2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Incremental Face Recognition using Rehearsal and Recall Processes

Sangwook Kim, Rammohan Mallipeddi, and Minho Lee*

***Abstract*—Most of the machine learning algorithms particularly suffer from the plasticity-stability dilemma. In this paper, we propose a model that adopts two types of memories i.e. short-term memory (STM) and long-term memory (LTM), which share their information through control processes called rehearsal and recall to alleviate the dilemma. In addition, the proposed model tries to integrate the advantages of generative and discriminative classifiers by employing them in STM and LTM respectively. Experimental results show the importance of rehearsal and recall process in improving the performance of the algorithm.**

I. Introduction

Face recognition is currently one of the most actively researched areas of computer vision. In real-time face recognition applications, entire training data are not available beforehand. Instead the data arrive in bits and pieces at discrete time intervals. Therefore, it is important that the learning algorithm incorporates new incoming data while retaining previously learnt knowledge. Traditional batch-type learning algorithms are not suitable for an embedded platform in which the computational and memory resources are limited. To counter this problem, incremental learning algorithms have been proposed. Such algorithms obtain continuous knowledge from a large number of data samples available at different time intervals [1], [2]. Generally, in incremental algorithms, there exists a dilemmatic relationship between stability and plasticity [3]-[7] where plasticity refers to the capability to extract and accumulate new knowledge, whereas stability refers to the capability to preserve previous knowledge during the accumulation of new knowledge.

Unlike machine learning algorithms, human learning takes place discretely over time by successfully retaining the previously acquired knowledge. Humans retain previous knowledge through a process called rehearsal and can easily incorporate new information with previously learned information through recall [8]. Recently, many researchers have tried to overcome the limitations of current state-of-the-art algorithms by mimicking the mechanism of the human brain [9], [10]. According to [11], [12], human memory can be separated into short-term memory (STM) and long-term memory (LTM). As the name explains, STM can store information only for a short duration (15~30 sec), while the LTM can retain the learnt information for a long time [13]. Rehearsal enables the information present in STM to be transferred to LTM for longer retention; whereas recall enables a new piece of information to be efficiently combined with existing information by re-accessing past information which is encoded and stored in the brain.

On the other hand, pattern classification methods including face recognition algorithms can be categorized as generative and discriminative model-based approaches. And these two approaches possess complementary strengths and weaknesses with regards to pattern recognition problems [14]. Generative models are adequate for incremental learning of data because of their flexibility while discriminative models provide robust classification of categories since they make use of the relationship between observed data and target variable to discriminate. And in discriminative models, it is hard to make the sample data coincident with certain target variable, whereas in generative models the data can be re-generated to have certain likelihood value [15].

In this paper, we propose a new incremental face recognition algorithm that uses the concept of STM/LTM dichotomy combined with two related processes, rehearsal and recall to efficiently enhance face recognition performance. In the proposed model, STM using generative models and LTM using discriminative models communicate with each other through rehearsal and recall mechanisms in order to overcome the plasticity-stability dilemma. The proposed method can alleviate not only the insufficient data problem in the STM learning process but also the robustness issue in LTM. Moreover, due to the less computational requirements of incremental learning for the generative models, STM module can be implemented onto an embedded platform while LTM is implemented onto a server. Since the proposed idea is general, any existing state-of-the-art method, including cascaded SVM, multi-class SVM (MC-SVM) [16] and classification algorithms based on sparse coding [17], [18] can be incorporated into LTM.

The remainder of this paper is organized as follows: Section 2 describes the structure of the proposed model, and experimental results are given in Section 3. Finally, concluding remarks are presented in Section 4.

S. Kim, R. Mallipeddi, and M. Lee are with the School of Electronics Engineering, Kyungpook National University, 1370 Sankyuk-Dong, Puk-Gu, Taegu 702-701, Republic of Korea (e-mail: sangwook@ee.knu.ac.kr, mallipeddi.ram@gmail.com, corresponding author to provide phone: +82-53-950-6436; fax: +82-53-950-5505; e-mail: mholee@knu.ac.kr).

This work was supported by the Industrial Strategic Technology Development Program (10044009, Development of a self-improving bidirectional sustainable HRI technology for 95% of successful responses with understanding user's complex emotion and transactional intent through continuous interactions) (50%) funded by the Korea Ministry of Knowledge and Economy (MKE) and the Korea Evaluation Institute of Industrial Technology (KEIT), and the Original Technology Research Program (2013034988) for Brain Science through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (50%).

II. THE PROPOSED MODEL

The proposed model tries to overcome the inherent plasticity-stability dilemma of incremental learning by mimicking the process of information flow in the human brain. In the proposed model, STM improves plasticity by adapting faster to new incoming information, whereas LTM improves stability by preserving the learnt information for a longer period of time.

To enable fast incremental learning, STM employs a generative model based on Gaussian mixture models (GMMs), which is trained incrementally with features extracted by incremental two-dimensional and two-directional principal component analysis ($I(2D)^2$ PCA) [2]. Usually feature extraction techniques are used to reduce the dimensionality by removing the redundant information present in the image data. And furthermore, feature extraction increases the robustness of the model. Principal component analysis (PCA) is a popular feature extraction technique [19]. But for high-dimensional data, the covariance matrix of the data becomes very large in PCA and takes long time to calculate related operations. To solve this problem, $(2D)^2$ PCA was introduced [20]. Because the covariance matrix in the $(2D)^2$ PCA is much smaller than the covariance matrix of conventional PCA, $(2D)^2$ PCA is much faster than PCA. However, to perform efficient on-line learning, where the data arrives at discrete time intervals, incremental techniques are required. $I(2D)^2$ PCA is incremental feature extraction scheme and doesn't require the entire dataset to learn the new data [2]. To provide robustness, LTM uses discriminative models such as cascaded SVM, multiclass SVM and sparse coding schemes (*l*1 and *l*2). Hence, the proposed model tries to combine the advantages of generative and discriminative models.

As shown in Fig. 1, STM and LTM have feature extraction parts as preprocessing steps. In training phase, the proposed model is incrementally trained to incorporate data corresponding to different classes. Initially, for the given class data, principal axes are constructed features are obtained by projection of data onto those axes. Then, Gaussian mixtures of STM are formed with those features. The combination of principal axes and Gaussian mixtures, referred to as rehearsal data, is provided to LTM for batch-type training and the information stored in STM is reset. In the next stage of the training process, when a new input belonging to the already trained class comes in, it is added to LTM. If LTM classifies the input correctly as rehearsal data, it is considered redundant. However, if LTM misclassifies the input, it is added to STM, which can construct new Gaussian mixtures to incorporate the misclassified data. Through the recall process, LTM also provides the reconstructed data needed to form new Gaussian mixtures in order to overcome the insufficient data problem.

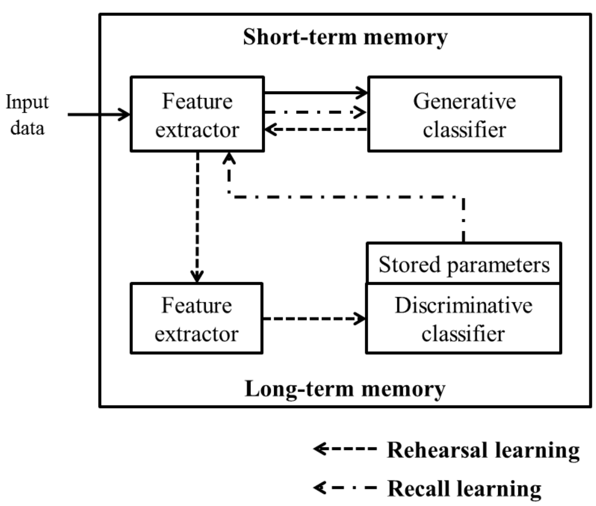

Fig. 1. The structure of the proposed model

Therefore, STM adapts faster to new incoming data and transfers the information to LTM for longer retention through rehearsal; whereas LTM provides the reconstructed data to STM through recall for accurate learning. From above reasoning, it can be observed that the process occurring in STM is computationally less expensive, whereas the process taking place in LTM is computationally expensive. As there is continuous communication between STM and LTM, the two modules can be implemented onto two different platforms depending on the application.

A. Short-term memory

As shown in Fig. 2, STM includes $I(2D)^2$ PCA [2] for incremental feature extraction and GMMs.

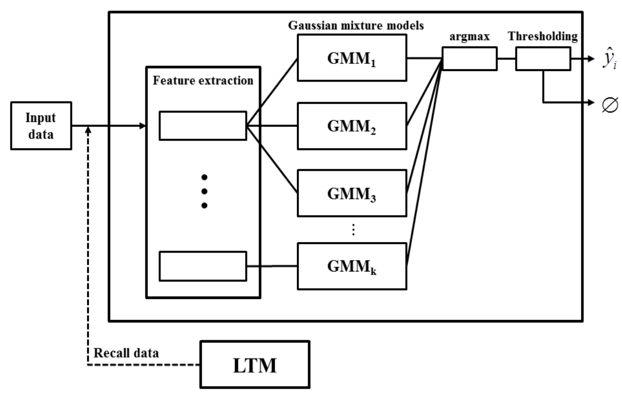

Fig. 2. The structure of the short-term memory

The GMM classifiers used in STM are computationally inexpensive and can model the distributions of input features with statistical probability. Generative models attempt to understand the basic formation of individual classes and can represent the independent relationships in the data. As generative models describe the distribution of the individual classes, it is easy to update a generative model by detecting changes in distribution. The distribution of the training data can be estimated by the most popular and well-established maximum likelihood estimation using the expectation-maximization algorithm [21].

The procedure of the training of STM can be summarized as Fig. 3.

In the training phase, for given class data, STM uses the first

The procedure of the training of STM:
input: dataset X of class y, STM S, LTM L
begin
  $X_{stack} := \{x_i \mid x_i \in X\}$;
  if $y \in L$
    then
      $X_L := \{x_i \mid y_i = y, x_i \in Recall(L)\}$;
      $X_{stack} := \{X_{stack}, X_L\}$;
    end
  end
  $A_{prev} :=$ previous feature extractor;
  $A_y :=$ I(2D)^{2}PCA with X_{stack} based on A_{prev};
  $U_{stack} :=$ extracted feature of X_{stack} using A_y;
  Build GMM of U_{stack};
end

Fig. 3. The procedure of training a class in STM

m samples of the class and obtains principal axes. The obtained features are used to train Gaussian mixtures. By using the GMM and principal axes, the rehearsal data are generated by inverse projection and are passed to LTM for training. After using the first *m* samples corresponding to a class, when n new samples corresponding to the same class that cannot be perfectly classified by LTM are presented, then LTM provides STM information regarding the principal axes. The information provided by LTM to STM is referred to as recall data. STM updates the principal axes to incorporate the misclassified data together with the recall data from LTM corresponding to the class. In other words, STM updates the axes provided by LTM, which are not representative for learning the class using I(2D)^2 PCA. Updated features are used to re-build the existing GMMs or form a new GMM if previously constructed GMMs are insufficient to represent the entire class. In STM, if available data samples are insufficient to learn the GMM i.e. $n<m$, then additional m-n data samples needed are supplied by LTM through the recall process.

When an input corresponding to a new class, which is not learnt by LTM, is presented to the model, LTM identifies it as a new class as there are no principal axes or Gaussian mixtures corresponding to the new class. The input is passed to STM, which extracts the features and trains a Gaussian mixture for the class as described above.

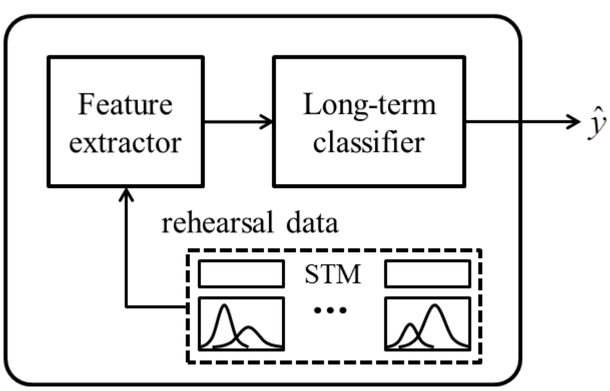

Fig. 4. The structure of the long-term memory

Therefore, by using I(2D)^2 PCA and Gaussian mixtures, STM can incrementally incorporate the information related to new classes of the training data. In addition, if an input corresponding to a learnt class is not covered properly in LTM, then STM incrementally learns the data and updates LTM by rehearsal.

B. Long-term memory

In the proposed model, LTM employs discriminative classifiers, which tries to learn the explicit decision boundary that maximizes the distance between samples of the positive and negative classes.

As shown in Fig. 4, LTM has a feature extraction module and a classification module. In LTM, any feature extraction method and discriminative model can be used. For example, the Histogram of Oriented Gradients (HoG) method can be used [22] as a feature extractor while cascaded SVM, multiclass SVM, or sparse coding schemes (*l*1 & *l*2) can be used as a classifier.

The procedure of training a class in LTM:
input: STM S, LTM L, class label y
begin
  $X_S := \{x_i \mid i = 1, ..., m, x_i \in Rehearsal(S)\}$;
  if $y \in L$
    then
      $X_{Lp} := \{x_i \mid y_i = y, x_i \in Recall(L)\}$;
      $X_{Ln} := \{x_i \mid y_i \neq y, x_i \in Recall(L)\}$;
      $(U_p, U_n) :=$ Extracted features of $(\{X_S, X_{Lp}\}, X_{Ln})$;
    else
      $X_{Ln} := \{x_i \mid x_i \in Recall(L)\}$;
      $(U_p, U_n) :=$ Extracted features of (X_S, X_{Ln});
    end
  end
  Train and update discriminative classifiers of L with (U_p, U_n);
end

Fig. 5. The procedure of training a class in LTM

The overall process of training a class in LTM is described in Fig. 5. As shown in Fig. 4 and 5, LTM is trained using the rehearsal data from STM instead of the original data. Therefore, in the test phase, a given test sample is not directly tested on LTM since it is trained with rehearsal data. The sample is projected and inverse-projected similar to rehearsal data using recently learned axis stored in LTM since a recent axis better represents the information corresponding to all classes. The obtained data through the rehearsal process, known as blurred data, are used for feature extraction using LTM's feature extractor.

C. Rehearsal and recall processes

Rehearsal is a process that is necessary to overcome the forgetting problem [6] and improve stability as well as robustness. As mentioned before, rehearsal data from STM are needed to train the classifiers in LTM. As the GMMs in the STM model input feature distributions, the rehearsal data could be reconstructed within the range of a certain confidence interval. The rehearsal process aids LTM in obtaining a decision boundary that can discriminate the new

Fig. 6.Cropped image samples of AR face dataset

class and the learnt classes by providing the rehearsal data. In addition, the rehearsal process can provide the data required without storing an entire dataset, since features are rehearsed using distribution parameters (mean vector and covariance matrix) of mixtures. Rehearsed features are reconstructed to the same dimensionality of the input face by inverse projection using principal axes, which are coupled with Gaussian mixtures. The rehearsal data look blurred and perturbed due to the loss in feature extraction and small variation in reconstruction. However, a little noise or distortion of the training data could enhance the robustness and generalization performance of the model. Furthermore, since blurred images are useful for robust recognition of deformed patterns [23], it enhances the robustness of LTM.

During the training, if the axes present in LTM are not representative enough to classify the incoming data corresponding to a class, then the axes and Gaussian mixtures should be updated to incorporate the data. Updating the axes and Gaussian mixtures occurs in STM. To properly update, the information present in the new samples should be combined with the information stored in the LTM. The information present in LTM is passed to STM through the recall process, which can efficiently alleviate the lack of training data problem in the incremental learning process of STM.

Fig. 7. Resized samples of training set and corresponding rehearsal data (left: original samples of training set, right: corresponding rehearsed data)

III. EXPERIMENTAL RESULTS

To show the effectiveness of the rehearsal and recall processes, AR face dataset [24] with sample images presented in Fig. 6 is used. As shown in Fig. 6, there are some variations such as facial expressions, illuminative changes and occlusions.

We use faces of 50 individuals with 13 gray-scaled faces per individual class. Each face is detected and cropped from the original picture by using Viola-Jones face detector [25] and resized to 32x32 resolution. For feature extraction of LTM with HoG, 10 orientation bins are used and the size of the cell is set to 4. As a result, the length of a feature vector becomes 1,960 ((32/4-1)2x4x10). The criteria for incremental learning of STM with $I(2D)^2$ PCA is set to 90% level [2].

The effectiveness of rehearsal is also evaluated using several state-of-the-art batch-type learning algorithms, cascaded SVM, MC-SVM, and sparse representation (*l*1 and *l*2), embedded in LTM.

As mentioned above, recall and rehearsal data are blurred and seem like noisy versions of the original data. Fig. 7 shows two resized samples of the training dataset (left) and corresponding rehearsed data (right). Rehearsed data seem to have lost fine details but they are robust to small deformations.

TABLE I
PERFORMANCE OF THE STM WITH/WITHOUT RECALL PROCESS FOR EACH CLASS

	STM without recall	STM with recall
Average accuracy	90.75 %	**97.75 %**

We compared the classification accuracy of STM with and without recall data in addition to the resized data. Five sampled images among the 13 images are used for training and the remaining eight faces are used for testing. The capacity of STM is limited by three. So three of five training images are used for initial batch modeling and the remaining two images are used for incremental learning. Performances of STM for classifying a single class are presented in Table 1.

The results show the enhancement of the performance using the recall process. Therefore, it is obvious that the use of recall data in STM training improves recognition

TABLE II
TEST PERFORMANCE OF PROPOSED METHOD FOR THE TRAINING OF 50 CLASSES

Methods	Without rehearsal	With rehearsal
Cascaded SVM	83.25 %	**86.75 %**
MC-SVM	80.25 %	**83.75 %**
$l1$	85.15 %	**89.5 %**
$l2$	81.75 %	**87.5 %**

performance by solving the insufficient data problem during incremental learning. Table 2 shows the effectiveness of the rehearsal process. As shown in Table 2, the rehearsal process using blurred data enhances the performance of all classifiers.

Therefore, from the experimental results, it has been demonstrated that rehearsal and recall are capable of incrementally incorporating information while giving better classification performance.

IV. CONCLUSION

We propose an incremental face recognition system comprising STM and LTM modules, which collaborate with each other by sharing their learnt information through natural processes referred to as rehearsal and recall. As a result, the proposed method can learn and model new data flexibly using STM and also stably classify using the discriminative classifiers of LTM. The experimental results showed the effectiveness of rehearsal and recall process.

Conventional face recognition algorithms are not suitable for embedded applications [26]. In the proposed system, STM can be implemented on an remote embedded system with limited memory and CPU speed, whereas LTM can be implemented on server and communicate the extracted results and information with STM through a wireless communication. Therefore, the proposed model is not only effective in alleviating the forgetting problem of real-world applications but also can be easily implemented onto an embedded platform.

As a future work, we would like to combine the proposed rehearsal and recall processes with a more advanced classifier in order to enhance the accuracy of facial recognition. And for the modeling of the data distribution and sampling procedures of the rehearsal and the recall process, we will consider the various techniques based on the uncertainty measure of class probability output networks [27] or information criteria such as Akaike information criterion (AIC) [28], the Bayesian information criterion (BIC) [29], and modulus of continuity information criterion (MCIC) [30].

ACKNOWLEDGMENT

This research was supported by the Industrial Strategic Technology Development Program (10044009, Development of a self-improving bidirectional sustainable HRI technology for 95% of successful responses with understanding user's complex emotion and transactional intent through continuous interactions) (50%) funded by the Korea Ministry of Knowledge and Economy (MKE) and the Korea Evaluation Institute of Industrial Technology (KEIT), and the Original Technology Research Program (2013034988) for Brain Science through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (50%).

REFERENCES

[1] Z. Haitao, Y. Pong Chi, and J. T. Kwok, "A novel incremental principal component analysis and its application for face recognition," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, pp. 873-886, 2006.

[2] C. Yonghwa, T. Tokumoto, L. Minho, and S. Ozawa, "Incremental two-dimensional two-directional principal component analysis (I(2D)²PCA) for face recognition," in *2011 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 1493-1496.

[3] S. Grossberg, "Competitive learning: From interactive activation to adaptive resonance," *Cognitive Science*, vol. 11, pp. 23-63, 1987.

[4] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures," *Neural Networks*, vol. 1, pp. 17-61, 1988.

[5] R. M. French, "Pseudo-recurrent Connectionist Networks: An Approach to the 'Sensitivity-Stability' Dilemma," *Connection Science*, vol. 9, pp. 353-380, 1997/12/01 1997.

[6] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, pp. 123-146, 1995.

[7] A. Robins, "Consolidation in Neural Networks and in the Sleeping Brain," *Connection Science*, vol. 8, pp. 259-276, 1996/06/01 1996.

[8] L. L. Jacoby and W. H. Bartz, "Rehearsal and transfer to LTM," *Journal of Verbal Learning and Verbal Behavior*, vol. 11, pp. 561-565, 1972.

[9] K. Nakamura, M. Yamazaki, and K. Ohzawa, "Face recognition evaluation of an association cortex-entorhinal cortex hippocampal formation model by successive learning," in *SICE 2001. Proceedings of the 40th SICE Annual Conference. International Session Papers*, 2001, pp. 214-219.

[10] S. Ozawa, S. L. Toh, S. Abe, S. Pang, and N. Kasabov, "Incremental learning of feature space and classifier for face recognition," *Neural Networks*, vol. 18, pp. 575-584, 2005.

[11] R. C. Atkinson and R. M. Shiffrin, "Human memory: A proposed system and its control processes," *The psychology of learning and motivation: Advances in research and theory*, vol. 2, pp. 89-195, 1968.

[12] A. D. Baddeley and E. K. Warrington, "Amnesia and the distinction between long-and short-term memory," *Journal of Verbal Learning and Verbal Behavior*, vol. 9, pp. 176-189, 1970.

[13] J. Jonides, R. L. Lewis, D. E. Nee, C. A. Lustig, M. G. Berman, and K. S. Moore, "The mind and brain of short-term memory," *Annu. Rev. Psychol.*, vol. 59, pp. 193-224, 2008.

[14] S. Chen, B. C. Lovell, and T. Shan, "Combining Generative and Discriminative Learning for Face Recognition," in *Digital Image Computing: Techniques and Applications, 2005. DICTA'05. Proceedings 2005*, 2005, pp. 5-5.

[15] I. Ulusoy and C. M. Bishop, "Generative versus discriminative methods for object recognition," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, pp. 258-265 vol. 2

[16] L. Zhao, Y. Song, Y. Zhu, C. Zhang, and Y. Zheng, "Face recognition based on multi-class SVM," in *Control and Decision Conference, 2009*. CCDC '09. Chinese, 2009, pp. 5871-5873.

[17] S. Qinfeng, A. Eriksson, A. van den Hengel, and S. Chunhua, "Is face recognition really a Compressive Sensing problem?," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, pp. 553-560.

[18] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and M. Yi, "Robust Face Recognition via Sparse Representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, pp. 210-227, 2009.

[19] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91, IEEE Computer Society Conference on*, 1991, pp. 586-591.

[20] D. Zhang and Z.-H. Zhou, "$(2D)^2$ PCA: Two-directional two-dimensional PCA for efficient face representation and recognition," *Neurocomputing*, vol. 69, pp. 224-231, 2005.

[21] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, pp. 1-38, 1977.

[22] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conf. Computer Vision and Pattern Recognition,* 2005, pp. 886-893 vol. 1.

[23] K. Fukushima, "Neocognitron trained with winner-kill-loser rule," *Neural Networks*, vol. 23, pp. 926-938, 2010.

[24] A. M. Martinez, "The AR face database," *CVC Technical Report*, vol. 24, 1998.

[25] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, pp. 137-154, 2004.

[26] H. Kondo, M. Nakajima, M. Bober, K. Kucharski, O. Yamamoto, and T. Shimizu, "Implementation of Face Recognition Processing Using an Embedded Processor," *J Rob Mechatron*, vol. 17, pp. 428-436, 2005.

[27] H.-G. Kim, R. M. Kil, and S.-Y. Lee, "Uncertainty Measure for Selective Sampling Based on Class Probability Output Networks," in *Neural Information Processing*, 2011, pp. 774-781.

[28] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Second international symposium on information theory*, 1973, pp. 267-281.

[29] G. Schwarz, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, pp. 461-464, 1978.

[30] I. Koo and R. M. Kil, "Model selection for regression with continuous kernel functions using the modulus of continuity," *JMLR, Special Topic on Model Selection*, vol. 9, pp. 2607-2633, 2008.