A Framework for Automatic Error Detection of Swedish Vowels Based on Audiovisual Data

SEBASTIEN PICARD

KTH Computer Science and Communication

Master of Science Thesis
Stockholm, Sweden 2010
A Framework for Automatic Error Detection of Swedish Vowels Based on Audiovisual Data

SEBASTIEN PICARD

Master’s Thesis in Speech Communication (30 ECTS credits)
School of Computer Science and Engineering
Royal Institute of Technology year 2010
Supervisors at CSC were Olov Engwall and Gopal Ananthakrishnan
Examiner was Björn Granström

TRITA-CSC-E 2010:027
ISRN-KTH/CSC/E--10/027--SE
ISSN-1653-5715
Abstract

Foreign learners of Swedish are usually experiencing difficulties when faced with the diversity of Swedish vowels. As part of the project Computer-Animated LAnguage TEAchers this thesis contributes to the creation of a framework for automatic detection of mispronunciation to be used in a Computer Assisted Language Learning (CALL) system. The aim is to provide users with informative feedback about their pronunciation.

This thesis introduces a general approach for binary classification based on audiovisual data. It was tested on vowels but is portable to any pair of phonemes. As a result it will be imbedded in an automatic mispronunciation detection system. The first contribution of this work is the use of a time-normalized 3D Discrete Cosine Transform (DCT) algorithm to extract visual features. Time-normalized DCTs are used to process vowels of different duration without the use of traditional Hidden Markov Models (HMMs) and generate visual feature vectors of constant length. Acoustic features of constant length were also obtained with a similar process, without resorting to the common use of Mel Frequency Cepstrum Coefficients (MFCCs). The resulting feature vectors of both modalities were subsequently concatenated and a combination of filter and wrapper feature selection methods was employed. This approach demonstrated a great ability to reduce the features to a subset suitable for classification. Support Vector Machines (SVMs) were used as classifiers and enabled the use of a sparse dataset.

We recorded an audiovisual database of two native and two non-native speakers with 118 words including all Swedish vowels in different contexts. This enabled comparison of the performance of different classifiers for each pair of vowels for samples using acoustic, visual and audiovisual features on a person-dependent framework. We concluded that the addition of visual cues contributed to improving the performance of the classifiers. We achieved 95 to 100% correct recognition rate for each pairwise classifier.
Preface

This Master’s thesis took place thanks to an exchange program between my home university ESCPE Lyon and KTH in fulfillment of the requirements for the obtention of my Engineering degree.

Although it was part of a project hosted by the Centre for Speech Technology, the main focus of this work concerns the extraction of visual features, audiovisual fusion and classification. Therefore it is assumed that the reader has basic background knowledge in the field of machine learning. Supplementary information may be found in Appendix 1 where a list of related textbooks is provided. On the other hand no background about speech processing is necessary for a clear understanding of this thesis.

I would like to thank the persons who contributed in any fashion to my coming to KTH for this 6-month stay. My special thanks go to Samer Al-Moubayed, who welcomed me as a friend and gave me useful advice about technical details concerning my work and practical matters, which made my life in Sweden more enjoyable. To Gopal Ananthakrishnan who indefatigably provided me with most of the technical background and ideas behind this work and guided me through my stay at KTH. To Olov Engwall, my supervisor at KTH, who helped me orchestrate my stay and made it possible for me to join this enriching project. To my girlfriend and family who have always been supportive and understanding at times I needed it most.
# Table of contents

I. Introduction ................................................................................................................ .. 1
   1. Context and motivations .......................................................................................... 1
      a) Difficulties of learning a foreign language and mispronunciation errors............ 1
      b) Computer Assisted Language Learning (CALL) systems .............................. 1
      c) The project Computer-Animated LAnguage TEAchers (CALATEA) ............... 2
   2. Audiovisual speech recognition ........................................................................... 2
      a) Bimodality of speech ....................................................................................... 2
      b) Shortcomings of auditory recognition and visual recognition ....................... 3
      c) Motivations of our approach ........................................................................... 3
   3. Overview of the thesis .......................................................................................... 4

II. Creation of a data set; recordings ........................................................................... 4
   1. Introduction......................................................................................................... 4
   2. Corpus ................................................................................................................ 4
   3. Settings ................................................................................................................ 5
   4. Alignment .......................................................................................................... 6
   5. Labeling ............................................................................................................. 7

III. Auditory features .................................................................................................... 8

IV. Visual features ....................................................................................................... 10
   1. Separation in different avi files and synchronization with wav files ................. 10
   2. ROI extraction ................................................................................................... 11
   3. Related work on image representation .............................................................. 12
      a) Appearance or low level features .................................................................. 12
      b) Shape or high level features ......................................................................... 13
      c) Joint appearance and shape features ............................................................. 14
   4. Our approach based on 3D DCT ....................................................................... 14
      a) Introduction .................................................................................................... 14
      b) 3D DCT ........................................................................................................... 15
      c) Determination of the parameter numbers \((N_x, N_y, N_t)\) .............................. 16
      d) Normalization schemes ............................................................................... 18

V. Classification by SVMs .......................................................................................... 21
   1. Introduction....................................................................................................... 21

VI. Combination of the audio and video features ....................................................... 23
   1. Introduction....................................................................................................... 23
   2. Presentation of different feature extraction and feature selection techniques ...... 24
      a) PCA ................................................................................................................. 24
      b) GAs .................................................................................................................. 25
      c) Filters and Wrappers ...................................................................................... 25
      d) MRMR ........................................................................................................... 25
      a) PCA based feature extraction ....................................................................... 26
      b) MRMR based feature selection ..................................................................... 27
      c) Improvement of the method by means of GA .............................................. 29

VII. Results ................................................................................................................... 31
    1. PCA-based feature extraction ........................................................................... 31
Table of figures

FIGURE 1 HISTOGRAM OF THE LENGTH OF THE VOWELS .................................................. 3
FIGURE 2 ORIGINAL VIDEO FRAMES ............................................................................. 5
FIGURE 3 PHONEME ALIGNMENT .................................................................................. 7
FIGURE 4 GUI TO GRADE THE QUALITY OF THE PRONUNCIATION OF VOWELS FROM NON-NATIVE SPEAKERS ........................................................................... 8
FIGURE 5 EXTRACTION OF AUDITORY FEATURES .......................................................... 9
FIGURE 6 EXTRACTION OF VIDEO FRAMES ................................................................... 10
FIGURE 7 ROI EXTRACTION .......................................................................................... 11
FIGURE 8 IMAGE FRAMES CONSTITUTING A VOWEL .................................................... 12
FIGURE 9 TIME NORMALIZED 3D DCT ALGORITHM ......................................................... 16
FIGURE 10 FAST ALGORITHM TO SELECT (N_x, N_y) ...................................................... 18
FIGURE 11 NORMALIZATION OF THE FIRST DCT COEFFICIENT ................................... 19
FIGURE 12 NORMALIZATION FOR THE FIRST K^K IN THE FIRST “TIME” FRAME ............ 19
FIGURE 13 EXTRACTION OF THE VISUAL FEATURES FROM IMAGE SEQUENCES ........ 20
FIGURE 14 GENERAL CLASSIFIER ............................................................................... 21
FIGURE 15 SAMPLE NUMBERS BY VOWEL CLASS .......................................................... 22
FIGURE 16 IMPACT OF THE TUNING PARAMETER (SIGMA) FOR RBF KERNEL ......... 23
FIGURE 17 FEATURE EXTRACTION BASED ON PCA FOR CONCATENATED DATA .......... 26
FIGURE 18 DIMENSIONALITY REDUCTION USING TWO LAYERS OF PCA ................. 26
FIGURE 19 MMR R FEATURE SELECTION .................................................................... 28
FIGURE 20 GENETIC ALGORITHM SCHEME ................................................................. 29
FIGURE 21 CREATION OF PAIR-WISE CLASSIFIER WITH EMBEDDED FEATURE SELECTION BY MEANS OF MMR AND GA ............................................................. 30
FIGURE 22 FORMAT FOR RESULT TABLES ..................................................................... 31
FIGURE 23 RESULT SAMPLES, EXTRACT OF RESULT TABLE 1 ...................................... 31
FIGURE 24 RESULT SAMPLES, EXTRACT OF RESULT TABLE 2 ...................................... 32
FIGURE 25 RESULT SAMPLES, EXTRACT OF RESULT TABLE 4 ...................................... 33
FIGURE 26 RESULT SAMPLES, EXTRACT OF RESULT TABLE 5 ...................................... 33
FIGURE 27 RESULT SAMPLES, EXTRACT OF RESULT TABLE 6 ...................................... 34
FIGURE 28 RESULT SAMPLES, EXTRACT OF RESULT TABLE 7 ...................................... 35
I. Introduction

1. Context and motivations

a) Difficulties of learning a foreign language and mispronunciation errors

Human beings have developed thousands of languages (Lewis, 2009) in constant evolution for more than 100,000 years (Johansson, 2006). Over the course of the 20th century, people have become increasingly inclined to learning foreign languages and continually encounter difficulties, which can vary greatly with the different (L1-L2) language pairs involved. The common origin for certain families of languages accounts for their similarities. Conversely the cleavage of a language into different families or the cultural isolation over the millennia can explain the great discrepancies between certain pairs of languages.

Although anyone can learn to produce the phonemes associated to one's mother tongue, it can be difficult to assimilate the subtleties of a foreign language, especially when this language presents phonemes absent in one's mother tongue. During childhood, we gradually form our brains and grow used to hearing only one language. As a result we lose the ability to speak and learn other languages (Yang, 2006). Learners whose mother tongue makes no distinctions between long and short vowels (e.g. French) will find it complicated to learn languages where the length of vowels plays an important role (e.g. Swedish). For some languages there is a high degree of correspondence between written representation and phonemes or sounds. Conversely in other cases the same combination of characters can lead to a multitude of different sounds and vice versa. This can prove to be problematic as in many cases one learns foreign languages by relying heavily on the written representation, which only confuses the learner. Other major differences between languages, such as the presence or absence of tones, the differences in prosody or the complexity of the grammar factor all contribute to making learning a foreign language a demanding task.

Learning a foreign language requires learners to make abstraction of their native tongue's idiosyncrasies and learn how to produce the phonemes proper to the target language. The difficulty of the task can be overwhelming when learners have developed flawed pronunciation habits through years of repetitive uncorrected mispronunciations. Moreover when learners have not familiarized themselves with foreign languages at a young age, the utterance of unknown phonemes, which necessitate novel pronunciation gestures, can prove to be unachievable. This can discourage the learners whose speech could even be unintelligible for the native speakers. Unfortunately the learning process is rarely accompanied by a strict and formative control of the pronunciation which could assist learners from the novice level. In some countries the speaking ability is utterly neglected during language classes and little effort is demanded from the learners along with the instructors.

b) Computer Assisted Language Learning (CALL) systems

CALL systems appeared in the 1960s and were primarily used by language laboratories in universities (Taylor, 1980) and are the subject of recent studies (Doremalen et al., 2009). The concept of assisting the learners by providing tailored feedback delivered by a tireless machine is appealing. However CALL has not encountered notable success in academia. Indeed the systems should be able to deal with all the aspects of the language, namely speaking, listening, writing and reading in order to be fully interactive and benefit the users. The constant advance in computer technology has not yet allowed for effective CALL systems, especially automatic and personalized speech analysis and feedback are far from being comparable to those of a human teacher. Moreover the language teaching community, who is not sufficiently familiar with computer technologies, is generally not inclined to see the benefit of CALL.
Therefore the advantages of CALL cannot overcome its high cost and the performances of the most state-
of-the-art systems cannot compete yet with the expectations of the consumers. Alternatively users may be
more seduced by the expanding language exchange social networks where one can receive personalized
feedback by native speakers. Ongoing research and recent progress in Automatic Speech Recognition
(ASR) could be decisive in making CALL widely popular.

c) The project Computer-Animated LAnguage TEAchers (CALATEA)

Several language learning softwares incorporate speech recognition technologies. However, the learners are
only given approximate scores and no information about the quality of the sounds produced is provided by
any software to date. The project CALATEA aims at bridging the gap between individualized teaching by
native speakers and impersonal learning methods. As we learn our mother tongue, constant feedback about
our speech abilities are available; therefore children manage to acquire the language. On the other hand
learning another language as an adult is fundamentally different insofar as one can hardly be immersed in a
foreign language environment and therefore no feedback is available. As a result, we tend to develop
flawed pronunciation habits, which constitute a major obstacle for language learning. Consequently, in
order to provide learners with a language learning method of higher quality, detailed feedback about the
quality of the pronunciation is targeted in this project.

The detection of errors will be enabled by combining theoretical linguistics knowledge, speech analysis and
machine learning algorithms. The purpose is to create a general classifier able to distinguish whether
sample vectors belong to one class or another. Those classes are defined from the linguistics point of view
and typically include correct pronunciation samples from native speakers in one class and mispronounced
samples in the other class. Then other speech analysis methods are to be applied in order to decide when to
use the classifiers so that errors are detected automatically. The goal of this Master’s thesis, as part of
CALATEA, is to design an algorithm able to create a 2-class classifier used for automatic detection of
mispronounced Swedish vowels, based on audiovisual data. The remaining parts of the project are not
investigated further in this thesis.

2. Audiovisual speech recognition

a) Bimodality of speech

Human speech is intrinsically bimodal. Visual information about the speaker’s face can greatly aid speech
intelligibility in presence of noise (Sumby & Pollack, 1954). It has been shown (Chen, 1998) that human
lip reading could benefit from the integration of video cues and in some cases the video channel can even
carry more information that the audio channel. However there is no simple mapping between phonemes and
visemes. For instance English speech comprises 14 visemes and 48 phonemes (Sumby & Pollack, 1954; Rabiner & Juang, 1993) /m/, /b/ and /p/ are grouped in the same viseme and cannot be visually
distinguished. On the other hand some cases of acoustic confusion (eg /k/ and /p/ sometimes hard to
distinguish with audio only) can be distinguished when using the visual modality (Chen, 1998).

Similarly one can easily understand the bimodality of speech production. The arrangement of the different
speech articulators results in speech production together with facial expressions (Chen, 2001). McGurk &
MacDonald (1976) showed that contradictory information from the audio and video channels led to
confused perception, which emphasized the bimodality of speech perception. For instance when the audio
produces /ba/ and the video /ga/ we perceive /da/, which was produced by neither of the speech modalities.
b) Shortcomings of auditory recognition and visual recognition

There are generally more phonemes than visemes in a given language and it can be the case that several different phonemes correspond to a single viseme. As a result one should not expect to obtain state-of-the-art performance with Automatic Speech Recognition (ASR) systems relying solely on visual features. On the other hand ASR based solely on auditory features is renowned for its vulnerability to noisy environments. This may in particular be a problem for ASR of non-native speech, which we are dealing with in this work. Several approaches aiming at tackling the problem with audio-only frameworks exist: noise filtering before classification (Juang, 1991), model adaptation to noise (Gales & Young, 1992; Nadas et al., 1989) and use of features robust to noise (Ghitza, 1986; Hermansky & Morgan, 1994). However they prove to be insufficient. This fact along with the knowledge that speech is bimodal triggered the interest in visual features to incorporate as an aid in an audiovisual framework.

ASR aided by video features outperforms audio-only in presence of noise is many cases (Dupont & Luettin, 2000; Potamianos & Graf, 1998; Teissier et al., 1999; Xie et al., 2003) and demonstrated on noise-free environments for the first time by Neti et al. (2000). There is no common agreement on what video features should be to achieve optimal classification results on audiovisual systems. Current techniques include three commonly used classes of video features: high level statistical models representing the lips (Adjoudani & Benoit, 1996; Petajan, 1984; Teissier et al., 1999; Kass et al., 1988; Chiu & Hwang, 1997; Cootes et al., 1995), low level region of interest (ROI) transformations (Duchnowski et al., 1994; Chiu & Hwang, 1997; Matthews et al., 1996; Neti et al., 2000; Potamianos & Graf, 1998), and combination of both (Dupont & Luettin, 2000; Luettin et al., 1996; Hennecke et al., 1996; Edwards et al., 1994; Cootes et al., 2001). Once audio and video features have been obtained, audiovisual integration is a crucial task. The purpose is to achieve better classification results than the best single modality stream. This field of research is subdivided into two categories of methods. Feature fusion concatenates audio and video vectors and after possible transformation trains a single bimodal classifier. Decision fusion on the other hand uses the output of two mono-modal classifiers. Combination of those methods can yield superior results (Potamianos, 2003).

c) Motivations of our approach

Our initial goal is to use a framework based on Support Vector Machines (SVMs), as it is the best classifier for sparse data. This is an important feature for a classifier of non-native pronunciation since training data may be sparse due to different native language backgrounds of the speakers. The use of SVMs requires feature vectors of fixed duration. However the extracted files, from both auditory and visual channels (cf. Figure 1), exhibit conspicuous differences in length.

![Figure 1](image_url)  
**Figure 1** Histogram of the length of the vowels  
(a) **acoustic files**, (b) **video files**
Some methods involve hybrid SVM-HMM models (Fine et al., 2001) while others use SVMs on each individual frame (Wei et al., 2009). Our method differs from those methods insofar as we normalize in time the feature vectors by means of (Discrete Cosine Transforms) DCT to produce feature vectors of fixed length which are therefore suitable for classification via SVMs.

3. Overview of the thesis

In this thesis we start by introducing our database (section II) and all the preprocessing steps associated to the collection of data. The acquisition of auditory features (section III) is treated only as a small section as the core of this thesis relies more on the other aspects of audiovisual speech recognition. We choose to consider an audiovisual framework for this study as we believe that audio and video modalities will complement each other and enable results unattainable with methods based on auditory- or visual-only modalities. In section IV we provide an overview of the most common approaches for extraction of visual features (section IV.3) and describe our strategy with regards to the visual front-end (section IV.4). The integration of auditory and visual modalities is discussed in section VI where a discussion about different dimensionality reduction techniques is also given (section VI.2). Section V details the classification strategy. Results and conclusions are to be found in sections VII and VIII.

II. Creation of a data set; recordings

1. Introduction

A database including audio-video recordings of native Swedish speakers is required to train the audiovisual classifiers. As we want to estimate the performance of the classification method also on incorrect pronunciations, the database should also contain utterances from non-native speakers. Moreover the objective is to target vowels, so we need a corpus which reflects the diversity and variation of the vowels in different contexts. Overall our task is very specific and it seems delusory to find an existing database exhibiting all the features we require, so we decided to record our own database.

2. Corpus

A corpus consisting of 118 Swedish one- or two-syllabic words in minimal word pairs\(^\text{A}\) was selected (cf. Table 1). This corpus was chosen so as to comprise all 18 vowels, long and short included, present in the Swedish language. Although a subset corresponds either to dialect or archaic terms, all words are existing Swedish words listed in the electronic version of Svenska Akademiens Ordbok at http://g3.spraakdata.gu.se/saob/. Empty slots in Table 1 correspond to non-existent words. Throughout the thesis we do not use the traditional International Phonetic Association (IPA) phonetic transcription but instead adopt the RULSYS (Carlson et al., 1988) notation used at the bottom of Table 1.

In this study we chose to focus only on vowels and we did so for several reasons. From a practical point of view, this framework will be used in an automatic mispronunciation detector for foreign speakers of Swedish. The Swedish language exhibits a wide variety of vowels comparatively to most other languages and this poses a problem for many foreign learners. Additionally, we aim at proving the usefulness of the visual modality and a clear distinction between rounded and unrounded lips is displayed for vowels. Moreover the auditory signals are more continuous for vowels and overall the vowels are more homogeneous, which make them in principle easier to process than consonants. The same framework could however also be applied to consonants.

\(^\text{A}\) Minimal pair denotes a pair of words differing in only one phoneme or other phonological element.
Table 1: Corpus of words grouped by vowels categories and base words

<table>
<thead>
<tr>
<th>Vowels</th>
<th>Word Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>rita</td>
<td>ryta reta räta röta rata råta (råda) rota ruta</td>
</tr>
<tr>
<td>ritt</td>
<td>rätt rött ratt rätt rott rutt</td>
</tr>
<tr>
<td>rist</td>
<td>rest röst rast rost rust</td>
</tr>
<tr>
<td>silar</td>
<td>sylar sälar sölar salar sålar solar sular</td>
</tr>
<tr>
<td>sill</td>
<td>säll sall såll sull</td>
</tr>
<tr>
<td>mys</td>
<td>mes mös mas mås mos mus</td>
</tr>
<tr>
<td>missa</td>
<td>myssa messa måssa mossa massa mossa mousse missa</td>
</tr>
<tr>
<td>lida</td>
<td>lyda leda löda lada låda loda luda</td>
</tr>
<tr>
<td>lydda</td>
<td>ledda ladda lädda ladda ladda ludda</td>
</tr>
<tr>
<td>lin</td>
<td>lyn len län lön LAN lån lon lun</td>
</tr>
<tr>
<td>fira</td>
<td>fyra föra förä förä förä fora fura</td>
</tr>
<tr>
<td>hitta</td>
<td>hyta hetta hätta hötta hatta hotta hutta</td>
</tr>
<tr>
<td>ni</td>
<td>ny nää nää nu</td>
</tr>
<tr>
<td>bida</td>
<td>Beda Böda bada båda Boda buda</td>
</tr>
<tr>
<td>bita</td>
<td>byta bota båta bota</td>
</tr>
</tbody>
</table>


3. Settings

The recording was performed in a soundproof recording booth in a noise-free environment. Audio waveforms were recorded with a high quality omnidirectional microphone (Haun MBNM 550 E-L [http://www.mbho.de/pdf/mbnm550el.pdf]) using a recording application based on the software Snack ([http://www.speech.kth.se/snack](http://www.speech.kth.se/snack)). It was decided to record high quality audio as this corpus may be used for other studies requiring this degree of quality. However in our study it is not of the utmost importance. It would also be possible to add noise to the audio to create a more realistic environment; but such experiments are not part of this study.

Most studies in the literature focused on audiovisual ASR in noisy environment. In our case the high quality of the sound is expected to render the error detection easier for the procedure based on the auditory features only. This will make it more difficult to notice improvement from the addition of video features to the auditory mono-modal framework. Moreover the quality of the video recordings is far from optimal. Indeed the face of the subject only accounts for a small proportion of the image (cf. Figure 2). As it will be explained in section IV.2 the region of interest consists of the area around the lips, which further reduces the resolution of the images. Therefore we anticipate the extraction of meaningful video features to be a delicate task. On the other hand it has the advantage that there should not be any significant drop for the results in real-life environment compared to the results obtained from our method. The properties of the video are such that it will enable other studies based on the same database and approximately recreates what a webcam would capture.

Figure 2 Original video frames
Image frames extracted from the video files for the four subjects of the database.
Initially four persons (cf. Figure 2) were gathered for the recordings: two native and two non-native speakers. Words were displayed in a predefined randomized order, which was independent of the target vowels. This was used in order not to influence the pronunciation of the non-native speakers. Therefore it was ensured that a significant quantity of mispronounced vowels was included in the database.

The experiment was conducted as follows for the four subjects:

1) For each word of the sequence:
   11) Subject reads the word displayed on a screen.
   12) Subject raises the clapperboard at face level, claps it, and subsequently lowers it back.
   13) Subject utters the word.
   14) Wave file is saved in a separate file.

2) Video is recorded continually along the process.

To increase the amount of correct data from the non-native speakers, the previous recording was repeated and altered for the two non-native speakers:

1) For each word of the sequence:
   11a) Subject reads the word displayed on a screen.
   11b) Subject hears the word uttered by a native speaker.
   12) Subject raises the clapperboard at face level, claps it, and subsequently lowers it back.
   13) Subject utters the word.
   14) Wave file is saved in a separate file.

2) Video is recorded continually along the process.

At first we decided to restrict the number of subjects in the database for practical reasons. Indeed it took a lot of manual processing to extract the video files corresponding to each word from the whole video obtained in 2).

Action #12) is performed for later synchronization between wave files and video files. Each wave file starts as the word is displayed on screen and the clap triggers a distinctive spike in the wave signal, which is also detected by the audio channel of the video recorder. Synchronization is performed by detecting the spike corresponding to the clap in both signals.

In the second recording, the non-native speakers hear the correct pronunciation and try to reproduce it. There are two fundamentally different sorts of mispronunciations. At beginner level it is very common to encounter mispronunciation errors induced by reading. More precisely non-native speakers tend to apply the same text-to-phonemes rules they have learned for their mother tongue when they encounter similar transcriptions. For example, /y/ is pronounced as /i/ in French and an inexperienced French learner would easily read /y/ as /i/ each time /y/ is encountered. To remedy this problem in this experiment, action #11a) can be skipped in favor of action #11b) only. The second sort of mispronunciation is when non-native speakers are unable to produce a phoneme of the target language and instead produce a phoneme inexistent in the target language.

In this thesis, we do not deal with pronunciation errors involving non-Swedish phonemes. Indeed in our database there are only two non-native speakers who are from different language background. Therefore it is not possible to gather sufficient data representing non-Swedish vowels from our database. However future work focusing on typical errors in pronunciation of a certain first language will emphasize on the detection of non-native pronunciations. This could be done using the proposed framework without any alteration.

4. Alignment

As we are only interested in processing the first vowel of each word, we need an automatic way to constrain the auditory and visual information to represent only the vowel. NALIGN (Sjölander, 2003; Sjölander & Heldner, 2004), a segmentation algorithm relying on Hidden Markov Models (HMMs) and
capable of automatically creating time-aligned transcription at phone-level for spoken Swedish, was used for this matter (cf. Figure 3).

![Figure 3 Phoneme alignment](image)

**Figure 3 Phoneme alignment**

*a) Original wav file with spike at the clap (dashed red line)*

*b) Segmented wav file after running NALIGN to obtain time segments.*

*Dashed red lines represent segment of phonemes, green lines represent starting and ending time of the first vowel*

For each wave file, the aligner gave the starting and ending time of the vowel, and after synchronization with the video file, we could also extract the corresponding video frames. However this is done after obtaining the ROI. Indeed the tracking algorithm has a latency of a few frames before it can track efficiently the face. Moreover it has proved to perform better then the starting frames of the sequence display unobstructed faces, which leads us to consider starting the tracking before the clapperboard is appearing on the frames.

As a measure of precaution it was chosen to add two additional frames, corresponding respectively to the frame immediately before the starting time and immediately after the ending time.

The purpose of this addition is twofold. First, the shortest image sequence only comprises two frames, which is not enough to allow compression in the temporal dimension. Second, there were possibly radical approximations involved when converting times from wav files sampled at 16 kHz to images at 25fps in the synchronization process. Addition of a greater number of frames could be detrimental for our task as those frames will not represent the target vowels and therefore only contribute to adding intra-class discrepancies during the classification process.

## 5. Labeling

The next step was to assess the quality of the pronunciation of the non-native speakers. This was done using a Graphical User Interface (GUI) developed specifically for the task (cf. Figure 4). The purpose of this simple GUI is to grade the pronunciation of a series of words. Although it would be possible to envision different configurations, it was chosen to be used only for the first vowel of each word and only four grades were considered.

Each word was displayed and played simultaneously and the pronunciation from native speakers was also accessible to enable comparison. Native speakers were asked to use the interface and assign a label to the first vowel of each word by clicking one of the four buttons on the bottom left (cf. Figure 4). CORRECT
(respectively **WRONG**) is chosen when there is no doubt that the vowel is correctly (respectively wrongly) uttered. The two buttons in the middle were used in case of doubt.

![GUI to grade the quality of the pronunciation of vowels from non-native speakers.](image)

*Figure 4* GUI to grade the quality of the pronunciation of vowels from non-native speakers. Buttons on the left assign a label to each vowel, and it is saved in a .mat file. Each word is played and displayed on the top right. The user has the possibility to replay and to compare any non-native speaker’s utterance with the one from native speakers. It is also possible to go back to the previous utterance at any time.

Labels collected from different graders constitute ground truth, which will be used for classification. Careful labeling is therefore an important part of the process; however the introduction of several levels can be confusing. As a result it was decided to confront the labels from different graders. In the experiments presented in this thesis, labels obtained from two graders were used. Vowels deemed correctly pronounced by both graders were considered and the rest were discarded. This process ensures that no contradiction in the labeling occurs, decreasing the proportion of sample belonging to the wrong class. Needless to say this is of paramount importance. However this process at the same time decreases the size of the training set. The remaining words corresponding to wrongly pronounced vowels in our corpus can be used for subsequent experiments on mispronunciation detection, yet this was not attempted in this work.

### III. Auditory features

In this section we briefly describe our strategy for the extraction of auditory features. As the main focus of this thesis is the visual front end and the integration of auditory and visual features, we do not expand greatly on this section.

Over several decades many studies have been focused on the choice of auditory features and Mel Frequency Cepstral Coefficients (MFCCs) (Mermerstein, 1976) are regarded as a choice of predilection as state-of-the-art auditory features for ASR. MFCCs are perceptually motivated, which means that it approximates the human auditory response. Many audiovisual ASR systems use MFCCs (Potamianos, 2003; Xie et al., 2003) as auditory features.

The auditory features are extracted by means of time-varying filter-banks based on the Equivalent Rectangular Band-width (ERB) scale (Moore & Glasberg, 1983). The ERB scale presents the same advantages as the Mel scale, as it is closely related to human perception, since it approximates the bandwidth of the filters in human hearing as MFCCs. The main difference of our method is the ability to compute 2D-cepstra for segments of different duration and to produce features of constant duration after post-processing via time-normalized 2D DCT.
The $k^{th}$ spectral component for the transform of the time signal $x(t)$: $1 \leq t \leq T$ ($T$ is the duration of the time signal) at sample frequency $F_s$ is $X(k,t)$ given by the following equations (1a, 1b, 1c) where $f$ is the frequency (Hz), $B_W$ is the bandwidth and $K$ is the total number of filters.

$$B_W = 6.23 \times 10^{-6} \cdot f^2 + 9.339 \times 10^{-2} \cdot f + 28.52$$ (1a)

$$W_k(t) = H(t) \cdot \sin \left( \frac{(t - (L(k)/2))B_W(k)}{F_s} \right)$$ (1b)

$$X(k,t) = \sum_{m=1}^{L(k)} W_k(m)x(t-m)\exp \left( -\frac{j2\pi C_f(k)}{F_s} \right)$$ (1c)

$$Y(k,t) = 10\log_{10}(|X(k,t)|^2)$$ (1d)

The window functions $W_k$ are Finite Impulse Response linear phase low pass filters. Their central frequencies $C_f(K)$ are calculated by dividing the ERB scale into $K$ equal parts with the constraint that $C_f(K)$ is not greater than $F_s/2$. The order $L(k)$ is calculated as $L(k) = 2 / B_W(k)$ and $H(n)$ is the Hann window.

The complex signal $X(k,t)$ is converted to a real signal and compressed using log scale approximation of loudness (1d).

For each vowel we compute $Y(k,t)$ for $0 \leq k \leq 45$ and $1 \leq t \leq T$ and obtain 2-dimensional frames with frequency on the x-axis and time on the y-axis (cf. Figure 5). In order to normalize over time we apply 2D DCT (cf. explanations about 3D DCT detailed in section IV 4 b). The whole process about extracting the auditory features from the vowels is detailed in Figure 5.

**Figure 5** Extraction of auditory features

*Time signal and spectral components before DCT are varying in duration.*
For our experiments we use 18 frequency coefficients and 3 time coefficients. Those values have been determined in another study (Ananthakrishnan & Engwall, 2010) which uses the same framework for auditory features.

The first coefficient of the DCT corresponds to the sum over all coefficients of \( Y \) (cf. explanations given in section IV 4 b) so we apply normalization by discarding this coefficient. Also we expect that the length of the vowel may be meaningful for the distinction between long and short vowels. The resulting auditory features for our experiments consist of 54 cepstral coefficients and 1 coefficient corresponding to the duration of the vowel.

IV. Visual features

As previously seen, the fewer number of visemes compared to phonemes exclude the possibility of accurate video-only lip-reading systems. Yet, the use of pertinent video features can be important when it comes to improving the performance of ASR systems. In this section we present the different approaches often considered in the literature and detail our strategy for the visual front-end. The first two subsections respectively deal with practicalities and preprocessing steps.

1. Separation in different avi files and synchronization with wav files

Avi files were manually extracted from the whole video based on the movement of the clapperboard. As the video recording is continuous, it is very helpful to detect when the clapperboard appears so that the video can be manually separated in avi files from the original recording. For each word, frames, comprised from the moment the clapperboard is starting to be raised to the moment the lips have stabilized after utterance, are stored. This corresponds to Figure 6 (b) to (h).

![Figure 6 Extraction of video frames](image)

Selection of frames (res. 720*576) extracted from the raw video for “lön”  
(a) Subject is reading input text. (b) Clapperboard is being raised.  
(c,d) Clapperboard is clapped. (e) Clapperboard is being lowered  
(f) Subject starts uttering word. (g) Vowel utterance. (h) Re-stabilization of the lips

The low video frame rate together with the latency in the movement of the subject makes it impossible to determine precisely when the clap occurs. Indeed several consecutive frames display the clapperboard completely closed as in Figure 6 (d) and (e), whereas some images are blurred because of the quick movement of the clapperboard (c). Furthermore, when one transposes the vowel length to video frame, the
longest sequence corresponds to 10 frames, overruling the possibility to locate the clap in the video channel. Instead we used the audio channel of the avi recordings to extract the clap position and report the corresponding time in the video channel. After synchronization with the wav file, corresponding starting and ending times were obtained for the video sequence. This is done before extraction of the ROI, yet the frame numbers delimiting the vowel are only used after ROI extraction.

2. ROI extraction

Visual features extraction generally requires pre-processing steps, namely face detection and ROI extraction. The robustness of the process will affect the quality of the whole system, and therefore it is considered an essential step (Iyengar et al., 2001). Common methods aim at extracting the entire face or only the area of the lips, alternatively visual information restricted to a rectangle above the nose, as suggested in (Davis & Kim, 2004), could carry useful information for ASR systems. (Potamianos, 2003) extracts a rectangle centered at the center of the mouth, which extends to the jaw and cheeks.

Some face detection algorithms make use of image processing techniques, such as color segmentation or template matching (Graf et al., 1997). Statistical models, such as likelihood tests (Schneiderman & Kanade, 2000) and neural networks (Rowley et al., 1998) along with combination of image processing and neural networks are also used (Lin et al., 2005). Potamianos (2003) uses neural networks to estimate the proportion of skin-tone pixels among candidate areas. Once the face is detected, statistical knowledge about the distribution of the facial features detectors makes it possible to locate several facial features, such as the corners of the lips. When the lips area is targeted, it is often obtained after location of the face. Cisař et al. (2004) use control points drawn on the faces in the training set. The control points are extracted by image processing and the 3D model obtained via stereo-vision. The ROI extraction process is based on image processing and histograms thresholding using knowledge from the training set. Lucey et al. (2005) achieve ROI extraction through a three-step process: face location, eye location and lip location. (Potamianos, 2003) also uses different steps of image processing on the output of the face detection algorithm.

Our goal is to extract a rectangular area encompassing the lips so that further processing of this area yield informative features. Among the diversity of tracking algorithms available, we shall choose one which conforms to the constraints and takes advantage of the characteristics of our setup.

Figure 6 shows how the subject is facing the camera with little head movement, however the target is not always accessible (c, d, e). Moreover we aim at extracting visual features from sequences of no more than 10 frames, emphasizing the importance of stability. Therefore we need an algorithm privileging tracking accuracy and automatic re-initialization over sensitivity to swift movement. The algorithm described in (Kjellström & Engwall, 2009) has been developed for extraction of stabilized images of the lips, and
therefore meets our needs. This algorithm tracks the upper part of the face using a particle filter. The limited head movement and the constant focus on the computer screen from the target subject allows for a 2D based template method. The mouth area which is the most deformable part of the face is deduced from the obtained position of the upper part of the face. The implementation of this algorithm from the same author was used for our ROI extraction.

The algorithm was run for each image sequence starting before appearance of the clapperboard. This was done to ensure that the initial tracking did not occur when the face was occluded by the clapperboard. Figure 7 displays the output of the algorithm for the same images shown in Figure 6.

![Image frames constituting a vowel](image)

**Figure 8** Image frames constituting a vowel

Frames corresponding to the vowel in “lön”

Using the information about the starting and ending time for the vowel, the frames corresponding to the vowel are selected. Figure 8 displays the entire frame sequence for the vowel in “lön”

### 3. Related work on image representation

**a) Appearance or low level features**

Appearance- or low-level-based methods suggest that every pixel of the ROI carries meaningful information. The simplest approach is to vectorize the pixel matrix of the ROI and directly use the result as feature vectors (Duchnowski et al., 1994; Chiou & Hwang, 1997). However the dimensionality of the resulting vector is foreboding, especially when the ROI extends over the sole lip area by adding the cheeks (Potamianos et al., 2000) or even the entire face (Matthews et al., 2001). In most cases features are obtained from direct transformation of the pixel values. Such transformations include datasieve (Matthews et al., 1996; Bangham et al., 1996) as in (Xie et al., 2003), where a recursive algorithm is used to obtain 1D vertical datasieves. Feature extraction is then used to produce the visual features and linear interpolation to upsample to audio sample rate.

Mostly traditional image transforms are used instead. Principal Component Analysis (PCA) (discussed in VI.2) is often considered as it optimizes the least square error between the original vectors and the reconstruction from the projected ones (Bregler & Konig, 1994; Chiou & Hwang, 1997; Duchnowski et al., 1994). PCA-based eigenfaces (Sirovich & Kirby, 1987), which emerged as a landmark method in the field of face recognition (Belhumeur et al., 1997; Heseltine et al., 2003; Turk & Pentland, 1991), have also been used in audiovisual ASR methods (Bregler & Konig, 1994; Li et al., 1995). Similar to PCA, Linear Discriminant Analysis (LDA) (Fisher, 1936) linearly projects the image space on a lower feature space. LDA maximizes the inter-class variance and minimizes the intra-class variance and is more suitable for subsequent classification as it is class specific. It was first used in pattern recognition (Duda & Hart, 1973), face recognition (Belhumeur et al., 1997) and automatic speechreading (Duchnowski et al., 1994; Martinez & Kak, 2001; Matthews et al., 2001; Potamianos et al., 2000). In (Belhumeur et al., 1997) LDA-based fisherface (which consists of decomposing an image in elemental masks given by LDA) outperforms
eigenface in all experiments; however (Chibelushi et al., 2002) found that no significant difference in speech recognition was achieved between methods using PCA and LDA visual features.

Image transforms such as Discrete Cosine Transforms (DCT) (Ahmed et al., 1974) or Discrete Wavelet Transforms (DWT) are also used as a substitute to PCA and LDA. DCT is closely related to the Discrete Fourier Transform (DFT). It decorrelates the features and exhibits energy compaction. Fast algorithms based on separability have been widely studied (Aggarwal & Gajski, 1998; Hung & Meng, 1994; Haque, 1985) and enable the use of 2D-DCT for audiovisual ASR systems (Duchnowski et al., 1994; Neti et al., 2000; Potamianos & Graf, 1998).

However, appearance video features are considered to be ineffective in extracting speech content by certain studies (Lucey et al., 2005). In that paper, subtracting the mean image helps removing the static speaker information and greater speech intelligibility is achieved. Use of free-parts (Lucey, 2004) helped remedy the reliance of area-based methods on the video front-end, especially the ROI extraction.

b) Shape or high level features

Shape- or high-level-based methods imply that the contours of the lips contain most of the information. There are two main categories of such methods, geometric and model features. Geometric features are obtained from a given lip contour and postulates that visible parameters such as the mouth aperture, its length or the thickness of the lips may carry meaningful information. Such geometric features are used in different speechreading approaches (Adjoudani & Benoit, 1996; Petajan, 1984; Teissier et al., 1999). Martínez & Gutiérrez (2004) use a motion tracking algorithm applied to the face images to extract five visual features (corners, tip of nose and chin, center of the mouth).

On the other hand model features do not restrict the features to simple measures from the lips’ contours but aim at obtaining a parametric or statistical model. Parametric curves such as Bezier are also used as in (Shdaifat & Grigat, 2003) where Bezier parameters and DCT for dimensionality reduction constitute the visual features. Active contour models or snakes (Kass et al., 1988) are deformable contours defined by control points which iteratively move towards a local minimum of a given energy function. Alternatively templates (Yuille et al., 1992) are another commonly used approach where parametric curves are fit to the lip contours also minimizing an energy function. Lip contour estimation can be achieved by snakes (Chiou & Hwang, 1997) or templates (Chandramohan & Silsbee, 1996), and then the parameters or the contours of the lips can serve as visual features. Snakes are suitable for the search of amorphous object but are unconstrained and can easily lead to aberrant results when it comes to objects of known form.

Cootes et al. (1995) propose an alternative which learns the possible variation of a set of annotated images, and then the model can fit to unseen images. When the training set is such that it covers the whole panel of the different possible configurations for the images, the model can generalize well to any unseen image. A careful selection of the representative images in the training set and the manual placement of the landmarks is therefore an important prerequisite. Active Shape Models (ASM) involve local optimization and require a plausible initial position of the shape while it requires very few tuning parameters and usually converges quickly to a stable solution with less than 20 iterations. Although ASMs have been used in ASR systems (Luettin et al., 1996), Cootes et al. shortly after introduced Active Appearance Models (AAMs) (1998), which exhibits more promising characteristics (Cootes et al., 1999). Therefore most algorithms tend to opt for AAMs over ASMs, notably because it is thought that pixel texture inside the mouth carry meaningful information which shape-based features simply ignore. AAMs are an example of joint appearance and shape features (detailed in the section below).

Some systems use different techniques which do not classify in either of the previous categories. In (Lucey et al., 2000) contour estimation is carried out from gray level pixel of the mouth ROI directly via a non-linear stochastic mapping technique know as direct estimation (Chen, 1998).
c) Joint appearance and shape features

Several ASR systems have combined appearance and shape features as both methods rely on intrinsically different assumption about where the information is to be found. Indeed the shape approach provides features which one can easily relate to lip movement, but discards the information contained in the pixels. Joint methods aim at using the information from both the shape and the appearance with the hope to obtain more meaningful features.

Some methods simply concatenate output of both methods (Chiou & Hwang, 1997; Dupont & Luettin, 2000; Luettin et al., 1996) while AAM incorporate everything in a single model. Chiou & Hwang (1997) use snake and PCA features, Dupont & Luettin et al. (2000) and Luettin et al. (1996) combine ASMs and PCA features. AAMs since their creation (Edwards et al., 1994) have been a major research theme in image processing and have come to play an important role in ASR systems. The basic algorithm is described as follows (Cootes et al., 1998; Edwards, 1994; Cootes et al., 2001)

In the training set:
1. Collection of a set of images spanning the possible configurations of the face, both in terms of shape and texture (gray-level). Setting of landmarks for each individual frame in the training set previously gathered.
2. Generation of a model of face shape variation using PCA on the training set.
3. Triangulation algorithm is used to match the landmarks of each frame to the landmarks of the mean shape. Subsequently PCA is applied to create a model of appearance variation of shape-normalized faces.
4. The two kinds of parameters are concatenated and a further PCA is applied to de-correlate shape and texture. This constitutes the appearance model.
5. Learning of the relationship between image error and parameter error. This is originally done by means of multivariate linear regression.

During the search: iteratively adjust the parameters and match synthesized image to target image.
6. Iteratively update the parameters to minimize the image error until convergence of the parameters.

Many ASR systems rely on AAMs to extract feature from the video front. Neti et al. (2000) reflected that the poor performance obtained via the use of AAMs (as opposed to 2D DCT) was due to insufficient training data and tracking error. Matthews et al. (2001) use AAMs to track the whole face and also reports similar results and explanations. The correlation between model parameters and resulting differences in image is fundamental and later improvements have been proposed in the image processing literature (Cootes & Kittipanya-ngam, 2002). (Baker & Matthews, 2001) introduced the use of an inverse compositional approach, subsequently augmented by Papandreou & Maragos (2008), where fitting performance of the AAMs were enhanced. Recent experiments on AAMs with a linear predictor described in (Ong & Bowden, 2008) display improvements. Lan et al. (2009) show that shape alone is meaningful but inner appearance of the mouth carries more information. In the same paper, AAMs implemented on audiovisual ASR are reported to outperform systems based on 2D DCT, Sieve and ASMs.

4. Our approach based on 3D DCT.

a) Introduction

All images are treated as RGB images and to simplify notation all calculation is done only for one channel as it would be exactly the same process for the two others. Unless there is explicit mention of the use of a specific color channel it is assumed that the same process is applied to all three channels.

Our method differs from most previous studies insofar as it is based on image sequences as opposed to individual frames. We treat each vowel as a sample. Therefore we need features which account for the similarity between consecutive frames. High-level approaches based on individual frames are expected to perform poorly as each frame would be processed independently of the others. Joint appearance and shape
models could be considered with a three-dimensional version of AAMs. However it is commonly admitted that the performance depends greatly on the quality and variety of the manually annotated frames. Rather, we opt for DCT.

Unlike the systems described in (Duchnowski et al., 1994; Neti et al., 2000; Potamianos & Graf, 1998) which use 2D DCT for audiovisual ASR, our purpose is to incorporate the temporal information in our model. We believe that the lips and their displacement in time carry information about the vowel uttered. Also, the use of 2D DCT for individual frames overlooks the fact that consecutive frames are also correlated in time. For instance, one can easily expect a great degree of correlation between consecutive frames from Figure 8. Therefore, the use of time as a third dimension for the DCT is expected to convey more information.

Figure 1b shows that video sequences have different lengths. This is problematic as we will use a standard SVM for classification, which imposes a constraint on the sample length. In order to tackle this aspect, we chose to normalize each sequence in time inside the DCT algorithm. The proposed algorithm therefore enables decorrelation and compression of the video sequence yielding outputs with standardized size, suitable for later classification.

b) 3D DCT

In order to compute the time-normalized 3D DCT as in equation (2a) and its inverse (2b) we implemented the algorithm explained in

\[
\begin{align*}
\text{DCT} & : S(w,v,u) = a(x,w)a(y,v)a_{dct}(t,u) \ast \sum_{i=1}^{x} \sum_{j=1}^{y} \sum_{k=1}^{t} [s(i,j,k) \cos(t_i) \cos(t_j) \cos(t_k)] \\
\text{IDCT} & : s(i,j,k) = \sum_{w=1}^{x} \sum_{v=1}^{y} \sum_{u=1}^{t} a(x,w)a(y,v)a_{idct}(t,u) [S(w,v,u) \cos(t_i) \cos(t_j) \cos(t_k)]
\end{align*}
\]

where

\[
\begin{align*}
t_i &= \frac{\pi(2i-1)(w-1)}{2x}, \quad t_j = \frac{\pi(2j-1)(v-1)}{2y}, \quad t_k = \frac{\pi(2k-1)(u-1)}{2t} \\
a(N,k) &= \begin{cases} 
\frac{1}{\sqrt{N}} & k = 1 \\
\frac{2}{N} & 2 \leq k \leq N 
\end{cases} \\
a_{dct}(t,u) &= \begin{cases} 
\frac{1}{t} & u = 1 \\
\sqrt{2} & 2 \leq u \leq t 
\end{cases} \\
a_{idct}(t,u) &= \begin{cases} 
\frac{1}{t} & u = 1 \\
\sqrt{2} & 2 \leq u \leq t
\end{cases}
\]

It is based on the concept of separability, which states that a multi-dimensional DCT may be computed as a series of one-dimensional DCTs as explained in

Figure 9.
Normalization in time for DCT is achieved by dividing each coefficient by the duration of the time signal \( t \). The DCT coefficients are then left invariant by compression or stretching of the initial time signal.

A given image sequence and its DCT have the same dimensionality, however we want to discard image artifacts and noise by reducing the number of DCT coefficients. By definition of the 1D DCT (3), coefficients corresponding to high frequencies represent only details of the image. As a result one can preserve most of the image information by discarding DCT coefficients for the higher frequencies. With the same notations as (3), for a signal \( s \) containing \( N \) samples, one can find an integer \( N_0 \) such that the DCT coefficients, \( S(u) \) for \( N_0 < u \leq N \) can be discarded without loss of useful information.

\[
S(u) = w(u) \sum_{i=1}^{N} s(n) \cos \left( \frac{\pi(2i-1)(u-1)}{2N} \right)
\]

where \( w(u) = \begin{cases} \frac{1}{\sqrt{N}} & u=1 \\ \frac{2}{\sqrt{N}} & 2 \leq u \leq N \end{cases} \)  \( \quad (3) \)
c) Determination of the parameter numbers \((N_x, N_y, N_t)\)

In our case the purpose is to determine the following parameters independently of the image sequence so as to obtain a fixed number of parameters \((N_x, N_y, N_t)\) where \(1 \leq N_x \leq x\), \(1 \leq N_y \leq y\), \(1 \leq N_t \leq t\) with \(x = 46\), \(y = 66\), and \(t\) being a random variable. It follows that \(N_i \leq \min_{im}(t(\text{im}))\).

Since the recordings were obtained from the same setting for all subjects, it is considered a reasonable assumption that the parameters are subject-independent. Therefore, those parameters are determined for the whole data set, however when other subjects will be recorded to supplement our database, one will need to compute them again.

We aim at finding \((N_x, N_y, N_t)\) such that their product is minimum and the average Peak Signal to Noise Ratio (PSNR) between original and reconstructed images is above a given threshold “th”.

The Mean Squared Error and the Peak Signal to Noise Ratio between two RGB images is defined in equation (4)

\[
\begin{align*}
\text{MSE}(I,J) &= \frac{1}{N \times M \times 3} \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{k=1}^{3} (I(i,j,k) - J(i,j,k))^2 \\
\text{PSNR}(I,J) &= 10 \log_{10} \left( \frac{\text{MAX}_I^2}{\text{MSE}(I,J)} \right) \text{ and } \text{MAX}_I = 255
\end{align*}
\]

PSNR is commonly used in image processing for applications such as denoising and is the standard tool to estimate the quality of image compression methods. We chose this approach here as the DCT space is delicate to interpret, whereas PSNR is thought as an approximation of how human beings perceive image reconstruction. We assumed that characteristics observed in the image space will be corresponding in the DCT space.

We decided to reduce the complexity of the search by imposing the constraint \(N_x / x = N_y / y\), which means that the percentage of coefficients preserved in the two space dimensions is equal. This is sensible as there is no apparent reason to privilege one space dimension over the other. PSNR requires two images as input. The first image is the original frame taken from the image sequence describing the lips for a given vowel (cf. Figure 8) and the other image is reconstructed via 3D IDCT with parameters \((N_x, N_y, N_t)\). For a given set of parameters, PSNR is computed for each image in each sequence, thereby requiring computation of IDCTs. For each image sequence, three DCTs are computed (one for each color channel); however the computation of IDCTs is dependent on the parameters. Therefore if \(p\) denotes the total number of parameter combinations, there will be approximately \(p\) times more IDCTs. Moreover the computation time is higher for the IDCTs and it also depends on the number of frames inside the image sequence. Typically it takes between 50 and 200 ms to compute 3 IDCTs for a 46*66*10 image sequence. Our limited database contains 677 sequences so we could approximate the computation time to be up to 2 minutes for each parameter configuration. If one wanted to conduct a thorough search, this would be unnecessarily time consuming.

Indeed it is clear that PSNR is an increasing function of both parameters. As a result if configurations \((N_{x'}, N_{y'}), (N_{x''}, N_{y''})\) are such that PSNR \(\geq\) Threshold, it will automatically be the case for \((N_{x''}, N_{y''})\). Also if PSNR\((N_{x'}, N_{y'}) < Th\), then PSNR\((N_{x''}, N_{y''}) < Th\) for all \(k\). Similar reasoning is applied for the time dimension. These properties allow for constrained optimization. The process we followed is described in

Figure 10. After running it on the whole database, the parameters obtained were \((13, 19, 4)\).
DCT coefficients of dimensionality (13, 19, 4) are computed for each color channel, which correspond to 2964-D vectors for classification. Such a number of features, in comparison to the number of samples (677 in the whole data set) likely to appear in each of the 2-class classifier we are to build, is too great. When we consider only correctly uttered vowels and consider pairwise classifiers (e.g. 'Y' vs. 'A') the number of samples decreases drastically. For example there are only 8 instances of correctly uttered 'O' out of the 18 words containing 'O' as first vowel. We are then faced with a classification problem where the number of features is far greater than the number of samples.

Although it is possible to synthesize new samples from estimated densities, it is not conceivable to estimate the densities in our case because of the great discrepancies between sample numbers and number of features. Instead the approach chosen is to apply dimensionality reduction to reduce the number of features. This will be detailed in section VI.

d) Normalization schemes
Another aspect to consider is that DCT coefficients were obtained from images of subjects exhibiting different skin complexion. There is reason to believe that the discrepancies existing between the different subjects will still exist in the DCT space. As a result we have considered different normalization strategies.

We have tried different normalization strategies which will be detailed below. First we need a way to estimate and compare those different strategies. This can be done by transposing the resulting DCT back to the image space. Here we no longer deal with a problem of loss compression as it was the case when we were searching for the number of DCT coefficients to preserve. As a result we cannot use PSNR or similar approaches to provide a score. Alternatively we can compare the original and reconstructed images and estimate qualitatively the effect of the normalization scheme. This approach is not rigorous and instead we may discuss the usefulness of the normalization scheme based on the impact on the classification error rate via cross validation.

The first DCT coefficient is given by equation (2a) applied for (1,1,1). It follows that $S(1,1,1)$ is proportional to the sum of all pixels (cf. equation (5)).

$$S(1,1,1) = \frac{1}{\sqrt{N_x \cdot N_y \cdot N_t}} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \sum_{k=1}^{N_t} s(i, j, k)$$  \(5\)

The first idea that came to mind is to normalize $S(1,1,1)$ over the whole data set. We expect that this will normalize the overall texture of the image after re-transformation in the image space. This process is detailed in Figure 11. Images from four different subjects were displayed before and after normalization of the first DCT coefficient. It is first observed that the shapes of the lips are not altered by this transformation. One can also notice the slight change in texture (especially for (c)) along with the smoothing which was induced by the reduction to the number of coefficients to $N_x$, $N_y$ and $N_t$. However, one can easily distinguish between the textures from (b) and (c) even after transformation.

**Figure 11 Normalization of the first DCT coefficient**

The previous scheme contributed to normalize the texture to a certain extent. If we come back to equation (2a) and think of the meaning of $S(w,v,1)$, for small values of $v$ and $w$ we can interpret this as that the coefficients are linked to the general texture of the image. For the next scheme, instead of using only the first coefficient for normalization, a $k*k$ square is considered (cf. Figure 12).

**Figure 12**
There is little difference between images obtained after normalizing the first (1:2, 1:2) coefficients (k=2) compared to the previous case (k=1). However when k=5 one can notice that even though the texture is less dependent on the subject, the shapes of the lips have been altered. Consequently this approach may not be the best way to normalize the DCT coefficients.

One can point out that for classification the value of the mean coefficients $S(1:k,1:k,1)$ is of no importance. Indeed their value would be the same for the whole set, contributing in no fashion to the distinction between two classes. As a result computation of the mean coefficients over the whole set are superfluous and only performed if one needs to return in the image space. This normalization scheme only boils down to ignoring a subset of coefficients for later classification. We will see in section VI that feature selection is a much better alternative.

Yet an advantage of the approach detailed above is that it is generalizable to unseen data. One should keep in mind that any transformation on the training data has to be reproduced on the test data. By definition only the training data is at our disposal during training and normalization schemes must remain class-independent as it must be applied to the test set for which the information about class labels is unknown. If we constrain the previous method to individual classes, that is, we compute the mean of $S(1:k,1:k,1)$ within a vowel class, and repeat the process for different vowels, then a fundamental problem occurs. For each sample in class $\{C_k\}$ a subset of coefficients will have the same value and this value will be different from one class to another. Therefore this process is similar to attributing a label for each class. Moreover $S(1:k,1:k,1)$ cannot be obtained for the testing set, which renders this approach unpractical.

Alternatively one might consider a normalization scheme occurring in the image space, which would affect all the DCT coefficients. The overall visual front end is summarized in Figure 13.
V. Classification by SVMs

A brief review of works using Support Vector Machines (SVMs) (Vapnik, 1995) as classifier for pronunciation error detection precedes explanations of our approach.

1. Introduction

The original purpose of audiovisual ASR is to improve the performance of speech recognition over systems solely based on auditory features. To this effect, auditory and visual features are combined and integrated together. Due to their ability to efficiently process samples of different length in time, Hidden Markov Models (HMMs) (Deller et al., 1993; Rabiner & Juang, 1993; Young et al., 1999) have been used as a state-of-the-art classifier for speech recognition. Although HMMs is the most commonly used classifier for speech processing, its inability to handle small data sets has encouraged research about other classifiers. Moreover HMMs are reported not to be suitable to distinguish between sounds with similar spectrum and different duration (Strik et al., 2009).

SVMs are renowned for their ability to generalize well even if the training data is of limited size (Vapnik, 1995), which makes SVMs good candidates for classification problems on sparse data. As a result, SVMs have been used in speaker identification (Fine et al., 2001), speaker verification (Le & Bengio, 2003), pronunciation scoring (Yoon et al., 2009) and mispronunciation detection (Wei et al., 2009), but are not yet prevalent in the field of ASR. Indeed speech classification involves the use of speech segments of variable length, whereas SVMs require fixed-length samples. Therefore different approaches have been taken to overcome this difficulty and benefit from the SVM generalization characteristics. A first class of methods comprises the various possibilities for time-normalization of the inputs. Methods based on hybrid HMM-SVMs constitute the second class. A detailed overview is given in (Solera et al., 2007).

2. Our approach based on SVMs.

The main purpose of this work is to create a general classifier used for the 2-class problem (cf. Figure 14). This classifier is designed to be class-dependent but run automatically in a similar fashion for any class involved. Each class will contain samples from a certain vowel; the features will either be auditory, visual or resulting from their combination. As a result for each pair-wise classifier a different set of parameters will be chosen so as to decrease the classification error as much as possible.

We adopt a general K-fold Cross Validation scheme to assess the performance of the classifier. A subset of each class composed of K-1 fold is used as training data, while the remaining fold is used a validation test to assess the generalization properties of the classifier obtained on the training set.
Support Vectors Machines are to be used for classification. Their ability to generalize well for sparse datasets is the target characteristic for our application. After clustering the dataset into 18 classes by vowel and restricting to the correctly pronounced samples, we obtain the distribution shown in Figure 15. Seven classes of vowel present fewer than 20 samples and none of the classes contain more than 50 samples. Such numbers considerably reduce the panel of classifiers at our disposal and reinforce the idea that SVMs are best suited for the task. Additionally one is less likely to opt for HMMs as they require far greater amount of training data, which justify our choice not to consider this approach.

Matlab provides an implementation of SVMs restricted to the two-class problem, which is sufficient as multiclass classification is not discussed in this thesis. Among the diversity of kernels this implementation allows to use, we decided to exclude those which involved several parameters. Indeed the larger the number of parameters in the model, the more the model is prone to overfitting. As a result we tried early experiments on two kernels, namely linear kernel and radial basis function (rbf) kernels. However results obtained pointed out that both kernels yielded similar results with a slight bias toward the rbf kernel in terms of performance. The training process was faster with rbf kernels which also present a better ability to fit to the configuration of the training set. Consequently we decided to opt for the rbf kernel for which one tuning parameter has to be adjusted. Indeed Figure 16 illustrates how the classification boundary along with the error rate is affected when this parameter varies. The tuning of this parameter is done via cross validation.

When considering the varying number of samples depending on the different classes, we have thought of equalizing their number. Yet we have no other sample points with correct labels at our disposal and we already agreed not to consider samples for which the grading was not unanimous. As a result it has been chosen to duplicate some of the samples to augment to less populated class for each binary classifier. This is primarily considered as a remedy against the possibility that the SVM ignore poorly represented classes and assign the same label to the whole test data. With the dataset used for our experiments, the lowest number of samples for a given class was 8, and the largest 46.
Even with our data where this equalization process seems unnecessary, we obtained results suggesting its usefulness. Moreover our classifier generation algorithm is to be applied to any data, for which the number of samples is unknown. For example one could choose to include ‘A’ ‘E’ ‘I’ ‘U’ ‘O’ from dozen of subjects and ‘Å’ for only one subject, in which case the sample number per class would be highly unbalanced. For this reason we believe it would be beneficial to keep this class-size equalization process in the general framework of our algorithm. All results displayed further will be based on this improvement.

Figure 16 Impact of the tuning parameter (sigma) for rbf kernel.
VI. Combination of the audio and video features

1. Introduction

Most integration techniques rely on various declinations of HMMs (Potamianos, 2003). Audiovisual integration is either performed before classification (feature fusion), or after (decision fusion). Feature fusion consists of applying a single classifier on the concatenation of auditory and visual features (Adjoudani & Benoit, 1996; Potamianos et al., 2000). On the other hand, in decision fusion the outputs of two single-modality classifiers are combined, generally inside the HMM framework (Hennecke et al., 1996; Dupont & Luettin et al., 2000; Neti et al., 2000).

As HMMs are not the subject of our work, they will not be explained in details. Similarly the problem of early, intermediate and late integration in decision fusion along with other particularities pertaining to HMMs will not be discussed. Indeed our work focuses on vowels and the concepts of integrating auditory and visual feature at word level is irrelevant for us.

We consider the standard approach of combining the features before classification. This simply consists of concatenating the auditory and visual feature vectors. In our current framework we use red, blue and green channels for the visual modality, which multiplies the number of visual features by three. As a result we obtain 2,964 visual and 55 auditory features. So far we have considered using the visual information as a supplement to improve the classification performance between different vowels. Consequently we expect that such a configuration cannot be optimal in a sense that the visual information will be predominant and the classifier will be unaffected by the auditory features. As a result we should consider a dimensionality reduction approach.

In fact we have already applied dimensionality reduction (cf. Figure 13) as we started from a image sequence of dimensionality 720*576 *3* t with $4 \leq t \leq 12$ to a feature vector of length 988*3, thereby achieving a dimensionality reduction of a factor varying between 400 and 1,200. Yet this is not sufficient and we cannot simply reduce the number of DCT coefficient by choosing a smaller set of $(N_x, N_y, N_t)$ coefficients. Instead we will use feature extraction and feature selection techniques.

A brief overview of some standard techniques with their respective advantages and drawbacks is given before we explain the methods we adopted in our framework.

2. Presentation of different feature extraction and feature selection techniques

---

B Figure 1b shows that $2 \leq t \leq 10$. When adding two additional frames (one before and one after the original sequence) we obtain $4 \leq t \leq 12$. 

Classification problems for data which comprise a great number of features can prove to be a computationally intensive task. When the dimensionality greatly exceeds the number of samples, the curse of dimensionality (Bellman, 2003; Donoho, 2000) further complicates the classification task.

Dimensionality reduction (Fodor1) which is subdivided into feature extraction and feature selection (Liu & Hiroshi, 1998; Guyon et al., 2003) is commonly used to remedy this problem. Feature extraction (Guyon et al., 2006; Nixon & Aguado, 2008) consists in transforming the original set of features into a smaller one. Principal Component Analysis (PCA) (Fodor1) and Linear Discriminant Analysis (LDA) (Fisher, 1936) are among the most popular linear techniques. In feature selection, a subset of the original features is selected and the remaining discarded. While an exhaustive search is theoretically bound to yield optimal results, it is unachievable in high-dimensional spaces due to the prohibitive number of combinations \(2^{n^1}\) in a n-D space. Therefore techniques aiming at producing an acceptable subset of features by testing a reasonable number of configurations are considered. Stepwise regression is a commonly employed greedy search and genetic algorithms (GAs) (Holland, 1975; Goldberg, 1989; Hussein, 2001) on the other hand are an alternative to the thorough search. Stepwise regression is generally used as a filter while GAs are an example of wrappers. Minimum redundancy maximum relevance (MRMR) (Ding & Peng, 2005; Peng et al., 2005) has recently emerged as a rapid and superior filter method which focuses on discarding redundant features and giving importance to relevant subsets.

a) PCA
PCA is one of the most commonly used dimensionality reduction techniques. PCA transforms the search space and orders the new features according to their weight in the overall variance. This usually benefits classification problems as features with little variance are given little importance. Those features are similar between and within different classes and offer no discriminative properties. Also, dimensionality reduction through PCA is a way to discard noisy features. The PCA features are also uncorrelated and it is generally considered to be an advantage for classification. Additionally the transformation involved in the computation of the PCA features has a closed-form solution, is reversible and based on Singular Value Decomposition, a classic linear algebra factorization and therefore PCA is implemented in a number of software.

The first PCA features, which contain most of the variance, often prove to be a good feature extraction approach before classification. However, those features can be redundant or even contradictory, thereby reducing the performance of the classifier. Similarly, from the fact the features are uncorrelated, one cannot conclude that it would necessarily benefit the classification. Indeed features which are not relevant for the classification may be given important weight and therefore the classifier is inclined to give modest performances.

b) GAs
GAs can easily be applied to high-dimensional data sets as they scan the search space regardless of its configuration. GAs do not require any prior knowledge about the data and its use is not restricted by any mathematical hypothesis. The search is global and relatively insensitive to local minima compared to gradient descent methods. It is a heuristic method and does not target the optimal solution, which in many cases is not problematic as the search space is so wide that it would be unrealistic to envision finding the optimal solution. Instead GAs often point to satisfactory or good solutions and discard the weak ones (as it is the case in nature where only the fittest individuals are meant to survive). Last but not least, GAs are extremely flexible and there is virtually no limit to the modifications each programmer can introduce.

The evolutionary aspect of the algorithm can be the origin of its weakness. In theory any point of the search space is likely to be considered by the GAs but in practice as the process evolves a subset of relatively similar candidates is susceptible to emerge and the subsequent iterations will generate candidates with a high degree of similarity, thereby reducing the likelihood of producing utterly different samples.
Evolution does not necessarily lead to suitable solution fast and when the cost function used to evaluate the pertinence of each candidate is time-consuming, the overall GAs suffer all the more in terms of computation time. The inherent character of randomness for each process (mutation, cross-over and initialization) makes GAs’ behavior unforeseeable. The outputs given by different runs can be disparate if the algorithm is constrained to terminate in too short a time.

c) Filters and Wrappers
Feature selection falls into two main subcategories, filters and wrappers. In the former, features are selected according to their inherent properties and scores obtained by means of statistical tests (t-test, F-test, entropy, mutual information) are used to estimate their discriminative properties on the target classes. Wrappers are used along with classifiers and the relevance and discriminative power of the selected subset is directly linked to the performance of the classifier. In comparison filters provide results much faster and independent of any classification algorithm to be subsequently applied. On the other hand wrappers are computationally costly and their results are optimized for the classifier used, with a risk of overfitting.

d) MRMR
MRMR is a feature selection algorithm belonging to the filter class. Unlike methods selecting top ranking features, it considers the relationships between the features in order to sort them according to their discriminative abilities. It relies on relevance and redundancy estimation, which is reduced to a 2-dimensional problem. Therefore greedy search is allowed. Processing time varies linearly with the number of features to be retained. As a filter method it gives generalized outputs as the selected subset of features are independent of any classifier.

3. Our approach

a) PCA based feature extraction

We first considered the classic PCA approach applied to the concatenated feature vectors (55 auditory and 2964 video features). We use the whole data set to find the number of principal components to keep in order to preserve 95% of the variance as it is usually assumed that the remaining 5% contain little but noise. Subsequently in the Cross Validation, we perform PCA on the training set and apply the same transformation for the testing set, and iterate the same process for each fold. For each fold used as testing, the PCA transformation is different, yielding different features each time. The feature fusion scheme based on PCA is described in Figure 17.

---

Cross validation

- Data set
- 2964
- 55
- PCA
- N
- PCA keep N features
- Train SVMs
- Test SVMs
- Projection on same PCA space
- Test
With our data set, we found $N = 386$. This is still a large number of features compared to the cardinality of each class. Moreover this approach is flawed as the concatenated features mainly consist of the visual information. This is undesirable as we expect the auditory channel to carry more information for separation of the different classes. Consequently we tried a slight alteration of this approach (cf. Figure 18). PCA is applied separately on the auditory and visual features so as to keep 95% of the variance for both cases. Then the output is concatenated and another PCA applied to it, again we keep 95% of the variance.

However results prove to be unsatisfactory, indeed the video channel is still predominant as we have $N_v = 380$ video features and $N_a = 38$ auditory features if we follow this process. We can act directly on the number of principal components ($N_v$, $N_a$, $N$) to keep instead of imposing a constraint on the variance. Different experiments have proved that the use of PCA, with one or two layers, imposing a constraint on the percentage of the variance or directly the number of principal components to preserve, is unsatisfactory. We have found that the addition of the visual channel signified an increase of the error rate (cf. section VII.1). This is due to the fact that PCA models the whole training data regardless of the class labels. Therefore the first components represent features which vary the most for the data as a whole. When one uses PCA before classification, special caution about the meaning of the transformation in PCA space is often overlooked as often the most discriminative features are also the ones exhibiting the greatest variance. It occurs that in our problem PCA fails to separate different vowels.

**b) MRMR based feature selection**

We then considered feature selection instead of feature extraction. Given the great number of features among which we aim at selecting a subset, we must rule out the possibility of a greedy search. Instead we have opted for MRMR which proved to be a fast algorithm, for which convergence is linearly dependent on the number of output features. A short description of the basic concepts (Ding & Peng, 2005; Peng et al., 2005) and the integration of the algorithm in our framework follow.

MRMR combines two main ideas. The first is to select features dissimilar to each other, that is, minimizing the redundancy. The second is to maximize the relevance, that is, the contribution of the features for the classification. Maximum relevance uses the class labels to select features which will have a good aptitude to separate the data in the given classes. Minimum redundancy (6a) and maximum relevance (6b) criteria are defined in terms of mutual information.

The mutual information for two discrete variables $(x, y)$, given their joint distribution probability $p(x, y)$ and marginal densities $p(x)$ and $p(y)$ is defined as:

$$ I(x, y) = \sum_{i,j} p(x_i, y_j) \log \left( \frac{p(x_i, y_j)}{p(x_i)p(y_j)} \right) $$

(5)

S denotes the feature space, $x_i$ the $i^{th}$ feature, $h$ the target classes.
Both conditions are incorporated into a single criterion: max \((V_I - W_I)\). The algorithm does not aim at selecting feature independently of each others. Instead it selects a compact subset of features and takes advantage of the correlation between features, as suggested by the use of mutual information to compute \(V_I\) and \(W_I\).

Although the algorithm enables the use of continuous variables, it is advised to discretized the data and use the discrete version. Indeed the computation of mutual information is considerably faster and a high level of details brought by the continuous character of the data is not useful. As a preprocessing step we normalize the data by adjusting its mean to 0 and standard deviation to 1. Discretization is then applied by taking the integer part of the resulting data. The MRMR algorithm implemented by Peng et al. (2005) is used in our framework to select a subset of features.

The output of the algorithm is a list consisting of feature indices sorted by order of relevance for later classification. The first index corresponds to the best single feature to maximize \((V_I - W_I)\), the first pair of indices corresponds to the best pair of features to maximize \((V_I - W_I)\) and so forth.

In Figure 19 we display an example of feature selection for /A/ versus /A:/.

The output can be interpreted as follows: if one wishes to use only one feature, this should \#28. When it comes to two features, by definition (Peng et al., 2005), \([\#28, \#2846]\) will be better than \([\#28, \#k]\) for any \(k\). This is highlighted by Figure 19 (b), (c) and (d) while (a) shows that using features not selected by MRMR are less suited for classification.

**Figure 19 MRMR feature selection**

MRMR is a filter method, which means there is no convolution to any classifier as the original purpose is to produce a subset of features exhibiting good discriminative properties regardless of the classifier used. As a result we must implement MRMR in a CV framework where SVMs are used to estimate the error rate for the data restricted to the features selected by MRMR. For each iteration we obtain a list of feature indices which fit the training data. As our data should be suitable enough for classification, we expect each class to have a distribution independent of the different folds created for cross validation. For this reason we expect MRMR to produce similar outputs from one CV training set to the next. However we envisioned the case where each fold corresponds to a particular subject in a subject-independent framework. There is to date no subject-independent visual features suited for ASR, therefore our previous assumption does not hold and we should consider a way to combine the output for each realization of MRMR.

The following approach was considered. In a preliminary CV loop, we compute the MRMR features for each fold. The first \(N\) (typically \(N\) is not greater than 50) features common to all the folds are kept and the
others ignored. Those features are later used in another CV loop to estimate which is the best configuration in terms of error rate for SVMs.

Table 2 First six feature indices for each fold in CV (person independent and 5-fold) for two different classifiers, based on audiovisual data.

<table>
<thead>
<tr>
<th>Person independent CV {A, A:} vs {E, E:}</th>
<th>Person independent CV {I, I:} vs {Y, Y:}</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 10 16 13 51 19 336</td>
<td>#1 113 2133 1045 2131 1060 2093</td>
</tr>
<tr>
<td>#2 10 2667 19 13 12 31</td>
<td>#2 57 111 551 88 1101 2236</td>
</tr>
<tr>
<td>#3 10 16 19 7 12 18</td>
<td>#3 57 1099 62 2159 2418 195</td>
</tr>
<tr>
<td>#4 10 19 16 13 12 31</td>
<td>#4 2033 111 88 551 1045 157</td>
</tr>
<tr>
<td>final 10 19 16 13 12 7</td>
<td>final 57 111 1045 2033 1101 2236</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5-fold CV {A, A:} vs {E, E:}</th>
<th>5-fold CV {I, I:} vs {Y, Y:}</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 10 19 13 12 16</td>
<td>#1 57 2062 1539 1145 113 111</td>
</tr>
<tr>
<td>#2 10 19 16 13 31</td>
<td>#2 2033 2087 1101 157 1050 265</td>
</tr>
<tr>
<td>#3 10 19 16 13 31</td>
<td>#3 2033 2087 157 57 1050 265</td>
</tr>
<tr>
<td>#4 10 19 13 16 12 7</td>
<td>#4 57 1099 597 113 157 2033</td>
</tr>
<tr>
<td>final 10 19 13 16 12 7</td>
<td>final 57 2033 157 2087 1045 2093</td>
</tr>
</tbody>
</table>

There are several observations to be made about those results. In the first classifier {A, A:} vs {E, E:}, the features selected were for the first six only coming from the auditory modality ($i < 56$) while it is the opposite for {I, I:} vs {Y, Y:}, indeed only visual features were selected. This reinforces the idea that visual information may be useful in certain cases. For the first classifier the first six features selected are the same whether we use a person dependent or independent scheme, while it is not the case for the other classifier. This shows that auditory features are suitable for a speaker independent scheme. For the other classifier, we need to display more than six features to notice similarities between the different CV schemes. This suggests that these visual features are not as suitable as the auditory features we use.

We can also notice that a given feature may be selected at a different position in several folds, (e.g. Feature #2093 is considered the 6th most important in 5-fold CV for {I, I:} vs {Y, Y:} but does not appear in the first 6 features for any of the different folds. It does not appear in the table, but the list of indices for which it is found in the different folds is (11, 11, 15, 7). Conversely, some features are selected early for some folds but are not used in others. In the same experiment, #597 is found at the following positions: 3 in fold 4, 77 in fold 1 and 90 in fold 2, whereas fold 3 and 5 do not count it within the first 200 features. Overall this scheme is thought to be useful for generalization to unseen data. Indeed when the final feature indices are highly correlated to those for each fold as it is the case for {A, A:} vs {E, E:}, we can expect a good generalization ability. Also features irrelevant for a given fold in CV, although considered more important for the remaining part of the data, are likely to be irrelevant for unseen data.

**c) Improvement of the method by means of GA**

MRMR does not impose any constraint on the number of features one should preserve. There is hence no reason to believe that using a larger number of features, although it may increase (V1 - W1), might be beneficial for classification. Moreover when the algorithm returns the indices of N features, it does not provide information of what is the best configuration. Indeed a group of three features ($i, j, k$) can have better discriminative properties than ($#1, #2$) for $i, j, k \in \{1, 2\}$. 

![GA diagram](image-url)
Moreover we can see that the order of the features is altered when we restrict the features to belong to the MRMR lists given for each fold in CV. As a result we may question the relevance of the order in the final feature indices list. Overall we may consider that MRMR is a preprocessing step providing a list of candidate features which exhibit satisfactory discriminative properties for classification. As a result we have considered using Genetic Algorithms (GA) to select among those features. Indeed, the global character of GAs will enable choosing among the best candidates regardless of the order in the MRMR feature list. We implemented a basic version of GA according to the scheme described in Figure 20.

We use this implementation of GA with a cross validation scheme in order to select both the feature indices and to determine the value of sigma. Indeed it is likely that sigma is correlated to the subset of features selected for classification. The individuals generated for GA are binary vectors of length L1+L2. L1 is the number of features considered and L2 is the number of bits used to code information regarding sigma. Previous experiments without GAs have shown that for some vowel pairs, classes are linearly separable when the first two or three features selected via MRMR are chosen. As a result we decided to force a given percentage of the initial population to correspond to the first few MRMR features. The reason for such an implement is purely practical. Indeed GAs are highly unstable in terms of convergence time, and the selection of best individuals requires training and testing of SVMs for the whole candidates, can take several minutes for a single iteration of GA. GA terminates if a candidate for which the CV error rate is lower than 1% is found or when the total computation time exceeds a given threshold.

For practical reasons we did not consider using GAs as a standalone feature selection method. Indeed the initial number of features is close to 3,000 when visual features are considered, which corresponds to a search space of dimensionality $2^{3000}$ roughly approaching $10^{900}$ combinations. Although GAs can be used for wide search spaces, the bottleneck of the method often resides in the estimation of the cost function for each individual generated. In our case this cost function corresponds to training a SVM, testing it and computing the error rate within a K-fold CV loop (cf. Figure 20), thereby multiplying the whole process by a factor K. Moreover training SVMs can be more time consuming when the data is not linearly separable. GAs without MRMR preprocessing are also likely to create individuals for which features yield inferior discriminative ability. This would result in a particularly delicate classification task, which would contribute to making computation time for SVM training increase. The overall MRMR+GA approach is detailed in Figure 21:

![Figure 20 Genetic algorithm scheme](image)

**Figure 20** Genetic algorithm scheme

N1, N2, N3 and N4 are fixed to (35,5,5,5). B is fixed to 5 and all other individuals are eliminated. Cross over, Mutations and Clone randomly select among B input individuals and perform those fundamental genetic alterations.

In Cross over and Mutations, k is randomly chosen for each new individual.

The computations inside are repeated until termination of the algorithm, either when the CV error rate falls below a given threshold or when the number of iteration reaches a given value. Termination is also forced if computation time exceeds a specified value.
Figure 21 Creation of pair-wise classifier with embedded feature selection by means of MRMR and GA
Person independent framework {I,I:} vs {Y,Y:} with audiovisual features.
(N1, N2, N3, N4 ,B) = parameters for GA (35,5,5,5). It = maximum number of iterations (200)
F0 = # features selected by MRMR for each fold in CV (200), F= min( # features common to all folds, 50)

VII. Results

Our system allows the use of single modality as well as audiovisual features. The classifier takes two classes which can be specified in different ways. Experiments were carried out by selecting samples corresponding to a single vowel for each class. Only correctly pronounced vowels were considered.

All results display the error rate in percentage as average error rate over the different folds for Cross Validation. We discuss the results for different framework (PCA, MRMR, MRMR+GA) and compare for auditory-only, video-only and audiovisual features. Two schemes we used for cross validation, 5-fold CV and subject-independent CV. The latter consists of training the SVMs with data from three subjects and use the remaining subject for testing. On the other hand 5-fold CV randomly affects a fold index for each sample so that we obtain five evenly populated folds, one of which is used as testing data, the rest constituting the training data. The general formats of the result tables (cf. section X.2) are detailed as follows.

Figure 22 Format for result tables.
(a) Error relative to class tested and baseline error for each pairwise classifier. (b) Average error rate

When relative errors are used, it is helpful to consider (C0/C1) along with (C1/C0). Indeed, for a given classifier, relative error for one class as test is only meaningful when compared to the error for the other
class as test. For instance (0, 100) corresponds to a classifier affecting the label $C_0$ to all the testing data and is the worst possible case.

1. PCA-based feature extraction

a) Original PCA framework

We first implemented the PCA approach for feature reduction and a subset of the results (cf. Result table 1 in Appendix 2) are given in Figure 23.

![Figure 23 Result samples, extract of Result table 1](image)

We notice that the visual features provide unacceptable results, indeed in many cases the relative error is (100/0) which means that the whole testing data is classified as one class. In comparison, auditory features display more suitable results. However one can notice that the classifier systematically gives poor results when only the length of the vowel changes between two classes. When it comes to audiovisual features, the results prove that far from enhancing the auditory-based framework the used of audiovisual features merely contribute to improving the visual-based scheme.

As we previously mentioned, such results are not fortuitous. From the use of a majority of video features in the audiovisual framework one should foresee the predominant character of the visual features. Indeed the results for the audiovisual framework are much like those from the visual-only framework.

b) Alternative PCA framework

Then two layers of PCA were used in order to give more importance to the auditory features. We also tested the equalization of sample size in the same experiment. Details for all pairwise classifiers are given in Result table 2 in Appendix 2 and Figure 24 gives an extract corresponding to the same classifiers displayed in Figure 23 for easier comparison. We notice that adding even a single visual feature to the auditory framework contributes to an overall increase of the error rate. This tendency is confirmed when comparing (b) and (c), indeed the addition of 5 visual features instead of 1 further deteriorate the classification error rate. (b) shows improvement induced by the use of classes with same number of samples.

![Figure 24 Result samples, extract of Result table 2](image)

We suspected that the limited number of samples (e.g. 8 for O) may have a detrimental impact on the results and therefore decided to remedy this aspect by merging long and short vowel classes (cf. Result
table 3). However, comparison between (b) and (c) where we use respectively 1 and 40 visual features point to the same conclusion drawn from Result table 1, Figure 23, Result table 2 and Figure 24. Indeed the use of visual features in a PCA-based feature fusion scheme contributes only to worsen the performance of the classifiers. As a result we decided to desist from the use of PCA altogether and present result obtained via our feature selection approach.

2. Feature selection

a) MRMR

Result table 4 shows the results for each pairwise classifier and Figure 25 displays only a selection of classifiers. We first compare the results for classifiers based on visual-only features from the previous method and the approach based on feature selection via MRMR (GAs are not used here). It can be seen that the use of MRMR (b) for visual features is much more pertinent that the use of PCA (a). Indeed we notice that no classifiers present error rates higher than 35 percent with clearly contrasts with the significant number of cases when the relative error is higher than the baseline error for the PCA-based method (a). Moreover classifiers such as \{I/I:} vs \{Y/Y:} display relative error rate close to 0 percent in average. This is a major improvement over the previous approach. However when we apply this method to audiovisual features (d) we notice that in most cases it yields results that are inferior to those produced when using auditory-only features (c). Yet \{I/I:} vs \{Y/Y:} is the perfect example to highlight the shortcomings of the auditory-only features while visual-only and audiovisual-based method display superior results for this classifier. In (e) we force the number of visual and auditory features but concluded that the overall performance was not increased over the unconstrained case (d). Plus we would lose the generative character of the algorithm. Indeed for a given classifier it can be the case that visual features are more meaningful than their auditory counterparts. Therefore it is advisable not to resort to this approach and rather give free rein to the algorithm for selection of auditory of visual features based solely on the CV error rate and not on expected behavior.

Figure 25 Result samples, extract of Result table 4.

(a) Visual features with PCA 380 dimension, (b) Visual features with MRMR 30 dimensions, (c) Auditory features with MRMR 50 dimensions, (d) Audiovisual feature with MRMR 30 dimensions, (e) Auditory features with MRMR 50 dimensions concatenated with visual features with MRMR 5 dimensions.

At this stage we have seen that MRMR enables significant improvement over PCA for classifiers based on visual and audiovisual features. One can also notice from the previous tables (cf. Result table 3 and Result table 4 in Appendix 2) that the results differ very slightly between the PCA and MRMR approaches for classifiers based on auditory-only features. There is however a slight bias towards MRMR in terms of performance but not as blatant as it was for visual features. In the previous experiment we only discarded five features with our approach based on MRMR, yet we could obtain better results. In the following
experiment, the number of features is incremented in a loop (we try the first $K$ features given by MRMR for $K$ varying from 1 to 20) and the best classifier is retained. Since the baseline is fluctuating around 50 for all classifiers and because the relative error rates are very similar from both side of the table (ie. it does not change whether it is one class or the other tested) we decided to show the average error rate from here on for more clarity (cf. Figure 22 (b) as opposed to (a)). The results are given in Result table 5 and an abstract follows (cf. Figure 26)

Figure 26 Result samples, extract of Result table 5.

(a) Visual features, (b) Auditory features, (c) Audiovisual features (d) Difference {Audio} - {Audiovisual} Count for auditory (e), visual (f) and audiovisual (audio|visual) (g) features

One can notice that there are many classifiers for which the error rate is 0 percent, not only for the auditory-based method (a), but also for the visual feature-based (b) and the audiovisual feature-based method (c). In most cases the visual information is not useful; this can be shown in (g) where the digit on the right indicates the number of visual features retained for the best classifier. In many cases only auditory features are kept. However one can notice that for {I,I:} vs {Y,Y:} the use of visual features is more efficient than the use of auditory features. MRMR only selected visual features for this classification task and this shows the superiority of the method compared to PCA. It is also worth mentioning that the number of features retained can be surprisingly small. Indeed some classifiers make use of only two features and achieve perfect separation of the data (e, f, g). Finally we can also notice that in some few cases the addition of visual features contribute a slight increase of the error rate. For this reason also we decided to supplement our approach by adding GAs.

b) MRMR and GA

As previously seen, MRMR gives a pool of features which will enable satisfactory distinction between samples from two different classes. Although the iterative loop provided promising results where we could notice the benefit of the addition of the visual channel, that slight decrease in terms of performance occurred for some classifiers. The results demonstrated the quality of the MRMR approach in selecting meaningful features. However the systematic iterative approach is unsatisfactory in a sense that it does not explore we whole space of possibilities. Indeed if one chooses three features it does not have to be the first three features given as output of MRMR and we want a framework allowing the use of three (or any other number) features among the subset selected by MRMR, without privileging features according to their order from MRMR. To this end, the approach detailed in Figure 21 was implemented and results with format similar to the previous experiment follow. We also add tables corresponding to the comparison between MRMR with and without GA for the same number of maximum features.

The following experiment uses this framework for speaker-dependent data, a maximum of 20 features and 30 iterations for GAs. A summary of the results from Result table 6 are presented here in Figure 27
20 features and 30 iterations max for GA
(a) Visual features, (b) Auditory features, (c) Audiovisual features (d) Difference \{Audio\}-{Audiovisual}\nDifference \{MRMR\} – \{MRMR+GA\} for auditory (e), visual (f), audiovideo features (g)
(h) Count for auditory features, (i) count for visual features (j) Count for audiovisual (audio|visual) features

Overall the same conclusions can be drawn when comparing with the previous approach when we constrain
the maximum number of features to 20 and the maximum number of iterations to 30. It can be noticed that
for some cases the introduction of GA contributes to improving the error rate for the auditory (f) but not
quite for the visual features (e) (a positive value corresponds to an improvement over the previous method
for which we take the best classifier from MRMR with a maximum number of features equal to 20). (d)
does that for some isolated classifiers there is still a rise in the error rate when including the visual
modality. However for most cases visual information is either unnecessary or beneficial.

In order to see the power of the GAs we tried the same experiment by increasing the maximum number of
features and iterations in the GAs. The results are displayed in Result table 7 and an extract in given in

Figure 28.

(e, f, g) show an overall improvement of the error rate compared to the previous method (MRMR only) and
(d) shows that the use of audiovisual features outperforms the use of auditory-only features. The
improvements at this level are better than in the previous method (GA 20 features, 30 iterations).
We notice in (h, i) that the number of features used to produce the best classifier do not exceed 30 whereas
the maximum number allowed is 50. In some cases when the error rate falls to 0, we notice that this number
is even smaller. When auditory and visual information is combined the number of features necessary is
even smaller (j), which highlights the benefit of the association of both modalities.

![Table and Figures](image-url)
50 features and 200 iterations max for GA
(a) Visual features, (b) Auditory features, (c) Audiovisual features (d) Difference {Audio}–{Audiovisual}
Difference {MRMR} – {MRMR+GA} for auditory (e), visual (f), audiovisual features (g)
h Count for auditory features, i count for visual features j Count for audiovisual (audio|visual) features

All results discussed hitherto concern speaker-dependent framework and in the next experiment we complicate the task by selecting a speaker independent framework. Our framework failed for a few classifiers, indeed MRMR did not find enough features common to all folds, which resulted in aberrant values (with a significant improvement for the person-independent framework) for a few classifiers, mainly for the visual modality. Result table 8 shows the results, aberrant values appear shaded in gray. Overall we notice that visual features are affected more than auditory features. The error rate increases (negative values for (e, f, g)) for half of the classifiers based on visual features and the classifiers based on auditory or audiovisual features are much less affected. We also notice that several classifiers suffer from the addition of visual features compared to purely auditory features.

VIII. Conclusions and criticism

We successfully designed a framework which has the capacity to automatically generate pairwise classifiers, optimizing the recognition rate for any given pair of vowels. This framework will be easily portable to other phonemes and will be implemented for automatic mispronunciation detection with the goal to provide foreign learners of Swedish with informative feedback about their pronunciation.

Time-normalized DCTs enabled us to process vowels of different duration without the traditional use of HMMs. We could consider each vowel as a whole and the combination of filter and wrapper feature selection methods demonstrated a great ability in reducing the features to a subset suitable for classification by SVMs.

Results detailed in section VII pointed out that our approach was successful in augmenting the quality of pairwise classifiers based on auditory features by adding visual features in a person-dependent framework. However it was shown that our approach is unlikely to generalize well on a person-independent framework, which is not surprising. For all these experiments the only normalization scheme adopted was to discard the first DCT coefficient for both auditory and visual channels. One could try to improve the visual features by proposing an alternative normalization scheme.

Although we could show the superiority of the audiovisual framework, our choice of high quality recordings led to already outstanding results for the auditory modality. As a result it was not possible to show improvement when we had a perfectly separable data set for the auditory features. To assess the usefulness of our method we could include noise for the audio channel, however time did not allow for such experiments. Also since we demonstrated unexpectedly good results for the audio modality, this framework will be promising when used in systems which do not rely on the visual channel.

In section VI we discussed about the computation time for SVMs and feature selection based solely on GAs. A similar problem arises when we preselect features with MRMR. Indeed we also incorporate sigma in the
GA scheme and we show (cf. Figure 16) how its value is critical for classification. This does not pose a major problem insofar as we restrict the features to a number not greater than 50, which is a radical limitation when compared to the original number of features. L2, the number of bits used to code sigma is typically small (i.e. 10). Also the tuning step performed when initializing the individuals enables GAs to pick a satisfying candidate relatively early. Yet, these precautions do not circumvent the problem as GAs will produce individuals whose value for sigma is irrelevant. This will contribute to making the average computation time increase. As a remedy, prior knowledge about the range of values spanned by sigma could help reducing computation time for creation of the classifiers. Similarly our GA framework is simple and involves no local search, nor do its parameters vary in time. A more sophisticated GA could be tried for improving computation time; however this is a minor concern so long as the training is done offline. Other global search methods could also be tried, but we restrain ourselves from considering local search as sigma and the subset of features have no apparent reason to be uncorrelated.

IX. References


X. APPENDICES

1. List of books the user may refer to


### 2. Result tables

**Result table 1:** Feature extraction performed with PCA on one level (95% of variance kept). Person-dependent classes, separation between long and short vowels. Add phoneme length as supplementary feature, discard first DCT coefficient for normalization.

#### NumFil = 45   Subject Dependent   PCA: N=38

<table>
<thead>
<tr>
<th>ErrorRate per class</th>
<th>Baseline</th>
<th>AUDIO + phoneme length + Lv1 DCT norm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### NumFil = 45   Subject Dependent   PCA: N=380

<table>
<thead>
<tr>
<th>ErrorRate per class</th>
<th>Baseline</th>
<th>VIDEO (Lv1 DCT norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Result: Table 2 Feature extraction performed with PCA on two levels. Person-dependent classes, separation between long and short vowels. Add phoneme length as supplementary feature, discard first DCT coefficient for normalization. (b) and (c) use classes with equalized number of samples.

### Error Rate per class | Baseline | AUDIO+VIDEO (phoneme length + L1 DCT norm)

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>AUDIO+VIDEO (phoneme length + L1 DCT norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(c)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SAME EXPERIMENT BUT WITH EQUALIZED CLASSES

Num Fil = 45 Subject Dependent PCA: N: 38_1_39

### Error Rate per class | Baseline | AUDIO+VIDEO (phoneme length + L1 DCT norm)

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>AUDIO+VIDEO (phoneme length + L1 DCT norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(b)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(c)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Result table 3

Feature extraction performed with PCA on two levels. Person-dependent classes, clustering of long and short vowels.

NumFil = 45  
Subject Dependent  
PCA: N: 38

<table>
<thead>
<tr>
<th>ErrorRate per class</th>
<th>Baseline</th>
<th>AUDIO {phoneme length + Lv1 DCT norm}</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>A / A:</td>
<td>E / E:</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>1.4 / 50</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>1.4 / 50</td>
</tr>
<tr>
<td></td>
<td>44</td>
<td>1.4 / 50</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>1.4 / 50</td>
</tr>
<tr>
<td></td>
<td>49</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>74</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>114</td>
<td>2.1 / 50</td>
</tr>
</tbody>
</table>

NumFil = 45  
Subject Dependent  
PCA: N: 38_1_39

<table>
<thead>
<tr>
<th>ErrorRate per class</th>
<th>Baseline</th>
<th>AUDIO+VIDEO {phoneme length + Lv1 DCT norm}</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b)</td>
<td>A / A:</td>
<td>E / E:</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>44</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>49</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>74</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>2.1 / 50</td>
</tr>
<tr>
<td></td>
<td>114</td>
<td>2.1 / 50</td>
</tr>
</tbody>
</table>

NumFil = 45  
Subject Dependent  
PCA: N: 40_40_80

<table>
<thead>
<tr>
<th>ErrorRate per class</th>
<th>Baseline</th>
<th>AUDIO+VIDEO {phoneme length + Lv1 DCT norm}</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c)</td>
<td>A / A:</td>
<td>E / E:</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>74</td>
<td>1.5 / 50</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>1.5 / 50</td>
</tr>
</tbody>
</table>
### Result table 4 Feature selection performed by MRMR, Person-dependent classes, clustering of long and short vowels.

#### VIDEO WITH 380 PCA

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>phoneme length + Lv1 DCT norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumFil = 45</td>
<td>Subject Dependent</td>
<td>EQUALIZED CLASSES</td>
</tr>
<tr>
<td>68 A / A: 0.39 31.51</td>
<td>0.39 0.39 0.41 0.39 5.948 22.49</td>
<td></td>
</tr>
<tr>
<td>58 I / I: 9.39 9.39 33.52 48.47 9.351 95.41 89.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43 O / O: 9.39 9.39 30.52 70.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 Y / Y: 8.841 0.48 88.41 2.147 10.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70 A / A: 0.39 34.59 10.03 0.51 0.52 39.48 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63 A / A: 0.39 0.41 29.48 0.41 0.41 1.643 0.41 25.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 O / O: 0.39 0.40 14.49 0.40 0.40 0.42 0.40 11.49</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### AUDIO&VIDEO WITH MRMR 30 features retained

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>phoneme length + Lv1 DCT norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumFil = 45</td>
<td>Subject Dependent</td>
<td>EQUALIZED CLASSES</td>
</tr>
<tr>
<td>68 A / A: 0.39 31.51</td>
<td>0.39 0.39 0.41 0.39 5.948 22.49</td>
<td></td>
</tr>
<tr>
<td>58 I / I: 9.39 9.39 33.52 48.47 9.351 95.41 89.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43 O / O: 9.39 9.39 30.52 70.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 Y / Y: 8.841 0.48 88.41 2.147 10.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70 A / A: 0.39 34.59 10.03 0.51 0.52 39.48 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63 A / A: 0.39 0.41 29.48 0.41 0.41 1.643 0.41 25.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 O / O: 0.39 0.40 14.49 0.40 0.40 0.42 0.40 11.49</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### AUDIO WITH MRMR 50 features retained

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>phoneme length + Lv1 DCT norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumFil = 45</td>
<td>Subject Dependent</td>
<td>EQUALIZED CLASSES</td>
</tr>
<tr>
<td>70 A / A: 0.39 31.51</td>
<td>0.39 0.39 0.41 0.39 5.948 22.49</td>
<td></td>
</tr>
<tr>
<td>58 I / I: 9.39 9.39 33.52 48.47 9.351 95.41 89.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43 O / O: 9.39 9.39 30.52 70.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 Y / Y: 8.841 0.48 88.41 2.147 10.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70 A / A: 0.39 34.59 10.03 0.51 0.52 39.48 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63 A / A: 0.39 0.41 29.48 0.41 0.41 1.643 0.41 25.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 O / O: 0.39 0.40 14.49 0.40 0.40 0.42 0.40 11.49</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### FEATURE SELECTION MRMR AUDIO 50 features + VIDEO 5 features

<table>
<thead>
<tr>
<th>Error Rate per class</th>
<th>Baseline</th>
<th>phoneme length + Lv1 DCT norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumFil = 45</td>
<td>Subject Dependent</td>
<td>EQUALIZED CLASSES</td>
</tr>
<tr>
<td>70 A / A: 0.39 31.51</td>
<td>0.39 0.39 0.41 0.39 5.948 22.49</td>
<td></td>
</tr>
<tr>
<td>58 I / I: 9.39 9.39 33.52 48.47 9.351 95.41 89.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43 O / O: 9.39 9.39 30.52 70.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 Y / Y: 8.841 0.48 88.41 2.147 10.47 9.362 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70 A / A: 0.39 34.59 10.03 0.51 0.52 39.48 98.41 91.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63 A / A: 0.39 0.41 29.48 0.41 0.41 1.643 0.41 25.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48 O / O: 0.39 0.40 14.49 0.40 0.40 0.42 0.40 11.49</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Result table 5: Average error percentages. Feature selection performed by MRMR, Person-dependent classes, clustering of long and short vowels.

<table>
<thead>
<tr>
<th>NumFil = 45</th>
<th>Subject Dependent</th>
<th>EQUALIZED CLASSES</th>
<th>Average ErrorRate</th>
<th>(phoneme length + L1 DCT norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIDEO Feature Selection MRMR 20 features max</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature count for the best VIDEO classifier</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature count for the best AUDIO classifier</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature count for the best classifier (AUDIO+VIDEO)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

48
Result table 6 Error rates in percentage for Audio-only, video-only, audiovideo features. Features selection based on MRMR and GAs. 20 features maximum and 30 iterations for GAs.

<table>
<thead>
<tr>
<th>Video Feature Selection MRMR 20 features max</th>
<th>Audio Feature Selection MRMR 20 features max</th>
<th>Audio&amp;Video Feature Selection MRMR 20 features max</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average Error Rate</strong> (phoneme length + L1 DCT norm)</td>
<td><strong>Average Error Rate</strong> (phoneme length + L1 DCT norm)</td>
<td><strong>Average Error Rate</strong> (phoneme length + L1 DCT norm)</td>
</tr>
<tr>
<td>44 E/E: 19</td>
<td>45 E/E: 0</td>
<td>44 E/E: 0</td>
</tr>
<tr>
<td>68 I/I: 11 15</td>
<td>74 I/I: 0 0</td>
<td>68 I/I: 0 0</td>
</tr>
<tr>
<td>43 O/O: 1,8 0 0</td>
<td>44 O/O: 1,8 0 0</td>
<td>43 O/O: 1,8 0 0</td>
</tr>
<tr>
<td>43 U/U: 2,7 1,1 0 19</td>
<td>45 U/U: 0 0 0 0</td>
<td>43 U/U: 0 0 0 0</td>
</tr>
<tr>
<td>48 Y/Y: 2,6 1,1 0 17 16</td>
<td>49 Y/Y: 0 0 0 0 3,3</td>
<td>48 Y/Y: 0 0 0 0 3,3</td>
</tr>
<tr>
<td>70 A/A: 18 22 11 0 0 2,2</td>
<td>74 A/A: 0 0 0 0 9</td>
<td>70 A/A: 0 0 0 0 9</td>
</tr>
<tr>
<td>65 A/A: 5,3 0 1,5 11 15 9,9 0</td>
<td>64 A/A: 3 0 4,6 0 0 0 0</td>
<td>65 A/A: 3,1 0 3,8 0 0 0 0</td>
</tr>
<tr>
<td>115 O/O: 16 0 1,5 3,7 9,3 14 0 0 18 0</td>
<td>115 O/O: 0 0 0 0 0 0 0 0</td>
<td>115 O/O: 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td><strong>Difference (MRMR 20 - GA (20,30))</strong> VIDEO</td>
<td><strong>Difference (MRMR 20 - GA (20,30))</strong> AUDIO</td>
<td><strong>Difference (MRMR 20 - GA (20,30))</strong> AUDIO &amp; VIDEO</td>
</tr>
<tr>
<td>44 E/E: -7</td>
<td>44 E/E: 0</td>
<td>44 E/E: 0</td>
</tr>
<tr>
<td>68 I/I: 4 -1</td>
<td>74 I/I: 0 0</td>
<td>68 I/I: 0 0</td>
</tr>
<tr>
<td>43 O/O: -0,3 1,1 0</td>
<td>44 O/O: -0,4 0 0</td>
<td>43 O/O: -0,4 0 0</td>
</tr>
<tr>
<td>43 U/U: 1 1,2 0 0</td>
<td>45 U/U: 0 0 0 1,1</td>
<td>43 U/U: 0 0 0 1,1</td>
</tr>
<tr>
<td>48 Y/Y: 0,3 -0,1 0 0 0 6 3</td>
<td>49 Y/Y: 0 0 0 -0,2</td>
<td>48 Y/Y: 0 0 0 -0,2</td>
</tr>
<tr>
<td>70 A/A: -1 2 -2,2 1,1 0 -0,1</td>
<td>74 A/A: 0 -0,1 0,7 0 0 1</td>
<td>70 A/A: 0 -0,1 0,7 0 0 1</td>
</tr>
<tr>
<td>65 A/A: 2,1 2,4 -0,8 0 1,1 0,7 0,79</td>
<td>64 A/A: 2,7 0 0 0 0 0</td>
<td>65 A/A: 2,7 0 0 0 0 0</td>
</tr>
<tr>
<td>115 O/O: -2 1,5 0,7 0 1,1 0,77 3</td>
<td>115 O/O: 1,4 0,75 0 0,75 0 0 -1,8 1,5</td>
<td>115 O/O: 1,4 0,75 0 0,75 0 0 -1,8 1,5</td>
</tr>
</tbody>
</table>

**FEATURE COUNT VIDEO**

| 44 E/E: 10 | 45 E/E: 9 | 44 E/E: 3 |
| 68 I/I: 9 | 74 I/I: 2 | 68 I/I: 2 |
| 43 O/O: 11 2 12 | 44 O/O: 13 4 4 | 43 O/O: 13 4 4 |
| 43 U/U: 11 10 6 8 | 45 U/U: 2 15 8 6 | 43 U/U: 2 15 8 6 |
| 48 Y/Y: 14 11 9 9 8 | 49 Y/Y: 12 11 2 12 3 | 48 Y/Y: 12 11 2 12 3 |
| 70 A/A: 9 5 9 7 9 | 74 A/A: 13 11 10 12 7 9 | 70 A/A: 13 11 10 12 7 9 |
| 65 A/A: 8 9 14 13 12 10 9 10 | 64 A/A: 8 9 14 13 12 10 9 10 | 65 A/A: 8 9 14 13 12 10 9 10 |

**FEATURE COUNT AUDIO**

| 44 E/E: 3 | 45 E/E: 2 | 44 E/E: 2 |
| 68 I/I: 2 | 74 I/I: 2 | 68 I/I: 2 |
| 43 O/O: 13 4 4 | 44 O/O: 13 4 4 | 43 O/O: 13 4 4 |
| 43 U/U: 2 15 8 6 | 45 U/U: 2 11 8 2 11 | 43 U/U: 2 11 8 2 11 |
| 48 Y/Y: 12 11 2 12 3 | 49 Y/Y: 12 11 2 12 3 | 48 Y/Y: 12 11 2 12 3 |
| 70 A/A: 13 11 10 12 7 9 | 74 A/A: 13 11 10 12 7 9 | 70 A/A: 13 11 10 12 7 9 |
| 65 A/A: 12 2 12 2 2 2 | 64 A/A: 12 11 3 11 4 8 13 7 | 65 A/A: 12 11 3 11 4 8 13 7 |

**DIFFERENCE (AUDIO) - (AUDIO+VIDEO) 20 features max**

| 44 E/E: 0 | 45 E/E: 0 | 44 E/E: 0 |
| 68 I/I: 0 | 74 I/I: 0 | 68 I/I: 0 |
| 43 O/O: 0 | 44 O/O: 0 | 43 O/O: 0 |
| 43 U/U: 0 | 45 U/U: 0 | 43 U/U: 0 |
| 48 Y/Y: 0 | 49 Y/Y: 0 | 48 Y/Y: 0 |
| 70 A/A: 9,1 0 0 0 0 | 74 A/A: 0 0 0 0 9 | 70 A/A: 9,1 0 0 0 0 |
| 65 A/A: 3,1 0 0 3,8 0 0 0 0 | 64 A/A: 3,1 0 0 3,8 0 0 0 0 | 65 A/A: 3,1 0 0 3,8 0 0 0 0 |
| 115 O/O: 0 0 0 0 0 0 0 0 18,0 | 115 O/O: 0 0 0 0 0 0 0 0 18,0 | 115 O/O: 0 0 0 0 0 0 0 0 18,0 |
### Result table 7

Error rates in percentage for Audio-only, video-only, audiovideo features. Features selection based on MRMR and GAs. 50 features maximum and 200 iterations for GAs

#### VIDEO WITH Feature Selection MRMR 50 features max

<table>
<thead>
<tr>
<th>Average Error Rate</th>
<th>(phoneme length + LV1 DCT norm)</th>
<th>Difference (MRMR 20 - GA (50,200)) VIDEO</th>
<th>Feature Count VIDEO</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/E</td>
<td>48</td>
<td>E/E: -2</td>
<td>E/E: 11</td>
</tr>
<tr>
<td>I/E</td>
<td>68</td>
<td>E/E: 8.4</td>
<td>I/E: 26</td>
</tr>
<tr>
<td>D/O/U/Y/Y/A/A/A</td>
<td>43</td>
<td>D/O: 1.1</td>
<td>D/O: 6</td>
</tr>
<tr>
<td>U/U</td>
<td>43</td>
<td>U/U: 1.5</td>
<td>U/O: 25</td>
</tr>
<tr>
<td>Y/Y/Y</td>
<td>68</td>
<td>Y/Y: 2.9</td>
<td>Y/Y: 29</td>
</tr>
<tr>
<td>A/A</td>
<td>70</td>
<td>A/A: 6</td>
<td>A/A: 20</td>
</tr>
<tr>
<td>A/A</td>
<td>63</td>
<td>A/A: 3</td>
<td>A/A: 28</td>
</tr>
<tr>
<td>A/A</td>
<td>110</td>
<td>A/A: 1</td>
<td>A/A: 18</td>
</tr>
</tbody>
</table>

#### AUDIO WITH Feature Selection MRMR 50 features max

<table>
<thead>
<tr>
<th>Average Error Rate</th>
<th>(phoneme length + LV1 DCT norm)</th>
<th>Difference (MRMR 20 - GA (50,200)) AUDIO</th>
<th>Feature Count AUDIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/E</td>
<td>45</td>
<td>E/E: 1</td>
<td>E/E: 3</td>
</tr>
<tr>
<td>I/E</td>
<td>45</td>
<td>I/E: 2</td>
<td>I/E: 2</td>
</tr>
<tr>
<td>D/O/U/Y/Y/A/A/A</td>
<td>45</td>
<td>D/O: 1.4</td>
<td>D/O: 4</td>
</tr>
<tr>
<td>U/U</td>
<td>45</td>
<td>U/U: 0</td>
<td>U/U: 2</td>
</tr>
<tr>
<td>Y/Y/Y</td>
<td>45</td>
<td>Y/Y: 2</td>
<td>Y/Y: 3</td>
</tr>
<tr>
<td>A/A</td>
<td>74</td>
<td>A/A: 3.3</td>
<td>A/A: 21</td>
</tr>
<tr>
<td>A/A</td>
<td>74</td>
<td>A/A: 0</td>
<td>A/A: 22</td>
</tr>
<tr>
<td>A/A</td>
<td>114</td>
<td>A/A: 0.75</td>
<td>A/A: 25</td>
</tr>
</tbody>
</table>

#### AUDIO&VIDEO Feature Selection MRMR 50 features max

<table>
<thead>
<tr>
<th>Average Error Rate</th>
<th>(phoneme length + LV1 DCT norm)</th>
<th>Difference (MRMR 20 - GA (50,200)) AUDIOVIDEO</th>
<th>Feature Count AUDIOVIDEO</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/E</td>
<td>64</td>
<td>E/E: 0.74</td>
<td>E/E: 3P</td>
</tr>
<tr>
<td>I/E</td>
<td>64</td>
<td>I/E: 0.74</td>
<td>I/E: R12</td>
</tr>
<tr>
<td>D/O/U/Y/Y/A/A/A</td>
<td>64</td>
<td>D/O: 0</td>
<td>D/O: 32</td>
</tr>
<tr>
<td>U/U</td>
<td>64</td>
<td>U/U: 2</td>
<td>U/O: 20</td>
</tr>
<tr>
<td>Y/Y/Y</td>
<td>64</td>
<td>Y/Y: 0</td>
<td>Y/Y: 20</td>
</tr>
<tr>
<td>A/A</td>
<td>64</td>
<td>A/A: 0</td>
<td>A/A: 91</td>
</tr>
<tr>
<td>A/A</td>
<td>64</td>
<td>A/A: 0.74</td>
<td>A/A: 92</td>
</tr>
<tr>
<td>A/A</td>
<td>114</td>
<td>A/A: 0.74</td>
<td>A/A: 810</td>
</tr>
</tbody>
</table>

#### DIFFERENCE (AUDIO) - (AUDIO+VIDEO) 50 features max

<table>
<thead>
<tr>
<th>Average Error Rate</th>
<th>(phoneme length + LV1 DCT norm)</th>
<th>Difference (MRMR 20 - GA (50,200)) AUDIO - (AUDIO+VIDEO)</th>
<th>Feature Count AUDIO - (AUDIO+VIDEO)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/E</td>
<td>44</td>
<td>E/E: 9</td>
<td>E/E: 9</td>
</tr>
<tr>
<td>I/E</td>
<td>44</td>
<td>I/E: 0.74</td>
<td>I/E: 0.74</td>
</tr>
<tr>
<td>D/O/U/Y/Y/A/A/A</td>
<td>44</td>
<td>D/O: 0.74</td>
<td>D/O: 0.74</td>
</tr>
<tr>
<td>U/U</td>
<td>44</td>
<td>U/U: 2</td>
<td>U/U: 2</td>
</tr>
<tr>
<td>Y/Y/Y</td>
<td>44</td>
<td>Y/Y: 0</td>
<td>Y/Y: 0</td>
</tr>
<tr>
<td>A/A</td>
<td>68</td>
<td>A/A: 0</td>
<td>A/A: 0</td>
</tr>
<tr>
<td>A/A</td>
<td>68</td>
<td>A/A: 0.74</td>
<td>A/A: 0.74</td>
</tr>
<tr>
<td>A/A</td>
<td>110</td>
<td>A/A: 0.74</td>
<td>A/A: 0.74</td>
</tr>
</tbody>
</table>

---

Result table 7: Error rates in percentage for Audio-only, video-only, audiovideo features. Features selection based on MRMR and GAs. 50 features maximum and 200 iterations for GAs.
**Result table 8** Error rates in percentage for Audio-only, video-only, audiovideo features. Features selection based on MRMR and GAs. 50 features maximum and 200 iterations for GAs. Speaker independent framework.

<table>
<thead>
<tr>
<th>VIDEO WITH Feature Selection MRMR 50 features max</th>
<th>AUDIO WITH Feature Selection MRMR 50 features max</th>
<th>AUDIO&amp;VIDEO Feature Selection MRMR 50 features max</th>
<th>DIFFERENCE [AUDIO] - [AUDIO+VIDEO] 50 features max</th>
</tr>
</thead>
<tbody>
<tr>
<td>NumFil = 45</td>
<td>NumFil = 45</td>
<td>NumFil = 45</td>
<td>NumFil = 45</td>
</tr>
<tr>
<td>Subject Independent</td>
<td>Subject Independent</td>
<td>Subject Independent</td>
<td>Subject Independent</td>
</tr>
<tr>
<td>EQUALIZED CLASSES</td>
<td>EQUALIZED CLASSES</td>
<td>EQUALIZED CLASSES</td>
<td>EQUALIZED CLASSES</td>
</tr>
<tr>
<td>Average ErrorRate</td>
<td>Average ErrorRate</td>
<td>Average ErrorRate</td>
<td>Average ErrorRate</td>
</tr>
<tr>
<td>(phoneme length + Lv1 DCT norm)</td>
<td>(phoneme length + Lv1 DCT norm)</td>
<td>(phoneme length + Lv1 DCT norm)</td>
<td>(phoneme length + Lv1 DCT norm)</td>
</tr>
<tr>
<td>44 E / E: 18</td>
<td>45 E / E: 0</td>
<td>44 E / E: 0</td>
<td>44 E / E: 0</td>
</tr>
<tr>
<td>68 I / I: 13</td>
<td>74 I / I: 0</td>
<td>46 I / I: 0</td>
<td>68 I / I: 0</td>
</tr>
<tr>
<td>45 U / U: 2.9</td>
<td>46 O / O: 0</td>
<td>43 O / O: -1.5</td>
<td>43 O / O: 0</td>
</tr>
<tr>
<td>48 Y / Y: 3.7</td>
<td>49 Y / Y: 0</td>
<td>48 Y / Y: 0</td>
<td>48 Y / Y: 0</td>
</tr>
<tr>
<td>70 A / A: 0.0</td>
<td>74 A / A: 0</td>
<td>50 A / A: 0</td>
<td>70 A / A: 0</td>
</tr>
<tr>
<td>63 A / A: 5.1</td>
<td>63 O / O: 14</td>
<td>63 O / O: 0</td>
<td>63 O / O: 0</td>
</tr>
<tr>
<td>110 O / O: 18</td>
<td>110 O / O: 4.6</td>
<td>110 O / O: 0</td>
<td>110 O / O: 0</td>
</tr>
<tr>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
</tr>
<tr>
<td>-5 -4.6</td>
<td>-3,1 -3,9</td>
<td>-2.2 0</td>
<td>-3,2 0</td>
</tr>
<tr>
<td>-5,8 16</td>
<td>-1,1 0 0</td>
<td>-0.7 0 0</td>
<td>-1,1 0 0</td>
</tr>
<tr>
<td>-9,2 -1,5</td>
<td>-3,1 -3,9</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>-1,5 -1,1</td>
<td>-3,2 0 1</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>-1,4 -2.2</td>
<td>-2.2 0</td>
<td>0 0 0</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>44 E / E: 0</td>
<td>46 E / E: 0</td>
<td>47 E / E: 0</td>
<td>44 E / E: 0</td>
</tr>
<tr>
<td>74 I / I: 0</td>
<td>74 I / I: 0</td>
<td>74 I / I: 0</td>
<td>74 I / I: 0</td>
</tr>
<tr>
<td>46 O / O: 0</td>
<td>46 O / O: 0</td>
<td>46 O / O: 0</td>
<td>46 O / O: 0</td>
</tr>
<tr>
<td>44 U / U: 0</td>
<td>44 U / U: 0</td>
<td>44 U / U: 0</td>
<td>44 U / U: 0</td>
</tr>
<tr>
<td>49 Y / Y: 0</td>
<td>49 Y / Y: 0</td>
<td>49 Y / Y: 0</td>
<td>49 Y / Y: 0</td>
</tr>
<tr>
<td>74 A / A: 0</td>
<td>74 A / A: 0</td>
<td>74 A / A: 0</td>
<td>74 A / A: 0</td>
</tr>
<tr>
<td>64 A / A: 2.1</td>
<td>64 A / A: 0</td>
<td>64 A / A: 0</td>
<td>64 A / A: 0</td>
</tr>
<tr>
<td>110 O / O: 0</td>
<td>110 O / O: 0</td>
<td>110 O / O: 0</td>
<td>110 O / O: 0</td>
</tr>
<tr>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
<td><strong>DIFFERENCE</strong></td>
</tr>
<tr>
<td>-1 1</td>
<td>-3,2 0 1</td>
<td>-1,1 0 0</td>
<td>-3,2 0 1</td>
</tr>
<tr>
<td>-0.7 0 0</td>
<td>0 0 0</td>
<td>-1.1 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>-0.7 0 0</td>
<td>0 0 0</td>
<td>-0.7 0 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>-1.4 -2.2</td>
<td>-2.2 0</td>
<td>-1.4 -2.2</td>
<td>-2.2 0</td>
</tr>
</tbody>
</table>