2006 International Joint Conference on Neural Networks
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

# KICA Feature Extraction in Application to FNN based Image Registration

Anbang Xu, Xin Jin, Ping Guo[*], *Senior Member, IEEE*, Rongfang Bie

*Abstract*—**In this paper, a novel image registration method is proposed. In the proposed method, kernel independent component analysis (KICA) is applied to extract features from the image sets, and these features are input vectors of feedforward neural networks (FNN). Neural network outputs are those translation, rotation and scaling parameters with respect to reference and observed image sets. Comparative experiments are performed between KICA based method and other six feature extraction based method: principal component analysis (PCA), independent component analysis (ICA), kernel principal component analysis (KPCA), the discrete cosine transform (DCT), Zernike moment and the complete isometric mapping (Isomap). The results show that the proposed method is much improved not only at accuracy but also remarkably at robust to noise.**

## I. INTRODUCTION

Image registration is the process of aligning two or more images of the same scene. Image registration techniques are embedded in a lot of visual intelligent systems, such as robotics, target recognition, remote medical treatment and autonomous navigation. The common image registration methods are divided into two types: intensity-based methods and feature-based methods. The analysis and evaluation for various techniques and methods of image registration are carried out on the basis of these two sorts, while the feature-based methods are emphasized.

Recently, Itamar Ethanany [1] proposed to use feedforward neural network (FNN) to register a distorted image through 144 Discrete Cosine Transform (DCT)-base band coefficients as the feature vector. But this method has too large lumber of input feature vectors for the un-orthogonality of DCT based space, thus suffered low computational efficiency and high requirements on computer performance. Later, Wu and Xie [2] used low order Zernike moments instead of DCT coefficients to register affine transform parameters but the estimation accuracy is still not satisfied. We proposed to use the complete isometric mapping (Isomap) [3] for feature extraction. Although the input vector dimension is reduced, to add new data to the basic Isomap has to be run again and the resulted solution may disarrange the features that were used to train the network.

A key step in these image registration schemes is to extract image features to form the FNN training data set. The main challenge in this step is how to reduce the high dimensional input data and retain the feature for image registration. In this paper, we develop a novel method to image registration, which uses kernel independent component analysis (KICA) for feature extraction, and then these features as input vectors are fed into a FNN to obtain register affine transform parameters. Experimental results show that the scheme we proposed is better than other methods in terms of accuracy and robustness.

This paper is organized as follows: In section II, the new KICA and FNN based image registration scheme and the algorithm are presented. Section III focuses on experimental result comparison with other methods under various noisy conditions. Finally, the conclusions are presented in section IV.

## II. KICA AND FNN BASED IMAGE REGISTRATION SCHEME

The image registration scheme consists of two stages: the pre-registration phase and the registration phase. In the pre-registration phase, first, a training set is synthesized by the reference image. The feature coefficients are extracted from the training set with the method of KICA, and then these feature coefficients as inputs are fed to a FNN. Second, a neural network is trained and its target outputs are affine parameters. In the registration phase, since the neural network is trained, the remainder work is simple: We just use the same method to extract features from the registered image and feed these features to the trained network to get the estimated affine parameters. This registration approach is shown in Fig. 1.

### A. *Affine Transformation*

Geometrical transformation can be represented in many different ways, affine transformation is one of the most common used transformations. An affine transformation is the transformation that preserves collinearity. Geometric contraction, expansion, dilation, reflection, rotation, shear, similarity transformations, spiral similarities, translation as well as these combinations are all belonging to affine transformations,. In this paper, in order to make comparisons with the methods proposed in the literatures [1, 2, 3], we adopt the affine transformation which is the composition of rotations, translations, dilations. Images can be represented with two dimensional matrices and the affine transformation can be described by the following matrix equation [1]:

The research work described in this paper was fully supported by a grant from the National Natural Science Foundation of China (Project No. 60275002).

Anbang Xu, Xin Jin, Ping Guo, and Rongfang Bie are all with Image Processing and Pattern Recognition Laboratory, Beijing Normal University, Beijing 100875, China. Prof. Ping Guo is the corresponding author (phone: +8610-5880-6662; fax: +8610-5880-9444; email: pguo@ieee.org).

0-7803-9490-9/06/$20.00/©2006 IEEE

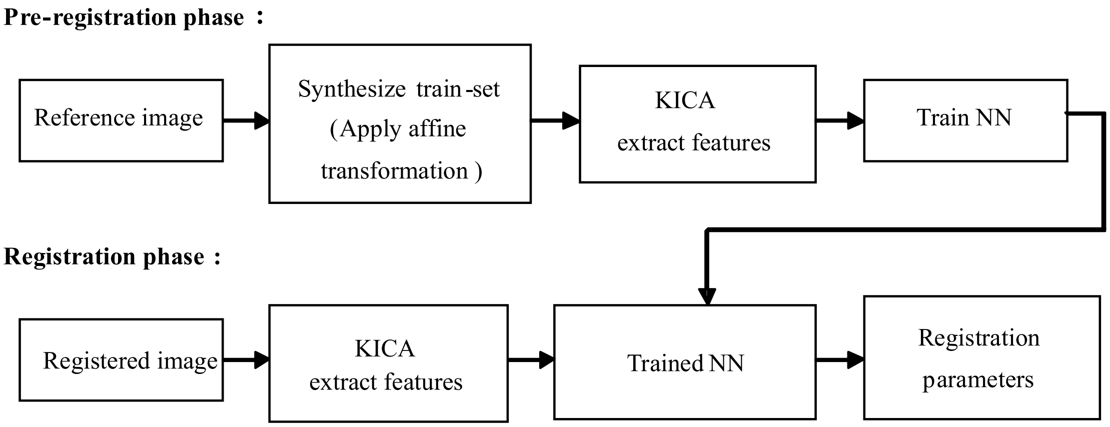


Fig. 1. The pre-registration and registration phases of the proposed scheme.

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix}. \qquad (1)$$

In the equation, there are four basic parameters for the transformation, where ($x_1$, $y_1$) denotes the original image coordinate, ($x_2$, $y_2$) denotes the transformed image coordinate in another image, $t_x$, $t_y$ are the translation parameters, $s$ is a scaling factor and $\theta$ is a rotation angle. In this paper, we will adopt this transformation model.

### *B. Feature Extraction Methods*

1) *Principal Component Analysis* (*PCA*): PCA is a classical multivariate data analysis method that is useful in linear feature extraction and data compression [4]. It is essentially equivalent to Karhunen-Loeve transformation and closely related to factor analysis. All these methods are based on second-order statistics of the data.

The PCA finds a linear transformation **y**=**Wx** such that the retained variance is maximized. It can be also viewed as a linear transformation that minimizes the reconstruction error [5]. Each row vector of **W** corresponds to the normalized orthogonal eigenvector of the data covariance matrix.

One simple approach to PCA is to use singular value decomposition (SVD). Let us denote the data covariance matrix by $\mathbf{R_x}(0) = E\{\mathbf{x}(t)\mathbf{x}^{\mathrm{T}}(t)\}$. Then the SVD of $\mathbf{R_x}(0)$ gives $\mathbf{R_x}(0) = \mathbf{UDU}^{\mathrm{T}}$, where $\mathbf{U}=[\mathbf{U}_{\mathrm{s}}, \mathbf{U}_{\mathrm{n}}]$ is the eigenvector matrix (i.e. modal matrix) and **D** is the diagonal matrix whose diagonal elements correspond to the eigenvalues of $\mathbf{R_x}(0)$ in descending order. Then the PCA transformation from $m$-dimensional data to $n$-dimensional subspace is given by choosing the first $n$ column vectors, i.e., $n$ principal component vector **y** is given by $\mathbf{y}=\mathbf{U}_{\mathrm{s}}^{\mathrm{T}}\mathbf{x}$.

2) *Independent Component Analysis* (*ICA*): ICA is a recently developed method [6] that can be considered as generalization of PCA. A most important application of ICA is unsupervised feature extraction, where the aim is to linearly transform the input data into uncorrelated components, along which the distribution of the sample set is the least Gaussian.

Suppose there are $m$ independent components, which are denoted as $\mathbf{Y}=[y_1, y_2, \ldots, y_m]$. The observed signals are $\mathbf{X}=[x_1, x_2, \ldots, x_n]$ and the classical model of ICA is:

$$\mathbf{X} = \mathbf{AY} \qquad (2)$$

where **A** is an unknown matrix. The ICA algorithm is to calculate a matrix **W**, which is inverse matrix of **A**. So the independent components can be estimated:

$$\tilde{\mathbf{Y}} = \mathbf{WX} \qquad (3)$$

where $\tilde{\mathbf{Y}}$ is the estimation of **Y**.

Many methods have been proposed to obtain the matrix **W**. For example, Bell and Sejnowski [6] designed a learning algorithm based on the information maximization, which is named as INFOMAX, and Amari [7] proposed a natural gradient method for better convergence. Their learning algorithm for **W** can be summarized as the following equation:

$$\Delta\mathbf{W} = (\mathbf{I} + g(s)s^T)\mathbf{W} \qquad (4)$$

where $s$ = **WX** and $g(s) = 1\text{-}2/(1+e^{-s})$. Before this learning procedure, a preprocessing operation $\tilde{\mathbf{W}}$, known as whitening or sphering, is required for most ICA learning algorithms [11]. The transformed data is zero mean and de-correlated data:

$$\tilde{\mathbf{W}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\tilde{\mathbf{W}}^{\mathrm{T}} = \mathbf{I} \qquad (5)$$

The transformation can be accomplished by eigenvalue decomposition. The Eq.(5) can be satisfied as $\tilde{\mathbf{W}} = \mathbf{\Lambda}^{-1/2}\mathbf{V}^T$. Here, **Λ** and **V** are the eigenvalues matrix and eigenvectors matrix of the covariance matrix of **X**, respectively.

3) *Kernel Independent Component Analysis* (*KICA*): ICA is a linear method that has been used widely to perform data redundancy reduction and feature extraction for linear model. However, most data is nonlinear distribution which is too complex to be represented well by a linear model. Recently Bach and Jordan proposed a new learning method of ICA, which use contrast functions based on canonical correlations

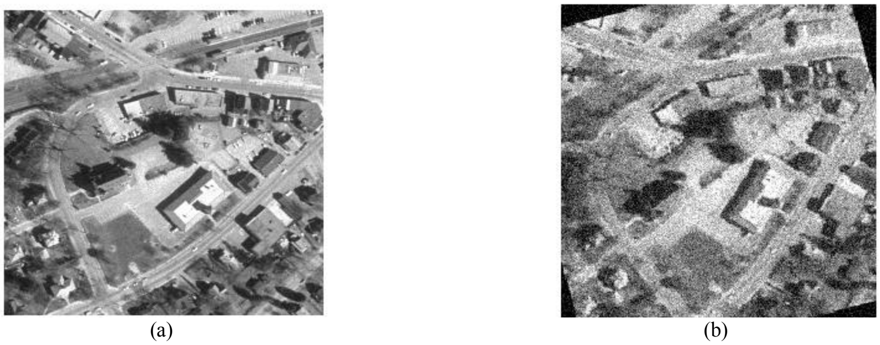


Fig. 2. (a) An original image and (b) a registered image in the training set with 15-degree rotation, 110% scaling, translation of 2 pixels and -3 pixels on X-axis and Y-axis respectively at a signal-to-noise ratio (SNR) of 15 dB.

in a Reproducing Kernel Hilbert Space (RKHS) [8]. Harmeling *et al* presented a kernel-based blind source separation algorithm in the blind separation of nonlinearly mixed speech signals [9] and Liu and Cheng *et al* proposed a new algorithm that incorporates ICA and the kernel trick to improve face recognition [10] and texture classification [11].

The main idea of KICA is to map the input data into an implicit feature space ***F*** firstly: $\mathbf{\Phi}: x \in \boldsymbol{R}^N \rightarrow \mathbf{\Phi}(x) \in \boldsymbol{F}$. Then ICA is performed in ***F*** to produce a set of nonlinear features of input data.

As ICA algorithm described in the above part, the input data **X** is whitened in feature space ***F***. The whitening matrix is: $\tilde{\mathbf{W}}_{\mathbf{\Phi}} = (\mathbf{\Lambda}_{\Phi})^{-\frac{1}{2}}(\mathbf{V}_{\Phi})^T$, here $\mathbf{\Lambda}_{\Phi}$, $\mathbf{V}_{\Phi}$ are the eigenvalues matrix and eigenvectors matrix of covariance matrix $\hat{C} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{\Phi}(x_i)\mathbf{\Phi}(x_i)^T$, respectively. Then we can obtain the whitened data $\mathbf{X}_{\Phi}^{\mathrm{W}}$ as

$$\mathbf{X}_{\mathbf{\Phi}}^{\mathbf{W}} = (\tilde{\mathbf{W}}_{\mathbf{\Phi}})^T \mathbf{\Phi}(\mathbf{X}) = (\mathbf{\Lambda}_{\Phi})^{-1}\alpha^T \mathbf{K}, \tag{6}$$

where **K** is defined by $K_{ij} := (\mathbf{\Phi}(x_i) \cdot \mathbf{\Phi}(x_j))$ and $\alpha$ is the eigenvectors matrix of **K**. After the whitening transformation, Then Eq. (4) is calculated by the following iterative algorithm:

$$\tilde{\mathbf{Y}}_{\Phi} = \mathbf{W}_{\Phi}\mathbf{X}_{\Phi}, \tag{7}$$

$$\Delta\mathbf{W}_{\mathbf{\Phi}} = [\mathbf{I} + (\mathbf{I} - \frac{2}{1+e^{-\tilde{\mathbf{Y}}_{\mathbf{\Phi}}}})(\tilde{\mathbf{Y}}_{\mathbf{\Phi}})^T]\mathbf{W}_{\mathbf{\Phi}}, \tag{8}$$

$$\hat{\mathbf{W}}_{\mathbf{\Phi}} = \mathbf{W}_{\mathbf{\Phi}} + \rho\Delta\mathbf{W}_{\mathbf{\Phi}} \rightarrow \mathbf{W}_{\mathbf{\Phi}}, \tag{9}$$

until $\mathbf{W}_{\Phi}$ converged, and $\rho$ is a learning constant. According to the above algorithm, the feature of a test data s can be obtained by:

$$y = \mathbf{W}_{\mathbf{\Phi}}(\mathbf{\Lambda}_{\Phi})^{-1}\alpha^T\mathbf{K}(\mathbf{X}, s) \tag{10}$$

where $\mathbf{K}(\mathbf{X}, s) = [k(x_1, s), k(x_2, s), ..., k(x_n, s)]^T$, $k$ is a kernel function.

In the above iteration algorithm, the function $\mathbf{\Phi}$ is an implicit form. The kernel function *k* can be computed to instead of $\mathbf{\Phi}$. This trick is named as Kernel Trick. Many functions can be chosen for the kernel such as polynomial kernel:

$$k(x, s) = (x \cdot s)^d \tag{11}$$

Gaussian kernel $k(x,s)=\exp(-\|x-s\|^2/2\sigma^2)$ and sigmoid kernel $k(x,s) = \tanh(k(x \cdot s) + \theta)$. Liu and Cheng *et al* use a cosine kernel function [10, 11] derived from the polynomial kernel function as shown in Eq.(11), which can give a better performance than the polynomial kernel function for feature extraction:

$$\hat{k}(x, s) = \frac{k(x, s)}{\sqrt{k(x, x)k(s, s)}}, \tag{12}$$

where $k$ is a polynomial kernel. Practically speaking, Kernel-ICA = Kernel-Centering + Kernel-Whitening + ICA. Selecting an appropriate kernel function for a particular application area can be difficult and remains largely an unresolved issue. Any new kernel function derived from the kernel $k(x, s)$ with form $\hat{k}(x, s) = c(x)c(s)k(x, s)$, has been proved to be a valid kernel function when $c(x)$ is a positive real valued function of $x$, which is always satisfied. So, the cosine kernel is a valid kernel function. In this paper, we will adopt cosine kernel in our experiments.

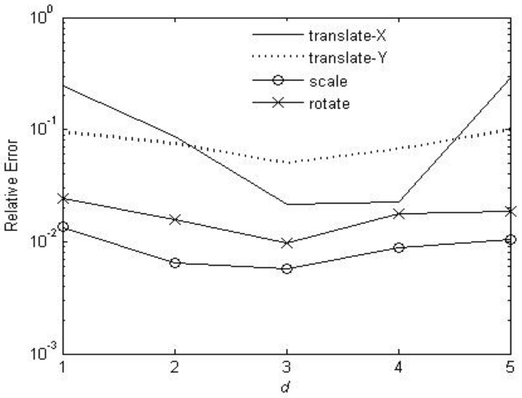


Fig. 3. Relative errors with respect to translation, scale and rotation vary with *d*.

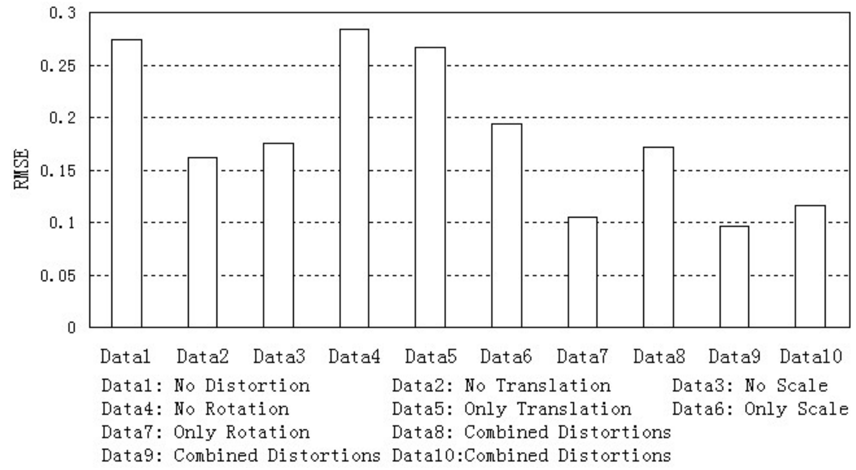
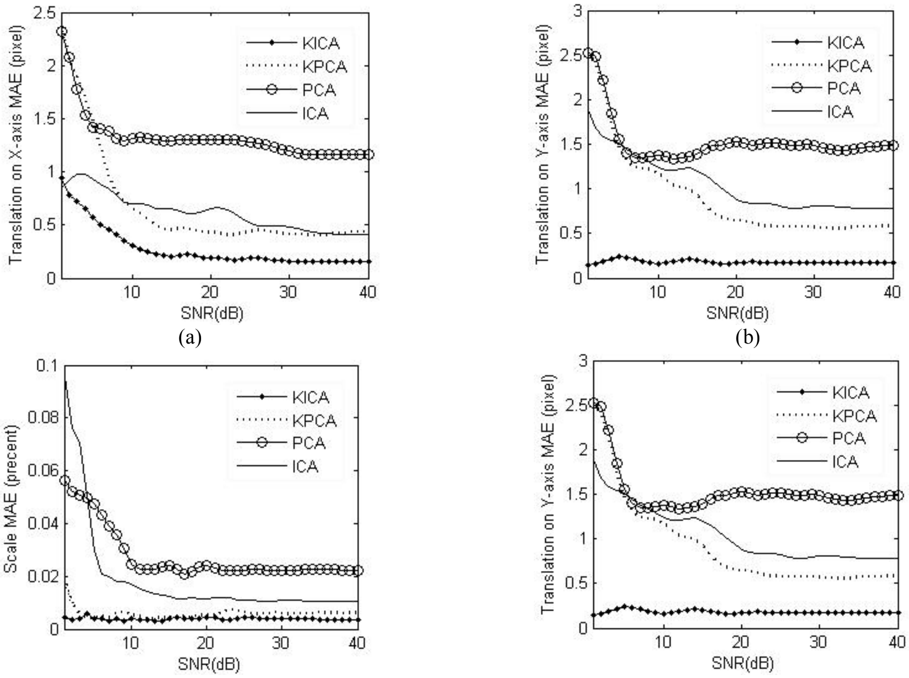


Fig. 4. RMSE between the registered image and the original image under 20 db SNR.



(a)

(b)

(c)

(d)

Fig. 5. Absolute registration error comparison with KICA, KPCA, PCA and ICA. (a) error with respect to translation on X-axis, (b) translation on Y-axis, (c) scale and (d) rotation errors.

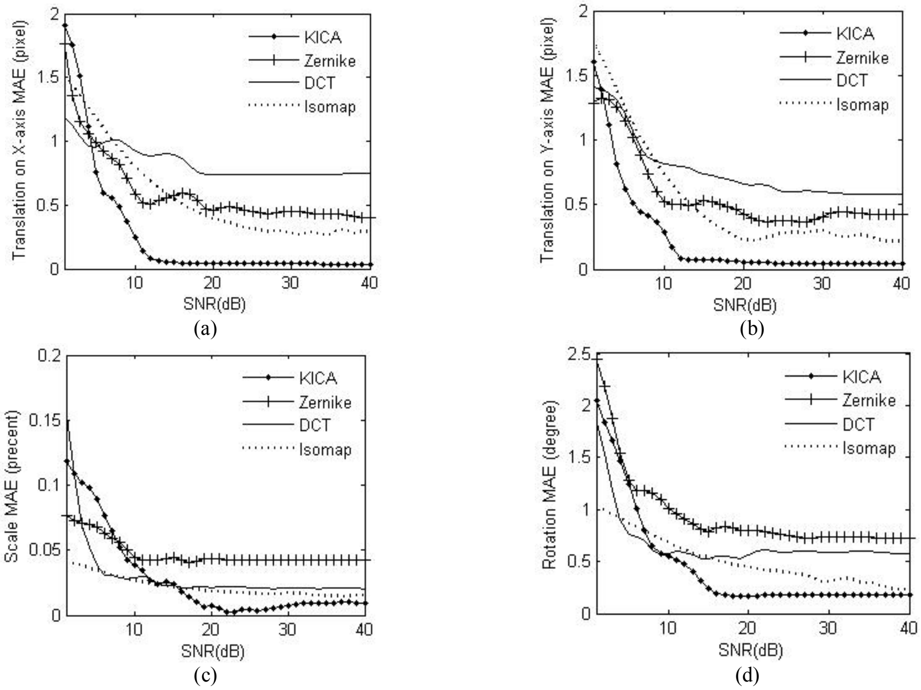
### *C. Image Registration Scheme with FNN*

The image registration scheme includes training the FNN to provide the required affine parameters. Each image in the training set is generated by applying an affine transformation. The affine parameters are randomly changed in a predefined range so as to reduce correlations among images. In order to improve the generalization and immunity of the FNN from over-sensitivity to distorted inputs, we introduce noise in the image synthesis. Then we employ KICA as a feature extraction mechanism presented to the FNN.

A good generalized FNN can be obtained with model selection or regularization techniques [12], in this work we intend to adopt model selection technique for generalization. Select a suitable FNN model can be implemented by designing the network structure with problem-dependent hidden neuron number. Because there is no theoretical guidance to decide the number of neurons in hidden layer [13] [14] and the low computational efficiency of some methods such as cross validation and bootstrap, for the FNN with three layer structure we can use the empirical formula [15] to compute the hidden neuron number:

$$p=\sqrt{0.43mn+0.12n^2+2.54m+0.77n+0.35}+0.5, \qquad (13)$$

where $p$ is the number of neurons in hidden layer, $m$ is the



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

Fig. 6. Absolute registration error comparison with KICA, Zernike, DCT and Isomap. (a) error with respect to translation on X-axis, (b) translation on Y-axis, (c) scale and (d) rotation errors.

number of input neurons and *n* is the number of output neurons. Here the structure of the FNN is that contains 60 inputs, 4 outputs and 17 hidden neurons. Sigmoid transfer functions are employed in the hidden layers while linear functions characterize the output-level neurons. The FNN is trained by using the Levenberg-Marquardt algorithm [16] based on gradient-descent learning technique.

## III. Experiments

### *A. Parameter Selection and Accuracy of the Proposed Method*

In the experiment, a pair of 256×256 resolution images was used. Fig. 2 shows one of the original images and a transformed image by the translation, rotation and scaling.

The training set consists of 300 images, each image is transformed from the reference image by translating, rotating and scaling randomly within a predefined range. Besides, additive Gaussian noise and Salt & Pepper type noise are applied on each image in various intensities. We also generate some test samples to demonstrate the registration accuracy of the proposed method. We apply KICA to the training samples and reduce the dimension of the sequence of vectors from 65536 to 60. These feature coefficients of images are inputs of FNN, the FNN is trained and its outputs are affine parameters. Finally, the feature coefficients are extracted from the registered image with the same method and fed as inputs into the trained neural network to get the estimated affine parameters.

Selecting an appropriate kernel function for a particular application area can be difficult and remains largely an unresolved issue. Those kernel functions given in section II have been explored in our experiments. We find that the cosine kernel function works best in our experiments. So, the cosine kernel function with various degrees *d* is discussed here. As shown in Fig. 3, it seems that $d$ = 3 is more appropriate for cosine kernel in our experiments.

Simulation results are shown in Table 1 as $d = 3$ and SNR = 15 db. The accuracy of parameter estimation can be also evaluated by root mean square error (RMSE) between the registered image and the original image, Fig. 4 shows the results under various test data sets under 20 db SNR. In the figure we can see that the untransformed images (Data 1 and Data 4) have the relative maximum error and the error of combined distortion images is usually low. The main reason is that the images in the training set are usually combined distortions and the generalization ability of the FNN is limited.

### *B. Comparison with Other Methods under Different Noisy Conditions*

In this experiment, we investigate the registration performance for the seven different feature extraction methods: KICA, ICA, PCA, KPCA DCT, Zernike moment and Isomap under different noisy conditions. In the proposed method, we choose $d = 3$ and each image is represented with a 60-dimentional feature vector. When we use PCA, KPCA and ICA based method, the dimension of input feature is also 60 in total. Same with the above experiment, a training set consisting of 300 images is synthesized. In order to evaluate registration performance with Gaussian noise, we take 40 images for each the evaluated SNR value. The test image is rotated 15 degree, 110% scaled, translated 2 pixels and -3 pixels on X-axis and Y-axis respectively, as shown in Fig.

TABLE I
DISTORTION PARAMETERS ESTIMATION UNDER 15 DB SNR

| Distortions Parameters | | | Registered Parameters |
|---|---|---|---|
| No Distortion | Translation | [0,0] | [0.14367, 0.050085] |
| | Scale | 1 | 1.0396 |
| | Rotation | 0 | 0.001 |
| No Translation | Translation | [0,0] | [-0.09011, 0.25429 ] |
| | Scale | 0.9 | 0.91493 |
| | Rotation | 21 | 21.253 |
| No Scale | Translation | [-2,1] | [-2.1276， 1.4066] |
| | Scale | 1 | 1.0167 |
| | Rotation | 3 | 3.0283 |
| No Rotation | Translation | [5,-5] | [4.674，-4.9044] |
| | Scale | 0.75 | 0.76998 |
| | Rotation | 0 | -0.2623 |
| Only Translation | Translation | [-5,5] | [-4.557，4.3894] |
| | Scale | 1 | 1.0399 |
| | Rotation | 0 | 0.4692 |
| Only Scale | Translation | [0,0] | [-0.1825，-0.1896] |
| | Scale | 1.25 | 1.2326 |
| | Rotation | 0 | 0.8346 |
| Only Rotation | Translation | [0,0] | [0.39705,0.16376] |
| | Scale | 1 | 1.0004 |
| | Rotation | 28 | 28.498 |
| Combined distortions | Translation | [-5,4] | [-4.9375，4.5481] |
| | Scale | 0.9 | 0.923 |
| | Rotation | 25 | 24.095 |
| Combined distortions | Translation | [5，-1] | [5.0092,-1.102] |
| | Scale | 1.134 | 1.1237 |
| | Rotation | 11.2 | 11.37 |
| Combined distortions | Translation | [-2，-3] | [-1.9296,-2.4307] |
| | Scale | 0.75 | 0.74787 |
| | Rotation | 16 | 16.197 |

2(a). Fig. 5 depicts the results of estimating the affine transform parameters under different SNR values by using the four methods. As can be seen from the results, our method shows more accurate than the other three methods. From Fig. 5 we also know that the performance of KPCA and KICA based methods are better than PCA and ICA based methods. Because PCA and ICA are linear methods in nature, these feature extraction methods are inadequate to describe the complex nonlinear variations.

Similarly, we compare our proposed method in this paper with the methods in reference [1], [2] and [3]. When we adopt DCT based method, the dimension of input feature is 64 in total. While Zernike and Isomap based method are implemented, the dimension of input feature is 57 and 10, respectively. We use test image "Lena" which is rotated 20 degree, scaled 80%, translated -4 pixels and 2 pixels on X-axis and Y-axis respectively. Fig. 6 described the results of estimating the affine transform parameters under different SNR values by using these four methods. As can be seen from the results, although the performance of the proposed method with respect to scale and rotation is not very satisfied as SNR is low, our method still shows more accurate than the other three methods especially when SNR is larger than 15 dB.

## IV. CONCLUSION AND FUTURE WORK

In this paper, a novel image registration scheme is proposed, which adopts KICA and the FNN to register affine transform parameters. Comparison experiments for feature extraction based image registration among PCA, ICA, KPCA, KICA, DCT, Zernike moment and Isomap are performed under different noisy conditions. It is shown that the proposed scheme has more accurate registration performance and robust to noise than the other methods. The experiment results suggest that KICA is a much efficient method to represent image registration feature.

The proposed method still deserves further study. First, there is not a systematic method for selection of the dimension of the input feature vectors and the parameter of degree *d* in the proposed scheme. Second, in the synthesizing a training set from the reference image, there is no proper image data due to rotation or scale. A feasible alternative to deal with second problem is to cut the border of the image, however, the information will be suffered for lost. Third, the generalization abilities of the FNN can be optimized in other methods such as regularization [17]. We will engage to find the solution for these problems in the further study.

## REFERENCES

[1] I. Elhanany, M. Sheinfeld, A. Beckl, Y. Kadmon, N. Tal, D. Tirosh , "Robust image registration based on feedforward neural networks," *Proceedings of IEEE International Conference on System, Man and Cybernetics*, vol. 1, pp. 1507-1511, 2000.

[2] J. Wu, J. Xie, "Image Registration Scheme Utilizing Feedforward Neural Networks," *Proceedings of the 5th World Congress on Intelligent Control and Automation*, vol. 5, pp. 4046-4048, 2004.

[3] A.B. Xu, P. Guo, "Isomap and Neural Networks based Image Registration Scheme", *International Symposium on Neural Networks* (ISNN 2006), *Lecture Notes in Computer Science*, Vol. 3972, pp. 486–491, Springer- Verlag, Berlin Heidelberg 2006.

[4] X.Y. Zeng, Y.W. Chen, Z. Nakao, H. Lu "A new texture feature based on PCA maps and its application to image retrieval," *IEICE Trans. Inf. and Syst.* pp. 929-936, 2003.

[5] K. I. Diamantaras, S.Y. Kung, *Principal Component Neural Networks: Theory and Applications*, John Wiley & Sons, INC. vol. 1, 1996.

[6] A.J. Bell and T.J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation*. vol. 7, pp. 1129-1159, 1995.

[7] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*. vol. 10, pp. 251-276, 1998.

[8] R. Bach Francis and Michael I. Jordan, "Kernel independent component analysis," *Journal of Machine Learning Research*. vol. 3, pp. 1-48, 2002.

[9] S. Harmeling, A. Ziehe, M. Kawanabe, K.R. Muller, "Nonlinear blind source separation using kernel feature spaces," in *Lee, T.W., ed.., Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation* 2001 (ICA2001), pp. 102-107.

[10] Q.S. Liu, J. Cheng, H. Lu, S. Ma, "Modeling Face Appearance with Kernel Independent Component Analysis," in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, vol. 2, pp. 761-766, 2004.

[11] J. Cheng, Q.S. Liu and H. Lu, "Texture Classification Using Kernel Independent Component Analysis," *Proceedings of 17th International Conference on Pattern Recognition*. vol. 1, pp. 620-623, 2004.

[12] P. Guo, "Studies of Model Selection and Regularization for Generalization in Neural Networks with Applications", PhD Thesis, the Chinese University of Hong Kong, 2001.

[13] P.M. Ravdin, G.M. Clark, S.G. Hilsenbeck, *et al*, "A demonstration that breast cancer recurrence can be predicted by neural network analysis," *Breast Cancer Res Treat*, vol. 21, no. 1, pp. 47-53, 1992.

[14] De M. Laurentiis, P.M. Ravdin, "Survival analysis of censored data: Neural network analysis detection of complex interaction between variables," *Breast Cancer Res Treat*, vol. 32, no. 1, pp. 113-118, 1994.

[15] H. White, "Connectionist Nonparametric Regression: Multilayer Feedforward Networks Can Learn Arbitrary Mapping," *Neural Networks*, vol. 3, no. 5, pp. 535-549, 1990.

[16] M.T. Hagan, M. Menhaj, "Training Feedforward Networks with Marquardt Algorithm," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 113-118, 1994.

[17] P. Guo, M. R. Lyu, C. L. Philip Chen, "Regularization Parameter Estimation for Feedforward Neural Networks," *IEEE Trans. Neural Networks*, vol. 33, no. 1, pp. 35-44, 2003.