the result of M times mode-d products on tensor $\boldsymbol{X}$ and vectors $\boldsymbol{\omega}_i(I=1,2\cdots,M)$: $\boldsymbol{X}\prod_{k=1}^{M}\times_k\boldsymbol{\omega}_k$ is a 0-order tensor, or a scalar. This operation is very important in tensor learning machine and it will be used many times in following.

2.2.4 Frobenius Norm

The Frobenius Norm of a tensor $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ is defined by:

$$\begin{aligned}\|\boldsymbol{X}\|_{\text{Fro}}&=\sqrt{[\boldsymbol{X}\otimes\boldsymbol{X};(1:M)(1:M)]}\\&=\sqrt{\sum_{l_1=1}^{L_1}\cdots\sum_{l_M=1}^{L_M}\boldsymbol{X}^2_{l_1,\cdots,l_M}}\end{aligned} \quad (4)$$

The Frobenius Norm describes the size of a tensor and its square is the energy of the tensor.

3 TENSOR LEARNING MACHINE

3.1 Convex optimization

Learning models are always formulated as optimization problems (Winston *et al.*, 2002; Zangwill, 1969). Therefore, mathematical programming is the heart of the machine learning research. The convex optimization has been used in machine learning successfully, such as linear programming used in linear programming machine (LPM) (Pedroso & Murata, 1999), and quadratic programming used for Support Vector Machine (SVM).

A mathematical optimization problem has the form:

$$\begin{bmatrix}\min & f(\boldsymbol{X}) & \boldsymbol{X}\in R^L\\ s.t. & g_i(\boldsymbol{X}) & i=1,2,\cdots,m\end{bmatrix} \quad (5)$$

Here the functions $g_i(X)(I=1,2,\cdots,m)$ are the equality or inequality constraint functions and the function $f(X)$ is the objective function.

The simplest convex optimization is called a linear program (LP) when the objective and constraint functions are all affine. A general linear program has the form:

$$\begin{bmatrix} \min\limits_{\boldsymbol{X}} & f(\boldsymbol{X})=\boldsymbol{C}^{\mathrm{T}}\boldsymbol{X} \\ \text{s.t.} & \boldsymbol{A}\cdot\boldsymbol{X}\leqslant\boldsymbol{B} \\ & \boldsymbol{Ae}\cdot\boldsymbol{X}=\boldsymbol{Be} \\ & \boldsymbol{lb}\leqslant\boldsymbol{X}\leqslant\boldsymbol{ub} \end{bmatrix} \tag{6}$$

where, $\boldsymbol{X}=[\,\boldsymbol{x}_1, \boldsymbol{x}_1,\ldots, \boldsymbol{x}_n]^{\mathrm{T}}$; $\boldsymbol{C}$, $\boldsymbol{B}$, $\boldsymbol{Be}$, $\boldsymbol{lb}$, $\boldsymbol{ub}$ are vectors; and $\boldsymbol{A}$, $\boldsymbol{Ae}$ are matrixes. The optimal point only appears at the vertexes of the feasible set. So if we select all of the vertexes $\boldsymbol{X}_i$ and compute their $f(\boldsymbol{X})=C^{\mathrm{T}}\mathrm{X}$, then we can get the solution of LP.

The convex optimization problem (5) is called a quadratic program (QP) if the objective function is convex quadratic, and the constraint functions are affine. A quadratic program can be expressed in the form:

$$\begin{bmatrix} \min\limits_{\boldsymbol{X}} & f(\boldsymbol{X})=\frac{1}{2}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{QX}+\boldsymbol{C}^{\mathrm{T}}\boldsymbol{X} \\ \text{s.t.} & \boldsymbol{A}\cdot\boldsymbol{X}\leqslant\boldsymbol{B} \\ & \boldsymbol{Ae}\cdot\boldsymbol{X}\leqslant\boldsymbol{Be} \\ & \boldsymbol{lb}\leqslant\boldsymbol{X}\leqslant\boldsymbol{ub} \end{bmatrix} \tag{7}$$

where $\boldsymbol{X}=[\,\boldsymbol{x}_1, \boldsymbol{x}_1,\ldots, \boldsymbol{x}_n]^{\mathrm{T}}$; $\boldsymbol{C}$, $\boldsymbol{B}$, $\boldsymbol{Be}$, $\boldsymbol{lb}$, $\boldsymbol{ub}$ are vectors; and $\boldsymbol{Q}$, $\boldsymbol{A}$, $\boldsymbol{Ae}$ are matrixes. In a quadratic program, we use the Lagrange multiplier approach and KKT conditions to get the solution of QP (7).

3.2 Support vector machine

Support vector machine (SVM) (Vapnik, 1995) is a machine learning method based on structural risk minimization and optimization theory, which finds a classification hyperplane to maximizes the margin between the positive measurements and the negative measurements, as shown in Fig. 2.

Suppose the N training samples $x_i\in R^L(1\leqslant i\leqslant N)$ and their labels $y_i\in\{+1, -1\}$ are known, and we will find an optimal hyperplane to classify the positive training samples from negative training samples. The SVM could find the projection vector $\boldsymbol{\omega}\in R^L$ and $b\in R$ through (Burges, 1998):

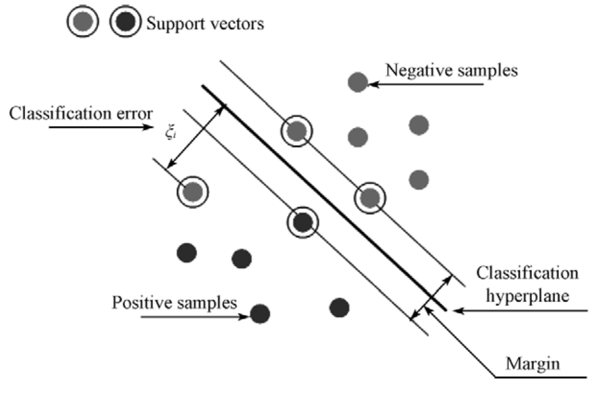


Fig. 2 SVM maximizes the margin between the positive and negative training measurements

$$\begin{bmatrix} \min\limits_{\boldsymbol{\omega},b,\boldsymbol{\xi}} & \frac{1}{2}\|\boldsymbol{\omega}\|^2+c\sum_{i=1}^{N}\xi_i \\ \text{s.t.} & y_i\left[\boldsymbol{\omega}^T\boldsymbol{x}_i+b\right]\geqslant 1-\xi_i,\quad 1\leqslant i\leqslant N \\ & \boldsymbol{\xi}\geqslant 0 \end{bmatrix} \tag{8}$$

where $\boldsymbol{\xi}=[\xi_1,\xi_2,\ldots,\xi_N]^{\mathrm{T}}\in R^N$ are additional slack variable. By introducing this additional slack variable, Eq. (8) could deal with the linearly nonseparable problem (see Fig. 2). When the classification problem is linearly separable, we can set $\boldsymbol{\xi}=0$. If we get the solution of Eq. (8), the decision function for classification is:

$$y(\boldsymbol{x})=\boldsymbol{\omega}^T\boldsymbol{x}+b \tag{9}$$

Afterward, we can use Eq. (9) to get the lables of unknown samples $x_j\in R^L(1\leqslant j\leqslant M)$.

We can use the following steps to get the optimal $\boldsymbol{\omega}$ and b:

The Lagrangian function of Eq. (8) is:

$$\begin{aligned} & L(\boldsymbol{\omega},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}) \\ & =\frac{1}{2}\|\boldsymbol{\omega}\|^2+c\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\kappa_i\xi_i \\ & \quad -\sum_{i=1}^{N}\alpha_i\left(y_i\left[\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x_i}+b\right]-1+\xi_i\right) \\ & =\frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+c\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x_\mathrm{i}} \\ & \quad -b\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}+\sum_{i=1}^{N}\alpha_i-\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\xi}-\boldsymbol{\kappa}^{\mathrm{T}}\boldsymbol{\xi} \end{aligned} \tag{10}$$

with Lagrangian multipliers α_i and κ_i $(1\leqslant i\leqslant N)$. Then we can get the partial derivative of function L by Eq.(10):

$$\begin{aligned} & \frac{\partial L}{\partial\boldsymbol{\omega}}=0 \;\Rightarrow\; \boldsymbol{\omega}=\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{x_i} \\ & \frac{\partial L}{\partial b}=0 \;\Rightarrow\; \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}=0 \\ & \frac{\partial L}{\partial\boldsymbol{\xi}}=0 \;\Rightarrow\; \boldsymbol{c}-\boldsymbol{\alpha}-\boldsymbol{\kappa}=0 \end{aligned} \tag{11}$$

Generally, we can first use the dual problem of Eq. (8) to get the multipliers α_i and κ_i:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\kappa}}\ \min_{\boldsymbol{\omega},b,\boldsymbol{\xi}}\ L(\boldsymbol{\omega},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}) \tag{12}$$

If we substitute the Eq. (11) into Eq. (12), then it can simplifying to the following form:

$$\begin{bmatrix} \max & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}y_i y_j\boldsymbol{x_i}^{\mathrm{T}}\boldsymbol{x_j}\alpha_i\alpha_j+\sum_{i=1}^{N}\alpha_i \\ \text{s.t.} & \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}=0 \\ & 0\leqslant\boldsymbol{\alpha}\leqslant\boldsymbol{c} \end{bmatrix} \tag{13}$$

If we set $\boldsymbol{Q}=[y_i y_j\boldsymbol{x}_i^T\boldsymbol{x}_j]_{1\leqslant i,j\leqslant N}$, $\boldsymbol{C}=1_{N\times 1}$, $\boldsymbol{Ae}=y$, $be=0$, $lb=0$, $\boldsymbol{ub}=C$, then the dual problem of SVM is a QP with the optimization variable $\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\cdots,\alpha_N]^{\mathrm{T}}$. We can use the convex optimization to get the solution of QP Eq. (13), after that, by the Eq. (11), we can get the solution of SVM.

3.3 Tensor learning machine

If we already known the N training samples $\boldsymbol{X}_i \in R^{L_1\times L_2\times\cdots\times L_M}$ ($1\leqslant i\leqslant N$) (they are M order tensors) with their corresponding labels $y_i\in\{+1,-1\}$, we must found a optimal tensor hyperplane to maximizes the margin between the positive samples and the negative samples. The tensor hyperplane (Fig.3) can be described as $y(\boldsymbol{X})=\boldsymbol{X}\prod_{k=1}^{M}\times_k \boldsymbol{\omega}_k+b$.

Tensor $\boldsymbol{X}_i \in R^{L_1\times L_2\times L_3}$ do the mode-d product with the projection vectors $\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$, $\boldsymbol{\omega}_3$ orderly at mode 1,2,3, and the result is a scalar. Then we can distinguish the positive samples from the negative samples.

To determine the tensor hyperplane, we can use this optimization to get the parameters $\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$, $\boldsymbol{\omega}_3$ and b (Tao *et al*., 2007):

$$\left[\begin{array}{ll} \min\limits_{\boldsymbol{\omega}_k|_{k=1}^{M},b,\boldsymbol{\xi}} & \dfrac{1}{2}\left\|\bigotimes_{k=1}^{M}\boldsymbol{\omega}_k\right\|^2+c\sum_{i=1}^{N}\xi_i \\ \text{s.t.} & y_i\left[\boldsymbol{X}_i\prod_{k=1}^{M}\times_k\boldsymbol{\omega}_k+b\right]\geqslant 1-\xi_i, 1\leqslant i\leqslant N \\ & \boldsymbol{\xi}\geqslant 0 \end{array}\right] \tag{14}$$

$\boldsymbol{\xi}\in R^N$ is slack variable to deal with the linearly nonseparable problem. We can set $\boldsymbol{\xi}$=0 if the samples are linearly separable.

The Lagrangian function for Eq. (14) is:

$$\begin{aligned} & L\left(\boldsymbol{\omega}_k|_{k=1}^{M},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}\right) \\ =& \frac{1}{2}\left\|\bigotimes_{k=1}^{M}\boldsymbol{\omega}_k\right\|^2+c\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\kappa_i\xi_i \\ & -\sum_{i=1}^{N}\alpha_i\left(y_i\left[\boldsymbol{X}_i\prod_{k=1}^{M}\times_k\boldsymbol{\omega}_k+b\right]-1+\xi_i\right) \\ =& \frac{1}{2}\prod_{k=1}^{M}\boldsymbol{\omega}_k^{\mathrm{T}}\boldsymbol{\omega}_k+c\sum_{i=1}^{N}\xi_i-b\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}+\sum_{i=1}^{N}\alpha_i \\ & -\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\xi}-\boldsymbol{\kappa}^{\mathrm{T}}\boldsymbol{\xi}-\sum_{i=1}^{N}\alpha_i y_i\left(\boldsymbol{X}_i\prod_{k=1}^{M}\times_k\boldsymbol{\omega}_k\right) \end{aligned} \tag{15}$$

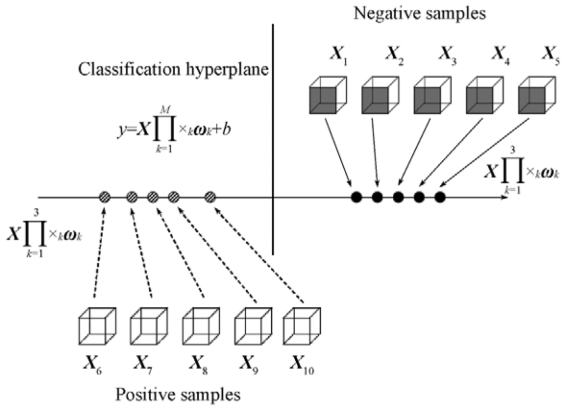

Negative samples
Classification hyperplane
$y=X\prod_{k=1}^{M}\times_k\omega_k+b$
X_1
X_2
X_3
X_4
X_5
$X\prod_{k=1}^{3}\times_k\omega_k$
$X\prod_{k=1}^{3}\times_k\omega_k$
X_6
X_7
X_8
X_9
X_{10}
Positive samples

Fig. 3 Tensor hyperplane for 3-order tensor classification

In which α_i and κ_i, ($1\leqslant i\leqslant N$) are Lagrangian multipliers. We can get the partial derivative of function L by Eq. (15):

$$\begin{aligned} & \frac{\partial L}{\partial \boldsymbol{\omega}}=0 \Rightarrow \\ & \boldsymbol{\omega}_j=\frac{1}{\prod\limits_{k=1}^{k\neq j}\boldsymbol{\omega}_k^{\mathrm{T}}\boldsymbol{\omega}_k}\cdot\sum_{i=1}^{N}\alpha_i y_i\left(\boldsymbol{X}_i\prod_{k=1}^{M}\bar{\times}_j\boldsymbol{\omega}_j\right) \\ & \frac{\partial L}{\partial b}=0 \Rightarrow \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}=0 \\ & \frac{\partial L}{\partial \boldsymbol{\xi}}=0 \Rightarrow \boldsymbol{c}-\boldsymbol{\alpha}-\boldsymbol{\kappa}=0 \end{aligned} \tag{16}$$

The dual problem for Eq. (15) is:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\kappa}}\ \min_{\boldsymbol{\omega}_k|_{k=1}^{M},b,\boldsymbol{\xi}} L\left(\boldsymbol{\omega}_k|_{k=1}^{M},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}\right) \tag{17}$$

It is a LP with the optimization variable $\boldsymbol{\alpha}$.

By the first equation of Eq. (16), the solution of $\boldsymbol{\omega}_j$ depends on $\boldsymbol{\alpha}$ and $\boldsymbol{\omega}_k$($1\leqslant k\leqslant M$, $k\neq j$). So we use alternating projection method to get $\boldsymbol{\omega}_j$. Tao (2007) had proved that this alternating projection is converge. The alternating projection procedure is executed as shown in Fig.4.

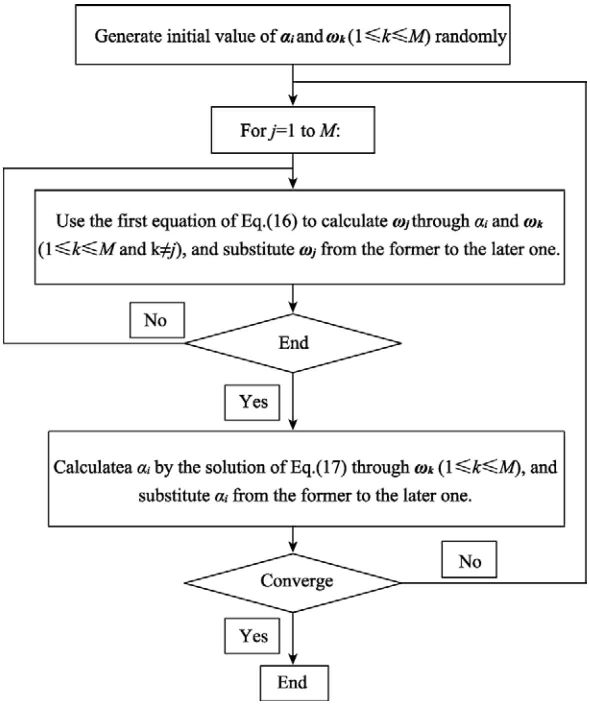

Generate initial value of α_i and ω_k ($1\leqslant k\leqslant M$) randomly
For j=1 to M:
Use the first equation of Eq.(16) to calculate ω_j through α_i and ω_k ($1\leqslant k\leqslant M$ and k≠j), and substitute ω_j from the former to the later one.
No
End
Yes
Calculatea α_i by the solution of Eq.(17) through ω_k ($1\leqslant k\leqslant M$), and substitute α_i from the former to the later one.
No
Converge
Yes
End

Fig. 4 Alternating projection procedure

3.4 Tensor-based learning machine for remotely sensed image target detection

Remotely sensed image target detection could be considered as a two-class classification, which classify the image into the target and background. Use the training samples of target and background, we can get the optimization hyperplane for classification by TLM. Take the hyperspectral and high-resolution remotely sensed images as examples for target detection, the procedures are as follows:

(1) Choose the targets and backgrounds as training samples in the preprocessed image, and convert them to 3 order-tensors as inputs of TLM use the method describe in section 2.1;

(2) Classify the training samples and get the tensor hyperplane using TLM;

(3) Select the window of the same size as the training samples from the first pixel in the image, take it as the start position of target detection;

(4) Convert the data in the window to 3 order-tensor, classify it use tensor hyperplane. Label it if it's classified as target, or ignore it if it's classified as background;

(5) Shift the window when the last window has been processed. Repeat the action in step 4;

(6) Finish the TLM when the last window in the image has been processed;

(7) Post-process the result of target detection. Merge the labels where they corresponding to the same target.

Fig.5 is the flow chart of tensor-based learning machine for remotely sensed image target detection:

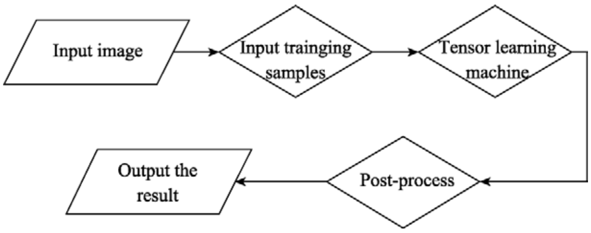


Fig. 5 Flow chart of tensor-based learning machine for remotely sensed image target detection

4 EXPERIMENTS

4.1 Hyperspectral image experiment

In this experiment, the Nuance's ground truth data is selected for target detection. The Nuance imaging spectrometer could obtain the images of the wavelength from 650nm to 1100nm, and the spectral resolution is 10nm. The targets in the image are ten stones while the other objects in the background including the soil, green grass, and withered grass, the spectral curves of above objects are showed in Fig. 6. From Fig.6 we know that the spectral information of target and withered grass are quite closed (curve A and B).

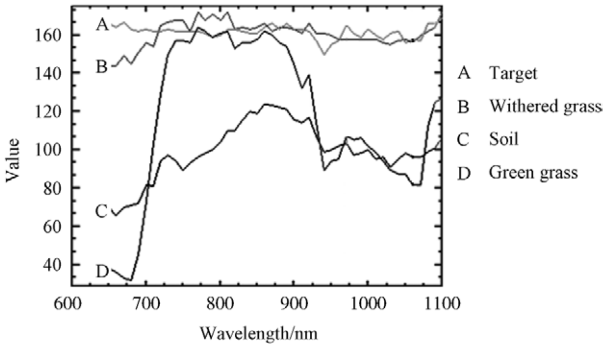


Fig. 6 Spectral curves of the main objects in the hyperspectral remote sensing image target detection experiment

The image in this experiment is 362 pixels width, 514 pixels height. Consider the redundancy of hyperspectral data, we choose 10 bands out of all 46 bands and the window size is set to 13 pixels × 13 pixels. The 5 positive samples and 5 negative samples are showed in Fig. 7. The tensor learning machine for this experiment is executed in MatLab R2007a, while the contrast experiment of SVM is executed in ENVI 4.5.

The target detection result of the proposed method and SVM are showed in Fig. 8. From Fig. 9 we can see TLM detect the all the ten targets in the right place, and the targets with different sizes could also be correctly detected. Because of the complex environment in background, both the TLM and SVM make incorrectly detections, from Fig. 8(a) and (b) we can see that only one target is detected incorrectly in TLM, while a large amount of pixels are misclassified into targets in SVM.

In order to quantitative evaluate the result of the proposed method, successful rate and false alarm rate are introduced, defines by successful rate = the number of targets which had been correctly detected / the total number of targets exist in the image and false alarm rate = the number of targets which had been incorrectly detected / the total number of targets which had been detected. The quantitative result of hyperspectral remote sensing image target detection is in Table 1.

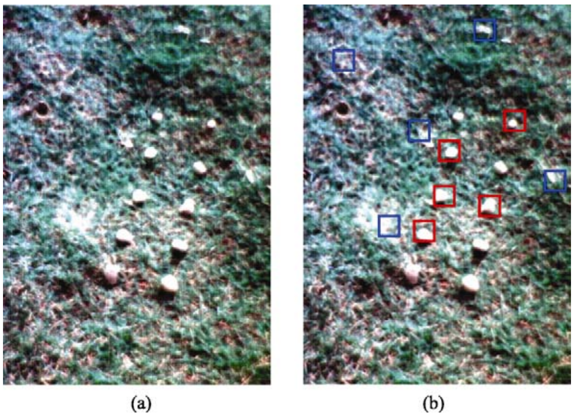


Fig. 7 Experiment data and the training samples
(a) Original image; (b) 10 Training samples in the image

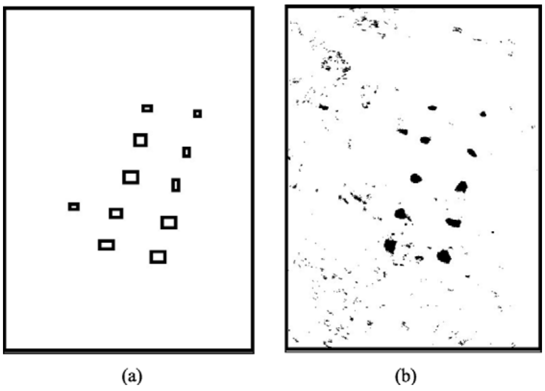


Fig. 8 Result of hyperspectral image target detection
(a) Proposed method; (b) SVM

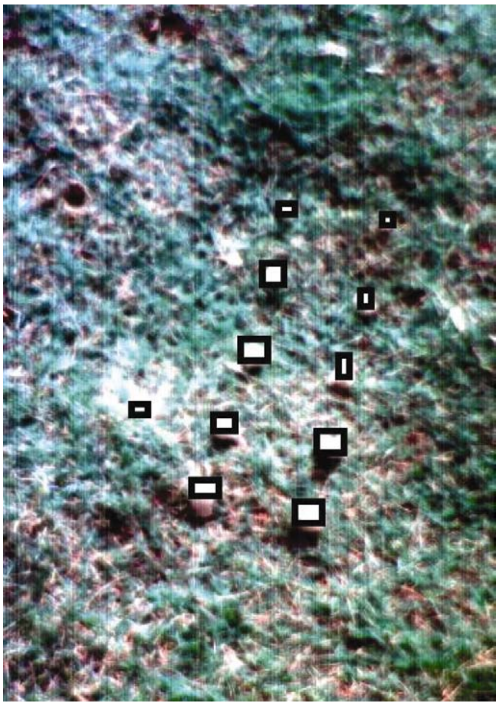


Fig. 9 Locations of the targets which have been detected in the image

Table 1 Quantitative result of hyperspectral remote sensing image target detection

/%

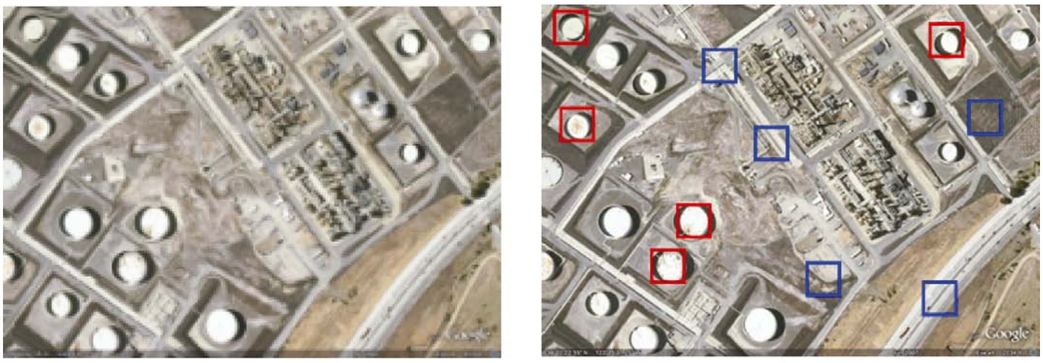
	TLM	SVM
Successful rate	100	100
False alarm rate	9.1	52.9

4.2 High resolution image experiment

In this experiment, the image is selected from GoogleEarth at west California Union Oil Company(38°02′22.55″N, 122°15′01.25″W), which has the size of 1024 pixels width, 724 pixels height and 3 bands. The targets in the image are 15 oilcans and the other objects in the image including soil, roads and roofs. The roads have the similar spectral information with the targets so they are difficult to distinguish. The 5 positive samples and 5 negative samples are selected for training samples with the size of 30 pixels × 30 pixels × 3 bands according to the size of the targets. The location of the training samples in the image is showed in Fig. 10.

In the high resolution remote sensing image target detection, the high resolution helps to observe more texture and small targets while cause the high intra-class and low inter-class variances (Huang, 2009). At this time, the pixels that have the similar spectral information with the target would be misclassified as target if we still take vectors as inputs and classify the all pixels by SVM. However, if we take tensor as inputs, each pixel would be constructed to 3-order tensor with the neighbor pixels, which consider spectral and structure information, so the probability of misclassify this tensor is decreased, that is the reason why the proposed method could keep a high probability of successful detection and reduce the false alarm.

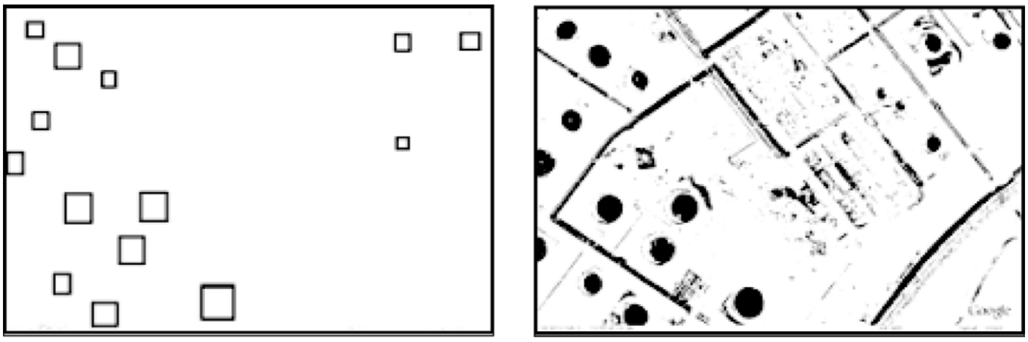
The target detection result of the proposed method and SVM are showed in Fig. 11. From Fig.12 we can see 14 targets with different sizes are correctly detected in right place without



(a) (b)

Fig. 10 Experiment data and the training samples

(a) Original image; (b) 10 training samples in the image



(a) (b)

Fig. 11 Result of high resolution image target detection

(a) Proposed method; (b) SVM

any false alarm, while large amount of pixels in the road are misclassified to target in SVM, because they have similar spectral information.

The quantitative result of high resolution remote sensing image target detection is in Table 2.

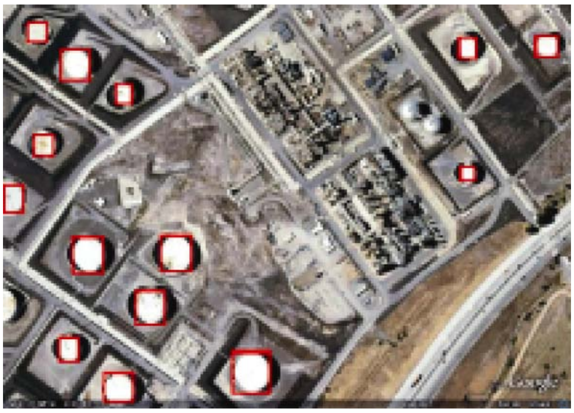


Fig. 12 Locations of the targets which have been detected in the image

Table 2 Quantitative result of high resolution remote sensing image target detection

/%

	TLM	SVM
Successful rate	93.3	100
False alarm rate	0	65.1

5 CONCLUSION

In this article a new way to detect the targets in remote sensing image based on the tensor learning machine (TLM) is proposed. The vector-based learning machine is generalized to the tensor-based learning machine which accepts tensors as input, alternating projection procedure is used to get the solution of the TLM. A great deal of experiments demonstrate that TLM could achieve 90% successful rate and keep the false alarm rate lower than 10%. Compare with the vector-based learning machine, the tensor learning machine for target detection could get high probability of successful detection and reduce the false alarm.

REFERENCES

Boyd S, Vandenberghe L. 2004. Convex Optimization. Cambridge: Cambridge University Press

Burges J C. 1998. A tutorial on support vector machines for pattern recognition. *Data Min Knowl Discov*, **2**(2): 121—167

Huang X. 2009. Multiscale Texture and Shape Feature Extraction and Object-Oriented Classification for Very High Resolution Remotely Sensed Imagery. Wuhan: Wuhan University.

Lathauwer L D. 1997. Signal Processing Based on Mutilinear Algebra. Belgium: Katholieke Universiteit Leuven

Li Y Z, Luo D Y and Liu S Q. 2008. Face recognition using tensor locality discriminant projection. *Acta Electronica Sinica*, **6**(10): 2070—2075

Pedroso J P and Murata N. 1999. Support vector machines for linear programming: motivation and formulations. BSIS Technical Report, 99-2

Ro D D, Hart P E and Stork D G. 2001. Pattern classification. New York: Wiley

Strohmann T R, Belitski A, Grudic G Z and DeCoste D. 2003. Sparse greedy minimax probability machine classification. 17th Annual Conference on Neural Information Processing Systems. Canada: Advances in Neural Information Processing Systems

Tan K and Du P J. 2008. Hyperspectral remote sensing image classification based on support vector machine. *J. Infrared Millim. Waves*, **27**(2): 321—821

Tao D C, Li X L, Wu X D, Hu W M and Stephen J.Maybank. 2007. Supervised tensor learning. *Knowledge and Information Systems* (Springer: KAIS), **13**(1): 1—42

Vapnik V. 1995. The Nature of Statistical Learning. New York: Springer-Verlag

Wi Z W. 1969. Nonlinear Programming: a Unified Approach. London: Prentice-Hall

Winston W L, Goldberg J B and Venkataramanan M. 2002. Introduction to Mathematical Programming: Operations Research. North Carolina: Baker & Taylor Books

张量分类算法的遥感影像目标探测

张乐飞[1]，张良培[1]，陶大程[2]

1. 武汉大学 测绘遥感信息工程国家重点实验室，湖北 武汉 430079;
2. 南洋理工大学 计算机工程系，新加坡 639798

摘 要： 提出了一种基于张量学习机的遥感影像目标探测方法。该方法基于张量数据模型和张量代数运算，针对遥感影像数据多维或高维的特点，将基于向量的监督法学习机扩展为基于张量的监督法学习机，然后利用凸函数最优化理论和交互投影迭代法求得张量学习机的最优解。最后分别以高光谱遥感影像和高分辨率遥感影像为例，使用张量学习机进行目标探测。实验表明，与支持向量机等方法相比，本文的方法在保持较高探测成功率的同时更好的抑制了虚警。

关键词： 张量，最优化理论，监督学习，目标探测

中图分类号： TP751.1 **文献标识码：** A

引用格式：张乐飞，张良培，陶大程. 2010. 张量分类算法的遥感影像目标探测. 遥感学报, **14**(3): 519—533

Zhang L F, Zhang L P and Tao D C. 2010. Tensor-based learning machine for remotely sensed image target detection. *Journal of Remote Sensing.* **14**(3): 519—533

1 引 言

目标探测是遥感影像处理中的一个热点问题。支持向量机 (SVM) 方法 (Vapnik, 1995) 在遥感影像目标探测中应用十分广泛。作为一种基于向量的学习机，其原理是通过对目标与背景这两类训练样本的光谱以向量的形式进行学习，找到最合适的分类依据，进而对影像的每个像元是否为目标进行判断。SVM 方法充分利用了目标和背景的光谱信息。随着遥感影像空间分辨率、光谱分辨率和时间分辨率的提高，在遥感影像中很多目标并不能合适的用一维向量表示，而是需要用二维或更高维的数组描述。此时，支持向量机方法的目标探测效率降低，主要表现为虚警率大大增高。针对高光谱、高分辨率遥感影像的目标探测，必须引入一种能够描述多维数组的数据模型。

张量 (Tao 等，2007)，作为描述多维数组的数据模型，能够更好的描述遥感影像中由多个像元组成的目标或背景地物。张量学习是机器学习和数据挖掘领域的一个较新的研究方向。现有比较成熟的监督法学习机例如支持向量机 (SVM) (谭琨 & 杜培军, 2008; Burges, 1998) 、最小最大概率机 (MPM) (Strohmann 等，2003) 和 Fisher 判别式分析 (FDA) (Duda 等，2001) 等，只是利用了一维的向量作为监督学习的训练样本。然而，现实中的数据从本质上是以张量这种多维数组的形式存在的 (李勇周等，2008)，仅用向量描述现实中的数据，必然会丢失数据的结构信息 (Tao 等，2007) 。因此必须引入张量作为监督学习的训练样本，实现以张量作为训练样本的学习机——张量学习机(tensor learning machine, TLM)。本文介绍了张量学习机的原理，并针对张量学习机在高光谱、高分辨率遥感影像中的目标探测问题进行了详细讨论。

2 张量与张量代数

2.1 张量

张量 $X \in R^{L_1 \times L_2 \times \cdots \times L_M}$ 表示一个多维数组(Tao 等，2007)。其中 M 表示张量的阶数，即多维数组的维数；L_i 表示张量第 i 阶的大小。张量中的某个元素表示为

收稿日期：2009-04-27；修订日期：2009-09-02

基金项目：国家 973 计划资助项目(编号 2009CB723905)，国家自然科学基金资助项目(编号 40930532，40771139)，国家 863 计划资助项目(编号 2009AA12Z114)。

第一作者简介：张乐飞(1986—)，男，硕士研究生，2008 年毕业于武汉大学遥感信息工程学院，目前主要从事遥感影像处理方面的研究工作。E-mail: zhanglefei.wh@gmail.com。

$\boldsymbol{X}_{l_1,l_2,\cdots,l_M}$ $(1\leqslant l_i\leqslant L_i$ 且 $1\leqslant i\leqslant M)$，其中 l_i 表示该元素在数组中第 i 阶的位置。

在特殊的低阶张量中，零阶的张量 $\boldsymbol{X}\in R$ 即是实数；一阶的张量 $\boldsymbol{X}\in R^{L_1}$ 即是向量；二阶的张量 $\boldsymbol{X}\in R^{L_1\times L_2}$ 表示矩阵，其中 $\boldsymbol{X}_{l_1,l_2}$ $(1\leqslant l_i\leqslant L_i)$表示矩阵 $\boldsymbol{X}\in R^{L_1\times L_2}$ 第 l_1 行、第 l_2 列的元素。常见的高分辨率、高光谱遥感影像，可以描述为三阶张量 $\boldsymbol{X}\in R^{L_1\times L_2\times L_3}$，其中 L_1 和 L_2 分别表示单波段影像的高度和宽度，L_3 表示影像的波段数，元素 $\boldsymbol{X}_{l_1,l_2,l_3}$ 表示影像中第 l_3 波段、第 l_1 行、l_2 列的像元灰度值。如图 1。

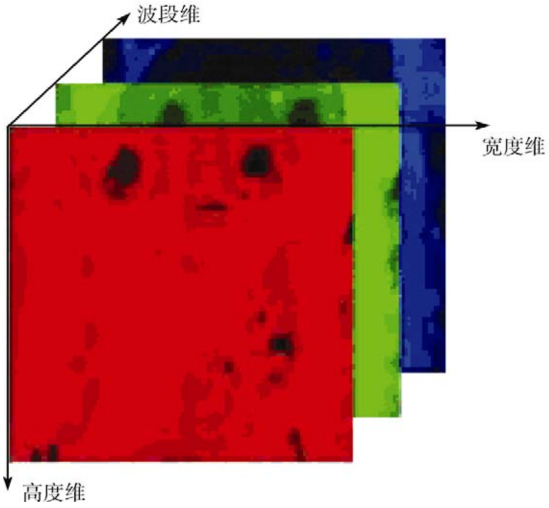


图 1　用三阶张量描述遥感影像
(高度、宽度和波段数分别为该张量的 3 个维度)

2.2　张量代数

与张量有关的常见代数运算 (Lathauwer，1997)定义如下：

2.2.1　外积

张量外积(outer product)的定义是：若 $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$，$\boldsymbol{Y}\in R^{L_1'\times L_2'\times\cdots\times L_M'}$，则外积：

$$(\boldsymbol{X}\otimes\boldsymbol{Y})_{l_1,l_2,\cdots,l_M,l'_1,l'_2,\cdots,l'_{M'}}=\boldsymbol{X}_{l_1,l_2,\cdots,l_M}\boldsymbol{Y}_{l'_1,l'_2,\cdots,l'_{M'}}\tag{1}$$

2.2.2　缩并

张量缩并(tensor contraction)的定义是：若 $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M\times L'_1\times L'_2\times\cdots\times L'_{M'}}$，$\boldsymbol{Y}\in R^{L_1\times L_2\times\cdots\times L_M\times L''_1\times L''_2\times\cdots\times L''_{M''}}$，则：

$$\begin{aligned}&\left[\boldsymbol{X}\otimes\boldsymbol{Y};(1:M)(1:M)\right]\\&=\sum_{l_1=1}^{L_1}\cdots\sum_{l_M=1}^{L_M}(\boldsymbol{X})_{l_1,l_2,\cdots,l_M,l'_1,l'_2,\cdots,l'_{M'}}(\boldsymbol{Y})_{l_1,l_2,\cdots,l_M,l''_1,l''_2,\cdots,l''_{M''}}\end{aligned}\tag{2}$$

张量缩并的条件是张量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 在某一阶上大小相等，即在某一维度上具有相等的元素数。每发生一次张量缩并，将会使阶数减少 2。在式(2)中，$M+M'$阶张量 $\boldsymbol{X}$ 与 $M+M''$阶张量 $\boldsymbol{Y}$ 在前 M 阶的大小都相等，因此进行 M 次张量缩并运算以后，得到 $M'+M''$阶张量。

2.2.3　*d* 维度与矩阵相乘

d 维度与矩阵相乘(mode-d product，简写为 ${}_d\boldsymbol{U}$)是一种特殊形式的张量缩并运算，必须满足张量 $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ 与矩阵 $\boldsymbol{U}\in R^{L'_d\times L_d}$ 在 d 阶上大小相等，即张量 $\boldsymbol{X}$ 在第 d 阶的大小为 L_d。定义：

$$\begin{aligned}&(\boldsymbol{X}\times{}_d\boldsymbol{U})_{l_1,l_2,\cdots,l_{d-1},l'_d,l_{d+1},\cdots,l_M}\\&=\sum_{l'_d}(\boldsymbol{X}_{l_1,l_2,\cdots,l_{d-1},l_d,l_{d+1},\cdots,l_M}\boldsymbol{U}_{l'_d,l_d})\end{aligned}\tag{3}$$

该运算的实质是一个 M 阶张量与一个二阶张量进行了一次缩并运算。

特殊的，如果是 M 阶张量与向量相乘，根据缩并的定义，将会得到$(M-1)$阶张量；如果该 M 阶张量依次与 M 个满足条件的向量连续相乘，将会得到零阶张量，即实数。这种运算在下文中求解分类张量平面的过程中多次被运用。

2.2.4　有理标准型

张量 $\boldsymbol{X}\in R^{L_1\times L_2\times\cdots\times L_M}$ 的有理标准型(frobenius norm)的定义是：

$$\begin{aligned}\|\boldsymbol{X}\|_{\mathrm{Fro}}&=\sqrt{\left[\boldsymbol{X}\otimes\boldsymbol{X};(1:M)(1:M)\right]}\\&=\sqrt{\sum_{l_1=1}^{L_1}\cdots\sum_{l_M=1}^{L_M}\boldsymbol{X}^2_{l_1,\cdots,l_M}}\end{aligned}\tag{4}$$

有理标准型描述了张量的大小，它的平方即是该张量全部元素的平方和，表示了这个张量的能量。

3　张量学习机

3.1　凸函数最优化理论

机器学习的数学模型总是可以转化为最优化问题 (Winston 等, 2002) 。因此，凸函数最优化理论是求解张量学习机的核心问题。在机器学习领域，凸函数最优化理论已经被成功的应用。例如，线性规划用于线性规划机(linear programming machine, LPM)以及二次规划用于支持向量机(support vector machine, SVM)。

最优化问题的数学模型可以描述为：

$$\begin{bmatrix}\min & f(\boldsymbol{X}) & \boldsymbol{X}\in R^L\\ \text{s.t.} & g_i(\boldsymbol{X}) & i=1,2,\cdots,m\end{bmatrix}\tag{5}$$

其中 m 个限制条件 $g_i(\boldsymbol{X})$共同限定了变量的取值范围，即可行域；求解该最优化问题即在可行域的范围内找到目标函数 $f(\boldsymbol{X})$的最小值。

最简单的最优化问题是线性规划问题(linear

programming, *LP*)，即限制条件 $g_i(\boldsymbol{X})$ 与目标函数 $f(\boldsymbol{X})$ 均为线性函数，该问题的数学模型为：

$$\begin{bmatrix} \min\limits_{\boldsymbol{X}} & f(\boldsymbol{X})=\boldsymbol{C}^{\mathrm{T}}\boldsymbol{X} \\ \text{s.t.} & \boldsymbol{A}\cdot\boldsymbol{X}\leqslant\boldsymbol{B} \\ & \boldsymbol{Ae}\cdot\boldsymbol{X}=\boldsymbol{Be} \\ & \boldsymbol{lb}\leqslant\boldsymbol{X}\leqslant\boldsymbol{ub} \end{bmatrix} \tag{6}$$

其中 $\boldsymbol{X}=[x_1,x_2,\cdots,x_n]^{\mathrm{T}}$; $\boldsymbol{C}$, $\boldsymbol{B}$, $\boldsymbol{Be}$, $\boldsymbol{lb}$, $\boldsymbol{ub}$ 是向量; $\boldsymbol{A}$, $\boldsymbol{Ae}$ 是矩阵。

若线性规划有最优解，则必在可行域的顶点处取得(Boyd & Vandenberghe, 2004) 。基于这个结论，只需逐个计算可行域的顶点处对应的函数值 $f(\boldsymbol{X})$，进行比较即可得到 LP 问题的最优解。

当约束条件为线性，而目标函数为二次型或二次函数时，该最优化问题成为二次规划(Boyd & Vandenberghe, 2004) (Quadratic programming, QP)。二次规划是研究的比较成熟的非线性规划问题，它的数学模型可以表示为：

$$\begin{bmatrix} \min\limits_{\boldsymbol{X}} & f(\boldsymbol{X})=\dfrac{1}{2}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{QX}+\boldsymbol{C}^{\mathrm{T}}\boldsymbol{X} \\ \text{s.t.} & \boldsymbol{A}\cdot\boldsymbol{X}\leqslant\boldsymbol{B} \\ & \boldsymbol{Ae}\cdot\boldsymbol{X}\leqslant\boldsymbol{Be} \\ & \boldsymbol{lb}\leqslant\boldsymbol{X}\leqslant\boldsymbol{ub} \end{bmatrix} \tag{7}$$

其中 $\boldsymbol{X}=[x_1,x_2,\cdots,x_n]^{\mathrm{T}}$; $\boldsymbol{C}$, $\boldsymbol{B}$, $\boldsymbol{Be}$, $\boldsymbol{lb}$, $\boldsymbol{ub}$ 是向量; $\boldsymbol{Q}$, $\boldsymbol{A}$, $\boldsymbol{Ae}$ 是矩阵。QL 问题的最优解可以使用拉格朗日方法和 K-T 定理联合求出。

3.2 支持向量机方法

支持向量机(support vector machine, SVM) (Vapnik, 1995; Pedroso & Murata, 1999)，是一种基于结构风险最小化原理和最优化理论的机器学习方法。该方法通过一系列已知类别的样本，利用某种准则找到一个最优的分类依据，从而实现对未知类别的样本的分类。具体方法如下：

假设 N 个训练样本：$\boldsymbol{x_i}\in R^L(1\leqslant i\leqslant N)$，并且已知它们所属的类别 $y_i\in\{+1, -1\}$，如图 2，要在特征空间中找到这样一个超平面：它能使得正样本与负样本之间的距离最大化。

解决这个问题是利用如下的最优化方法，求出投影向量 $\boldsymbol{\omega}\in R^L$ 和偏移量 $b\in R$ (Burges, 1998)：

$$\begin{bmatrix} \min\limits_{\omega,b,\xi} & \dfrac{1}{2}\|\boldsymbol{\omega}\|^2+c\sum\limits_{i=1}^{N}\xi_i \\ \text{s.t.} & y_i\left[\boldsymbol{\omega}^T\boldsymbol{x}_i+b\right]\geqslant1-\xi_i,\quad 1\leqslant i\leqslant N \\ & \boldsymbol{\xi}\geqslant0 \end{bmatrix} \tag{8}$$

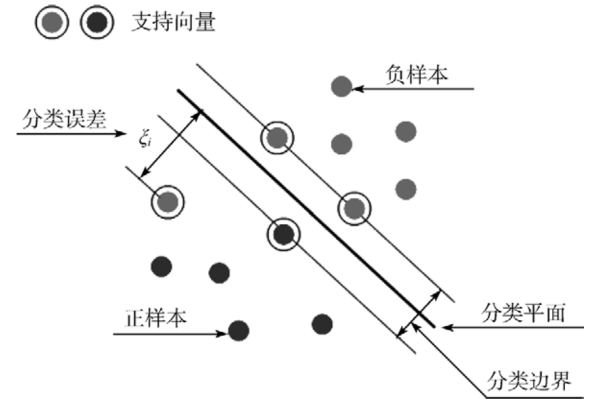


图 2 SVM 分类器使得正样本与负样本之间的距离最大

其中，$\boldsymbol{\xi}=[\xi_1, \xi_2,\cdots, \xi_N]^{\mathrm{T}}\in R^N$ 表示松弛变量，以解决某些样本的线性不可分问题(图 2)。如果训练样本是线性可分的，则令 $\xi_i=0$。

当解出投影向量 $\boldsymbol{\omega}\in R^L$ 和偏移量 $b\in R$ 以后，判决平面可以表示为：

$$y(\boldsymbol{x})=\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x}+b \tag{9}$$

该判决平面可以用于对未知样本进行分类。

最优化问题式(8)的解法如下：首先构造拉格朗日函数：

$$\begin{aligned} &L(\boldsymbol{\omega},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}) \\ &=\frac{1}{2}\|\boldsymbol{\omega}\|^2+c\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\kappa_i\xi_i \\ &\quad-\sum_{i=1}^{N}\alpha_i\left(y_i\left[\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x_i}+b\right]-1+\xi_i\right) \\ &=\frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+c\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x_i} \\ &\quad-b\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}+\sum_{i=1}^{N}\alpha_i-\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\xi}-\boldsymbol{\kappa}^{\mathrm{T}}\boldsymbol{\xi} \end{aligned} \tag{10}$$

式中，α_i 和 κ_i 为拉格朗日乘子$(1\leqslant i\leqslant N)$。由式(10)求出 L 对各变量的偏导数：

$$\begin{aligned} &\frac{\partial L}{\partial\boldsymbol{\omega}}=0\ \Rightarrow\ \boldsymbol{\omega}=\sum_{i=1}^{N}\alpha_i y_i\boldsymbol{x_i} \\ &\frac{\partial L}{\partial b}=0\ \Rightarrow\ \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}=0 \\ &\frac{\partial L}{\partial\boldsymbol{\xi}}=0\ \Rightarrow\ \boldsymbol{c}-\boldsymbol{\alpha}-\boldsymbol{\kappa}=0 \end{aligned} \tag{11}$$

为了先解求拉格朗日乘子 α_i 和 κ_i，将原问题转化为其对偶问题：

$$\max_{\alpha,\kappa}\ \min_{\omega,b,\xi}\ L(\boldsymbol{\omega},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\kappa}) \tag{12}$$

将式(11)各式带入式(12)，消去 $\boldsymbol{\omega},b,\boldsymbol{\xi}$ 可得到如下最优化问题：

$$\begin{bmatrix} \max & -\dfrac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j \boldsymbol{x_i}^{\mathrm{T}} \boldsymbol{x_j} \alpha_i \alpha_j + \sum_{i=1}^{N}\alpha_i \\ \text{s.t.} & \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y} = 0 \\ & 0 \leqslant \boldsymbol{\alpha} \leqslant \boldsymbol{c} \end{bmatrix} \tag{13}$$

若令 $\boldsymbol{Q}=[y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j]_{1\leqslant i,j\leqslant N}$，$C_i=1$，$\boldsymbol{Ae}=\boldsymbol{y}$，$be_i=0$，$lb_i=0$，$\boldsymbol{ub}=C$，则 SVM 的对偶问题式(12)可以视为关于变量 $\boldsymbol{\alpha}=[\alpha_1, \alpha_2,\cdots, \alpha_N]^{\mathrm{T}}$ 的 QP 问题。利用凸函数最优化理论，可以求解出 $\boldsymbol{\alpha}$；然后带入式(11)即可求出 $\boldsymbol{\omega}$。这样，我们就得到了支持向量机(SVM)的解。

3.3 张量学习机方法

已知训练样本：$\boldsymbol{X_i}\in R^{L_1\times L_2\times\cdots\times L_M}$，$(1\leqslant i\leqslant N)$，以及它们对应的所属类别 $y_i\in\{+1, -1\}$。要找到一个最优的分类平面使得两类训练样本能够被最大距离的分开。由于训练样本 $\boldsymbol{X_i}$ 为 M 阶张量，因此分类平面为如下张量平面(图 3)：

$$y(\boldsymbol{X}) = \boldsymbol{X}\prod_{k=1}^{M}\times_k \boldsymbol{\omega_k} + b \text{。}$$

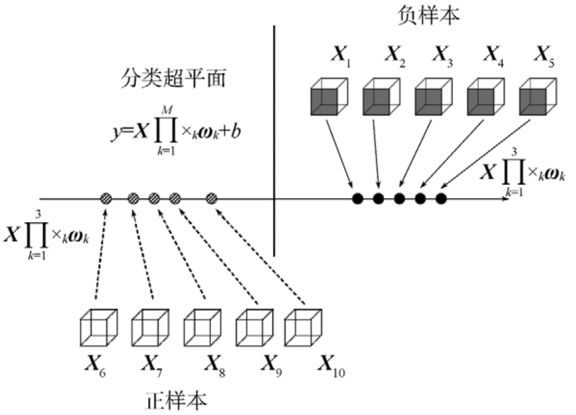

负样本
X_1
X_2
X_3
X_4
X_5
分类超平面
$y=X\prod_{k=1}^{M}\times_k\omega_k+b$
$X\prod_{k=1}^{3}\times_k\omega_k$
$X\prod_{k=1}^{3}\times_k\omega_k$
X_6
X_7
X_8
X_9
X_{10}
正样本

图 3 三阶张量分类平面原理示意图

(样本张量 $X_i\in R^{L_1\times L_2\times\cdots\times L_M}$ 通过与投影向量 ω_1, ω_2, ω_3 依次进行张量缩并运算 $\boldsymbol{X}\prod_{k=1}^{3}\times_k \boldsymbol{\omega_k}$ 得到零阶张量，即实数。然后在一维空间中对正负样本分类)

为确定该张量平面，需要利用如下最优化问题求解投影向量 $\boldsymbol{\omega_k}\in R^{L_k}$ $(1\leqslant k\leqslant M)$和偏移量 $b\in R$ (Tao 等, 2007)：

$$\begin{bmatrix} \min\limits_{\omega_k|_{k=1}^{M},b,\xi} & \dfrac{1}{2}\left\| \bigotimes_{k=1}^{M} \boldsymbol{\omega_k} \right\|^2 + c\sum_{i=1}^{N}\xi_i \\ \text{s.t.} & y_i\left[\boldsymbol{X_i}\prod_{k=1}^{M}\times_k\boldsymbol{\omega_k} + b\right] \geqslant 1-\xi_i, 1\leqslant i\leqslant N \\ & \boldsymbol{\xi}\geqslant 0 \end{bmatrix} \tag{14}$$

其中，$\boldsymbol{\xi}=[\xi_1,\xi_2,\cdots, \xi_N]^{\mathrm{T}}\in R^N$ 表示松弛变量，以解决某些样本的线性不可分问题。如果训练样本本身是线性可分的，则可以令 $\xi_i=0$。

为了解决最优化问题式(14)，构造如下拉格朗日函数：

$$\begin{aligned} & L\left(\boldsymbol{\omega_k}\big|_{k=1}^{M}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\kappa}\right) \\ = & \frac{1}{2}\left\| \bigotimes_{k=1}^{M} \boldsymbol{\omega_k} \right\|^2 + c\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\kappa_i\xi_i \\ & -\sum_{i=1}^{N}\alpha_i\left(y_i\left[\boldsymbol{X_i}\prod_{k=1}^{M}\times_k\boldsymbol{\omega_k} + b\right] - 1 + \xi_i \right) \\ = & \frac{1}{2}\prod_{k=1}^{M}\boldsymbol{\omega_k}^{\mathrm{T}}\boldsymbol{\omega_k} + c\sum_{i=1}^{N}\xi_i - b\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y} + \sum_{i=1}^{N}\alpha_i \\ & -\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\xi} - \boldsymbol{\kappa}^{\mathrm{T}}\boldsymbol{\xi} - \sum_{i=1}^{N}\alpha_i y_i\left(\boldsymbol{X_i}\prod_{k=1}^{M}\times_k\boldsymbol{\omega_k}\right) \end{aligned} \tag{15}$$

式中，α_i 和 κ_i 为拉格朗日乘子$(1\leqslant i\leqslant N)$。由式(15)求出 L 对各变量的偏导数：

$$\begin{aligned} & \frac{\partial L}{\partial \boldsymbol{\omega}} = 0 \ \Rightarrow \\ & \boldsymbol{\omega_j} = \frac{1}{\prod\limits_{k=1}^{k\neq j}\boldsymbol{\omega_k}^{\mathrm{T}}\boldsymbol{\omega_k}}\cdot\sum_{i=1}^{N}\alpha_i y_i\left(\boldsymbol{X_i}\prod_{k=1}^{M}\bar{\times}_j\boldsymbol{\omega_j}\right) \end{aligned} \tag{16}$$

$$\frac{\partial L}{\partial b} = 0 \ \Rightarrow \ \boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y} = 0$$

$$\frac{\partial L}{\partial \boldsymbol{\xi}} = 0 \ \Rightarrow \ \boldsymbol{c} - \boldsymbol{\alpha} - \boldsymbol{\kappa} = 0$$

原问题的对偶问题为：

$$\max_{\boldsymbol{\alpha},\boldsymbol{\kappa}} \ \min_{\omega_k|_{k=1}^{M},b,\boldsymbol{\xi}} L\left(\boldsymbol{\omega}_k\big|_{k=1}^{M}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\kappa}\right) \tag{17}$$

该问题是一个线性规划问题。

由(16)的第一个等式可知，求解 $\boldsymbol{\omega_j}$，必须知道 α_i 以及 $\boldsymbol{\omega_k}$ $(1\leqslant k\leqslant M$ 并且 $k\neq j)$。因此，使用交互投影迭代法解求 $\boldsymbol{\omega_j}$。Tao 等(2007)(Supervised Tensor Learning)证明了该方法的收敛性。交互投影迭代法的步骤见图 4。

3.4 基于张量学习机的遥感影像目标探测

遥感影像中的目标探测问题可以视为二分类问题：将图像分为目标或背景两类。利用已知类别的目标和背景样本，使用上文提出的张量学习机方法能够找到最优的分类平面。在下文中，分别对高光谱、高分辨率的遥感影像进行目标探测实验。实验方法如下：

(1) 在经过预处理的遥感影像中，选取合适的目标和背景作为训练样本，使用 2.1 节介绍的方法将训练样本以三阶张量的数据形式读入作为张量学习机的训练样本；

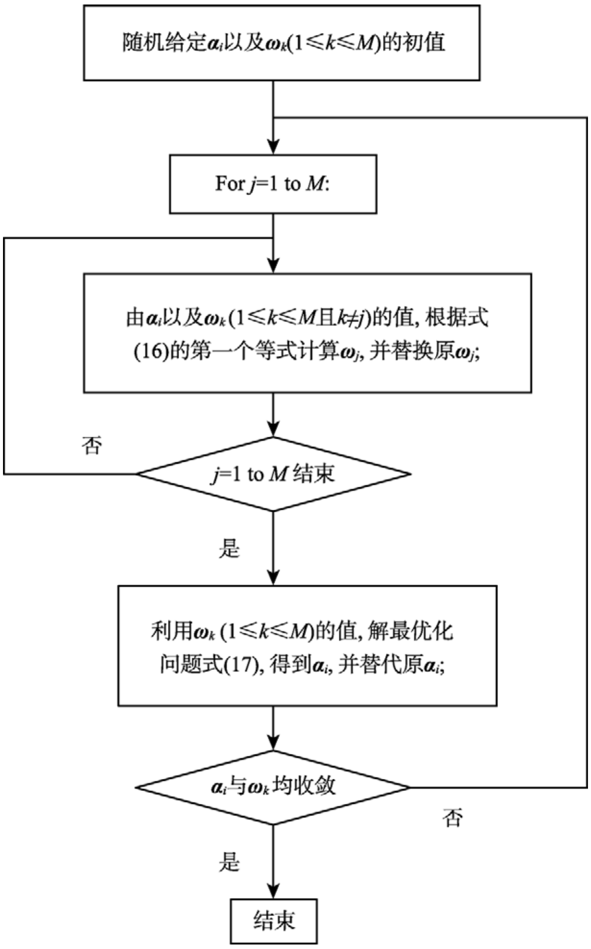

随机给定α_i以及$\omega_k(1\leq k\leq M)$的初值
For j=1 to M:
由α_i以及ω_k $(1\leq k\leq M$且$k\neq j)$的值, 根据式(16)的第一个等式计算ω_j, 并替换原ω_j;
否
j=1 to M 结束
是
利用ω_k $(1\leq k\leq M)$的值, 解最优化问题式(17), 得到α_i, 并替代原α_i;
α_i与ω_k均收敛
否
是
结束

图 4 交互投影迭代法的步骤流程图

(2) 使用张量学习机方法，对已知样本进行分类，得到分类张量平面;

(3) 在遥感影像中，从第一个像元开始，选取与训练样本相同大小的窗口，作为目标探测的起始位置;

(4) 将该窗口内的数据转化为三阶张量的形式，并利用张量分类平面对该张量进行分类。如果是目标，则对其进行标注；如果不是目标，则不进行任何操作;

(5) 前一窗口处理完毕后，将窗口进行移动，得到新的窗口。重复 4 的操作，对新窗口内的数据进行判断;

(6) 当窗口移动到图像的最后一个像元，整幅图像处理完毕;

(7) 对目标检测结果进行处理。对于被重复检测的同一目标，需要将多次标记进行合并。

基于张量学习机的遥感影像目标探测流程图见图 5。

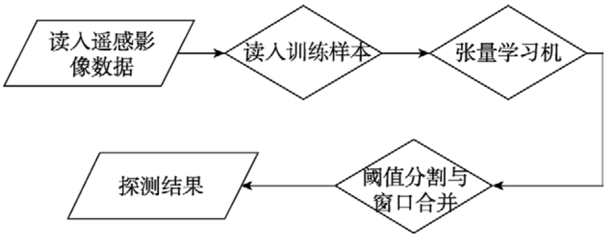

读入遥感影像数据
读入训练样本
张量学习机
阈值分割与窗口合并
探测结果

图 5 基于张量学习机的遥感影像目标探测流程图

4 实验与结果分析

4.1 高光谱遥感影像目标探测

实验数据选自 Nuance 高光谱成像仪采集的地面数据。该光谱成像仪的成像波长范围是 650—1100nm, 光谱分辨率为 10nm。实验中待探测的目标是 10 个大小、形状不一的石块，影像中的背景地物主要有青草、枯草、裸土，各类地物的光谱特征如图 6 所示。其中枯草与目标的光谱特征较为接近(如图 6 中曲线 A、B 所示)，因而是容易被误判的区域。

实验影像大小为 362×514 像元。考虑到高光谱数据的冗余性，实验中从全部影像的 46 波段中选取了信息量较大、相关性较小的 10 个波段。根据待探测目标的大小，实验选取的窗口为 13×13 像元，并将该窗口内 13×13×10(波段)的数据转化为张量。实验选择的 5 个目标张量样本和 5 个背景张量样本影像中的分布如图 7。实验中使用的张量学习机程序在 MatLab R2007a 平台下编写。此外，本文选用了商用软件 EVNI4.5 中的 SVM 方法作为对比实验。

使用本文的方法与支持向量机方法分别对高光谱影像进行目标探测结果如图 8。从图中可以看出，张量学习机方法正确的探测到全部 10 个目标。进一步从图 9 可以看出，全部目标在正确的位置被探测，并且不同尺寸的目标也能被正确探测。由于背景区域的复杂性，局部背景区域的光谱特征与目标的光谱特征相近，因此导致了一定程度的错误探测。从

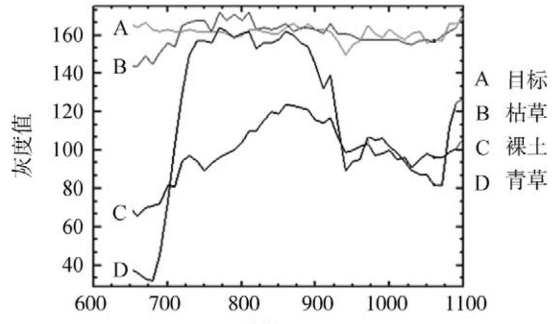


图 6 高光谱遥感影像目标探测实验中主要地物的光谱曲线

图 8(a)中可以看出，本文的方法仅出现了一个被错误探测的目标；而在图 8(b)中，支持向量机方法则出现了大面积较离散的错误探测。

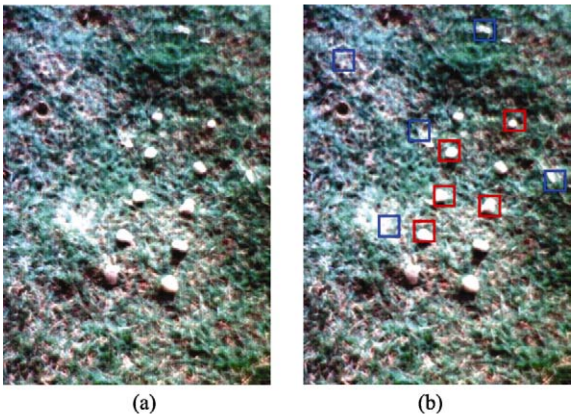


图 7 实验数据与训练样本
(a) 原始图像; (b) 训练样本在原始图像中的位置

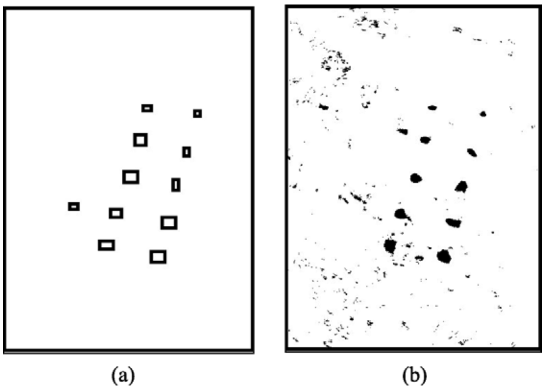


图 8 高光谱遥感影像目标探测实验结果
(a) 本文所采用的方法; (b) 支持向量机方法

为了定量评判本文提出的方法进行目标探测的效率，给出如下定义：探测率=成功探测到的目标数/实际存在的目标总数；虚警率=错误探测的目标数/探测到的目标总数。则高光谱遥感影像目标探测实验结果如表 1。

表 1 高光谱遥感影像目标探测实验结果

/%

	张量学习机	支持向量机
探测率	100	100
虚警率	9.1	52.9

4.2 高分辨率遥感影像目标探测

实验数据来自 GoogleEarth 美国 California 州西部 (38°02′22.55″N, 122°15′01.25″W) 的 Union Oil Company。影像大小为 1024×724×3(波段)，空间分辨率为 2.5m。影像中待探测的目标为油罐，背景主要有土地、道路、屋顶和空地。其中道路的光谱与目标光谱特征较为接近，因而是容易被误判的区域。实验中选择了 5 个包含目标的正样本和 5 个背景作为负样本。根据目标的大小，训练样本确定为 30×30×3(波段)的三阶张量，它们在影像中的位置如图 10。

在高分辨率遥感影像目标探测实验中，由于影像的空间分辨率较高，在观测到更多细节特征和小目标的同时，也造成了同种地物内部的光谱变化和异质性，以及不同地物之间的光谱差异减小 (黄昕, 2009) 。此时，如果仍以向量作为训练样本，使用支持向量机对影像逐像元的进行判断，必然会使得背景中与目标光谱特征相似的像元被错误判断为目标，从而大大的提高错误率。但是如果使用本文提出的张量学习机方法，目标和背景样本全部以三阶张量的形式进行学习，在目标探测过程中也是以张量作为对象。此时，容易被错误判断的背景像元将会与它的邻域像元共同构成张量进行判断。考虑到邻域像元以后，该张量被错误判断的可能性将会大大降低，因此本文的方法能够有效的降低目标探测的错误率。

使用本文的方法与支持向量机方法分别对高光谱遥感影像进行目标探测结果如图 11。从图中可以看出，总共 15 个目标，张量学习机方法正确探测到了其中的 14 个。从图 12 可以看出，上述目标在正确的位置被探测，并且不同尺寸的目标也能被正确探测。而从图 11(b)中可以看出，支持向量机方法出现了大面积的错误探测，主要表现在对道路的大量错误探测，因为道路与目标具有相似的光谱特征。而本文的方法对各种背景地物均没有出现错误探测。

进一步统计实验结果如表 2。

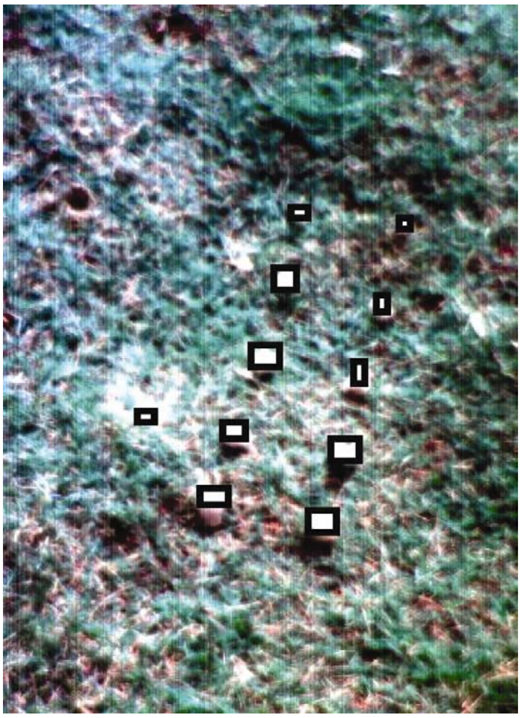


图 9 目标位置示意图

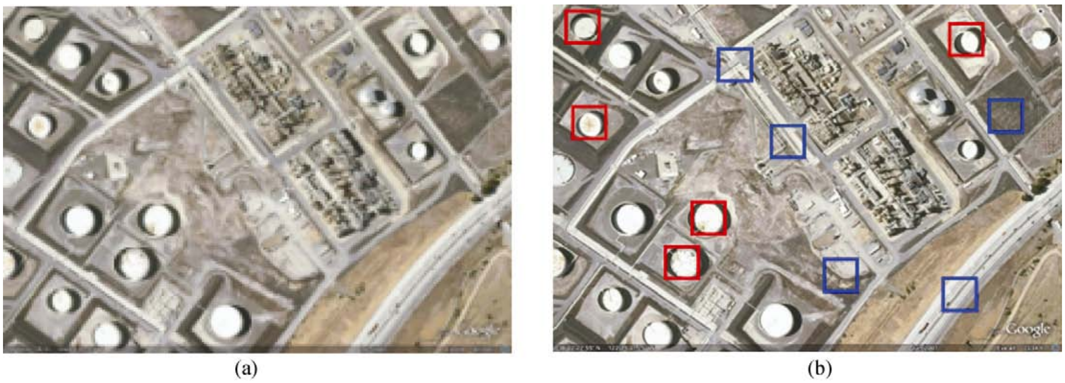


图 10　实验数据与训练样本

(a) 原始影像; (b) 训练样本在原始影像中的位置

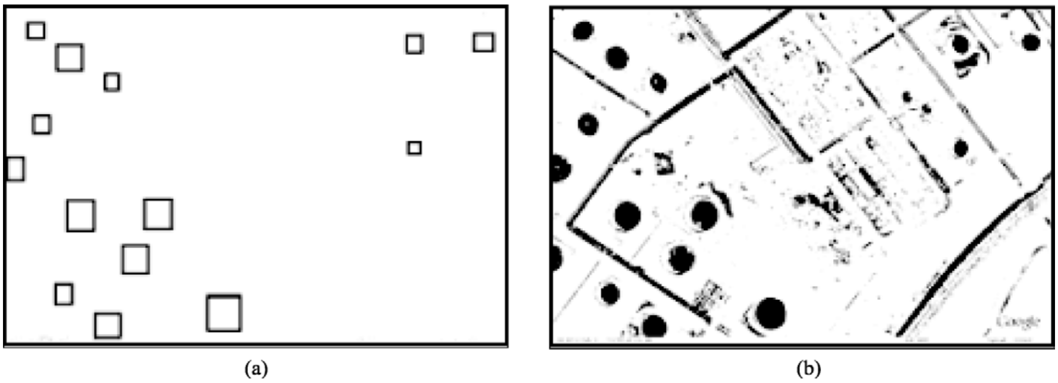


图 11 高分辨率遥感影像目标探测实验结果

(a) 本文所采用的方法; (b) 支持向量机方法

表 2　高分辨率遥感影像目标探测实验结果

/%

	张量学习机	支持向量机
探测率	93.3	100
虚警率	0	65.1

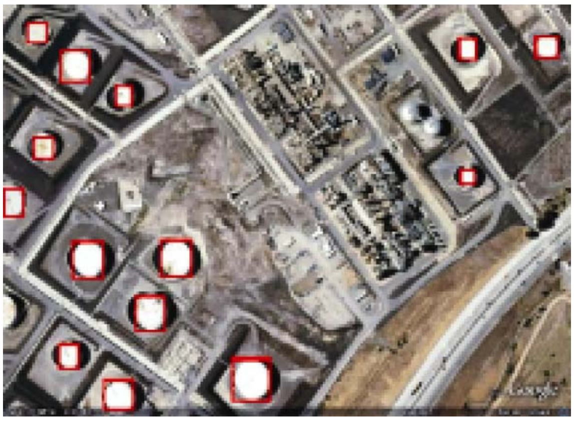


图 12　目标位置示意图

5　结　论

在本文中，提出了一种基于张量学习机的遥感影像目标探测方法。以向量作为输入的监督学习方法 SVM 被扩展成为以张量作为输入的张量学习机(TLM)，通过交互投影迭代法求得张量学习机的解，并将该方法用于遥感影像的目标探测。通过大量的高光谱、高分辨率的遥感影像目标探测实验，结果表明：使用张量学习机方法探测率能够保持在 90%以上，并且保证 10%以内的虚警率。实验证明张量能够更准确的描述遥感影像中被探测的目标，与支持向量机方法相比，张量学习机应用于遥感影像目标探测中能够保持较高的目标探测率，同时能够更有效的抑制虚警。

REFERENCES

Boyd S, Vandenberghe L. 2004. Convex Optimization. Cambridge: Cambridge University Press

Burges J C. 1998. A tutorial on support vector machines for pattern recognition. *Data Min Knowl Discov*, **2**(2): 121—167

Huang X. 2009. Multiscale Texture and Shape Feature Extraction and Object-Oriented Classification for Very High Resolution Remotely Sensed Imagery. Wuhan: Wuhan University

Lathauwer L D. 1997. Signal Processing Based on Mutilinear Algebra. Belgium: Katholieke Universiteit Leuven

Li Y Z, Luo D Y and Liu S Q. 2008. Face recognition using tensor locality discriminant projection. *Acta Electronica Sinica*, **6**(10): 2070—2075

Pedroso J P and Murata N. 1999. Support vector machines for linear programming: motivation and formulations. BSIS Technical Report, 99-2

Ro D D, Hart P E and Stork D G. 2001. Pattern classification. New York: Wiley

Strohmann T R, Belitski A, Grudic G Z and DeCoste D. 2003. Sparse greedy minimax probability machine classification. 17th Annual Conference on Neural Information Processing Systems. Canada: Advances in Neural Information Processing Systems

Tan K and Du P J. 2008. Hyperspectral remote sensing image classification based on support vector machine. *J. Infrared Millim. Waves*, **27**(2): 321—821

Tao D C, Li X L, Wu X D, Hu W M and Stephen J.Maybank. 2007. Supervised tensor learning. *Knowledge and Information Systems* (Springer: KAIS), **13**(1): 1—42

Vapnik V. 1995. The Nature of Statistical Learning. New York: Springer-Verlag

Wi Z W. 1969. Nonlinear Programming: a Unified Approach. London: Prentice-Hall

Winston W L, Goldberg J B and Venkataramanan M. 2002. Introduction to Mathematical Programming: Operations Research. North Carolina: Baker & Taylor Books

附中文参考文献

谭琨，杜培军．2008．基于支持向量机的高光谱遥感图像分类．红外与毫米波学报, **27**(2): 321—821

李勇周，罗大庸，刘少强．2008．张量局部判别投影的人脸识别．电子学报, **6**(10): 2070—2075

黄昕．2009．高分辨率遥感影像多尺度纹理、形状特征提取与面向对象分类方法研究．武汉大学博士学位论文