# SPECTRO TEMPORAL ANALYSIS OF SPEECH FOR SPANISH PHONEME RECOGNITION

*Sara Sharifzadeh, Javier Serrano, Jordi Carrabina*

Microelectronic Department
Univesitat Autònoma de Barcelona
sara.sharifzadeh@uab.cat, javier.serrano@uab.cat, Jordi.carrabina@uab.cat

## ABSTRACT

State of the art speech recognition systems (ASR), mostly use Mel-Frequency cepstral coefficients (MFCC), as acoustic features. In this paper, we propose a new discriminative analysis of acoustic features, based on spectrogram analysis. Both spectral and temporal variations of speech signal are considered. This will improve the recognition performance specially in case of noisy situation and phonemes with time domain modulations such as stops. In this method, the 2D Discrete Cosine Transform (DCT) is applied on small overlapped 2D hamming windowed patches of spectrogram of Spanish phoneme's and enhanced by means of bi-cubic interpolation. An adaptive strategy is proposed for the size of patches over the time to construct unique length vectors for different phonemes. These vectors are classified based on K-nearest neighbor (KNN) and linear discriminative analysis (LDA) and reduced rank LDA (RLDA). Experimental results demonstrate improvement in recognition performance for noisy speech signals and stops.

***Index Terms***— Automatic speech recognition, Spectrogram, DCT transform, TF, MFCC

## 1. INTRODUCTION

Automatic speech recognition (ASR) is the task of speech to text conversion by intelligent powerful computers. Today, ASR is widely used in many applications such as multimedia, medical and industrial systems. Although the generation of ASR goes back many years ago and existing systems work satisfactory in many cases, there are still many challenges remained unsolved. Great deal of research have been accomplished during last decades, to overcome these systems limitations.

The first step in recognition is the extraction of acoustic features. Signal processing methods are used to map the speech signal characteristics to a proper representation well suited for statistical modeling. A good feature extraction strategy should reduce the dimensionality and preserve the relevant information. Because of varying and sequential nature of speech, the features usually are extracted from very short time slots or frames. A frame has usually 10 to 25 $ms$. length.

In state of the art ASR, features are extracted from spectral variation of speech using Fourier transform [1]. Therefore, any temporal information are ignored. Just $\Delta$ and $\Delta\triangle$ features as first and second derivatives are added at each frame to embedded some information from a larger time span than single frame. But, they can not explicitly describe temporal modulations in speech signal. This is contradictory to the recent psycho-acoustical and neuro-physiological findings, about unified multi-resolution representation of the spectral and temporal features [2].

Regarding to these facts, recently, some researches have been performed on acoustic feature extraction based on both time and frequency information of speech signal. In [3], information in the joint time-frequency domain are studied and a sequential design of spectral discriminants followed by temporal discriminants are proposed and classified based on a linear discriminant framework. The sensitivity of temporal modulation in spectrogram is investigated in [4]. In this work, a 1D-FFT is applied to the critical energy bands of a spectrogram. But, the temporal features could not describe the joint and localized spectro-temporal modulation sensitivity.

In this paper, we propose an improved time-frequency (ITF) feature extraction method for Spanish phonemes recognition. These features are more similar to the work explained in [5]. The idea is to enhance the time-frequency (TF) acoustic features introduced in that work. We apply a localized 2D-DCT transform on the spectrogram of Spanish phonemes. This will provide us both temporal and spectral speech features inspired from the bio-scientific findings explained above. In order to evaluate these features, the classification is performed using K-nearest neighbor (KNN) and linear discriminant analysis (LDA). We have also applied Fiche's reduced rank LDA (RLDA) on acoustic features. This will significantly reduce the dimensionality of the classification task with out any loss. Another important advantage of the proposed approach is that, by developing such acoustic features with high spectro temporal resolution, they are highly robust against noise. In addition, these (ITF) features are capable to do better recognition on phonemes with time domain modulations (such as stops).

The rest of the paper is organized as follows; Section 2 describes different steps for both TF and ITF feature extraction. Section 3 is about classification on acoustic vectors. In section 4, we present the experimental results and a discussion. Finally, there would be a conclusion for this paper.

## 2. SPECTROGRAM TIME-FREQUENCY ANALYSIS

In this section different steps for building the primary TF feature vectors as well as the suggested ITF features and MFCC will be explained.

### 2.1. Time Frequency features (TF)

Each utterance of the corpus is first per-emphasized and then its spectrogram is obtained using $16KHz$ sampling frequency, 300 samples of Hamming window with 268 samples overlap and $2^{10}$ FFT points. Then, the logarithm of the magnitude of the spectrogram is normalized to have the global zero mean and unit variance. In the next step, the spectrogram of each phoneme is separated from the whole utterance according to the time labels of the corpus. When separating, an overlap of $30ms$ with both neighbor sides is considered due to co-articulation effects. Therefore, we obtain an independent spectrogram for each phoneme. The higher 113 frequency bins (frequencies higher than 6.23 KHz) are ignored because, there is no important acoustic information there. We have tested this, and no improvement was observed, by keeping those frequencies. Also, as the DCT transform will be applied later on different patches of the spectrogram, we should avoid to loos important information of the signal near the edge in low frequencies. Therefore, a spectrum of 25 frequency bins (390 Hz) is reflected about 0 Hz which means that they are appended over time axis symmetrically.

#### *2.1.1. Adaptive Local Patching and 2D DCT Transform*

At this step, each phoneme's spectrogram is divided into different local overlapped patches and 2 dimensional hamming window is applied on each patch. Then, 2 dimensional DCT transform with 2x oversampling is applied on each patch. The concept is to pool down the coefficients of patches and construct a vector of pooled coefficients over the time. Then, it would be possible to build an acoustic vector with consistent size for all phonemes by averaging over a fixed number of time intervals. But, we know that the time duration of phonemes differ due to their nature, speaker rate and etc. Therefore, the patch width should be chosen in an appropriate way to provide the minimum number of time intervals. In this paper, we proposed an adaptive patching approach to maintain at least the minimum number of required time intervals even for short length phonemes. Then, a map of patches is created out of initial spectrogram. This patch map is indexed with the averages of time and frequencies at each patch. For example, if there are $N$ frequency indexes and $P$ number of time indexes, there would be totally $N \times P$ number of patches in the map.

$$PatchMap = \sum_{t=1}^{P} \sum_{f=1}^{N} P_{tf} \tag{1}$$

First, the dimensionality of each patch is defined as $50 \times 20$ bins ($781Hz \times 1.25ms$ ) with $25 \times 2$ bins overlap ($390Hz \times 0.125ms$). The overlap frequency is the same as the one used for reflection of the low frequencies. Then, the algorithm checks whether there would be enough patches to form the minimum number of time intervals (in our algorithm it is 5). In case of insufficient number of patches, the patch width would be calculated adaptively to maintain the required intervals.

After patching, a 2 dimensional DCT transform with twice oversampling in both time and frequency direction is applied on each hamming windowed patch in the map. The highest energies in a DCT region always are collected in the top-left. Accordingly, in TF method, the 6 low order coefficients from the $3 \times 3$ triangular located at the top-left of each patch are extracted and the remaining higher harmonic's coefficients are ignored. Then, there are 6 coefficients per patch. A new cubic map is formed with 6 DCT coefficients at each time-frequency index of the patch map:

$$CubMap = \sum_{t=1}^{P} \sum_{f=1}^{N} \sum_{d=1}^{6} C_{ftd} \tag{2}$$

To create the final acoustic vector, first, the cubic map is divided along the time into five segments and averaged over time within each segment. The first and last segments correspond to the co-articulation effects which was considered as $30ms$ overlap before. The remaining bins, fall inside the rest 3 segments show the phoneme and we model them with the proportion of 3:4:3. This is because the central segment is the main part of each phoneme with higher durability. Finally, all coefficients at each time segment with the length of $6 \times N$ are pooled down and concatenated into a vector with the unique size of $6 \times N \times 5$ for all phonemes. Figure 2.1.1. shows the different steps to form the acoustic vector for a phoneme.

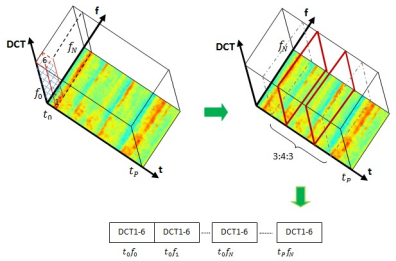


**Fig. 1**. Three main steps for forming an acoustic vector.

Besides the DCT features, the logarithm of the phoneme duration is added to the end of the vector and the total number

of features is $6 \times N \times 5 + 1$. In our case, $N$ is 17, then the total number of TF features per vector is 511.

### 2.2. ITF Method for Feature Extraction

The main two differences of the TF method and ITF method are in the 2 dimensional DCT transform and its coefficient selection strategy.

Instead of applying a 2D dimensional DCT transform with 2x oversampling on each hamming windowed patch, we have enhanced the coefficients by applying the 2D DCT transform, on an interpolated version of the hamming windowed patch. The interpolation scale is two and it is based on the "bi-cubic" interpolation method. Another difference is in the selected DCT coefficients. We have tested different numbers and selection strategies for extraction of DCT coefficients and found that instead of the 6 top left triangular coefficients, selection of the coefficients shown in figure 2.2. results in better performance.

**Fig. 2**. TF DCT coefficient selection strategy (Left), ITF DCT coefficient selection strategy (Right).

Therefore, the number of features in ITF is also 511 per phoneme vector.

### 2.3. Mel-frequency Cepstral Coefficients (MFCC)

The MFCC feature vectors are constructed by computing 13 MFCCs from each frame of the spectrogram. Then $\triangle$ and $\triangle\triangle$ features are also included, which gives the classical 39-dimensional feature vectors for each frame. Similar to TF and ITF features, the time axis is divided up into five segments. The two regions including spectra before and after the phoneme are $30ms$ wide and are centered at the beginning and end of the phoneme. A log-duration feature is also added.

## 3. CLASSIFICATION

In order to evaluate the acoustic feature vectors, the k-Nearest Neighbor (KNN) classifier and Linear discriminant analysis (LDA) have been used. Also, reduced rank LDA (RLDA) is implemented. In the following subsections, LDA and RLDA will be explained briefly.

### 3.1. Linear Discriminant Analysis (LDA)

Linear discriminant analysis for classification is based on the class posteriors $Pr(G|X)$ for optimal classification [6]. Assuming $f_k(x)$ is the class-conditional density of the feature $x$ in class $G = k$ and let $\pi_k$ be the prior probability of class $k$, with $\sum_{k=1}^{K} \pi_k = 1$. If we model each class density as multivariate Gaussian:

$$f_k(x) = \frac{1}{(2\pi)^{\frac{p}{2}} |\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma^{-1}(x-\mu_k)} \quad (3)$$

and consider a common covariance matrix $\Sigma_k = \Sigma, k = 1, .., K$, in comparing multiple classes, the linear discriminant function is:

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + log\pi_k \quad (4)$$

which is an equation linear in $x$. This linear function implies that the decision boundary between classes is linear in $x$; in $p$ dimensions a hyperplane. This function is calculated for all classes and the decision rule is $G(x) = argmax_k \delta_k(x)$. In practice, the parameters of the Gaussian distributions are unknown, and should be estimated using the training data:

- Prior probability of each class: $\hat{\pi}_k = \frac{N_k}{N}$ where $N_k$ is the number of observations in $class - k$ and $N$ is total number of observations;
- Mean value of each class: $\hat{\mu_k} = \sum_{gi=k} (\frac{x_i}{N_k})$;
- $\hat{\Sigma} = \sum_{k=1}^{K} \sum_{gi=k} \frac{(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T}{(N-K)}$.

### 3.2. Reduced-Rank Linear Discriminant Analysis (RLDA)

RLDA is an informative low-dimensional projections of the data. The $K$ class centroids in $p - dimensional$ input space could be considered as an affine subspace of $dimension \leq K$–1, and if $p$ is much larger than $K$, there will be a considerable drop in dimension. This subspace that is spanned by the $K$ centroids is of dimension $K$–1. Therefor, data can be viewed in $K - 1$ dimension without losing any information. This corresponds to our case where there are just $K = 31$ classes but $p = 511$ feature space. Thus there is a fundamental dimension reduction in LDA.

The reduced-rank LDA is performed using Fisher's optimization criterion, where the idea is to spread out the projected centroids as much as possible comparing with variance. In other words, the between-class variance $B$ is maximized relative to the within-class variance $W$; for more details we refer to [6]. Figure3.2. illustrates the LDA and RLDA classification for a 3 class problem.

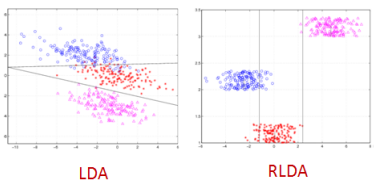


**Fig. 3**. Comparison between LDA and RLDA classification for a 3 class problem

**Table 1**. Comparison of the classification performance for MFCC, TF and ITF features using KNN classifier.

| KNN | clear | Pink(SNR=20) | Vowel | Stops | Length |
|---|---|---|---|---|---|
| MFCC | 64.73 | 45.83 | 69.77 | 59.32 | 196 |
| TF | 63.14 | 51.62 | 64.90 | 64.94 | 511 |
| ITF | 66.15 | 56.67 | 66.87 | 66.07 | 511 |

**Table 2**. Comparison of RLDA and LDA classification on MFCC, TF and ITF features of clear signal.

| | LDA | length | RLDA | length |
|---|---|---|---|---|
| MFCC | 72.71 | 196 | 72.71 | 30 |
| TF | 69.97 | 511 | 69.97 | 30 |
| ITF | 70.57 | 511 | 70.57 | 30 |

## 4. EXPERIMENTAL RESULT

In this section, the experimental results of different algorithms explained in previous sections would be described. The Spanish "Albayzin" corpus have been used for the experiments. There are 43897 number of data points, which have been divided into the standard training and test sets. There are totally 31 different phonemes in the Spanish language.

Three different feature extraction methods; MFCC, TF and ITF have been applied on the training and test data. The test data features have been extracted from both clean and noisy signals. These features are classified into one of the 31 class labels in the next step.

First, all the three feature sets are classified using $KNN$ classifier. Table 1, shows the results of classification for clear signal as well as pink noisy signal of 20 $SNR$. The performance of the features for Spanish "vowels" and "stops" has been also calculated separately. This is because, we are interested to know how the features work in case of both vowels and stops with frequency modulation effects and time domain modulation effects, respectively. The frequency modulations are captured well by MFCC features, while TF and ITF features consider both time and frequency changes.

The LDA classification results show that, the performance for MFCC is fairly better than the two other methods for clear signal , while in noisy condition, it is worse.

In the last table 2, the results of reduced rank LDA for three feature sets on clear signal is presented.

### 4.1. Discussion

According to the results, KNN classifier shows better results for time-frequency features in clear and specially noisy situation. On the other hand, the LDA classifier increases the performance of all methods and also works fairly better for MFCC features in clear situation and for TF and ITF features in noisy condition. Although, the LDA performance for both TF and ITF noisy features are better, the difference is not as significant as it was in KNN.

In addition, regarding to the results for vowels and stops, it is clear that MFCC works better for frequency domain modulations because, its mainly constructed out of spectral variations. However, TF and ITF features are capable of handling time domain modulations better. The results from RLDA Fisher method, shows the possibility to decrease the classification dimension, while saving the same performance which is an interesting characteristic of RLDA.

According to these results, the proposed ITF features outperform the TF features almost in all cases. It is also clear that the time frequency features are more robust in noisy situations. The dimensionality increase should not be a problem regarding to the increase in computers power.

## 5. CONCLUSION

In this paper, we have proposed an improved time-frequency (ITF) feature extraction method for Spanish phonemes recognition. These features have been compared with the previous time frequency features (TF), and the state of the art mel-frequency cepstral coefficients (MFCC). K-nearest neighbor (KNN), linear discriminant analysis (LDA) and reduced rank LDA (RLDA) classification methods have been used for classification, in both clean and noisy situation. In addition, the performance has been estimated for vowel and stop phonemes separately. The proposed features show better performance than the previous TF features in most cases. They also outperform the MFCC features in case of noisy situation and stop phonemes.

## 6. REFERENCES


[1] L. Rabiner and B.H. Juang, *Fundamentals of speech recognition*, Prentice-Hall, Upper Saddle River, NJ, USA, 1993.

[2] T. Chih, P. Ru, and S. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *Journal of the Acoustical Society of America*, vol. 118, pp. 887–906, 2005.

[3] S. Kajarekar, B. Yegnanarayana, and H. Hermansky, "A study of two dimensional linear discriminants for asr," *In proc. ICASSP*, 2001.

[4] L. Atlas and S. Shamma, "Joint acoustic and modulation frequency," 2003.

[5] J. Bouvrie, T. Ezzat, and T. Poggio, "Localized spectro-temporal cepstral analysis of speech," *In proc. ICASSP*, 2008.

[6] T. Hastie, R. Tiboshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, 2009.