On the Use of a Parallel-Branch Subunit Model in Continuous HMM for improved Word Recognition

연속분포 HMM에서 평행분기 음성단위를 사용한 단어인식율 향상연구

Yong Kyoo Park*, Chong Kwan Un*

박 용 규*, 은 종 관*

ABSTRACT

In this paper, we propose to use a parallel-branch subunit model for improved word recognition. The model is obtained by splitting off each subunit branch based on mixture component in continuous hidden Markov model(continuous HMM). According to simulation results, the proposed model yields higher recognition rate than the single-branch subunit model or the parallel-branch subunit model proposed by Rabiner et al[1]. We show that a proper combination of the number of mixture components and the number of branches for each subunit results in increased recognition rate. To study the recognition performance of the proposed algorithms, the speech material used in this work was a vocabulary with 1036 Korean words.

요 약

단어인식의 성능향상을 위하여 평행분기 음성단위(subunit) 모델의 사용을 제안하였으며 연속 분포 HMM에서 이 모델은 각 음성단위를 확률분포함수 (mixture components)를 이용하여 분기시킴에 의해 얻어잔다. 제안된 방법을 사용한 결과에 따르면 기존에 제안된 평행분기 [1] 음성단위 모델이나 단일분기 모델보다 높은 인식율을 얻을 수 있었다. 본 연구에서는 각 음성단위에 대해 확률분포함수나 분기수의 적절한 결합을 통해 높은 인식율을 얻는데 이 1036 한국어 격리단어가 인식실험에 사용되었다.

I. Introduction

In the past few years, hidden Markov model (HMM) has been successfully applied to many speech recognition systems[2] [5]. The HMM-based speech recognition system uses training algorithm, which adjusts paramenters to obtain an approximation to the maximum-likelihood estimates (MLE) of HMM parameters [6] [7] [8] [9].

It has been shown that when a large amount of training data is available, the performance of a

*한국과학기술원 전기 및 전자공학과

접수일자 : 1995년, 3월 6일

speech recognizer can generally be improved by creating more than one template or statistical model for each of the recognition units [1]. In fact, the speech recognizer using either multiple templates or statistical models have been successfully applied to word recognition. In the course of our research on isolated word recognition, however, the use of a parallel-branch subunit model appeared more efficient for word recognition, because it provides more accurate representation of the variants of speech, such as sex, age, coarticulation and articulation manner. With this observation, we have studied an algorithm for creating a parallel-branch subunit model in continuous HMM, and tested its recognition performance of isolated Korean words.

It has been known that the conventional training scheme of maximum likelihood is easily combined with a parallel-branch subunit model [1] [2] [3]. One of the most important issues in speech recognition is how to initialize the HMMs for training. While the training algorithm guarantess improvement in every iteration, it does not guarantee the global maximum. If the initialization is poor, one may find a local maximum that results in poor recognition rate. For example, a zero probability in the parallel-branch subunit model will remain zero in every iteration. As will be seen in the next section, the proposed parallel-branch subunit model allows many degrees of freedom(i.e., many continuous parameters which will be estimated from observations) with respect to the amount of training data. Thus, a more sophisticated form of initialization is needed. According to our experimental results, good initialization becomes more important when the number of parallel branchs increases. We show that the initialization approach using the maximizing likelihood method proposed by Rabiner et al. [1] does not work well in the case of the parallel-branch subunit model.

In the following, we first describe various algorithms of generation and training of the parallel-branch subunit model in Section 2. In this section we develop a procedure which initializes a new parallel-branch subunit model by splitting off each subunit branch with Gaussian mixture components in continuous HMM. Next in Section 3, we present the speech database and conditions for experiments, and discuss experimental results. Finally, we make a conclusion in Section 4.

II. Generation and Training of Multiple Models

The speech subunit used here is represented by a single first-order, left-to-right, hidden Markov model having 3 states, with self-and forward transitions without skipping. Although either the segmental k-means algorithm or the Baum-Welch algorithm can be used, we used the former. For every state, the transition probability of moving from state s_j at time $t+1$ is represented by a_{ij}. And, for the spectral density within s_j of a continuous HMM, the observation pdf in s_j is given by

$$b_j(\mathbf{O}_t) = \sum_{m=1}^{M} c_{jm} N(\mathbf{O}_t, \mu_{jm}, \Sigma_{jm})$$

$$= \sum_{m=1}^{M} c_{jm} (2\pi)^{-N/2} |\Sigma_{jm}^{-1}|^{1/2}$$

$$\exp\left[-\frac{1}{2} (\mathbf{O}_t - \mu_{jm})^T \Sigma_{jm}^{-1} (\mathbf{O}_t - \mu_{jm}) \right] \qquad (1)$$

where M is the number of Gaussian mixture components ; c_{jm} is the mixture gain for the mth mixture : $N(\mathbf{O}_t, \mu_{jm}, \Sigma_{jm})$ denotes the Gaussian probability density function for an observation vector $\mathbf{O}_t$ with the mean vector μ_{jm} and the covariance matrix Σ_{jm} ; N is the dimension of each observation vector ; superscript T denotes vector transpose ; and $|\cdot|$ represents a determinant. In Fig. 1, a single-branch subunit model and a parallel-branch subunit model are shown, where the superscript indicates the branch number.

2.1 The splitting algorithm based on maximum likelihood method

Rabiner et al. proposed a splitting procedure of subunit branches whose likelihood scores are rela-

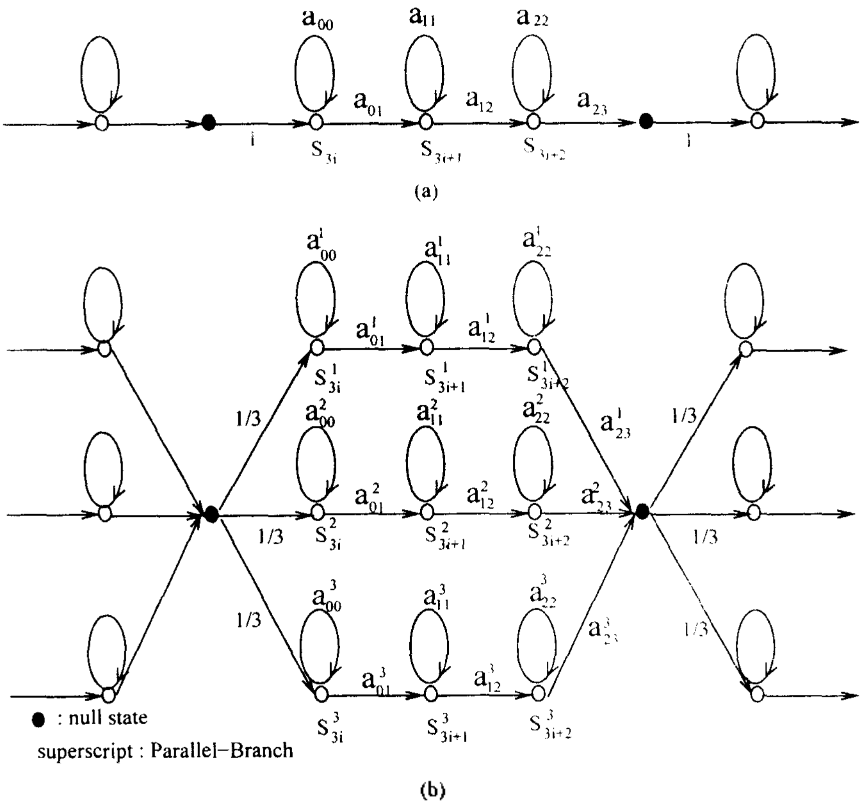


그림. 1. (a) Single-branch subunit model and
(b) Parallel-branch subunit model.

tively low. The objective of maximum likelihood model estimation is to obtain a set of parallel-branch model parameters which gives the maximum likelihood based on the given set of training data. This algorithm is basically related to the one-mixture continuous HMM. The procedure for splitting off subunit branchs with low likelihood scores and creating new ones from these subunit branches is as follows :

1. Training is done on a set of subunits with one branch and one mixture until convergence is reached.

2. For each subunit, one of the branches per each subunit associated with the lowest likelihood score is used to initialize an additional branch for that subunit by splitting it off.

3. The training procedure is repeated on the new set of parallel-branch models until convergence is achieved.

4. The above procedure(i.e., steps 2 and 3) is iterated until the desired number of branches per subunit is obtained.

After those branches for each subunit are obtained, the number of mixture components, K, can be estimated by the vector quantization(VQ) procedure on segmented data. And then the segmental k-means training procedure is used on the parallel-branch subunit model until convergence is reached.

The above initialization method is based on the concept of likelihood maximization, starting from a small set of subunits and iteratively splitting off the subunit branches with low likelihood scores. The above procedure does not provide good initial representations of the parallel-branch subunit model in the training set. The approach of maximizing likelihood tends to give an improved model of subunits for which the initial estimates were generally quite good, but tend to lead to poor estimates for subunits which are poorly represented by the initial model.

2.2 The splitting algorithm based on mixture components

A block diagram for generation and training of a parallel-branch subunit model with M branches and K mixtures is illustrated in Fig. 2. For creation of the parallel-branch subunit model from a single-branch subunit model with M mixtures, the number of initial mixture components is set to the desired number of parallel branches per subunit.

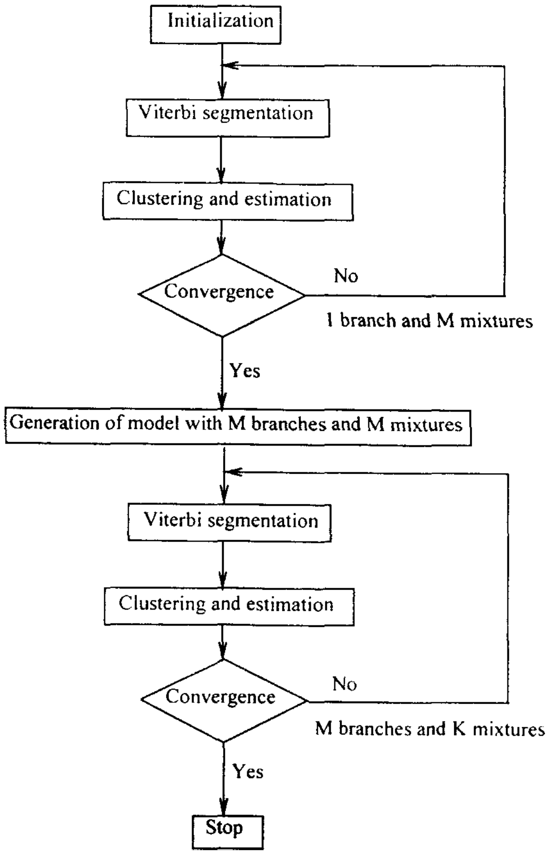

Initialization
Viterbi segmentation
Clustering and estimation
Convergence
No
1 branch and M mixtures
Yes
Generation of model with M branches and M mixtures
Viterbi segmentation
Clustering and estimation
Convergence
No
M branches and K mixtures
Yes
Stop

그림 2. Flow chart of generation and training of a parallel-branch subunit model based on mixture components.

For the continuous HMM, the model parameters, i.e., a_{ij} and$(\mu_{jm}, \Sigma_{jm}, c_{jm})$, can be estimated using the segmental k-means algorithm as follows.

Stpe 1 : Initialization

Linearly segment all training vectors into phoneme units and HMM states. By chustering, the parameters$(\mu_{jm}, \Sigma_{jm}, c_{jm})$ are initialized.

Stpe 2 : Segmentation

The HMM parameters estimated in Step 1 or Step 3 are used to (re)segment each training utterance into phoneme units and HMM states via Viterbi decoding. The transition probabilities are obtained from the segmentation.

Stpe 3 : Clustering and estimation

All the observation vectors corresponding to a particular state of each phoneme model are partitioned into M clusters using the standard VQ design method, and the parameters$(\mu_{jm}, \Sigma_{jm}, c_{jm})$ are estimated as

$$\mu_{jm} = \frac{1}{L_{jm}} \sum_{\mathbf{O}_t \in V_{jm}} O_t \tag{2}$$

$$\Sigma_{jm} = \frac{1}{L_{jm}} \sum_{\mathbf{O}_t \in V_{jm}} (\mathbf{O}_t - \mu_{jm})(\mathbf{O}_t - \mu_{jm})^T \tag{3}$$

$$c_{jm} = \frac{L_{jm}}{\sum_{k=1}^{M} L_{jk}} \tag{4}$$

where V_{jm} denotes a set of vectors that have been partitioned to the mth mixture of state j, and L_{jm} is the number of members in V_{jm}.

Step 4 : Steps 2-3 are repeated until the convergence condition is satisfied.

After convergence is achieved, we can use the i-th trained 3-state subunit model to obtain i-th parallel-branch subunit model. The three states with self-loops can represent a transition into the subunit, a steady-state portion, and a transition out of the subunit, respectively, where they are denoted by s_{3i}, s_{3i+1}, and s_{3i+2}. The basic idea of splitting is that the steady-state portion contributes the largest amount to the likelihood of the three states. In Fig. 1., a null state is skipped, and therefore does not produce the output pdf. Fig. 2 shows the creation of a parallel-branch subunit model parameters except for the pdf of the steady-state part are set to the parameters of the single subunit model to be split off as shown in Fig. 1. Only the output pdf of the steady state portion is set as follows.

$$B_{3i}^{1}(\mathbf{O}_t) = B_{3i}^{2}(\mathbf{O}_t) = \cdots = B_{3i}^{m}(\mathbf{O}_t) = \cdots = B_{3i}^{M}(\mathbf{O}_t) = B_{3i}(\mathbf{O}_t) \qquad (5)$$

$$B_{3i+2}^{1}(\mathbf{O}_t) = B_{3i+2}^{2}(\mathbf{O}_t) = \cdots = B_{3i+2}^{m}(\mathbf{O}_t) = \cdots = B_{3i+2}^{M}(\mathbf{O}_t) = B_{3i+2}(\mathbf{O}_t) \qquad (6)$$

$$B_{3i+1}^{1}(\mathbf{O}_t) = N(\mathbf{O}_t, \mu_{3i+1,1}, \Sigma_{3i+1,1})$$
$$B_{3i+1}^{2}(\mathbf{O}_t) = N(\mathbf{O}_t, \mu_{3i+1,2}, \Sigma_{3i+1,2})$$
$$\cdots$$
$$B_{3i+1}^{m}(\mathbf{O}_t) = N(\mathbf{O}_t, \mu_{3i+1,m}, \Sigma_{3i+1,m})$$
$$\cdots$$
$$B_{3i+1}^{M}(\mathbf{O}_t) = N(\mathbf{O}_t, \mu_{3i+1,M}, \Sigma_{3i+1,M}) \qquad (7)$$

$$a_{kj}^{1} = a_{kj}^{2} = \cdots = a_{kj}^{m} = \cdots = a_{kj}^{M} = a_{kj} \qquad (8)$$

where M is the total number of parallel branches for each subunit and superscript m denotes the m-th branch. The m-th branch pdf of the steady-state part is set to the m-th pdf of mixture components. Once the subunit model has been split, new segmentation can be done in step 2 of the above algorithm. Also in step 3, the desired number of mixture components, K, can be estimated by the VQ procedure. And then the segmental k-means procedure is reiterated on the parallel-branch subunit model until convergence is obtained. As in the case of clustering, once the parallel-branch subunit model has been clustered, the training algorithm is used to give an optimal set of parameters for each of the parallel branches.

III. Experimental Evaluation

3.1 Task and Data Base

To study the recognition performance of the proposed algorithms and to compare these results with other algorithms, the speech material used in this work was a vocabulary with 1036 Korean words produced by 48 speakers(32 males and 14 females). 39 speakers(28 males and 11 females) were used for training data, and 9 others for test data. The total number of training words is 12227 and that of the test word is 2810. They were used for estimating HMM parameters of 32 recognition and testing of the proposed algorithm.

The speech data was first low-pass filtered with a cutoff frequency of 7.2 kHz, and then digitized at a sampling rate of 16 kHz using a 16 bit A/D converter. The digitized speech was then preemphasized with the digital filter, $1-0.95z^{-1}$. A 12th-order LPC analysis was performed on a Hamming-windowed speech segment of 20 ms, and a feature vector consisting of 13 cepstral coefficients($\boldsymbol{C}_t$) including log energy was generated every 10 ms. The 13 liftered cepstral coefficients ($\hat{\boldsymbol{C}}_t$) were computed as :

$$\hat{\boldsymbol{C}}_t(m) = C_t(m)W_c(m) \qquad 0 \le m \le 12$$

where $\hat{\boldsymbol{C}}_t(0)$ is a log energy and $W_c(m)$ is the window of the form $W_c(m) = 1 + \frac{Q}{2}\sin(\frac{\pi m}{Q})$, $Q = 12$. And then the corresponding delta cepstral vector ($\Delta\hat{\boldsymbol{C}}_t$) called linear regression coefficients can be obtained by

$$\Delta\hat{\boldsymbol{C}}_t(m) = [\frac{\sum_{k=-K}^{K} k\hat{c}_{t-k}(m)}{\sum_{k=-K}^{K} k^2}]G \qquad 0 \le m \le 12$$

where G is a gain term which makes the variances of the 13 liftered cepstral coefficients and the corresponding delta cepstral vector equal. In our current implementation, $K = 2$ and $G = 3.16$ were used. And the 13 liftered cepstral vector and the corresponding delta cepstral vector were concatenated, resulting in a 26-dimensional observation vector [2]. We assumed a diagonal matrix in our experiments.

3.2 Simulation Results and Discussion

The word recognition performances in the training and test set are shown in Tables 1, 2, and 3, where the values in the parenthesis denote the recognition accuracy including second candidate words.

As shown in Table 3, the recognition rates of test data based on mixture components are monotonically increasing depending on the number of parallel branches per subunit. In Table 2, it is seen that the word accuracy of the subunit model based on mixture components is significantly higher than that for the case of the splitting algorithm based on the maximum likelihood method by Rabiner et al[1]. Hence, one can conclude that the proposed algorithms are efficient and give good recognition performances in word recognition, and that the initialization in the parallel-branch subunit model is very important when the number of parallel branches per subunit increases.

The proposed parallel-branch subunit model is shown to give improvement in recognition accuracy in comparison with the single-branch subunit model. Also, the performance of the single branch continuous HMM is evaluated by various number of mixtures, 1, 2, 3, 6, 9, and 15. This is shown in Table 1. The recognition rate increases monotonically up to the number of mixtures being equal to 15. As shown in Table 3, the recognition rate is increasing, its amount depending on the

표 1. Word Recognition Performance(%) of Single-Branch Subunit Model(No. of Test Words was 2810)

No. of mixtures	Recognition rate
1	54.26(68.54)
2	63.18(76.19)
3	69.69(82.07)
4	71.23(83.15)
6	74.28(85.75)
9	73.94(87.06)
12	75.98(88.11)
15	74.25(86.97)

표 2. Word Recognition Performance(%) of a Parallel-Branch Subunit Model Based on Maximum Likelihood proposed by Rabiner et al.(No. of Test Words was 2810)

No. of	Recognition rate with different number of branches			
mixture	2	3	4	5
1	60.21(75.23)	63.95(78.22)	66.58(80.21)	63.38(76.66)

표 3. Word Recognition Performance(%) Of a Parallel-Branch subunit Model Based on the Mixture Components For The Test Set(No. of Test Words was 2810)

No. of	Recognition rate with different number of branches			
mixture	2	3	4	5
1	61.82(75.41)	65.59(78.68)	67.54(80.71)	68.65(81.64)
2	70.07(83.31)	73.70(85.45)	74.52(85.48)	74.41(85.84)
3	73.99(85.91)	76.55(87.47)	76.83(87.37)	74.88(86.62)
4	74.73(86.37)	77.05(88.61)	77.40(88.33)	77.87(88.33)
6	76.90(88.33)	-	-	-
9	77.94(88.68)	-	-	-

number of mixtures and the number of parallel branches. The highest recognition rate using the proposed method is 77.94%. This is obtained when a set of subunits with 2 parallel branch model and 9 mixtures is used. Nevertheless, since the amount data is limited, the number of branches per subunit and the number of mixture components must also be limited. The reason why the proposed method improves the recognition accuracy may be explained as follows. In the conventional continuous HMM, the single-branch subunit model involved is not capable of providing good representations of the variants of speech within a subunit. On the other hand, in the proposed training method it is based on HMM-based clustering according to variations of speech within the subunit. Hence, a proper combination of the number of mixture components and the number of branches for each subunit would be helpful in improving the recognition rate.

Ⅳ. Conclusions

We have shown that a parallel-branch subunit model and a proper number of mixture components yields significant performance improvement in speaker-independent word recognition. According to the simulation results for improved recognition, the use of a parallel-branch subunit model in continuous HMM is effective. A key issue in design and implementation of a parallel-branch subunit model is how to efficiently initialize the model that yields the best recognition performance. We have developed a splitting procedure which initializes each new parallel-branch subunit model by splitting off all subunits in the training sets.

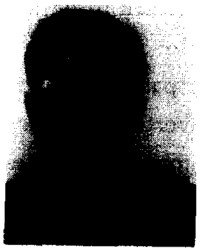
While the training algorithm guarantess improvement in every iteration, it does not guarantee finding a global maximum. If the initialization is poor, one may find a local maximum that results in poor recognition rate. The proposed model has many degrees of freedom(i.e., many continuous parameters) with respect to the amount of training data. Thus, a more sophisticated form of initialization is needed. This model will increase the number of parameters which will be estimated from observations and require a large amount of training utterances. A proper combination of the number of mixture components and the number of model per subunit would increase the recognition rate significantly.

참 고 문 헌

1. L. R. Rabiner, C. H. Lee, B. H. Juang, and J. G. Wilpon, "HMM Clustering for Connected Word Recognition," *IEEE Int. Conf. on Acoustics, Speech, And Signal Processing*, pp. 405-408, 1989.
2. C. H. Lee, L. R. Rabiner, R. Pieraccini, and J. G. Wilpon, "Acoustic modeling for large vocabulary continuous speech recognition," *Computer Speech and Language*, vol. 4, pp. 127-165, 1990.
3. L. Rabiner and B. H. Junag, Fundamentals of Speech Recognition, Prentice Hall, New York, 1993.
4. R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons, New York, 1973.
5. K. Lee, Automatic Speech Recognition : The Development of the SPHINX system. Boston, MA : Kluwer Arcademic, 1989.
6. S. Furui and M. Sondhi, Advances in Speech Signal Processing, Dekker, pp. 509, 1992.
7. L. R. Rabiner, J. G. Wilpon, and B. H. Juang, "A Segmental K-Means Training Procedure for Connected Word Recognition," *AT&T Tech. Journal.*, vol. 65, No. 3, pp. 21-32, May-June. 1986.
8. L. E. Baum, "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes," Inequalities, vol. 3, pp. 1-8, 1972.
9. L. A. Liporace, "Maximum likelihood estimation for multivariate observations of Markov sources," *IEEE Trans. Information Theory*, vol. IT-28, PP. 729-734, 1982.
10. B. H. Juang and L. R. Rabiner, "The segmental k-means algorithm for estimating parameters of hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-38, pp. 1639-1641, Sep. 1990.

11. A. Viterbi, "Error bounds for conventional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform Theory*, vol. IT-13, pp. 260-269, Apr. 1967.
12. B. H. Juang and L. R. Rabiner, "Mixture autoregressive hidden Markov Models for speech signals," *IEEE Trans. Acoust. Speech. Signal Processing*, vol. 33, pp. 1404 1413, 1985.
13. L. R. Rabiner, K. C. Pan and F. K. Soong, "On the performance of isolated word speech recognition using vector quantization and temporal energy contours," *AT&T Bell Lab. Tech. J.*, vol. 63, pp. 1245 1260, Sep. 1984.
14. S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, "An introduction to the application of the theory of probabilistic foundations of a Markov process to automatic speech recognition," *AT&T Bell Lab. Tech. J.*, vol. 62, pp. 1035-1074, Apr. 1983.

▲박용규(Yong Kyoo Park) 1960년 12월 30일생



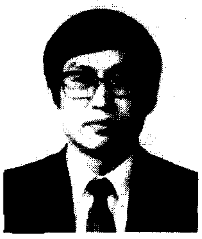
1980년~1984년 : 한양대학교 전기공학과(B.S.)

1985년~1987년 : 한국과학기술원 전기 및 전자공학과(M.S.)

1987년~1991년 : 한국통신 연구개발원

1991년~현재 : 한국과학기술원 전기 및 전자공학과(Ph.D.)

▲은종관 1940년 8월 25일생



1964년 6월 : Univ. of Delaware 공학사

1966년 6월 : 동대학원 공학석사

1969년 6월 : 동대학원 공학박사

1964년~1969년 : Univ. of Delaware Research Fellow

1969년~1973년 : Univ. of Maine 전자공학 조교수

1973년~1977년 : Stanford 연구소 책임연구원

1982년 1월~1983년 6월 : 한국과학기술원 공학장

1983년 7월~1989년 6월 : 한국과학기술원 통신공학연구실장

1977년 7월~현재 : 한국과학기술원 전기 및 전자공학과 교수

1987년 11월~1989년 11월 : 한국음향학회 회장