Blind Source Separation with Evolution based KICA

Peng Wu1,2, Qian Yin1,* and Ping Guo1
1Image Processing and Pattern Recognition Laboratory, Beijing Normal University
Beijing 100875, China
2Shandong Provincial Key Laboratory of Network based Intelligent Computing, University of Jinan
Jinan 250022, China
Email: pengwiseujn@gmail.com, yinqian@bnu.edu.cn, pguo@ieee.org

Abstract—Kernel independent component analysis (KICA) has been widely used in the field of blind source separation. The selection of kernel function and its parameters plays an important role in KICA algorithm performance. An optimal kernel model should be rich enough to well map the given samples. However, users usually use a singular kernel based model in their experiments, which leads to a suboptimal kernel model. In order to solve this problem, we propose the evolution based multiple kernel independent component analysis (EMKICA), in which a convex combination of multiple base kernels is used instead of single kernel of KICA. The combination weights are learned by particle swarm optimization algorithm. Firstly, we elaborate the basic theory of KICA and concept of EMKICA, also the combination form of the composition kernel used in EMKICA. Secondly, we describe the presentation of the individuals in the particle swarm optimization algorithm, the settings of the evaluation function and general algorithm. Finally, we evaluate the separation ability of EMKICA on three different data sets including one-dimensional mixed signals, composite images and images with reflection. The experimental results verify the effectiveness of EMKICA.

I. INTRODUCTION

Independent component analysis (ICA) is an important technology in the field of signal processing. ICA is a simple and effective method for blind source separation and has improved considerably over the past several years [1]. ICA is based on the higher-order statistical properties, so original signals can be estimated, which are statistics independent and mixed by unknown factor, from the observed signals. Due to the ability of reflecting higher-order statistics characteristic of image data, ICA has successful application in many fields of image processing. ICA uses a single fixed nonlinear objective function, and always assumes that signal follows the Gaussian distribution. However, usually the probability distribution of a signal component is not known, so it has some limitations while dealing with complex signal. On the other hand, ICA is a linear algorithm, so it has a better separation effect if the observed signals are linear mixed by some signal components, as compared to the case of nonlinear mixed.

To solve the aforementioned problems, F. R. Bach et al. [2] proposed a kernel independent component analysis (KICA). The key idea of KICA is getting a higher-dimensional reproduced kernel Hilbert space (RKHS) by nonlinear mapping of the signals in lower-dimensional space. In RKHS, a nonlinear contrast function can be built and searches its minimum value. Compared with the traditional ICA, KICA is robust to noise, adapt to nonlinear mixed signal, and has an excellent performance in the case of signals with approximate Gaussian distribution. It plays an important role to affect the separation effect of KICA to select the kernel function and its parameters in RKHS. Usually, simple kernel is used by the rule of thumb, which is often time-consuming also inaccurate, which limits the further development of KICA. Huihua Yang et al. proposed a kernel model selection method to remove ECG artifact from the EEG signal [3], in which a known signal is added to the mixed signals and the final kernel model is selected by minimizing the known signal separation error. Lingli Jiang et al. proposed a kernel selection method on the basis of the similarity of source default signals and kernel independent component [4]. These methods are some primary trials to the kernel model selection in KICA, all of which only use one certain kernel function, such as Gaussian kernel, then find the optimal parameter, i.e., the bandwidth of Gaussian function. Each kernel functions represent one kind of mapping. When the samples are heterogeneous or non homologous, using only one kernel to map the whole of samples is inappropriate, then using the combination of multiple kernels to replace a single kernel is more reasonable, which is called multiple kernel learning (MKL) [5]. MKL has achieved better results than SVM with a single kernel in many applications [6] [7] [8] [9] [10]; therefore it is worthy to research how to utilize the ideas of MKL to make better performance of KICA. In this paper, we propose a multiple kernel version of KICA, called the evolution based multiple kernel independent component analysis (EMKICA), in which a convex combination of multiple base kernels is used instead of single kernel of KICA, where combination weights are learned by particle swarm optimization algorithm.

The rest of this paper is organized as follows: Section 2 introduces the basic theory of MKL and EMKICA; Section 3 describes in detail the presentation of the individuals in the particle swarm optimization algorithm, the setting of evaluation function and the general algorithm; the separating experiments and the related analysis on one-dimensional mixed signals, composite images and photos with reflection, are reported in Section 4; the conclusion and the further work are discussed in the final section.

II. BACKGROUND AND RELATED WORK

In this section, the basic theory of MKL and KICA algorithm are described.

A. Multiple Kernel Learning

MKL is a supervised learning process. Given a set of training samples \( D = \{x_i, y_i\} | 1 \leq i \leq n \), where \( x_i \in \mathbb{R} \) is the feature vector of sample and \( \mathbb{R} \) denotes a real space, \( y_i \) is...
the corresponding class label, \( n \) is the total number of samples, then the classical MKL can be formulated as an optimization problem as follow:

\[
\min_{\mathbf{K}} \min_{f \in \mathbf{H}_k} \lambda \| f \|_{\mathbf{H}_k} + \sum_{i=1}^{n} \ell(y_i, f(x_i)),
\]

where \( \ell \) denotes a loss function, e.g., hinge loss, \( \ell(t) = \max(0, 1 - t) \), \( \mathbf{H}_k \) is RKHS derived from a kernel function \( k \), \( \mathbf{K} \) is the optimization space of candidate kernels, \( \lambda \) is a regularization parameter. The optimization target of Eq. 1 is to find the optimal kernel function \( k \) in \( \mathbf{K} \) and the optimal function \( f \) in \( \mathbf{H}_k \), simultaneously. The final decision function is shown as follow:

\[
f(x) = \sum_{i=1}^{n} \beta_i k(x, x_i),
\]

where \( x \) denotes a sample of unknown type, \( x_i \) is a training sample, \( n \) is the total number of training samples, \( \beta_i \) is a weight. For more details of MKL, please refer to [5] [6] [7] [8].

### B. Principle of Kernel Independent Component Analysis

The key points of KICA include two aspects: object function and its optimization. Bach et al. [2] presented two kinds of contrast functions based on RKHS, i.e., kernel canonical correlation analysis (KCCA) and kernel generalized variance (KGV), and then they have used the contrast functions to build the corresponding objective functions. Both KCCA and KGA are tightly related to \( F - \)correlation. Considering the case of two univariate random variables \( x_1 \) and \( x_2 \), we get two Mercer kernels \( K_1 \) and \( K_2 \) as well as the corresponding feature spaces \( F_1 \) and \( F_2 \) after feature mapping by \( \Phi_1 \) and \( \Phi_2 \), the regularized \( F - \)correlation \( \rho_F^2 \) can be written as:

\[
\rho_F^2 = \max_{f_1, f_2 \in \mathcal{F}} \frac{\varphi(f_1(x_1) f_2(x_2))}{\| f_1 \|^2_\mathcal{F}} \beta, \quad \beta > 0,
\]

where \( \theta \) is a small positive constant. Expand \( \varphi(f_1(x_1) + \theta f_1) \) up to second order in \( \theta \), to obtain [2]:

\[
\varphi(f_1(x_1) + \theta f_1) \approx \frac{\alpha_1^T}{\alpha_0^T} \theta K_1 \alpha_1 + \frac{\alpha_0^T}{\alpha_1^T} \theta K_2 \alpha_0 \approx \frac{\alpha_0^T}{\alpha_1^T} (K_1 + \theta^2 K_2), \quad \theta \ll 1.
\]

The regularized KCCA problem can be written as [2]:

\[
\rho_F^2(K_1, K_2) = \max_{\alpha_0, \alpha_1 \in \mathbb{R}^N} \frac{\alpha_0^T K_1 \alpha_1}{\alpha_0^T K_2 \alpha_1},
\]

with its equivalent formulation as a generalized eigenvalue problem as follow [2]:

\[
\begin{pmatrix}
K_1 & K_2 \\
K_2 & K_1
\end{pmatrix}
\begin{pmatrix}
\alpha_0 \\
\alpha_1
\end{pmatrix} = \lambda
\begin{pmatrix}
\alpha_0 \\
\alpha_1
\end{pmatrix}
\]

For more than two variables, the \( F - \)correlation can be obtained by solving the following generalized eigenvalue problem [2]:

\[
\begin{pmatrix}
(K_1 + \Delta_1 I)^2 & K_2 K_1 & \cdots & K_2 K_m \\
K_2 K_1 & (K_2 + \Delta_2 I)^2 & \cdots & \cdots \\
\vdots & \vdots & \ddots & \vdots \\
K_m K_1 & K_m K_2 & \cdots & (K_m + \Delta_m I)^2
\end{pmatrix}
\begin{pmatrix}
\alpha_1 \\
\alpha_2 \\
\vdots \\
\alpha_m
\end{pmatrix} = \lambda
\begin{pmatrix}
\alpha_1 \\
\alpha_2 \\
\vdots \\
\alpha_m
\end{pmatrix}
\]

For simplicity, the above equation is usually written as \( \mathbf{K}_\theta \alpha = \lambda \mathbf{D}_\theta \alpha \), where both \( \mathbf{K}_\theta \) and \( \mathbf{D}_\theta \) are \( mN \times mN \) matrices. The minimal eigenvalue of this equation is referred to as the first kernel canonical correlation and denoted by \( \lambda_1^\theta(K_1, K_2, \ldots, K_m) \). The contrast function based on KCCA can be formulated as [2]:

\[
C(W) = I_{K \Phi}(K_1, K_2, \ldots, K_m) = -\frac{1}{2} \log \lambda_1^\theta(K_1, K_2, \ldots, K_m).
\]

The contrast function based on KGV is [2]:

\[
C(W) = I_{K \Phi}(K_1, K_2, \ldots, K_m) = -\frac{1}{2} \log \delta_2^\theta(K_1, K_2, \ldots, K_m),
\]

where \( \delta_2^\theta(K_1, K_2, \ldots, K_m) = \det \mathbf{K}_0 / \det \mathbf{D}_0 \).

The description of KICA algorithm is as follow:

- Input kernel function \( k(x, y) \) and observed signals \( \mathbf{X} = [x_1, x_2, \ldots, x_m] \);
- Whiten the observed signals \( \mathbf{X} \), i.e., \( \hat{\mathbf{X}} = \mathbf{P} \mathbf{X} \), where \( \mathbf{P} \) is a whiten matrix;
- Minimize the value of \( C(W) \) by gradient-descending until convergence, where the approach to calculating the value of \( C(W) \) is as follow: firstly, estimate the source signals using \( \mathbf{S} = \mathbf{W} \mathbf{X} \); secondly, calculate the centered Gram matrices, i.e., \( \mathbf{K}_1, \mathbf{K}_2, \ldots, \mathbf{K}_m \); thirdly, get the minimal eigenvalue by solving (7), and finally, obtain the value of \( C(W) \);
- Calculate the estimated source signals \( \mathbf{S} = \mathbf{W} \mathbf{X} = \mathbf{W} \mathbf{X} \), where \( \mathbf{W} \) obtained in the former step is the demixing matrix.

### III. EVOLUTION BASED MULTIPLE KERNEL INDEPENDENT COMPONENT ANALYSIS

#### A. Expression of Composite Kernel

Let \( \mathbf{K}_\text{em} \) be the composite kernel in EMKICA, which is a linear combination of several predefined base kernels. Considering \( m \) predefined base kernels, i.e., \( \mathbf{K}_1, \mathbf{K}_2, \ldots, \mathbf{K}_m \), \( \mathbf{K}_\text{em} \) can be computed as:

\[
\mathbf{K}_\text{em} = \sum_{i=1}^{m} \beta_i \mathbf{K}_i = \{k_{\text{em}}(\cdot, \cdot) = \sum_{i=1}^{m} \beta_i k_i(\cdot, \cdot)\},
\]

where \( k_i(\cdot, \cdot) \) denotes the entry of kernel matrix \( \mathbf{K}_i \), \( \beta_i \) is the weight and subject to \( \sum_{i=1}^{m} \beta_i = 1 \). For the base kernel functions, the common used types of functions are as follow:

\[
k_{\text{linear}}(x_i, x_j) = \langle x_i, x_j \rangle, \quad k_{\text{poly}}(x_i, x_j) = (\langle x_i, x_j \rangle + 1)^d, \quad k_{\text{gaussian}}(x_i, x_j) = \exp(-\frac{\| x_i - x_j \|^2}{2\delta^2}),
\]

where \( d \) is the degree of polynomial, \( \delta \) is the width of Gaussian kernel.

#### B. General Algorithm for Evolving Kernel in EMKICA

This section introduces how to optimize the combination coefficients and the related parameters of the base kernels to enhance the performance of EMKICA. We presented the issue by solving the global optimization problem by designing a specific object function. Due to the simplicity and it’s easy to be parallelization, PSO [11] algorithm has gain great success in the global optimization problem, which is utilized to optimize the related parameters of EMKICA.
1) Expression of Particles: The parameters include the combination coefficients and the related parameters of the given base kernels. We treat these parameters as the decisive variables of PSO algorithm. The number of variable depends on the specific type of kernel function. Three different kernel functions are used in our experiments, i.e., linear, Gaussian and polynomial function. One particle is a vector as, \{d, \delta, \beta_1, \beta_2, \beta_3\}, where d is the degree of polynomial and its value, ranging from 1.0 to 5.0, is rounded during the computation for the sake of validly computing; \delta is the width of Gaussian kernel and its value ranges from $10^{-2}$ to $10^2$, $\beta_1, \beta_2$ and $\beta_3$ are the weights of linear, Gaussian and polynomial kernel, respectively.

2) Parameter of Objective Function: The parameter of the objective function guides the evolving process of PSO algorithm, objective function is also known as the fitness function. In order to evolve the optimization composition kernel for EMKICA, we need to transform the original unsupervised problem to a supervised problem by designing a specific fitness function. In [3], in which a known signal is added into the observed function. We have adopted the same fitness function used in [3], in which a known signal is added into the observed signals, and all of them are separated by EMKICA. Due to the known signal, the obtained demixing matrix \( W \) has some special characteristic. Huihua Yang et al. [3] proved that if KICA can correctly separate the addition known signal and the special characteristic. Therefore, two evaluation functions are given as follows [3]:

\[
E^1(W) = \frac{1}{2m|a_0|} \left( \sum_{i=1}^{m}|a_i| + \sum_{i=1}^{m}|b_i| \right) = \frac{1}{2m|a_0|} \left( \sum_{i=1}^{m}|a_i| + \sum_{i=1}^{m}|b_i| \right), \quad (14)
\]

\[
E^2(W) = \frac{1}{2m|a_0|} \left( \sum_{i=1}^{m}a_i^2 + \sum_{i=1}^{m}b_i^2 \right) = \frac{1}{2m|a_0|} \left( \sum_{i=1}^{m}a_i^2 + \sum_{i=1}^{m}b_i^2 \right), \quad (15)
\]

where \( a_i \) and \( b_i \) denote the other elements located in the same column or row as \( a_0 \). The physical significance of Eqs. 14 and 15 are the relative errors in amplitude and energy between the observed signals and the additional one, respectively. We use Eq. 14 as the fitness function for the PSO algorithm in EMKICA.

3) General Algorithm: The general algorithm of EMKICA can be described as follows:

1) Input the samples \( X \) and define three kinds of kernel functions, i.e., linear, Gaussian and polynomial kernel.
2) Initial the global parameters in PSO including: the maximum iteration number, the size of particles, the inertia weight of velocity update \( V_w \), the control parameter of local searching ability \( C_1 \), the control parameter of global searching ability \( C_2 \), the range of each variable.
3) Generate randomly the velocity vector \( v \) and the position vector \( p \) of each particle.
4) Calculate the fitness value of each particle as follows: firstly, calculate \( X \)'s three kernel matrices, i.e., \( K_1, K_2, K_3 \), according to the kernel functions defined in Step 1; secondly, using the position vector \( p \) and Eq. 10 to calculate \( K_{em} \); thirdly, obtain the demixing matrix \( W \) by KICA with \( K_{em} \) and substitute the value of \( W \) into Eq. 14 for calculating the fitness value of each particle.

5) Update each particle’s personal best value \( P_{best} \) and global best value \( G_{best} \).
6) Update each particle’s velocity and position as follows:

\[
v_t = V_w \times v_t + C_1 \times \text{rand} \times (P_{best} - p_t) + C_2 \times \text{rand} \times (G_{best} - p_t) \quad (16)
\]

\[
p_t = p_t + v_t \quad (17)
\]

7) Stop the algorithm if the predefined condition is reached; otherwise, jump into Step 4.

IV. EXPERIMENTAL STUDIES

To validate the performance of EMKICA, we have conducted experiments in two groups, which include separating one dimensional mixed signals and two-dimensional signals. To compare algorithm performance, we conducted the same experiments using the canonical KICA [2]. In KICA, Gaussian kernel is adopted and the values of two necessary parameters, i.e., the regularization parameter \( k \) and the width of Gaussian function, are set to 0.02 and 1, respectively [2]. As for EMKICA, the value of \( k \) uses the same setting as KICA and the parameters related to PSO are initialized according to [11]. Table I lists the settings of EMKICA. All experiments run on a PC of Intel core i7-3610M 2.3GHz CPU, 4 GB DDR, Windows 7 OS.

A. Separating one-dimensional signals
To get the mixed signals, we used the following method: firstly, construct a one-dimensional signal matrix \( X = [x_1; x_2; x_3] \), where \( x_1, x_2 \) and \( x_3 \) are a random signal, a sine signal and a stochastic signal with exponential distribution, respectively. All of them have the same length of 100; secondly, generate randomly a mix matrix \( A = [-0.5505 \ 0.7148 \ 0.4312; -0.7473 \ -0.1917 \ -0.6362; -0.3721 \ -0.6725 \ 0.6397]; \) thirdly, calculate the mixed signals using \( \hat{X} = AX \).

An additional known signal is necessary for EMKICA, so a square signal whose value has only two options to insure less computational work will be added when EMKICA separates the additional signal from the observed signals. Figure 1 shows the related information of this experiment. The first row of Figure 1 depicts the three source signals denoted by Source1, Source2 and Source3; the second row shows the three mixed signals denoted by Mixture1, Mixture2, Mixture3, and the additional signal denoted by Addition; the fourth and the fifth rows are the estimated signals by KICA and EMKICA, respectively.

As shown in Figure 1, both EMKICA and KICA can correctly separate the mixed signals. In order to compare the difference of the two methods, Tables II and III list the correlation coefficients between the source and the estimated signals. As we can see from Tables II and III, EMKICA

<table>
<thead>
<tr>
<th>Source signal</th>
<th>Estimated signal</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source1</td>
<td>Estimated source2</td>
<td>0.908</td>
</tr>
<tr>
<td>Source2</td>
<td>Estimated source1</td>
<td>-0.9985</td>
</tr>
<tr>
<td>Source3</td>
<td>Estimated source3</td>
<td>0.9681</td>
</tr>
</tbody>
</table>

TABLE II. THE CORRELATION COEFFICIENTS BETWEEN THE SOURCE AND THE ESTIMATED SIGNALS SEPARATED BY KICA.
enhances the correlation coefficients by 0.0879, 0.0004 and 0.0250, compared to KICA. The experimental results suggest that EMKICA can find an optimal composition kernel through an evolving procedure and guarantee a better separating effect. Additionally, we compare the computation cost of the two algorithms. It takes 0.9047 and 49.0061 seconds to run KICA and EMKICA, respectively. This suggests that EMKICA needs an evolving procedure and guarantee a better separating effect.

Additionally, we compare the computation cost of the two algorithms. It takes 0.9047 and 49.0061 seconds to run KICA and EMKICA, respectively. This suggests that EMKICA needs an evolving procedure and guarantee a better separating effect.

B. Separating two-dimensional signals

To separate two-dimensional signals such as images, we can transform them to one-dimensional signals. Considering the resolution of an image is $M \times N$, we can get a one-dimensional signal with the length of $M \times N$ after progressive scanning.

1) Separating Mixed Nature Images: Three images have been selected with the resolution of $256 \times 256$, and transformed into three one-dimensional signals with the length of 131072 denoted by $Y = [y_1; y_2; y_3]$. A random mix matrix is generated as $A = [0.8671, 0.1239, 0.9824; 0.2218, 0.9217, 0.7845; 0.7653, 0.5432, 0.022]$, and calculated the mixed signals using $\hat{Y} = AY$.

In Figure 2, the first row denotes three source images, and the third ones in the second row are the mixed images. The third and forth row of Figure 2 display the separated images using KICA and EMKICA, respectively. It is worth noting that the last image in the second row is an additional noise image which is only involved with the separating procedure of EMKICA. We have conducted the experiment three times and reported only one result of them due to the limitation of space. From Figure 2 we can see both KICA and EMKICA can correctly separate the mixed images, but EMKICA outperform KICA in the term of RMSE, which is defined as follow:

$$RMSE = \sqrt{\frac{1}{M \times N} \sum_{1 \leq x \leq M, 1 \leq y \leq N} (p_s(x, y) - p_e(x, y))^2},$$

where $M, N, p_s(x, y)$ and $p_e(x, y)$ denote the row index, the column index, the gray value of a pixel of a source image and the gray value of the pixel at the same position of an estimated image, respectively.

Table IV lists the mean RMSE of three running experiments, from which we can see the RMSE between the source and the estimated images dropped by 0.1746, 0.2562 and 0.3908 on three images, i.e., “Lena”, “Nvsheng” and “Cameraman”, respectively, when compare the performance of EMKICA to KICA. These results suggest that kernel model evolved by EMKICA is better than one single kernel model used by KICA.

2) Separating Relection in Photo: When we take a picture of an object in our daily life, it has some incidental variations.
such as a reflection of an unrelated object. As shown in Figure 3, we want to take a picture of Fuwa in a frame, however, the picture contains the Fuwa and the reflection of a bottle in front of the frame due to the reflecting action of glass. We consider the Fuwa as intrinsic aspect of the photo, the reflection of the bottle as incidental, and the combination of incidental and intrinsic components can often be approximated as a linear mixing process. Some researchers used the statistical tool of KICA to separating these components [12] [13]. Here, we test the ability of EMKICA on separating reflection from a picture. In this experiment, we photographed the Fuwa framed behind glass with the reflection of a bottle filled with colorful candies. The Fuwa with reflection was photographed twice through a linear polarizer with different orientation, in the angle between the picture and camera view direction was about 90 degrees. We tailored the pictures into size of $250 \times 386$ and treated them as the separating objects, shown as Figures 4.(a),(b). Figures 4.(e),(f) are the difference between the separated Fuwa image by KICA (Figure 4.(c)) and EMKICA (Figure 4.(d)) and the completely separated Fuwa image by polarizing filters. As can be seen from these results, EMKICA is more efficient than KICA.

V. CONCLUSION

As an efficient algorithm, KICA has been applied for the blind source separation with great success. However, selection of an optimal kernel model is a difficult task. In this paper, an evolution based multiple kernel ICA is proposed, in which a composition kernel is learned by PSO algorithm and used instead of single kernel of KICA. Experimental results validate the performance of proposed method. However, the computational time of EMKICA is higher than KICA due to the evolving procedure. In future we would like to reduce the computational cost of EMKICA.

ACKNOWLEDGEMENTS

The research work in this paper was supported by the grants from National Natural Science Foundation of China (Project No. 61375045 and 61472043), Beijing Natural Science Foundation (4142030) and Natural Science Foundation of Shandong Province (ZR2013FL002). Professor Qian Yin (Email: yinqian@bnu.edu.cn) is the corresponding author of this paper.

REFERENCES