CONVOLUTIONAL NEURAL NETWORKS-BASED CONTINUOUS SPEECH RECOGNITION USING RAW SPEECH SIGNAL

Dimitri Palaz[⋆†] *Mathew Magimai.-Doss*[⋆] *Ronan Collobert*[⋆]

[⋆] Idiap Research Institute, Martigny, Switzerland
[†] Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
{dimitri.palaz, mathew}@idiap.ch ronan@collobert.com

ABSTRACT

State-of-the-art automatic speech recognition systems model the relationship between acoustic speech signal and phone classes in two stages, namely, extraction of spectral-based features based on prior knowledge followed by training of acoustic model, typically an artificial neural network (ANN). In our recent work, it was shown that Convolutional Neural Networks (CNNs) can model phone classes from *raw* acoustic speech signal, reaching performance on par with other existing feature-based approaches. This paper extends the CNN-based approach to large vocabulary speech recognition task. More precisely, we compare the CNN-based approach against the conventional ANN-based approach on Wall Street Journal corpus. Our studies show that the CNN-based approach achieves better performance than the conventional ANN-based approach with as many parameters. We also show that the features learned from raw speech by the CNN-based approach could generalize across different databases.

Index Terms— automatic speech recognition, convolutional neural networks, raw signal, feature learning

1. INTRODUCTION

State-of-the-art automatic speech recognition (ASR) systems typically divide the task into several sub-tasks, which are optimized in an independent manner [1]. In a first step, the data is transformed into features, usually composed of a dimensionality reduction phase and an information selection phase, based on the task-specific knowledge of the phenomena. These two phases have been carefully handcrafted, leading to state-of-the-art features such as mel frequency cepstral coefficients (MFCCs) or perceptual linear prediction cepstral features (PLPs). In a second step, the likelihood of subword units such as, phonemes is estimated using generative models or discriminative models. In a final step, dynamic programming techniques are used to recognize the word sequence given the lexical and syntactical constraints.

Recent advances in machine learning have made possible systems that can be trained in an end-to-end manner, i.e. systems where every step is *learned* simultaneously, taking into account all the other steps and the final task of the whole system. It is typically referred to as *deep learning*, mainly because such architectures are usually composed of many layers (supposed to provide an increasing level of abstraction), compared to classical "shallow" systems. As opposed to "divide and conquer" approaches presented previously (where each step is independently optimized) deep learning approaches are often claimed to lead to more optimal systems, as they alleviate the need of finding the right features by instead training a stack of features in a end-to-end manner, for a given task of interest. While there is a good success record of such approaches in the computer vision [2] or text processing fields [3], deep learning approaches for speech recognition still rely on spectral-based features such as MFCC [4]. Some systems have proposed to learn features from "intermediate" representation of speech, like mel filter bank energies and their temporal derivatives.

In our recent study [5], it was shown that it is possible to estimate phoneme class conditional probabilities by using raw speech signal as input to convolutional neural networks [6] (CNNs). On TIMIT phoneme recognition task, we showed that the system is able to learn features from the raw speech signal, and yields performance similar or better than conventional ANN, more specifically multilayer perceptron (MLP), based system that takes cepstral features as input.

The goal of the present paper is to ascertain two aspects of the CNN-based system: its scalability to large vocabulary speech recognition and the invariance of the features learned from raw speech across domains. For the first aspect, we compare the CNN-based approach against the conventional ANN-based approach with different architectures on Wall Street Journal corpus. Our studies show that the CNN-based approach yields better performance than ANN-based approach with as many parameters. For the second aspect, we propose a cross-domain experiment, where the features learned on one database are used on another one. We show that these features could generalize given enough training data.

The remainder of the paper is organized as follows. Section 2 presents a brief survey of related literature. Section 3 presents the architecture of the proposed system. Section 4 presents the experimental setup and Section 5 presents the results. Section 6 presents the discussion and concludes the paper.

2. RELEVANT LITERATURE

Hybrid HMM/ANN approach was originally developed with ANNs that have single hidden layer and classify context-independent phonemes given cepstral feature as input. More recently, ANNs with deep learning architectures, more precisely, deep belief network or deep neural networks (DNNs) [7, 8], which can yield better system than a single hidden layer MLP have been proposed to address various aspects of acoustic modeling. More specifically, use of context-dependent phonemes [9, 10]; use of spectral features as opposed to cepstral features [4, 11]; CNN-based system with

This work was supported by the HASLER foundation (www.haslerstiftung.ch) through the grant "Universal Spoken Term Detection with Deep Learning" (DeepSTD). The authors also thank their colleague Ramya Rasipuram for providing the HMM/GMM baseline.

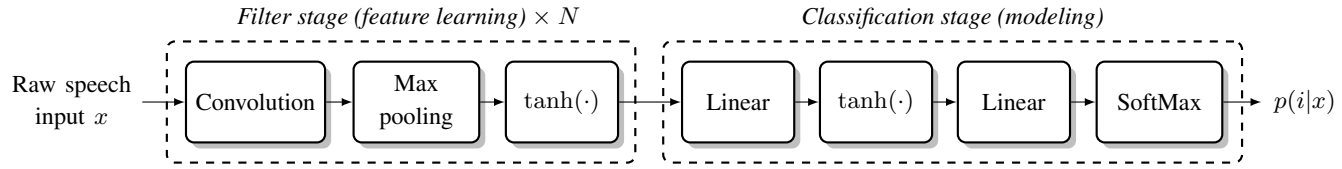


Fig. 1. *Convolutional Neural Network. Several stages of convolution/pooling/tanh might be considered. Our network included 3 stages. The classification stage can have multiple hidden layers.*

mel filter bank energies as input [12, 13]; combination of different features [14]; CNN-based phoneme recognition system with raw speech signal input trained in end-to-end manner [15]; multichannel processing using CNNs [16], to name a few.

Features learning from raw speech using neural networks-based systems has been investigated in [17]. In this approach, the learned features are post-processed by adding their temporal derivatives and used as input for another neural network. In comparison to that, in our approach, the features are learned jointly with the acoustic model. A recent study investigated acoustic modeling using raw speech as input to a DNN [18]. The study showed that raw speech based system is outperformed by spectral feature based system.

3. CONVOLUTIONAL NEURAL NETWORKS

3.1. Architecture

The proposed network is given a sequence of raw input signal, split into frames, and outputs a score for each classes, for each frame. It is presented in Figure 1. The network architecture is composed of several filter stages, followed by a classification stage. A filter stage involves a convolutional layer, followed by a temporal pooling layer and a non-linearity ($\tanh()$). Our optimal architecture included three filter stages. Processed signal coming out of these stages are fed to a classification stage, which in our case is a multilayer perceptron, which can have multiple hidden layers. It outputs the conditional probabilities $p(i|x)$ for each class i, for each frame x.

3.2. Convolutional layer

While "classical" linear layers in standard MLPs accept a fixed-size input vector, a convolution layer is assumed to be fed with a sequence of T vectors/frames: $X = \{x^1 \quad x^2 \quad \dots \quad x^T\}$. A convolutional layer applies the same linear transformation over each successive (or interspaced by dW frames) windows of kW frames. For example, the transformation at frame t is formally written as:

$$M \begin{pmatrix} x^{t-(kW-1)/2} \\ \vdots \\ x^{t+(kW-1)/2} \end{pmatrix}, \tag{1}$$

where M is a $d_{out} \times d_{in}$ matrix of parameters, d_{in} denotes the input dimension and d_{out} denotes the output dimension of each frame. In other words, d_{out} filters (rows of the matrix M) are applied to the input sequence.

3.3. Max-pooling layer

These kind of layers perform local temporal $\max$ operations over an input sequence. More formally, the transformation at frame t is written as:

$$\max_{t-(kW-1)/2 \leq s \leq t+(kW-1)/2} x_s^d \qquad \forall d \tag{2}$$

with x being the input, kW the kernel width and d the dimension. These layers increase the robustness of the network to minor temporal distortions in the input.

3.4. Network training

The network parameters θ are learned by maximizing the log-likelihood L, given by:

$$L(\theta) = \sum_{n=1}^{N} \log(p(i_n|x_n, \theta)) \tag{3}$$

for each input x and label i, over the whole training set, with respect to the parameters of each layer of the network. Defining the $\texttt{logsumexp}$ operation as: $\text{logsumexp}_i(z_i) = \log(\sum_i e^{z_i})$, the likelihood can be expressed as:

$$L = \log(p(i|x)) = f_i(x) - \underset{j}{\text{logsumexp}}(f_j(x)) \tag{4}$$

where $f_i(x)$ described the network score of input x and class i. Maximizing this likelihood is performed using the stochastic gradient ascent algorithm [19].

4. EXPERIMENTAL SETUP

In this section, we present the two studies, the databases, the baselines and the hyper-parameters of the networks.

4.1. Study 1: Large vocabulary speech recognition

We evaluate the scalability of the proposed system on a large vocabulary speech recognition task on the WSJ corpus. The CNN-based system is used to perform the feature learning and acoustic modeling steps, by computing the posterior probabilities of context-dependent phonemes from raw speech.

The decoder is an HMM. The scaled likelihoods are estimated by dividing the posterior probability by the prior probability of each class, estimated by counting on the training set. The hyper parameters such as, language scaling factor and the word insertion penalty are determined on the validation set.

4.2. Study 2: Feature invariance

The filter stage of the CNN-based system can be seen as a feature extractor or matching filters [5]. In order to ascertain the invariance capability of these filters, we propose a cross-domain experiment,

where the filter stage is first trained on one domain, then it is fixed and used as feature extractor on another domain. More precisely, we propose the following procedure, as illustrated in Figure 2:

1. The whole network is trained on one database.
2. The weights of every convolutional layer are fixed, and only the classification stage, as presented in Figure 1, is trained on a second database.

For the experiments, in addition to the WSJ corpus, we use the TIMIT corpus. We present two studies. First, a word recognition study on WSJ with the features learned on TIMIT corpus and a second study on TIMIT phoneme recognition task with the features learned on WSJ corpus. The network has the same hyper-parameters in both cases. For the phoneme recognition study, the decoder is a standard HMM decoder, with constrained duration of 3 states, and considering all phonemes equally probable.

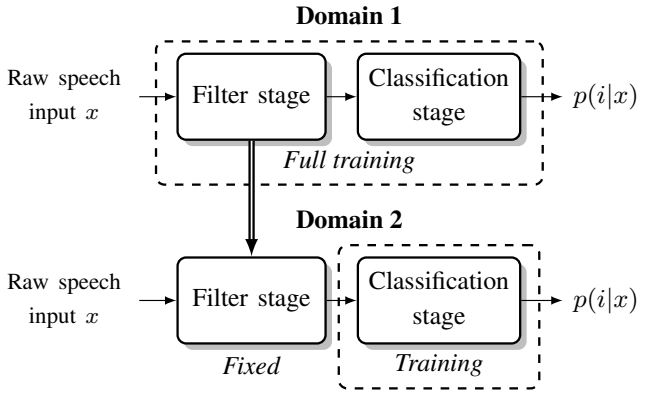


Fig. 2. *Illustration of the cross-domain experiment. The filter stage is trained on domain 1, then used as feature extractor on domain 2.*

4.3. Databases

The SI-284 set of the Wall Street Journal (WSJ) corpus [20] is formed by combining data from WSJ0 and WSJ1 databases, sampled at 16 kHz. The set contains 36,416 sequences, representing around 80 hours of speech. Ten percent of the set was taken as validation set. The Nov'92 set was selected as test set. It contains 330 sequences from 10 speakers. The dictionary was based on the CMU phoneme set, 40 context-independent phonemes. 2776 tied-states were used in the experiment. They were derived by clustering context-dependent phones in HMM/GMM framework using decision tree state tying. The dictionary and the bigram language model provided by the corpus were used. The vocabulary contains 5000 words. The HMM/GMM system yields a performance of 5.1% word error rate. It is comparable to the performance reported in literature [21].

The TIMIT acoustic-phonetic corpus consists of 3,696 training utterances (sampled at 16kHz) from 462 speakers, excluding the SA sentences. The cross-validation set consists of 400 utterances from 50 speakers. The core test set was used to report the results. It contains 192 utterances from 24 speakers, excluding the validation set. The 61 hand labeled phonetic symbols are mapped to 39 phonemes with an additional garbage class, as presented in [22].

4.4. Features

Raw features are simply composed of a window of the temporal speech signal (hence, $d_{in} = 1$ for the first convolutional layer). The window is normalized such that it has zero mean and unit variance. We also performed several baseline experiments, with MFCC as input features. They were computed (with HTK [23]) using a 25 ms Hamming window on the speech signal, with a shift of 10 ms. The signal is represented using 13th-order coefficients along with their first and second derivatives, computed on a 9 frames context.

4.5. Baseline systems

We compare our approach with the standard HMM/ANN system using cepstral features. We train ANNs with two different architectures. More precisely, we use an ANN with one single hidden layer, referred to as *ANN-1L* and an ANN with three hidden layers, referred to as *ANN-3L*. The input to the ANNs are MFCC with several frames of preceding and following context. The number of context frame was tuned on the validation set. We do not pre-train the network.

4.6. Networks hyper-parameters

The hyper-parameters of the network are: the input window size w_{in}, corresponding to the context taken along with each example, the kernel width kW_n, the shift dW_n and the number of filters d_n of the n^{th} convolution layer, the pooling width kW_{mp} of maxpooling layers and the hidden layers width. They were tuned by early-stopping on the validation set. Ranges which were considered for the grid search are reported in Table 1. It is worth mentioning that, for the first layer of convolution, the best performance was found with a kernel width (kW_1) of 50 samples (as for this layer, each frame contains only one sample), corresponding to 3 ms of speech, and a shift of 10 samples.

We train two architectures: the first one is composed of 3 convolutional layers and 1 hidden layer and is referred to as *CNN-1L*. The second one is composed of 3 convolutional layers and 3 hidden layers and is referred to as *CNN-3L*. The best performance was found with: 310 ms of context, 5 frames kernel width, 80, 60 and 60 filters, 500 hidden units and 2 pooling width. The second architecture has the same hyper-parameters, with 1000 hidden units for the three hidden layers. For the baselines, the *ANN-1L* uses 1000 nodes for the hidden layer and 9 frames as context. The *ANN-3L* system uses 1000 nodes for each hidden layer and 9 frames as context. For the cross-domain study, the classifier stage has one hidden layer of 500 units for each case. The experiments were implemented using the *torch7* toolbox [24].

Table 1. Network hyper-parameters

Parameters	Units	Range
Input window size (w_{in})	ms	100-700
Kernel width of the first conv. (kW_1)	samples	10-90
Kernel width of the n^{th} conv. (kW_n)	frames	1-11
Number of filters per kernel (d_{out})	filters	20-100
Max-pooling kernel width (kW_{mp})	frames	2-6
Number of hidden units in the classifier	units	200-1500

5. RESULTS

5.1. Large vocabulary speech recognition

The results for the LVCSR study, expressed in terms of Word Error Rate (WER) for the baseline systems and the proposed system, are presented in Table 2, along with the number of parameters of the network. As it can be observed, the *CNN-1L* based system outperforms the *ANN-1L* based baseline system, and the *CNN-3L* based system also outperform the *ANN-3L* based system. with as many parameters. Furthermore, the *CNN-1L* based system performance is comparable to the *ANN-3L* based system. These results indicate that CNNs result in simpler features which can be classified easily when compared to MFCC features.

Table 2. Word Error Rate on the Nov'92 testset

Features	System	#Params.	WER
MFCC	*ANN-1L*	3.1M	7.0 %
MFCC	*ANN-3L*	5.6M	6.4 %
RAW	*CNN-1L*	3.1M	6.7 %
RAW	*CNN-3L*	5.6M	5.6 %

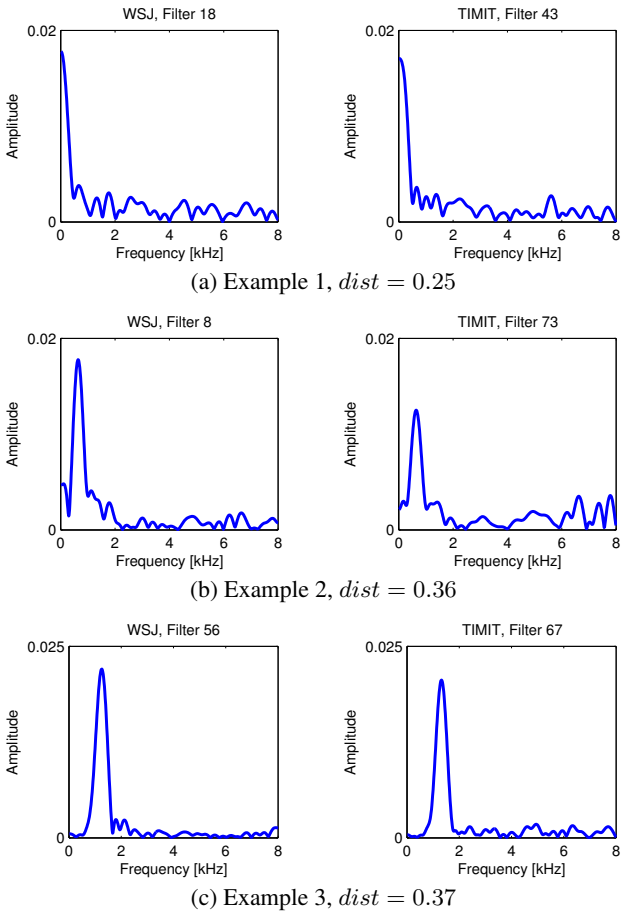
5.2. Features invariance

The results for the cross-domain study are presented in Table 3. On the TIMIT corpus, the features trained on WSJ yield similar performance with the features trained on TIMIT. On the WSJ corpus, the features trained on TIMIT yield lower performance. These results suggest that there is some level of dependency on the data used for training the CNN. More specifically, the low performance on WSJ corpus could be explained by the fact that TIMIT is small corpus with few amount of variability.

We compared the filters learned on WSJ corpus with the filters learned on TIMIT corpus. This was done by: computing the magnitude of the Fourier transform of the filters of the first convolution layer, learned on TIMIT and on WSJ; normalizing it; and finding the closest filter using symmetric Kullback-Lieber divergence as metric. Figure 3 presents normalized frequency responses of a few filters learned on WSJ (on the left column) and the closest filters learned on TIMIT (on the right column). It can be observed that the peaks are centered around the same frequency between the two corpora, although there is a difference in the spectral balance, specially see Figure 3(b). These differences in the spectral balance could possibly be related to the variability in the data across domains and explain performance differences. This needs further investigation and is part of our future work.

Table 3. Cross-domain results. The TIMIT results are given in PER, and the WSJ results are given in WER.

Test domain	Features	Error Rate
TIMIT	Learned on TIMIT	32.3 %
	Learned in WSJ	32.4 %
WSJ	Learned on WSJ	6.7 %
	Learned on TIMIT	10.1 %



(a) Example 1, $dist = 0.25$

(b) Example 2, $dist = 0.36$

(c) Example 3, $dist = 0.37$

Fig. 3. *Examples of three close pairs of filters learned. The left column is on WSJ, the right on TIMIT.*

6. CONCLUSION

In this paper, we investigated the scalability of an ASR approach based on CNNs, which takes as input the raw speech signal, to large vocabulary task. Our studies on WSJ corpus showed that the CNN-based system is able to achieve better performance than the ANN-based system, which takes standard cepstral features as input. These findings are inline with the phoneme recognition studies reported on TIMIT corpus in [5]. In comparison to [18], where poor ASR performance was achieved with raw speech signal as input to DNN, our LVCSR study indicates that CNNs have an edge over DNNs in modeling raw speech signal. We also studied the generalization capability of the features learned by the CNN. The cross-domain experiment indicated that the features learned on large amount of data could generalize across domains.

Our future work will focus on studying the language independence of the CNN-based approach, as the standard cepstral feature extraction process does not have any such dependency.

7. REFERENCES

[1] H. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*, vol. 247, Springer, 1994.

[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[3] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[4] A. Mohamed, G.E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14 –22, January 2012.

[5] D. Palaz, R. Collobert, and M. Magimai.-Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks," in *Proc. of Interspeech*, 2013.

[6] Y. LeCun, "Generalization and network design strategies," in *Connectionism in Perspective*, R. Pfeifer, Z. Schreter, F. Fogelman, and L. Steels, Eds., Zurich, Switzerland, 1989, Elsevier.

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

[8] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. N. Sainath, "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 8297, 2012.

[9] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. of Interspeech*, 2011, pp. 437–440.

[10] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 3042, 2012.

[11] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems 22*, 2009, pp. 1096–1104.

[12] O. Abdel-Hamid, A. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Proc. of ICASSP*, 2012, pp. 4277–4280.

[13] T. N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for lvcsr," in *Proc. of ICASSP*, 2013, pp. 8614–8618.

[14] E. Bocchieri and D. Dimitriadis, "Investigating deep neural network based transforms of robust audio features for lvcsr," in *Proc. of ICASSP*, 2013, pp. 6709–6713.

[15] D. Palaz, R. Collobert, and M. Magimai. -Doss, "End-to-end phoneme sequence recognition using convolutional neural networks," *ArXiv e-prints*, Dec. 2013.

[16] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *Signal Processing Letters, IEEE*, vol. 21, no. 9, pp. 1120–1124, September 2014.

[17] N. Jaitly and G. Hinton, "Learning a better representation of speech soundwaves using restricted boltzmann machines," in *Proc. of ICASSP*, 2011, pp. 5884–5887.

[18] Z. Tüske, P. Golik, R. Schlüter, and H. Ney, "Acoustic modeling with deep neural networks using raw time signal for lvcsr," in *Interspeech*, Singapore, Sept. 2014, pp. 890–894.

[19] L. Bottou, "Stochastic gradient learning in neural networks," in *Proceedings of Neuro-Nmes 91*, Nimes, France, 1991, EC2.

[20] P.C. Woodland, J.J. Odell, V. Valtchev, and S.J. Young, "Large vocabulary continuous speech recognition using htk," in *Proc. of ICASSP*, apr 1994, vol. ii, pp. II/125 –II/128 vol.2.

[21] W. Reichl and W. Chou, "Robust decision tree state tying for continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, pp. 555–566, 2000.

[22] K. F Lee and H. W Hon, "Speaker-independent phone recognition using hidden markov models," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.

[23] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, "The htk book," *Cambridge University Engineering Department*, vol. 3, 2002.

[24] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A matlab-like environment for machine learning," in *BigLearn, NIPS Workshop*, 2011.