# Online LDA which can perform both successive learning and incremental learning

**Kazuyuki Hiraoka, Ken-ichi Hidai,**
**Hiroshi Mizoguchi, Taketoshi Mishima, and Shuji Yoshizawa**
**Saitama University**
**Dept. of Information and Computer Sciences**
**255 Shimo-Okubo, Urawa, 338, Japan**
**hira@ics.saitama-u.ac.jp**

## Abstract

Adaptability is a desired property for the computation in coming years. In particular, adaptability is important for face identification because change of situations can occur in various cases. For face identification, linear discriminant analysis (LDA) is applied extensively [8][9]. However, LDA is poor at adaptability.

Recently, the authors have proposed an *online* version of LDA [11], which is referred to online LDA (OLDA). By OLDA, the face identification system can be updated with low computational cost when new additional images are presented. Hence OLDA has the ability of adaptation to the change of environment. OLDA also has an advantage that $N \times N$ matrices never appear in its calculation when the number of pixels in each image is $N$.

In the present paper, we will show experimental results that OLDA works efficiently for both two typical scenarios of the change of situations, *successive learning* and *incremental learning*.

**Keywords:** Linear discriminant analysis, Face identification, Online learning, Adaptability

## 1 Introduction

Adaptability is a desired property for the computation in coming years. In particular, adaptability is important for face identification because change of situations can occur in various cases. The following are a few examples: (a) The number of persons to be identified may increase or decrease. (b) The frequency of each person may change as time passes. (c) The face itself can change. That is, a person may begin to use glasses one day. (d) Image condition can change. For example, low contrast images may increase when the rainy season has set in.

For face identification, linear discriminant analysis (LDA) is applied extensively [8][9]. However, LDA is poor at adaptability. This is because LDA is a batch learning algorithm. Namely, LDA is designed in the follow manner: (1) all sample images are given at once, (2) the discriminant matrix $A$ is calculated for the sample images, and then (3) identification is performed by use of $A$. Owing to this design, we have to re-calculate $A$ every time when we add new data to update the identification system. This calculation is heavy for high dimensional data such as face images.

When the situation changes gradually or suddenly, one time learning is not sufficient and additional learning is indispensable for adaptability. Thus the identification system must have the ability to learn new data and update itself with small calculations. Such an algorithm that has this ability is called *online* learning algorithm. As we have mentioned above, conventional LDA is not online learning.

Recently, the authors have proposed an *online* version of LDA [11], which is referred to online LDA (OLDA). By OLDA, the face identification system can be updated with low computational cost when new additional images are presented. Hence OLDA has the ability of adaptation to the change of environment. OLDA also has an advantage that $N \times N$ matrices never appear in its calculation.

In the present paper, we will show experimental results that OLDA can perform both *successive learning* and *incremental learning*.

A scenario of successive learning is as follows (Figure 1).

1. Face images of three persons A, B, and C are presented to the identification system in random order. After leaning of these examples, the system can identify A, B, and C.

2. Suddenly, a new person D appears. At this time, the system cannot identify D.

3. From this time, *all the four persons* A, B, C, and D are presented. After some transient period, the system can identify A, B, C, and D.

**successive learning:**

`A,B,A,C,C,B,A,B,C,...`
$\Rightarrow$ `D,C,A,A,D,B,B,C,A,C,D,B,A,C,B,D,...`

**incremental learning:**

`A,B,A,C,C,B,A,B,C,...`
$\Rightarrow$ `D,D,D,...`

Figure 1: Comparison between successive learning and incremental learning

On the other hand, in a scenario of incremental learning, the last stage is changed as follows.

3'. Then *the last person* D is presented repeatedly. After some transient period, the system can identify A, B, C, and D.

# 2 Face identification via LDA

The procedure of identification via LDA is as follows. Suppose that we are trying to identify face images of $M$ persons by extracting $L$ features [1] from each image. If the image is $n \times n$ size, it is represented by an $N = n^2$-dimensional vector $\boldsymbol{x}$. Let the discriminant matrix $A$ be an $N \times L$ matrix. Before the identification, the mean face $\bar{\boldsymbol{x}}^c$ of each person $c$ is transformed to the $L$-dimensional feature vector $\bar{\boldsymbol{y}}^c = A^T \bar{\boldsymbol{x}}^c$, $c = 1, \cdots, M$. When a new image $\boldsymbol{x}$ is presented, its feature vector $\boldsymbol{y} = A^T \boldsymbol{x}$ is calculated and the nearest vector $\bar{\boldsymbol{y}}^{c^*}$ among $\bar{\boldsymbol{y}}^1, \cdots, \bar{\boldsymbol{y}}^M$ is selected. Then the image $\boldsymbol{x}$ is guessed to be the person $c^*$.

To obtain accurate identification, selection of $A$ is critical. LDA is a method to select an appropriate $A$.

In the conventional LDA, the discriminant matrix $A$ is determined by use of the solution of a generalized eigenvalue problem. This calculation is heavy when the image size $N$ is large. Therefore, in a time-varying situation, conventional LDA is not suitable. Indeed, when we add new data and update the identification system, we have to re-calculate a generalized eigenvalue problem every time. Thus, online version of LDA is demanded.

Though iterative algorithms have been proposed[2] for neural network based LDA [6][5], they are not sufficiently "online". Since those algorithms keep $N \times N$ matrices, they require $O(N^2)$ time for one step updating and $O(N^2)$ memory. These computational costs are still too large to update the system on the fly.

# 3 OLDA algorithm

## 3.1 Basic form

At every time step $t = 1, 2, 3, \cdots$, a new pair $(\boldsymbol{x}(t), c(t))$ is presented, where $\boldsymbol{x}(t)$ is an $N$-dimensional data vector, $c(t) \in \{1, \cdots, M\}$ is the class of $\boldsymbol{x}(t)$, and $M$ is the number of classes. Based on this pair, auxiliary variables are updated as fol-

[1] Since $M$ mean faces span an $(M-1)$ dimensional hyperplane, $L$ is set to be at most $L-1$. It is redundant to choose $L \geq M$.

[2] The original algorithm in [5] is a batch learning. However, we can easily modify it to an iterative learning by replacing the calculation of the standard deviation in [5] with an iterative one.

lows:

$$
\begin{aligned}
t^c(t) &= t^c(t-1) + \delta(c, c(t)), & (1)\\
\bar{\boldsymbol{x}}(t) &= \left(1 - \frac{1}{t}\right)\bar{\boldsymbol{x}}(t-1) + \frac{1}{t}\boldsymbol{x}(t), & (2)\\
\bar{\boldsymbol{x}}^c(t) &= & \\
&\begin{cases} \left(1 - \frac{1}{t^c(t)}\right)\bar{\boldsymbol{x}}^c(t-1) + \frac{1}{t^c(t)}\boldsymbol{x}(t) & (c = c(t)),\\ \bar{\boldsymbol{x}}^c(t-1) & (c \neq c(t)), \end{cases} & (3)\\
\boldsymbol{v}^c(t) &= \bar{\boldsymbol{x}}^c(t) - \bar{\boldsymbol{x}}(t), & (4)\\
\boldsymbol{w}(t) &= \boldsymbol{x}(t) - \bar{\boldsymbol{x}}^{c(t)}(t), & (5)\\
B(t) &= \frac{1}{M}\sum_{c=1}^{M}\boldsymbol{v}^c(t)\boldsymbol{v}^c(t)^T, & (6)
\end{aligned}
$$

where $c = 1, \cdots, M$ and $\delta(c, c(t)) = 1\,(c = c(t))$, $0\,(c \neq c(t))$. Then $N \times L$ discriminant matrix $A$ is updated:

$$
\begin{aligned}
A(t) = A(t-1) + \eta\Big[&B(t)A(t-1)\\
&- \frac{1}{2}B(t)A(t-1)\,A(t-1)^T(\boldsymbol{w}(t)\boldsymbol{w}(t)^T + \epsilon I)A(t-1)\\
&- \frac{1}{2}(\boldsymbol{w}(t)\boldsymbol{w}(t)^T + \epsilon I)A(t-1)\,A(t-1)^T B(t)A(t-1)\Big],
\end{aligned} \qquad (7)
$$

where the learning coefficient $\eta$ and the regularization coefficient $\epsilon$ are small positive numbers, and $I$ is the identity matrix. The term $+\epsilon I$ is useful for stabilization of the algorithm [13].

### 3.2 Fast calculation

In the present paper, we assume that the number $N$ of pixels in each image is much larger than the number $M$ of classes. Then, note that we can calculate the right hand side of (7) without generating $N \times N$ matrices explicitly. The procedure is as follows: First, calculate

$$
\begin{aligned}
\boldsymbol{u} &= A^T\boldsymbol{w}, & (8)\\
A^T BA &= \frac{1}{M}\sum_{c=1}^{M}(A^T\boldsymbol{v}^c)(A^T\boldsymbol{v}^c)^T. & (9)
\end{aligned}
$$

Then, calculate each term in (7),

$$
\begin{aligned}
BA &= \frac{1}{M}\sum_{c=1}^{M}\boldsymbol{v}^c(A^T\boldsymbol{v}^c)^T, & (10)\\
BAA^T\boldsymbol{w}\boldsymbol{w}^T A &= ((BA)\boldsymbol{u})\boldsymbol{u}^T, & (11)\\
\boldsymbol{w}\boldsymbol{w}^T AA^T BA &= \boldsymbol{w}(\boldsymbol{u}^T(A^T BA)), & (12)\\
BAA^T A &= (BA)(A^T A), & (13)\\
AA^T BA &= A(A^T BA). & (14)
\end{aligned}
$$

## 4 Experimental results

In this section, performance of online LDA algorithms is tested for face identification task (Fig. 2). In the present paper, we will show experimental results that OLDA can perform both *successive learning* and *incremental learning*.

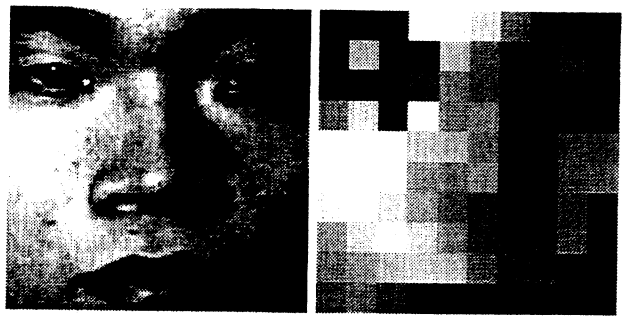

Figure 2: *A sample image for the simulation (left: original, right: reduced to* $10 \times 10$*)*

The setting of the simulation is written in Table 1. Face images of three persons A, B, and C are prepared for the simulation. In the first half, for both the successive learning and the incremental learning, face images of two persons A and B are presented in a random order. Then, a new person C is added for the latter half. For the successive learning, face images of three persons A, B, and C are presented in a random order. On the other hand, for the incremental learning, face images of only the third person C are presented in a random order. Through the simulation, the correct identification ratio is evaluated on every 10 samples. Different image sets of same persons are used for the learning and the evaluation. The result of the simulation is shown in Fig. 3 and Fig. 4.

Table 1: Setting of the simulation

| task | identification of face images |
|---|---|
| data vector $\boldsymbol{x}(t)$ | face images under various illumination conditions<br>(front view, 256 level gray scale, normalized to $[-1, +1]$) |
| size of $\boldsymbol{x}(t)$<br>number of classes to be identified<br>number of features | $N = 10 \times 10 = 100$ (pixels)<br>first half: $M = 2$(persons), latter half: $M = 3$(persons)<br>$L = M - 1$ (= number of columns in $A$) |
| initial values of elements in $A$<br>learning coefficient<br>regularization coefficient | random values from the uniform distribution on $[-0.001, +0.001]$<br>$\eta = 0.01$<br>$\epsilon = 0.01$ |
| procedure of learning<br>number of face images for learning | Face images for learning is presented in a random order.<br>— first half —<br>500(images per person) $\times$ 2(persons) = 1000<br>— latter half —<br>*successive:* 500(images per person) $\times$ 3(persons) = 1500<br>*incremental:* 1500(images per person) $\times$ 1(persons) = 1500 |
| procedure of evaluation<br>number of face images for evaluation | The ratio of the correct identification is evaluated for face images which are different from the face images for learning.<br>— first half —<br>100(images per person) $\times$ 2(persons) = 200<br>— latter half —<br>100(images per person) $\times$ 3(persons) = 300 |
| number of trials | 100 independent trials with different random seeds |

Despite that only the third person C is presented in the incremental learning, deterioration of performance is small (Fig. 5). Note that the identification ratio for the latter half in the simulation is evaluated for all three persons A, B, and C.

# 5 Conclusion

In the present paper, it is experimentally shown that OLDA can perform both *successive learning* and *incremental learning*. In the case of successive learning, OLDA can adapt to the new situation. On the other hand, in the case of incremental learning, OLDA can keep high performance even for the persons who are not presented for long time.

# References


[1] E. Oja, "A simplified neuron model as a principal component analyzer", J. Math. Biol., Vol. 15, pp. 267–273, 1982.

[2] T. D. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network", Neural Networks, Vol. 2, No. 6, pp. 459–473, 1989.

[3] K. Hornik and C.-M. Kuan, "Convergence analysis of local feature extraction algorithms", Neural Networks, Vol. 5, No. 2, pp. 229–240, 1992.

[4] W.-Y. Yan, U. Helmke, and J. B. Moore, "Global analysis of Oja's flow for neural networks", IEEE Tr. NN, Vol. 5, No. 5, pp. 674–683, 1994.

[5] J. Mao and A. K. Jain, "Artificial neural networks for feature extraction and multivariate

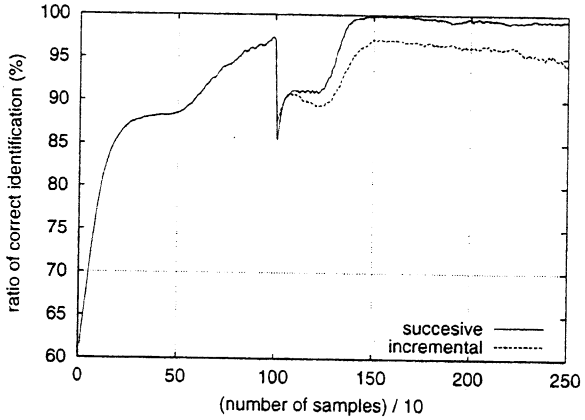


Figure 3: *Mean learning curves of the successive learning and the incremental learning for* 100 *independent trials. Horizontal axis: number of presented samples. Vertical axis: percentage of correct discrimination.*

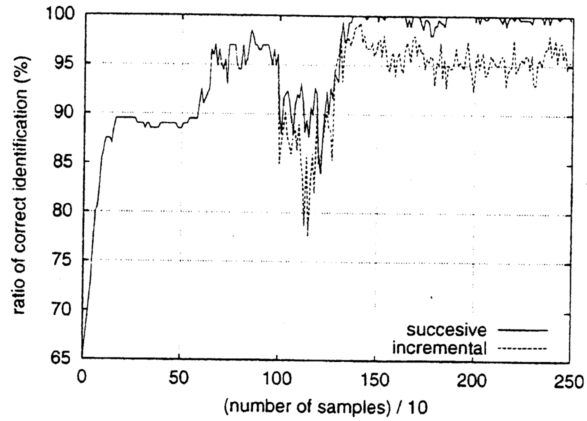


Figure 4: *Learning curves of the successive learning and the incremental learning for one trial. Horizontal axis: number of presented samples. Vertical axis: percentage of correct discrimination.*

data projection", IEEE Tr. NN, Vol. 6, No. 2, pp. 296–317, 1995.

[6] C. Chatterjee and V. P. Roychowdhury, "On self-organizing alogorithms and networks for class-separability features", IEEE Tr. NN, Vol. 8, No. 3, pp. 663–678, 1997.

[7] M. B. Nevel'son and R. Z. Has'minskiĭ, *Stochastic Approximation and Sequential Estimation*, Nauka, 1968 (in Russian; translated into Japanese by T. Kitagawa and K. Tajima, 1983).

[8] T. Kurita and S. Hayamizu, "Gesture Recognition using HLAC Features of PARCOR Images and HMM based Recognizer", Proc. 3rd int. conference on automatic face and gesture recognition, April, 1998, pp. 422-427.

[9] W. Zhao, et al., "Discriminant Analysys of Principal Components for Face Recognition", Proc. 3rd int. conference on automatic face and gesture recognition, 1998, pp. 422-427. pp. 336-341.

[10] K. Fukunaga, *Statistical Pattern Recognition*, Academic Press, New York, 1989.

[11] K. Hiraoka and M. Hamahira, "On Successive Learning Type Algorithm for Linear Discriminant Analysis", IEICE Technical Report, NC99-73, pp. 85–92, 1999 (in Japanese).

[12] Kazuyuki Hiraoka, Ken-ichi Hidai, Masashi Hamahira, Hiroshi Mizoguchi, Takaomi Shigehara, and Taketoshi Mishima "Derivation of online LDA algorithm and its application for face identification", Fifth Robotics Symposia, pp. 226–231, 2000 (in Japanese).

[13] Kazuyuki Hiraoka, Shuji Yoshizawa, Ken-ichi Hidai, Masashi Hamahira, Hiroshi Mizoguchi, and Taketoshi Mishima, "Convergence Analysis of Online Linear Discriminant Analysis", International Joint Conference on Neural Networks (IJCNN), 2000, *to appear.*

[14] K. Hiraoka, K. Hidai, M. Hamahira, H. Mizoguchi, T. Mishima, and S. Yoshizawa, "Successive Learning of Linear Discriminant Analysis: Sanger-type Algorithm", 15th International Conference on Pattern Recognition (ICPR), 2000, *to appear.*

[15] K. Hiraoka, K. Hidai, M. Hamahira, H. Mizoguchi, T. Mishima, and S. Yoshizawa, "Comparison of Various Algorithms for Online LDA", Proceedings of ROBOMEC'00 (in Japanese), pp. 1A1-77-110(1)–(2), 2000.

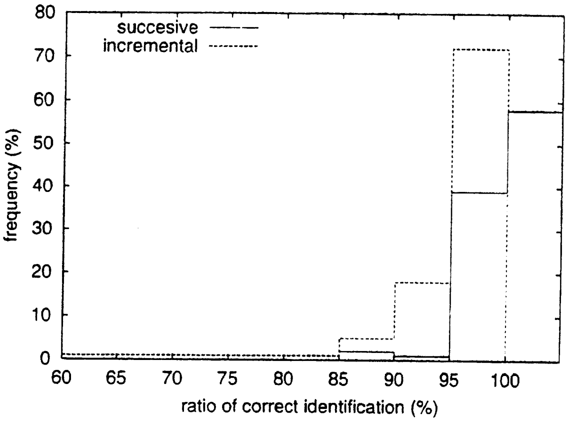


Figure 5: *Histogram of the final identification ratio after* 2500 *steps of learning. Horizontal axis: percentage of correct discrimination. Vertical axis: frequency in* 100 *trials.*

[16] Kazuyuki Hiraoka, Ken-ichi Hidai, Hiroshi Mizoguchi, Taketoshi Mishima, and Shuji Yoshizawa, "Fast algorithm for online linear discriminant analysis", International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC '2000), July 11-13, 2000, Hotel Paradise, Pusan, Korea, *to appear.*

[17] Kazuyuki Hiraoka, Ken-ichi Hidai, Hiroshi Mizoguchi, Taketoshi Mishima, and Shuji Yoshizawa, "One-parameter family of nonlinear dynamics for online linear discriminant analysis", 2000 International Symposium on Nonlinear Theory and its Applications (NOLTA2000), TU Dresden Conference Center, Dresden, Germany, September 17 - 21, 2000, *to appear.*