ROBUST ISOLATED SPEECH RECOGNITION USING BINARY MASKS

Seliz Gülsen Karadoğan¹, Jan Larsen¹, Michael Syskind Pedersen², Jesper Bünsow Boldt²

¹) Informatics and Mathematical Modelling, Technical University of Denmark,
DK-2800, Kgs. Lyngby, Denmark
²) Oticon A/S, Kongebakken 9, DK-2765 Smørrum, Denmark
{seka, jl}@imm.dtu.dk , {msp,jeb}@oticon.dk

ABSTRACT

In this paper, we represent a new approach for robust speaker independent ASR using binary masks as feature vectors. This method is evaluated on an isolated digit database, TIDIGIT in three noisy environments (car, bottle and cafe noise types taken from DRCD Sound Effects Library). Discrete Hidden Markov Model is used for the recognition and the observation vectors are quantized with the K-means algorithm using Hamming distance. It is found that a recognition rate as high as 92% for clean speech is achievable using Ideal Binary Masks (IBM) where we assume priori target and noise information is available. We propose that using a Target Binary Mask (TBM) where only priori target information is needed performs as good as using IBMs. We also propose a TBM estimation method based on target sound estimation using non-negative sparse coding (NNSC). The recognition results for TBMs with and without the estimation method for noisy conditions are evaluated and compared with those of using Mel Frequency Cepstral Coefficients (MFCC). It is observed that binary mask feature vectors are robust to noisy conditions.

1. INTRODUCTION

Automatic Speech Recognition (ASR) systems have been improving significantly since the 50's. However, there are still many challenges to be surpassed to reach the human performance or beyond. It is well known that one of the key challenges is the robustness under noisy conditions. Another challenge is the need for innovative modeling frameworks. Most of the work has been focusing on the successful representations such as mel frequency cepstral coefficients (MFCC). However, because of a long history of research within the current ASR paradigm, the performance enhancement usually reported is very little. We will suggest a new approach which gives the state of the art performance that is robust to noisy environments.

Since the human auditory system has a great performance, it is tempting to use the human auditory system as an inspiration for an efficient ASR system. Auditory Scene Analysis (ASA) studies perceptual audition and describes the process how the human auditory system organizes sound into meaningful segments[1]. Computational ASA (CASA) makes use of some of the ASA principles and it is claimed that the goal of CASA is the ideal binary mask (IBM) [2]. IBM is a binary pattern obtained with the comparison of the target and the noise signal energies with priori information of target and noise signals separately. IBMs have been shown to improve speech intelligibility when applied to noisy speech signals. The listeners have been imposed to the resynthesized speech signals from the IBM-gated signal and almost perfect recognition results have been obtained even for a signal-to-noise-ratio (SNR) as low as -60 dB which corresponds to pure noise [3, 4]. Having proven to make improvements on speech intelligibility of humans, it is inevitable not to make the use of CASA and thus IBMs for machine recognition systems. Green et. al. have studied this in [5]. They used CASA as a preprocessor to ASR and used only the time-frequency regions of the noisy speech which are dominated by the target signal to obtain the recognition features. Therefore, they concluded that occluded (incomplete) speech might contain enough information for the recognition.

In this work we go one step further and explore the possibility that not only the occluded speech but the mask itself might carry sufficient information for ASR. The most obvious benefit of this new approach is the simplicity with the use of the binary information on the mask. The difficulty about using this method would be the need for the priori information of the target and noise signals to estimate the IBM. However, we minimize this need by using Target Binary Mask (TBM) where only target information is needed and compared to a speech shaped noise (SSN) matching the long term spectrum of a large collection of speakers. Using TBMs has also been shown to give high human speech intelligibility [4]. In addition, we propose a TBM estimation method based on non-negative sparse coding (NNSC)[6].

This paper will focus on a speaker-independent isolated digit recognizer with hidden Markov model (HMM) using the binary masks as the feature vectors. In Section 2 we give the modeling framework. The experiments and results are explained in Section 3. Finally Section 4 states the conclusion.

2. MODELING FRAMEWORK

2.1 Ideal Binary Masks

The computational goal of CASA, the IBM, is obtained by keeping the time-frequency regions of a target sound which have more energy than the interference and discarding the other regions. More specifically, it is one when the target is stronger than the noise for a local criteria (LC), and zero elsewhere. The time-frequency (T-F) representation is obtained by using the model of the human cochlea as the basis for data representation [7]. If \( T(t, f) \) and \( N(t, f) \) denote the target and noise time-frequency magnitude, then the IBM is defined as

\[
IBM(t, f) = \begin{cases} 
1, & \text{if } T(t, f) - N(t, f) > LC \\
0, & \text{otherwise} 
\end{cases} \]

(1)

Figure 1 shows time-frequency representations of the target, noise and mixture signals. The target is digit six by a male speaker while the noise is SSN with 0dB of SNR. The corresponding IBM with LC of 0dB is also seen in Figure 1. Calculating an IBM requires that the target and the noise are available separately. One of the other properties of an IBM is that it sets the ceiling performance for all binary masks. Therefore, it is crucial that we know the results with IBMs before exploring any alternative mask definitions.

LC and SNR values in Equation 1 are two important parameters in our system. If LC is kept constant, increasing or decreasing the SNR makes the mask get closer to all-ones mask or all-zeros mask respectively. The change in IBMs for a fixed LC with different SNR values is shown in Figure 2 for a digit sample. As also seen from this figure, with fixed threshold, low or high SNR values result in masks with little or redundant information respectively. Meanwhile, increasing the SNR value is identical to decreasing the LC value and vice versa. Therefore, the relative criterion \( RC = LC - SNR \) was defined in [4] and the effect of RC of an IBM on speech perception was
studied. They calculated IBM’s with priori target and noise information and multiplied the mixture signal with the corresponding IBM’s. They exposed human subjects to resynthesized IBM-gated mixtures and found high human speech intelligibility (over 95%) for the RC range of [-17dB, 5dB]. We took this RC range as a reference and the results of our ASR system coincided with human speech perception results in terms of RC range which is shown in section 3.

2.2 Target Binary Masks

The binary mask calculated based on only the target signal was studied and is called Target Binary Mask (TBM) [8]. TBM’s were further investigated in [4] in terms of speech intelligibility and the results were comparable to those of IBM’s. The definition of TBM as seen in equation 2 is very similar to that of IBM except that while obtaining TBM the target T-F regions are compared to a reference SSN matching the long-term spectrum of the target speaker. (It is also possible to compare the target to a frequency dependent threshold corresponding to the long term spectrum of SSN)

$$TBM(t,f) = \begin{cases} 1, & \text{if } T(t,f) - SSN(t,f) > LC \\ 0, & \text{otherwise} \end{cases}$$  

Figure 3 illustrates the T-F representation of a target signal and the mixture signal with cafe noise at 0dB SNR. That figure also shows the resultant IBM and TBM patterns with LC of 0dB, and the difference between them is discernible. The TBM mimics the target pattern better, whereas the IBM pattern depends on the noise type.

Some of the properties of TBM can be very practicable. First of all, acquiring a TBM needs only the priori information of the target. Therefore, estimating the TBM can be much more convenient in some applications, especially if speech enhancement techniques are used. In the case of an ASR system that is robust to noise types, use of TBM in the training stage require less computational effort as opposed to the use of IBM’s where it is needed to include all IBM’s for all different noise types in the training stage.

2.3 ASR Using Binary Masks

As mentioned previously, we investigate if the mask itself can be used to recognize different words. The distinctivity of the masks can be observed easily in Figure 4 in which IBM’s for four different digits with SNR of -6dB using SSN as interference are shown. (Note that IBM is identical to TBM when the noise type is SSN) Moreover, as seen in Figure 5, the masks for different speakers for the same digit are very similar. Thus, the patterns in every mask are characteristic for each digit which concludes that these patterns are promising representations for speech recognition.

Figure 3: Illustration of T-F representations of a target (digit six), mixture (target+cafe noise) and mixture signals with the resultant IBM and TBM red regions: highest energy, blue regions: lowest energy.

Figure 4: IBMs for different digits for the same speaker

We use a discrete Hidden Markov Model (HMM) as the recognition engine [9]. As the vector quantization method before HMM, we choose to use K-means algorithm which has been shown to perform as well as many other clustering algorithms and is computationally efficient [10] and proven to be successfully applicable to classify binary data [11]. Figure 6 illustrates the acquisition of the feature vectors to be classified by K-means. We stack the columns of the IBM into a vector. The number of columns to be stacked...
is a parameter that has been optimized for this work (it is 3 for this study) as well as other parameters: the codebook size, the state number of the HMM, the number of frequency bands, and the window length of the IBM. The optimization process can be found in detail in [12].

![Diagram](image)

Figure 6: Acquisition of the feature vectors to be clustered by K-means.

The whole system is summarized in Figure 7. First, the masks for training and test data are calculated. The feature vectors obtained from IBMs are quantized with K-means to acquire the observed outputs for discrete HMM. One HMM for each digit is trained with the corresponding data. Finally, the test masks are input to each HMM and the test digit is assigned to the one with the highest likelihood. We use only clean data for training. However, for testing we use clean data to see the best performance that can be obtained with our system, unprocessed mixture signal to see the improved results can be obtained with our system, unprocessed mixture signal to see the best performance that can be obtained with our system.

![Diagram](image)

Figure 7: The schematics representation of the system used.

2.4 Estimation of TBMs

Estimation of TBM is simpler compared to that of an IBM as mentioned previously. Once the target signal is estimated, it is compared to a reference SSN signal in T-F domain. For speech and noise separation, non-negative sparse coding (NNSC), combination of sparse coding and non-negative matrix factorization, is used [6]. This method was proven to be successful for wind noise reduction in [13], and we took this work as reference for our method.

The principle in NNSC is to factorize the non-negative signal, X into a dictionary, W and a code, H:

\[ X \approx WH. \]  

The columns of the dictionary can be considered as the basis and the code matrix can be considered to have the weights for each of the basis vectors constituting the signal X. In our case X is the T-F representation of a signal which is non-negative (Details about the acquisition of T-F spectrogram is in section 3). We use the method described in [13] that is based on the algorithm in [14]. W and H are initialized randomly, and updated according to the equations below until convergence:

\[ H \leftarrow H \times \frac{W^T X}{W^T WH + \lambda}, \]  

\[ W \leftarrow W \times \frac{XH^T + W \times (1. \cdot W^T H^2 \times W)}{W^T H^2 + W \times (1. \cdot (XH^T \times W))}, \]

Here, (.) indicate direct multiplication, while others indicate point wise multiplication and division. 1 is a square matrix of ones of suitable size.

When the speech signal is noisy, and if the noise signal is assumed to be additive, then

\[ X = X_s + X_n \approx [W_i H_i] \begin{bmatrix} H_i \\ H_n \end{bmatrix}, \]

where \( X_s \) and \( X_n \) denote the speech and noise. We precompute the noise dictionary \( W_n \) using noise recordings and using equations 4 and 5. We keep this precomputed \( W_n \) fixed and learn speech \( X_s \) using the following iterative algorithm,

\[ H_i \leftarrow H_i \times \frac{W_i^T \cdot X}{W_i^T \cdot W_i \cdot H_i + h_i}, \]

\[ H_n \leftarrow H_n \times \frac{W_n^T \cdot X}{W_n^T \cdot W_n \cdot H_n + h_n}, \]

\[ W_i \leftarrow W_i \times \frac{X_i H_i^T + W_i \times (1. \cdot W_i^T H_i^2 \times W_i)}{W_i^T H_i^2 + W_i \times (1. \cdot (X_i H_i^T \times W_i))}. \]

The clean speech is estimated as

\[ X_s = W_i H_i. \]

Finally, the TBM is estimated by comparing the estimated speech signal \( X_s \) to the reference SSN signal spectrogram using equation 2. As mentioned previously, different RC values lead to masks with different densities and only choosing the right RC values leads high recognition results. However, we learn the right RC values for ASR after training and testing with IBMs, where we have the pure target and noise signals. (The results can be seen in section 3 in figure 8). We assume that after NNSC we have the pure target and noise signals. (The results can be seen in section 3 in figure 8). We assume that after NNSC and LC values for the right RC value. However, to obtain the SNR between the estimated target and speech, we do not go back to time domain which would be a waste of time and computational power. Thus, we defined a new SNR in the T-F domain which is calculated by the ratio between the sum of all T-F bins of the target signal to the sum of all T-F bins of the noise signal and will be called as \( SNR_{TFD} \). We observed that \( RC_{TFD} = LC_{TFD} - SNR_{TFD} \) range is similar to RC range found before. (The results can be seen in section 3 in figure 10).

3. EXPERIMENTAL EVALUATIONS

Through the experiments, data from TIDIGIT database were used. The spoken utterances of 37 male and 50 female speakers for both training and test data were taken from the database. There are two examples from every speaker for each 11 digits (zero-nine, oh) making 174 training, 87 test and 87 verification utterances for each digit. The verification set has been used to obtain the optimized parameters for HMM and for NNSC and the final results.
are obtained using the test set. The experiments were carried out in MATLAB and an HMM toolbox for MATLAB by Kevin Murphy was used [15]. The experiments have also been verified using the HMMs in Statistical Toolbox of MATLAB. For NNSC the NMF-DTU toolbox for MATLAB [16] has been adjusted for our system and used. The time-frequency representations of the signals sampled at 8kHz have been obtained using gammatone filter with 32 frequency channels equally distributed on ERB scale within the range of [80Hz, 4000Hz]. The output from each filterbank channel was divided into 20 ms frames with 10 ms overlap. SSN, car, bottle and cafe noise were used through the experiments [17]. A left-to-right HMM with 10 states was used to model each digit. The binary vectors were quantized into a codebook of size 256 with K-means. The HMMs were trained with IBMs obtained with LC of 0 dB and with different SNR values in the range of [0dB, 16dB] with 2dB steps only using SSN as the reference noise signal. We compare the method with a standard approach using 20 static MFCC features. All parameters used for the MFCC are the same except for the optimized codebook size of 32. The optimal codebook size is smaller since we have less training data for MFCC. One minute of SSN, car, bottle and cafe noise used to obtain the dictionaries for NNSC. For train, verification or test noise samples different parts of corresponding noise types were used.

Recognition results obtained for the test set for IBMs with SSN for LC of 0 dB and different SNR values are presented in Figure 8. As seen, the rate curve is bell-shaped, i.e. the rate does not increase monotonously while SNR increases. This is because of the previously mentioned fact that either increasing or decreasing the SNR value results in masks closer to all-ones or all-zeros masks and thus in the decrease of the recognizability of the masks. If we look at the RC value, Figure 8 shows that 92% recognition rate is obtained for RC of -6 dB. Thus, the masks with RC of -6 dB gives the maximum performance.

![Figure 8: The recognition rates with IBMs for LC=0dB and SNR=[-2dB, 16dB]](image)

If the LC value can be adjusted so that the mask is as close to the maximum-performance mask as possible (RC is close to -6dB), we can obtain high recognition results for different SNR values. However, under noisy conditions choosing the correct LC value is a challenge since we do not know neither the SNR value nor the noise spectrogram in real life applications. This problem will be solved by using NNSC method assuming we have information about the noise characteristics. However, it is reasonable to check the recognition results that can be obtained comparing unprocessed mixture signals to SSN with adjusted LC values (results are obtained with different LC values and the best result is recorded) before exploring that method. Figure 9 shows the recognition rates obtained using HMMs trained with IBMs obtained by clean data and SSN, with test set added different noise types at an SNR range of [0dB, 20dB] (with adjusted RC value for the best performance). In that figure, the results obtained using static MFCC features is also shown. It can be seen that using IBM features yields more noise-robust recognition rates than using MFCC features. We point out the fact that we used only static MFCC features and did not use any of the improvement methods suggested for MFCC that results in a better performance [18]. Nevertheless, we did not use dynamical features that could be obtained from IBMs neither. In addition, we believe that the performance of IBMs for ASR can also be improved in various ways such as mask estimation methods [19]. Moreover, if we consider the ASR results obtained using MFCC within recent works, our results are comparable [18]. (We can not make a direct comparison though, since they use a different system and database) In addition, our method establishes a new route for robust ASR that is open for further improvements. (Some additional results and figures of the whole system can be found at [12]).

![Figure 9: The recognition rates for TBMs and MFCC features at SNR range of [0dB, 20dB]](image)

As mentioned previously, for NNSC we needed to find RC_{TFD} range giving high recognition results. The corresponding results can be seen in Figure 10 and -6dB of RC_{TFD} gives the maximum performance and RC between -16dB and 2dB gives reasonable recognition results (over 80%). The optimized parameters for NNSC for this work is the size of the dictionary of noise and speech, W_n and W_s. Other parameters \lambda, \lambda_s and ln were just equaled to be a very small number taking reference the results in [13]. To find the optimal parameters for the size of W_n and W_s, we checked the recognition results for different size numbers between 4 and 512 for all noise types with SNR_{TFD} of 10dB and LC of 0dB. We choose 64 for W_n and 128 for W_s based on the results seen in Figure 11.

![Figure 10: The recognition rates with IBMs for LC=0dB and SNR_{TFD}=[-2dB, 16dB]](image)

In Figure 12, the recognition rates obtained with noisy mixtures before and after using NNSC is shown. (with reference SSN at SNR_{TFD} of 0dB) As seen on the left of this figure, before NNSC, different LC values within right RC range found before (-4 dB to 2dB), result in sparse recognition rates. For cafe noise at 10dB SNR, it is seen that before NNSC the rates can change from 30% to 60% for those different LC values. However, after using NNSC to estimate the masks as explained, it is seen that the rates for those LC values gives the best performances solving the choice of the right LC values for our ASR system. Using NNSC not only solves this problem but also leads higher recognition results especially for low SNR values at the price of a decrease in recognition results for high SNR values. However, the decrease in high SNR values is not as much as the increase in low ones. Finally, we obtain 60% to 70%,
16% to 73% and 40% to 70% recognition rates for SNR values between 0dB and 20dB for car, bottle and cafe noises respectively which are comparable to the state-of-the-art results [18, 20].

4. CONCLUSION

In this paper, we investigated a new feature extraction method for ASR using ideal and target binary masks. It is found that using binary information from the masks directly as feature vectors results in high recognition performance. We constructed a speaker-independent isolated digit recognition system. The experiments were carried out with TIDIGIT database, using discrete HMM as the recognition engine. The K-means algorithm with hamming distance was used for vector quantization. The maximum recognition rate achieved for clean speech is 92%. In comparison to the MFCC features results, A TBM estimation method using non-negative sparse coding has been demonstrated to give state of the art performance. It is concluded that noise-robust ASR systems can be built using binary masks.

Acknowledgments: We acknowledge the independent work similar to our work that we became aware of after our model was developed [21].

References

[15] K. Murphy, “Hidden markov model(HMM) toolbox for MATLAB.”
[16] IMM Technical University of Denmark, “Nmfedtu toolbox”.

Figure 11: The recognition rates for different size of W<sub>c</sub> and W<sub>s</sub>

Figure 12: The recognition rates before and after NNSC