Alzheimer's Disease Detection Using Sparse Autoencoder, Scale Conjugate Gradient and Softmax Output Layer with Fine Tuning

Debesh Jha and Goo-Rak Kwon

***Abstract*—Accurate diagnosis of Alzheimer's disease (AD) plays an important role for patients care particularly in the early phase of the disease. Although numerous studies have used machine learning techniques for the computer aided diagnosis (CAD) of AD, an obstacle in the diagnostic efficiency was shown in the former methods, due to deficiency of effective strategies for characterizing neuroimaging biomarkers and limitation in choosing the learning models. In this study, we propose a deep learning model, which consists of sparse autoencoders, scale conjugate gradient (SCG), stacked autoencoder and a softmax output layer, to subdue the bottleneck and support the analysis of AD and healthy controls. Compared to the former workflows, out technique requires less labeled training examples and minimal prior knowledge. The proposed methods provides a significant improvement in classification output when compared to other studies, resulted in high and reproducible accuracy rates of 91.6% with a sensitivity of 98.09% and a specificity of 84.09%.**

***Index Terms*—Alzheimer's disease, sparse autoencoder, scale conjugate gradient, softmax layer.**

I. Introduction

Alzheimer's disease is a most familiar dementia type which mostly occurs in the elderly people. The AD is impaired cognitive functions and with memory loss. The subject of AD is seen to be increasing rapidly all over the world in every year time. At present, the cause of AD not well understood. However, the disorder is related to plague and tangles inside the brain.

Scholars have proposed different approaches based on computer vision and machine learning to support the diagnosis of AD by Magnetic resonance imaging (MRI). They presented the performance of their methods in their own experiments on AD/NC classification. The limitations of the former works are that they only took simple low-level features for e.g. cortical thickness and/or gray matter tissue volumes. In this work, we consider the whole MRI scans which holds hidden or latent high level information that can be beneficial to build a new robust model for the diagnosis AD/NC.

We assume that the prior workflows can be optimized by designing a new framework to efficiently represent the different stage of AD using different biomarkers. The conventional methods with shallow structures often results in feature repetition [1]. Hence, the deep data representation based learning is found to be much more effective than shallow architectures with regard to computational elements and parameters necessary for representing the new functions. Deep learning architectures draw out high-level features gradually via various layers of feature representation [2]. The high-level features are likely to be much more separable in classification issue because of the sequential transformations of feature space.

Some previous studies reported that multilayered learning structure was efficient in capturing shape dissimilarity of the brain area that corresponds with demographic and disease knowledge. Suk *et al*. [3] came up with the idea of stacked autoencoder (SAEs) for training every image modality; later, the trained high-level features were additionally passed to the multi-kernel support vector machine (MKSVM). Nevertheless, SVM alone could not also classify patients with a high performance when the training data is large.

In this paper, we utilized a new framework for early diagnosis of AD based on deep learning approaches, consisting of stacked sparse autoencoders and a softmax output layer. In addition, our technique is semi supervised that can be lengthened to use unlabeled training model, which are accessible and economical to obtain.

II. Methodology

A. Autoencoder

An autoencoder is a symmetrical neural network which can grasp the features in an unsupervised way by minimizing reconstruction errors [4]. The training process in an autoencoder is based on the optimization of a cost function. The cost function computes the error between the input x and its rebuilding at the output $\hat{x}$. An autoencoder constitutes of an encoder followed by decoder. The encoder and decoder can possess multiple layers; nevertheless for simplicity we consider that all of them have only one layer.

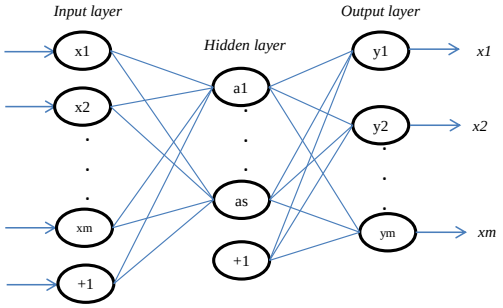

Fig. 1. Structure of an Autoencoder.

Manuscript received January 2, 2017; revised February 20, 2017.

Debesh Jha and Goo-Rak Kwon are with Dept. of Information and comm., Engineering Chosun University, Gwanju, Korea (email: debeshjha1@gmail.com, grkwon@chosun.ac.kr).

doi: 10.18178/ijmlc.2017.7.1.612

Autoencoder is demonstrated in Fig. 1. If the input data to an autoencoder is a vector $x \in R^{D_x}$, then the encoder projects the vector x to other vector $z \in R^{D^{(1)}}$ as follows:

$$z^{(1)} = h^{(1)}(W^{(1)}x + b^{(1)}), \quad (1)$$

where, the superscript (1) signifies the first layer. $h^{(1)} : R^{D^{(1)}} \to R^{D^{(1)}}$ is a transfer function for the encoder, $W^{(1)} \in R^{D^{(1)} \times D_x}$ is weight matrix, and $b^{(1)} \in R^{D^{(1)}}$ is a bias vector. Later, the decoder projects the encoded representation z away into an estimate of the initial input vector, x, as follows:

$$\hat{x} = h^{(2)}(w^{(2)}x + b^{(2)}), \quad (2)$$

where, the superscript (2) describes the second layer. $h^{(2)} : R^{D_x} \to R^{D_x}$ is the transfer function for the decoder, $w^{(1)} \in R^{D_x \times D^{(1)}}$ is a weight matrix, and $b^{(2)} \in R^{D_x}$ is a bias vector.

B. Sparse Autoencoders

Promoting sparsity of an autoencoder is feasible by incorporating a regularizer to the cost function. This regularizer is a function of the average output activation value of a neuron. The average output activation measure of a neuron i is described as:

$$\hat{\rho}_i = \frac{1}{n}\sum_{j=1}^{n} z_i^{(1)}(x_j) = \frac{1}{n}\sum_{j=1}^{n} h(w_i^{(1)\mathrm{T}} x_j + b_i^{(1)}), \quad (3)$$

where n is the total number of training examples. x_j is the *jth* training example, $w_i^{(1)T}$ is the ith row of the weight matrix $w^{(1)}$, and $b_i^{(1)}$ is the *ith* entry of the bias vector, $b_i^{(1)}$. A neuron is evaluated as 'firing', if the output activation value is high. A low output activation value indicates that the neuron in the hidden layer fires in reaction to a tiny number of the training examples. Accumulating a term to the cost function that forces the values of $\hat{\rho}_i$ to be low, boosts the autoencoder to study a representation, where every neuron in the hidden layer fires to a tiny number of training examples. That is, every neuron is trained by responding to few features that is only available in a small subset of the training examples.

C. Sparsity Regularization

Sparsity regularizer tries to accomplish a constraint on the sparsity of the output from the hidden layer. One such sparsity regularization term can be the Kullback-Leibler divergence.

$$\Omega_{\text{sparsity}} = \sum_{i=1}^{D^{(1)}} KL(\rho \| \hat{\rho}_i) = \sum_{i=1}^{D^{(1)}} \rho \log(\frac{\rho}{\hat{\rho}_i}) + (1-\rho)\log(\frac{1-\rho}{1-\hat{\rho}_i}) \quad (4)$$

We can define the desired value of the average activation value utilizing the Sparsity Proportion name-value pair argument while training an autoencoder.

D. L2 Regularization

When training a sparse autoencoder, it is possible to build the sparsity regularizer small by magnifying the values of the weights $w^{(l)}$ and reducing the values of $z^{(1)}$. Accumulating a regularization term on the weights to the cost function prevents it from occurring. This term is called the L_2 regularization term and is defined by:

$$\Omega_{\text{weights}} = \frac{1}{2}\sum_{l}^{L}\sum_{j}^{n}\sum_{i}^{k}(w_{ji}^{(l)})^2, \quad (5)$$

where L is the number of hidden layers, n is the number of observations (examples), and k is the number of variables in the training data.

E. Cost Function

The cost function for training a sparse autoencoder is an adjusted mean squared error function defined as follows:

$$E = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{k}(x_{kn} - \hat{x}_{kn})^2 + \lambda \times \Omega_{\text{weights}} + \beta \times \Omega_{\text{sparsity}}, \quad (6)$$

Here, λ is the coefficient for the L2 regularization term and β is the coefficient for the sparsity regularization term. We can designate the values of λ and β by utilizing the L2 Weight Regularization and Sparsity Regularization name-value pair arguments, respectively, while training an autoencoder.

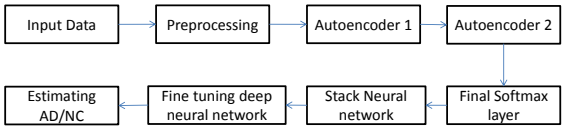

Fig. 2. Block diagram of the proposed system.

F. Softmax Layer

For the classification of AD, a softmax output layer is summed on the tip of the trained autoencoder stack including only former hidden layers [5], [6]. The Softmax layer utilizes a different activation function that might have nonlinearity, dissimilar from the one used in previous layers. The Softmax activation function is given by

$$h_i^l = \frac{e^{w_i^l h^{l-1} + b_i^l}}{\sum_j e^{w_i^l h^{l-1} + b_i^l}} \quad (7)$$

Here, w_i^l is the i-*th* row of w^l and b_i^l is the *i-th* bias term of final layer. We can employ h_i^l as an estimator of $P(Y = i / x)$. Moreover, Y is the concerned label of input data vector X. In our study, we have two output neurons at the Softmax layer can be explained as the probabilities of identifying the subjects with NC or AD. Alike to the operation of training the Deep Belief Net (DBN) [7], furthermore, all the parameters can be fine-tuned in the network with regard to the complete classification loss by unfolding every autoencoders and employing the back

propagation algorithm on the whole network [5], [8].

III. Experimental Result and Discussion

We implemented the deep learning framework described in this paper using Matlab 2015b environment on Intel(R) core (TM) i5-7500 CPU, 3.30 GHz processing speed, and 16 GB RAM, Microsoft windows 7. Readers can repeat our results on any machine where MATLAB is a platform. The block diagram of the proposed system is shown in Fig. 2.

A. Database

In our study we have downloaded the structural MRI data from open Access Series of Imaging Studies (OASIS) database. The OASIS is an attempt at making MRI data set of the brain freely available to the scientific community. OASIS covers two types of data: cross-sectional MRI data and longitudinal MRI data. We used cross sectional MR image in our study because the main purpose of our study is developing an automated tool to detect AD, which is not related to longitudinal data in which AD subjects were gathered together over long duration. In our study, we recruited 51 AD subjects (35with CDR=0.5 and 16with CDR=1) out of 100 having dementia and 44 normal subjects out of 98 normal subjects. Only right-handed subjects are included constituting of both men and women. The database contains various details of the patient such as age, gender, education, socio-economic status, CDR, and MMSE. The statistical data used in our learning are demonstrated in Table I.

TABLE I: Statistical Data of the Subject Used in Our Learning

Factors	Normal	Very Mild & Mild AD
No. of Patients	44	51
Age	84.40 (76-96)	82.11 (76-96)
Education	3.34 (1-5)	3.13 (1-5)
Socioeconomic status	2.31 (1-5)	2.82 (1-5)
CDR (0.5/1)	0	35/16
MMSE	28.72 (25-30)	24.82 (18-30)

B. Image Preprocessing

The images are imported from the backup folder and subjects are extracted using ONIS software. We have selected 32 center slices from each subject containing most predictive information about the brain tissues. The same method is used to all the subjects (95 including both AD and NC). All of these obtained images are in PNG format and the dimension of the each slice is 176×256. The image is enlarged to 256×256 before further processing.

C. Training Autoencoder and Final Softmax Layer

At first we train a sparse autoencoder on the training data without using labels. Neural networks have weights randomly initialized before training. Therefore, the results from the training are different each time. To avoid this behavior, explicitly we set the random generator seed. We set the size of the hidden layer for the autoencoder. For the autoencoder that we are going to train, it is a good idea to make this smaller than the input size. The type of autoencoder that we are going train is a sparse autoencoder. This autoencoder uses regularizers to learn a sparse representation in the first layer. We can control the influence of these regularizers by setting various parameters: L2 Weight Regularization controls the impact of an L2 regularizer for the weights of the network (and not the biases). This should typically be quite small. Sparsity proportion is a parameter of the sparsity regularizer. It controls the sparsity of the output from the hidden layer. A low value for sparsity proportion usually leads to each neuron in the hidden layer specializing by only giving a high output for a small number of training examples. For example, if Sparsity Proportion is set to 0.1, this is equivalent to saying that each neuron in the hidden layer should have an average output of 0.1 over the training examples. This value must be between 0 and 1. The ideal value varies depending on the nature of the problem.

Now, the autoencoder is trained specifying the values for the regularizers which are mentioned above. After training the first autoencoder the second autoencoder is trained in a similar manner. The principal difference is that we utilized features that were produced from the first autoencoder as the training data in the second autoencoder. Also, the size of the hidden representation is decreased, so that the encoder in the second autoencoder learns a tinier representation of the input data.

We then train a softmax layer to categorize the obtained feature vectors. Unlike the autoencoders, training of the softmax layer is done in a supervised fashion utilizing the labels for the training data.

D. Forming a Stacked Neural Network

We have trained three independent components of a deep neural network in seclusion. At this stage, it might be beneficial to view the three neural networks that have been trained by us. These are autoenc1, autoenc2, and softnet. As described, the encoders from the autoencoders have been utilized for feature extraction. The encoders can be stacked from the autoencoders jointly with the softmax layer to design a deep network. The network is created by the encoders from the autoencoders and the softmax layer. Let's consider that the number of units in the input layer is equal to the dimension of the input feature vector. But the number of hidden units in the upper layers can be decided according to the description of the input, i.e., even extensive than the input dimension. The full deep network is formed, and we can estimate the outcome on the test set. To utilize images with the stacked network, we have to reshape the test images into a matrix. We can do this by stacking the columns of an image to form a vector, and then forming a matrix form these vectors. We can visibly see the outcome with a confusion matrix. The overall accuracy is seen in the bottom right-handed square of the matrix.

E. Fine Tuning the Deep Neural Network

The results for the deep neural network can be upgraded by performing backpropagation on the entire multilayer network. This method is often mentioned to as fine tuning. We fine tune the network by retraining it on the training data in a supervised approach. Before carrying out this, we have to reshape the training images into a matrix, as was accomplished for the test images.

F. Evaluation

The AD brains are considered as positive, while NC brains are regarded as negative. Ultimately, the network is trained and we obtain the output in terms of the confusion matrix. Now, we calculate the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The AD brains are supposed to true and normal ones to false, following common convention. The formula for accuracy, sensitivity and specificity are given below.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (9)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (10)$$

G. Comparison to State-of-the-Art Approaches

To further demonstrate the usefulness of our proposed method we compared it with to 7 state-of the-art approaches in Table II, which utilized dissimilar statistical settings, making comparison complex. The result in Table II shows that "US+SVD-PCA+SVM-DT [9]" achieved a classification accuracy of 90.00%, a sensitivity of 94.00%, and a specificity of 71%, "BRC+IG+SVM [10]" achieved an accuracy of 90.00%, a sensitivity of 96.88%, and a specificity of 77.78%. We can observe that their specificities are less when compared with other methods. Therefore, both of these methods are not considered. Likewise, "BRC+IG+VFI [15]" achieved an accuracy of 78%, sensitivity of 65.63% and specificity of 100%. Despite of high specificity, the accuracy and sensitivity obtained with this proposed method is low. Therefore, this method is also not considered under the review. "Curvelet+ PCA+KNN [11]" achieved classification accuracy of 89.47%, a sensitivity of 94.12%, and a specificity of 84.09%. Debesh Jha [11] achieved promising result with 4-level curvelet features, PCA and KNN. The proposed method obtained good results.

Other three state-of-the-art algorithm considered in our study outlines for both mean and standard deviation values. They also achieved convincing results. VBM + RF [12] achieved an accuracy of 89.0 ± 0.7%, a sensitivity of 87.9 ± 1.2%, and a specificity of 90.0± 1.1. The convincing result is achieved because of the voxel based morphometry (VBM). Actually, VBM has been frequently utilized to study the changes in the brain. Maguire (2000) signified that taxi driver will have larger back part of posterior hippocampus usually. Good (2001) figured that global gray matter decreased linearly with aging, but the global white matter remains identical. However, it needs an accurate spatial normalization; or the classification result may minimize notably. DF + PCA + SVM [13] acquired an accuracy of 88.27 ± 1.89%, a sensitivity of 84.93 ± 1.21%, and a specificity of 89.21 ± 1.63%. This technique is based on a new approach called displacement field (DF). This review estimates and measures the displace field of different slices between AD subjects and NC subjects. Similarly, "EB+WTT+SVM+RBF [14]" achieved an accuracy of 86.71± 1.93%, a sensitivity of 85.71 ± 1.91%, a specificity of 86.99 ± 2.30%.

Finally, our proposed method achieves an accuracy of 91.6%, a specificity of 98.09%, and a specificity of 84.09%. Considering classification accuracy, our approach outperforms 7 state-of-the-arts. We achieved outstanding sensitivity which if far better than the existing methods. We also achieved a promising specificity which is comparable to other state-of-the-art algorithms. Hence, our results are either outperforms or are comparable to the existing methods. Fig. 3 shows performance comparison of algorithms comparison.

TABLE II: ALGORITHM PERFORMANCE COMPARISON FOR MRI BRIAN IMAGES

Algorithm	Accuracy (%)	Sensitivity (%)	Specificity (%)
Proposed method	91.6	98.09	84.09
US + SVD-PCA + SVM-DT [9]	90	94	71
BRC + IG + SVM [10]	90.00	96.88	77.78
CURVELET+ PCA+ KNN [11]	89.47	94.12	84.09
VBM + RF [12]	89.0 ±0.7	87.9 ±1.2	90.0 ±1.1
DF + PCA + SVM[13]	88.27 ±1.89	84.93 ±1.21	89.21 ±1.63
EB + WTT + SVM + RBF [14]	86.71±1.93	85.71±1.91	86.99±2.30
BRC + IG + VFI [15]	78	65.63	100

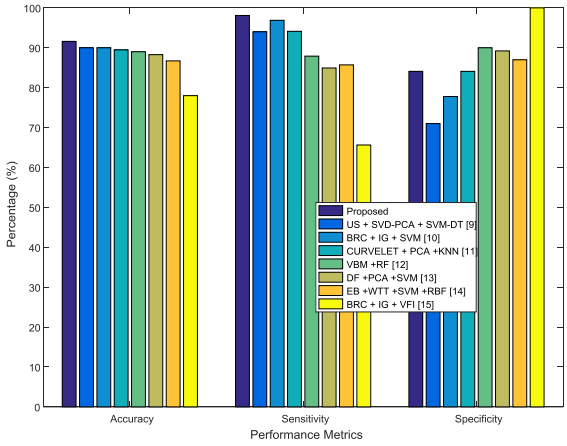

Fig. 3. Performance comparison of the proposed system.

IV. CONCLUSION

We proposed a new method for diagnosis of AD based on deep learning algorithms. This framework can distinguish AD with minimal clinical prior knowledge required. The proposed technique also performs dimensionality reduction and data fusion at the same moment. A performance gain is achieved with the binary classification. We have also showed that multi-layered parametric learning model can be applied on biomedical datasets with smaller size to extract high-level biomarkers. Based on MR data our method outperformed 7-state-of-the-art and SVM based framework. Therefore, we argue, that the proposed technique can be powerful method for computer-aided examination in other biomedical fields as well.

ACKNOWLEDGMENT

This research was supported by the Brain Research

Program through the National Research Foundation of Korea funded by the Ministry of Science, ICT & Future Planning (NRF-2014M3C7A1046050). The corresponding author is Goo-Rak Kwon (grkwon@chosun.ac.kr).

REFERENCE

[1] S. Liu, S. Liu, W. Cai, H. Che, S. Pujol, R. Kikinis, D. Feng, and M. J. Fulham, "Multimodal neuroimaging feature learning for multiclass diagnosis of Alzheimer's disease," I*EEE Trans. on Biomed. Engin*., vol. 62, pp. 1132-1140, Apr. 2015.

[2] Y. Bengioet, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 1798–1828, Aug. 2013.

[3] H.-L. Suk, S. W. Lee, and D. Shen "Latent feature representation with stacked auto-encoderfor AD/MCI diagnosis," *Brain Struct. Funct*., pp. 841-859, 2015.

[4] H. C. Shin, M. R. Orton, D. J. Collins, S. J. Doran, and M. O. Leach, "Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 1930–1943, Aug. 2013.

[5] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends® in Mach. Learn.*, vol. 2, pp. 1-127, Nov. 2009.

[6] J. S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," *Neurocomputing*, pp. 227-236, 1990.

[7] Y.-L. Boureau and Y. L. Cun, "Sparse feature learning for deep belief networks," *Adv. in Neural Infor. Proces. Syst*., pp. 1185-1192, 2007.

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504-507, 2006.

[9] Y. Zhang, S. Wang, and Z. Dong "Classification of Alzheimer disease based on structural magnetic resonance imaging by kernel support vector machine decision tree" *Prog. Electromagn. Res*., vol. 144, pp. 171–184, 2014.

[10] C. Plant, S. J. Teipel, A. Oswald, C. Bohm, T. Meindl, J. M. Miranda, A. W. Bokde, H. Hampe, and M. Ewers, "Automated detection of brain atrophy patterns based on MRI for the prediction of Alzheimer's disease," *Neu. Image*, vol. 50, pp. 162–174, March 2010.

[11] D. Jha and G. R. Kwon, "Alzheimer Disease detection in MRI using curvelet transform with KNN," *The Jour. Korean Inst. Infor. Tech.*, vol. 14, Aug. 2016.

[12] K. R. Gray, P. Alijabar, R. A. Heckemann, A. Hammers, and D. Rueckert, "Random forest-based similarity measures for multi-modal classification of Alzheimer's disease," *Neu. Ima*., vol. 65, pp. 167–175, Jan. 2013.

[13] Y. Zhang and S. wang, "Detection of Alzheimer's disease by displacement field and machine learning," *Peer J.*, vol. 3, Oct. 2015.

[14] Y. Zhang, Z. Dong, P. Phillips, S. Wang, J. Genlin, J. Yang, and T.-F. Yuan, "Detection of subjects and brain regions related to Alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning," *Front. Comput. Neurosci*., vol. 9, p. 66, Jun. 2015.

[15] C. Plant, S. J. Teipel, A. Oswald, C. Bohm, T. Meindl, J. M. Miranda, A. W. Bokde, H. Hampe, and M. Ewers, "Automated detection of brain atrophy patterns based on MRI for the prediction of Alzheimer's disease," *Neu. Image*, vol. 50, pp. 162–174, Mar. 2010.

Debesh Jha received his B.E in 2013 at Khwopa Engineering College, Purbanchal University, Nepal and is currently pursuing his M.S. in the Department of Information and Communication Engineering, Chosun University, Korea. Besides, he works as a research assistant at Chosun University. His research interest include big data analysis, bio-medical image processing, machine learning, pattern recognition, computer vision, and artificial neural network.

Goo-Rak Kwon received the Ph.D. degree at the Department of Mechatronic Engineering of KoreaUniversity in 2007 and the M.S. degree in the Schoolof Electrical and Computer Engineering at the Sung Kyun Kwan University in 1999. He has also served as the chief executive officer and director of Dalitech Co. Ltd. From May 2004 to Feb. 2007. He joined the Department of Electronic Engineering at Korea University where he was a Postdoc supporting the BK21 Information Technique Business from Mar. 2007 to Feb. 2008. At present, he is working an associate professor at Chosun University. He has contributed 80 articles to journals and conference proceedings. He also holds 20 patents on security of multimedia contents for digital rights management. He was a member in the IEEE, IEICE, and IS&T in the international institute. In the domestic institute, he had a member of signal processing society in the IEEK, KMMS, KIPS, and KICS. His interest research fields are A/V signal processing, video communication, and applications.