Vowel controlled software

Magnus Raunio
Panagiotes Mousikides
mraunio@kth.se
pmou@kth.se

May 30, 2013

Abstract

We create a voice recognition system which classifies sound segemnts as one of the four following Swedish vowels: A, I, O and Ä. The purpose of this is to create a voice controlled game where you can use the four mentioned vowels to control the game. We use formant frequencies as features for the vowels and manage to get a decent classification rate of 85%, but we have too litle data to truly create a voice independet system that can classify vowels. Given more data you should be able to get the system up to at least 95%.

Contents

1 Introduction

The overarching goal for this project was to create a voice controlled application where you could use the sounds of long vowels to perform some simple tasks, i.e. the user will make a continuous sound of a vowel and you the use the recognized vowel to generate a signal to some application.

The application of choice is the game Sokoban. Sokoban allows the player to move up, down, left and right, as such we need to be able to distinguish four vowels from each other in order to be able to play the game of Sokoban. To win the game you have to push a set of boxes onto a set of goals but it's possible for the player to push a box so that they lock the game and can not win,

In order to play Sokoban we will also need to have a very precise recognizer, both in regards to eliminating noise as well as classifying vowels, since moving in the wrong direction may cause the player to lose the game by ending up in a deadlock.

2 Method

2.1 Features

Since we only wish to recognize vowels/sounds for our application we don't really have any need for lexical or language models so the focus will be on feature extraction and decoding of said features. A well studied feature for vowels is formants which we will use to classify sounds as vowels.

A formant can be described as the concentration of energy around around a frequency, i.e, the strength of the frequency [2], and the frequency with the highest energy is called the first formant, the second highest concentration of energy is called the second formant and so on. We will be using the first three formants for vowel recognition since the higher formants are more speaker dependent [2]).

Since we only need to select four vowels for recognition to play the game of Sokoban we can try and do so by looking on previous research about formants and try to make an initial guess about which four vowels should be easiest to build a classifier for.

Vowel	I	Y	U	O	E	Ö	Å	Ä	A
Phonetic	i	y	ʉ	u	e	ø	o	æ	ɒ

Table 1: The table shows which vowels the phonetic transcriptions in figure 1 corresponds to.

Figure 1 shows an image from a study about swedish vowels [1] and it shows how the first and second formant changes over time as the vowels

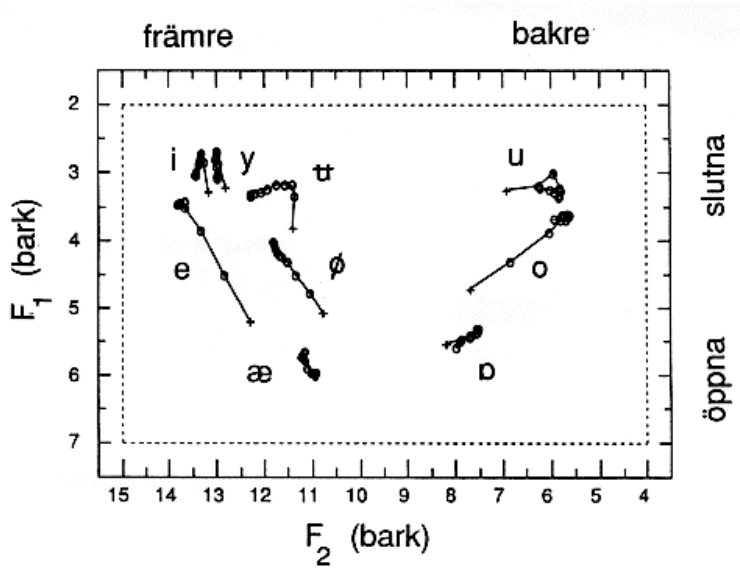

Figure 1: The image shows how the first two formants for Swedish vowels change over time. The image is from a study by Ingegerd Eklund and Hartmut Traunmüller [1]

are pronounced. In the study they find that vowels Ä and A isn't as prone to changes in formant frequencies while they are pronounced as the other vowels. While our recognition system isn't meant to classify pronunciation of vowels, i.e. just saying a vowel, but instead classify sound segments of a continuous sound signal, i.e. "sounding" a vowel. However, a reasonable guess is that the sound signal for A and Ä is more stationary in terms of formant frequencies than the other vowels and as such they should be easy to distingusih. The other two best candidates for vowel recognition ought to be I and O since they seem to be pretty stationary as well but are also far from A and Ä in the first and second formant as well as each other.

We use windows of a 100 milliseconds of captured audio from which we extract the mean formants from and use as feature vector for that audio segment. The larger the segments of captured audio the more robust our system should be to noise or intra speaker variation since such variation should get averaged out. Of course, the larger the time segment we capture and classify, the larger the delay will be for the user.

2.2 Classifier

To classify the vowels we will use a gaussian multivariate mixture model for each vowel. We will fit a mixture of gaussians to each vowels first three formants using the Expectation maximization algorithm. Our initial guess

for the mixture positions are estimated using the k-means algorithm and then we run the expectation maximization algorithm to optimize it. We're using a mixture of gaussians, instead of just one gaussian, to better be able to capture speaker variation between the vowels. Our gaussians only use diagonal covariance matrices to estimate the formants. Assuming that the formant frequencies are independent of each other is probably not true but by just using diagonal covariance matrices we should get better generalisation capabilities since we won't be as prone to overfitting our model on our training data as if we used full covariance matrices and since we don't have that much training data we need good generalisation if we wish our vowel recognition to be speaker independent.

Since the frequencies for the first three formants can range up to several thousand Hertz we need to scale down the values. Firstly because the initial k-means algorithm has to perform decently and secondly because we wish to avoid underflow errors. If we have covariance matrices with too large values the probability density functions will approach close to zero and we may get underflow errors.

Formant	i	e	y	ä	ö	u	a	å	o
F1	321	407	318,5	683,5	476,5	357	612,5	403	347
F2	2281	2326	2153,5	1695,5	1750,5	1791,5	957	695	678,5
F3	3317,5	2961,5	2936,5	2623,5	2531	2525,5	2719,5	2692	2543

Table 2: The table lists formant averages frequencies for Swedish vowels. The values are from a study by Ingegerd Eklund and Hartmut Traunmuller [1]

Formant	i	ä	a	o	Average (Hz)
F1	321	683,5	612,5	347	491
F2	2281	1695,5	957	678,5	1403
F3	3317,5	2623,5	2719,5	2543	2800,875

Table 3: The table lists formant averages frequencies for Swedish vowels I,Ä,A and O and their average. The values are from a study by Ingegerd Eklund and Hartmut Traunmulle [1]

Table 3 shows estimated means for the first three formants. Since we only need a rough scale down of the formants to get an initial decent estimate for k-means and just wish to scale down the formant vectors to avoid underflows we can apply a scaling vector of (1/500, 1/1400, 1/2800) to get the formants feature vector down into small values of similar range.

2.3 Noise reduction

To reduce the chance of recognizing noise as vowels we threshold input audio based on the probability distribution function value so that we ignore formant vectors that lie on the tails of the gaussian mixtures. The threshold values are manually set to eliminate just the most obvious cases and the threshold value depends on the probability density functions for the mixtures which in turn depends on how much we have scaled down the values for the formants. But since we only wish to eliminate the tails the exact choice of threshold the threshold value is not so important as long as it's not so large that we cut off areas where real data occurs.

2.4 Voice control

Even though we record and classify all input sound in 100 ms segments we can't send a control signal every 100 ms since that would make it very hard for the user to have any fine control over the character movements. So even though we generate a new vowel every 100 ms we only send a signal to the application if 500 ms have passed since the last signal was sent. We decrease the time between signals from 500 ms to 100 ms by a 100 ms each time the new signal is the same as the previous signal though so that the generated movement signals get sent faster and faster as vowels are detected to allow the user to still move the character quickly in one direction.

3 Results

The data we have on which we can evaluate the accuracy of our gaussian mixture model classifier on consists of 50 samples of each vowel from five different speakers, i.e. we have 1000 samples total. Each sample is a 100 millisecond recording of a vowel recorded using the same microphone and environment. In table 4 you can see the gender of each speaker. In table 5 you can see the percentage of correcly classified vowels when we do crossvalidation by training on all but one speaker and testing on the speaker we didn't train on.

Speaker	Sp1	Sp2	Sp3	Sp 4	Sp5
Gender	Male	Male	Female	Male	Male

Table 4: This table shows speaker gender since there's some variation among female and male speakers.

When using three formants, like we planned from the start, we find that we can't capture the speaker variation very well between the speakers as we increase the number of mixtures. In fact, using fewer mixtures is better for us if we look at the trend the percentage of correct classifications seems to

Mixtures	Sp1	Sp2	Sp3	Sp4	Sp5	Average
1	87%	76%	90%	76%	80%	81,8%
2	83%	72,5%	88,5%	75%	74,5%	78,7%
4	84,5%	70%	86,5%	89%	67%	79,4%
8	72,5%	72%	87%	78,5%	67%	75,4%

Table 5: The table shows the percentage of correctly classified vowels when we do cross validation in relation to how many mixtures we're using. The features used are the first three formants.

follow as can be seen in table 5. Most of the errors consists of confusing the vowel A with Ä or confusing O with A. An 80% classification rate, which we get as best on average, is clearly not good enough for a voice controlled software.

When training on all the data and evaluating on all the training data we get at least 90.8% correct for just one mixture though as seen in tabe 6. We should not be in any danger of overfitting the model to the training data, since we're using just one mixture as well as using diagonal covariance matrices over full covariance matrices, it's quite obvious that we lack enough data to create a speaker independent classifier, there's just too much variation between the speakers.

Mixtures	1	2	4	8
Correct	90,8%	95,1%	96,4%	97,4%

Table 6: The percentage of correct classifications when training on all the data and testing on all the data for three formants.

Since we can't create a speaker independent system just using the limited data we have we will try to reduce the number of formants we use for features from three to two. Having a lower dimensional feature space should reduce the speaker variation some. In table 7 you can see the percentage of correctly classified vowels when doing cross validation using two formants and in table 8 you can see the results of testing and training on all the data.

Mixtures	Sp1	Sp2	Sp3	Sp4	Sp5	Average
1	87%	76%	91,5%	92,5%	77,5%	84,9%
2	81,5%	80,5%	88,5%	88%	90,5%	85,8%
4	83,5%	83%	90%	92%	85%	86,7%
8	81%	79%	87,5%	91,5%	91,5%	86,1%

Table 7: The table shows the percentage of correctly classified vowels when we do cross validation in relation to how many mixtures we're using. The features used are the first two formants.

Mixtures	1	2	4	8
Correct	91%	95,6%	96,8%	97,5%

Table 8: The percentage of correct classifications when training on all the data and testing on all the data for two formants.

For the two first formants we don't have have as much speaker variation since we have a more stable average between the speakers instead of a decreasing average as seen when comparing table 5 and table 7. When comparing table 2 and table 4 we see that we don't really have any change in performance when testing on the training data so we probably don't need the third formant to distinguish between our four vowels among these speakers.

Next we test how well our thresholding of noise based on the probability density function of our gaussian mixtures work using the first two formants as our feature vectors. We do this by adding 200 samples of noise data to the test samples when doing cross validation, so we have 200 samples of vowels and 200 samples of noise for each cross validation. The noise consists of silence and speaker noise, such as throat clearing and lip smacking. The percentage of correct classifications of sounds as vowels and the percentage of incorrect classifications of sounds as noise, when using a threashold of 0.01 to cut of the tails, can be seen in table 9 and table 10 respectivly.

Mixtures	Sp1	Sp2	Sp3	Sp4	Sp5	Average
1	95%	95,2%	96,8%	94,8%	94,8%	95,32%
2	97,5%	98,5%	96,8%	97,6%	97,5%	97,58%
4	96,5%	97,5%	98,7%	94,2%	98,9%	97,16%
8	94,2%	97,5%	92,4%	93,8%	93,5%	94,28%

Table 9: The table shows the percentage of correct classifications of sounds as vowels, i.e. the precision.

Mixtures	Sp1	Sp2	Sp3	Sp4	Sp5	Average
1	4,5%	0,5%	20,7%	0%	0%	5,14%
2	2,5%	1%	20,4%	0%	2%	5,18%
4	3,5%	1%	20,5%	3,1%	5,3%	6,68%
8	3,1%	1%	22,6%	2,1%	0%	5,76%

Table 10: The table shows the percentage of incorrect classifications as noise, i.e. the percentage of the data classified as noise that is vowels.

When looking at table 10, which shows the percentage of vowels threashholded as noise, we clearly see that speaker three deviates from the other speakers in the number of vowels thresholded away as noise. The primary thresholded vowel is I where almost all of them are removed as noise. Since

speaker three is the only female speaker we apparently dont capture the vowel I just using gaussian mixtures fitted to male speaker data. In table 11 we can see the percentage of correct classifications of sounds as vowels and the percentage of incorrect classifications of sounds as noise when training and testing on all the data.

Mixture	1	2	4	8
Correct	94,79%	97,51%	98,95%	93,46%
Incorrect	0%	2,01%	5,26%	0%

Table 11: Table shows the percentage of correct classifications as vowels and the percentage of incorrect classifications as noise when training and testing on all the data.

4 Discussion

When selecting our four vowels to use for classification we chose them initially based on how distinguishable they were in the first and second formant, because of this the third formant became pretty much useless. However, if we would need more than four vowels wed probably need the third formant to distinguish more vowels, for example I and E are quite close in the first and second formant but differ in the third formant if you look at the means in table 2. As its now we dont have enough data to capture the third formant well and it just introduces errors in our classifier if we try to train a speaker independent classifier.

Threasholding the probability density function to remove noise works reasonably well. We had pretty much a 95% correct classification rate and threasholded away 5% of the vowels as noise. The 95% classification rate as vowels is good enough for our purpose and a 5% lose of vowels is also acceptable. That we had a 85% correct classification rate inbetween the vowels, when doing cross validation, is more troublesome since there is too much confusion between vowels.

However, looking at the test data in table 8 we should be able to build a speaker independent classifier if we just had more data. We were able to get 95% correct classifications when we just used two mixtures for five speakers so we should be in no danger of having overfitted our classifier to our data which suggests that with more data we should be able to reach higher than 85%.

5 Conclusions

To distinguish the vowels A, I, O and Ä we find that just using the first two formants works better than using the first three formants when you have

a small dataset to train on and you wish to have a speaker independent system. More data will probably not make the third formant more useful when trying to distinguish these four vowels, but it will become useful when trying to distinguish more vowels since some vowels overlap a lot in the first and second formant.

6 References

[1] Ingegerd Eklund and Hartmut Traunmüller. Comparative study of male and female whispered and phonated versions of the long vowels of swedish. Technical report, Institutionen for Lingvistik, Stockholms Universitet, 1996.

[2] Xuedong Huang, Alex Acero, and Hsiao-Wuen Hon. *Spoken Language Processing: A Guide to Theory, Algorithm and System Development.* Prentice Hall, Upper Saddle River, New Jersey 07458, 2001.