We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

11

Automatic Target Recognition Based on SAR Images and Two-Stage 2DPCA Features

Liping Hu[1], Hongwei Liu[2] and Hongcheng Yin[1]
[1]*National Key Laboratory of Target and Environment Electromagnetic Scattering and Radiation, Beijing Institute of Environmental Characteristics, Beijing,*
[2]*National Key Laboratory of Radar Signal Processing, Xidian University, Xi'an, China*

1. Introduction

In recent years, radar Automatic Target Recognition (ATR) based on target synthetic aperture radar (SAR) images has received more and more attentions. So far, many literatures based on MSTAR public dataset are released, which focus on the SAR target recognition related techniques including target segmentation, feature extraction, classifier design, and so on. A template matching was proposed (Ross et al., 1998). Support Vector Machine (SVM) has been applied to SAR ATR (Zhao & Principe, 2001; Bryant & Garber, 1999). The drawbacks of them are that none of them have any pre-processing and feature extraction. However, efficient pre-processing and feature extraction may help to improve recognition performance.

Principal Component Analysis (PCA) is a classical feature extraction technique. But when PCA is used for images feature extraction, 2D image matrices must be previously transformed into 1D image vectors. This usually leads to a high dimensional vector space, where it is difficult to evaluate the covariance matrix accurately. To solve this problem, 2-dimensional PCA (2DPCA) for image feature extraction is proposed (Yang et al., 2004). As opposed to PCA, 2DPCA constructs the covariance matrix directly using 2D image matrices rather than 1D vectors, and evaluates the covariance matrix more accurately. Moreover, the size of the covariance matrix is much smaller. A drawback of 2DPCA is that it only eliminates the correlations between rows. So it needs more features, and this will lead to large storage requirements and cost more time in classification phase. To further compress dimension of features, two-stage 2DPCA is applied in this chapter.

The remainder of this chapter is organized as follows: in Section 2, the SAR images pre-processing method is described. 2DPCA is first reviewed, and two-stage 2DPCA is described in Section 3. In Section 4, classifiers are described. In Section 5 and 6, experimental results based on Moving and Stationary Target Acquisition and Recognition (MSTAR) data and conclusions are presented.

2. SAR image pre-processing

The original SAR images provided by MSTAR contain not only targets of our interest, but also background clutters, as shown in Fig. 1 (a). If targets are recognized based the original images, clutters would depress the system performance. Thus, it is necessary to pre-process the original images to segment targets from background clutters.

2.1 Logarithmic transformation

We transform the original images using logarithm conversion, which can convert speckles from multiple model to additional model and make the image histogram more suitable be approximated with a Gaussian distribution. The logarithmic transformation is given by

$$\boldsymbol{G}(x,y) = 10\lg\left[\boldsymbol{F}(x,y) + 0.001\right] + 30 \tag{1}$$

where $\boldsymbol{F}$ denotes the magnitude matrix of the original SAR image. Since the logarithm is not defined at 0, we add an arbitrary constant (for example 0.001) to the original image before the logarithm. To ensure the pixel values to be nonnegative, we add a corresponding constant (30).

2.2 Adaptive threshold segmentation

In order to obtain the target image, the adaptive threshold segmentation method is adopted. First of all, estimating the mean μ and the variance σ of the current image $\boldsymbol{G}$, for each pixel (x,y) of $\boldsymbol{G}$

$$\begin{cases} (x,y) \in \boldsymbol{T}_{\text{ar}}, \boldsymbol{T}_{\text{ar}}(x,y) = 1, & if \quad \boldsymbol{G}(x,y) > \mu + c\sigma \\ (x,y) \in \boldsymbol{B}_{\text{ac}}, \boldsymbol{T}_{\text{ar}}(x,y) = 0, & else \end{cases} \tag{2}$$

Where $\boldsymbol{T}_{\text{ar}}$, $\boldsymbol{B}_{\text{ac}}$ denote the target and the background respectively, c can be obtained statistically from training samples.

2.3 Morphological filter and geometric clustering operation

Due to the presence of speckles, the result of threshold segmentation contains not only target, but smaller objects inevitably, as shown in Fig. 1 (b). To remove these small objects and obtain smoothing the target image, morphological filter (Gonzalez &Woods, 2002) and geometric clustering operation (Musman & Kerr, 1996) are adopted to $\boldsymbol{T}_{\text{ar}}$.

Morphological filter aims to smooth boundary, remove sharp protrusions, fill small concaves, remove small holes, joint gaps, and so on.

In general, filtered image $\boldsymbol{T}_{\text{ar}}$ may also contain some non-target regions, which are much smaller than target itself, as shown in Fig. 1 (c). To remove small regions, we apply geometric clustering operation: firstly, detect and label all the independent connected regions in $\boldsymbol{T}_{\text{ar}}$. Then, compute areas for each region. The largest region is of our interest. In this way, we obtain the resulting $\boldsymbol{T}_{\text{ar}}$, as shown in Fig. 1 (d). Overlaying the resulting $\boldsymbol{T}_{\text{ar}}$ on the logarithmic image $\boldsymbol{G}$ obtains target intensity.

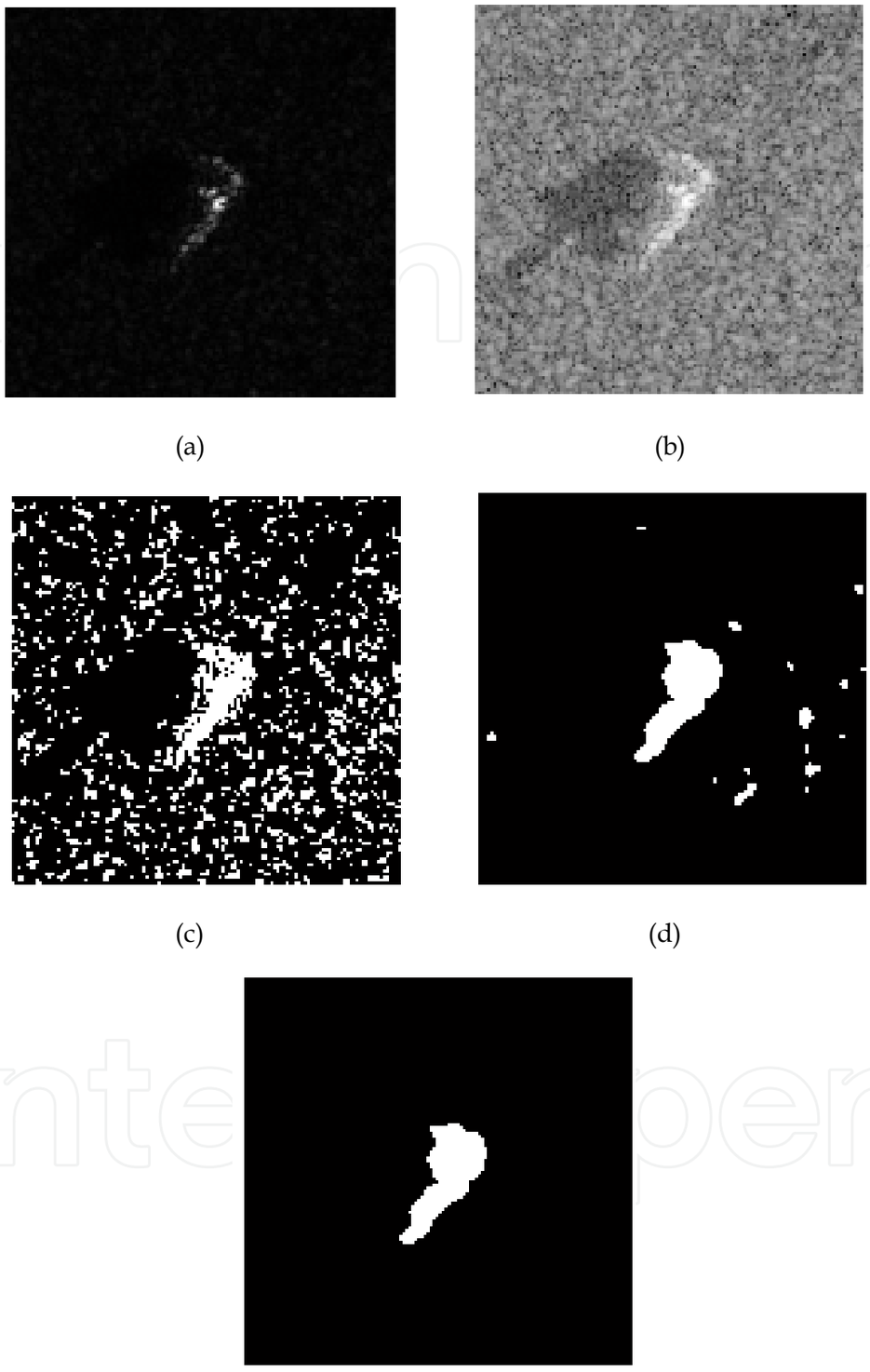
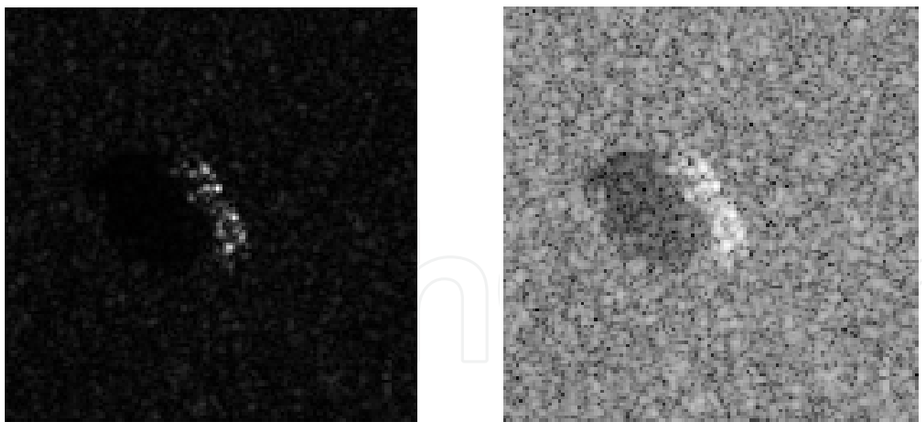
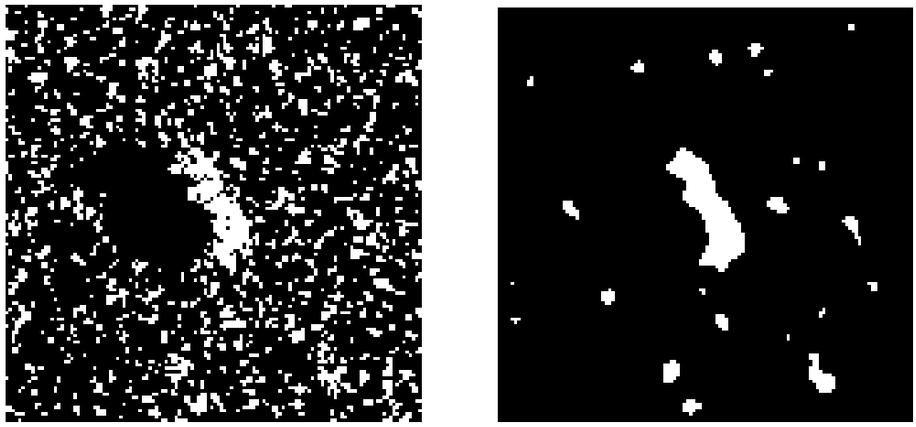


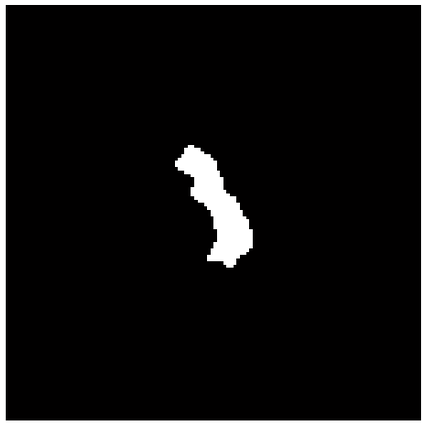
Fig. 1. SAR image pre-processing (taking T72 for example). (a) Original image, (b) Logarithmic image, (c) Threshold segmented image, (d) Result of the morphological filtering, (e) Result of geometric clustering.



(a) (b)

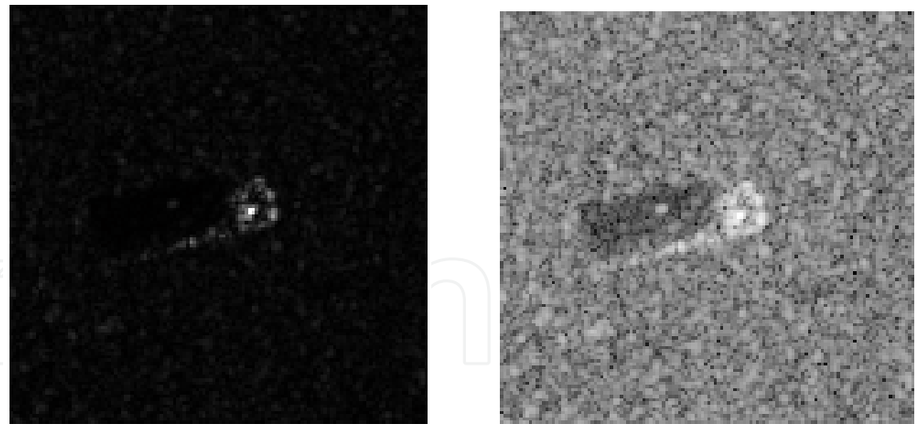


(c) (d)

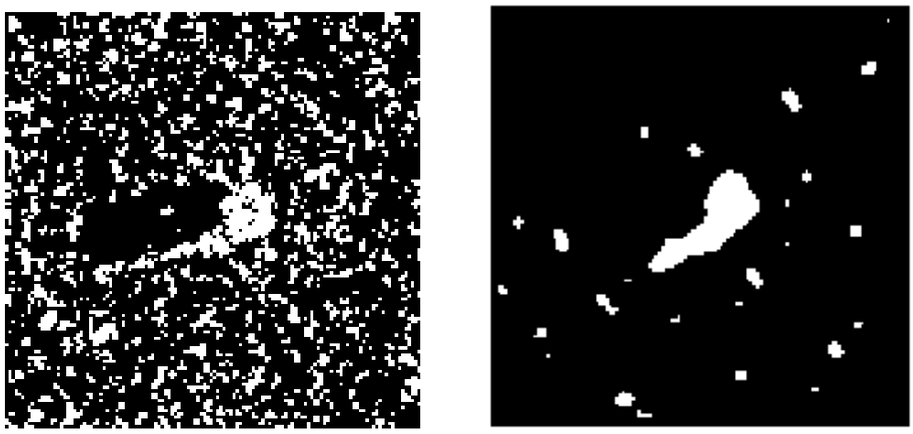


(e)

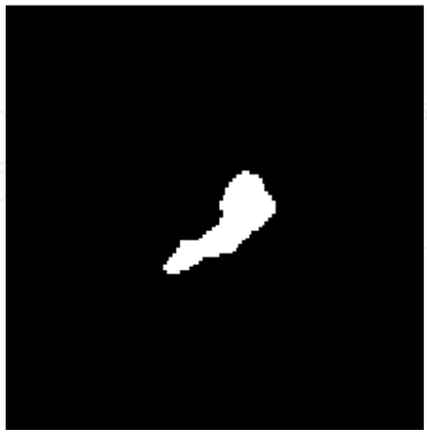
Fig. 2. SAR image pre-processing (taking BTR70 for example). (a) Original image, (b) Logarithmic image, (c) Threshold segmented image, (d) Result of the morphological filtering, (e) Result of geometric clustering.



(a) (b)



(c) (d)



(e)

Fig. 3. SAR image pre-processing (taking BMP2 for example). (a) Original image, (b) Logarithmic image, (c) Threshold segmented image, (d) Result of the morphological filtering, (e) Result of geometric clustering.

2.4 Image enhancement and normalization

Image enhancement (Gonzalez &Woods, 2002) can weaken or eliminate some useless information and give prominence to some useful information, which aims to enhance image quality by adopting a certain technology for a specific application. Here, we apply the power-law transformation to enhance the target image

$$\boldsymbol{K}(x,y)=\left[\boldsymbol{H}(x,y)\right]^{\alpha} \tag{3}$$

where $\boldsymbol{H}$, $\boldsymbol{K}$ denotes the former and latter transformed image respectively, α is an constant.

In practice, due to the difference of the distance between a target and radar, the intensity of echoes differs greatly. Thus, it is necessary to normalize the image. Here, a normalized method adopted is

$$\boldsymbol{J}(x,y)=\frac{\boldsymbol{K}(x,y)}{\sqrt{\sum_{x}\sum_{y}\left|\boldsymbol{K}(x,y)\right|^{2}}} \tag{4}$$

where $\boldsymbol{J}$, $\boldsymbol{K}$ denotes the former and latter normalized image respectively.

Due to the uncertainty of target location in a scene, 2-dimensional fast Fourier transform (2DFFT) is applied. Only half of the amplitude of Fourier is used as inputs of feature extraction due to its translation invariance and symmetric property, so that it can decrease the dimension of samples and reduce computation.

3. Feature extraction

Feature extraction is a key procedure in SAR ATR. If all pixels of an image are regarded as features, this would result in large requirements, high computation and performance loss. Therefore, it is necessary to extract target features.

3.1 Feature extraction based 2DPCA

Suppose that we have M pre-processed training samples $\{\boldsymbol{I}_1,\boldsymbol{I}_2,\cdots,\boldsymbol{I}_M\}$ with $\boldsymbol{I}_i\in \mathrm{R}^{m\times n}, i=1,2,\cdots,M$. Center them $\mathbf{I}_i=\boldsymbol{I}_i-\bar{\boldsymbol{I}}$, where $\bar{\boldsymbol{I}}=\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{I}_i$ is the mean of total training samples. For each centered sample $\mathbf{I}_i$, let project it onto W by the following linear transformation:

$$\boldsymbol{A}_i=\mathbf{I}_i\boldsymbol{W} \tag{5}$$

where the projection matrix $\boldsymbol{W}\in \mathrm{R}^{n\times r}$ satisfies: $\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\mathrm{I}_r$, and I_r is $r\times r$ identity matrix. Let us reconstruct the sample $\boldsymbol{I}_i$: $\boldsymbol{I}_i^{(\mathrm{Rec})}=\bar{\boldsymbol{I}}+\boldsymbol{A}_i\boldsymbol{W}^{\mathrm{T}}=\bar{\boldsymbol{I}}+\mathbf{I}_i\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}}$, the reconstruct error is $\left\|\boldsymbol{I}_i-\boldsymbol{I}_i^{(\mathrm{Rec})}\right\|$. The optimal projection matrix should minimize the sum of the reconstruct errors sum of all the training samples

$$\begin{aligned}\boldsymbol{W}_{\text{opt}} &= \arg\min_{\boldsymbol{W}} \sum_{i=1}^{M} \left\| \boldsymbol{I}_i - \boldsymbol{I}_i^{(\text{Rec})} \right\|_{\text{F}}^2 \\ &= \arg\min_{\boldsymbol{W}} \sum_{i=1}^{M} \left\| \boldsymbol{I}_i - \left(\overline{\boldsymbol{I}} + \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right) \right\|_{\text{F}}^2 \\ &= \arg\min_{\boldsymbol{W}} \sum_{i=1}^{M} \left\| \boldsymbol{I}_i - \overline{\boldsymbol{I}} - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right\|_{\text{F}}^2 \\ &= \arg\min_{\boldsymbol{W}} \sum_{i=1}^{M} \left\| \mathbf{I}_i - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right\|_{\text{F}}^2 \end{aligned} \tag{6}$$

where $\|\cdot\|_{\text{F}}$ denotes matrix F-norm. We have

$$\begin{aligned}\sum_{i=1}^{M} \left\| \mathbf{I}_i - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right\|_{\text{F}}^2 &= \sum_{i=1}^{M} \text{tr}\left[\left(\mathbf{I}_i - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right) \left(\tilde{\boldsymbol{I}}_i - \tilde{\boldsymbol{I}}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right)^{\text{T}} \right] \\ &= \sum_{i=1}^{M} \text{tr}\left[\left(\mathbf{I}_i - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \right) \left(\mathbf{I}_i^{\text{T}} - \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} \right) \right] \\ &= \sum_{i=1}^{M} \text{tr}\left(\mathbf{I}_i \mathbf{I}_i^{\text{T}} - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} + \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} \right) \\ &= \sum_{i=1}^{M} \text{tr}\left(\mathbf{I}_i \mathbf{I}_i^{\text{T}} - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} - \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} + \mathbf{I}_i \boldsymbol{W} \boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} \right) \\ &= \sum_{i=1}^{M} tr\left(\mathbf{I}_i \mathbf{I}_i^{\text{T}} \right) - \sum_{i=1}^{M} tr\left(\boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} \mathbf{I}_i \boldsymbol{W} \right) \end{aligned} \tag{7}$$

So, equation (6) is equivalent to the following formula

$$\begin{aligned}\boldsymbol{W}_{\text{opt}} &= \arg\max_{\boldsymbol{W}} \sum_{i=1}^{M} tr\left(\boldsymbol{W}^{\text{T}} \mathbf{I}_i^{\text{T}} \mathbf{I}_i \boldsymbol{W} \right) \\ &= \arg\max_{\boldsymbol{W}} \sum_{i=1}^{M} tr\left[\boldsymbol{W}^{\text{T}} \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right)^{\text{T}} \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \boldsymbol{W} \right] \\ &= \arg\max_{\boldsymbol{W}} tr\left(\boldsymbol{W}^{\text{T}} \boldsymbol{G}_{\text{t}} \boldsymbol{W} \right) \end{aligned} \tag{8}$$

where $\boldsymbol{G}_{\text{t}} = \sum_{i=1}^{M} \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right)^{\text{T}} \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right)$ is the covariance matrix of training samples. So, the optimal projection matrix $\boldsymbol{W}_{\text{opt}} = \left[\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_r \right] \in \text{R}^{n \times r} (r < n)$ with $\left\{ \boldsymbol{w}_i \middle| i = 1, 2, \cdots, r \right\}$ is the set of eigenvectors of $\boldsymbol{G}_{\text{t}}$ corresponding to the r largest eigenvalues.

For each training image $\boldsymbol{I}_i$, its feature matrix is

$$\begin{aligned}\boldsymbol{B}_i &= \left[\boldsymbol{y}_1^{(i)}, \cdots, \boldsymbol{y}_r^{(i)} \right] \\ &= \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \boldsymbol{W}_{\text{opt}} \\ &= \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \left[\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_r \right] \\ &= \left[\left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \boldsymbol{w}_1, \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \boldsymbol{w}_2, \cdots, \left(\boldsymbol{I}_i - \overline{\boldsymbol{I}} \right) \boldsymbol{w}_r \right] \in \text{R}^{m \times r} \end{aligned} \tag{9}$$

Given an unknown sample $\boldsymbol{I} \in \mathrm{R}^{m\times n}$, its feature matrix $\boldsymbol{B}$:

$$\begin{aligned}\boldsymbol{B} &= [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_r] \\ &= (\boldsymbol{I} - \overline{\boldsymbol{I}})\boldsymbol{W}_{\text{opt}} \\ &= \left[(\boldsymbol{I} - \overline{\boldsymbol{I}})\boldsymbol{w}_1, (\boldsymbol{I} - \overline{\boldsymbol{I}})\boldsymbol{w}_2, \cdots, (\boldsymbol{I} - \overline{\boldsymbol{I}})\boldsymbol{w}_r\right] \in \mathrm{R}^{m\times r}\end{aligned} \tag{10}$$

3.2 Feature extraction based two-stage 2DPCA

2DPCA only eliminates the correlations between rows, but disregards the correlations between columns. So it needs more features. This will lead to large storage requirements and cost much more time in classification phase. To further compress the dimension of feature matrices, two-stage 2DPCA is applied in this chapter. Its detailed implementation is described as follows (shown in Fig.4):

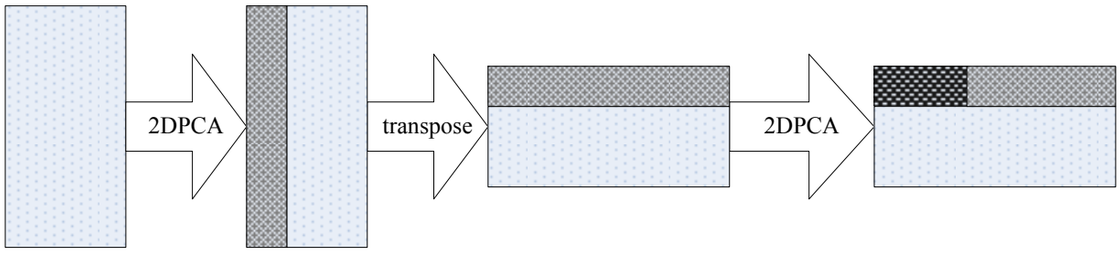


Fig. 4. Illustration of two-stage 2DPCA.

(1) Training images $\boldsymbol{I}_i \in \mathrm{R}^{m\times n}$ with $i = 1,2,\cdots,M$, calculate $\boldsymbol{G}_t$ by the section 3.1, and then obtain the row projection matrix $\boldsymbol{W}_{\text{ropt}} \in \mathrm{R}^{n\times r_1}\ (r_1 < n)$. Compute feature matrices $\boldsymbol{A}_i = (\boldsymbol{I}_i - \overline{\boldsymbol{I}})\boldsymbol{W}_{\text{ropt}} \in \mathrm{R}^{m\times r_1}$ for each training image.

(2) Regard the matrices $\boldsymbol{Z}_i = \boldsymbol{A}_i^{\mathrm{T}}\ (i = 1,2,\cdots,M)$ as new training samples, repeat the course of 2DPCA and get the column projection matrix $\boldsymbol{W}_{\text{copt}} \in \mathrm{R}^{m\times r_2}\ (r_2 < m)$.

Feature matrix of each training image is obtained

$$\begin{aligned}\boldsymbol{B}_i &= \boldsymbol{Z}_i\boldsymbol{W}_{\text{copt}} \\ &= \boldsymbol{A}_i^{\mathrm{T}}\boldsymbol{W}_{\text{copt}} \\ &= \boldsymbol{W}_{\text{ropt}}^{\mathrm{T}}\left(\boldsymbol{I}_i - \overline{\boldsymbol{I}}\right)^{\mathrm{T}}\boldsymbol{W}_{\text{copt}} \in \mathrm{R}^{r_1\times r_2}\ (i = 1,2,\cdots,M)\end{aligned} \tag{11}$$

Given a unknown image $\boldsymbol{I} \in \mathrm{R}^{m\times n}$, its feature matrix $\boldsymbol{B}$:

$$\boldsymbol{B} = \boldsymbol{W}_{\text{ropt}}^{\mathrm{T}}\left(\boldsymbol{I} - \overline{\boldsymbol{I}}\right)^{\mathrm{T}}\boldsymbol{W}_{\text{copt}} \in \mathrm{R}^{r_1\times r_2} \tag{12}$$

4. Classifier design

In this chapter, the nearest neighbor classifier based Euclid distance is used. Compute distances of feature matrices between unknown and all training samples. Then, the decision is that this test belongs to the same class as the training sample, which minimizes the distance.

4.1 Classification based 2DPCA features

2DPCA features of training image $\boldsymbol{I}_i$ and test $\boldsymbol{I}$ are $\boldsymbol{B}_i = \left[\boldsymbol{y}_1^{(i)}, \cdots, \boldsymbol{y}_r^{(i)}\right] = \left(\boldsymbol{I}_i - \bar{\boldsymbol{I}}\right)\boldsymbol{W}_{\text{opt}} \in \mathrm{R}^{m\times r}$, $\boldsymbol{B} = \left[\boldsymbol{y}_1, \cdots, \boldsymbol{y}_r\right] = \left(\boldsymbol{I} - \bar{\boldsymbol{I}}\right)\boldsymbol{W}_{\text{opt}} \in \mathrm{R}^{m\times r}$ where $\boldsymbol{y}_k^{(i)} = \left(\boldsymbol{I}_i - \bar{\boldsymbol{I}}\right)\boldsymbol{w}_k \in \mathrm{R}^{m\times 1}$, $\boldsymbol{y}_k = \left(\boldsymbol{I} - \bar{\boldsymbol{I}}\right)\boldsymbol{w}_k \in \mathrm{R}^{m\times 1}$, $k = 1,2,\cdots,r$, $i = 1,2,\cdots,M$. From the expressions, we see that column vectors $\boldsymbol{y}_k$ and $\boldsymbol{y}_k^{(i)}$ of $\boldsymbol{B}$ and $\boldsymbol{B}_i$ derive from the projections of $\boldsymbol{I}$ and $\boldsymbol{I}_i$ onto eigenvector $\boldsymbol{w}_k$ corresponding to eigenvalue λ_k. Therefore, the distance of feature matrices between the test and ith training image is defined as

$$d\left(\boldsymbol{B}, \boldsymbol{B}_i\right) = \sum_{k=1}^{r} \left\| \boldsymbol{y}_k - \boldsymbol{y}_k^{(i)} \right\|_2 \tag{13}$$

4.2 Classification based two-stage 2DPCA features

Given feature matrices $\boldsymbol{B} \in \mathrm{R}^{r_1\times r_2}$ and $\boldsymbol{B}_i \in \mathrm{R}^{r_1\times r_2}$ of a test $\boldsymbol{I}$ and training image $\boldsymbol{I}_i$ by two-stage 2DPCA.

(1) Definition Distance along row

Feature matrices $\boldsymbol{B} \in \mathrm{R}^{r_1\times r_2}$ and $\boldsymbol{B}_i \in \mathrm{R}^{r_1\times r_2}$ are written the following form $\boldsymbol{B} = \left[\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_{r_1}\right]^{\mathrm{T}}$, $\boldsymbol{B}_i = \left[\boldsymbol{x}_1^{(i)}, \boldsymbol{x}_2^{(i)}, \cdots, \boldsymbol{x}_{r_1}^{(i)}\right]^{\mathrm{T}}$, $\boldsymbol{x}_{k_1}$, $\boldsymbol{x}_{k_1}^{(i)}$ are row vectors with their dimension of r_2. Define the distance between the two feature matrices

$$d_1\left(\boldsymbol{B}, \boldsymbol{B}_i\right) = \sum_{k_1=1}^{r_1} \left\| \boldsymbol{x}_{k_1} - \boldsymbol{x}_{k_1}^{(i)} \right\|_2 \tag{14}$$

(2) Definition Distance along column

Feature matrices $\boldsymbol{B} \in \mathrm{R}^{r_1\times r_2}$ and $\boldsymbol{B}_i \in \mathrm{R}^{r_1\times r_2}$ are written as $\boldsymbol{B} = \left[\boldsymbol{y}_1, \cdots, \boldsymbol{y}_{r_2}\right]$, $\boldsymbol{B}_i = \left[\boldsymbol{y}_1^{(i)}, \cdots, \boldsymbol{y}_{r_2}^{(i)}\right]$, $\boldsymbol{y}_{k_2}$, $\boldsymbol{y}_{k_2}^{(i)}$ are row vectors with the dimension of r_1. Define their distance

$$d_2\left(\boldsymbol{B}, \boldsymbol{B}_i\right) = \sum_{k_2=1}^{r_2} \left\| \boldsymbol{y}_{k_2} - \boldsymbol{y}_{k_2}^{(i)} \right\|_2 \tag{15}$$

(3) Definition Distance along row and column

The distance between the test and the ith training image is defined

$$d\left(\boldsymbol{B}, \boldsymbol{B}_i\right) = d_1\left(\boldsymbol{B}, \boldsymbol{B}_i\right) + d_2\left(\boldsymbol{B}, \boldsymbol{B}_i\right) \tag{16}$$

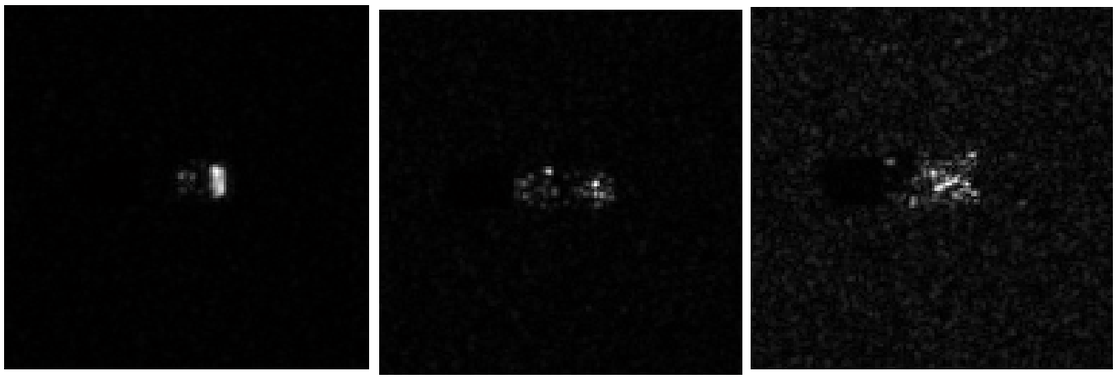
5. Experimental results

Experiments are made based on the MSTAR public release database. There are three distinct types of ground vehicles: BMP, BTR70, and T72. Fig.5 gives the optical images of the three classes of vehicles, and Fig.6 shows their SAR images.

There are seven serial numbers (i.e., seven target configurations) for the three target types: one BTR70 (sn-c71), three BMP2's (sn-c21, sn-9593, and sn-9566), and three T72's (sn-132, sn-812, and sn-s7). For each serial number, the training and test sets are provided, with the target signatures at the depression angles 17° and 15°, respectively. The training and test datasets are given in Table 1. The size of target images is converted 128×128 into 128×64 by our pre-processing described in section 2.

(a) BMP2 (b) BTR70 (c) T72

Fig. 5. Optical images of the three types of ground vehicles.



(a) BMP2 (b) BTR70 (c) T72

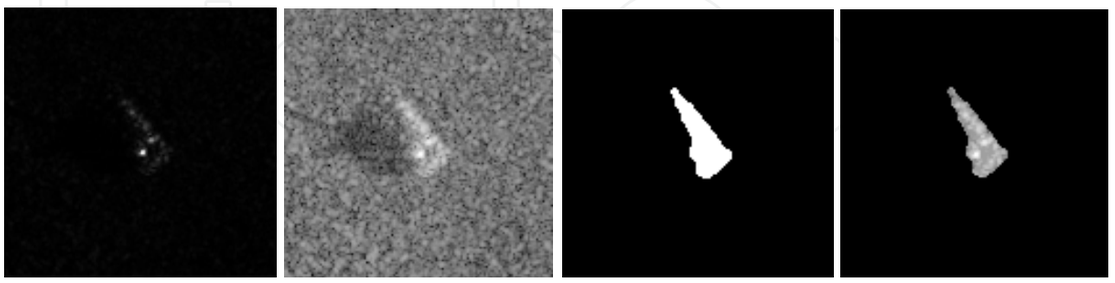
Fig. 6. SAR images of the three types of ground vehicles.

Training set	Number of samples	Testing set	Number of samples
BMP2sn-9563	233	BMP2sn-c21	196
		BMP2sn-9563	195
		BMP2sn-9566	196
BTR70sn-c71	233	BTR70sn-c71	196
T72sn-132	232	T72sn-132	196
		T72sn-812	195
		T72sn-s7	191

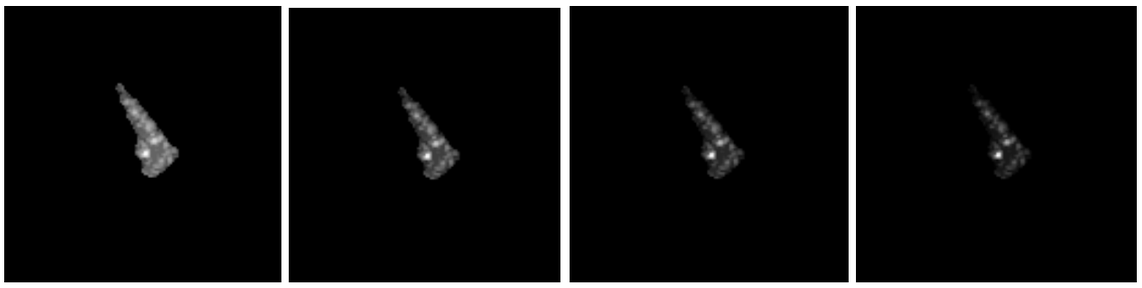
Table 1. Training and testing datasets.

5.1 The effects of logarithm conversion and power-law transformation with different exponents on the recognition rates in our pre-processing method

Let us illustrate the effects of logarithm conversion and power-law transformation with different exponents in our pre-processing using an image of T72, shown in Fig.7.



(a) (b) (c) (d)



(e) (f) (g) (h)

Fig. 7. The effects to image quality with different α . (a) Original image, (b) Logarithmic image, (c) Segmented binary target image, (d) Segmented Target intensity image, (e) Enhanced image with $\alpha = 2$, (f) Enhanced image with $\alpha = 3$, (g) Enhanced images with $\alpha = 4$, (h), Enhanced images with $\alpha = 5$.

From Fig.7 (a), we see that the total gray values are very low, and many details are not visible. On the one hand, logarithmic transformation converts speckles from multiple to additional model and makes image histogram more suitable be approximated with a Gaussian distribution. On the other hand, it enlarges the gray values and reveals more details.

However, image contrast in the target region decreases as shown in Fig.7 (d). Therefore, it is necessary to enhance image contrast, which can be accomplished by power-law transformation with $\alpha \geq 1$. The values of α corresponding to Fig.7 (e) ~ (h) are 2, 3, 4, and 5. We note that as α increases from 2 to 4, image contrast is enhanced distinctly. But when α continues to increase, the resulting image become dark and lose some details. By comparisons of these resulting images, we think that the best image enhancement result is at α taking 4 approximately.

We use the 698 samples of BMP2sn-9563, BTR70sn-c71, and T72sn-13 for training, 973 images of BMP2sn-9563, BMP2sn-9566, BTR70sn-c71, T72sn-132, T72sn-812, and T72sn-s7 for testing. The variation of recognition performance with α of 2DPCA is given in Fig. 8. We can see that 2DPCA obtains the highest recognition rate at $\alpha = 3.5$ (α is equal to 3.5 by default).

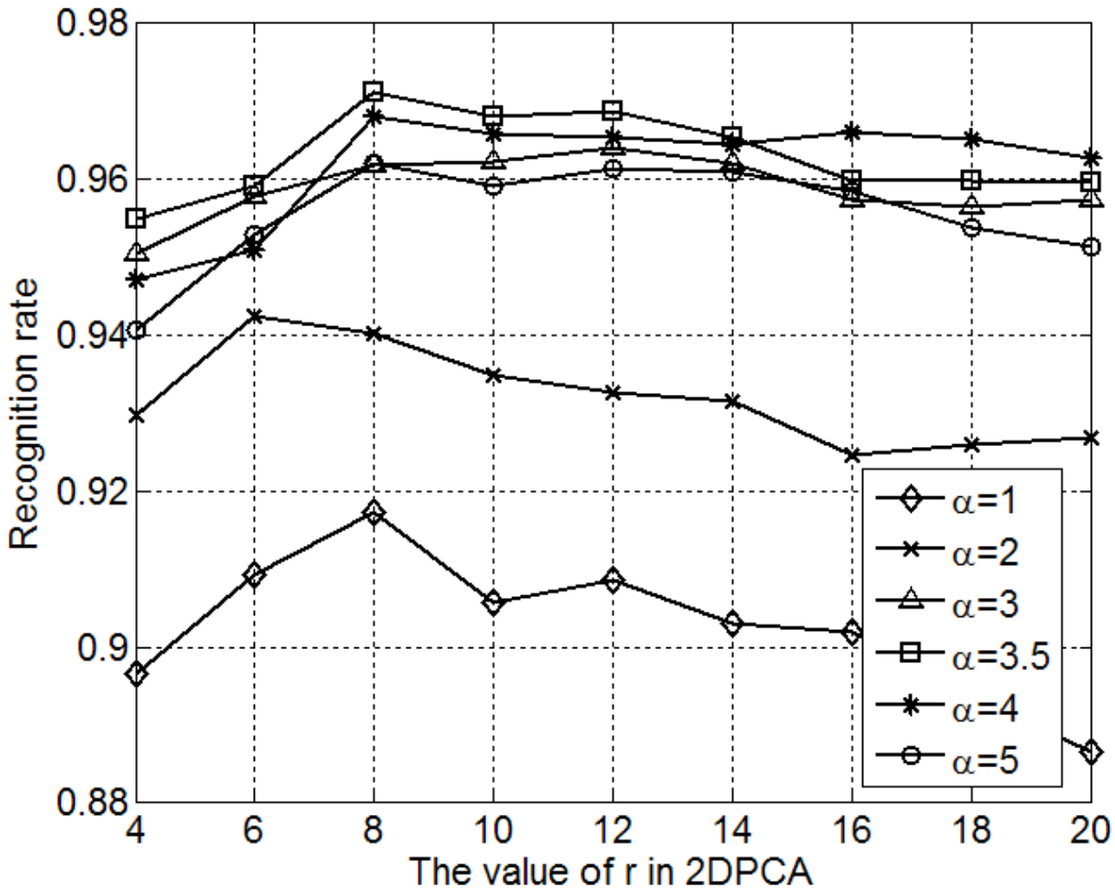


Fig. 8. Recognition rate with different α.

5.2 Comparisons of different pre-processing methods

In SAR recognition system, pre-processing is an important factor. Let us evaluate the performance of several pre-processing approaches as follows.

Method 1: the original images are transformed by logarithm. Then, half of the amplitudes of the 2-dimensional fast Fourier transform are used as inputs of feature extraction.

Method 2: overlaying the segmented binary target $\boldsymbol{T}_{\text{ar}}$ on the original image $\boldsymbol{F}$ gets target image, normalize it. Half of the amplitudes of 2-dimensional fast Fourier transform are used.

Method 3: overlaying $\boldsymbol{T}_{\text{ar}}$ on $\boldsymbol{F}$ obtains target image. First, enhance it using power-law transformation with an exponent 0.6. Then normalize it. Half of the amplitudes of 2-dimensional fast Fourier transform are used.

Method 4: overlaying $\boldsymbol{T}_{\text{ar}}$ on the logarithmic image $\boldsymbol{G}$ obtains target image, normalize it. Half of the amplitudes of 2-dimensional fast Fourier transform are used.

Method 5 (our pre-processing method in section 2): That is, overlaying $\boldsymbol{T}_{\text{ar}}$ on $\boldsymbol{G}$ obtains target image. First, enhance it using power-law transformation with an exponent 3.5. Then normalize it. Half of the amplitudes of 2-dimensional fast Fourier transform are used.

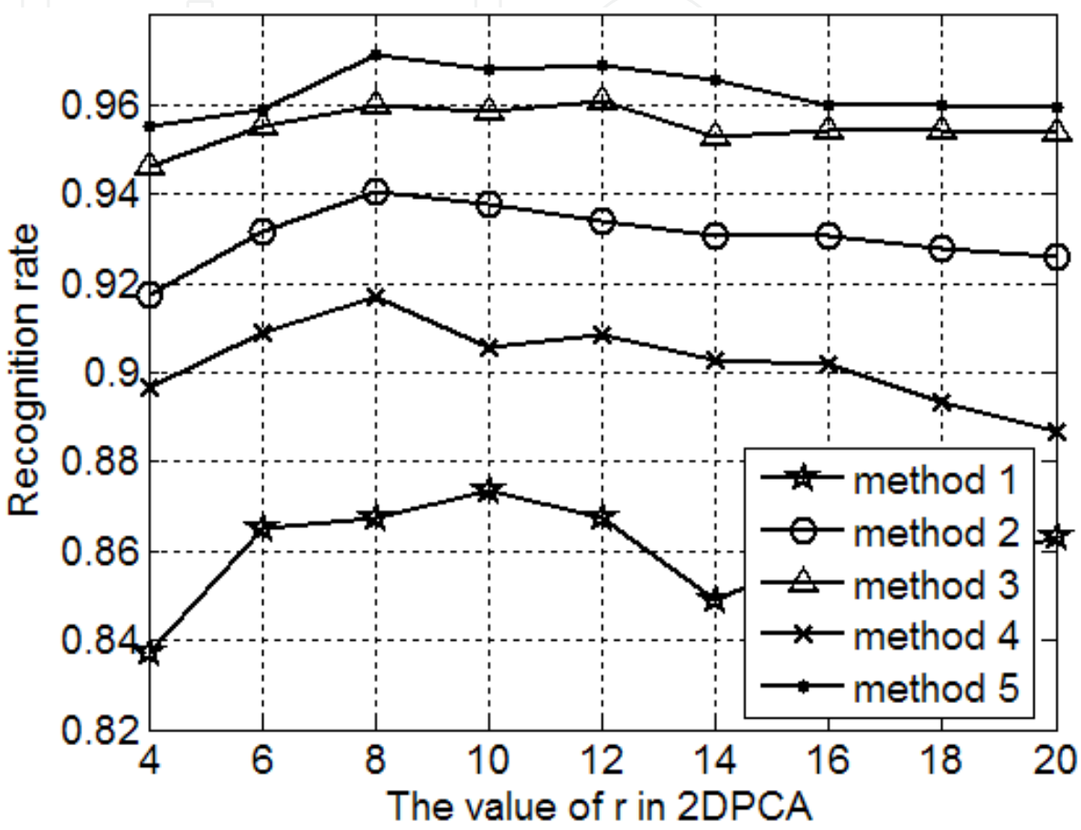


Fig. 9. Performances of 2DPCA with five pre-processing methods.

We also use the 698 samples of BMP2sn-9563, BTR70sn-c71, and T72sn-13 for training, 973 images of BMP2sn-9563, BMP2sn-9566, BTR70sn-c71, T72sn-132, T72sn-812, and T72sn-s7 for testing. The recognition rates of 2DPCA with these five pre-processing methods are given in Fig. 9.

We can see that the performance of method 1 is the worst, because it does not segment the target from background clutters, which disturb recognition performances.

Comparing method 3 with 2 and 5 with 4, we easily find that image enhancement based on power-law transformation is very efficient.

The difference between method 3 and method 5 (our pre-processing method) is that the former is obtained by overlaying $\boldsymbol{T}_{\text{ar}}$. on the original image $\boldsymbol{F}$, and then enhanced by power-law transformation with a fractional exponent 0.6. The latter is obtained by overlaying $\boldsymbol{T}_{\text{ar}}$ on the logarithm image $\boldsymbol{G}$, and then enhanced by power-law transformation

with an exponent 3.5. Due to the effects of logarithm in our method, the performance of method 5 (our pre-processing method) is better than that of method 3.

All the five experimental results testify that our pre-processing method is very efficient.

5.3 Comparisons of 2DPCA and PCA

To further evaluate our feature extraction method, we also compare 2DPCA with PCA. The flow chart of experiments is given in Fig.10.

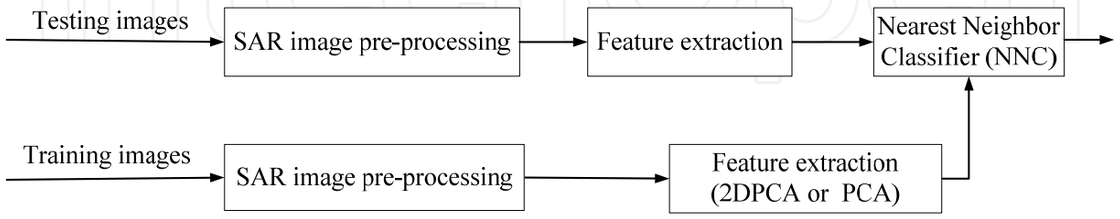


Fig. 10. Flow chart of our SAR ATR experiments.

For all the training and testing samples in Table 1, Fig.11 gives the variation of recognition rates of PCA with feature dimensions, that is, the number of principal components. PCA achieves the highest recognition rate when the number of principal components (d) equal 85.

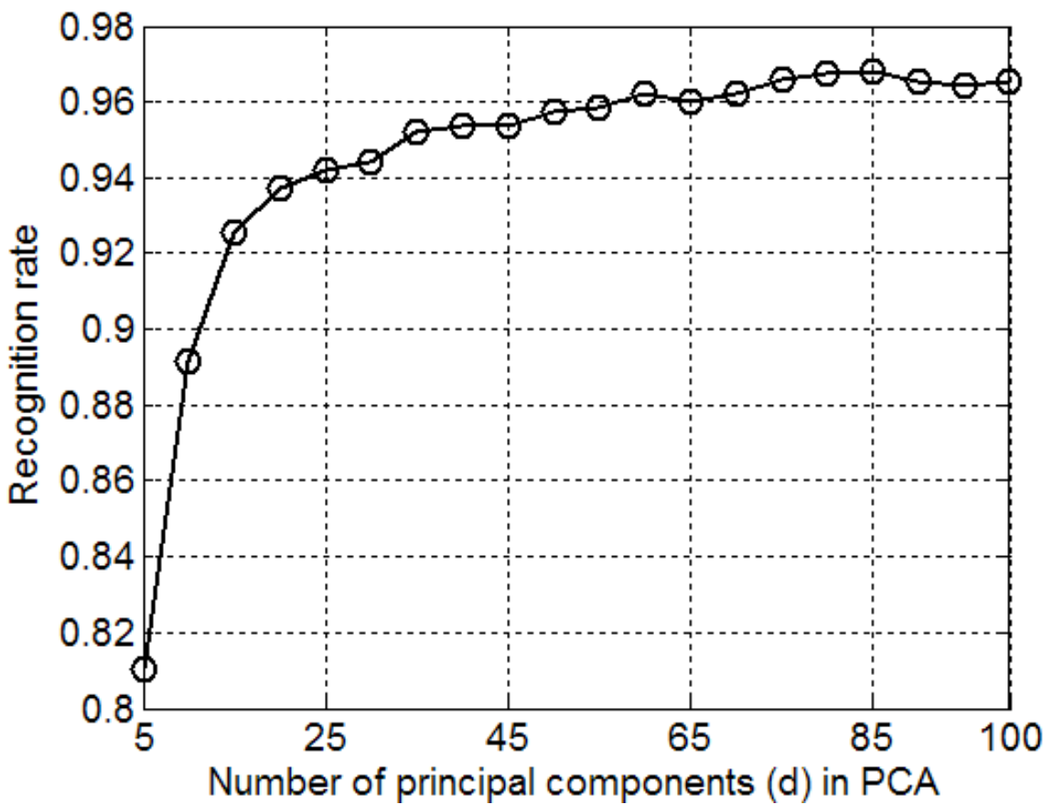


Fig. 11. Variation of recognition rates of PCA with the number of principal components.

For all the training and testing samples in Table 1, Fig.12 gives the variation of recognition rates of 2DPCA with the number of principal components. 2DPCA achieves the highest recognition rate when the number of principal components (r) equal 8.

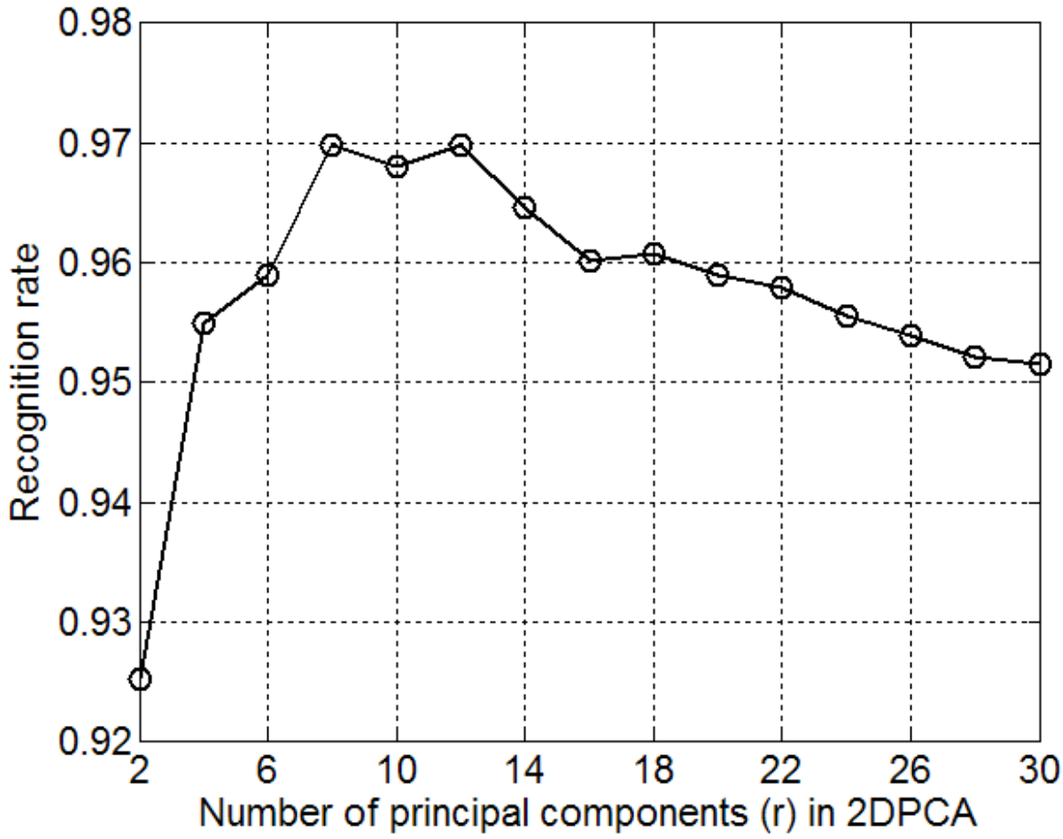


Fig. 12. Variation of recognition rates of 2DPCA with the number of principal components.

Table 2 shows the highest recognition rates of PCA and 2DPCA. We see that the highest recognition performance of 2DPCA is slightly better than that of PCA.

	Highest recognition rate (dimension of feature vectors or feature matrices)
PCA+NNC	96.81 (85)
2DPCA+NNC	96.98 (128×8)

Table 2. Comparisons of the highest recognition rates of PCA and 2DPCA

By Comparing Fig.11 with Fig.12, we find that recognition performance of 2DPCA is better than PCA. This is due to the facts that 2-dimensional image matrices must be transformed into 1-dimensional image vectors when PCA used in image feature extraction. The image matrix-to-vector transformation will result in some problems: (1) This will destroy 2-dimensional spatial structure information of image matrix, which brings on performance loss; (2) This leads to a high dimensional vector space, where it is difficult to evaluate the covariance matrix accurately and find its eigenvectors because the dimension of the

covariance matrix is very large ($m \cdot n \times m \cdot n$). 2DPCA estimates the covariance matrix based on 2-dimensional training image matrices, which leads to two advantages: (1) 2-dimensional spatial structure information of image matrix is kept very well; (2) the covariance matrix is evaluated more accurately and the dimensionality of the covariance matrix is very small ($n \times n$). So, the efficiency of 2DPCA is much greater than that of PCA.

Table 3 gives the computation complexity and storage requirements of PCA and 2DPCA, in which the storage requirements include two parts: the projection vectors and features of all training samples ($M = 698, m = 128, n = 64, r = 8, l = 20$). From this table, we see that although the storage requirements are comparative, the computation complexity of 2DPCA is much smaller than that of PCA when seeking the projection vectors. So, we think that 2DPCA is much greater than PCA in computation efficiency.

	PCA	2DPCA
Size of covariance matrix	$m \cdot n \times m \cdot n$ $= (128 \cdot 64) \times (128 \cdot 64)$ or $M \times M$ $= 698 \times 698$	$n \times n$ $= 64 \times 64$
Complexity of finding projection vectors	$o(M^3)$ $= o(698^3)$	$o(n^3)$ $= o(64^3)$
Complexity of finding features	$d \times (m \cdot n) \times 1$ $= 85 \times (128 \cdot 64) \times 1$ $= 696{,}320$	$m \times n \times r$ $= 128 \times 64 \times 8$ $= 65{,}536$
Storage of features	$(m \cdot n) \times d + d \times M$ $= 128 \times 64 \times 85 + 85 \times 698$ $= 755{,}650$	$n \times r + m \times r \times M$ $= 64 \times 8 + 128 \times 8 \times 698$ $= 715{,}264$

Table 3. Comparisons of the computation complexity and storage requirements of PCA and 2DPCA

From Table 2 and Table 3, we can conclude that 2DPCA is better than PCA in computation efficiency and recognition performance.

From Table 2, we also see that feature matrix obtained by 2DPCA is considerably large. This may lead to massive memory requirements and cost too much time in classification phase. So, we proposed two-stage 2DPCA to reduce feature dimensions.

5.4 Comparisons of 2DPCA and two-stage 2DPCA

2DPCA only eliminates the correlations between rows, but disregards the correlations between columns. The proposed two-stage 2DPCA can eliminate the correlations between images rows and columns simultaneously, thus reducing feature dimensions dramatically and improving recognition performances.

Table 4 shows the highest recognition rates of 2DPCA and two-stage 2DPCA.

Table 5 gives the computation complexity and storage requirements of 2DPCA and two-stage 2DPCA. The storage requirements also include two parts: projection matrices and feature matrices of all training samples ($M = 698$, $m = 128$, $n = 64$, $r = 8$, $l = 20$, $r_1 = 12$, $r_2 = 22$).

From Table 4, we see that two-stage 2DPCA achieves the highest recognition performance with smaller feature matrices.

From Table5, we find that the storage requirements of two-stage 2DPCA are smaller than those of 2DPCA.

From Table 4 and Table 5, we can conclude that two-stage 2DPCA is better than 2DPCA in recognition performance and storage requirements.

From Table 4, we also see that the results of two-stage 2DPCA are comparative no matter how the distance between two features is defined and the recognition performance of the way of the distance along row and column is slightly better.

Recognition approaches	Recognition rate (feature dimension)
2DPCA	96.98 (128×8)
Two-stage 2DPCA (Definition Distance along column)	97.21 (12×22)
Two-stage 2DPCA (Definition Distance along row)	97.32 (10×30)
Two-stage 2DPCA (Definition Distance along row and column)	**97.55 (12×22)**

Table 4. Comparisons of recognition performances of 2DPCA and two-stage 2DPCA (%).

	Complexity of finding projection vectors	Complexity of finding features	Storage of features
2DPCA	$o\left(n^3\right)$ $= o\left(64^3\right)$	$m \times n \times r$ $= 128 \times 64 \times 8$ $= 65{,}536$	$n \times r + m \times r \times M$ $=64 \times 8+128 \times 8 \times 698$ $=715{,}264$
Two-stage 2DPCA	$o\left(m^3\right) + o\left(n^3\right)$ $= o\left(128^3\right) + o\left(64^3\right)$	$r_1 \times n \times m + r_1 \times m \times r_2$ $=12 \times 64 \times 128+12 \times 128 \times 22$ $=132{,}096$	$n \times r_1 + m \times r_2 + r_1 \times r_2 \times M$ $=64 \times 12 + 128 \times 22 + 12 \times 22 \times 698$ $=187{,}856$

Table 5. Comparisons of storage requirements of 2DPCA and Two-stage 2DPCA.

5.5 Comparisons of 2DPCA and two-stage 2DPCA under different azimuth intervals

In some cases, we can obtain target azimuth. Using it, recognition performances may be improved. Group training samples with equal intervals for each class within $0^\circ \sim 360^\circ$, then extract features within the same azimuth range for the three types of training samples in the phase of training. In the phase of testing, the test sample is chosen to be classified in the corresponding azimuth range according to its azimuth. In this experiment, training samples of each class are grouped with equal intervals by 180°, 90°, 30° respectively.

Recognition results of 2DPCA and two-stage 2DPCA under different azimuth intervals (180°, 90°, and 30°) are given in Table 6. From it, we obtain that performances of the two-stage 2DPCA method is better than those of 2DPCA. Moreover, two-stage 2DPCA is robust to the variation of azimuth. This table further proves that two-stage 2DPCA combining with our pre-processing method is efficient.

Recognition approaches	180°	90°	30°
2DPCA	98.18	94.75	93.40
Two-stage 2DPCA (Definition Distance along column)	98.27	**95.49**	*94.91*
Two-stage 2DPCA (Definition Distance along row)	98.25	95.07	*94.71*
Two-stage 2DPCA (Definition Distance along row and column)	**98.43**	95.23	***95.04***

Table 6. Performances of 2DPCA and two-stage 2DPCA under different azimuth intervals (%)

5.6 Comparisons of two-stage 2DPCA and methods in literatures

The recognition rates of two-stage 2DPCA and methods in literatures are listed in Table 7.

Recognition approaches	Recognition rate (feature dimension)
Template matching (Zhao & Principe, 2001)	40.76
SVM (Bryant & Garber, 1999)	90.92
PCA+SVM (Han et al., 2003)	84.54
KPCA+SVM (Han et al., 2003)	91.50
KFD+SVM (Han et al., 2004)	95.75
$(2D)^2PCA$[9]combining our pre-processing (Definition Distance along column)	97.38 (22×12)
$(2D)^2PCA$[9] combining our pre-processing (Definition Distance along row)	97.15 (22×12)
$(2D)^2PCA$[9] combining our pre-processing (Definition Distance along row and column)	97.61 (22×12)
G2DPCA[10] combining our pre-processing (Definition Distance along column)	97.44 (22×12)
G2DPCA[10] combining our pre-processing (Definition Distance along row)	97.32 (24×12)
G2DPCA[10] combining our pre-processing (Definition Distance along row and column)	97.66 (22×12)
Two-stage 2DPCA (Definition Distance along column)	97.21 (12×22)
Two-stage 2DPCA (Definition Distance along row)	97.32 (10×30)
Two-stage 2DPCA (Definition Distance along row and column)	97.55 (12×22)

Table 7. Performances of two-stage 2DPCA and several methods in literatures (%)

We see that performances of literatures (Zhao et al., 2001; Bryant & Garber, 1999) are the worst, since they do not have any pre-processing and feature extraction. However, efficient pre-processing and feature extraction can help to improve recognition performances.

In literatures (Han et al., 2003; Han et al., 2004), PCA, KPCA, or KFD is employed. These feature extraction methods seek projection vectors based on 1-dimensional image vectors.

In our ATR system, target is firstly segmented to eliminate background clutters. Then, enhanced by power-law transformation to stand out useful information and strengthen target recognition capability. Moreover, feature extraction is based on 2-dimensional image matrices, so that the spatial structure information is kept very well and the covariance matrix is estimated more accurately and efficiently. Therefore, two-stage 2DPCA combining with our proposed pre-processing method can obtain the best recognition performance.

By comparisons of two-stage 2DPCA and the similar techniques, such as $(2D)^2PCA$ (Zhang & Zhou, 2005), G2DPCA (Kong et al., 2005), we can conclude that our pre-processing method is very efficient and two-stage 2DPCA is comparable to $(2D)^2PCA$ and G2DPCA in performance and storage requirements.

Table 8 gives the results of two-stage 2DPCA, and other approaches in literatures under different azimuth intervals. From it, we obtain that performances of our method is better than those of literatures. This table further validates that two-stage 2DPCA combining with our pre-processing method is the best.

Recognition approaches	180°	90°	30°
Template matching (Zhao & Principe, 2001)	45.79	56.92	70.55
SVM (Bryant & Garber, 1999)	89.89	88.35	90.62
PCA+SVM (Han et al., 2003)	88.79	95.02	94.73
KPCA+SVM (Han et al., 2003)	92.38	95.46	95.16
KFD+SVM (Han et al., 2004)	95.46	97.14	95.75
$(2D)^2PCA$ (Zhang & Zhou, 2005) combining our pre-processing (Definition Distance along column)	98.14	95.01	94.76
$(2D)^2PCA$ (Zhang & Zhou, 2005) combining our pre-processing (Definition Distance along row)	98.27	95.47	94.70
$(2D)^2PCA$ (Zhang & Zhou, 2005) combining our pre-processing (Definition Distance along row and column)	98.31	95.26	94.75
G2DPCA (Kong et al., 2005) combining our pre-processing (Definition Distance along column)	98.30	95.07	94.53
G2DPCA (Kong et al., 2005) combining our pre-processing (Definition Distance along row)	98.27	95.45	94.92
G2DPCA (Kong et al., 2005) combining our pre-processing (Definition Distance along row and column)	98.31	95.16	94.84
Two-stage 2DPCA (define distance along row)	98.27	95.49	*94.91*
Two-stage 2DPCA (define distance along column)	98.25	95.07	*94.71*
Two-stage 2DPCA (define distance along row and column)	98.43	95.23	*95.04*

Table 8. Performances of two-stage 2DPCA and several methods in literatures under different azimuth intervals (%)

From this table, we also see that recognition performances of two-stage 2DPCA achieve the best under the 180° azimuth intervals.

However, with the azimuth interval decreasing, recognition performances of two-stage 2DPCA become worse. This is because number of training samples at intervals becomes smaller; it is not useful for estimating the covariance matrix exactly, thus resulting in recognition performance loss. While in literatures (Bryant & Garber, 1999; Han et al., 2003; Han et al., 2004), the classifier of SVM is employed, this is fit for a small sample classification.

Comparisons of two-stage 2DPCA and the $(2D)^2PCA$ and G2DPCA methods under different azimuth intervals, we think that our pre-processing method is very efficient and two-stage 2DPCA is comparable to $(2D)^2PCA$ and G2DPCA.

6. Conclusions

An efficient SAR pre-processing method is first proposed to obtain targets from background clutters, and two-stage 2DPCA is proposed for SAR image feature extraction in this chapter. Comparisons with 2DPCA and other approaches prove that two-stage 2DPCA combining with our pre-processing method not only decreases sharply feature dimensions, but also increases recognition rate. Moreover, it is robust to the variation of target azimuth, and decreases the precision requirements for the estimation of target azimuth.

7. References

Bryant, M. & Garber, F. (1999). SVM Classifier applied to the MSTAR public data set, *SPIE*, Vol. 3721, No. 4, (April 1999), pp. 355-359.

Gonzalez, R.C. & Woods R. E. (2002) *Digital Image Processing*, Publishing House of Electronics Industry, Beijing, China.

Han, P.; Wu, R. B. & Wang, Z. H. (2003). SAR Automatic Target Recognition based on KPCA Criterion, *Journal of Electronics and Information Technology*, Vol. 25, No.10, (October 2003), pp.1297-1301.

Han, P.; Wu, R. B. & Wang, Z. H. (2003). SAR Target Feature Extraction and Automatic Recognition Based on KFD Criterion, *Modern Radar*, Vol. 26, No.7, (July 2004), pp. 27-30.

Kong, H.; Wang, L. & Teoh, E. K. (2005) Generalized 2D Principal Component Analysis for face image representation and recognition, Neural Networks, Vol. 18, (2005), pp. 585-594.

Musman, S. & Kerr, D. (1996). Automatic Recognition of ISAR Ship Images, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 32, No. 4, (October 1996), pp. 1392-1404.

Ross, T.; Worrell, S. & Velten, V. (1998). Standard SAR ATR evaluation experiment using the MSTAR public release data set, *SPIE*, Vol.3370, No.4, (April 1998), pp. 66-573.

Yang, J.; Zhang, D. & Frangi, A. F. (2004). Two-dimensional PCA: a new approach to appearance-based face representation and recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 1, (January 2004), pp. 131- 137.

Zhang, D. Q. & Zhou Z. H. (2005). $(2D)^2PCA$: Two-directional two-dimensional PCA for efficient face representation and recognition, *Neurocomputing*, Vol. 69, (2005), pp. 224-231.

Zhao, Q. & Principe, J. C. (2001). Support Vector Machine for SAR automatic target recognition, *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, No. 2, (April 2001), pp. 643-654.

Principal Component Analysis - Engineering Applications
Edited by Dr. Parinya Sanguansat

ISBN 978-953-51-0182-6
Hard cover, 230 pages
Publisher InTech
Published online 07, March, 2012
Published in print edition March, 2012

This book is aimed at raising awareness of researchers, scientists and engineers on the benefits of Principal Component Analysis (PCA) in data analysis. In this book, the reader will find the applications of PCA in fields such as energy, multi-sensor data fusion, materials science, gas chromatographic analysis, ecology, video and image processing, agriculture, color coating, climate and automatic target recognition.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen