(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
31 March 2011 (31.03.2011)

PCT

(10) International Publication Number
WO 2011/037587 A1

(51) International Patent Classification:
G10L 15/02 (2006.01)

(21) International Application Number:
PCT/US2009/058563

(22) International Filing Date:
28 September 2009 (28.09.2009)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: **NUANCE COMMUNICATIONS, INC.** [US/US]; 1 Wayside Road, Suite 100, Burlington, MA 01803-4613 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **GRUHN, Rainer** [DE/DE]; Riedlenstrasse 85, 89079 Ulm (DE). **CANO, Daniel Andres, Vazquez** [CO/DE]; Basteistr. 29, 89073 Ulm (DE). **ARADILLA, Guillermo** [ES/DE]; Donautalstrasse 20, 89079 Ulm (DE).

(74) Agents: **JAKOBSCHE, George, J.** et al.; Sunstein Kann Murphy & Timbers LLP, 125 Summer Street, Boston, MA 02110 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,

[Continued on next page]

(54) Title: DOWNSAMPLING SCHEMES IN A HIERARCHICAL NEURAL NETWORK STRUCTURE FOR PHONEME RECOGNITION

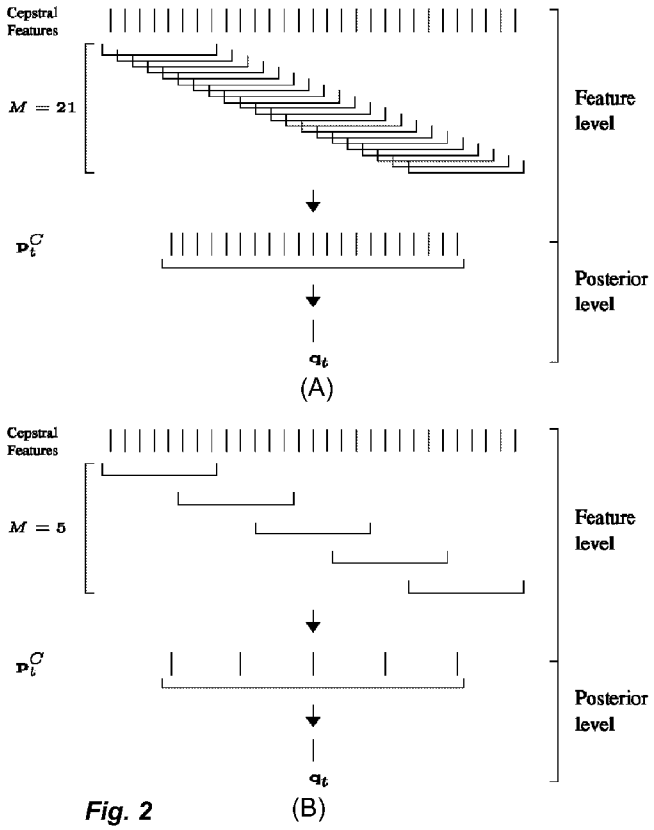


Fig. 2

(57) Abstract: An approach for phoneme recognition is described. A sequence of intermediate output posterior vectors is generated from an input sequence of cepstral features using a first layer perceptron. The intermediate output posterior vectors are then downsampled to form a reduced input set of intermediate posterior vectors for a second layer perceptron. A sequence of final posterior vectors is generated from the reduced input set of intermediate posterior vectors using the second layer perceptron. Then the final posterior vectors are decoded to determine an output recognized phoneme sequence representative of the input sequence of cepstral features.

WO 2011/037587 A1

MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

TITLE

Downsampling Schemes In A Hierarchical Neural Network Structure For Phoneme Recognition

FIELD OF THE INVENTION

[0001] The present invention relates to automatic speech recognition, and more specifically phoneme recognition in a hierarchical neural network.

BACKGROUND ART

[0002] A phoneme is the minimal unit of speech sound in a language that can serve to distinguish meaning. Phoneme recognition can be applied to improve automatic speech recognition. Other applications of phoneme recognition can also be found in speaker recognition, language identification, and keyword spotting. Thus, phoneme recognition has received much attention in the field of automatic speech recognition.

[0003] One common and successful approach for phoneme recognition uses a hierarchical neural network structure based on a hybrid hidden Markov model (HMM)-Multilayered Perceptron (MLP) arrangement. The MLP outputs are used as HMM state emission probabilities in a Viterbi decoder. This approach has the considerable advantage that the MLP can be trained to discriminatively classify phonemes. The MLP also can easily incorporate a long temporal context without making explicit assumptions. This property is particularly important for phoneme recognition because phoneme characteristics can be spread over a large temporal context.

[0004] Many different approaches have been proposed to continue to exploit the contextual information of a phoneme. One approach is based on a combination of different specialized classifiers that provides considerable improvements over simple generic classifiers. For instance, in the approach known as TRAPS, long temporal information is divided into frequency bands, and then, several classifiers are independently trained using specific frequency information over a long temporal range. *See* H. Hermansky and S. Sharma, *Temporal Patterns (TRAPS) in ASR of Noisy Speech,* in Proc. ICASSP, 1999, vol. 1, pp. 289–292, incorporated herein by reference. Another different technique splits a long

temporal context in time. *See* D. Vasquez et al., *On Expanding Context By Temporal Decomposition For Improving Phoneme Recognition*, in SPECOM, 2009, incorporated herein by reference. A combination of these two approaches which splits the context in time and frequency is evaluated in P. Schwarz et al., *Hierarchical Structures Of Neural Networks For Phoneme Recognition*, in Proc. ICASSP, 2006, pp. 325–328, incorporated herein by reference.

[0005] Another phoneme recognition structure was proposed in J. Pinto et al., *Exploiting Contextual Information For Improved Phoneme Recognition,* in Proc. ICASSP, 2008, pp. 4449–4452, (hereinafter "Pinto", incorporated herein by reference). Pinto suggested estimating phoneme posteriors using a two-layer hierarchical structure. A first MLP estimates intermediate phoneme posteriors based on a temporal window of cepstral features, and then a second MLP estimates final phoneme posteriors based on a temporal window of intermediate posterior features. The final phoneme posteriors are then input to a phonetic decoder for obtaining a final recognized phoneme sequence.

[0006] The hierarchical approach described by Pinto significantly increases system accuracy, compared to a non-hierarchical scheme (a single layer). But computational time is greatly increased because the second MLP has to process the same number of speech frames as were processed by the first MLP. In addition, the second MLP has an input window with a large number of consecutive frames, so there are a high number of parameters that must be processed. These factors make it less practical to implement such a hierarchical approach in a real time application or in an embedded system.

SUMMARY OF THE INVENTION

[0007] Embodiments of the present invention are directed to phoneme recognition. A sequence of intermediate output posterior vectors is generated from an input sequence of cepstral features using a first layer perceptron. The intermediate output posterior vectors are then downsampled to form a reduced input set of intermediate posterior vectors for a second layer perceptron. A sequence of final posterior vectors is generated from the reduced input set of intermediate posterior vectors using the second layer perceptron. Then the final posterior vectors are decoded to determine an output recognized phoneme

sequence representative of the input sequence of cepstral features.

[0008] In further embodiments, intra-phonetic information and/or inter-phonetic information may be used for decoding the final posterior vectors. The downsampling may be based on a window downsampling arrangement or a temporal downsampling arrangement, for example, using uniform downsampling or non-uniform downsampling. The downsampling may be based on using an intermediate phoneme decoder to determine possible phoneme boundaries. And the perceptrons may be arrangements within a Hybrid Hidden Markov Model-Multilayer Perceptron (HMM-MLP) phoneme classifier.

[0009] Embodiments also include an application or device adapted to perform the method according to any of the above. For example, the application may be a real time application and/or an embedded application.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] Figure 1 shows a two-layer hierarchical structure for estimating posterior vectors for phoneme recognition.

[0011] Figure 2 compares the standard hierarchical approach for handling intermediate posterior vectors with a window downsampling approach.

[0012] Figure 3 A-B shows sequences of posterior vectors in a temporal context C which stretches over a period of frames.

[0013] Figure 4 compares the standard hierarchical approach for handling intermediate posterior vectors with a uniform downsampling approach.

[0014] Figure 5 illustrates a non-uniform downsampling approach.

DETAILED DESCRIPTION OF SPECIFIC EMBODIMENTS

[0015] In phoneme recognition, the goal is to find the most probable corresponding phoneme or phoneme-state sequence $\mathbf{S} = \{s_1, s_2, ..., s_L\}$ given a sequence of observations

or cepstral vectors $\mathbf{X}_o = \{\mathbf{x}_1, ..., \mathbf{x}_t, ..., \mathbf{x}_K\}$, where $K \geq L$. In the following discussion, the phoneme recognition approach is discussed based on a Hybrid HMM/MLP where the MLP outputs are used as HMM state emission probabilities. An MLP has the additional advantage of incorporating a long context in the form of a temporal window of N cepstral features. Therefore, the set of acoustic vectors at the input of the MLP is $\mathbf{X} = \{\mathbf{x}_1^N, ..., \mathbf{x}_t^N, \mathbf{x}_K^N\}$ where:

$$\mathbf{x}_t^N = \{\mathbf{x}_{t-\frac{N-1}{2}}, ..., \mathbf{x}_t, ..., \mathbf{x}_{t+\frac{N-1}{2}}\}$$

[0016] The emission probabilities in an HMM state can be estimated by scaled likelihoods according to the Bayes' rule:

$$\frac{p(\mathbf{x}_t^N \mid s_t = i)}{p(\mathbf{x}_t^N)} = \frac{p(s_t = i \mid \mathbf{x}_t^N)}{P(s_t = i)} = \frac{p_t^i}{P(s_t = i)}$$

where $p_t^i = p(s_t = i \mid \mathbf{x}_t^N)$ refers to the output of the MLP. The prior state probability $P(s_t =$ i) can be estimated after performing a forced alignment of the training data to its true labels. Since $p(\mathbf{x}_t^N)$ is independent of the HMM state, it can be ignored for deriving scaled likelihoods. The scaled likelihoods are applied to a Viterbi decoder for obtaining a recognized phoneme sequence.

[0017] At each time instant *t,* the MLP estimates a posterior probability for each phoneme or phoneme-state. Thus the output of the MLP is a posterior feature vector:

$$\mathbf{p}_t = [p_t^1, ..., p_t^i, ..., p_t^O]^T$$

with O being the total number of phonemes or phoneme-states. In this discussion, we refer to 1-state or 3-state modeling when the number of output units equals the total number of phonemes or phoneme-states respectively.

Hierarchical Downsampling

[0018] Embodiments of the present invention are directed to an improved neural network for phoneme recognition based on a two-layer hierarchical structure based on a multi-layer perceptron (MLP). Intermediate posterior vectors generated by a first layer MLP (e.g., cepstral features level) are input to a second layer MLP (e.g., intermediate

posteriors level). Figure 1 shows a two-layer hierarchical structure for estimating posterior vectors according to embodiments of the present invention. In the first layer which is defined as context modeling at the *feature level,* an MLP estimates intermediate phoneme posterior vectors p_t from a context of N cepstral features x_t^N. Redundant information contained in the intermediate posterior vectors is removed by one or more downsampling arrangements, and the second layer MLP, which provides context modeling at the *posterior level,* based on a window of C intermediate posterior vectors:

$$p_t^C = \{p_{t-\frac{C-1}{2}}, \ldots, p_t, \ldots, p_{t+\frac{C-1}{2}}\}$$

to estimate final phoneme posterior vectors $q_t^i = p(s_t = i \mid \mathbf{p}_t^C)$ given phoneme posterior trajectories which constitute to some extended language model properties.

[0019] For example, in one approach referred to as window downsampling, redundant information is removed at the input window of the second layer MLP. That allows the number of parameters to be greatly reduced and reduces computational time while maintaining system accuracy requirements. Another approach referred to as temporal downsampling reduces the number of frames at the input of the second layer MLP by downsampling the intermediate posteriors based on a intermediate phonetic decoder. Such downsampling arrangements make it feasible to implement a hierarchical scheme in a real time application or in a embedded system.

Inter-Phonetic Information

[0020] Inter-phonetic information can be modeled by the second layer of the hierarchical MLP, and a window downsampling technique can then be used to make feasible a real time implementation. As explained above, a two-layer MLP hierarchical structure can be used that estimates the posterior vectors. The first layer MLP provides context modeling at the feature level by estimating intermediate phoneme posterior vectors $\mathbf{p}_t$ from a context of N cepstral features x_t^N. The second layer MLP provides context modeling at the posterior level by estimating phoneme posterior vectors $\mathbf{q}_t$ based on a window of M posterior features $\mathbf{p}_t^C$ covering a context of C frames. In the hierarchical approach, $M = C$ since all posterior vectors M contained in a context window of C frames were taken.

[0021] Figure 2A shows the standard hierarchical approach where a temporal window of M consecutive intermediate posterior vectors are input to a second layer MLP without downsampling. At the feature level, M different windows of cepstral features, shifted from each other by one frame, are used to generate M consecutive intermediate posterior vectors:

$$p_t^C = \{p_{t-\frac{C-1}{2}}, \ldots, p_t, \ldots, p_{t+\frac{C-1}{2}}\}$$

covering a context of C frames. Then, at the posterior level, the window of intermediate posterior vectors $\mathbf{p}_t^C$ is input to a second layer MLP to estimate a final posterior vector $\mathbf{q}_t$.

[0022] In practice, the sequence of intermediate posterior vectors $\mathbf{p}_t^C$ is generated by a single MLP which is shifted over time. The M consecutive posterior vectors can be viewed as if they were derived by M MLPs situated at M consecutive time instances. In this case, each MLP has been trained based on its corresponding label l_{t+j} for $-\frac{C-1}{2} \le j \le \frac{C-1}{2}$. The M generated posterior vectors may have inter-phonetic information among them since the labels, which cover a context of C frames, may belong to different phonemes. Then, the final posterior vectors $q_t^i = p(s_t = i \mid \mathbf{p}_t^C)$ are estimated based on some inter-phonetic posterior trajectories which correspond to some extended language model properties.

[0023] Context modeling at the posterior level as described in the foregoing is based on inter-phonetic information, which is useful for improving recognition accuracy since it captures how a phoneme is bounded by other phonemes. The number of posterior vectors M at the input of the second layer MLP is given by:

$$M = \frac{C-1}{T_s} + 1$$

where C is the total context of frames and T_s is the frame sampling period. As an example, Figure 3A shows a context of $C = 31$ frames with a period of $T_s = 5$, giving a number of $M = 7$ intermediate posterior vectors at the input of the second layer MLP. On the other hand, when $T_s = 1$, then $M = C$ and the hierarchical approach given in Pinto is obtained.

[0024] In Figure 3 A-B, sequences of posterior vectors $\mathbf{p}_t^C$ are given showing only those relevant components with highest posterior values within a temporal context *C*. It can be observed that a particular component p_t^i dominates during certain sub-intervals of the whole context *C* since a phoneme stretches over several frames. Therefore, there is similar information contained in each sub-interval. Having all this repeated information may be irrelevant for the task of phoneme classification based on inter-phonetic information.

Testing Framework

[0025] In one specific set of testing experiments, the TIMIT corpus was used without the SA dialect sentences, dividing the database into three parts. The training data set contained 3346 utterances from 419 speakers, the cross-validation data set contained 350 utterances from 44 speakers, and the standard test data set contained 1344 utterances from 168 speakers. A 39 phoneme set (from Lee and Hon) was used with the difference that closures were merged to the regarding burst. Feature vectors were of 39-dimensions with 13 PLPs with delta and double delta coefficients. The feature vectors were under global mean and variance normalization, with each feature vector extracted from a 25 msec speech window with a 10 msec shift.

[0026] The MLPs were trained with the Quicknet software tool to implement three layer perceptrons with 1000 hidden units. The number of output units corresponded to 39 and 117 for 1-state and 3-state modeling respectively with the softmax nonlinearity function at the output. A standard back-propagation algorithm with cross-entropy error criteria was used for training the neural network, where the learning rate reduction and stop training criteria were controlled by the frame error rate in cross-validation to avoid overtraining. In addition, a phoneme insertion penalty was set to give maximum phoneme accuracy in the cross-validation. A Viterbi decoder was implemented with a minimum duration of three states per phoneme. It also was assumed that all phonemes and states were equally distributed. No language model was used and silences were discarded for evaluation. Finally, phoneme accuracy (PA) was used in all experiments as a measure of performance.

[0027] Table 1 shows the results of a set of experiments for a standard HMM/MLP

system representing context modeling at the feature level for 1-state and 3-state modeling with a window of $N = 9$ consecutive cepstral vectors.

System	PA	
	1-state	3-state
no hierarchy	68.13	71.21
hierarchy	71.79	73.61

Table 1-Phoneme recognition accuracies for different context modeling levels. The MLPs estimate phoneme posteriors or state posteriors for 1-state or 3-state modeling respectively.

In addition, results of the hierarchical approach are also given for $M = 21$ in the second row of Table 1 where it can be observed that a remarkable increase in performance has been achieved by the hierarchical approach, as in Pinto.

Window Downsampling

[0028] Figure 2B shows how a window downsampling approach can be used to select fewer posterior vectors which are separated by some given number of frames T_s. In this case, the number of MLPs M has been highly reduced according to the equation in the preceding paragraph, where each MLP is separated a number of samples T_s. This reduces the number of inputs to the second layer MLP, thereby decreasing thus the number of parameters while still preserving acceptable performance levels. However, the posterior vectors generated by the hierarchical downsampling approach are still covering the same temporal context C.

[0029] Several experiments were performed to test this approach in which the frame sample period T_s was varied while keeping almost constant the temporal context C. Table 2 gives the results for 1-state and 3-state modeling showing phoneme recognition accuracies:

Sampling Parameters			PA	
C	M	T_s	1-state	3-state
21	21	1	71.79	73.61
19	7	3	71.62	74.10
21	5	5	71.61	74.20
21	3	10	69.39	72.80

Table 2- Phoneme recognition accuracies for the hierarchical downsampling approach. Several sampling rates have been tested giving M intermediate posterior vectors at the input of the second layer MLP covering a temporal context of C frames.

As expected, the performance significantly decreases when T_s increases up to 10 frames. On the other hand, it is interesting to observe that the performance remains almost constant for $T_s = 3$ and $T_s = 5$. In particular, for $T_s = 5$, good performance was obtained, but the number of input posterior vectors was greatly reduced from 21 to 5. These results confirm that there is redundant information at the input of the second layer MLP which can be ignored.

Intra-Phonetic Information

[0030] The foregoing shows how system performance can be increased when inter-phonetic information is used. In addition, the transition information within a particular phoneme can be modeled better to reflect the fact that a phoneme behaves differently at its beginning, middle, and end. The MLPs that have been trained as described above can also be used to obtain this intra-phonetic information. The inter-phonetic information in different MLPs trained according to Fig. 2B was obtained with $M = 5$, with the same label l_t, which belongs to the label of the middle of the context C. Thus, there is an implicit assumption that a phoneme occupies a total context of C frames.

[0031] Table 3 gives the results when $M = 5$ different MLPs were used for modeling inter and intra-phonetic information, as shown in Figure 2B:

System	PA	
	1-state	3-state
no hierarchy	68.13	71.21
hierarchy inter	71.61	74.20
hierarchy intra	70.91	73.65

Table 3- Phoneme recognition accuracies in a hierarchical framework under inter or intra phonetic constraints. The number of intermediate posterior vectors at the input of the second layer MLP is M=5, covering a temporal context of C=21 frames.

Indeed, the *inter-phonetic* arrangement is the same as shown in Table 2 for $T_s = 5$. According to Table 3, it can be observed that the inter-phonetic approach out performs the intra-phonetic approach, meaning that inter- information is more useful to better classify a

phoneme. However, both approaches considerably out perform the non-hierarchical approach. And while these approaches are based on different criteria, they can serve to complement each other and further improve performance.

Combining Inter- and Intra-Phonetic Information

[0032] The foregoing discussion describes how considering inter- and intra-phonetic information can be used to considerably improve phoneme recognition performance over a non-hierarchical system. Since these two approaches were generated based on different criteria, we now consider the complementariness of both approaches in order to achieve further improvements. To that end, the intra-phonetic can be used as a first hierarchical step, then the posterior vectors generated used as the input to a second hierarchical step, given by the inter-phonetic approach. The aim such an arrangement is first to better classify a phoneme based on the temporal transition information within the phoneme. Then, temporal transition information among different phonemes is utilized to continue improving phoneme classification.

[0033] In one set of experiments, both inter-phonetic derivations at the second hierarchical step (i.e., the conventional hierarchical approach given in Pinto) and the window downsampling approach above. Table 4 shows results of the combination together with the downsampling parameters utilized in the inter-phonetic step.

System	C	M	T_s	PA	
				1-state	3-state
no hierarchy	-	-	-	68.13	71.21
hierarchy intra	-	-	-	70.91	73.65
hierarchy intra+inter	21	21	1	73.04	74.93
hierarchy intra+inter	21	5	5	72.50	75.07

Table 4-Phoneme recognition accuracies for a combination of inter- and intra-phonetic information.

The introduction of intra-phonetic information as an intermediate step in a hierarchical approach achieves further improvements. These results verify the assumption that both criteria carry complementary information, useful for improving system performance. On the other hand, results concerning the downsampling technique, using different

sampling periods T_s, show the advantage of removing redundant information while keeping good performance.

Temporal Downsampling.

[0034] Another approach for removing redundant information from the intermediate posterior vectors is based on temporal downsampling. Figure 4A shows the intermediate posterior vectors for each frame sampling period T_t according to the standard hierarchical approach as in Pinto. The MLP receives a set of intermediate posterior vectors which are sampled based on a impulse train which denotes how often the sampling is performed. This can be viewed as a filtering process where the intermediate posterior vectors that are not sampled are ignored by the MLP. Each intermediate posterior vector has a corresponding final posterior vector since the sampling period is $T_t = 1$ frame. In Fig. 4A, each intermediate posterior vector corresponds to a window of C consecutive posterior vectors.

[0035] Figure 4B shows a uniform downsampling scheme when $T_t = 3$. Under this approach, it can be observed that a significant number of frames are reduced when T_t is considerably increased. It can be shown that having just a few number of frames of final posterior vectors decreases system accuracy under constrain of minimum phoneme duration of 3-states in the Viterbi decoder. This suggests further testing this approach with minimum phoneme duration of 1-state. It is also important to mention that during training, the true labels together with the training set of intermediate posterior vectors are also downsampled, significantly reducing the training time.

Non-Uniform Downsampling

[0036] The uniform downsampling arrangement described above has the disadvantage that important information can be lost when the sampling period is highly increased. In particular, the intermediate posterior vectors corresponding to short phonemes can be totally ignored after performing the downsampling. For this reason, it may be useful to sample the set of intermediate posterior vectors every time potentially important information appears. The sampling points can be estimated by an intermediate Viterbi decoder which takes at its input the set of intermediate posteriors. Then, the sampling

points correspond to those points in time where different phonemes have been recognized, generating a non-uniform downsampling arrangement. By this means, the loss of possible important information is highly alleviated, while significantly reducing the number of frames at the input of the second layer MLP.

[0037] Figure 5 illustrates the idea of a non-uniform downsampling arrangement. A set of intermediate posterior vectors generated by the first layer MLP is input to an intermediate Viterbi decoder, which gives at its output an intermediate recognized phoneme sequence together with the phoneme boundaries. Then the recognized phoneme sequence not used further, but just the time boundaries. Each segment corresponding to a recognized phoneme is uniformly divided into three sub-segments. Then, the sampling points are indicated by the central frame of all sub-segments.

[0038] Three sampling points per recognized phoneme were used in order to keep consistency with a phoneme model of minimum 3-state duration. In addition, no word insertion penalty was used in the intermediate Viterbi decoder. Moreover, as it is performed for the uniform downsampling scheme, the training set of intermediate posteriors together with the true labels were also downsampled based on the non-uniform sampling points. For training the second layer MLP, a window of C consecutive posterior vectors was used.

[0039] In one set of experiments testing these ideas, we tested various specific MLP arrangements at the posterior level, corresponding to different hierarchical approaches: no downsampling (standard), uniform downsampling and non-uniform downsampling. All the tested MLPs, had the same number of parameters, and therefore, the computational time reduction was only due to a decrease in number of frames at the input of the second layer MLP. As in other experiments, the input of the second layer MLP for all approaches consisted of a window of $C = 21$ intermediate posterior vectors.

[0040] For 1-state modeling, the second MLP gave an average speed of 359.63 MCPS (Million Connections Per Second) for forward propagation phase. This MLP was able to process 418.6 frames/sec. On the other hand, the MLP for 3-state modeling

gave an average speed of 340.87 MCPS which processed 132.4 frames/sec. Table 5 shows the average number of frames per utterance at the input of the second layer MLP:

Downsampling	T_t	frames/utt.		time/utt. [sec]	
		1-state	3-state	1-state	3-state
uniform	1	304	304	0.72	2.29
uniform	3	100	100	0.24	0.75
uniform	5	60	60	0.14	0.45
non-uniform	-	123	101	0.29	0.76

Table 5—Performance of downsampling arrangements measured in average frames per utterance and computational time.

Table 6 shows phoneme accuracies of the different downsampling methods:

Downsampling	T_t	min. dur. 1-state		min. dur. 3-state	
		1-state	3-state	1-state	3-state
uniform	1	71.86	70.36	71.79	73.61
uniform	3	71.21	67.18	58.80	55.93
uniform	5	68.15	67.17	39.92	37.27
non-uniform	-	72.04	70.69	71.86	73.39

Table 6—Phoneme accuracies of downsampling arrangements. The second layer MLP estimates phoneme posteriors (1-state) or state posteriors (3-state). Minimum duration constraints of 1-state and 3-state per phoneme are evaluated in the final Viterbi decoder.

[0041] For uniform downsampling, three different sampling periods T_t were tested. And for $T_t = 1$, the standard hierarchical approach was obtained as it is given in Pinto. It can be seen in Table 5 that the computational time was considerably reduced when T_t was increased, since there were fewer frames to process by the second layer MLP. But, the system accuracy dropped significantly. For this approach, it can be observed that it is necessary to use a minimum duration of 1-state in the final decoder in order to keep a reasonably good accuracy.

[0042] One reason of the low accuracy obtained when T_t was increased was caused by the poor classification of short phonemes. When the sampling period is extremely increased, only a few samples corresponding to short phonemes may remain, or they even may totally disappear. This effect was verified by measuring the phoneme recognition of short phonemes when $T_t = 5$. The shortest phonemes of the TIMIT corpus are given by */dx/* and */dh/* with an average number of frames of 2.9 and 3.5, respectively. And in fact,

those phonemes suffered the highest deterioration after downsampling with a decrease in phoneme recognition from 69% to 50% and from 68% to 59%, for */dx/* and */dh/* respectively.

[0043] For the non-uniform downsampling arrangement, Table 5 shows that a high reduction in the number of frames is achieved—by 59.5% and 66.8% for 1-state and 3-state modeling, respectively—which reduced computational time in the same proportion. Moreover, the computational time required by the intermediate decoder for processing one utterance was neglected compared to the time required by the second layer MLP in processing the same utterance. Table 6 shows that a similar accuracy is obtained, compared to the standard approach. This shows a significant advantage of non-uniform downsampling where a high decrease of computational time is obtained while keeping good performance.

[0044] These results confirm that there is a large amount of redundant information contained in the intermediate posterior vectors, and therefore, it is not necessary for the second layer MLP to again process each individual frame of the entire utterance. Various strategies have been described for downsampling the intermediate posterior vectors. One especially promising approach was obtained by non-uniform downsampling where the sampling frequency is estimated by an intermediate Viterbi decoder. This reduced the computational time by 67% while maintaining system accuracy comparable to the standard hierarchical approach. These results suggest the viability of implementing a hierarchical structure in a real-time application.

[0045] Embodiments of the invention may be implemented in any conventional computer programming language. For example, some or all of an embodiment may be implemented in a procedural programming language (*e.g.*, "C") or an object oriented programming language (*e.g.*, "C++", Python). Alternative embodiments of the invention may be implemented as pre-programmed hardware elements, other related components, or as a combination of hardware and software components.

[0046] Embodiments can be implemented in whole or in part as a computer program

product for use with a computer system. Such implementation may include a series of computer instructions fixed either on a tangible medium, such as a computer readable medium (*e.g.,* a diskette, CD-ROM, ROM, or fixed disk) or transmittable to a computer system, via a modem or other interface device, such as a communications adapter connected to a network over a medium. The medium may be either a tangible medium (*e.g.,* optical or analog communications lines) or a medium implemented with wireless techniques (*e.g.,* microwave, infrared or other transmission techniques). The series of computer instructions embodies all or part of the functionality previously described herein with respect to the system. Those skilled in the art should appreciate that such computer instructions can be written in a number of programming languages for use with many computer architectures or operating systems. Furthermore, such instructions may be stored in any memory device, such as semiconductor, magnetic, optical or other memory devices, and may be transmitted using any communications technology, such as optical, infrared, microwave, or other transmission technologies. It is expected that such a computer program product may be distributed as a removable medium with accompanying printed or electronic documentation (*e.g.,* shrink wrapped software), preloaded with a computer system (*e.g.,* on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the network (*e.g.,* the Internet or World Wide Web). Of course, some embodiments of the invention may be implemented as a combination of both software (*e.g.*, a computer program product) and hardware. Still other embodiments of the invention are implemented as entirely hardware, or entirely software (*e.g.*, a computer program product).

[0047] Although various exemplary embodiments of the invention have been disclosed, it should be apparent to those skilled in the art that various changes and modifications can be made which will achieve some of the advantages of the invention without departing from the true scope of the invention.

CLAIMS

What is claimed is:

1. A method for phoneme recognition comprising:

generating a sequence of intermediate output posterior vectors from an input sequence of cepstral features using a first layer perceptron;

downsampling the intermediate output posterior vectors to form a reduced input set of intermediate posterior vectors for a second layer perceptron.

2. A method according to claim 1, further comprising:

generating a sequence of final posterior vectors from the reduced input set of intermediate posterior vectors using the second layer perceptron.

3. A method according to claim 2, further comprising:

decoding the final posterior vectors to determine an output recognized phoneme sequence representative of the input sequence of cepstral features.

4. A method according to claim 3, wherein decoding the final posterior vectors uses intra-phonetic information.

5. A method according to claim 3, wherein decoding the final posterior vectors uses inter-phonetic information.

6. A method according to claim 1, wherein the downsampling is based on a window downsampling arrangement.

7. A method according to claim 1, wherein the downsampling is based on a temporal downsampling arrangement.

8. A method according to claim 7, wherein the downsampling is based on a uniform downsampling arrangement.

9. A method according to claim 7, wherein the downsampling is based on a non-uniform

downsampling arrangement.

10. A method according to claim 9, wherein the downsampling is based on using an intermediate phoneme decoder to determine possible phoneme boundaries.

11. A method according to claim 1, wherein the perceptrons are arrangements within a Hybrid Hidden Markov Model-Multilayer Perceptron (HMM-MLP) phoneme classifier.

12. An application adapted to perform the method according to any of claims 1-11.

13. An application according to claim 12, wherein the application is a real time application.

14. An application according to claim 12, wherein the application is an embedded application.

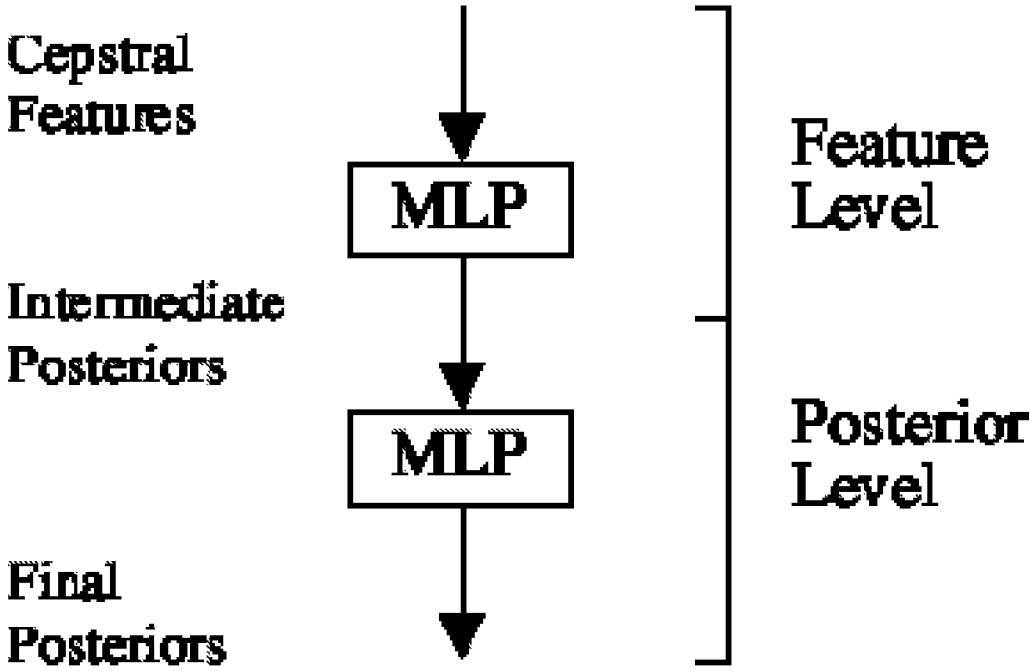

Cepstral Features
MLP
Intermediate Posteriors
MLP
Final Posteriors
Feature Level
Posterior Level

Fig. 1

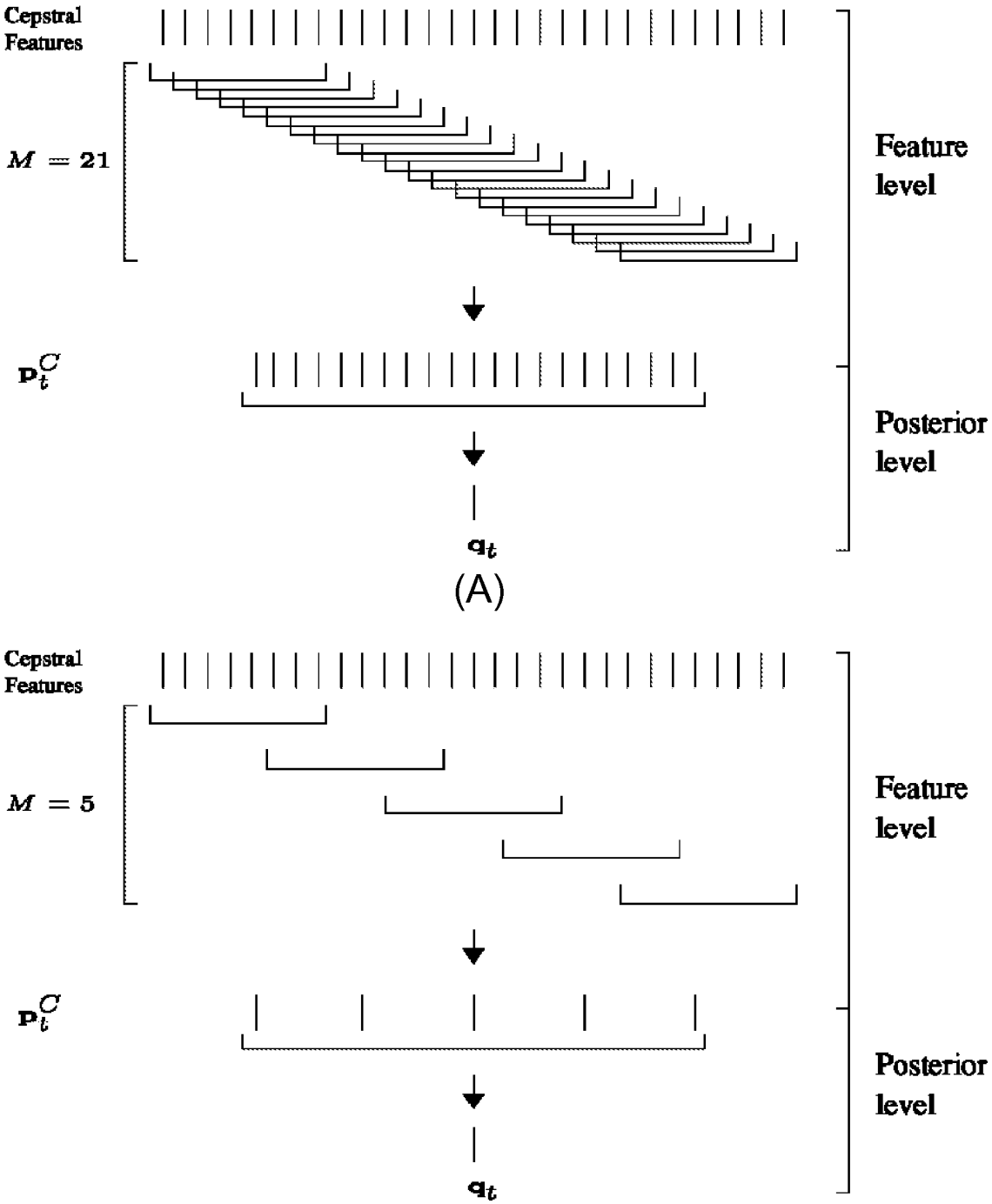
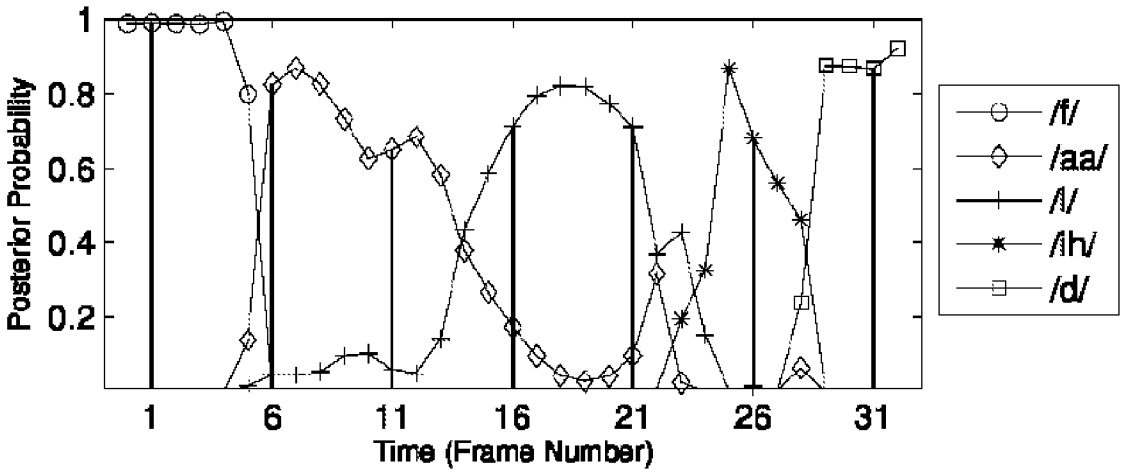
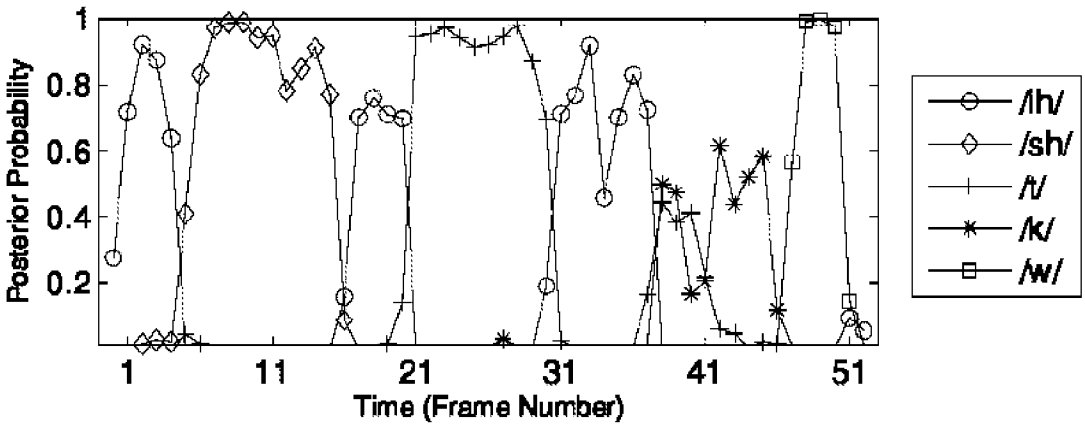

Cepstral Features
$M = 21$
Feature level
$\mathbf{p}_t^C$
Posterior level
$\mathbf{q}_t$
(A)
Cepstral Features
$M = 5$
Feature level
$\mathbf{p}_t^C$
Posterior level
$\mathbf{q}_t$
(B)

Fig. 2


Posterior Probability
1
0.8
0.6
0.4
0.2
1
6
11
16
21
26
31
Time (Frame Number)
/f/
/aa/
/l/
/ih/
/d/

(A)


Posterior Probability
1
0.8
0.6
0.4
0.2
1
11
21
31
41
51
Time (Frame Number)
/lh/
/sh/
/t/
/k/
/w/

(B)

Fig. 3

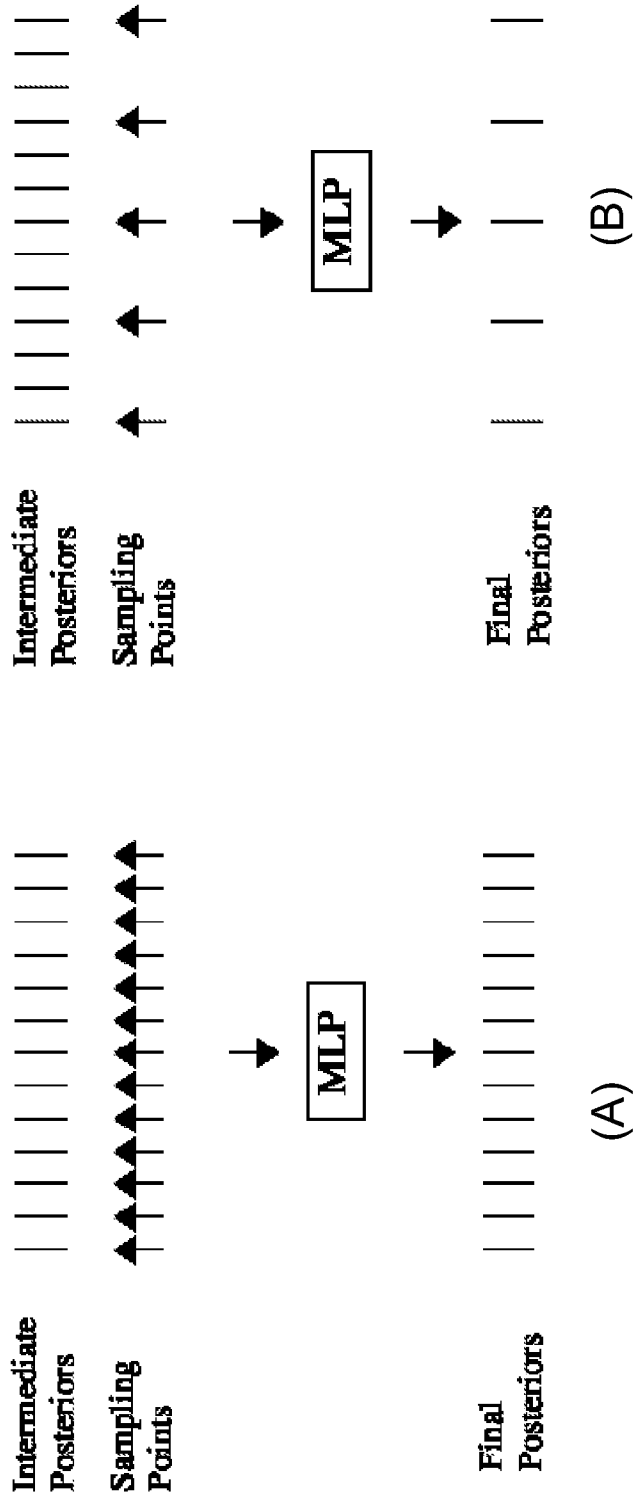

Intermediate Posteriors
Sampling Points
MLP
Final Posteriors
(A)
Intermediate Posteriors
Sampling Points
MLP
Final Posteriors
(B)

Fig. 4

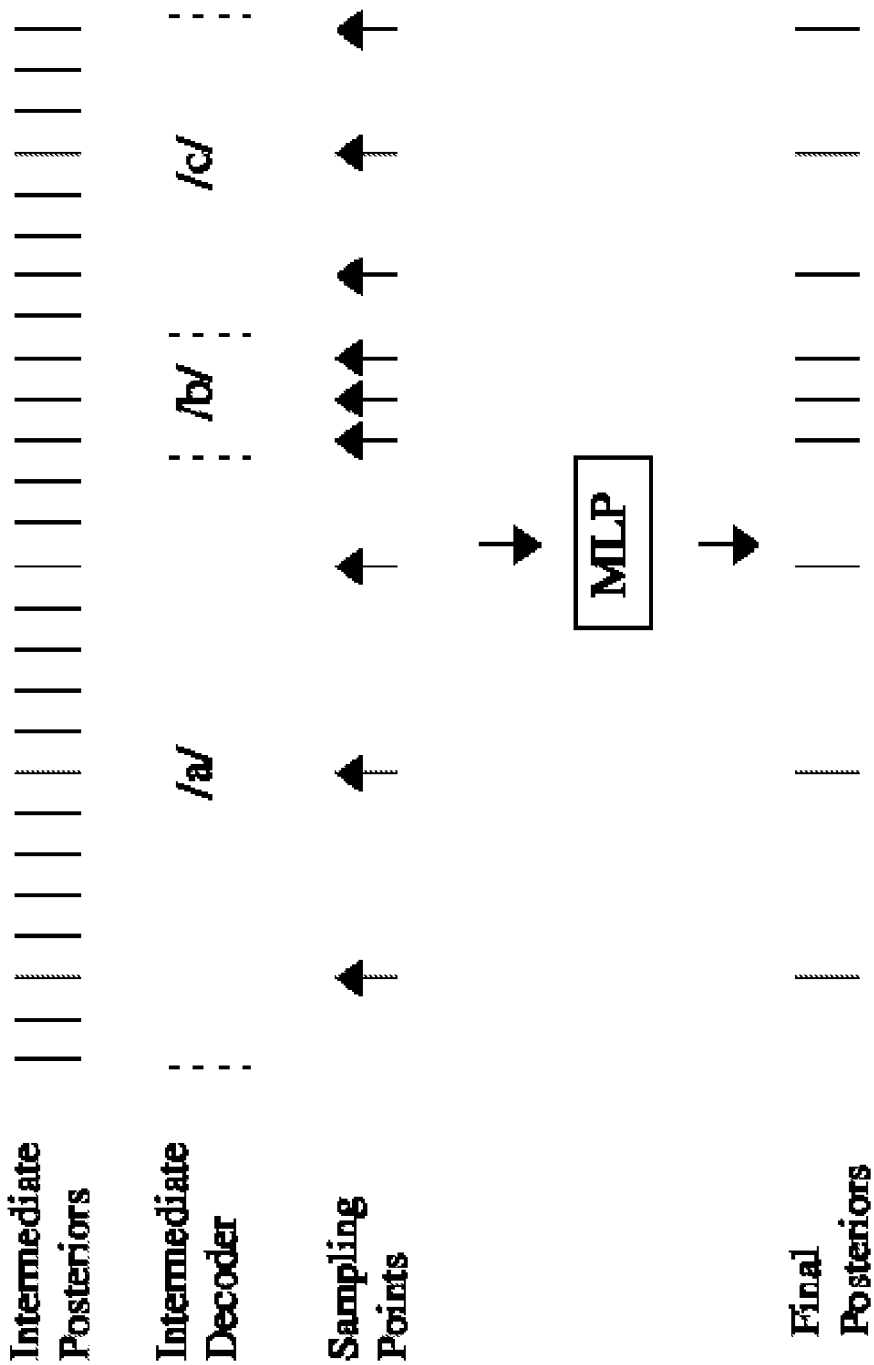

Intermediate Posteriors
Intermediate Decoder
Sampling Points
/a/
/b/
/c/
MLP
Final Posteriors

Fig. 5

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2009/058563

A. CLASSIFICATION OF SUBJECT MATTER
INV. G10L15/02
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, INSPEC, WPI Data, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JOEL PINTO ET AL: "Exploiting contextual information for improved phoneme recognition" ACOUSTICS, SPEECH AND SIGNAL PROCESSING, 2008. ICASSP 2008. IEEE INTERNATIONAL CONFERENCE ON, IEEE, PISCATAWAY, NJ, USA, 31 March 2008 (2008-03-31), pages 4449-4452, XP031251585 ISBN: 978-1-4244-1483-3	1-9, 11-14
A	abstract page 4449, right-hand column, paragraphs 2,4 page 4451, left-hand column, paragraph 2	10

-/--

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
8 June 2010	16/06/2010

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Greiser, Norbert

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2009/058563

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	HERMANSKY H ET AL: "Tandem connectionist feature extraction for conventional HMM systems" ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 2000. ICASSP '00. PROCEEDING S. 2000 IEEE INTERNATIONAL CONFERENCE ON 5-9 JUNE 2000, PISCATAWAY, NJ, USA,IEEE, vol. 3, 5 June 2000 (2000-06-05), pages 1635-1638, XP010507669 ISBN: 978-0-7803-6293-2 abstract page 1635, paragraph [2. Approach] figure 1	1,12
A	US 2004/172247 A1 (YOON SU-YEON [KR] ET AL) 2 September 2004 (2004-09-02) abstract	1,12

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2009/058563

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004172247 A1	02-09-2004	JP 2004258658 A	16-09-2004
		KR 20040076035 A	31-08-2004