Oscillometric Blood Pressure Estimation Based on Deep Learning

Soojeong Lee and Joon-Hyuk Chang, *Senior Member, IEEE*

***Abstract*—Oscillometric measurement is widely used to estimate systolic blood pressure (SBP) and diastolic blood pressure (DBP). In this paper, we propose a deep belief network (DBN)-deep neural network (DNN) to learn about the complex nonlinear relationship between the artificial feature vectors obtained from the oscillometric wave and the reference nurse blood pressures using the DBN-DNN-based-regression model. Our DBN-DNN is a powerful generative network for feature extraction and can address to stick in local minima through a special pretraining phase. Therefore, this model provides an alternative way for replacing a popular shallow model. Based on this, we apply the DBN-DNN-based regression model to estimate the SBP and DBP. However, there are a small amount of data samples, which is not enough to train the DBN-DNN without the overfitting problem. For this reason, we use the parametric bootstrap-based artificial features, which are used as training samples to efficiently learn the complex nonlinear functions between the feature vectors obtained and the reference nurse blood pressures. As far as we know, this is one of the first studies using the DBN-DNN-based regression model for BP estimation when a small training sample is available. Our DBN-DNN-based regression model provides a lower standard deviation of error, mean error, and mean absolute error for the SBP and DBP as compared with the conventional methods.**

***Index Terms*—Blood pressure (BP), bootstrap, deep neural networks (DNNs), machine learning, oscillometric blood pressure estimation.**

I. Introduction

BLOOD PRESSURE (BP) is one of the most important vital signals and plays a key role in determining the cardiovascular activity of patients. Oscillometric BP devices have recently gained popularity [1]–[4] and monitoring using this device is now readily available for the home, office, and hospital. The MAA is commonly utilized to estimate the mean of the BP as the cuff pressure at which the maximum oscillation occurs and then linearly relates the SBP and DBP to the mean pressure using heuristically derived ratios [1]. These characteristic ratios are used to determine time points where the cuff pressure corresponds to the SBP on the ascending phase of the OMW's envelope and DBP on the descending phase of the OMW's envelope [1]. Although it is possible to estimate the arterial BP using the oscillometric measurement technique, an established standard is not available to determine the SBP and DBP [1]. Even if the SBP and DBP ratios in the conventional MAA method are assumed to be fixed, this assumption is not supported by the evidence because the BPs (SBP and DBP) vary consistently over time according to various physiological factors [2]. Generally, the characteristic fixed ratio may be viewed as a parameter dependent on the oscillometric BP measurements derived from a specified group of subjects by minimizing the MAE relative to the auscultatory measurements recorded by the trained nurses [3], [4]. Liu *et al.* [5] studied an error mechanism of the fixed ratio using the oscillometric BP measurement. Raamat *et al.* also found that physiological components have significant effects on the characteristic ratios [6]. More recently, a pilot study using the coefficient-free BP estimation based on the pulse transit time-cuff pressure dependence was proposed to address the fixed-ratio problem of the conventional MAA [7].

Alternative methods such as artificial neural networks [8], [9] have been considered to overcome the characteristic fixed-ratio problem of the conventional MAA. These methods are not based on a mathematically complicated model and are suitable only for nonlinear physiological structures [10]. Based on these methods, Baker [11] proposed a two-layer FFNN based on the backpropagation training algorithm in order to estimate the BP. In a similar approach, Narus *et al.* [9] used one-hidden-layer FFNN using the backpropagation training to estimate the BP at the supraorbital artery. However, in the literature, the raw-signal-based method using the OMW has limits such as the complexity of the input data, the amount of time needed, and the computational expense for larger training measurement sets [10]. Also, a new technique using the feature-based FFNN, which consists of two-layer FFNNs with a linear output layer, was performed to compute the gradient using the backpropagation algorithm [10]. The activation function of hidden layers was the sigmoid-type function that allows the network to learn the complex nonlinear relationships between the inputs and targets [10]. However, the backpropagation algorithm often suffers from the local minima problem, which does not achieve consistently acceptable performance to satisfy the recommendations of the American National Standards Institute and the Association for the Advancement of Medical Instrumentation SP10 standard [12]. For these reasons, an improved technique is required to estimate the BP accurately without the characteristic fixed ratio.

Manuscript received February 4, 2016; revised May 21, 2016, June 28, 2016, July 20, 2016, and August 4, 2016; accepted September 14, 2016. Date of publication September 26, 2016; date of current version April 18, 2017. This work was supported by the National Research Foundation of Korea under Grant 2014R1A2A1A10049735. Paper no. TII-16-0130. *(Corresponding author: J.-H. Chang.)*

The authors are with the Department of Electronic Engineering, Hanyang University, Seoul 133-791, South Korea (e-mail: leesoo86@hanyang.ac.kr; jchang@hanyang.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2016.2612640

1551-3203 © 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

In this paper, we propose a novel technique using a DBN-DNN-based regression model [12], [13] to learn the complex nonlinear relationship between the feature vectors obtained from the OMW and target BPs; we note herewith that the DBN proposed by Hinton *et al.* [14] is a probabilistic generative model that contains many layers of hidden variables [15] and has recently attracted attention in both the machine learning and signal processing areas as a superior technique [16]. Based on this motivation, the DBN-DNN-based regression model is proposed to estimate the SBP and DBP, for which the greedy layerwise pretraining algorithm using the stack of RBMs is first performed to initialize the DBN parameters in the high-dimensional input data and then the fine-tuning is applied to the DNN as a discriminative training. However, our feature space has a small sample size due to a limited subject population, which yields a fatal weakness when using the DBN-DNN [14], since the DBN-DNN works well with a big data [17]. To address this problem, the parametric bootstrap technique is originally employed to create artificial features for the SBP and DBP estimation. As a result, the number of the artificial feature samples is dramatically increased by using the parametric bootstrap technique [18], and thus, the DBN-DNN can be efficiently learned to mimic the complex nonlinear relations between the feature vectors obtained from the OMW and the reference nurse BPs. As far as we know, this is one of the first studies employing the DBN-DNN-based regression model for BP estimation with a small training sample. The key idea behind our proposed technique is that the SBP and DBP are directly estimated using the DBN-DNN with artificial features. The paper has the following additional enhancements and contributions when compared to [10].

1) It provides a novel technique to obtain accurate BP estimates from a small sample of oscillometric BP measurements using the DBN-DNN-based regression model.
2) The DBN-DNN is designed to address a local minima to the training feature set through a special pretraining phase, which is a greedy layerwise unsupervised pretraining stage of the stacked RBMs to obtain initial parameters. It is also used as a regularizer of the supervised learning stage that hinders the overfitting of the training set in the DBN-DNN [16]. Therefore, it is very worthwhile to consider replacing the popular shallow models such as FFNN [10] and SVR [19] with an excellent generative model so that it can represent largely variant functions and find the fundamental regularity of multiple features quite well.
3) We devise a technique to address the problem caused by small samples of features using the parametric bootstrap because small samples cause a serious disadvantage as overfitting on using the machine learning technique [20]. We can thus mitigate the overfitting problem using artificial feature samples that are generated based on the parametric bootstrap technique.
4) We confirm the distribution of artificial features is asymptotical Gaussian that fits very well to the DBN-DNN model as the number of training data increases.
5) It provides the evidence for superiority of the proposed method by utilizing the grading criteria used by the BHS[21] and the ANSI/AAMI [12].

The structure of this paper is organized as follows. In Section II, a short review of conventional FFNN is represented. We first propose the DBN-DNN-based regression model with the artificial features in Section III. Section IV describes the experimental dataset and measurement methodology and provides results, while Section V discusses the results and provides conclusions.

II. Review of Conventional FFNN

The FFNN is probably the most widely used structure that can be learned by set of input features and output targets. This architecture is composed of a network of processing units called neurons, which compute a weighted sum of inputs. The trained information is retained in the weights and biases, and neurons are connected to neurons of the next layers through the weights in each layer. The weighted sum of all inputs is fed into each hidden unit, in which an activation function is used to find the weighted sum to produce the hidden units' output. All of the hidden units are activated using the logistic activation function. To minimize the mean squared error (MSE) between the predicted and target values [20], the scaled conjugate gradients technique [22] is used as a backpropagation algorithm. This algorithm was originally proposed as a popular technique to train multilayer architectures. However, the FFNN with the multilayer has been replaced by other techniques such as support vector machines, and boosting because their training turned out difficult and backpropagation algorithms got frequently stuck in local minima when starting from random initialization. Even though improvements can be acquired by trying different practical techniques such as multilayer training using random initialization; nevertheless, their generalization ability may not be competitive with other techniques. This weakness was quite serious when more than two hidden layers were utilized. The more hidden layers one uses, the more difficult the training leads to. Also, the probability that suffers from local minima is increased [17], [23]. Based on these, there are two issues that need to be considered for the FFNN technique.

1) Compared to the training methods of traditional FFNN, the DBN with deeper layers can be used to prevent overfitting to the training set via an unsupervised pretraining step.
2) In order to train the deep structure of the neural network, advanced techniques to optimize the deeper neural network that have many layers of hidden unit are required against the small size of the input data.

The following section will be presented about the issues. The detailed parameter setting for the FFNN is also summarized in Section IV.

III. DBN-DNN for SBP and DBP Estimation

One of the key advantages of the DBN-DNN technique is its powerful generalizability over the conventional FFNN [16]. In particular, a greedy layerwise unsupervised learning technique initializes a very good set of parameters because the unsupervised training makes it easier to optimize many parameters before overfitting occurs, which is called pretraining [14]. After pretraining, a supervised learning step is performed through the

backpropagation algorithm to fine-tune the parameters acquired from the pretraining [14], [16]. The pretraining step makes it possible to discover abstractions from the lowest level features to highest level concepts. In other words, the input data are to be mapped into gradually higher levels of representation through the deep architectures [15]. Based on these advantages, we devise a novel way to use the DBN-DNN for estimating the SBP and DBP. But, unfortunately, the sample number of input feature vectors in this work is limited for each subject because it is not practically feasible to acquire a large number of oscillometric BP measurements for each subject or guarantee repeatable conditions to reproduce measurements [1], which is a serious disadvantage in the learning step of the DBN-DNN technique. Actually, the backpropagation fine-tuning in the DNN yields an unstable performance when small samples are used. Specifically, the problem of small sample in the training step is known as a factor to raise the overfitting problem [20]. Although the dropout and early stopping is known to alleviate the overfitting problem as practical techniques, however, we cannot use the dropout and early stopping because these techniques basically require a suitable amount of the input data. Thus, the artificial features from the parametric bootstrap technique [18] are presented as input features of the DBN-DNN, as shown in the following subsection. Moreover, the distribution of the artificial input features will assist to improve the performance of the DBN-DNN-based regression model because the DBN in the pretraining step consists of statistical RBM structures [14].

The proposed DBN-DNN-based regression model is composed of four stages: feature extraction, artificial feature generation, the DBN pretraining and DNN fine-tuning, and DBN-DNN estimation. The more detailed description follows. Also, see Table I which includes the list of acronyms.

TABLE I
LIST OF ACRONYMS

acronym	meaning
BP	blood pressure
MAA	maximum amplitude algorithm
SBP	systolic blood pressure
DBP	diastolic blood pressure
OMW	oscillometric waveform
FFNN	feed-forward neural network
ANSI	American National Standards Institute
AAMI	Association for Advancement of Medical Instrumental
SP10	sphygmomanometer committee
DBN	deep belief networks
DNN	deep neural networks
RBM	restricted Boltzmann machine
SVR	support vector regression
BHS	British Hypertension Society
MMSE	minimum mean-squared error
RSBP	reference systolic blood pressure
RDBP	reference diastolic blood pressure
MAP	mean arterial pressure
MA	maximum amplitude
AE	area of the OMW's envelope
AR	asymmetry ratio
MAPL	a length position of the maximum amplitude
EL	envelope's length
CDF	cumulative distribution function
MAE	mean absolute error
SDE	standard deviation of the error
ME	mean error
mmHg	millimeters of mercury
MLR	multiple linear regression

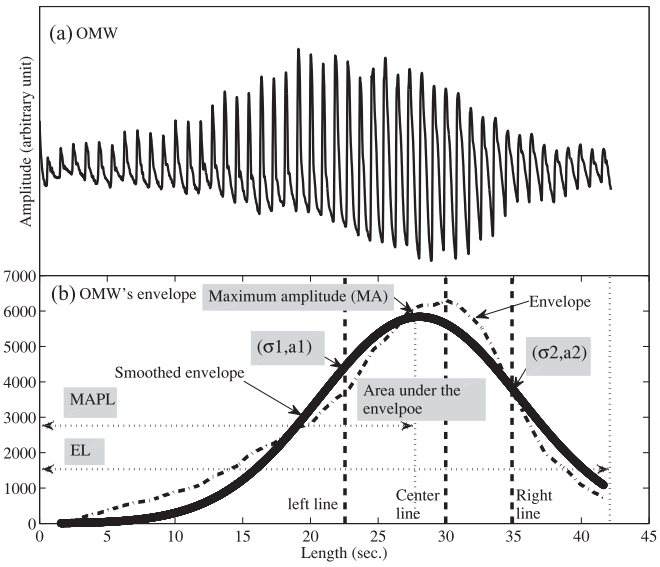


Fig. 1. Basic concept of (a) an example of the OMW and (b) OMW's envelope, where the dashed-dotted line represents an envelope obtained from the OMW and the bold line represents a smoothed envelope using asymmetrical Gaussian fitting [1].

A. Feature Extraction

In order to handle linear and nonlinear transformation techniques, a set of features are generated from a set of original features on the conventional machine learning model such as the FFNN and SVR [10], [19]. To be more specific, the basic reasoning behind generated new features is that an appropriately selected transform can exploit and remove redundant information [24]. However, the proposed DBN-DNN-based regression model does not require feature generation techniques such as principal component analysis because it can readily mix the merit of our feature vectors to estimate the target BPs efficiently [16]. In our work, we basically focus on the feature extraction [25] obtained from the OMW and OMW's envelope [1], [3] because the preprocessing of the OMW is not the main interest. For more details on the deflation curve, OMW extraction, and envelope detection, refer to [10]. We first carefully analyze the original features obtained from the OMW's envelope in order to estimate the nurse's RSBP and the nurse's RDBP. The features, which represent the characterization of the BPs of individual subjects, could naturally include the MAP estimated using the MAA technique, which is mapped to the cuff pressure [3]. The MA, the AE [3], and the AR of the OMW's envelope (= MAPL/EL) [3] calculated by MAPL and EL [3] are included as shown by the gray marks in Fig. 1. We also obtain the feature vectors from the OMW's envelope using the asymmetrical Gaussian curve fitting given as follows [1], [10]:

$$f(x) = a_1 \cdot \exp\left\{\frac{-(x-\mu)^2}{2\cdot\sigma_1^2}\right\} + a_2 \cdot \exp\left\{\frac{-(x-\mu)^2}{2\cdot\sigma_2^2}\right\} \quad (1)$$

where a_1 and a_2 represent the amplitudes, σ_1 and σ_2 denote the different standard deviation of the asymmetrical Gaussian function, and μ is the same mean location [1] based on the left, center, and right lines, as shown in Fig. 1. Using this function, two features are acquired from different standard deviations (σ_1 and σ_2). Therefore, eight feature vectors were prepared to estimate the target BPs (RSBP and RDBP). A general normalization method is then used to restrict the interval of the allowed feature values to lie in between the minimum and maximum of the predefined intervals [3]. This method is to help the training speed of the networks and prevent the saturation of the networks as described in [10]. Particularly, the feature ζ can be mapped into a normalized feature η as given by

$$\eta = \left\{ \frac{\zeta - \zeta_{\min}}{\zeta_{\max} - \zeta_{\min}} \right\} (\eta_{\max} - \eta_{\min}) + \eta_{\min} \tag{2}$$

where $\zeta_{\min}$ is the minimum measured feature, $\zeta_{\max}$ is the maximum measured feature, $\eta_{\min} = 0.1$ is the minimum scaled value, and $\eta_{\max} = 0.99$ is the maximum scaled value.

B. Artificial Features Using Bootstrap

As mentioned before, we use the artificial feature generated by the bootstrap technique [1], [18], which is a computer-intensive method to improve the accuracy of the estimates from a small number of measurements in situations, where conventional methods are not valid to improve the accuracy [1], [18]. In this work, suppose $X = \{x_1, \cdots, x_N\}$ is a random sample of the distribution F with unknown parameters $\{\theta, \sigma\}$. Thus, F can be approximated by $\hat{F}(\hat{\theta}, \hat{\sigma}|X)$, where the mean and standard deviation are given by

$$\mathbb{E}(\theta|X) \simeq \hat{\theta} = \bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_i \tag{3}$$

$$\mathbb{E}(\sigma|X) \simeq \hat{\sigma} = \sqrt{\left[\frac{1}{N-1} \sum\nolimits_{n=1}^{N} (x_n - \bar{x})^2 \right]} \tag{4}$$

where $\hat{F} \simeq \mathcal{N}(\hat{\theta}, \hat{\sigma})$ is approximated by a normal distribution, which is called the parametric bootstrap [18]. The parametric bootstrap is used to generate the artificial features obtained from the original feature vectors that have *a priori* information such as the mean and standard deviation for each subject, whereas the nonparametric bootstrap technique is used to estimate F without any *a priori* assumptions.

Instead of sampling with replacement from the $X = \{x_1, \cdots, x_N\}$, we generated B samples X_b^* with size $N = 5$ and $\forall b \in \{1, \cdots, B\}$ from $\hat{F}_{\hat{\theta},\hat{\sigma}}$, as shown in Table II, where N denotes the number of measurement for each subject. In the parametric technique, the bootstrap samples based on $\hat{F}_{\hat{\theta},\hat{\sigma}}$ were calculated using the Monte Carlo method [1]. Thus, the sufficient feature samples are used to efficiently learn the complex nonlinear relationship between the artificial feature vectors obtained from the OMW and the target BPs as the train data, whereas the unseen feature vectors are used as the test data. As shown in step 5 in Table II , we computed the average of all measurements in $\mathbf{X}^*$.

TABLE II
ALGORITHM IS AN EXAMPLE OF ARTIFICIAL FEATURE CREATION USING THE PARAMETRIC BOOTSTRAP TECHNIQUE FOR EACH SUBJECT, WHERE I DENOTES THE NUMBER OF INPUT FEATURE (MAP, AR,...), J IS THE NUMBER OF TARGET BPS (SBP AND DBP), AND B REPRESENTS THE NUMBER OF PARAMETRIC BOOTSTRAP REPLICATION [1]

Algorithm	(The DBN-DNN with parametric bootstrap technique)
step 1:	input feature vectors: $X_i = \{x_1, x_2, \cdots, x_N\}$ and $Y_j = \{y_1, y_2, \cdots, y_N\}, \forall i = 1$ to $I(=8)$ and $\forall j = 1$ to $J(=2)$, explanatory matrix: $\mathbf{X} = \{X_1, X_2, \cdots, X_I\}$ and response (targets) matrix: $\mathbf{Y} = \{Y_1, Y_2\}$, $\mathbf{S} = \{\mathbf{X}, \mathbf{Y}\}$
step 2:	calculate parameters: $\mathbb{E}(\mu, \sigma|X_i)$ and $\mathbb{E}(\mu, \sigma|Y_j)$,
step 3:	call parametric bootstrap replication: $X_{i,b}^* = \{x_1^*, x_2^*, \cdots, x_N^*\}$ and $Y_{j,b}^* = \{y_1^*, y_2^*, \cdots, y_N^*\}, \forall b = 1$ to B,
step 4:	calculate average: $\bar{X}_{i,b}^* = \frac{1}{N}\sum_{n=1}^{N} x_n^*$ and $\bar{Y}_{j,b}^* = \frac{1}{N}\sum_{n=1}^{N} y_n^*$
step 5:	generate artificial feature: $\mathbf{X}_i^* = \{\bar{X}_{i,1}^*, \bar{X}_{i,2}^*, \cdots, \bar{X}_{i,B}^*\}$ and $\mathbf{Y}_j^* = \{\bar{Y}_{j,1}^*, \bar{Y}_{j,2}^*, \cdots, \bar{Y}_{j,B}^*\}$
step 6:	call pretraining phase: $\mathbf{RBM}^1\ \{\hat{f}^*(\mathbf{X}_i^*)\}, \mathbf{RBM}^2\ \{\hat{f}^*(\mathbf{X}_i^*)\}, \mathbf{RBM}^3\{\hat{f}^*(\mathbf{X}_i^*)\}$ where b denotes the index of parametric bootstrap replication.
step 8:	output: parameters (weights and biases) for fine tuning
step 10:	call backpropagation phase:
step 11:	call parametric bootstrap replication: $X_{i,b}^* = \{x_1^*, x_2^*, \cdots, x_N^*\}$ and $Y_{j,b}^* = \{y_1^*, y_2^*, \cdots, y_N^*\}, \forall b = 1$ to B,
step 12:	calculate average: $\bar{X}_{i,b}^* = \frac{1}{N}\sum_{n=1}^{N} x_n^*$ and $\bar{Y}_{j,b}^* = \frac{1}{N}\sum_{n=1}^{N} y_n^*$
step 13:	generate artificial feature: $\mathbf{X}_i^* = \{\bar{X}_{i,1}^*, \bar{X}_{i,2}^*, \cdots, \bar{X}_{i,B}^*\}$ and $\mathbf{Y}_j^* = \{\bar{Y}_{j,1}^*, \bar{Y}_{j,2}^*, \cdots, \bar{Y}_{j,B}^*\}, \forall i = 1$ to $I(=8)$ and $\forall j = 1$ to $J(=2)$
step 14:	call learning: $\mathbf{backpropagation}\ \{\hat{f}^*(\mathbf{X}_m^*, \mathbf{Y}_m^*)\}$, output: $\hat{\mathbf{Y}}_m$, $\forall m = 1$ to M where b denotes the index of parametric bootstrap replication.

1) Kolmogorov–Smirnov Test and Its Analysis: From the above artificial feature distribution, we conduct the statistical analysis through the Kolmogorov–Smirnov test [26] to validate the normality of each artificial feature distribution as shown in Fig. 2, where these artificial feature distributions fit the Gaussian distributions very well. Assume that we obtain $\hat{F}$, a distribution of artificial feature $\{\hat{\theta}_1^*, \hat{\theta}_2^*, \cdots, \hat{\theta}_B^*\}$ from unknown sample distribution F. Thus, it is needed to test the hypothesis that $\hat{F}$ is equal to the Gaussian distribution F_0 as given by

$$\mathbb{H}_0 : \hat{F} = F_0, \quad \mathbb{H}_1 : \hat{F} \neq F_0. \tag{5}$$

As shown in Table III, we find that the hypothesis test results of all artificial features are 0. These indicate a failure to reject the null hypothesis at the 0.05 significance level. We can also reject the null hypothesis if the critical value is less than the statistic value of the Kolmogorov–Smirnov test [26]. Also, all p-values are larger than the level of significant (0.05). The other subjects also show similar results as in Table III. Therefore, we can accept the normality of the distributions of the artificial features.

2) Asymptotic Accuracy of Artificial Feature: Next, we need to evaluate the consistency and convergence of the artificial feature acquired by the bootstrap technique [1], [18], and our artificial features should have been a good approximation of the original features based on the bootstrap convergence for the sample mean [18] based on the theorem by Singh [27] as follows:

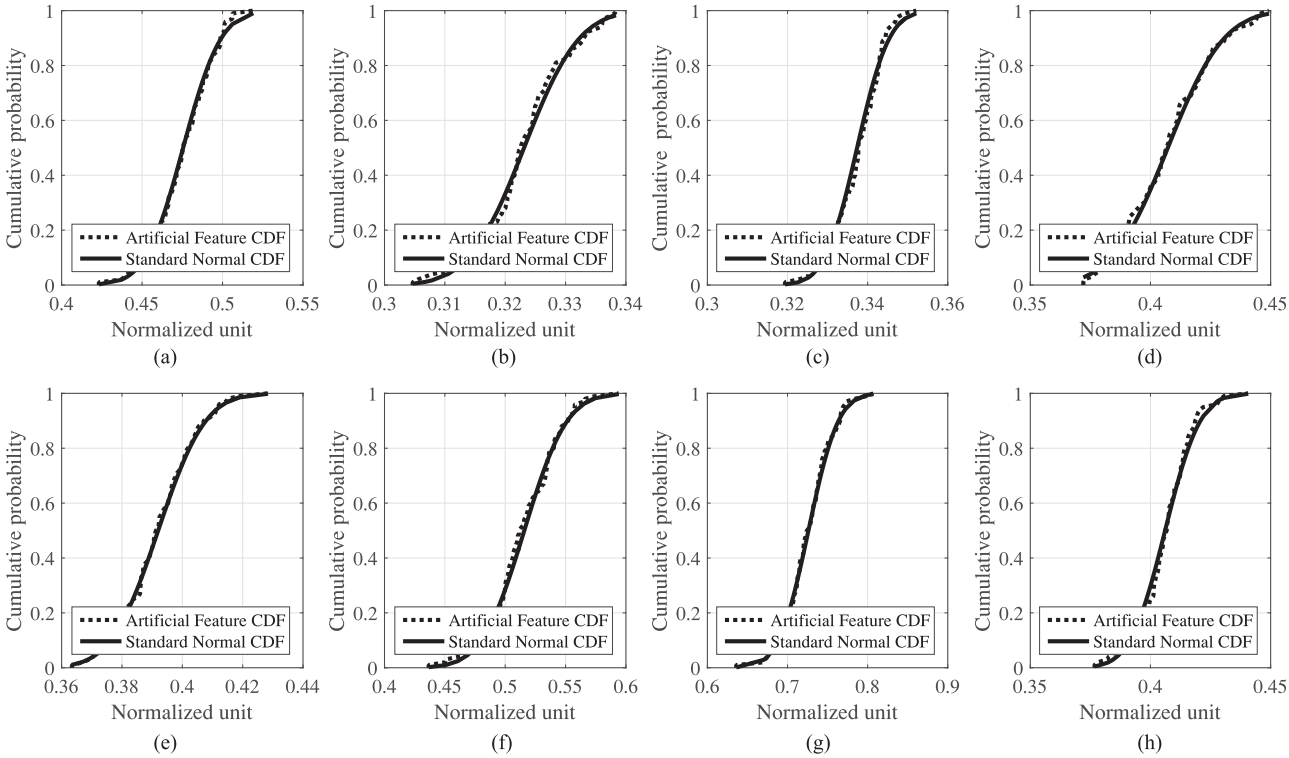


Fig. 2. CDF of artificial features based on the parametric bootstrap approach with replication numbers ($B = 100$). Note that these artificial figures are examples obtained from one subject with five samples for each feature [3].

TABLE III
HYPOTHESIS TEST VALUES OF ALL ARTIFICIAL FEATURES BASED ON THE PARAMETRIC BOOTSTRAP APPROACH WITH REPLICATION NUMBERS ($B = 100$), WHERE h DENOTES THE HYPOTHESIS TEST RESULT, *ks* DENOTES THE STATISTIC VALUE OF THE KOLMOGOROV–SMIRNOV TEST, *p* DENOTES THE *p*-VALUE, AND *cv* IS THE CRITICAL VALUE [26]

Feature/Test values	h	ks	p	cv
MAP	0	0.044	0.985	0.134
AR	0	0.064	0.787	0.134
AE	0	0.080	0.519	0.134
EL	0	0.054	0.920	0.134
MA	0	0.052	0.934	0.134
σ_1	0	0.061	0.832	0.134
σ_2	0	0.040	0.995	0.134
MAPL	0	0.084	0.464	0.134

Theorem: If $\mathbb{E}(x^2) < \infty$, then

$$\sup_{x\in\mathbb{R}} \mid \mathbb{P}^*\{\sqrt{n}(\hat{\theta}^* - \hat{\theta}) \le x\} - \mathbb{P}\{\sqrt{n}(\hat{\theta} - \theta) \le x\} \mid \longrightarrow 0 \tag{6}$$

where $\mathbb{P}^*$ denotes the conditional probability using the parametric bootstrap based on the original feature X and sup as the sup-norm. On the basis of this theorem, we can prove that the distribution of $\sqrt{n}(\hat{\theta}^* - \hat{\theta})$ converges to that of $\sqrt{n}(\hat{\theta} - \theta)$, which is almost surely zero [27]

$$\text{bias}(\hat{\theta}(\cdot)) = \mathbb{E}\left[\hat{\theta}(X) - \theta\right] \tag{7}$$

where θ is the true parameter. If the bias is zero, the estimator is called unbiased. Thus, we compute the prediction error (bias)

TABLE IV
EXEMPLARY RESULT (ONE SUBJECT) SHOWING AN EVALUATION BETWEEN THE ARTIFICIAL FEATURE AND ORIGINAL FEATURE FOR CONSISTENCY AND CONVERGENCE [18]

Features	$\hat{\theta}$	$\hat{\theta}^*$	bias$(\hat{\theta}^*(\cdot))$	$\hat{s_e}$	$\hat{s_e}^*$
RSBP	137.0	136.95	−0.05	4.90	2.06
RDBP	72.2	71.64	−0.56	5.85	2.83
MAP	0.4733	0.4729	−0.004	0.036	0.016
AE	0.322	0.3229	0.007	0.015	0.0071

and standard error of the interested parameter as follows:

$$\begin{aligned}\text{bias}(\hat{\theta}^*(\cdot)) &= \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}_b^* - \mathbb{E}(\theta|X) \\ &= \mathbb{E}(\theta|X^*) - \mathbb{E}(\theta|X) \\ &= \hat{\theta}^* - \mathbb{E}(\theta|X) \cong \mathbb{E}\left[\hat{\theta}^*(X^*) - \hat{\theta}(X)\right]\end{aligned} \tag{8}$$

where bias$(\hat{\theta}^*(\cdot))$ denotes the prediction error (i.e., bias)

$$\hat{s}_e^* = \sqrt{\left[\frac{1}{B-1}\sum\nolimits_{b=1}^{B}(\hat{\theta}^* - \hat{\theta}^*(\cdot))^2\right]} \tag{9}$$

where $\hat{s}_e^*$ is the standard error based on the parametric bootstrap and $\hat{\theta}^*(\cdot)$ denotes $B^{-1}\sum_{b=1}^{B}\hat{\theta}_b^*$. From the results in Table IV, we confirm that the bias is very small in the exemplary sample and that the standard error $\hat{s_e}^*$ obtained from the parametric bootstrap more closely approximates $\hat{\theta}$ than $\hat{s_e} = \hat{\sigma}$, so the

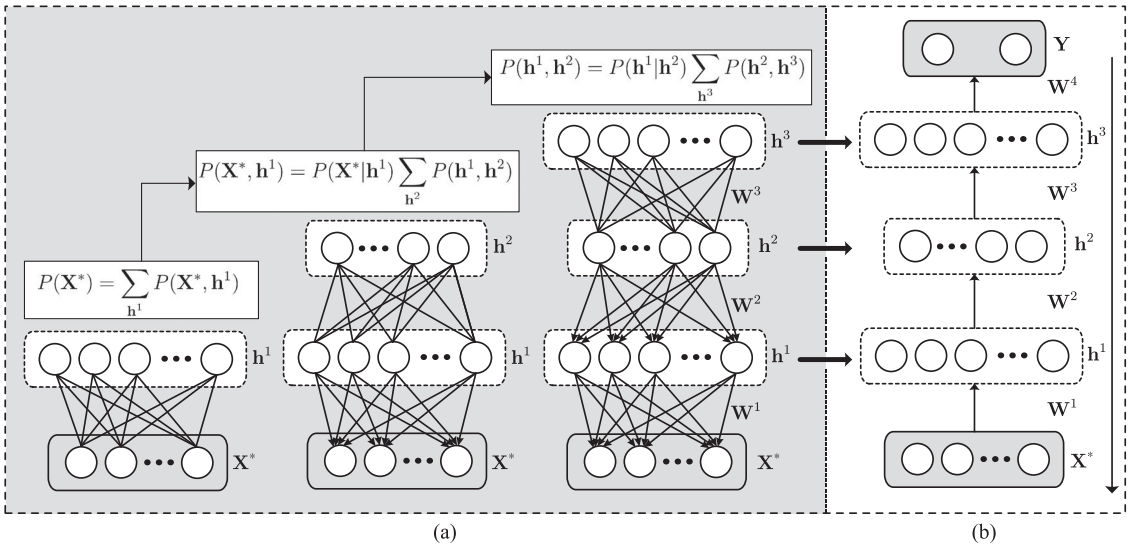


Fig. 3. Concept of DBN-DNN [14]. (a) Arrows denote the direction of information flow through the RBMs during top-down generation. (b) Arrows are the direction of information flow through the networks during bottom-up propagation, where $\mathbf{W}$ denotes the weighted matrix and long arrow in (b) is the direction of error flow.

parameter bootstrap can be used as a good generator for increasing the number of samples for the RSBP, RDBP, MAP, and AE as shown in Table IV. The artificial features have very similar statistical information as the original feature vectors. The remaining features also show similar results as in Table IV. In addition, the distribution of the artificial features obtained using the parametric bootstrap is shown in Fig. 2 ($B = 100$). As the number of replication B gets large, the distribution becomes more closely the Gaussian distribution [1]. Based on the statistical test and asymptotic accuracy, it is confirmed that the distributions of the artificial features are asymptotical Gaussian.

C. Design of DBN-DNNs

The training of the DBN-DNN regression model is composed of a greedy layerwise unsupervised pretraining and supervised backpropagation fine-tuning [13], [14] as shown in Fig. 3, where the DBN is a top-down graph computed sequentially from the top hidden layer to the input data. It should be noted that the FFNN-based DNN is a bottom-up architecture calculated sequentially from the input data to the top layer through a nonprobability approach. Indeed, the FFNN-based DNN architecture used to fail in finding a good set of model parameters because it may be locked in local minima under the many layers of nonlinearities and millions of parameters. Unlike the FFNN-based DNN, the DBN is a probabilistic generative architecture [16] that consists of multihidden layers of stochastic latent variables. Our top two layers are undirected connections, whereas other hidden layers have a top-down directed acyclic graph as shown in Fig. 3(a), where the units of the lowest layer are visible units, which denote an artificial input feature vector. As shown in Fig. 3(a), successively connected two layers are called the RBM, which is a fundamental component of DBN with multiple layers of hidden units. The DBN model can thus be defined as $P(\mathbf{X}^*,\mathbf{h}^1,\mathbf{h}^2,\cdots,\mathbf{h}^l)=P(\mathbf{X}^*|\mathbf{h}^1)P(\mathbf{h}^1|\mathbf{h}^2)\cdots P(\mathbf{h}^{l-2}|\mathbf{h}^{l-1})P(\mathbf{h}^{l-1},\mathbf{h}^l)$ and the condition layers $P(\mathbf{h}^i|\mathbf{h}^{i+1})$ are factorized conditional distribution [15], where $\mathbf{h}^i$ denotes the hidden units at layer i and $\mathbf{X}^*$ is the artificial input feature vector. The hidden layer $\mathbf{h}^i$ is a binary random vector with $\mathbf{h}^i_j$. Thus, a joint probability with input layer activations $\mathbf{X}^*$ (for the visible unit) and hidden layer activations $\mathbf{h}$ is given by

$$P(\mathbf{X}^*,\mathbf{h})=\frac{1}{Z}\exp^{-\mathbf{h}^{\mathrm{T}}\mathbf{W}\mathbf{X}^*-\mathbf{b}^{\mathrm{T}}\mathbf{X}^*-\mathbf{c}^{\mathrm{T}}\mathbf{h}} \tag{10}$$

where Z denotes the normalization constant for the joint probability, $\mathbf{b}$ represents the bias vector for the visible units, $\mathbf{c}$ denotes the bias vector for the hidden units, and $\mathbf{W}$ represents the weight matrix for the layer, respectively. The conditional probability of one layer given the other is thus defined as

$$\begin{aligned}P(\mathbf{h}^i|\mathbf{h}^{i+1})&=\prod_{j=1}^{n^i}P(\mathbf{h}^i_j|\mathbf{h}^{i+1})\\&=\mathrm{sigm}\left(-\mathbf{b}-\sum_{k=1}^{n^{i+1}}\mathbf{W}\mathbf{h}^{i+1}_k\right)\end{aligned} \tag{11}$$

where $\mathrm{sigm}(x)$ denotes $1/(1+\exp(-x))$, $\mathbf{b}=[b^i_j]$ represents the bias vector for unit j of layer i, and $\mathbf{W}=[W^i_{kj}]$ represents the weight matrix for layer i. Specifically, the RBM consists of a two-layer, bipartite, generative undirected model with a set of binary hidden units $\mathbf{h}$, and a set of visible units $\mathbf{X}^*$ in our case. Also, a weight matrix $\mathbf{W}$ connects the visible units and the hidden units [16]. The logistic function is also used to activate in the hidden units and the type of the output is used as a linear value.

Practically, the Gaussian–Bernoulli RBM [14] in this work is adopted for connecting between a Gaussian visible layer and a binary hidden layer because the artificial input feature vectors are asymptotically Gaussian distribution. The multiple Bernoulli–Bernoulli RBMs are then stacked behind the first Gaussian–Bernoulli RBM [14]. A faster learning

procedure called contrastive divergence is utilized to train the first Gaussian–Bernoulli RBM [14], [15] as an unsupervised learning to minimize the negative log probability of the training vectors. Then, the second Bernoulli–Bernoulli RBM can be trained using the first Gaussian–Bernoulli RBM's hidden layer as the second RBMs visible layer [14].

As mention before, the DNN is trained by an algorithm known as backpropagation [15]. In the training stage of this work, the weights and biases are initialized by pretraining [15]. After the pretraining, the weight matrix $\mathbf{W}$ with the biases for layer i obtained the multiple RBMs can be used as an effective starting point for fine-tuning by using backpropagation [14] as shown in Fig. 3. The objective criterion employs the MMSE criterion [13] based on a minibatch scaled conjugate gradient optimizer [22] between the estimated BP and reference BP as

$$C = \frac{1}{N}\sum_{n=1}^{N}\sum_{d=1}^{D}\left(\hat{\mathbf{Y}}_n^d(\mathbf{W},\mathbf{b}) - \mathbf{Y}_n^d\right)^2 \tag{12}$$

where C denotes the cost function, $\hat{\mathbf{Y}}_n^d(\mathbf{W},\mathbf{b})$ and $\mathbf{Y}_n^d$ represent the dth estimated and reference BP vectors at the sample index n, N and D represents the mini-batch size and the feature vector's size, and (W^i, b^i) represent the weights and bias parameters, which are learned at the ith layer. The estimated weights and bias can then be updated iteratively as follows:

$$\begin{aligned}\left(\mathbf{W}_{n+1}^i, \mathbf{b}_{n+1}^i\right) &= -\epsilon\frac{\partial C}{\partial(\mathbf{W}_n^i, \mathbf{b}_n^i)} \\ &\quad + \eta\left(\mathbf{W}_n^i, \mathbf{b}_n^i\right), 1 \le i \le L+1\end{aligned} \tag{13}$$

where ϵ is the learning rate, η denotes the momentum parameter, L represents the number of hidden layers, and $L+1$ is the output layer. Note that the trained networks are known to suffer from the poor local minima, and indeed, it is not possible to make the model reach the global minimum exactly if the entire error surface is not discovered. However, it is reported in the literature [28] that the poor local minima are rarely a problem since all local minima with well-trained large networks are likely to have an error very close to that of the global minimum, even though the number of the local minima increases in proportional to the network size. Interested readers are referred to recent theoretical and empirical studies, which suggest that local minima are not a serious issue in general [29].

To summarize, the DBN-DNN-based regression model is developed to learn the complex mapping function between the artificial feature vectors and the target BPs, which can automatically earn the complicated relationship to estimate the target BPs from the artificial feature vectors given the sufficient number of training samples by using the parametric bootstrap [18], in the DBN-DNN training stage. Finally, the SBP and DBP in our DNN estimation stage are wonderfully estimated by using the DBN-DNN obtained from the training stage.

IV. Experimental Results

A. Experimental Procedure

Our oscillometric BP measurements were acquired using an automated wrist BP monitors given by (Biosign Technologies Inc., Toronto, ON, Canada). In particular, the experimental BP measurement set was obtained from 85 healthy subjects aged from 12 to 80 consisting of 37 females and 48 males with no history of cardiovascular disorders [1]. Oscillometric BP measurements were measured from each volunteer with five sets ($5 \times 85 = 425$ total measurements according to the recommendations of the ANSI/AAMI SP 10 standard protocol [12]. The two observer readings were averaged to provide one SBP and one DBP reading as the target BP. More specifically, our BP measurements were composed of SBP and DBP readings guided by two trained observers following 1 min of rest. This was then repeated after another minute of rest. This step was repeated again four more times to build a recording of five BP measurements. During the BP data collection, each subject comfortably sat upright in a chair and the UFIT monitor's cuff was strapped to the left wrist of the subject, which was raised to heart level. The auscultatory cuff, which was the reference device, was located on the upper left arm, which was also at heart level. In particular, the upper cuff was inflated around the arm to occlude the brachial artery. Turbulent blood flow created Korotkoff sounds when the cuff pressure was deflated, which could be quite easily heard by using a stethoscope. The first Korotkoff sound, which was determined in units of mmHg by a manometer connected to the upper cuff, was utilized to estimate the SBP, while the fifth sound was used to determine the DBP [31].

However, concurrent brachial and wrist measurements were impossible due to the problem of occlusion of brachial arteries by upper arm sphygmomanometers. So, nearly 1.5 min after each pulse wave signal obtained by the UFIT write BP device, two trained nurses simultaneously recorded (SBP_1 and SBP_2) and (DBP_1 and DBP_2) using a classic upper arm sphygmomanometer provided with means for simultaneous readings. Readings with subscript 1 were acquired by the first nurse, and readings with subscript 2 were acquired by the second nurse. Therefore, the results of five pairs are given by $\{\text{SBP}_{1,j}, \text{SBP}_{2,j}, |j = 1, \cdots, 5\}$ and $\{\text{DBP}_{1,j}, \text{DBP}_{2,j}, |j = 1, \cdots, 5\}$ for each subject, respectively.

In time, each of five classic upper arm sphygmomanometer measurements acquired by the first and second nurses $\{(\text{SBP}_{1,j}, \text{SBP}_{2,j}), (\text{DBP}_{1,j}, \text{DBP}_{2,j}), |j = 1, \cdots, 5\}$ corresponded to an interval of roughly 1.5 min to each of the five pulse wave signal recorded by the automated oscillometric BP device $\{\text{UFIT}_j | j = 1, \cdots, 5\}$ for each subject. This interval between the classic upper arm and wrist measurements could be not only as short as possible to minimize the natural BP variability over time but also long enough to let the system settle down after the occlusion of arteries during measurements. The interval of about 1.5 min between arm and wrist measurements was chosen by a compromise in minimizing the method errors [12]. For more details on the data collection, the interested reader is referred to [12] and [30].

We conducted the training and test experiments in order to verify the proposed DBN-DNN-based regression model. In the proposed test scenario, the measurements of subjects were sequentially divided into the training set (250 measurements obtained from 50 subjects with five measurements) and testing set (175 measurements obtained

TABLE V
STATISTICAL INFORMATION FOR 85 SUBJECTS [30]

Statistical information	Value
Arm size	25 (cm) to 42 (cm)
Wrist size	13.5 (cm) to 23 (cm)
Deflation rate	3.0 (mmHg/s)
Male	48 of 85 (56.5%)
Female	37 of 85 (43.5%)
Age (Male)	12 to 80
Age (Female)	17 to 65

TABLE VI
PARAMETER SETTING [14]–[16] OF THE DBN-DNN-BASED REGRESSION MODEL, WHERE 8 IS THE NUMBER OF INPUT UNITS (NAMELY, THE INPUT VECTOR'S DIMENSION) AND 2 DENOTES THE NUMBER OF OUTPUT UNITS (NAMELY, THE TARGET VECTOR'S[SBP AND DBP] DIMENSIONS)

Number of the hidden unit in three layers:	$[(8,(32),(32),(32),2)]$
Number of feature	8
Number of sample over each artificial feature	100
Number of hidden layers	3
Number of hidden unit on the layers	16 to 128
Learning rate for weight	0.001
Learning rate for biases of visible units	0.01
Learning rate for biases of hidden units	0.01
Momentum rate	0.9
Activation type	logistic function
Number of epoch in the pretraining	100 to 400
Number of epoch in the fine-tuning	100 to 1000
Initial weights and biases	randomly between $(-1, 1)$

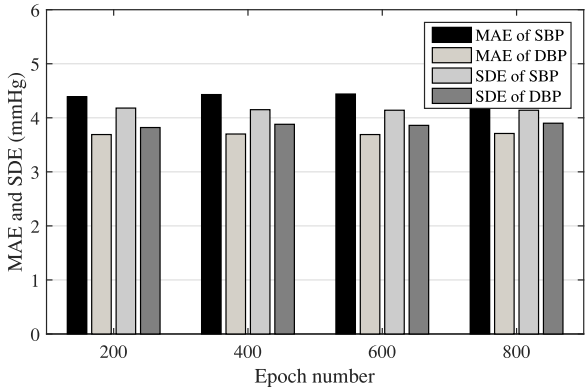


Fig. 4. Summary of the MAE and SDE obtained using the proposed DNN-based regression model with different epoch numbers relative to the auscultatory nurse measurements [21].

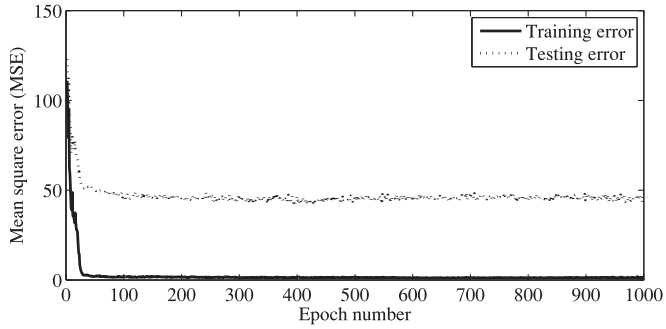


Fig. 5. MSE using the DBN-DNN regression model over different epoch numbers relative to the auscultatory measurements (target BPs) [35].

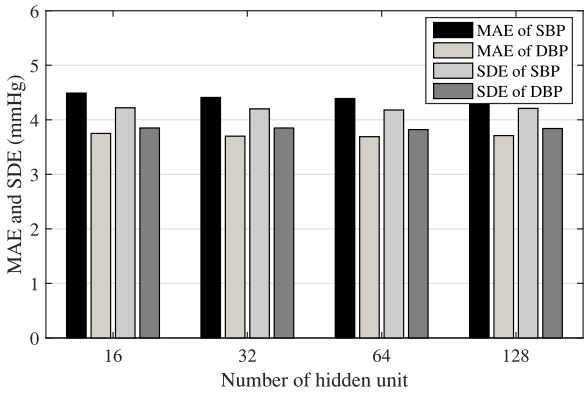


Fig. 6. Summary of the MAE and SDE obtained using the proposed DNN-based regression model with different numbers of hidden unit relative to the auscultatory nurse measurements [10].

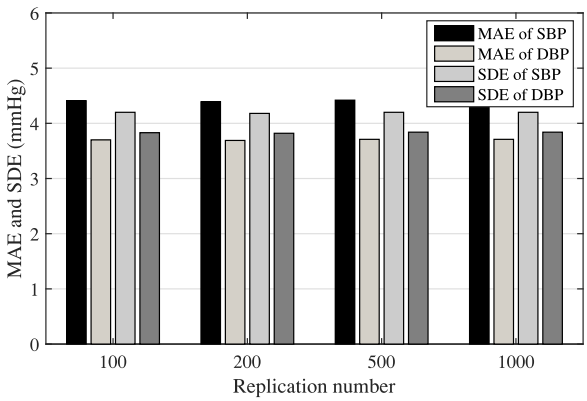


Fig. 7. Summary of the MAE and SDE obtained using the proposed DNN-based regression model with different replication numbers relative to the auscultatory nurse measurements [18].

from 35 subjects with five measurements). This procedure was then repeated such that each subject was involved once in the testing stage. Testing in the proposed test scenario may be viewed as examining the generalizing ability of the proposed approach in measurements from new subjects. Indeed, five measurements (i.e., five samples for each feature) from each subject constitute a small amount of the input data in the training stage. Therefore, we applied the artificial features obtained from the original feature vectors. Note that in the test stage, we used the unseen original feature set to verify the proposed algorithm. As previously mentioned, our feature vectors were obtained from the OMW, which were used to generate the artificial feature vectors based on the parameter bootstrap [18]. We thus acquired the artificial feature vectors with (from $B = 5$ to $B = 2000$) training samples for each subject, which implies that each feature has (from $B = 5$ to 2000) training samples. Namely, we obtained (5000 input vectors $=$ 50 subject $\times$ B, where at $B = 100$) over each feature in the training stage.

TABLE VII
FFNN ARCHITECTURE [10] AND TRAINING PARAMETERS

Number of the hidden unit in three layers:	$[(8,(32),(32),(32),2)]$
Number of feature	8
Number of sample over each artificial feature	100
Number of hidden layers	3
Number of hidden unit on the layers	16 to 128
Initial weights and biases	randomly between $(-1, 1)$
Activation type	logistic function
Learning rate for hidden layers	0.01
Number of epoch in the training	100 to 1000
Momentum rate	0.9

TABLE VIII
COMPARISON OF THE COMPLEXITY OF IN TERMS OF RUNNING TIME [32], WHERE THE SPECIFICATIONS OF SYSTEM ARE INTEL CORE(TM) I7-4790 CPU 3.60 GHz, RAM 32.0 GB, OS 64 BIT, AND MATLAB 2015 (THE MATHWORKS INC., NATICK, MA, USA)

Method	FFNN [10]	DBN-DNN
Total time (s)	308.442	319.814

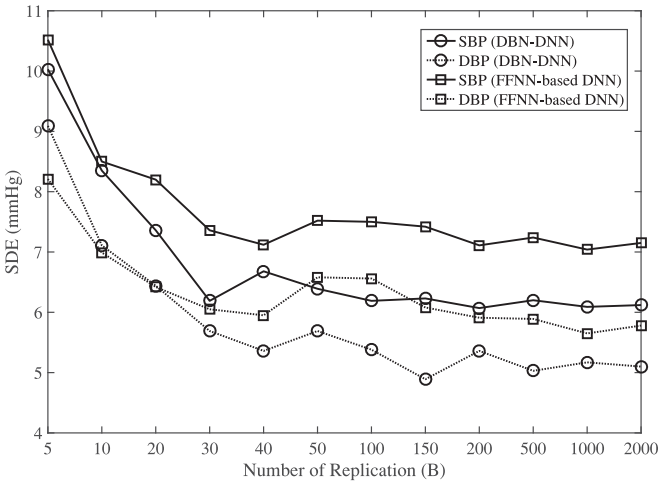


Fig. 8. SDE of ME [12] using the proposed DBN-DNN-based regression model compared to the that of ME using the FFNN [10] over different training sample, where B denotes the number of replication about the artificial feature vectors with (from $B = 5$ to 2000) training samples for each subject, which implies that each feature has (from $B = 5$ to 2000) training samples. As a result, we acquired 50 000 input vectors (i.e., 50 subject $\times$ B, in case of $B = 1000$) over each feature in the training stage.

B. Statistical Analysis

Table V summarizes the statistical information for all subjects, and the configuration and parameter setting of our DBN-DNN model is shown in Table VI. Specifically, the number of feature (8) represents the eight feature vectors such as (e.g., the MAP) the number of sample over each artificial feature (100) obtained using the parametric bootstrap as shown in Section III. Based on these, we first tuned the epoch number in the DBN-DNN formations as shown in Fig. 4. As a result, the performance was drastically improved until 50 epochs, and then, it was up and down constantly when the epoch number was increased until 1000 as shown in Fig. 5. We then tested the performance of the proposed algorithm on the same conditions with a different number of hidden units from 16 to 128 due to the dimension $\mathbb{R}(= 10)$ of the input data. The best result was observed at 64 hidden units as shown in Fig. 6. Also, it was demonstrated that our DBN-DNN model was robust across the different hidden unit from 16 to 128 based on the MAE and SDE [21].

We also performed another experiment to verify the performance of the proposed algorithm based on the same conditions with the number of replications (B) from 100 to 1000 for the artificial features. From Fig. 7, we observed that the performance of the proposed algorithm had the best performance at 200 for B. However, the smaller number of replication (B) yields that the MAE and SDE were increased. In addition, we compared the computational complexity of the conventional FFNN and the proposed algorithm to investigate the additional computational cost when using identical parameters as shown in Tables VI and VII. We set the number of hidden unit and the number of

TABLE IX
ME AND SDE RELATIVE TO THE REFERENCE AUSCULTATORY METHOD [12] OBTAINED WITH THE CONVENTIONAL MAA [3], MLR [33], MLR WITH LASSO REGULARIZER MLR_{LR} [34], FFNN [10], SVR [19], AND USING THE DBN-DNN IN THE TEXT, WHERE THE RESULTS ARE THE AVERAGE VALUES FOR OUR TEST DATA

mmHg	MAA		MLR		MLR_{LR}		FFNN		SVR		DBN-DNN	
Test	SBP	DBP	SBP	DBP	SBP	DBP	SBP	DBP	SBP	DBP	SBP	DBP
ME	1.07	1.12	0.11	−0.37	−0.02	0.01	−0.30	−1.79	−0.05	−0.24	0.01	0.03
SDE	9.03	7.40	10.94	7.78	10.41	7.67	7.61	6.79	7.12	6.09	6.35	5.28

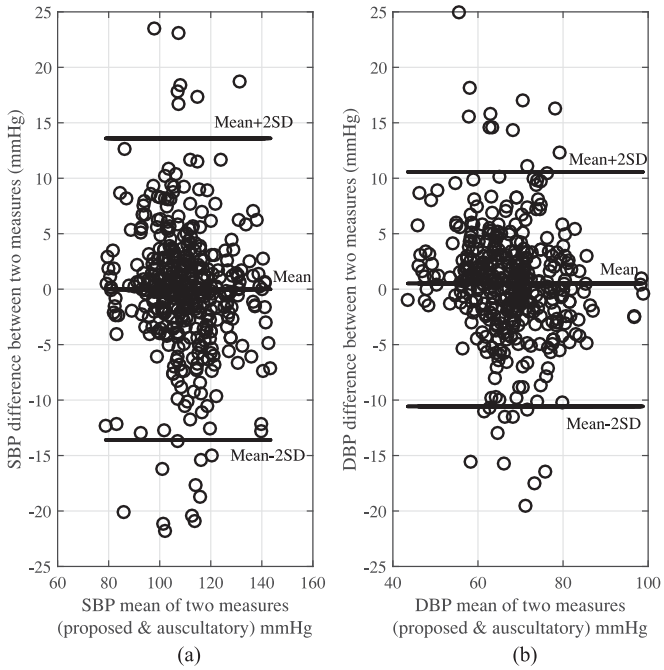


Fig. 9. Bland–Altman plots comparing the performance between the proposed DBN-DNN-based regression model and the auscultatory nurse measurements [12]. (a) Bland–Altman plot for the SBP. (b) Bland–Altman plot for the DBP.

TABLE X
GRADING OF THE PROPOSED ALGORITHM BASED ON THE BHS STANDARD [21] USING THE RESULTS OF MAA [3], MLR [33], MLR_{LR} [34], FFNN [10], SVR [19], AND DBN-DNN ON ($5 \times 85 = 425$) MEASUREMENTS

	SBP Absolute difference (%)			DBP Absolute difference (%)			Standard (SBP/DBP) BHS
Tests	≤5 mmHg	≤10 mmHg	≤15 mmHg	≤5 mmHg	≤10 mmHg	≤15 mmHg	Grade
MAA	47.06	85.88	96.47	56.47	88.24	97.65	C/B
MLR	45.88	70.82	87.76	51.06	80.94	93.65	C/B
MLR_{LR}	47.06	72.00	87.76	52.94	82.12	93.88	C/B
FFNN	53.88	85.65	95.53	66.12	94.12	98.82	B/A
SVR	62.59	86.12	95.53	74.12	93.65	96.94	A/A
DBN-DNN	69.65	90.12	95.53	76.19	95.76	98.82	A/A

epoch as 32 and 100 to compare the computational running time between the conventional FFNN and the proposed algorithm in Table VIII. The remaining parameters was set as in Tables VI and VII. Note that the running time is eventually calculated based on the performance in MATLAB 2015 [32]. The result showed that the proposed algorithm did give slightly higher computation 11.371 (s) compared with the conventional FFNN [10], which is mainly due to the pretraining step.

In order to evaluate the overall performance of the proposed algorithm, we investigated the ME and the SDE between the estimated BP and the reference BP according to the recommendations of the AAMI standard protocol[12]. Supposing a device's measurement error has an ME value of less than 5 mmHg with an SDE of no more than 8 mmHg, the device could pass the AAMI protocol [12]. Thus, lower values of the ME and SDE indicate better overall performance. However, the SDE is more important than the ME because a device may be very inaccurate and the measurements can have small ME with large errors, which are equally probable to be positive or negative. The SDE of ME using the DBN-DNN model decreased gradually as compared with that using the FFNN with different training sample. Most importantly, the advantage of the DBN-DNN model did not vanish as the number of training sample increases as shown in Fig. 8. This implies that the distribution of the artificial input data fits better to the DBN-DNN than the FFNN architecture because the proposed DBN-DNN is the probabilistic graphical model. Meanwhile, it turned out that the FFNN architecture did not take full advantage of the increased input data. Note that we used the FFNN model with same training and testing conditions to fairly evaluate the performance of the proposed DBN-DNN-based regression model as shown in Fig. 3(b) and Table VII. The MEs of the SBP and DBP acquired using the proposed DBN-DNN-based regression model were compared with those of the MAA, SVR [19], MLR [33], MLR with lasso regularizer [34] which was developed in MATLAB 2015 (The MathWorks Inc., Natick, MA, USA), and FFNN [10], as the opponent algorithms as shown in Table IX. Note that the artificial features were used in the opponent algorithms except the MAA that is the conventional BP estimation algorithm using the characteristic fixed ratio based on the oscillometric envelope. Thus, artificial features are not used in the MAA. From Table IX, the SDE acquired by our DNN was found to be 6.35 and 5.28 mmHg for the SBP and DBP, respectively. The proposed DNN was improved by 2.68 and 2.12 mmHg for the SBP and DBP compared with the conventional MAA with respect to the SDE as shown in Table IX. There were differences of 1.78 and 1.01 mmHg for the SBP and DBP, respectively, in the SDE between the proposed DNN and FFNN [10], [20]. Moreover, comparing the proposed DNN to the SVR, it was observed that the SDEs of the SBP and DBP were reduced by 0.77 and 0.81 mmHg, respectively. In addition, the MLR and MLR with lasso regularizer MLR_{LR} were presented as shown in Table IX. Specifically, we used the SVR model using a linear epsilon insensitive cost [19] as the opponent algorithm in the current state of the art in order to fairly evaluate the performance of the proposed DBN-DNN-based regression model.

Moreover, the percentages of the MAE for three categories, ≤5 mmHg, ≤10 mmHg, and ≤15 mmHg, are computed for all measurements (425 measurements). The protocol of the BHS with an A–D graded system would grant a grade of A to a device if 60% of its error measurements are within 5 mmHg, 85% of its error measurements are within 10 mmHg, and 95% of its error measurements fall within 15 mmHg [21]. In addition, Bland–Altman plots comparing the performance of the proposed DBN-DNN-based regression model and the auscultatory nurse measurements (425 measurements) are shown in Fig. 9. Furthermore, the results of the BHS protocol [21] indicate that our DNN provided accurate BP estimates when compared with the MAA, MLR, MLR_{LR}, FFNN, and SVR [19]. In Table X, we presented the BHS grading obtained by our proposed DNN-based regression algorithm. The results of the proposed method were 69.52% (≤5 mmHg), 90.00% (≤10 mmHg), and 95.48% (≤15 mmHg) for the SBP in the test scenario and 76.19% (≤5 mmHg), 95.71% (≤10 mmHg), and 98.82% (≤15 mmHg) for the DBP in the given test scenario.

V. DISCUSSION AND CONCLUSION

We provided the BP estimation using the DBN-DNN with a help of the parametric bootstrap technique. As mentioned in Section IV, it is discovered that the proposed DBN-DNN-based regression model was superior to the conventional algorithms as shown in Tables IX and X. These indicate that sufficient artificial features are of importance for improving the generalization capacity of the proposed DBN-DNN model because the distribution of artificial input data is better match to the

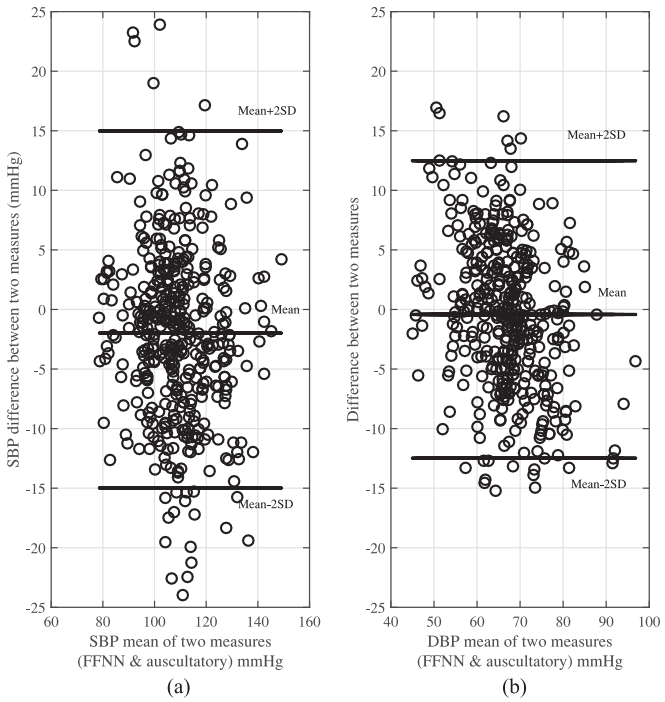


Fig. 10. Bland–Altman plots comparing the performance between the FFNN model and the auscultatory nurse measurements [12]. (a) Bland–Altman plot for the SBP. (b) Bland–Altman plot for the DBP.

DBN-DNN statistical model than the FFNN-based DNN structure. Consequently, the proposed DBN-DNN-based regression model acquires an overall grade of A based on the BHS grading system. In addition, we evaluated the performance of the DBN-DNN-based regression model by using the Bland–Altman plot as shown in Fig. 9. This implies that the BP estimates obtained by the proposed DNN-based regression model are in close agreement with those obtained by the referenced BPs. The limits of agreement (see bold horizontal lines in Fig. 9) that we used were ME $\pm$ 2 $\times$ SDE for two plots; the most black circles lie within the limits of agreement. The bias (see horizontal center lines) for the two plots turned out to be very small ($\leq \pm$ 0.5 mmHg). Note that Fig. 9 well represents a result corresponding to the last row in Table IX. The accordance between the FFNN and referenced BPs was also compared by the Bland–Altman plot (see Fig. 10). Therefore, from the results of the overall evaluation, we clearly showed that the performance of the proposed DBN-DNN-based regression with artificial features using the parametric bootstrap is as excellent as those obtained using the conventional methods.

In conclusion, the DBN-DNN-based regression model resulted in a lower SDE and ME for the SBP and the DBP compared with conventional methods. To the best of the authors' knowledge, this paper is the first study to estimate the SBP and DBP using the DBN-DNN-based regression model with artificial features based on the bootstrap technique. We believe that the proposed DBN-DNN model on small samples accounts for a major contribution to estimate BPs (SBP and DBP). We also confirm that the distribution of an artificial feature is asymptotical Gaussian that is very well fit for the DBN-DNN model as the number of training data increases. Even though the proposed approach was performed on the 85 healthy subjects based on the ANSI/AAMI SP 10 and BHS protocols, our future work will try the clinical testing on a wide range of new subject populations.

Acknowledgment

The authors would like to thank Biosign Technologies Inc. for providing them with equipment and data.

References

[1] S. Lee, M. Bolic, V. Z. Groza, H. R. Dajani, and S. Rajan, "Confidence interval estimation for oscillometric blood pressure measurements using bootstrap approaches," *IEEE Trans. Instum. Meas.*, vol. 60, no. 10, pp. 3405–3415, Oct. 2011.

[2] K. Soueidan, S. Chen, H. Dajani, M. Bolic, and V. Groza, "Augmented blood pressure measurement through the noninvasive estimation of physiological arterial pressure variability," *Physiol. Meas.*, vol. 33, no. 6, pp. 881–899, May 2012.

[3] S. Lee *et al.*, "Oscillometric blood pressure estimation based on maximum amplitude algorithm employing Gaussian mixture regression," *IEEE Trans. Instum. Meas.*, vol. 62, no. 12, pp. 3387–3389, Dec. 2013.

[4] S. Lee C.-H. Park and J.-H. Chang, "Improved Gaussian mixture regression based on pseudo feature generation using bootstrap in blood pressure measurement," *IEEE Trans. Ind. Informat.*, doi: 10.1109/TII.2015.2484278, to be published.

[5] J. Liu, J. O. Hahn, and R. Mukkamala, "Error mechanisms of the oscillometric fixed-ratio blood pressure measurement method," *Ann. Biomed. Eng.*, vol. 41, no. 3, pp. 587–597, Mar. 2013.

[6] R. Raamat, J. Talts, K. Jagomagi, and J. Kivastik, "Errors of oscillometric blood pressure measurement as predicted by simulation," *Blood Press. Monit.*, vol. 16. no. 5, pp. 73–76, Jun. 2011.

[7] M. Forouzanfar, S. Ahmad, I. Batkin, H. Dajani, V. Groza, and M. Bolic, "Coefficient-free blood pressure estimation based on pulse transit time-cuff pressure dependence," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 3, pp. 608–618, Mar. 2012.

[8] P. D. Baker, J. A. Orr, D. R. Westenskow, and T. P. Egbert, "Method for determining blood pressure utilizing a neural network," U.S. Patent 5 339 818, Aug. 23, 1994.

[9] S. Narus, T. Egbert, T.-K. Lee, J. Lu, and D. Westenskow, "Noninvasive blood pressure monitoring from the supraorbital artery using an artificial neural network oscillometric algorithm," *J. Clin. Monit.*, vol. 11, no. 5, pp. 289–297, Sep. 1995.

[10] M. Forouzanfar, H. Dajani, V. Groza, M. Bolic, and S. Rajan, "Feature-based neural network approach for oscillometric blood pressure estimation," *IEEE Trans. Instum. Meas.*, vol. 60, no. 8, pp. 2786–2796, Aug. 2011.

[11] P. D. Baker, "Neural network processing of oscillometric waveforms and blood pressure estimation from the superficial temporal artery," Ph.D. dissertation, Univ. Utah, Salt Lake City, UT, USA, 1990.

[12] *Association for the Advancement of Medical Instrumentation (AAMI), American National Standard Manual, Electronic or Automated Sphygmonanometers*, AASI/AAMI SP 10:2002, 2003.

[13] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE Trans. Audio, Speech, Language Process.*, vol. 23. no. 1, pp. 7–19, Jan. 2015.

[14] G. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[15] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.

[16] X. L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 4, pp. 697–710, Apr. 2013.

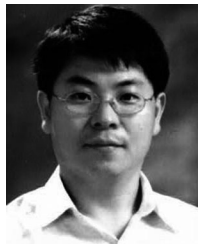
[17] D. Erhan, Y. Bengio, A. Courville, P. A. Manzagol, and P. Vincent, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, 2010.

[18] B. Efron, *Introduction to the Bootstrap*, London, U.K.: Chapman & Hall, 1993.

[19] A. Rakotomamonjy, "Analysis of SVM regression bound for variable ranking," *Neurocomputing*, vol. 70, pp. 1489–1491, Mar. 2007.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[21] E. O'Brien *et al.*, "The British Hypertension Society protocol for the evaluation of blood pressure measuring devices," *J. Hypertension*, vol. 11, no. 2, pp. 43–62, 1993.
[22] M. Moller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Netw.*, vol. 6, no. 4, pp. 525–533, 1993.
[23] S. Theodoridis, *Machine Learning*. San Francisco, CA, USA: Academic, 2015.
[24] S. Theodoridis, *Pattern Recognition*. San Francisco, CA, USA: Academic, 2009.
[25] A. Suzuki and K. Ryu, "Feature selection method estimating systolic blood pressure using the Taguchi method," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1077–1085, May 2014.
[26] J. Frank and Jr. Massey, "The Kolmogorov-Smirnov test for goodness of fit," *J. Amer. Statist. Assoc.*, vol. 46, no. 253, pp. 68–78, 1951.
[27] K. Singh, "On the asymptotic accuracy of Efron's bootstrap," *Ann. Statist.*, vol. 9, no. 6, pp. 1187–1195, 1981.
[28] L. Yann, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
[29] Y. N. Dauphin, P. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *Proc. Adv. Neural. Inf. Process. Syst. Conf.*, Jun. 2014, vol. 10, pp. 1–14.
[30] S. Ahmad, M. Bolic, H. Dajani, V. Groza, I. Batkin, and S. Rajan, "Measurement of Heart rate variability using an oscillometric blood pressure monitor," *IEEE Trans. Instum. Meas.*, vol. 59, no. 10, pp. 2575–2590, Oct. 2010.
[31] M. R. Neuman, "Measurement of blood pressure," *IEEE Pulse*, vol. 2, no. 2, pp. 39–43, Mar./Apr. 2011.
[32] M. Knapp-Cordes and B. McKeeman, "Improvements to tic and toc functions for measuring absolute elapsed time performance in MATLAB, " in, *Matlab Technical Articles and Newsletters*. Natick, MA, USA: The MathWorks Inc., 2011.
[33] S. Lee, C. Lim, and J.-H. Chang, "A new *a priori* SNR estimator based on multiple linear regression technique for speech enhancement," *Digital Signal Process.*, vol. 30, no. 7, pp. 154–164, Jul. 2014.
[34] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *J. Roy. Statist. Soc.*, vol. 58, no. 1, pp. 267–288, 1996.
[35] S. Lee, S. Rajan, C. H. Park, J.-H. Chang, H. Dajani, and V. Groza, "Estimated confidence interval from single blood pressure measurement based on algorithm fusion," *Comput. Biol. Med.*, vol. 62, pp. 154–163, Jul. 2015.



Soojeong Lee received the Ph.D. degree in computer engineering from Kwangwoon University, Seoul, South Korea, in 2008.

From 2009 to 2011, he was a Postdoctoral Fellow in the School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada. He is a Research Professor with the Institute of Electrical Information and Communication, Hanyang University, Seoul. His research interests include biosignal processing, instrumentation and measurement, speech enhancement, noise estimation, and reduction.

Joon-Hyuk Chang (SM'12) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 1998, and the M.S. and Ph.D. degrees in electrical engineering from Seoul National University, Seoul, South Korea, in 2000 and 2004, respectively.

From March 2000 to April 2005, he was a Chief Engineer with Netdus Corp., Seoul. From May 2004 to April 2005, he was a Postdoctoral Researcher with the University of California, Santa Barbara, where he worked on adaptive signal processing and audio coding. In May 2005, he joined Korea Institute of Science and Technology, Seoul, as a Research Scientist to work on speech recognition. From August 2005 to February 2011, he was an Assistant Professor in the School of Electronic Engineering, Inha University, Incheon, South Korea. He is currently an Associate Professor in the School of Electronic Engineering, Hanyang University, Seoul. His research interests include speech coding, speech enhancement, speech recognition, audio coding, and adaptive signal processing.

Dr. Chang has received the IEEE/IEEK IT Young Engineer Award of the year 2011. He is the Editor-in-Chief of the *Signal Processing Society Journal* of the Institute of Electronics Engineers of Korea.