Incremental Learning in Autonomous Systems: Evolving Connectionist Systems for On-line Image and Speech Recognition

Nikola Kasabov, David Zhang, Paul S. Pang
Knowledge Engineering & Discovery Research Institute,
Auckland University of Technology,
Auckland, New Zealand,
nkasabov@aut.ac.nz

Abstract – The paper presents an integrated approach to incremental learning in autonomous systems, that includes both pattern recognition and feature selection. The approach utilizes evolving connectionist systems (ECoS) and is applied on on-line image and speech pattern learning and recognition tasks. The experiments show that ECoS are a suitable paradigm for building autonomous systems for learning and navigation in a new environment using both image and speech modalities.

Index Terms – Autonomous Systems; Incremental Learning; Adaptive Systems; Multimodal Systems; Image Recognition; Speech Recognition; Evolving Connectionist Systems (ECOS), Evolving Growing Cluster Classifier (EGCC), Online Adaptation.

I. Introduction

Building autonomous systems that continuously learn from incoming information is a difficult task that falls in the area of computational intelligence. A solution to this task would be applicable to robotics, on-line decision support systems, intelligent systems for Web information processing, business intelligence.

Here we propose an integrated approach to solving this task using evolving connectionist systems (ECoS) where an autonomous system is evolving its structure and input features to learn incrementally new images and new spoken commands from continuous streams of image and speech data.

The paper first introduces the ECoS paradigm and a new algorithm called EGCC (evolving growing cluster classifier). The algorithm is illustrated on on-line learning and recognition of image and speech data.

Next, the paper introduces an incremental principal component analysis algorithm for feature selection and feature modification as part of an incremental learning process. The algorithm is illustrated on a task of learning face image data stream.

II. Evolving connectionist systems

Evolving connectionist system (ECoS) [1] are systems that evolve their structure and functionality over time through interaction with the environment – fig.1.

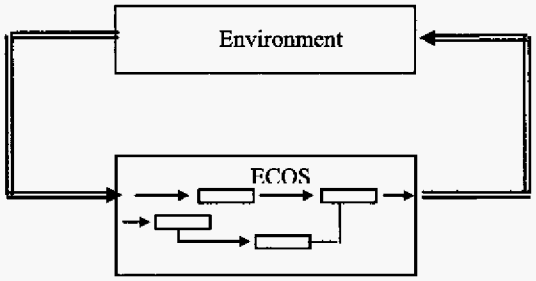

Fig.1. ECoS evolve their structure and functionality through incremental learning from data streams and through interaction with the environment

There are various implementations of the ECoS paradigm [2, 3]. In this paper, a new type of ECoS implementation called Evolving Growing Cluster Classifier (EGCC) is introduced.

The EGCC is a knowledge-based neural network model for classification and is modification of the Radial Basis Function (RBF) type networks. EGCC is similar to the Evolving Classifier Function (ECF) network [1]. The implementation of the concept of "growing" in EGCC is different from that of other growing neural networks, such as growing neural gas (GNG) and growing cell structures (GCS) [4]. EGCC has simple training and test procedures that require only two pass training (no further iterations are needed). There is no need for a parameter setting. A cluster center (CC) grows gradually (according to a pre-defined growth speed) till its influence field reaches the maximum influence field or the influence field of cluster center from a different class. The main factor that affects the speed of the training process of EGCC is the growing speed of the CCs that can be set by users. A CC is learned by a neuron.

The EGCC has three different operation modes: learning, adaptation, and classification. They are introduced in the following sections.

A. EGCC learning algorithm

In the EGCC learning mode, the network is built from incoming data starting with no nodes (CC) at all. The EGCC learning algorithm is described here in 5 steps:

1. Finding cluster centers (CCs) for each category according to a set maximum radius of influence field (maxinf), initially set to 0;

   The method of finding CCs of one category is the following one: the first sample is used to create the first CC. For this category. From the second sample on, calculate the Manhattan distances between this sample and all the existing CCs. If any of the distances is less than the maxinf, nothing happens; otherwise, create a new CC for this sample. The same process is used for establishing the CCs for each category.

2. The influence field of each CC starts to grow at a uniform speed. The growth of the influence field of a CC lasts till it reaches the maxinf, or it reaches the influence field of another CC of different category.

3. Check all the samples against the established CCs. If one sample falls into the influence field of any CC with the same category, nothing happens; otherwise, if it falls out of the influence field of all the CCs, create a new CC for this sample, set the influence field to 0; otherwise, if it falls into the influence field of a CC with a different category, shrink this CC so that the influence field = Distance between this sample and this CC, also create a new CC for this sample, set the influence field to 0;

4. Check all the samples against the existing CCs. If any sample falls out all CCs, create a new CC for it, set the influence field to 0.

5. All the CCs grow again at the same uniform speed. The growth of each CC lasts till its influence field reaches maxinf, or it starts to overlap with the influence field of another CC with different category.

Note that according to the growth method of the EGCC learning algorithm, there will be no influence field overlapping between any two different categories. This leads to an easier classification decision process.

Fig. 2 shows how the influence field of each CC grows gradually from status (a) to (d). Suppose there are 3 CCs in the network, two of them from one category (represented by the cross), one from another category (represented by the star).

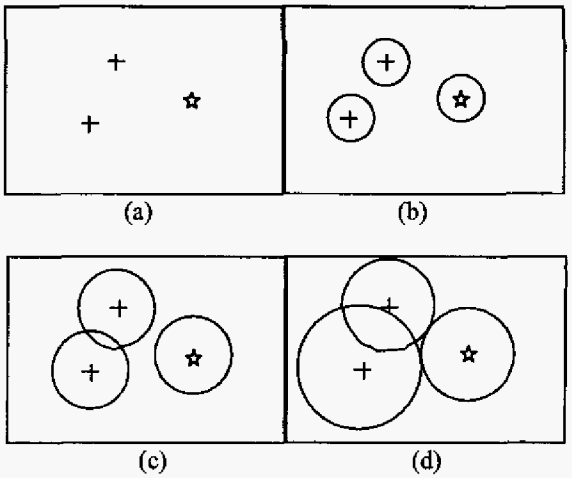

Fig. 2 A example shows how the influence field of the CCs grow from status (a) to (d) gradually

B. EGCC adaptation algorithm

As EGCC belongs to the ECoS family, a very important feature of the EGCC network is its adaptation ability. This means that an EGCC network can be adapted online by presenting new samples, either of existing category or new category. This feature is crucially useful for robots to learn new knowledge on the fly. The EGCC online adaptation algorithm consists of the following 3 steps:

When a new sample is presented to an existing EGCC network,

1. Find all the CCs with the same category as this new sample;

2. If this sample falls into the influence field of any of these CCs, nothing happens, adaptation process finishes. Else, if it doesn't fall into the influence field of any of these CCs, create a new CC for this sample, set its influence filed to 0.

3. Set the influence field of all the CCs (from all the categories) to 0. Then all the CCs start to grow at the same uniform speed as that of the learning algorithm. The growth of each CC lasts till its influence field reaches maxinf or it starts to overlap with the influence field of another CC with different category.

Assume that the network before adaptation is given by the status (d) from Fig. 2. Upon arrival of a new sample (of the class represented by a star), the status of the network changes as shown in Fig. 3.

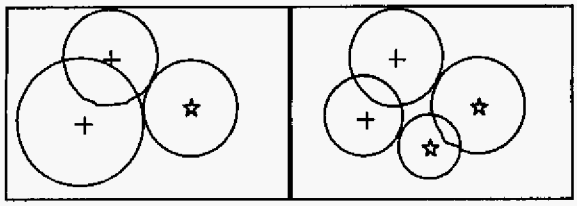

Fig. 3 This example shows how the adaptation algorithm works in the EGCC ECoS

C. *The EGCC classification algorithm*

The EGCC classification algorithm is described below:

1. Enter a new sample for classification.
2. If the sample falls into the influence field of any CC, announce the winning category with the shortest distance between the new sample and the CC;
3. If the new sample doesn't fall into the influence field of any CC, find the shortest distance from this sample to all the CCs; announce the winning category with the shortest distance.

III. DATA ACQUISITION

Many robotic applications require using both auditory and visual signals. The auditory signals can be human speech instructions. The visual information can be images captured using camera installed on the robot. A popular robotic application is described in [5]. A verbal instruction is given by the user to the robot. For processing this verbal instruction, usually a parser is used with a simple grammar optimized for command sentences, in order to extract the main auditory information. E.g., extract the word "cup" from the user's instruction "pick up the cup". After receiving and parsing this instruction, the robot reacts by finding the "cup" and picking it up. The whole process of this scenario is illustrated in Fig. 4.

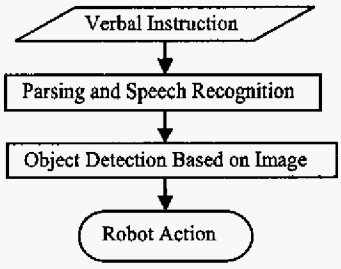

Fig. 4 Flowchart of verbal and image based robot control process

According to this scenario, after receiving the verbal instruction, there are two modules employed, one for speech recognition, and another - for image recognition. According to this, in order to testify the generalization ability of the EGCC network, a simple scenario was used, where the following speech and image datasets were collected.

A speech dataset was collected upon the vocabulary shown in the Table 1, using close-mouth microphone. In the dataset, there are 8 speakers; each utterance was repeated 3 times by each speaker. So, totally there are 168 samples presented to the EGCC system.

Table 1. Vocabulary used in the speech recognition module

	Speech Utterance
1	"Pen"
2	"Rubber"
3	"Cup"
4	"Orange"
5	"Circle"
6	"Ellipse"
7	"Rectangle"

The image dataset was collected upon 4 objects: pen, rubber, cup and orange. There are 20 images for each object, so totally 80 samples were presented top the system. Originally, the images were taken using a digital camera with a resolution of 640 by 480. Then, all the images were re-sampled to 64 by 48 in a grayscale. Some image samples are shown in Fig. 4.

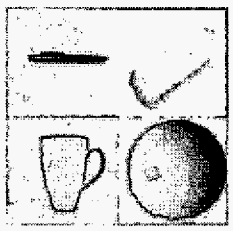

Fig. 5 Some image samples in the image dataset

IV. FEATURE EXTRACTION

In this section, signal processing and feature extraction methods for speech and image used in this paper are introduced.

A. *Speech processing and feature extraction*

Features were extracted from each speech sample using the following method. First, Mel Frequency Cepstrum coefficients (MFCC) were extracted as acoustic features from the raw speech signal. Then, spectral analysis of the speech signal was performed over 20ms using Hamming window with 50% overlap. Discrete cosine transformation (DCT) was applied on the MFCC of the whole word in the following manner. For an m frame segment, DCT transformation will result in a set of m DCT coefficients. This sequence is truncated to achieve a fixed-size input

vector consisting of 20 * d, where d is the dimensionality of the feature space [6].

B. *Image processing and feature extraction*

Features were extracted for each face image using the composite profile technique. The composite profile features are composed of the average value of the columns in the image followed by the average value of rows in the image. It is a relevant feature to characterize symmetric and circular patterns, or patterns isolated in a uniform background. This feature can be useful to verify the alignment of objects. Then, the interpolation method was applied to reduce the number of features from original 112 (64 row composite profiles features and 48 column composite profile features) to 64 [7].

While the procedure above describes the feature extraction process that results in a fixed number of feature that can be used though the whole operation of the resulted system, the next section introduces a novel method of incremental PCA for updating the features of a system during an incremental learning from a stream of data.

V. ON LINE FEATURE EXTRACTION THROUGH INCREMENTAL PCA

Principle component analysis (PCA) is a typical method for feature extraction. However, the original PCA is not suited for incremental learning purposes. To achieve the end of online pattern recognition, we used a method of incremental principle component analysis (IPCA) to update eigenvectors and eigenvalues in an incremental way as the following.

Assume that N training samples $x_i \in R^n \ (i = 1, \cdots, N)$ have been presented so far, and an eigenspace model $\Omega = (\bar{x}, U, \Lambda, N)$ is constructed by calculating the eigenvectors and eigenvalues from the covariance matrix of x_i, where $\bar{x}$ is a mean input vector, U is a $n \times k$ matrix whose column vectors correspond to the eigenvectors, and Λ is a $k \times k$ matrix whose diagonal elements correspond to the eigenvalues. Here, k is the number of dimensions of the current eigenspace that is often determined such that a specified fraction of energy in the eigenvalue spectrum is retained.

Let us consider the case that the $(N+1)th$ training sample y is presented. The addition of this new sample will lead to the changes in both of the mean vector and covariance matrix; therefore, the eigenvectors and eigenvalues should also be recalculated. The mean input vector $\bar{x}$ is easily updated as follows:

$$\bar{x}' = \frac{1}{N+1}(N\bar{x} + y)$$

Therefore, the problem is how to update incrementally the eigenvectors and eigenvalues.

When the eigenspace model Ω is reconstructed to adapt to a new sample, we must check if the dimensions of the eigenspace should be changed or not. If the new sample has almost all energy in the current eigenspace, the dimensional augmentation is not needed in reconstructing the eigenspace. However, if it has some energy in the complementary space to the current eigenspace, the dimensional augmentation cannot be avoided. This can be judged from the norm of the following residue vector h:

$$h = (y - \bar{x}) - Ug$$

where

$$g = U^T (y - \bar{x}).$$

When the norm of the residue vector h is larger than a threshold value η, it must allow the number of dimensions to increase from k to $k+1$, and the current eigenspace must be expanded in the direction of h. Otherwise, the number of dimensions remains the same.

It has been shown that the eigenvectors and eigenvalues should be updated based on the solution of the following intermediate eigenproblem [9]:

$$\left(\frac{N}{N+1} \begin{bmatrix} \Lambda & 0 \\ 0^T & 0 \end{bmatrix} + \frac{N}{(N+1)^2} \begin{bmatrix} gg^T & \gamma g \\ \gamma g^T & \gamma^2 \end{bmatrix} \right) R = R\Lambda'$$

where $\gamma = h^T (y - x)$, R is a matrix $(k+1) \times (k+1)$ whose column vectors correspond to the eigenvectors obtained from the above intermediate eigenproblem, Λ' is the new eigenvalue matrix, and 0 is a k-dimensional zero vector. Using this solution R we can calculate the new $n \times (k+1)$ eigenvector matrix U' as follows:

$$U' = [U, \hat{h}]R$$

where

$$\hat{h} = \begin{cases} h / \|h\| & if \|h\| > \eta \\ 0 & \text{otherwise.} \end{cases}$$

As we can see from the above equation, R operates as a rotation of the eigenvectors; we will call R a rotation matrix. Note that $\hat{h} = 0$, R degenerates into a $n \times k$ matrix; that is, the dimensions of the updated

eigenspace remains the same as those of the previous eigenspace.

To implement the above IPCA technique to online face recognition, as rotation happening, eigenvectors are rotated. Fig. 6 (a) gives an example of eigenspace rotation. As we can see, the eigenfaces are changed between two incremental learning sessions, but the number of eighenfaces are kept the same. Whereas as augmentation happening, since it is usually accompanied by rotation, the eigenfaces are not only rotated , but also their number increases. Fig. 6 (b) is an example of eigenspace augmentation with rotation.

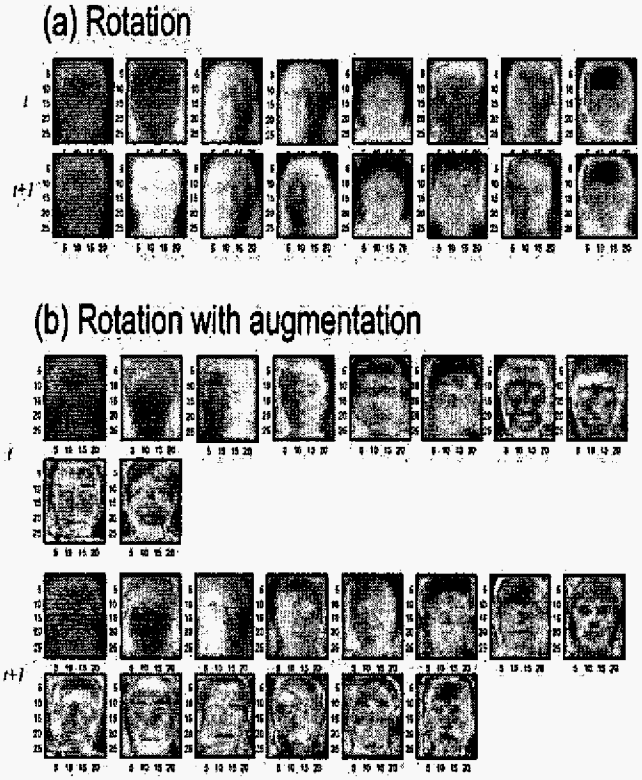

Fig. 6. Examples of eigenspace rotation and Augmentation

VI. Experiments

In order to testify whether the EGCC network is an efficient tool for building classification models for speech and image recognition tasks, the following experiments were conducted in two distinct phases based on simple speech and image datasets.

A. Speech recognition using EGCC network

This experiment was divided into two steps. In the first step, the generalization ability of the EGCC network was testified. In the second step, the adaptation ability of the EGCC network was verified.

First, an EGCC model was built to recognize 4 words ("Pen", "Rubber", "Cup" and "Orange"). Using 96 samples, a 10-fold cross-validation test was performed. The confusion table is shown in Table 2.

Table 2. Speech recognition confusion table before adaptation

	Pen	Rubber	Cup	Orange
Pen	22	0	2	0
Rubber	0	21	2	1
Cup	0	3	21	0
Orange	0	2	2	20

Then, the rest 3 words ("Circle", "Ellipse" and "Rectangle") were added to the trained EGCC model. Another 10-fold cross-validation test was performed. The confusion table of this experiment is shown in Table 3. (For simplicity, only the first 3 letters of each word are shown in the table).

Table 3. Speech recognition confusion table after adaptation

	Pen	Rub	Cup	Ora	Cir	Ell	Rec
Pen	21	0	2	0	1	0	0
Rub	0	21	2	1	0	0	0
Cup	0	3	20	0	1	0	0
Ora	0	3	2	18	0	1	0
Cir	1	0	0	0	23	0	0
Ell	0	0	0	2	21	1	0
Rec	2	1	0	3	1	0	17

As shown in tables 2 and 3, the EGCC network is an efficient tool for building incrementally adaptive systems, for speech recognition in this case. Testified by the confusion table in Table 3 is also the ability of the EGCC model to learn new words (classes) while maintaining its performance on pervious words. This illustrates the adaptive characteristic of the EGCC model.

B. Image recognition using EGCC network

A similar experimental setup was applied for incremental adaptive image recognition. First, an EGCC model was built to recognize 3 objects (pen, rubber, and cup). With the 60 samples, a 10-fold cross-validation test was performed. The confusion table is shown in Table 4.

Table 4. Image recognition confusion table of a GCC model before adaptation

	Pen	Rubber	Cup
Pen	20	0	0
Rubber	1	19	0
Cup	0	2	18

Then, another object (orange) was added to the trained EGCC model. Another 10-fold cross-validation test was performed. The confusion table of this experiment is shown in Table 5.

Table 5. Image-based object recognition confusion table after adaptation

	Pen	Rubber	Cup	Orange
Pen	20	0	0	0
Rubber	1	19	0	0
Cup	0	2	16	2
Orange	0	1	1	18

As shown by table 4 and 5, the EGCC network is an efficient tool for building incremental, image recognition model. Moreover, as illustrated by the confusion table in Table 3, while the EGCC model was adapted to be able to recognize a new object, it maintains its performance on the pervious objects.

VI. Autonomous Adaptive Systems for Incremental Speech and Image Learning and Recognition Using ECOS

A block diagram of an integrated autonomous adaptive system for speech and image incremental learnin g and recognition using ECoS is shown in Fig. 7.

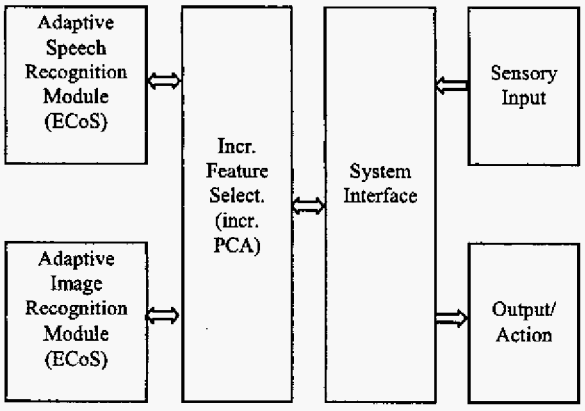

Fig. 7 Model of autonomous adaptive system using ECoS

VII. Conclusions and Future Work

This research and experiments show that the EGCC is an appropriate network for the creation of models for the task of speech and image-based object recognition. The models are adaptive, which means the model can be adapted to accommodate new objects without degrading its performance over already learned objects. Hence, the EGCC network can be applied to robotic application for object detection and recognition (examples of robots are given in Fig. 8).

Fig. 8 Some real robots

Future work can be directed towards improved learning algorithms and real applications.

Acknowledgment

This work is supported by the Foundation of Research, Science and technology FRST of New Zealand through a NERF grant NERFAUTX02001.

References

[1] N. Kasabov, "Evolving connectionist systems: Methods and applications in bioinformatics, brain study and intelligent machines", Springer, London, 2002.

[2] N. Kasabov, "Evolving fuzzy neural networks for supervised / unsupervised online knowledge-based learning", *IEEE Trans. On Systems, Man and Cybernetics, Part B: Cybernetics,* 31(6), 195-202, 2001.

[3] D. Zhang, N. Kasabov, Q. Song and I. Nishikawa, "Evolving connectionist modeling of auditory, visual and combined stimuli perception based on EEG data", 7th Joint Conference on Information Sciences, North Carolina, USA. pp. 1361-1364, 2003.

[4] B. Fritzke, "Growing cell structures - A self-organizing network for unsupervised and supervised learning", Neural Networks, vol.7, no.9, 1994.

[5] O. Rogalla6, M. Ehrenmann, R. Zollner, R. Becher and R. Dillmann, "Using Gesture and Speech Control for Commanding a Robot Assisant", Institute of Computer Design and Fault Tolerance, University of Kalsruhe, Germany, unpublished

[6] D. W. Tank and J. J. Hopfield, "Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit", *IEEE Trans. on Circuits and Systems*, 533-541, 1986.

[7] A. Ghobakhlou, D. Zhang and N. Kasabov, "An Evolving Neural Network Model for Person Verification Combining Speech and Image", Proceedings of 11th International Conference on Neural Information Processing, LNCS, vol.. 3316, 381-386, Springer, 2004.

[8] P. Hall and R. Martin, "Incremental eigenanalysis for classification," British Machine Vision Conference, Vol. 1, pp. 286-295, 1998.