Fusion Based FastICA Method: Facial Expression Recognition

Humayra B. Ali and David M W Powers
Computer Science, Engineering and Mathematics School, Flinders University, Australia
Email: {ali0041, david.powers}@flinders.edu.au

***Abstract*—With the continuous progress of human computer interaction, face detection as well as facial expression recognition is gaining the attention of researchers from the fields of security, psychology, image processing, and computer vision. In this area the most challenging thing is to recognize accurate facial expression with minimum time requirement. In this work, our main focus is to minimize the time using fusion based Independent Component Analysis (ICA). Research studies show ICA has significant success on face image analysis. Among several architectures of ICA we mainly used here Gaussian kernel based FastICA algorithm due to time efficiency. We apply FastICA on whole faces to recognize facial expressions. Also we apply FastICA on different facial parts, by proposing two algorithms namely WAPA-FastICA and OEPA-FastICA, to analyze the influence of different parts for several basic emotions. Our experiment shows OEPA-FastICA and WAPA-FastICA outperforms existing predominant FastICA algorithm. We also compared these proposed algorithms with our previous PCA based facial expression recognition work.**

***Index Terms*—OEPA: optimal expression specific parts accumulation, WAPA: weighted all parts accumulation algorithm, ICA: independent component analysis, FER: facial expression recognition, LS-ICA: locally salient ICA**

I. Introduction

In this work our main target is to increase the correct recognition rate of facial expression and at the same time decrease the recognition time. As ICA works for higher order derivatives, and facial expression data in images lie on high dimensional data matrix, so ICA is a good choice for facial expression recognition. Among different architectures of ICA algorithm, we use here FastICA algorithm for time efficiency. Our first contribution here is to implement several kernels, like Tangent, Gaussian and Cubic for FastICA and compare among them for independent component extraction. We find FastICA with Gaussian kernel is more efficient. Our second contribution is we propose two algorithms namely WAPA-FastICA and OEPA-FastICA for part based facial expression recognition. These algorithms are discussed in corresponding section. We also implement LS-ICA because of its robustness to partial occlusion and local distortion. For LS-ICA 8x8 window is applied to perform scanning process and finding ICs. Our main target is to analyze the influence of different parts for facial expression recognition.

II. Related Work

The work from [1] claims that the structural information of sensory inputs stays in the redundancy if the sensory input system. PCA and ICA are the most well known methods for redundancy as well as finding useful components for attaining distinguishable properties. This redundancy provides information to develop a factorial system and independent components (ICs) develop from this representation. Complex object of higher order dependencies need such representation to be encoded. Independent component representation from this sort of redundant data is a general coding strategy for the visual systems [2].

The most prominent subspace learning algorithms are PCA, ICA, NMF, LDA etc. For feature extraction from the facial expression images, most of the early FER research works extracted useful features using Principal Component Analysis (PCA). PCA is a second-order statistical method, which creates the orthogonal bases. These orthogonal bases provide the maximum variability. These variables are good source fro distinguishing features in image analysis. It is also commonly used for dimension reduction. [3] and [4] employed PCA as one of the feature extractors to solve FER with the Facial Action Coding System (FACS). We previously investigated PCA on facial expression recognition [5] and [6]. The work in [5] and [6] shows applying PCA on face parts rather whole face gives more accuracy to recognize expressions from facial image data.

In this work, we investigate ICA on facial data to recognize basic facial expressions. Independent Component Analysis (ICA) has the ability to extract local facial features [7], [8]. In recent years ICA has been extensively utilized for FER [7]-[9]. As much of the information that distinguishes different facial expressions stays in the higher order statistics of the images [9], ICA is a better choice for FER than PCA. In [10], Bartlett *et al.* implemented ICA to recognize twelve different facial expressions referred to FACS. In [11], Chao-Fa and Shin utilized ICA to extract the IC features of facial expression images to recognize the Action Units (AU) in the whole face as well as the lower and upper part of the faces. In [8], Buciu *et al.* proposed ICA for the emotion-specified facial expression recognition and applied ICA on the

Manuscript received January 11, 2013; revised May 7, 2014.

©2014 Engineering and Technology Publishing

doi: 10.12720/joig.2.1.1-7

Japanese female facial expression database [12]. In [13], Bartlett *et al.* again introduced ICA on the PCs for face recognition in two different architectures where the first architecture finds the spatially local basis images and the second one the factorial face codes.

III. Independent Component Analysis

The principal component analysis (PCA) performs by the Karhunen-Loeve transformation and produces mutually uncorrelated features y (i), i=0,1…N. When the goal of image or data processing is to reduce the dimension as well as to minimize the mean-square error, Karhunen-Loeve transformation is a good solution. However, certain applications, such as the one depicted in Figure 1, the mathematical or analytical solution does not meet the expectations. In addition the more recently development of Independent Component Analysis (ICA) algorithm seeks to achieve much more complicated features than simple decorrelation of data or image analysis. Then ICA task is defined as follows: Given the set of input samples x, determine an M x M invertible matrix W such that the entries y (i), i=0, 1…M-1, of the transformed-vector are mutually independent. The uncorrelatedness required by PCA is less important feature than the statistical independence which independence can be implemented by ICA algorithms. Only for Gaussian random variables, this two conditions are equivalent to each other.

$$y = Wx \quad (1)$$

Seeking for statistical independence of data gives the mean of exploiting a lot more information, which lies in the higher order statistics of the data.

Before developing the techniques for performing ICA, we need be confident that such as a solution and under specific conditions. For this we have to assume that our input random data vector x is principally generated by a linear combination of statistically independent sources, which is.

$$x = Ay \quad (2)$$

Now the task is to exploit the information of x to define under what conditions the matrix W can be computed so as to recover the components of y from equation of (2). Usually A is mixing matrix and W is the demixing matrix.

The first condition is all independent components y (i), i=1,2…N, must be non-Gaussian and the second condition is that matrix A must be invertible. It means, in contrast to PCA, ICA is meaningful only if the random variables are non-Gaussian. And mathematically for Gaussian random variables, statistical independence is equivalent to the uncorrelatedness nature of PCA. So we have to assume that the obtained Independent Components y (i),i=1,2…N, are all Gaussian, then by using any unitary matrix, a linear transformation will be a solution [7]. On the other hand, by imposing a specific orthogonal structure onto this transformation matrix, PCA achieves a unique solution.

IV. Problem Specification and Possible Solution

Here, in the following figures, Fig. A should match with Fig. B as they both are of same facial expression, anger (from CK dataset). But when we apply ICA on whole faces, our system shows close match with Fig. C. This is because same person's faces are in both train and test dataset. In this situation, ICA decomposition finds a close match with same person than similar expression. It may happen for same hair style, same cheek size or same face pattern (fat or this face). This situation happens in two cases: firstly, when same or similar test face (same person's face) is present in train image set as well as similar (not exactly same) test expression is in the train image set. For this reason, the overall recognition rate decreases.

Figure 1. Problem specification

To overcome this situation we focus on only facial parts like, eye, nose and mouth. These parts are more prone for expressing any facial emotion. Our part based system (WAPA and OEPA-FastICA) shows better performance than whole face FastICA. The general scheme of the method is shown in figure.

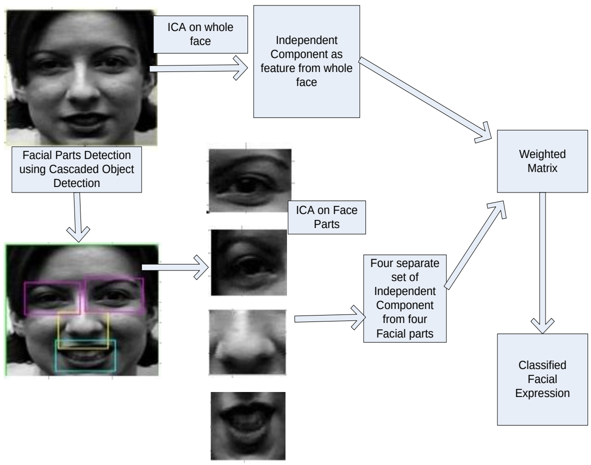

Figure 2. Proposed solution

V. ICA Algorithm

A. ICA by Maximization of Non Gaussianity

Non-Gaussian components are Independent [1]. Maximization of non-Gaussianity is the simple principles for estimating ICA model. Under certain conditions, the distribution of a sum of independent random variables tends towards a Gaussian distribution, which is the concept of central limit theorem. By finding the right linear combinations of the mixture variables, independent

©2014 Engineering and Technology Publishing

components can be estimated. The mixing can be inverted as,

$$S = A^{-1}X \tag{3}$$

Now it becomes,

$$Y = b^T X = b^T AS \tag{4}$$

Here b stand for one of the rows A^{-1}, but this linear combination $b^T X$ actually stands for one of the independent components. But as we have no knowledge of matrix A, we cannot determine such ‘b’, although we can find an approximate estimator. Non- Guassianity has two different practical implementations.

Kurtosis: Kurtosis or the fourth order cumulant is the classical measure of non-gaussianity. It is stated by

$$Kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 \tag{5}$$

We should assume variable y to be standardized, so it can say

$$Kurt(y) = E\{y^4\} - 3 \tag{6}$$

The normalized version of the fourth moment $E\{y^4\}$ is defined as Kurtosis. The fourth moment is equal to 3 For the Gaussian case implementation and hence $Kurt(y) = 0$. So it means for the gaussian variable kurtosis is zero but it is non-zero for the nongaussian random variables.

Hence the kurtosis is simply a normalized version of the fourth moment. For the Gaussian case the fourth moment is equal to 3 and hence $Kurt(y) = 0$. Thus for gaussian variable kurtosis is zero but for nongaussian random variable it is non-zero.

Negentropy: An alternative very important measure of nonguassianity is negentropy. From mathematical analysis, the measure of non-gaussianity is zero for a Gaussian variable and always non negative for a non Gaussian random variable. We can use a slightly modified version of the definition of differential entropy as negentropy. Negentropy J is defined as

$$J(y) = H(y_{gauss}) - H(y) \tag{7}$$

where y_{gauss} is a Gaussian random variable of the same covariance matrix.

B. Negentropy in Terms of Kurtosis

The largest entropy among all the random variables is the Gaussian variable. The negentropy for the random variables is zero if and only if it is a Gaussian variable, otherwise it is always positve. Moreover, the negentropy has an extra property that it is invariant for invertible transformation. But the estimation of negentropy is difficult, as it requires an estimate of the probability density function. Therefore in practice using higher order moments approximates negentropy.

$$J(y) \approx \frac{1}{12}E\{y^3\}^2 + \frac{1}{48}kurt(y)^2 \tag{8}$$

Another approach is to generalize the higher order cumulant approximation in order to increase the robustness. Again the random variable y is assumed to be a standard variable. So that it uses expectations of general non-quadratic functions. To make it simple, we can take any two nonquadratic functions G1 & G2 such that G1 is odd & G2 is even & we obtain the following approximation [equation 9].

$$J(y) \approx K_1(EG_1(y))^2 + K_2(EG_2(y)) - EG_2(U)^2 \tag{9}$$

where K1 & K2 are positive constant & U is standardized Gaussian variable.

VI. Fast Fixed Point Algorithm for ICA (FastICA)

Assume that we have a collection of prewhitened random vector x. Using the derivation of the preceding section, the following steps show the fast fixed point algorithm for ICA.

1. Take a random initial vector w(0) of norm 1.Let $k=1$.
2. Let. $w(k) = E\{x(w(k-1)^T x)^3\} - 3w(k-1)$ Byusing a large sample of x vectors, the expectation can be estimated.
3. Divide $w(k)$ by its norm.
4. If $|w(k)^T w(k-1)|$ is not close enough to 1,let $k = k+1$ and go back to step 2. Otherwise, output the vector $w(k)$.

The final vector $w(k)$ stands for one of columns of the (orthogonal) mixing matrix [B]. This means that $w(k)$ separates one of the non-Gaussian source signals. The most dominant property of this algorithm is that it requires a very small number of iterations; normally 5-10 iterations seem to be enough to obtain the maximal accuracy allowed by the sample data set. This is due to the cubic convergence property of the algorithm.

A. Performance Index

A well-known formula for measuring the separation performance is Performance Index (PI) which is defined as

$$PI(A) = \frac{1}{m(n-1)} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(\frac{[A]_{i,j}}{\max_j [A]_{i,j}} - 1\right) \tag{10}$$

where $[A]_{i,j}$ is the (i,j)th-element of the matrix [A]. As because, the knowledge of the mixing matrix [A] is required, the smaller value of PI usually mentions the better performance evaluation for experimental settings.

B. Proposed Algorithm

As pre-processing we did differentiation on whole set of images. Then we applied ICA on whole faces and on different facial parts, which hold the prominent

©2014 Engineering and Technology Publishing

characteristics for expression detection. In [5 & 6], we implemented PCA on whole and part based faces. In this paper we implemented ICA incorporating (i) Weighted All Parts Accumulation (WAPA) algorithm and (ii) Optimal Expression-specific Parts Accumulation (OEPA). We consider four parts left eye, right eye, mouth and nose and we found better results.

First Approach: In first option, we consider all the four parts to train an expression and utilize their weighted value in order to identify the expression from the test data.

Weighted All Parts Accumulation (WAPA) algorithm

Step: Apply the relevant equation to identify an expression

--Ehap = W1.LEhap + W2.REhap + W3.Nhap + W4.Mhap

--Eang = W1.LEang + W2.REang + W3.Nang + W4.Mang

--Edis = W1.LEdis + W2.REdis + W3.Ndis + W4.Mdis

--Efear = W1.LEfear + W2.REfear + W3.Nfear + W4.Mfear

--Esur = W1.LEsur + W2.REsur + W3.Nsur + W4.Msur

--Esad = W1.LEsad + W2.REsad + W3.Nsad + W4.Msad

LE=Left eye, RE=Right Eye, M=Mouth, N=Nose In these equations, we calculate the weighted average of the four parts of faces, eye (left, right), nose and mouth.

Second Approach:

Sometimes a subset of all the four parts of the face is optimal in terms of processing time and accuracy for identifying an expression. In second approach, we adapt similar approach and named it as Optimal Expression-specific Parts Accumulation (OEPA). In case of identifying an expression, if more than one subset of four parts give almost equal accuracy within a threshold value, this algorithm picks the subset of minimal number of parts in order to reduce the processing time. It increases the efficiency of the program in terms of time and as well as accuracy.

Algorithm of Optimal Expression-specific Parts Accumulation (OEPA)

Initialization: First we initialize the random population.

Evaluation:

Step 1: Assume I = [i1, i2, i3, i4] is the vector of different segments of facial region, like: both eyes, mouth and nose.

Step 2: Evaluate fitness f (I (i)) representing the accuracy of detection based on a particular instance of I (i), where i = 1to 4.

Step 3: Assume E = [e1, e2, e3, e4, e5, e6] is the vector for six basic emotions, like: happy, sad, disgust, anger, fear and surprise. For each expression E (j), we need to run step 4 and step 5, where j = 1to 6.

Step 4: Assume P = I (i) for each region, where i = 1to4 and K1 = max (f (P)), accuracy value for detection of expression E(j).

K2 = max (f (P1 + P2)), accuracy value for detection of region P1 and P2 for the expression E(j), where P1 ≠ P2.
K3 = max (f (P1 + P2 + P3) =accuracy value for detection of P1, P2 and P3 expression where P1 ≠ P2 ≠ P3.
K4=max (f(P1+P2+P3+P4)=accuracy value for detection of P1,P2,P3 and P4 expression where P1 ≠ P2 ≠ P3 ≠ P4.

Step 5: Get the facial region $\sum P_i$, for which L = max (Ki)| here L is the final accuracy value for the particular expression.

VII. LOCALLY SALIENT ICA

The LS-ICA method imposes additional localization constraint in the process of the kurtosis maximization. Thus it creates component based local basis images. Each iterative solution step is weighted. And it becomes sparser by only emphasizing larger pixel values. This sparsification contributes to localization. Let *V* be a solution vector at an iteration step, we can define a weighted solution, *W* where $W_i = |(V_i)|^{\beta} V_i$ and *W* = *W* /‖ *W* ‖. β > 1 is a small constant. The kurtosis is maximized in terms of *W* (weighted solution) instead of *V* as in equation 11.

$$Kurt(w) = \left| E(W)^4 - 3(E(b)^2)^2 \right| \tag{11}$$

By applying equation 12, which is a update rule we can derive a solution for equation 11.

$$S^{t+1} = E|V_i|^{\beta} Q(|V_i|^{\beta} S^T(t)Q)^3 \tag{12}$$

Here (in equation 12) *S* is a separating vector and *Q* contains whitened image matrix. Then the resulting basis image is $W^i = |V_i|^{\beta}(S^T Q)_i$. Then according to the algorithm LS- ICA basis images are created from the ICA basis images selected in the decreasing order of the class separability P [11], where $P = \Omega_{between} / \Omega_{within}$. Thus both dimensional reduction and good recognition performance can be achieved. The output basis images contain localized facial features that are discriminant for change of facial expression recognition.

VIII. DATASET

For experimental purpose we benchmark our results on Cohn Kanade Facial Expession dataset. e Nearly 2000 image sequences from over 200 subjects are in this dataset. All the expression dataset maintain a sequence from neutral to highest expressive grace. We took two highest graced expressive image of each subject. As we took 100 subjects, so the total image becomes 1200. 100 subject x 6 different expression x 2 of each expression. SO it becomes 100 x 6 x 2=1200. There is a significant variation of age group, sex and ethnicity.

The following figure (Fig. 3) shows a portion of the dataset of our experiment. The first row is the frontal faces from CK dataset. Second, third, fourth and fifth

©2014 Engineering and Technology Publishing

rows show mouth, left eye, and right eye and nose respectively.

Figure 3. Original dataset from CK database and the four parts of every image obtained from our algorithm and used as data.

IX. Experimental Setup

A. Face and Facial Parts Detection

In CK dataset, the background is large with all the face images. First we apply the Viola-Jones algorithm to find the faces. For eyes, nose and mouth detection we applied cascaded object detector with region set on already detected frontal faces Fig. 4. This cascade object detector with proper region set can identify the eyes, nose and mouth. Actually it uses Viola-Jones Algorithm as an underlying system. This object uses a cascade of classifiers to efficiently process image regions for the presence of a target object. Each stage in the cascade applies increasingly more complex binary classifiers, which allows the algorithm to rapidly reject regions that do not contain the target. If the desired object is not found at any stage in the cascade, the detector immediately rejects the region and processing is terminated.

Figure 4. Finding face and face parts

B. Training and Testing Data

We used here 100 subjects 1200 images and four face parts of the images. For every case (whole face, eyes, nose and mouth) we used 65% of the images for training and 35% for testing. We make separate face spaces for six different facial expressions. Then after ICA decomposition on the test images Euclidian distance is used for recognizing the closely match. When we applied ICA on whole faces, it happens that the system finds similar faces rather than similar expression when the same or similar person's face is in the both the train and test folder. For this reason part based analysis, WAPA, OEPA and also LS-ICA performs better than whole face ICA decomposition.

X. Result Analysis

A. FastICA

We performed differentiation on the vector of images as a preprocessing step. Fig. 5 shows the filtered mixed signals after differentiation. Different source images are depicted in Fig. 6. Independent components of the images are shown in Fig. 7. Fig. 8 Shows global 3D matrix of Performance Index. As described before Performance Index is a well-known formula for measuring the separation performance which is defined in equation [10].

Finally Fig. 9 shows the estimated Inverse matrix.

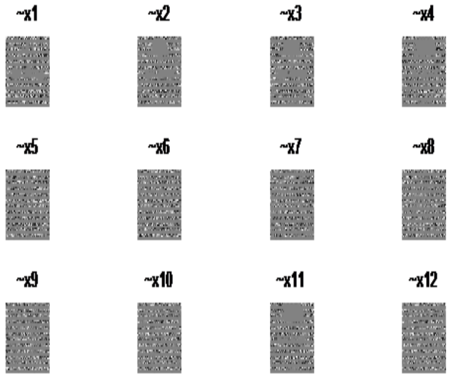

Figure 5. Filtered mixed signals (differentiation)

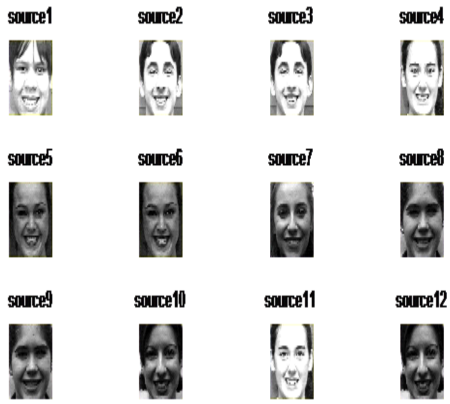

Figure 6. Source images

Figure 7. Independent components

©2014 Engineering and Technology Publishing

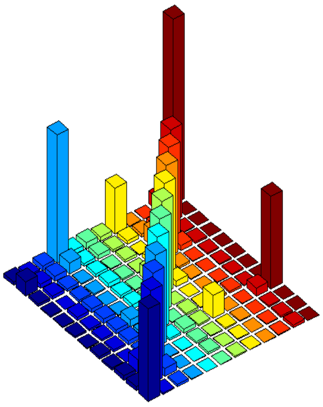

Figure 8. Global matrix 3D- performance index=0.0375

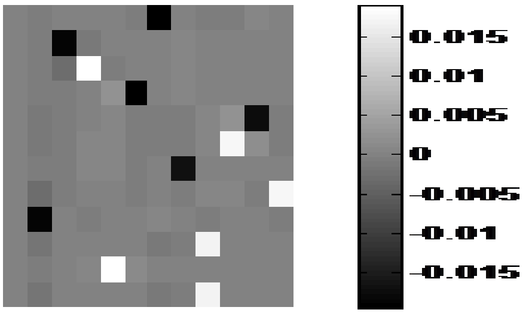

Figure 9. Estimated Matrix Inverse (W)

B. FastICA with Different Kernels

We implemented FastICA with three kernels: Hyper Tangent, Gaussian and Cubic kernels. Our experiment clearly shows the Gaussian kernel needs the less time compared to other kernels of FastICA. So for the next step we choose the FastICA algorithm with Gaussian kernel.

C. Influence of Different Parts

As described before we detected facial parts like eyes, mouth and nose and applied FastIca with Gaussian kernel and calculated the weighted accuracy. Sometimes a subset of all the four parts of the face is optimal in terms of processing time and accuracy for identifying an expression. As for example from the following table (Table I) we can see that to find the highest accuracy for fear both eyes and mouth are needed whereas only mouth is enough to achieve the highest accuracy of surprise expression. The other results are shown below.

TABLE I. EFFECTS OF FACIAL PARTS FOR EXPRESSION RECOGNITION. LE=LEFT EYE, RE=RIGHT EYE, N=NOSE, M=MOUTH, OEPA=OPTIMAL EXPRESSION-SPECIFIC PARTS ACCUMULATION

Facial Parts	Surprise	Anger	Sad	Happy	Fear	Disgust
LE	82%	65%	66%	70%	40%	55%
RE	82%	65%	66%	70%	40%	55%
LE+RE	82%	65%	66%	70%	40%	55%
N	15%	15%	50%	15%	30%	55%
M	98%	50%	55%	90%	80%	50%
LE+RE+N	60%	55%	50%	75%	56%	85%
LE+RE+M	98%	80%	70%	100%	96%	80%
N+M	69%	50%	50%	60%	50%	70%
LE+RE+N+M	85%	90%	86%	85%	85%	78%
WAPA-FastICA	92%	88%	82%	88%	90%	82%
OEPA-FastICA	98% (M)	90% (LE+RE+N+M)	86% (LE+RE+N+M)	100% (LE+RE+M)	96% (LE+RE+M)	85% (LE+RE+N)

D. LS-ICA

As described before the Locally Salient ICA (LS-ICA) has the ability to identify locally distorted parts more accurately. So we choose here LS-ICA to compare against FastICA methods even with the WAPA and OEPA based FastICA to understand which algorithm plays better role for specific applications of Facial Expression Recognition. Fig. 10 shows a set of LS-ICA components after performing the LS-ICA decomposition over the images of facial expression.

XI. COMPARISON AMONG ALL THE PROPOSED METHODS

The following table (Table II) shows the comparison among Whole face based ICA, Weighted All Parts Accumulation based FastICA, Optimal Expression Specific Parts Accumulation based FastICA, and Locally Salient ICA. For some expressions the LS-ICA outperforms the conventional FastICA and WAPA-FastICA methods. This is because LS-ICA algorithm has the strength to identify the local distortion more accurately. But the OEPA based FASTICA outperforms LS-ICA methods as we are choosing here only that facial parts which has the influence for specific facial expressions rather than choosing all the four face parts.

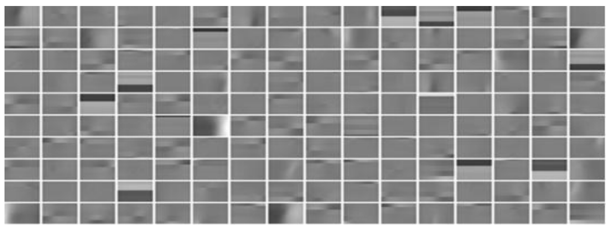

Figure 10. Locally Salient Independent components based on 8-by-8 windows and 40 dimensions.

TABLE II. COMPARISON AMONG PROPOSED ALGORITHMS. (W-FASTICA: WHOLE FACE ICA, WAPA: WEIGHTED ALL PARTS ACCUMULATION, OEPA: OPTIMAL EXPRESSION SPECIFIC PARTS ACCUMULATION, LS-ICA: LOCALLY SALIENT ICA.)

	Happy	Anger	Sad	Surprise	Fear	Disgust
W-FAstICA (FastICA with Gaussian Kernel on whole face)	58%	60%	65%	70%	50%	55%
WAPA-FASTICA	82%	68%	72%	82%	70%	72%
OEPA-FASTICA	90%	80%	78%	88%	80%	85%
LS-ICA	82%	70%	72%	82%	68%	70%

©2014 Engineering and Technology Publishing

XII. CONCLUSION

In this research work, we investigate FastICA and LS-ICA on CK facial expression images. We apply FastICA with Gaussian kernel on whole faces and different facial parts, like: eyes, mouth and nose. When we apply ICA on whole faces, the system finds similar faces rather than similar expression. This happens when the same or similar person's face lies in both train and test folder. So the overall recognition rate decreases. To overcome this, we apply part based analysis, WAPA, OEPA and LS-ICA. This WAPA-FastICA, OEPA-FastICA and LS-ICA outperform the prevalence FastICA method on whole face.

ACKNOWLEDGMENT

This work was supported in part by the Chinese Natural Science Foundation under Grant No. 61070117, and the Beijing Natural Science Foundation under Grant No. 4122004, the Australian Research Council under ARC Thinking Systems Grant No. TS0689874 and/or Discovery Projects 0988686 and 110101473, as well as the importation and development of High-Caliber Talents Project of Beijing Municipal Institutions.

REFERENCES

[1] S. B. Marian, R. M. Javier, and J. S. Terrence, "Face recognition by independent component analysis," *IEEE Transaction Neural Networks*, vol. 13, no. 6, pp. 1450-1464, 2002.

[2] J. J. Atick and A. N. Redlich, "What does the retina know about natural scenes?" *Neural Comput.*, vol. 4, pp. 196–210, 1992.

[3] G. Donato, M. S. Bartlett, J. C. Hagar, P. Ekman, and T. J. Sejnowski, "Classifying facial actions," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 974-989, 1999.

[4] P. Ekman and W. V. Priesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*, Palo Alto, CA: Consulting Psychologists Press, 1978.

[5] A. B. Humayra, P. M. W. David, L. Richard, and L. W. Trent, "Comparison of region based and weighted principal component analysis and locally salient ICA in terms of facial expression recognition," in *Proc. International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD),* Springer, Sydney, Australia, 2011, pp. 81-89.

[6] A. B. Humayra and P. M. W. David, "Facial expression recognition based on weighted all parts accumulation and optimal expression-specific parts accumulation," in *Proc. the International Conference on Digital Image Computing: Techniques and Applications*, Hobart, Tasmania, Australia, 2013, pp. 229-235.

[7] A. Hyvarinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Computation*, vol. 9, no. 7, pp. 1483- 1492, 1997.

[8] I. Buciu, C. Kotropoulos, and I. Pitas, "ICA and gabor representation for facial expression recognition," *in Proc. the IEEE Conference on Image Processing*, 2003, pp. 855-858.

[9] F. Chen and K. Kotani, "Facial expression recognition by supervised independent component analysis using MAP estimation," *IEICE Transactions on Information and Systems*, vol. E91-D, no. 2, pp. 341- 350, 2008.

[10] M. S. Bartlett, G. Donato, J. R. Movellan, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Face image analysis for expression measurement and detection of deceit," *in Proc. the Sixth* Joint *Symposium on Neural Computation*, 1999, pp. 8-15.

[11] C. Chao-Fa and F. Y. Shin, "Recognizing facial action units using independent component analysis and support vector machine," *Pattern Recognition*, vol. 39, pp. 1795-1798, 2006.

[12] M. J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with Gabor wavelets," *in Proc. the 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, pp.200-205.

[13] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Transaction on Neural Networks*, vol. 13, no. 6, pp. 1450-1464, 2002.

[14] A. Hyvarinen, "Survey of independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94-128, 1999.

Humayra B. Ali is currently pursuing PhD in CSEM School of Flinders University, South Australia. She graduated from the Dept. of Computer science and Engineering, Dhaka University, Bangladesh, in 2007 with a first class honors. She was involved in research project of MAGICIAN, Multiple Autonomous Ground-robotic International Challenge by Intelligent Adelaide Navigators (2009-2010), in Flinders University. She also worked as a research assistant in Deakin University, Melbourne with the project of "Early Heart Rate Detection with Machine Learning". Her research interests are in pattern recognition, machine learning, computer vision, image analysis and facial expression recognition. Her contact email address is: humayra_binte02@yahoo.com for any further correspondence.

David M. W. Powers is professor and director of KIT Centre, School of Comp. Science Engineering & Mathematics of Flinders University, Adelaide, South Australia. His specific research framework takes Language, Logic and Learning as the cornerstones for a broad Cognitive Science perspective on Artificial Intelligence and its practical applications. Prof. Powers is known as a pioneer in the area of Natural Language Learning, Unsupervised Learning and Evaluation of Learning, and was Founding President of ACLSIGNLL, as well as initiating the CoNLL conference. His previous positions include Telecom Paris, University of Tilburg, University of Kaiserslautern, Macquarie University, as well as work with industry, and commercialization of research through several start-up companies. Prof. Powers is also a Visiting Professor at the Beijing University of Technology in Beijing Municipal Lab for Multimedia & Intelligent Software, and is Editor-in-Chief of the new Springer journal Computational Cognitive Science and the book series Cognitive Science and Technology. His email address is powers@ieee.org <mailto:powers@ieee.org>.

©2014 Engineering and Technology Publishing