This article was downloaded by: *[Hoya, Tetsuya]*
On: *4 February 2010*
Access details: *Access Details: [subscription number 919053762]*
Publisher *Taylor & Francis*
Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK

Connection Science
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713411269

A cascaded neuro-computational model for spoken word recognition

Tetsuya Hoya [ab]; Cees van Leeuwen [c]
[a] Department of Mathematics, College of Science & Technology, Nihon University, Tokyo, Japan [b] Laboratory for Advanced Brain Signal Processing/Laboratory for Perceptual Dynamics, Brain Science Institute, RIKEN, Saitama, Japan [c] Laboratory for Perceptual Dynamics, Brain Science Institute, RIKEN, Saitama, Japan

Online publication date: 03 February 2010

To cite this Article Hoya, Tetsuya and van Leeuwen, Cees(2010) 'A cascaded neuro-computational model for spoken word recognition', Connection Science, 22: 1, 87 — 101

To link to this Article: DOI: 10.1080/09540090903383171

URL: http://dx.doi.org/10.1080/09540090903383171

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: http://www.informaworld.com/terms-and-conditions-of-access.pdf

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

Connection Science
Vol. 22, No. 1, March 2010, 87–101

A cascaded neuro-computational model for spoken word recognition

Tetsuya Hoya[a,b]* and Cees van Leeuwen[c]

[a] *Department of Mathematics, College of Science & Technology, Nihon University, Tokyo, Japan;* [b] *Laboratory for Advanced Brain Signal Processing/Laboratory for Perceptual Dynamics, Brain Science Institute, RIKEN, Saitama, Japan;* [c] *Laboratory for Perceptual Dynamics, Brain Science Institute, RIKEN, Saitama, Japan*

(*Received 3 December 2008; final version received 31 July 2009*)

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

In human speech recognition, words are analysed at both pre-lexical (i.e., sub-word) and lexical (word) levels. The aim of this paper is to propose a constructive neuro-computational model that incorporates both these levels as cascaded layers of pre-lexical and lexical units. The layered structure enables the system to handle the variability of real speech input. Within the model, receptive fields of the pre-lexical layer consist of radial basis functions; the lexical layer is composed of units that perform pattern matching between their internal template and a series of labels, corresponding to the winning receptive fields in the pre-lexical layer. The model adapts through self-tuning of all units, in combination with the formation of a connectivity structure through unsupervised (first layer) and supervised (higher layers) network growth. Simulation studies show that the model can achieve a level of performance in spoken word recognition similar to that of a benchmark approach using hidden Markov models, while enabling parallel access to word candidates in lexical decision making.

Keywords: automatic speech recognition; human speech recognition; spoken word recognition; artificial neural networks; network growing model

1. Introduction

A central problem in processing of auditory information is the recognition of spoken words. In the field of automatic speech recognition, approaches based upon hidden Markov models (HMMs) have been most successful (Holmes and Huckvale 1994) and constitute a benchmark for any approaches aiming for greater plausibility of their neural layout. The latter differ from conventional HMMs, often having an explicit hierarchical structure that incorporates both pre-lexical and lexical levels (McClelland and Elman 1986; Allen 1994; Norris 1994; Plaut and Kello 1999; Scharenborg, Norris, ten Bosch, and McQueen 2005; Norris and McQueen 2008). This hierarchical structure enables these models to generate expectations about word candidates before the utterance is completed. Because of this, they can accommodate a growing body of

*Corresponding author. Email: hoya@math.cst.nihon-u.ac.jp

ISSN 0954-0091 print/ISSN 1360-0494 online
© 2010 Taylor & Francis
DOI: 10.1080/09540090903383171
http://www.informaworld.com

literature on the dynamic integration of information about word candidates during the course of the utterance (Allopenna, Magnuson, and Tanenhaus 1998).

Illustrative is the model known as TRACE (McClelland and Elman 1986). TRACE was motivated by classical models of word recognition such as Cohort (Marslen-Wilson 1987) and the conceptual *logogen* model (Morton 1969). The TRACE model uses sub-word level information by combining feature, pre-lexical and lexical representation layers. Activation in the lexical layers builds up, based on pre-lexical information collected during the utterance; the system generates multiple expectations simultaneously for various word candidates that are consistent with the pre-lexical information so far as presented.

The TRACE model was able reasonably to account for segmenting a continuous speech stream into isolated words and to simulate the effects of lexical competition and prediction. In this early model, network structure is completely pre-specified, leaving no room for learning (Christiansen and Chater 1999). It is still under debate whether feedback connections as used in TRACE are needed for word recognition (McClelland, Mirman, and Holt 2006; Mirman, McClelland, and Holt 2006).

Word recognition in the TRACE model is limited, as it takes only artificial speech segments (i.e., ‘wickelphones’) as input, leaving aside the problem of processing real speech data. To our knowledge, no neurally plausible hierarchical models of spoken word recognition exist that can actually handle real speech samples (see also Scharenborg 2007). Scharenborg et al. (2005) proposed the SPeM model; a model that can process real speech input in a hierarchical fashion, comprising of both the pre-lexical and lexical levels. While incorporating some aspects of psychological models, SPeM can hardly be seen as a hierarchical connectionist model but rather is an augmented version of automatic speech recognition based upon HMM.

The present article proposes the Constructive Cascaded-Layer Neural Network, a neurocomputational model for spoken word recognition that is equipped with both the pre-lexical and lexical layers and deals with real speech data. The theory behind the actual network construction is based upon the concept of kernel memory (Hoya 2005). Like a conventional neural network, the kernel memory uses units to represent items in memory, linked by weighted connections.[1] In three respects, the system differs from a conventional neural network. First, rather than using artificially segmented speech as input, what each unit represents is determined by a particular ‘kernel’ function, a concept of kernel regression theory. Second, rather than having static receptive fields, the receptive fields in this model are adaptive. In the pre-lexical layer, the receptive field is modelled by a radial basis function (RBF). The RBF has become accepted in a neurobiological context: Hopfield (1995) suggests that, in order to recognise an object, the functional unit based upon an RBF yields a more powerful biological device, compared with a sigmoidal function as extensively exploited in multi-layered perceptron neural networks (MLP-NNs).

Third, in order to benefit from sub-word level information in the recognition process, a three-layered structure is prescribed for the network. The first layer passes on its activity in a winner-takes-all manner to the second layer. The second layer is the lexical layer. It is composed of units that measure the similarity between their own, internal template and a series of labels, corresponding to the winning receptive fields in the first, pre-lexical layer. Thereby, each unit in the second layer performs serial-order detection of winning units in the first layer and eventually outputs the value of similarity as its activation. Pulvermüller (2002) proposed a related idea in the field of syntactical processing. Supported by neuropsychological/psycholinguistic findings, he proposed that such a serial-order detection mechanism could be found at sub-cortical level in the brain. The third layer is the output layer that eventually performs lexical competition based upon the total sum values of activation from the second layer units.

The three-layered structure operates merely as a scaffold; the content of each level, unlike in the TRACE model, is not determined *a priori*. Instead of the static and pre-defined network topology, the proposed model is topologically unconstrained beyond the three-layer prescription. There is initially only a single unit in the pre-lexical layer and no other units in the succeeding layers

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

within the network. The entire network, including the interlayer connections, grows automatically according to an incremental training rule, and thus the approach is data-driven. An advantage is that the model can acquire new patterns, where necessary, without retraining the whole network from scratch.

Moreover, even though the analysis of input speech is on a frame-to-frame basis at each layer, importantly, the lower-level representations vary at a faster time scale than the higher-level ones, as a result of *abstraction*. This feature of the model corresponds to the observation that, in human speech recognition, auditory signals are analysed through a layered architecture of the temporal cortex, where the signals are subdivided into smaller units and for speech signals bottom-up processing from the pre-lexical to lexical level occurs (cf. Allen 1994). Abstraction, in combination with unsupervised/supervised network growing, automatically yields a system in which the layers represent a meaningful, hierarchical segmentation of spoken word input. A related model for visual object recognition supported by neurophysiological data from visual cortex (Serre, Oliva, and Poggio 2007) similarly uses the efficacy of bottom-up processing through different hierarchical levels of abstraction.

In comparison with the TRACE model, our model does not operate with explicitly defined features at the phoneme level, but instead representations are implicitly determined by the RBFs.

Note also the discrepancy from ordinary, fully connected 'three-layered' perceptron models: First, while the proposed model has the aforementioned feed-forward-type three-layered architecture as in conventional MLP-NNs, the interlayer connections (i.e., both the connections between Layers 1 and 2 and those between Layers 2 and 3) are not necessarily fully connected. In fact, connectivity can be rather sparse; as a result of how it was established during the construction phase. Second, while each node in all the three layers of MLP-NNs is uniformly represented by a sigmoidal activation function, the units in the respective layers are represented by different types of activation functions within the cascaded-layer neural network model (i.e., RBFs in the first, another type of nonlinear functions measuring the similarity between the inner-held templates and inputs obtained from the activation of the first layer units in the second, and the linear sums in the third layer).

A detailed description of the proposed cascaded-layer neural network model is given in the next section.

2. The constructive cascaded-layer neural network

Figure 1 shows an illustrative example of a layered network structure for spoken word recognition that results after construction has been completed.

The first analysis of raw speech proceeds on a frame-to-frame basis. Raw speech is encoded into a sequence of mel-frequency cepstral coefficients (MFCCs; Furui 1981) and presented to the system as input (far left). Each frame is processed through three consecutive layers, each consisting of multiple units. The first layer represents the input frames in a winner-takes-all manner; thus a frame leads to a selection from multiple RBF units. The selection implies a process of abstraction. Activation from the winning unit is being fed into the next layer. Here, the selection process is repeated, leading to a higher level of abstraction, and likewise at the third level that eventually yields a final score for a particular lexical category.

2.1. *Construction of the network*

The model structure is built from scratch. Starting from a blank slate, the network is constructed based on the principle that at each level, a new unit is created whenever the ones already available

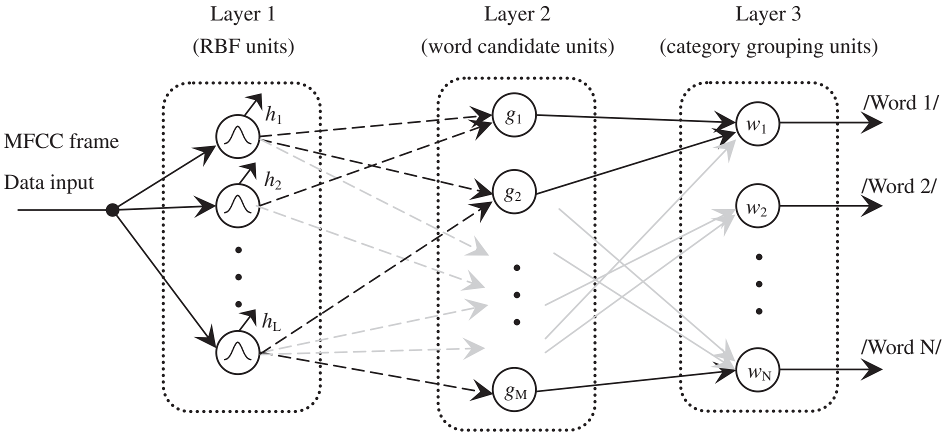

Figure 1. Cascaded neural network model for spoken word recognition (after completion of the construction) – in the figure, each unit in Layer 1 represents an RBF, and its activation value h_i $(i = 1, 2, \ldots, L)$ is calculated using Equation (1), given the MFCC frame data input x, whereas g_j $(j = 1, 2, \ldots, M$, in Equation (4)) and w_k $(k = 1, 2, \ldots, N$, in Equation (6)) are the activations from the word candidate units and category grouping units, respectively. The dashed connections between Layers 1 and 2 depict the forwarded input to g_j, consisting of not activation values (as in conventional artificial neural network models) obtained from h_i but symbols (i.e., representing maximally activated RBFs).

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

fail to generalise properly the information given. This automatically applies in the beginning, when no units are available.

In Figure 1, Layer 1 is constructed independently from Layers 2 and 3, while both Layers 2 and 3 are simultaneously constructed after all RBF units in Layer 1 are configured. The construction of the network as in Figure 1 thus takes the following two major steps:

Step 1: Construction of Layer 1 (i.e., the pre-lexical layer): addition of RBF units that are responsible for sub-word level information.

Step 2: Based upon RBF units added/configured in Layer 1 in Step 1), construction of Layers 2 and 3 (i.e., the word and category grouping units, respectively), as well as the interlayer connections, i.e., the connections between Layers 1 and 2 and those between Layers 2 and 3.

In Step 2, we have opted for a process of network growth: when a particular group of neuronal units are co-activated, given a stimulus input, a connection between them is added to the network. This connection relates the RBFs in Layer 1 and a category grouping unit in Layer 3 via a newly established unit in Layer 2. Connection growth takes place during a construction step (to be described later in detail): when a certain spoken word is given to the system, the units excited by/responsible for it all become connected to each other. This manner of establishing the link connections between the units in the layers is motivated by the postulate of Hebb (1949): ‘When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A’s efficiency, as one of the cells firing B, is increased’.

2.2. *RBF units*

As aforementioned, sub-word level information is obtained from a raw speech signal in the form of a set of MFCCs. The cepstral coefficients obtained for a single frame of raw speech signal are input to the sub-word level units, i.e., RBF units in the first layer. These units are set automatically to represent information in the form of ‘centroid’ vectors. In essence, centroid vectors represent the midpoints of clusters detected in the input data of the layer. The addition of these units, as well as setting their centroid vectors, occurs during the construction phase. In Figure 1, the output activation from the RBF unit h_i $(i = 1, 2, \ldots, L)$ is given as a Gaussian response function, i.e.,

$$h_i(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_i\|_2^2}{\sigma^2}\right) \tag{1}$$

where $\|\cdot\|_2$ denotes the $L2$-norm and $\mathbf{x}$ is the MFCC input data (for a single frame) and $\mathbf{c}_i$ and σ are the centroid vector and radius of the ith RBF unit, respectively. Then, as represented by Equation (1), pattern matching between $\mathbf{x}$ and $\mathbf{c}_i$ is essentially performed by measuring their Euclidean distance, followed by a nonlinear transformation. In comparison with an ordinary template matching scheme, this manner of pattern matching can be quite beneficial, since each centroid vector can be seen as a template vector and every data point around the centroid can be (moderately) covered by a regional space defined by a single template and the nonlinear function. Thus, by tuning effectively both the parameters $\mathbf{c}_i$ and σ, memory load can be dramatically reduced in order to cover a particular pattern space.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

2.3. *Construction of Layer 1*

The addition of RBF units to Layer 1 takes place in an unsupervised mode, according to Steps 1–3. RBF units are added to Layer 1 in a manner similar to the previously proposed network growing schemes (Carpenter, Grossberg, and Reynolds 1991; Platt, 1991; Hoya and Washizawa 2007)[2]:

Step 1: Collect all the MFCC frame data obtained from all available speech samples.

Step 2: Initially, there is only a single RBF, of which the centroid vector is identical to the input pattern obtained from the very first frame data of the first speech sample.

Step 3: Repeat the following, until all the frame data have been presented:

- *Step 3.1*: Calculate the activation of all RBF units in Layer 1, given the frame data as input (i.e., Equation (1)).
- *Step 3.2*: For each RBF unit, if the activation is above a certain threshold θ, which is given *a priori*, mark it as EXCITED.
- *Step 3.3*: If there is no excited RBF, add a new RBF in Layer 1, with its centroid vector identical to the input pattern. Then, unmark all the excited RBFs for the next iteration. Otherwise, do nothing.

According to Equation (1), each RBF unit has two free parameters, i.e., the radius σ and centroid vector $\mathbf{c}_i$. The former is given *a priori* as a scalar constant, while the latter is set during the construction of Layer 1 as specified by Steps 2 and 3 of the algorithm.

2.4. *Neural mechanisms for serial-order detection*

Since the formulation by McCulloch and Pitts (1943), the problem of how serial-order detection of multiple events can be neurally performed has been studied in the neurophysiological domain (Lasyley 1951; Kleene 1956; Reichardt and Varju 1959; Barlow and Levick 1965; Abeles 1991; Hubel 1995; Braitenberg, Heck, and Sultan 1997; Pulvermüller 2002). In terms of connectionist models, Elman (1990) utilised Jordan's recurrent network model (Jordan 1986) for finding a particular structure in sequentially given data. Exploiting this feature could realise functionality similar to a serial-order detection mechanism. However, since the network model is a variant of MLP-NN augmented with reciprocally connected 'memory layer' units, in practice, the network training normally involves an iterative and rather long tuning of the parameters, and thus it may quite often suffer from problems of numerical instability such as being stuck in local minima or slow convergence/divergence. Hence, this approach does not facilitate online processing.

The role of each unit g_j in Layer 2 in Figure 1 is to integrate the sub-word information obtained in time-wise from the pre-lexical layer (i.e., Layer 1). This involves the processing due to a generalised version of the so-called *mediated serial-order processing* (Pulvermüller 2002): when events $A_1, A_2, \ldots, A_N$ occur, the neuronal units responsible for the respective events $\alpha_1, \alpha_2, \ldots, \alpha_N$ are activated, and there is another neuronal unit β that is connected to all these units and becomes active, if and only if the sequence of activations from $\alpha_1, \alpha_2, \ldots, \alpha_N$ is detected.

In the proposed model, the notion of mediated serial-order processing is applied to the processing by the units in Layer 2 for detection of the serial-order of winning receptive fields in Layer 1 (i.e., maximally activated RBF units for the respective MFCC frame data).

2.5. *Input data representation for the units in Layer 2*

Whenever a maximally activated RBF unit in Layer 1 is determined for a single frame of MFCC data, rather than its activation value, the *label* of the unit will be transferred to Layer 2. (For instance, if the maximally activated unit is h_1 as in Figure 1, the numerical value '1' is transferred.) Activation of the same RBF by subsequent input frames are mapped onto a single value, e.g., 4, 4, 4 → 4. This feature of the model reduces input variability to a certain extent. Then, all the values obtained for all the subsequent frames of a whole word pattern are collected and given as an input vector to the second layer units $g_1, g_2, \ldots, g_M$ in Figure 1. Serial-order between frames, therefore, is not explicitly encoded by the model. However, serial-order is reproduced by the read-out algorithm that the word candidate units use for measuring the similarity between these vectors and their own intrinsic templates.

2.6. *Units representing word candidates*

As with the RBF units in the first layer, the role of each unit g_j $(j = 1, 2, \ldots, M)$ in the second layer is to measure the similarity between its template and the input vector. Also, word candidate units are configured automatically during the construction phase. We assume that there can be multiple word candidates for a single spoken word input. This is because of the variability in spoken word input; the abstraction in the first layer does not sufficiently reduce this variability to yield a unique matching lexical candidate at this level. Because multiple second layer units may correspond to the same word (or, interchangeably, lexical category), we hereafter call them 'word candidate units'.

The input vector $\mathbf{y}$ and template $\mathbf{u}_j$ corresponding to the jth word candidate are defined, respectively, as:

$$\begin{aligned} \mathbf{y} &= [y(1), y(2), \ldots, y(N_y)] \\ \mathbf{u}_j &= [u_j(1), u_j(2), \ldots, u_j(N_u^j)] \end{aligned} \tag{2}$$

where each element $y(p)$ $(p = 1, 2, \ldots, N_y)$ denotes the label of a maximally activated RBF unit in the first layer, given the data points of the pth MFCC frame data (i.e., last frame: N_y):

$$y(p) = \arg(\max(h_i(\mathbf{x}_p))). \tag{3}$$

For example, the numerical values in the template vector

$$\mathbf{u}_j = [1, 2, 3, 4]$$

represent the case where the RBF units with the labels '1', '2', "3', and '4' subsequently were maximally activated, as an encoding of four consecutive frames of MFCC data obtained in a

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

particular spoken word. The activation of the jth word candidate unit is defined as

$$g_j = \frac{\min(N_y, N_u^j)}{\max(N_y, N_u^j)} \delta(\mathbf{y}, \mathbf{u}_j) \tag{4}$$

where $\delta_j(\mathbf{y}, \mathbf{u}_j)$ is a function expressing the similarity between $\mathbf{y}$ and $\mathbf{u}_j$.

For performing lexical competition in the simulation study to be described later, only $T(< N_y)$ consecutive frames are taken into account for measuring the similarity, i.e.,

$$\begin{aligned} g_j(T) &= \frac{\min(T, N_u^j)}{\max(T, N_u^j)} \delta(\mathbf{y}', \mathbf{u}_j) \\ \mathbf{y}' &= [y(1), y(2), \ldots, y(T)]. \end{aligned} \tag{5}$$

(In the above, the parameter T therefore determines the proportional length of the initial segment.)

Note that, because of the way the input is encoded eliminates repetitions of identical labels, measuring the similarity is equivalent to detecting a particular serial-order of maximally activated RBFs in Layer 1. For instance, Figure 2 shows a part of the network involved in detecting the serial order represented by $\mathbf{u}_j$.

In practice, the actual reading-out computation described in the previous subsection can be essentially achieved by the one similar to calculating the 'edit-distance' between two strings (Duda, Hart, and Stork 2001). For example, if $\mathbf{y}$ is given as

$$\mathbf{y} = [1, 3, 1, 2, 2, 4, 4, 5]$$

a calculation of the edit-distance is illustrated in Figure 3. Then, in this case, $\delta_j(\mathbf{y}, \mathbf{u}_j) = 3$, and thus $g_j = 4/8 \times 3 = 1.5$.

By virtue of the edit distance procedure in Figure 3, the activation from RBF units with the labels '3' and '5', which appear in $y(2)$ and $y(8)$, respectively, can automatically be classified as irrelevant insertions, which are treated as 'noise' and thus not taken into account.

In sum, we may compare the four operations in the edit-distance procedure in Figure 3 to the neural detection of serial-order within the sub-network of Figure 2: the 'no change' operation corresponds to increase in the activation of a word-candidate unit, while the other three operations of 'deletion', 'insertion', and 'exchange' have no effect in terms of activation. For instance, the activation of h_3 (i.e., the value of $y(2) = 3$) is not taken into account, since, during the first-half round of serial-order detection, g_j waits only for the occurrence of the subsequent activation from h_1 followed by h_2. This skipping of redundant insertions in the input is equivalent to the deletion operation in the edit-distance procedure. Similarly, the activation of h_5 (i.e., $y(8) = 5$) is ignored,

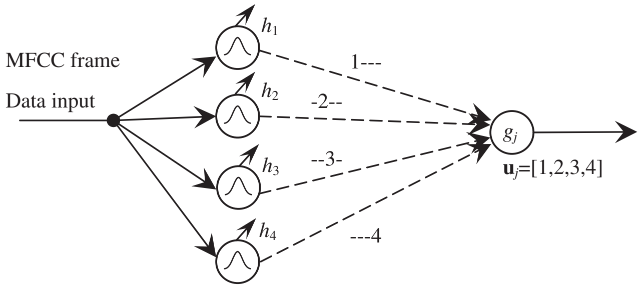

Figure 2. Example of a part of the network involved in detecting a serial-order – where the word candidate unit g_1 has the connections with the four RBFs h_1–h_4 in Layer 1 and detects the serial-order of the activations from the four RBFs.

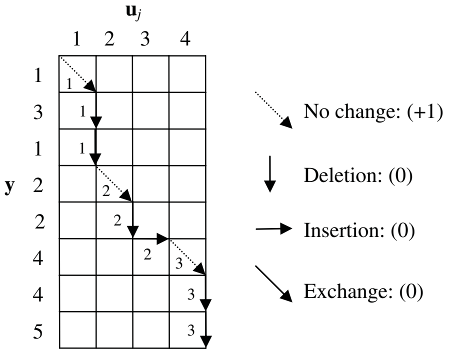

Figure 3. Illustrative example of calculating the modified edit-distance between $\mathbf{y}$ and $\mathbf{u}_j$ (cf. Duda et al. 2001); in the procedure, the four operations of 'no-change', 'deletion', 'insertion', and 'exchange' are involved. (Then, the priority of the four operations is given in this order).

simply because no connection has been established between h_8 and g_j. Repeated activation of the same RBF h_2 or h_4 is ignored. This, again, corresponds to the deletion operation in the edit-distance procedure: the pairs $y(4)$ and $y(5)$ and $y(6)$ and $y(7)$ are mapped on to the respective single values 2 and 4.

2.7. *Category grouping units*

As in Figure 1, each unit w_k ($k = 1, 2, \ldots, N$) in the third layer summates over the activations from all the word candidate units that fall in a particular lexical category and eventually outputs the final score for the corresponding lexical category. Then, the activation of the category grouping unit w_k up to the presentation of the pth ($p = 1, 2, \ldots, N_y$) frame data is defined as

$$\begin{aligned} w_k(p) &= \frac{1}{\xi}\sum_{l} g_l(p), \quad l \in [\text{all the word units for a particular lexical category}] \\ \xi &= \sum_{j=1}^{M} g_j(p). \end{aligned} \tag{6}$$

Finally, the recognition result at the pth frame data presentation is obtained by applying the 'winner-takes-all' strategy, i.e.:

$$[\text{Final Recognition Result}] = \arg(\max(w_k(p))). \tag{7}$$

2.8. *Construction of Layers 2 and 3 and the interlayer connections*

Once the construction of Layer 1 is completed, both Layers 2 and 3 are subsequently constructed on the basis of the activation patterns obtained from RBF units in Layer 1. Similar to the construction of Layer 1, the addition of both the units representing word candidates and category grouping units is performed in a growing fashion, while the interlayer connections, i.e., the connections between RBF units in Layer 1 and the word candidate units in Layer 2, and those between the word candidate units and category grouping units in Layer 3, are also established during the construction. However, in contrast to the unsupervised addition of RBF units in Layer 1, the addition of units in Layers 2 and 3, as well as establishment of interlayer connections, is performed in a supervised mode; the category label assigned to a spoken word is used as the target value. In the following, the construction of Layers 2 and 3 is summarised:

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

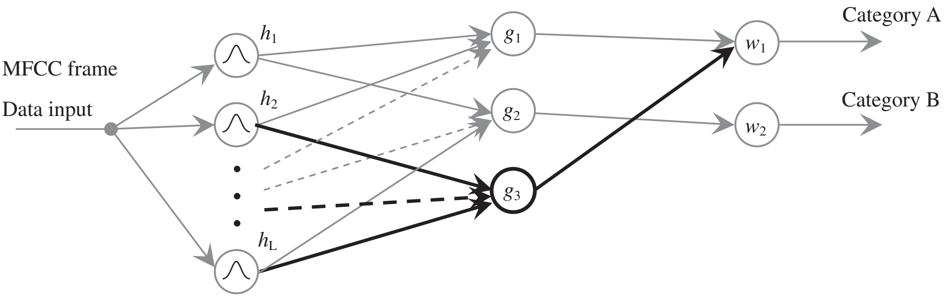

Figure 4. Construction of Layers 2 and 3 and the establishment of the interlayer connections: the resultant network structure after adding a new word candidate unit g_3 that falls into category A.

Step 1: Repeat the following for all available speech samples:

- *Step 1.1*: For all the frame data for a spoken word, Layer 2 receives a serial-order pattern of maximally activated RBFs in Layer 1.
- *Step 1.2*: If Layer 2 is empty or if the category to which a maximally activated word candidate unit in Layer 2 belongs (i.e., the category represented by the link connection with a category grouping unit in Layer 3) differs from that of the input word, do the following:
  - *Step 1.2.1*: Add a new word candidate unit into Layer 2 with setting its template vector as the serial-order pattern received in Step 1.1.
  - *Step 1.2.2*: Establish connections between the newly added word candidate unit and all RBF units in Layer 1 that appeared in the serial-order pattern.
  - *Step 1.2.3*:
    - If there is already a unit representing the same category in Layer 3, establish a connection between the category grouping unit and the newly added word candidate unit.
    - Otherwise, add a new category grouping unit in Layer 3 with the same category label as the input word candidate unit and establish the link connection in between.

    Otherwise, do nothing.

In Step 1.2.1, note that the nodes in Layer 2 intrinsically exhibit a many-to-one mapping: for a single category, there can eventually be multiple word candidate units with different serial-order templates in Layer 2 during the construction. Suppose, for instance, that there are two word candidate units g_1 and g_2 that belong, respectively, to the categories A and B (i.e., connected with the respective category grouping units w_1 and w_2 in Layer 3) with the template vectors $\boldsymbol{u}_1 = [1, 2, 3, 4]$ and $\boldsymbol{u}_2 = [1, 3, 5, 6, 7, 9]$, respectively. Then, suppose that a new word candidate unit g_3 that falls into category A is about to be added in Layer 2 with setting $\boldsymbol{u}_3 = [2, 5, 6, 7, 8]$ and establishing the link connection with w_1; the resultant network structure will be illustrated in Figure 4.

3. Simulation study and discussion

In order to test the proposed model, a simulation study was performed, using data sets consisting of spoken digit words in Japanese.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

Table 1. Spoken digit datasets used for the simulation study.

	Num. training patterns	Num. testing patterns	Num. digits used
Data set 1 (single speaker)	780	910	13
Data set 2 (seven speakers)	504	336	12 (-/SHICHI/)
Data set 3 (88 speakers)	720	336	12 (-/JU/)

3.1. *Data sets used*

For the simulation study, we used two different Japanese spoken digit data sets as summarised in Table 1.

Non-trained Japanese native speakers produced these utterances in an ordinary room. Recordings were made with a moderate level of background noise, no noise reduction was performed. Words were originally sampled at 44.1/22.05 kHz, with the frame rate/analysis window size of 16/32 ms, and converted into a training/testing pattern consisting of 12 MFCC frame data points (using the HCopy program available from HTK package (Young et al. 2005)).

Data set 1 was recorded from a single speaker, Data set 2 from seven speakers. In the recording session, each speaker was asked to utter a Japanese digit word at once, and, for each digit, this was repeated for 130 (Data set 1)/10 times (Data set 2). In contrast, Data set 3 was obtained from a total of 88 different speakers, each producing only a single utterance. The Japanese digit words recorded were: ‘ichi’ (one), ‘ni’ (two), ‘san’ (three), ‘yon’ (four), ‘go’ (five), ‘roku’ (six), ‘nana’ (seven-1), ‘shichi’ (seven-2; for Data set 1 only), ‘hachi’ (eight), ‘kyu’ (nine), ‘ju’ (ten), ‘zero’ (zero-1), ‘rei’ (zero-2) (i.e., 13 digits in total). In the simulation study, both Data sets 1 and 2 were used for speaker-dependent tasks, while Data set 3 was used for speaker-independent tasks (tasks in which the training and testing patterns were produced by different speakers; utterances obtained from 60 speakers were used for the training, and the remaining 28 for the testing).

3.2. *Overall recognition performance*

Table 2 compares the overall recognition performance of the cascaded neural network to a benchmark. For this purpose, a state-of-the-art HMM approach was used (for an introduction, e.g., see Rabiner and Juang 1993). In Table 3, both the numbers of RBF units and word candidate units generated within the cascaded neural network during the construction phase are shown.[3]

Table 2 shows that the overall recognition performance obtained by the network is comparable to that of the HMM for Data sets 1–3. The number of word candidate units can be regarded as a

Table 2. Simulation results – overall recognition performance (best scores obtained).

	Data set 1 (%)	Data set 2 (%)	Data set 3 (%)
Cascaded neural network	98.5	92.9	90.8
HMM	99.1	89.9	94.6

Table 3. Simulation results – number of sub-word/word candidate units generated during construction of the cascaded neural network model.

	Data set 1	Data set 2	Data set 3
Num. RBF units	134	3635	100
Num. word candidate units	91	206	311

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

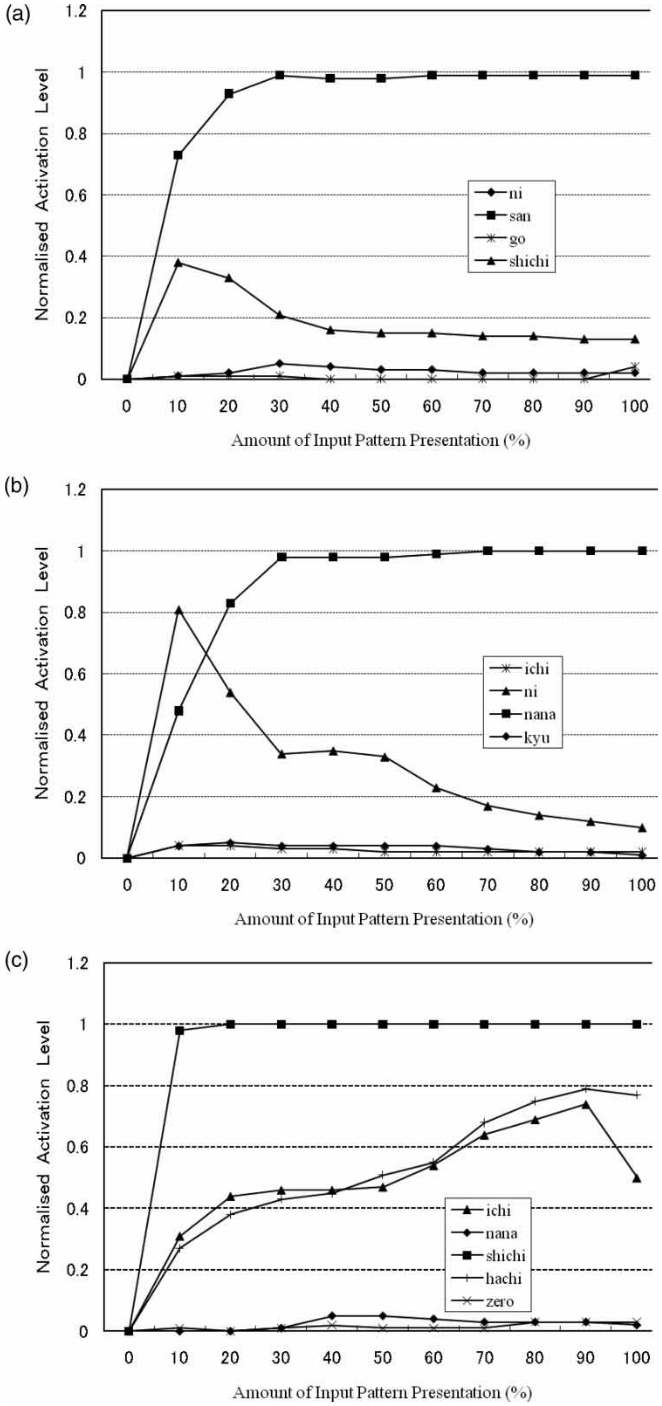

Figure 5. Results of lexical competition using cascaded neural network model – where the target digits were (a) ‘san’, (b) ‘nana’, and (c) ‘shichi’, respectively; in the figures, y-axis is the normalised activation level of the category grouping unit corresponding to the target digit, whereas x-axis is the proportional length of input patterns in terms of number of MFCC frame data.

direct reflection of the number of speakers included in the data sets: relatively small for Data set 1, larger for Data set 2, and largest for Data set 3. This illustrates that the number of these units captures the variability of utterances between different speakers.

3.3. *Simulation results on lexical competition*

To investigate how a cascaded neural network behaves during the recognition phase, we here consider the lexical competition among several candidates under presentation of the initial part of the word. The simulation strategy follows essentially the same principle as in the study using the TRACE model (Allopenna et al. 1998).

More specifically, using Data set 1, we presented each testing pattern, varying the length of the input pattern in terms of number of MFCC frame data. Then, the activation from each word candidate unit is calculated by Equation (5). Figure 5(a)–(c) show the results of lexical competition using the cascaded neural network where the target digits were ‘san’, ‘nana’, and ‘shichi’, respectively. Each plot shown in the figures was the normalised activation of the category grouping unit corresponding to the digit (i.e., calculated using Equation (6)) averaged over all the testing patterns for a particular lexical category (i.e., a digit), as a function of the proportional length of the initial segment.

In Figure 5(a), the target digit was ‘san’, and it clearly shows that ‘shichi’ can be regarded as a cohort, since the initial unvoiced part ‘sh’ is considered to be shared by ‘san’ and ‘shichi’, and that both the digits ‘go’ and ‘ni’ are unrelated to the target. Similarly, it is shown in Figure 5(b) that the initial nasally voiced part is shared by ‘nana’ and ‘ni’, at a relatively higher level of activation than that of ‘nana’, but, at around 30% of presentation amount, the activation level of ‘ni’ quickly drops. Figure 5(c) compares the target digit ‘shichi’ with the two rhymes ‘hachi’ and ‘ichi’. As expected, the activation levels for both ‘hachi’ and ‘ichi’ were increased monotonically and kept relatively high at the later amount of input presentation.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

4. Conclusion and discussion

We proposed a cascaded neuro-computational model for spoken word recognition. The model is based upon the concept of two-staged processing as suggested in human speech recognition research. Each pre-lexical unit represents a neurally plausible receptive field, modelled by an RBF. Each lexical candidate represents an abstract word template consisting of the dominant RBFs detected in the input sequence. The two-stage structure is built incrementally from data, using a combination of simple unsupervised and supervised network growing. Performance of the network functions is comparable to that of a state-of-the-art engineering model based on HMM (e.g., see Rabiner and Juang 1993).

HMM is a statistical model for a general pattern classification purpose.[4] For the application to spoken word recognition, a word is modelled by a limited (or usually small) number of states and their transition probabilities.[5] After training, each word will have its own HMM, in which each state represents a macroscopic representation of sub-word information, approximated by a mixture of probability density functions, and it is normally assumed that the transitions between the respective states occur only in a single direction in time (i.e., from early to late).

On the other hand, in our model, each RBF unit in the first layer represents a particular word segment that can be common to several word representations. The segment can repeatedly be activated during the course of a single word presentation. Then, even for a single word category, multiple word candidate units can be created in the second layer during the construction, if they differ sufficiently from each other. Therefore, the combination of multiple locally clustered data representations in the currently proposed model enables the representation of sub-word information. Thereby, it is possible to perform online local tuning of the system against variations in the training set (i.e., incremental training), which is impossible in conventional HMM approaches, due to their holistic representation and non-one-shot parameter tuning by the use of Baum-Welch and Viterbi algorithms. In other words, it is considered that the feature of online network growing as in the present approach could greatly contribute to develop a speech recognition system with adaptability and capability of dealing with context sensitivity, both the abilities of which are inherently held in human language faculty.

Unlike conventional artificial neural network approaches, the cascaded neural network is grown from input data. Growth of the network in the respective layers and their connections follows essentially the same principle across layers: adding a unit, whenever the system encounters a new pattern that cannot be covered by the receptive fields of the existing structure.

The model assumed a serial-order read-out mechanism in Layer 2 that can be applied to any superordinate–subordinate relationship of a given data structure. For instance, connectivity between multiple words in a sentence, i.e., syntactical structure could, in principle, be modelled by applying such a sequence detection mechanism (cf. Pulvermüller 2002).

The present model bears resemblance to a feed-forward architecture that has recently been applied to visual object recognition (Serre et al. 2007). This model was developed in order to account for rapid categorisation in visual system. Multi-layered feed-forward network structure may be vital for rapidly categorising sensory information. In the auditory domain, a feed-forward structure must be combined with a sequence detection mechanism, for temporal pattern recognition. The present article presents a possible strategy how this combination can be implemented.

Although the application of the proposed model in this article is limited to the recognition of a rather small number of isolated spoken words, it can still be extensible to apply for a larger lexicon and connected word/continuous speech recognition tasks. The latter can be possible, with a slight modification to the manner of computation for the activation from a word candidate unit in Equation (4), by taking into account the averaged total number of consecutive frames calculated for each word category during the training.

In the present article, only interlayer connections with binary values (i.e., 1: connected, 0: not connected) between Layers 2 and 3 were considered during the network construction phase (in Section 2.8). However, the model could benefit from introducing variable weight connections, thus enabling neural plasticity of the Hebbian type (Hebb 1949). This manner of modifying link weight connections has previously been introduced successfully to kernel memory in a pattern classification application (Hoya 2005). We expect that with this extension more robust performance will ensue on word recognition under noisy conditions.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

Acknowledgements

The authors would like to thank the three anonymous reviewers for their invaluable comments to improve the manuscript. The authors would also like to thank the students who participated in the recording session at the Department of Electronics and Communications, School of Science and Technology, Meiji University and Department of Mathematics, College of Science and Technology, Nihon University.

Notes

[1] In this article, however, without loss of generality, we consider only uniformly weighted connections between units.

[2] In another view, the node-adding manner can also be viewed as a simplified version of Gaussian ARTMAP (Williamson 1996); Gaussian ARTMAP is a variant of ARTMAP model (Carpenter et al. 1991), with incorporating

Gaussian distribution functions as category units into its adaptive resonance theory modules; a new unit (i.e., new category node) may be created when an input pattern sufficiently differs from other category nodes, depending upon the value of the vigilance parameter. However, unlike the proposed approach, both the mean (i.e., corresponding to the centroid vector here) and standard deviation variables of a certain category node (cf. Equation (1)) are updated, whenever an input pattern matches (or resonates) it.

3. The generation of RBF units depends upon the selection of σ (in Equation (1)). In the simulation, we selected the value of σ for Data set 2 much smaller than that for Data set 3, in order to achieve a reasonable performance, and thus the number of RBF units generated during the training phase increased as such.
4. Although Rao (2004) studies the neural-plausibility of HMM in depth, we do not dig into this issue here.
5. Sub-word information corresponds in an HMM to states which are defined as mixtures of probability density functions (i.e., normally represented by a multivariate Gaussian mixture). A matching score between word input and an HMM is eventually evaluated as a joint probability, obtained via the sequential product of each state output and their transition probabilities.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010

References

Abeles, M. (1991), *Corticonics – Neural Circuits of the Cerebral Cortex*, Cambridge: Cambridge University Press.

Allen, J.B. (1994), ‘How do Humans Process and Recognize Speech?’, *IEEE Transactions on Speech and Audio Processing*, 2(4), 567–577.

Allopenna, P.D., Magnuson, J.S., and Tanenhaus, M.K. (1998), ‘Tracking the Time Course of Spoken Word Recognition Using Eye Movements: Evidence for Continuous Mapping Models’, *Journal of Memory and Language*, 38, 419–439.

Barlow, H., & Levick, W.R. (1965), ‘The Mechanism of Directionally Selective Units in Rabbit’s Retina’, *Journal of Physiology*, 178, 477–504.

Braitenberg, V., Heck, D., and Sultan, F. (1997), ‘The Detection and Generation of Sequences as a Key to Cerebellar Function: Experiments and Theory’, *Behavioral and Brain Sciences*, 20, 229–245.

Carpenter, G.A., Grossberg, S., and Reynolds, J. (1991), ‘ARTMAP: Supervised Real-time Learning and Classification of Nonstationary Data by a Self-organizing Neural Network’, *Neural Networks*, 4, 565–588.

Christiansen, M.H., and Chater, N. (1999), ‘Connectionist Natural Language Processing: The State of the Art’, *Cognitive Science*, 23(4), 417–437.

Duda, H.E., Hart, P.E., and Stork, D.G. (2001), *Pattern Classification* (2nd ed.), NewYork: Wiley.

Elman, J.L. (1990), ‘Finding Structure in Time’, *Cognitive Science*, 14, 179–211.

Furui, S. (1981), ‘Cepstral Analysis Technique for Automatic Speaker Verification’, *IEEE Transactions on Acoustic Speech and Signal Processing*, 29, 254–272.

Hebb, D.O. (1949), *The Organization of Behavior: A Neuropsychological Theory*, New York: John Wiley.

Holmes, W., and Huckvale, M.A. (1994), ‘Why have HMMs been so Successful for Automatic Speech Recognition and How Might They be Improved?’, Internal Report: Speech, Hearing and Language – Work in Progress, Phonetics and Linguistics, University College London, 8.

Hopfield, J.J. (1995), ‘Pattern Recognition Computation Using Action Potential Timing for Stimulus Representation’, *Nature*, 376(6), 33–36.

Hoya, T. (2005), ‘Artificial Mind System – Kernel Memory Approach’, *Studies in Computational Intelligence*, (Vol. 1), Heidelberg: Springer-Verlag.

Hoya, T., and Washizawa, Y. (2007), ‘Simultaneous Pattern Classification and Multidomain Association Using Self-structuring Kernel Memory Networks’, *IEEE Transactions on Neural Networks*, 18(3), 732–744.

Hubel, D. (1995), *Eye, Brain, and Vision* (2nd ed.), New York: Scientific American Library.

Jordan, M.I. (1986), ‘Serial Order: A Parallel Distributed Processing Approach’, Internal Report: Tech. Rep. No. 8604, San Diego: University of California, Institute for Cognitive Science.

Kleene, S.C. (1956), ‘Representation of Events in Nerve Nets and Finite Automata’, in *Automata Studies*, eds. C.E. Shannon and J. McCarthy, Princeton, NJ: Princeton University Press, pp. 3–41.

Lasyley, K.S. (1951), ‘The Problem of Serial Order in Behavior’, in *Cerebral Mechanisms in Behavior. The Hixxon Symposium*, ed. L.A. Jeffress, New York: John Wiley, pp. 112–136.

Marslen-Wilson, W.D. (1987), ‘Functional Parallelism in Spoken Word Recognition’, *Cognition*, 25, 71–102.

McClelland, J.L., and Elman, J.L. (1986), ‘The TRACE Model of Speech Perception’, *Cognitive Psychology*, 18, 1–86.

McClelland, J.L., Mirman, D., and Holt, L.L. (2006), ‘Are There Interactive Processes in Speech Perception?’, *Trends in Cognitive Sciences*, 10(8), 363–369.

McCulloch, W.S. and Pitts, W.H. (1943), ‘A Logical Calculus of Ideas Immanent in Nervous Activity, *Bulletin of Mathematical Biophysics*, 5, 115–133.

Mirman, D., McClelland, J.L., and Holt, L.L. (2006), ‘An Interactive Hebbian Account of Lexically Guided Tuning of Speech Perception’, *Psychonomic Bulletin & Review*, 13(6), 958–965.

Morton, J. (1969), ‘Interaction of Information in Word Recognition’, *Psychological Review*, 76, 165–178.

Norris, D. (1994), ‘Shortlist: A Connectionist Model of Continuous Speech Recognition’, *Cognition*, 52, 189–234.

Norris, D., and McQueen, J.M. (2008), ‘Shortlist B: A Bayesian Model of Continuous Speech Recognition’, *Psychological Review*, 115(2), 357–395.

Platt, J. (1991), ‘A Resource-allocating Network for Function Interpolation’, *Neural Computation*, 3(2), 213–225.

Plaut, D.C., and Kello, C.T. (1999), 'The Emergence of Phonology from the Interplay of Speech Comprehension and Production: A Distributed Connectionist Approach', in *The Emergence of Language*, ed. B. MacWhinney, Mahwah, NJ: Erlbaum, pp. 381–415.

Pulvermüller, F. (2002), *The Neuroscience of Language: On Brain Circuits of Words and Serial Order*, New York: Cambridge University Press.

Rabiner, L., and Juang, B.-H. (1993), *Fundamentals of Speech Recognition*, Englewood Cliffs, NJ: Prentice Hall.

Rao, P.N.R. (2004), 'Bayesian Computation in Recurrent Neural Circuits', *Neural Computation*, 16, 1–38.

Reichardt, W., and Varju, D. (1959), 'Übertragungseigenschaften im Auswertesystem für das Bewegungssehen', *Zeitschrift für Naturforschung*, 14b, 674–689.

Serre, T., Oliva, A., and Poggio, T. (2007), 'A Feedforward Architecture Accounts for Rapid Categorization', *Proceedings of the National Academy of Sciences*, 104(15), 6424–6429.

Scharenborg, O., Norris, D., ten Bosch, L., and McQueen, J.M. (2005), 'How Should a Speech Recogniser Work?', *Cognitive Science*, 29, 867–918.

Scharenborg, O. (2007), 'Reading Over the Gap: A Review of Efforts to Link Human and Automatic Speech Recognition Research', *Speech Communication*, 49, 336–347.

Williamson, J.R. (1996), 'Gaussian ARTMAP: A Neural Network for Fast Incremental Learning of Noisy Multidimensional Maps', *Neural Networks*, 9(5), 881–897.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. (2005), *The HTK Book (Version 3.3)*, Department of Engineering, Cambridge University.

Downloaded By: [Hoya, Tetsuya] At: 08:33 4 February 2010