School of Engineering
Department of Mechanical Engineering

Disproving Visemes As The Basic Visual Unit Of Speech

Matthew David Ramage

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University

December 2013

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgement has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature: .....

Date: .....

Abstract

Phonemes are the standard audio unit of speech. They are the smallest segment of sound which, if replaced with another, can change the meaning of the word. In visual speech recognition, visemes have commonly been used as the basic visual unit of speech. There is a many-to-one mapping from phonemes to visemes, with the phonemes contained within a viseme considered visually indistinguishable.

While visemes are widely used, there are a number of doubts that have not been examined regarding their suitability for use within visual speech recognition. For visemes to be suitable for continued use, they must provide benefits when compared to the use of phonemes as the visual unit of speech. As image processing advances, the mouth shape can be found more accurately, capturing the more subtle visual nuances of each phoneme. As each phoneme can be better distinguished visually, the justification for creating viseme groupings diminishes.

In this thesis, a visual speech recogniser is constructed to test the validity of visemes. A novel energy method, known as "wrapping snakes", is developed to extract lip shapes from standard video datasets of people speaking. Taking this sequence of lip shapes as input, a Hidden Markov Model based recogniser is used to perform the speech recognition and output a phoneme transcript.

Examining the phoneme output of the recogniser shows that it is not possible to construct a viseme grouping that exhibits the required phoneme confusion characteristics. Some phonemes displayed very little confusion, and the remainder did not exhibit any significant clustering. The phoneme confusion was generally directional, showing that although substitutions do occur, phonemes are mostly visually distinguishable. This conclusively proves that it is phonemes, and not visemes, that should be used as the basic visual unit of speech.

Acknowledgements

To Euan, my supervisor, thank you for all the guidance you have given, and the many years you have put into helping me with this research.

I thank my colleagues and students for all your help, advice, and encouragement.

To my family and friends, thank you for all your encouragement, and keeping me sane during this time.

And last, but by no means least, to Mum and Dad. Thank you for everything you have done for me, for supporting me, and for giving me this opportunity. Without you, none of this would have been possible.

The work was supported by iVEC through the use of advanced computing resources located at the Australian Resources Research Centre (ARRC), Western Australia.

Table of Contents

List of Figures

List of Tables

Chapter 1

An Introduction To Phonemes And Visemes

Audio-visual speech recognition is the use of both audio and visual features in the process of performing speech recognition, instead of using audio alone. The key principle of visual, and audio-visual, speech recognition is that the mouth shape is based on the underlying structure of the word being spoken (Holden and Owens, 2000). There are three reasons that visual information benefits speech perception: it helps speaker localisation, it contains speech information that supplements the audio, and it provides complementary information about the place of articulation (Summerfield, 1987).

It has long been known that speech is bimodal in nature. In the 1950s it was established that seeing the talker can improve the comprehension of speech in noise by an amount equivalent to that produced by increasing the signal to noise ratio by 15dB (Sumby and Pollack, 1954). This was widely interpreted to mean that visual modality only helps with noisy audio, however a re-examination of the findings showed the benefit was apparent to all signal to noise ratios (Campbell, 2008).

The perception of speech has been shown to depend on both the audio and visual modes, as demonstrated by the *McGurk effect* (McGurk and MacDonald, 1976). McGurk and MacDonald showed that if the audio of the [ga] syllable is superimposed over the video of the [ba] syllable, most people perceive it as the [da] syllable as being spoken.

Traditional audio-only speech recognition is not always reliable, as the recognition accuracy falls dramatically as the audio signal is degraded (Dupont and Luettin, 2000), and audio is not always available. Degraded and noisy audio signals can be due to low quality recording equipment or noisy environments, such as in factories, public areas, windy environments, and automotive environments. Another common cause is multiple people speaking within range of the microphone, such as in shared office environments, or public kiosks. The inclusion of visual speech features has been shown to improve the recognition performance, not only when the audio is degraded, but also with clean audio if large vocabularies are used (Chen and Rao, 1998).

Audio speech recognition has a long history, with the basic aural unit of speech firmly established as the phoneme. Phonemes are the smallest segment of sound for which, if that segment is replaced with another, the meaning of the word changes (International Phonetics Association, 1999). The number of phonemes used for recognition, in the English language, is commonly accepted to be approximately 42 (Potamianos et al., 2004).

Visual speech recognition is a much newer field. Traditionally, visemes have been used as the basic visual unit of speech (Neti et al., 2000). Visemes are groups of phonemes that are visually indistinguishable, creating a many-to-one or many-to-many mapping from phonemes to visemes. While there is general agreement on the number and set of phonemes, there is much less agreement for visemes.

The number of visemes, and the phonemes they contain, varies wildly between researchers, with little consistency between them. While this can partially be attributed to visual speech recognition being younger than the well-established

audio speech recognition, there are still a number of reasons for doubting the validity of using visemes for visual speech recognition.

In this introduction, a history of phonemes and visemes is given, covering what they are, how they evolved, and why they have been used. This is followed by a discussion of a number of doubts regarding the validity of visemes for visual speech recognition.

1.1 Phonemes

Phonemes are the basic unit of acoustic speech. They are the smallest segment of sound for which, if that segment is replaced with another, the meaning of the word changes (International Phonetics Association, 1999). By concatenating a sequence of phonemes together, words and sentences can be formed.

1.1.1 Phonetic Alphabet

The study of phonetics and speech has a long history. The International Phonetic Association was formed in 1886 under the name of *Dhi Fonètik Tîcerz' Asòciécon* in Paris. Their aim was to encourage phonetic notation to be used in schools as a method of helping children to acquire a realistic pronunciation of foreign languages (International Phonetics Association, 1999). In 1887 development was started on a phonetic alphabet, known as the International Phonetic Alphabet (IPA), guided by six principles:

1. There should be a separate sign for each distinctive sound; that is, for each sound which, being used instead of another, in the same language, can change the meaning of a word.
2. When any sound is found in several languages, the same sign should be used. This applies also to very similar shades of sound.

3. The alphabet should consist as much as possible of the ordinary letters of the roman alphabet; as few new letters as possible being used.

4. In assigning values to the roman letters, international usage should decide.

5. The new letters should be suggestive of the sounds they represent, by their resemblance to the old ones.

6. Diacritic marks should be avoided, being trying for the eyes and troublesome to write.

The phonetic alphabet established by the Association rapidly developed, such that the alphabet issued in August 1899 is very similar to the one still in use today (International Phonetics Association, 1999). This demonstrates the long established agreement on a set of phonemes for use in describing speech in various languages.

The phonemes are typically split into two main groups consisting of consonants, and vowels. This is due to the nature of how they are formed. Consonants are any sounds in which the flow of air out of the mouth is impeded at least enough to cause a disturbance of the airflow. Vowels are sounds in which the air flows out of the mouth unimpeded (International Phonetics Association, 1999).

Consonants have traditionally been classified in terms of the place of articulation. The sound produced for each consonant phoneme is determined by this place of articulation. Figure 1.1 shows the grouping of the consonants by the place of articulation. Figure 1.2 illustrates these places of articulation along the vocal tract.

Vowels involve a less extreme narrowing of the vocal tract than consonants, and cannot be described in terms of a place of articulation. They are instead described in terms of an abstract "vowel space", which is roughly related to the position of the tongue in vowel production. Figure 1.3 shows the placement of the vowels within this space.

CONSONANTS (PULMONIC) © 2005 IPA

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	ʙ			r					ʀ		
Tap or Flap		ⱱ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

Figure 1.1: IPA chart showing consonants grouped by the place of articulation (International Phonetic Association, 2005)

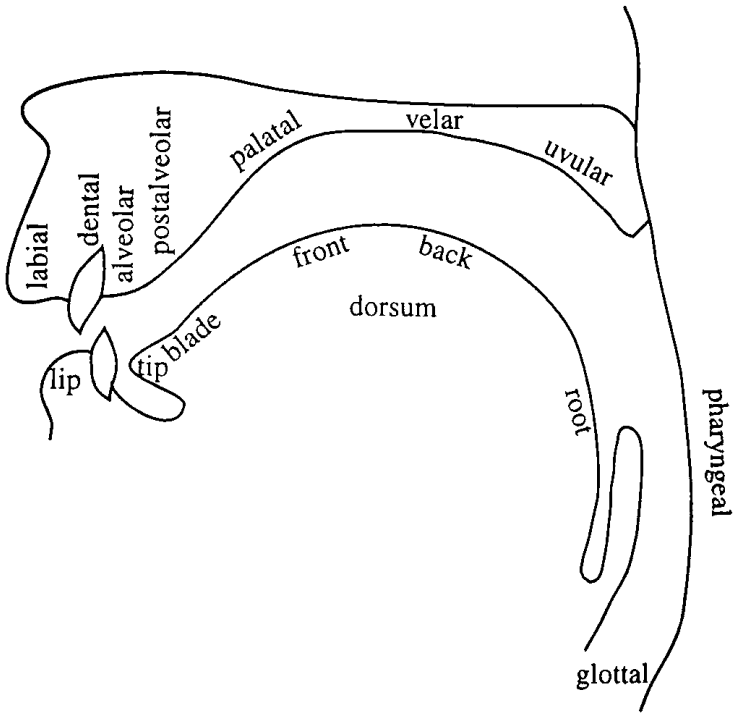


Figure 1.2: Section of the vocal tract, with places of articulation labelled (International Phonetics Association, 1999)

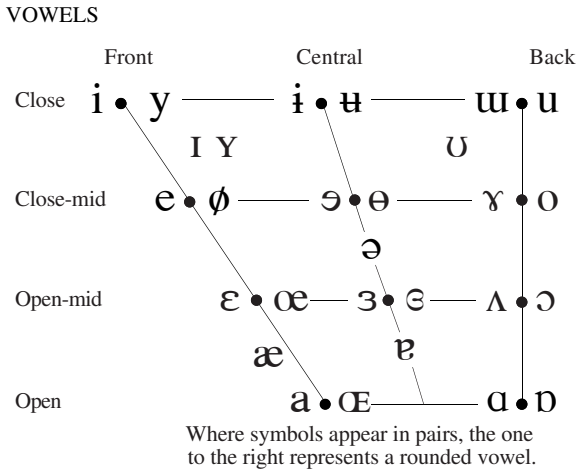


Figure 1.3: IPA chart showing the placement of vowels in the "vowel space" (International Phonetic Association, 2005)

Another phonetic alphabet is the ARPAbet (Carnegie Mellon University, 2008). It was developed for use in speech recognition software, which is traditionally limited to the ASCII character set. For this reason, the names of each phoneme in the ARPAbet is comprised of one or two ASCII letters only. This phonetic alphabet is used in the CMU Pronouncing Dictionary, which is a machine readable pronunciation dictionary containing over 125,000 words (Carnegie Mellon University, 2008). Table 1.1 shows the list of phonemes used, including the equivalent IPA symbol, along with a sample word and pronunciation.

1.1.2 Use Of Phonemes For Speech Recognition

Phonemes are almost always used as the basis for automatic speech recognition (ASR). Simple recognisers train a model for each phoneme, whereas the state of the art uses context dependent sub-phonetic models, such as triphones (Rabiner and Juang, 2008). In these recognisers, each contextual variation of a phoneme is modelled separately. The exception is for very small vocabulary systems where a model can be trained for each word in the vocabulary.

Table 1.1: Sample pronunciations from the CMU Pronouncing Dictionary, demonstrating each phoneme, including the equivalent IPA symbol (Carnegie Mellon University, 2008)

Phoneme		Example	Translation	Phoneme		Example	Translation
AA	ɑ	odd	AA D	L	l	lee	L IY
AE	æ	at	AE T	M	m	me	M IY
AH	ə	hut	HH AH T	N	n	knee	N IY
AO	ɔ	ought	AO T	NG	ŋ	ping	P IH NG
AW	au	cow	K AW	OW	o	oat	OW T
AY	aɪ	hide	HH AY D	OY	oɪ	toy	T OY
B	b	be	B IY	P	p	pee	P IY
CH	tʃ	cheese	CH IY Z	R	r	read	R IY D
D	d	dee	D IY	S	s	sea	S IY
DH	ð	thee	DH IY	SH	ʃ	she	SH IY
EH	ɛ	Ed	EH D	T	t	tea	T IY
ER	ər	hurt	HH ER T	TH	θ	theta	TH EY T AH
EY	e	ate	EY T	UH	ʊ	hood	HH UH D
F	f	fee	F IY	UW	u	two	T UW
G	g	green	G R IY N	V	v	vee	V IY
HH	h	he	HH IY	W	w	we	W IY
IH	ɪ	it	IH T	Y	y	yield	Y IY L D
IY	i	eat	IY T	Z	z	zee	Z IY
JH	ʤ	gee	JH IY	ZH	ʒ	seizure	S IY ZH ER
K	k	key	K IY				

The use of phonemes in speech recognition is long established (Rabiner and Juang, 2008). From the 1930s to the 1950s, speech research was dominated by the study of the spectral characteristics of phonemes. In this period, the relationship between sound classes and signal spectrum was firmly established (Potter, Kopp, and Kopp, 1966). In this first generation of speech recognition research, it was shown that reliable identification of the phonetic nature of a speech sound was tied to the reliable estimation of the spectral properties of the sound (Rabiner and Juang, 2008).

The second generation of speech recognition research was in the 1950s to 1960s. In this period, the focus was on using algorithmic approaches to identify phonemes based on spectral properties of the sound as a function

of time (Rabiner and Juang, 2008). By identifying the trajectories of the resonances of the vocal tract (formant regions) during vowels, simple isolated digit recognisers were built that achieved between 50% and 100% accuracy, depending on the speaker (Davis, Biddulph, and Balashek, 1952).

In this period, one of the first attempts at building a speaker independent speech recogniser was described (Forgie and Forgie, 1959). It was trained to recognise a set of 10 vowels using a 35 channel filter bank to estimate the prominent spectral regions. It achieved an accuracy of 93%, which was a significant achievement.

The next generation of speech recognition research was in the 1960s to 1980s. This generation was characterised by the use of pattern recognition for small to medium size vocabulary problems. In a review at the end of the 1960s (Hyde, 1972), several key observations and conclusions were made, including the following two points:

- Speech cannot be directly segmented into phonemes because of the importance and influence of context dependency. Hence all segmentation and labelling systems were essentially doomed without applying linguistic constraints over multiple phonemes.

- The need to utilize linguistic information in order to properly and accurately recognise fluent speech. This is basically the only way of handling the word perplexity problem and making the recognition task manageable, even for large word vocabularies (Hyde, 1972).

The fourth generation of ASR was from the 1980s to 2000s. The main development during this period was a move away from simple pattern recognition techniques, to statistical modelling frameworks. The primary methodology developed for ASR was the use of Hidden Markov Models for modelling speech dynamics and statistics in continuous speech recognition systems (Rabiner and Juang, 2008). The use of HMMs grew rapidly during this period, becoming the preferred model by the end of the 1980s. HMMs are now a

vital component of almost all speech recognition systems today (Rabiner and Juang, 2008). Typically, a model is trained for each triphone, which are context dependent phonemes. They model each context the phoneme appears in as a separate model, allowing contextual dependencies to be modelled more accurately.

HMMs also have the ability to include language models, allowing information regarding the structure of the language itself to be included in the recogniser. This can significantly improve the performance of ASR, in particular for restricted grammar scenarios, such as interactive voice prompts for automated answering services (Rabiner and Juang, 2008). In the 1990s, with the advent of more powerful computers, work began on using HMMs for continuous speaker-independent recognition (Young, 2008).

Throughout the 1990s and 2000s, the Defence Advanced Research Projects Agency (DARPA) pushed the development of ASR towards real world applications by devising a series of increasingly difficult tasks for researchers to work towards. This started with the Resource Management task to query a database of ships about the locations and properties of naval ships throughout the world (Pallett, 2003). It had a vocabulary of 1000 words, with a computer generated list of possible queries. The word error rate (WER) by the end of this task was approximately 2% (Rabiner and Juang, 2008).

The next task was the Airline Travel Information System (Pallett, 2003; Rabiner and Juang, 2008). This task had a user create travel plans via an interactive voice system. As the structure of the prompts were determined by the users, the systems had to handle out-of-vocabulary words. The task had a vocabulary of around 2500 words, and the WER was approximately 2.5%.

Another task was the North American Business task (Pallett, 2003; Rabiner and Juang, 2008). This had users speak a story from one of several sources, such as the Wall Street Journal. The task vocabulary was about 64,000 words, and the systems again had to handle out-of-vocabulary words. Another issue was new words which appeared in the news frequently as a result of the

changing events being reported day to day. The end performance for this task was a word error rate of 6.6%.

The next task was the Broadcast News task (Pallett, 2003; Rabiner and Juang, 2008). This required the systems to provide a running text transcript at the bottom of the screen for segments from television news broadcasts. The vocabulary size was about 210,000 words, and achieved word error rates of approximately 13-17%.

The Switchboard task used speech from standard telecom switchboards, containing conversations between two people over a standard telephone channel (Pallett, 2003; Rabiner and Juang, 2008). The speech was conversational, of telephone quality and bandwidth, and had a task vocabulary of about 45,000 words. The final performance was a WER of approximately 25-29%.

The final task was the Call Home task (Pallett, 2003; Rabiner and Juang, 2008). This used speech between individuals calling their home telephone number, and talking to someone from their family. The resultant speech was highly fragmented and unstructured, with a vocabulary size of about 28,000 words, and the recognition performance achieved WERs of approximately 40%.

The conclusion from the DARPA tasks is that conversational speech is significantly harder to recognise due to the unstructured nature. Another consistent finding was that more training data always improved the recognition results, which shows the capacity limit of HMMs to represent the speech for recognition purposes has not yet been reached (Rabiner and Juang, 2008).

In the current generation of ASR from 2000s onwards, the focus is on improving the robustness of systems to the affects of noise, and on the use of parallel processing to increase the recognition decision reliability (Rabiner and Juang, 2008). Some of the problems with only using audio for speech recognition are discussed in the following section.

1.1.3 Problems With Audio Only Speech Recognition

There are many problems associated with speech recognition when only audio is used. The major problem is the significant drop in recognition accuracy as the signal-to-noise ratio drops (Chen and Rao, 1998). Audio-only speech recognisers are particularly susceptible to problems with noisy audio or multiple simultaneous speakers. These conditions are very common in many everyday environments. Some examples include outdoor locations that are windy or near traffic, call centres or meetings with multiple people having discussions, and vehicle cabins where road noise, music from the stereo, and multiple people talking all combine to provide a very challenging environment for audio-only speech recognition.

The effect of acoustic noise is illustrated in Figure 1.4, which shows results obtained by Chen (2001). It is clear that as the signal to noise ratio (SNR) drops, the audio recognition accuracy drops significantly. Similar results have been presented by many other authors, for example Chen and Rao (1998), Dupont and Luettin (2000), and Potamianos et al. (2004).

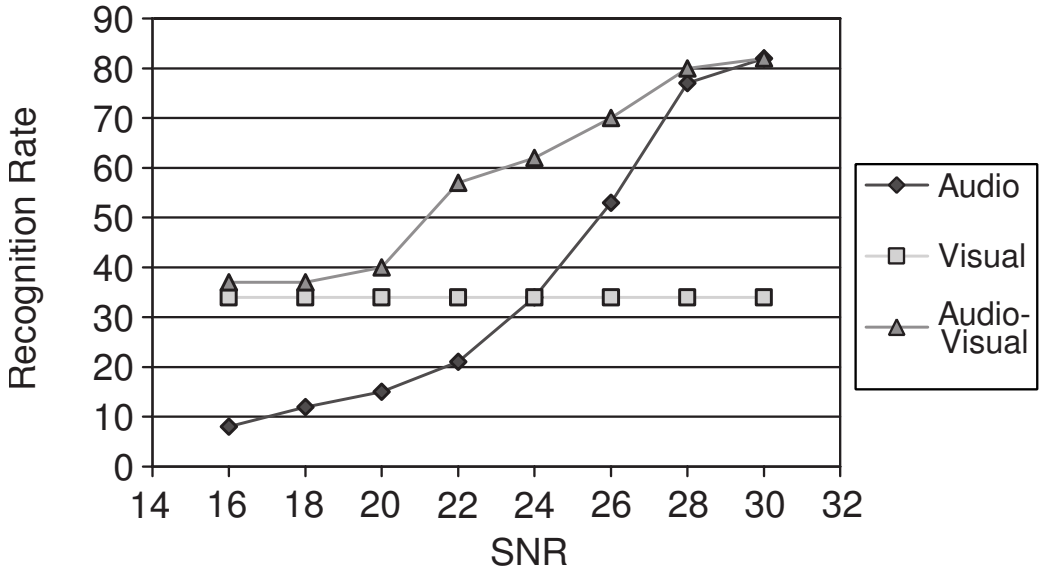


Figure 1.4: Recognition rate versus signal to noise ratio (SNR) of the acoustic signal (Chen, 2001)

The significant drop in performance in the presence of noise is a major issue for audio-only automatic speech recognition. It limits the use of ASR to relatively controlled, quiet environments, or to situations that can use predefined sentence structures with limited complexity, to prevent significant drops in recognition accuracy.

Audio-visual speech recognition is being used to improve the performance of speech recognition. In AVSR, the recogniser makes use of both the audio and visual speech information to perform the speech recognition. This has been shown to dramatically improve recognition results when compared to audio-only speech recognition (Chen, 2001; Dupont and Luettin, 2000; Hazen et al., 2004; Potamianos et al., 2004).

1.2 Complementary Nature Of Aural And Visual Components Of Speech

By performing studies of which phonemes are most easily confused aurally and visually, it has been shown that the visual component of speech is particularly useful at distinguishing between phonemes that are often confused in the acoustic component (Summerfield, 1987). This highlights the complementary nature of the aural and visual components of speech perception.

The tree in Figure 1.5 summarises the auditory confusions between consonants presented as consonant-vowel (CV) syllables in white noise (Summerfield, 1987). The vertical direction shows the SNR calculated with reference to the peak level of the vowel. The figure shows that for SNR below -18 dB, no consonants could be identified. Between -15 and -12 dB, the first major split occurs, with voiceless consonants being able to be distinguished from voiced and nasal consonants. As the SNR increases, the number of distinguishable groups of consonants increases, until all places of articulation are able to be distinguished once the SNR reaches +15 dB. (Summerfield, 1987)

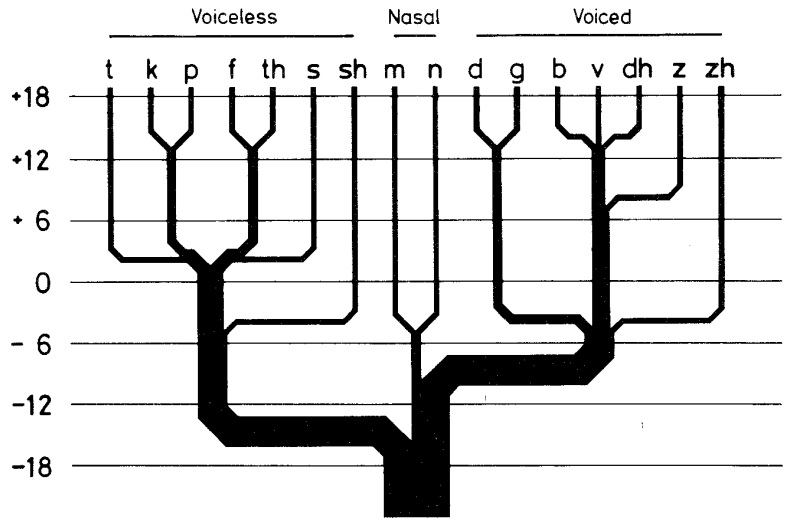


Figure 1.5: Auditory confusions among consonants presented in noise. The tree summarises consonant confusion when presented as a C-V syllable at various SNR levels (Summerfield, 1987)

The tree in Figure 1.6 shows the visual confusion among consonants presented as consonant-vowel syllables (Summerfield, 1987). It is clear that the visual confusions among the consonants are significantly different to the auditory confusions. For example, the /f/ and /v/ phonemes were found to be highly confused visually, whereas they were easily distinguished acoustically even in the presence of significant noise. In contrast, the /k/ and /p/ phonemes were very difficult to distinguish acoustically, whereas they were easily distinguishable visually.

The auditory confusions are first separated into voiced/voiceless groups of consonants, indicating these are the easiest to distinguish between using audio. In comparison, the consonants that are articulated using different external mouth features have the lowest visual confusion. For example, the /f/ and /v/ consonants are both formed using the same configuration of the lips and teeth, explaining their high visual confusion. In comparison, they have very low auditory confusion because /f/ is unvoiced, whereas /v/ is voiced.

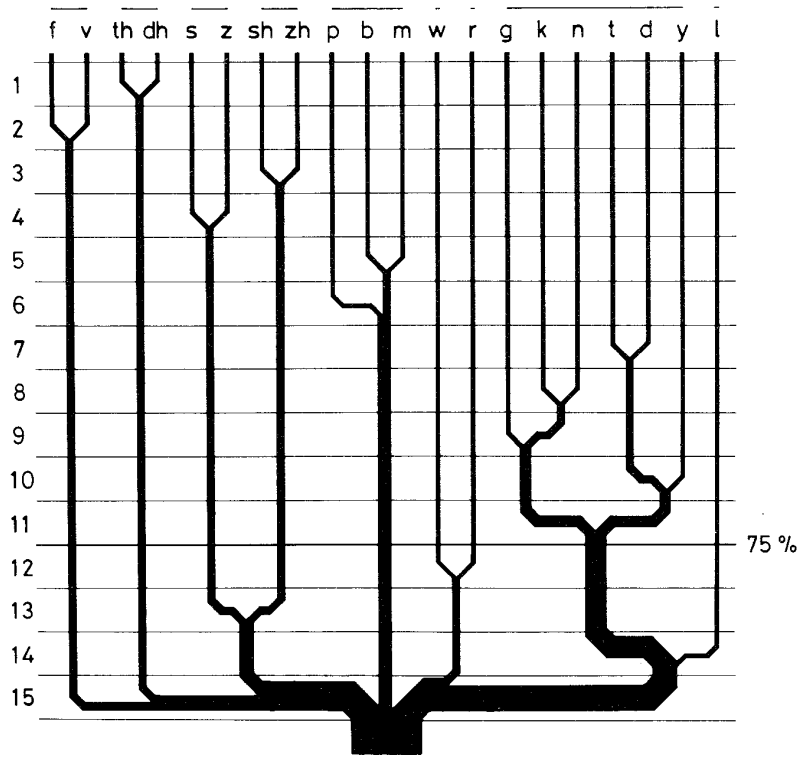


Figure 1.6: Visual confusion among consonants presented as C-V syllables. The nine groups of consonants after forming the 11th cluster can be considered to be distinct visemes – that is on 75% of presentations these were identified to belong to the same group (Summerfield, 1987)

In contrast, /k/ and /p/ are hard to confuse visually and more easily confused aurally. This is due to /k/ being formed by an open mouth and /p/ having the lips closed, making them very easy to distinguish visually. Aurally, both are voiceless, and have similar acoustic features that are easily masked by noise.

Figure 1.5 and Figure 1.6 highlight the complementary nature of the auditory and visual modes of speech, specifically showing that some consonants that are easily confused acoustically are likely to be easily distinguishable visually. As a result of this complementary nature, audio-visual speech recognition can significantly outperform audio-only speech recognition by using the

combination of aural and visual features to distinguish between otherwise similar phonemes (Chen and Rao, 1998; Hazen et al., 2004; Potamianos et al., 2004).

It has been shown that speech perception in humans is bimodal, in that we make use of both auditory and visual cues. Visual cues can supplement the auditory cues to assist in the perception of speech, even in people with normal hearing (Reisberg, McLean, and Goldfield, 1987).

If the auditory and visual cues are not consistent, we can perceive speech that is not present in either the auditory or visual cues (McGurk and MacDonald, 1976). For example, it has been shown that if a normal adult person is shown video of someone producing the [ga] syllable, while hearing the audio of the [ba] syllable, they typically report perceiving the [da] syllable (McGurk and MacDonald, 1976). This is known as the McGurk effect.

The suggested cause for this effect is that the audio contains some cues that are common to several phonemes, and the visual component contains some cues that are also common to several phonemes. If there is an overlap between these two sets, this common phoneme can be perceived by the subject when it was not actually present in either the audio or video of the speaker (McGurk and MacDonald, 1976).

This effect demonstrates the way in which speech perception in humans is bimodal, and how visual features are important in the perception of speech. It shows the human speech perception makes use of both the auditory and visual features, and the perceived speech is based on a combination of these two components. By making use of the complementary nature of the auditory and visual components of speech, speech recognition can also be made more accurate and reliable, but to do so, both audio and visual representations for sounds are required.

1.3 Origin Of Visemes

Similar to a phoneme for audio, a viseme has been defined as the smallest visually distinguishable unit of speech (Fisher, 1968). The viseme groups contain one or more auditory sounds that are visually indistinguishable. Visemes are not static lip shapes, but are sequences, or transitions, between different shapes. These movements can be captured over several frames of video. The appearance of each viseme can depend on neighbouring visemes, as the mouth moves to form each required sound (Owens and Blazek, 1985).

As the sounds of speech are formed in different parts of the mouth and throat, phonemes are not always visually distinguishable. This produces a many-to-one mapping from phonemes to visemes. For example, the /p/ and /b/ phonemes are both formed by a closed mouth shape, so they are grouped into the same viseme (Chen and Rao, 1998).

The primary characteristic of a viseme is a high level of confusion between phonemes belonging to the viseme, and a low level of confusion between phonemes belonging to different visemes (Chen and Rao, 1998; Summerfield, 1987).

The method often used to establish viseme groupings is based on the ability of human test subjects to identify consonant phonemes in vowel-consonant-vowel (VCV) or consonant-vowel-consonant (CVC) sequences. Clusters are identified in the confusion matrices, and labelled as visemes (Chen and Rao, 1998; Goldschen, Garcia, and Petajan, 1994; Owens and Blazek, 1985). In many studies, visemes were defined where the total number of responses to any group of stimuli in the matrix, divided by the total number of responses possible for these stimuli, had to be above a given threshold (Owens and Blazek, 1985). Common thresholds used include 70% (Binnie, Jackson, and Montgomery, 1976), and 75% (Owens and Blazek, 1985; Walden et al., 1981).

Unlike for phonemes, there is no generally agreed upon set of visemes, used by all researchers (Chen, 2001; Goldschen, Garcia, and Petajan, 1994; Potamianos

et al., 2004). Table 1.2 shows the viseme groups used in a number of studies. While most of these studies only looked at the consonant phonemes, some of the later studies did look at both consonant and vowel phonemes. It is clear that the viseme groups vary significantly, even between different studies performed by the same researchers.

While there is no definitive set of viseme groups, there are common trends among the 14 studies shown in Table 1.2:

- Of the 13 studies that listed both /f/ and /v/, all of them grouped these two phonemes into the same viseme, without any other phonemes. This is expected, as these two phonemes are formed in the same way, the only difference being that /f/ is unvoiced, and /v/ is voiced.
- Another common grouping is /p/ and /b/. These two phonemes are paired in all 13 studies that listed both phonemes. This group sometimes included the /m/ phoneme, as well as the occasional /em/.
- The /dh/ and /th/ phonemes are another pair often grouped together. Of the 10 studies that listed both of these phonemes, they were grouped together 9 times.
- The phonemes /sh/, /zh/, /ch/ and /jh/ are also commonly grouped together, as are /s/ and /z/.
- The remaining phonemes are much less consistent, with /t/, /d/, /j/, /k/, /g/, /n/, and /l/ sometimes being included in a "super group", sometimes also containing /s/ and /z/. Other times, these phonemes are split across multiple groups.

An interesting characteristic noted in the study that first defined a viseme (Fisher, 1968), is that not all phonemes within an identified viseme are confused in the same way, with the confusion sometimes being directional. In the study by Fisher, it was found that /n/ would often be confused for /t/, but /t/ would rarely be confused for /n/. It was also found that for

Table 1.2: A compilation of studies on visemes. Expanded from (Owens and Blazek, 1985)

(Heider and Heider, 1940)	(Woodward and Barber, 1960)	(Fisher, 1968)	(Fisher, 1968)	(Binnie, Montgomery, and Jackson, 1974)	(Binnie, Jackson, and Montgomery, 1976)
Consonants in CV pairs	*Consonants in C_1VC_1V or C_1VC_2V groups*	*Initial consonants in words*	*Final consonants in words*	*Consonants in VCV groups*	*Consonants in CV pairs*
/p, b, m/	/p, b, m/	/p, b, (m, d)/	/p, b/	/p, b, m/	/p, b, m/
/f, v/	/f, v/	/f, v/	/f, v/	/f, v/	/f, v/
/r/	/w, r, hw/	/w, hw, (r)/			/r/
					/w/
/sh, ch, jh/			/sh, zh, jh, (ch)/	/sh, zh/	/sh, zh/
/n, t, d/	/t, d, n, l, th, dh, s, z, ch, jh, sh, zh, j, k, g, h/		/(t, d, n, th, dh, s, z, r, l)/	/t, d, n, s, z, k, g/	/t, d, s, z/
/th/				/th/	/th, dh/
/k, g/					/k, g/
/l/					
( ) indicates directional confusions					

Table 1.2: (Continued) A compilation of studies on visemes. Expanded from (Owens and Blazek, 1985)

(Walden et al., 1977)	(Walden et al., 1981)	(Benguerel and Pichora-Fuller, 1982)	(Goldschen, Garcia, and Petajan, 1994)	(Chen and Rao, 1998)
Consonants in CV pairs	*Consonants in VCV groups*	*Consonants in VCV groups*	*Consonants in words*	*Consonants*
/p, b, m/	/p, b, m/	/p/	/p, b/ /bcl, m, pcl/	/p, b, m/
/f,v/	/f, v/	/f/	/f, v/	/f, v/
/w/	/w, r/	/w/	/w/	/w/
/r/			/r/	/r/
/sh, zh/	/sh, zh, ch, jh/	/ch, sh/	/ch/ /jh/ /zh/	/sh, zh/
/s, z/			/d, dcl, g, gcl, k, kcl, l, n, t, tcl/	/d, g, k, t, y/
/th, dh/	/th, dh/	/th/	/th/	/th, dh/
/t, d, n, k, g, j/			/dh, epi/	
/l/			/en/	/l/
			/ng/	
			/dx, nx, q/	
			/hh/	
			/hv/	
			/y/	

Table 1.2: (Continued) A compilation of studies on visemes. Expanded from (Owens and Blazek, 1985)

(Lucey, Martin, and Sridharan, 2004)	(Hazen et al., 2004)	(Potamianos et al., 2004)
Consonants and vowels in words	*Consonants and vowels in words*	*Consonants and vowels in words*
/p, b, m, em/	/b, p/ /bcl, pcl, m, em/	/p, b, m/
/f, v/	/f, v/	/f, v/
/w, wh, r/	/w, aw, uh, uw, ow, ao, oy/	/l, el, r, y/
	/r, er, axr/	
/ch, jh, sh, zh/	/ch, jh, sh, zh/	/sh, zh, ch, jh/
/k, g, n, l, nx, hh, y, el, en/	/t, d, th, dh, g, k/	/t, d, n, en/
		/th, dh/
	/gcl, kcl, ng/	/ng, k, g, w/
/iy, ih/	/el, l/	
/aa/	/ah, aa/	
/ah, ax, ay/	/ax, ih, iy, dx/	/ih, iy, ax/
/eh, ey, ae, aw/	/ae, eh, ay, ey, hh/	/ae, eh, ey, ay/
/uh, uw/		/uw, uh, ow/
/ao, oy, ix, ow/		/ao, ah, aa, er, oy, aw, hh/
/er/	/y/	

consonants at the start of words, the /p, b/ group also included /m/ and /d/ as directional phonemes (Fisher, 1968).

These characteristics are important, as they suggest that the visual appearance of a phoneme is not always consistent, and is dependent on the context in which it occurs. This means that phonemes that have been grouped together as a viseme may be distinguished visually in certain situations. It also shows that while a phoneme may be confused for a second phoneme, the second is not always confused for the first. This complicates the process of creating a many-to-one mapping from phonemes to visemes.

1.4 Evolution Of Viseme Groupings

A trend that has emerged is an increase in the number of viseme groups identified (see Figure 1.7). Initially, only a handful of groups were identified, but more recently the number of groups has increased significantly. This can be partially explained by the number of phonemes considered in the studies increasing over time, but this is not the only contributing factor.

The sizes of the groups have decreased in the more recent studies (see Figure 1.8), suggesting that the phonemes are now able to be better distinguished in the visual mode. A good example of this is the lack of a "super group", containing /t,d,n,s,z,g,k,l,j/, in the later studies (Chen and Rao, 1998; Goldschen, Garcia, and Petajan, 1994; Hazen et al., 2004; Lucey, Martin, and Sridharan, 2004; Potamianos et al., 2004).

Another indication that phonemes are able to be better distinguished visually, is the separation of the closure forms of /p/ and /b/ (/pcl/ and /bcl/, respectively) from the regular forms (Hazen et al., 2004). The closure form differs from the regular form in that it is the specific transition from an open to closed lip shape. Using more sophisticated techniques, these /pcl/ and /bcl/ phonemes have been categorised into a separate viseme from the

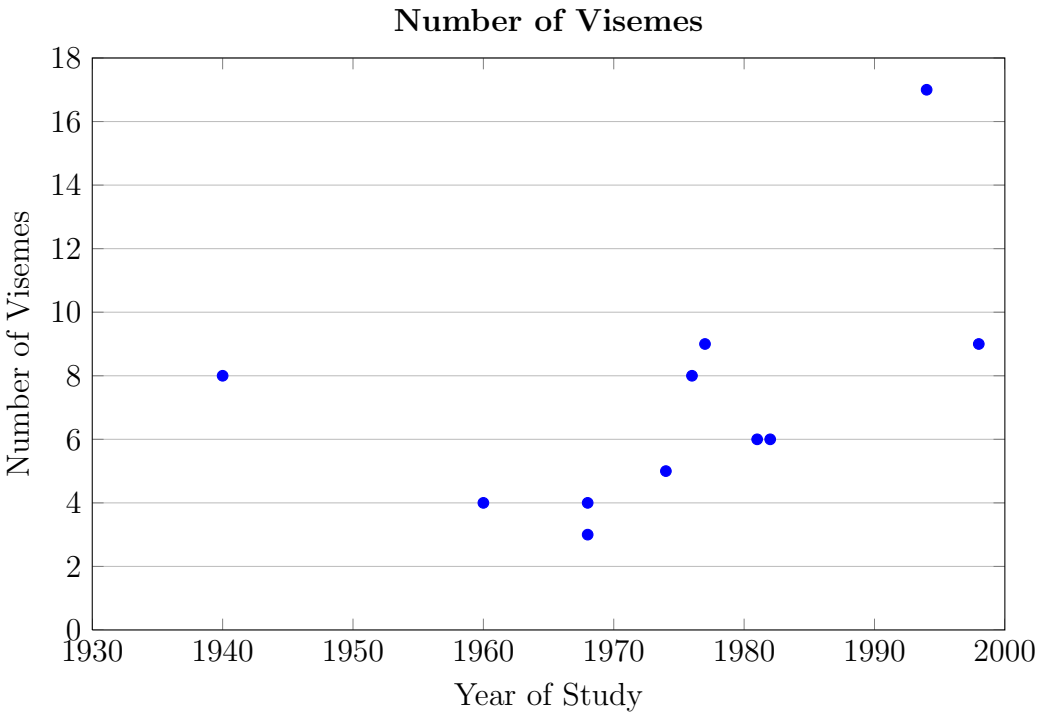


Figure 1.7: Number of visemes, for studies of consonants

/p/ and /b/ phonemes (Hazen et al., 2004). This suggests that with better recognisers, more phonemes can be successfully distinguished visually.

The current trend is for more visemes of smaller sizes. This indicates that the techniques used to recognise phonemes visually have been improving over time. The early methods of creating visemes involved using human participants and clustering their responses to identify clusters of confused phonemes. This approach is highly variable based on the abilities of the participants, which can explain the large number of phonemes assigned to each viseme.

As the availability and capabilities of computers improved, their ability to visually distinguish between phonemes has also improved. This has reduced the size of the viseme, and increased the number of visemes that can be distinguished, even being able to distinguish between different forms of the same phoneme. The rapid increase in the computational power of computers

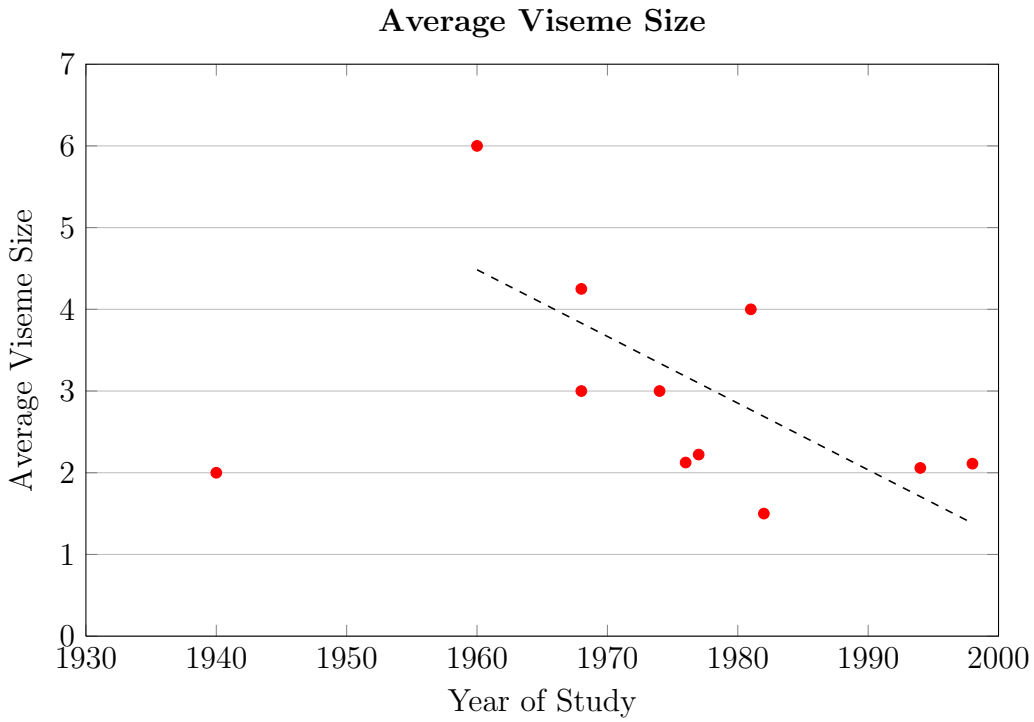


Figure 1.8: Average viseme size, for studies of consonants

has allowed for sophisticated algorithms to be used that were not feasible on earlier generations. This has allowed more complex visual features to be extracted and modelled. The increase in computing power has also allowed for larger, higher resolution datasets to be used which improves the effectiveness of training a speech recogniser.

The trend towards a larger number of smaller visemes suggests that there are subtle visual cues that are available to help visually distinguish between phonemes of generally similar appearance. As Figure 1.8 illustrates, when examining the average viseme size, it is clear that there is a downward trend, which has reached approximately two phonemes per viseme. With an average viseme size of two, there are already a number of visemes containing a single phoneme (see Table 1.2 on page 18). If this trend continues, there will be even more visemes containing single phonemes, which goes against the concept of a viseme being a group of highly visually confounded phonemes. As this trend

continues, it raises the question as to whether visemes are in fact needed at all, and if it is appropriate to create a static many-to-one mapping from phonemes to visemes.

1.5 Use Of Visemes

Visemes have been used for a number of purposes, including visual and audio-visual speech recognition, visual speech synthesis, and facial animation. When used for speech recognition, they allow a mouth shape to be mapped to a sound. In contrast, when used in visual speech synthesis and facial animation, they are used to allow a sound to be mapped to a shape. In this section, the use of visemes in each area is discussed, and the implications they may have.

1.5.1 Use Of Visemes For Speech Recognition

Visemes are often used as the visual unit of speech for audio-visual speech recognisers (Bregler et al., 1993; Hazen, 2006; Neti et al., 2000; Rogozan, 1999; Silsbee, 1994; Terry and Katsaggelos, 2008). The most common classifier used for speech recognition is the Hidden Markov Model (HMM), for both audio and visual speech recognition (Chen, 2001; Dupont and Luettin, 2000; Potamianos et al., 2004). Typically, an HMM is built and trained for each viseme, with context dependent models often being used for improved accuracy. These context dependent models allow the system to better model the variations in viseme appearance depending on preceding and following visemes.

In a study comparing phonemes and visemes as the visual speech class in an audio-visual speech recogniser, the phoneme based recogniser achieved a correctness of 64.3% and an accuracy of 39.4%, while the viseme based recogniser achieved 66.7% and 45.9% for correctness and accuracy respectively (Terry and Katsaggelos, 2008). It is interesting to note that the correctness

only varied by 2.4%, meaning that both classes resulted in a similar number of words being correctly recognised. The slightly larger difference in accuracy indicates that the use of visemes reduced the number of word insertions, when compared to the use of phonemes as the visual class.

The difference between accuracy and correctness is due to the accuracy metric penalising insertion errors, while correctness is simply a measure of how many of the inputs were correctly recognised. These metrics are calculated using the following two equations

$$Percentage\ Correct = \frac{H}{N} * 100\% \tag{1.1}$$

$$Accuracy = \frac{H - I}{N} * 100\% \tag{1.2}$$

where H is the number of correctly output labels, N is the number of labels in the reference transcription, and I is the number of insertions. These are commonly used metrics to measure performance of speech recognition systems (Young et al., 2005).

In a study comparing audio, visual, and audio-visual accuracy for the German language, it was found that the addition of visual features improved the accuracy when compared to the audio-only and visual-only recognisers, especially if noisy audio was used (Bregler et al., 1993). The performance of the recogniser for two speakers ("msm" and "mcb") is shown in Table 1.3.

Table 1.3: Comparing word accuracy for audio-only, visual-only, and audio-visual recognition, for clean and noisy audio from two speakers (Bregler et al., 1993)

Speaker	Audio Quality	Acoustic	Visual	Combined
msm	clean	88.8%	31.6%	93.2%
msm	noisy	47.2%	31.6%	75.6%
mcb	clean	97.0%	46.9%	97.2%
mcb	noisy	59.0%	46.9%	69.6%

The recogniser used a set of 65 phoneme states (including phoneme-to-phoneme transition states) for the audio component. It used a set of 42

visemes states (including viseme-to-viseme transition states) which is similar to the number of phonemes within the English language, and much larger than sets used by other researchers (see Table 1.2 on page 18).

A study using artificially coloured lips to improve lip visibility reported an increase in accuracy from 90.8% for audio-only recognition, to 95.4% for audio-visual recognition (Rogozan, 1999). For visual only recognition, a viseme accuracy of 49.3% was reported for a Hidden Markov Model based recogniser, and less than 30% when using a time-delay neural network (TDNN) (Rogozan, 1999).

Visemes are also typically used for visual-only speech recognition systems (Hazen et al., 2004; Hilder, Theobald, and Harvey, 2010; Neti et al., 2000; Potamianos et al., 2001; Revéret and Benot, 1997). In these systems, a recogniser is trained using viseme or sub-viseme (context dependent visemes) models. In visual-only recognisers, the performance is relatively low in comparison to audio recognisers. For example, Potamianos et al. (2001) reported a visual-only word accuracy of 36.5% for a connected letter task (26 word problem). Another study reported word error rates of 51% using viseme-based decision tree clustering of HMMs (Neti et al., 2000).

The Word Error Rate (WER) of visual speech recognition systems varies with the approach used, the size and quality of the corpus, and the type of recognition task, such as continuous speech or isolated digit recognition.

In a study of different visual features, a range of WERs from 58.1% to 64.0% were obtained (Potamianos et al., 2004). These results were obtained by first training HMMs using noisy audio, before being rescored using visual-only features. This significantly reduced the WER. When pure visual features were used, the WERs ranged from 89.2% to 82.3%, which is significantly worse than the rescored HMMs. These results were for speaker-independent large vocabulary continuous speech recognition (LVCSR). For a simple isolated digit recognition task, the WERs achieved were much lower at 16.8%, due to the lower complexity of the task.

In another study, the frontal and profile views were used in a speaker-dependent isolated word recognition task (Kumar, Chen, and Stern, 2007). The WERs obtained ranged from 51.45% to 76.83%. In another study, a WER of 40.3% was achieved for a multi-speaker connected digits task (Kumar, Chen, and Stern, 2007). This was a highly constrained task with the digit order always the same (digits from "0" to "9"), significantly reducing the complexity of the recognition task.

As a comparison, the word recognition rate for human lipreaders varies significantly, and is greatly enhanced by providing contextual information (Gailey, 1987). It has been reported that the average performance of adults with hearing is approximately 20–30% of words within sentences (Bernstein and Benoit, 1996).

The results obtained in a study by Gailey (1987) give accuracies of between 33% and 46% for monologues (a series of short sentences) and phrases. The conditions of these tests involved shining a 500W lamp into the speakers face to illuminate the oral cavity, while the speaker spoke without voice. By illuminating the oral cavity, the lipreader is given as much visual information as possible, as many phonemes are articulated deep in the oral cavity. These optimal viewing conditions should allow for increased recognition accuracy when compared to typical viewing conditions.

1.5.2 Integration Of Audio And Visual Features For Audio-Visual Speech Recognition

The use of visemes in audio-visual speech recognition (AVSR) complicates the audio-visual feature fusion process (Potamianos et al., 2004). In AVSR, the audio features and visual features can be combined using either early or late integration. In early integration, the audio and visual features are combined before being fed to a single audio-visual recogniser, whereas late integration uses separate audio and visual recognisers and combines the results of each to produce the final output.

In the case of early integration, phonemes are required to be used, as the recogniser is very similar to a traditional audio recogniser, but with the addition of visual features to the feature vector. With late integration, it is possible to use different classes for the acoustic and visual classifiers, as the output of each is combined to form the final output. This allows for visemes to be used as the visual classifier class, while phonemes can be used for the acoustic class. However, the use of different feature classes for the audio and visual components complicates the fusion process (Potamianos et al., 2004).

The use of late integration does not dictate using visemes as the visual unit of speech. As late integration uses the output from each classifier to determine the correct output phoneme, the classes used for the audio and visual classifiers simply have to be compatible in some way. If phonemes were to be used for the visual classifier, the output would be easily combined with the phoneme output from the audio classifier. When the output from the two classifiers are not identical, the recogniser can determine the most likely phoneme to have produced output by using knowledge of the confusions in both the audio and visual components.

Audio-visual speech recognisers typically place more weight on the audio classifier than the visual classifier. It has been shown that the lower the signal to noise ratio in the audio, the higher the weight that should be applied to the visual classifier. The proper integration of audio and visual features has been shown to always outperform audio-only speech recognition (Chen, 2001; Potamianos et al., 2003). This is illustrated in Figure 1.9.

In a set of studies reported by Potamianos et al. (2004), it has been shown that the proper integration of audio and visual features can achieve an increase in recognition performance equivalent to a 7–7.5dB increase in signal to noise ratio, when compared to the audio-only performance at 10dB SNR. The results of these studies are shown in Figure 1.10 for a large vocabulary connected speech recognition task (LVCSR), and for a simpler connected digits task (DIGITS). An important result is that all algorithms outperformed

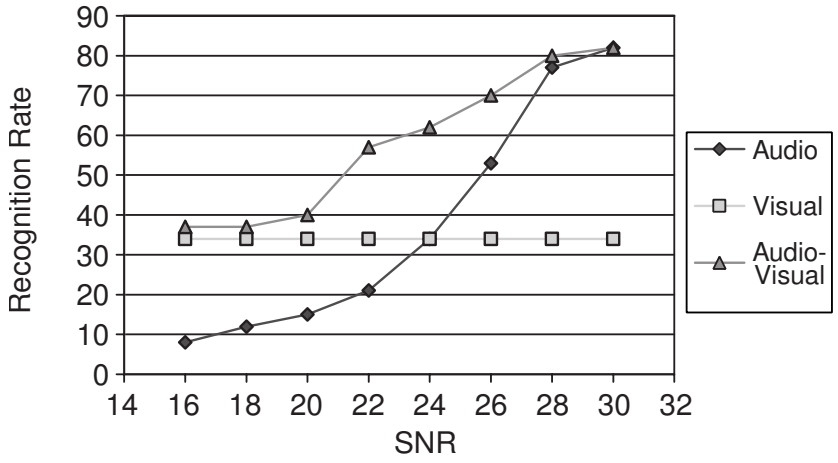


Figure 1.9: Comparing the performance of audio-only, visual-only, and audio-visual speech recognition in the presence of noisy audio (Chen, 2001)

the audio-only performance, with the margin increasing for lower signal to noise ratios.

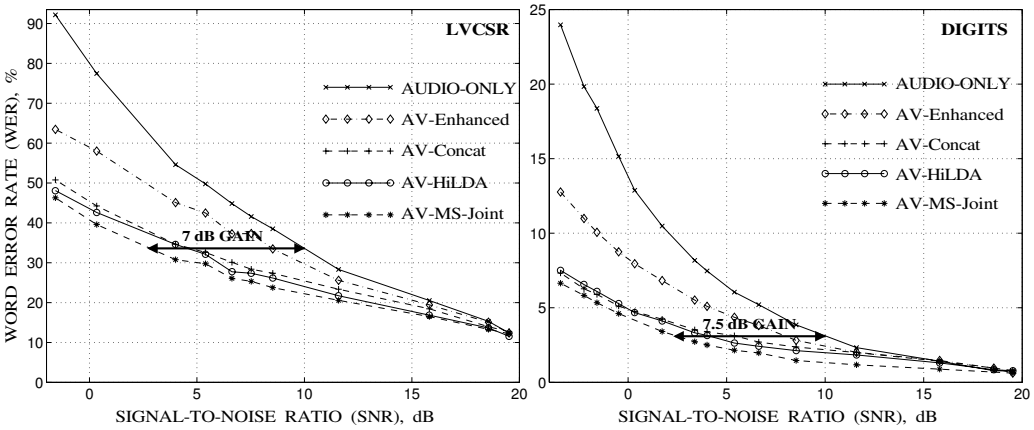


Figure 1.10: Comparing the word error rates of various feature and decision fusion algorithms for audio-visual speech recognition against audio-only recognition. The comparisons are made for both large vocabulary continuous speech recognition (LVCSR) and for a simple connected digits task (DIGITS) (Potamianos et al., 2004)

The results illustrated in Figure 1.10 show that for all algorithms, the word error rate is significantly higher in the LVCSR task, than in the much simpler DIGITS task. This is due to the highly constrained dictionary and grammar

for the DIGITS task in comparison to the LVCSR task, allowing for better recognition results.

The majority of speech recognition research involving the use of visual information has focused on audio-visual speech recognition, with visual-only speech recognition being the minority. This is due to most scenarios having audio information available for use, and the intended goal of improving existing audio-only recognisers through the use of video.

1.5.3 Other Uses Of Visemes

Visemes are not restricted to use in speech recognition. Another area where they are used is facial animation, particularly for visual speech synthesis, where a transcription or audio is used to animate a face.

The MPEG-4 standard (ISO/IEC 14496-2, 2001) contains a facial animation specification, including the definition of 14 visemes. This standard was designed to allow transmission of realistic animated faces in an interoperable way. This would allow for video conferencing without the requirement of a high bandwidth connection, as well as for use in traditional animation.

The MPEG-4 facial animation specification allows custom face models to be controlled using 84 facial feature points, as illustrated in Figure 1.11. Through the use of ten groups of facial animation paramaters (FAPs), the face can be deformed in a standardised way, allowing for portability between face models. (Ostermann, 1998).

To allow simplified animation of speech using the MPEG-4 facial animation specification, visemes are used to provide a standardised way of specifying how the face model is to be transformed to produce the desired appearance (Ostermann, 2002). Using the defined visemes, an animated face can mimic the facial movements produced while speaking, giving a realistic appearance with minimal effort in comparison to traditional animation techniques.

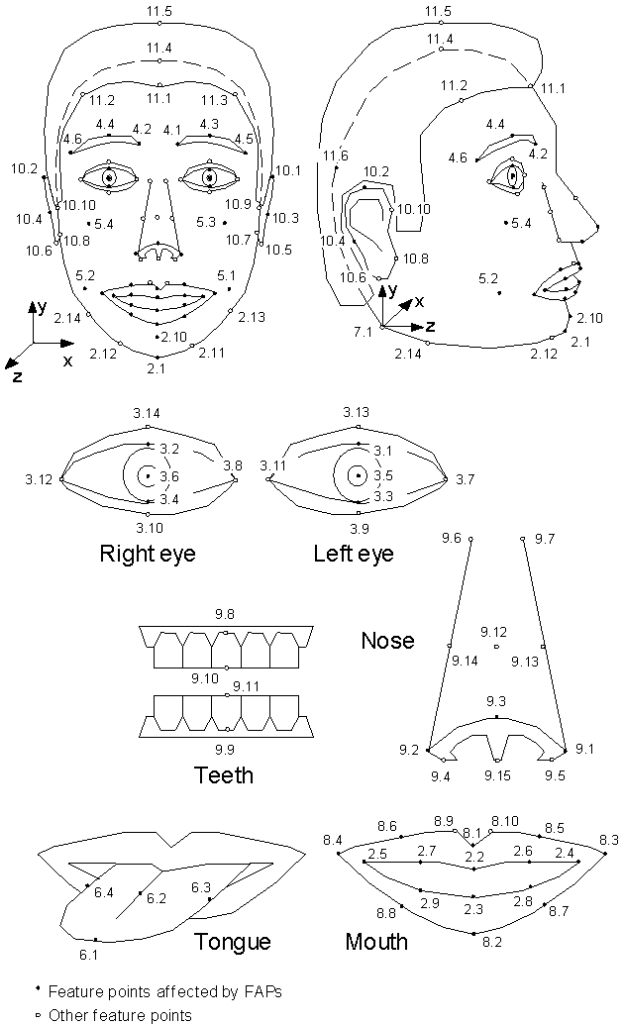


Figure 1.11: The facial animation control points supported by the MPEG-4 facial animation standard (Aleksic, Potamianos, and Katsaggelos, 2005)

The inclusion of high resolution and realistic images of the lip and mouth regions, at sufficient frame rates, has been shown to significantly improve the intelligibility of the speech (Williams et al., 1998). The MPEG-4 facial animation standard has been designed to allow the lip and mouth shape to be encoded through the use of visemes to allow for more realistic transmission and reproduction of the visual component of speech.

For the purposes of animation, a realistic mouth shape is desirable, but it does not have to be 100% accurate, as it is not being used for recognition. For example, the MikeTalk text-to-audiovisual-speech synthesiser makes use of visemes to determine the mouth shape to display (Ezzat and Poggio, 1998).

Another visual speech synthesiser is the BEAT animation toolkit (Cassell, Vilhjálmsson, and Bickmore, 2001). It uses visemes to automatically animate a virtual avatar, animated human figure, or characters within computer games, from just the text dialogue. For the intended use of the BEAT animation toolkit, a suitable estimate of the mouth shape for each viseme is sufficient.

Visemes are used to animate a real-time 3D talking head in a visual speech synthesiser created by Niswar et al. (2009). It uses a database of 17 synthetic visemes to generate sufficiently realistic mouth movements for a 3D head model. By using a cubic interpolation between visemes, the mouth movements can be animated. Figure 1.12 shows frames from an animation generated from the set of synthetic visemes, for the letters "VRCAI".

The use of visemes for facial animation allows for improved animation performance, particularly on embedded devices such as smart phones. By reducing the size of the models required for the animation, faster loading times, and smoother animation can be achieved (Danihelka and Kencl, 2010). The research investigated the effects of reducing the number of visemes from 16 to 10, by merging similar viseme models, on the speed and smoothness of the animation of a talking head. It was reported that the loading time was reduced from 18 to 8 seconds, and the frames per second increased from 5.4 FPS to 12.2 FPS for the reduced model (Danihelka and Kencl, 2010).

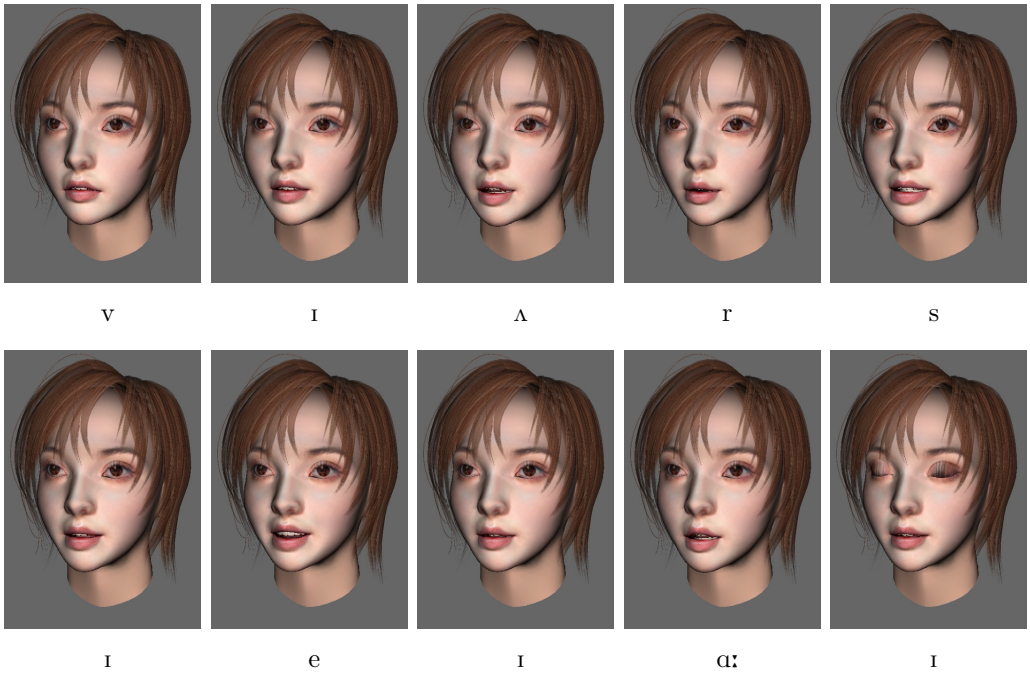


Figure 1.12: Animating the mouth movements for the letters "VRCAI", from a set of visemes (Niswar et al., 2009)

The use of visemes in animation is very different to their use in speech recognition. In visual speech synthesis, or animation, the goal is to identify a mouth shape given a sound, whereas in visual speech recognition the goal is to identify a sound given the mouth shape. As a result of these differences, the use of visemes for visual speech synthesis and animation is desirable, as a unique mouth shape is not required for every phoneme when trying to animate a virtual avatar. In contrast, the use of visemes for visual speech recognition raises a number of doubts as to their suitability. These doubts are discussed in the following section.

1.6 Doubts Regarding Visemes For Visual Speech Recognition

When considering the history of visemes, despite their many uses, there are several doubts that have not been addressed over their suitability to be used as a basic visual unit of speech. There are three main types of doubts: method, measurement, and consistency. The doubts regarding the method include inconsistency in phoneme appearance, and directionality of phoneme confusion. The doubts regarding the measurement of viseme performance include the hiding of substitution errors and rogue phonemes. Finally, there is the doubt regarding the inconsistency of viseme groupings. These five doubts are discussed in the remainder of this chapter.

1.6.1 Inconsistency In Phoneme Appearance

An issue that is not properly handled by creating a static mapping of phonemes to visemes is the inconsistency in the appearance of phonemes. One cause of visual inconsistency is due to coarticulation effects. It has been found that the appearance of some phonemes can change dramatically based on surrounding phonemes (Benguerel and Pichora-Fuller, 1982; Owens and Blazek, 1985).

When people speak, the mouth shape and movement is not simply a concatenation of individual movements used to form each phoneme. Instead, the mouth blends these movements together, with some phonemes dominating others. This is due to the place of articulation of certain phonemes being inside the mouth and throat, where the lip shape is inconsequential in the formation of the sound, while others are formed by the lips, with the internal mouth structures being inconsequential. When sequences of these phonemes are produced, the mouth structures anticipate the upcoming sounds and move into place while the previous sound is still being produced. This is not

limited to neighbouring phonemes, but can also occur due to other nearby phonemes as well (Benguerel and Pichora-Fuller, 1982).

It has been shown that certain vowel-consonant-vowel sequences have more pronounced coarticulation effects. In particular, Owens and Blazek (1985) found the /u/ vowel hides almost all articulation movements other than bilabial and labiodental. While this made the /u/ phoneme very reliable to recognise, its usefulness is counteracted by the inability to recognise surrounding consonants. As a result of this coarticulation effect, the visemes formed by grouping consonants varied based on the vowel context (see Table 1.4).

Table 1.4: Visemes associated with different vowels when presented in vowel-consonant-vowel nonsense words (Owens and Blazek, 1985)

/aa/C/aa/	/ah/C/ah/	/iy/C/iy/	/u/C/u
/p, b, m/	/p, b, m/	/p, b, m/	/p, b, m/
/f, v/	/f, v/	/f, v/	/f, v/
/th, dh/	/th, dh/	/th, dh/	
/w, r/	/w, r/	/w, r/	
/ch, jh, sh, zh/	/ch, jh, sh, zh/	/ch, jh, sh, zh/	
/k, g, n, l/	/t, d, s, z/	/t, d, s, z/	
/h/			

These coarticulation effects demonstrate the variability of the appearance of phonemes. This variability raises doubts over the suitability of viseme groups being formed using a simple many-to-one mapping of phonemes to visemes. In some contexts, a phoneme may appear similar to one group of phonemes, while in other contexts it may appear to belong in another group entirely.

1.6.2 Directionality Of Phoneme Confusion

If phonemes within a viseme are truly indistinguishable, there should be significant intra-viseme substitutions between all members of the viseme, not just some of them, and in all directions. In spite of this, even the earliest viseme research has found that confusability between consonants in a viseme

group can be directional (Fisher, 1968). In one of the first studies of visemes, the phonemes /t/ and /n/, among others, were grouped into a viseme. It was found that /n/ would often be confused for /t/, but /t/ would rarely be confused for /n/.

While there may often be confusion in one direction, the lack of confusion in the other direction would indicate that these two phonemes can in fact be distinguished in the visual domain to a significant degree. This is clearly a contradiction of the concept of the phonemes in a viseme being visually indistinguishable.

The handling of this directional confusability by grouping a large number of phonemes into a single viseme is not well suited to dealing with this type of relationship. This process results in phonemes that are able to be distinguished, being grouped into a viseme that is supposed to contain visually indistinguishable phonemes. For example, even if phoneme A is directionally confused for B and B is directionally confused for C, there is no justification for grouping A, B and C into the same viseme, as A and C are able to be distinguished.

Furthermore, grouping hides information that could be used by a later step in the process of converting sequences of phonemes into sequences of words. If this information were retained, the word/sentence constructor could make use of the information about directional confusability of phonemes when determining which words best fit the sequence of phonemes.

The nature of the confusions between phonemes can allow a word fitting algorithm to make better informed choices. For example, if a phoneme is rarely substituted for other phonemes, the word fitting algorithm can give priority to fitting words containing this trusted phoneme, and fit other words around it. It can also apply heavy penalties to words that require this phoneme to be substituted to fit.

In this same way, if a particular phoneme is almost never confused for the phoneme that occurred in the recogniser output, the word fitting algorithm

can be designed to heavily penalise any word which contains the unlikely phoneme in that position. Conversely, if a particular phoneme is known to often get confused for the phoneme that was output, the words containing that phoneme can be favoured. This would not be possible if these low level characteristics were hidden through the use of visemes which group several phonemes into a single label to be output by the recogniser. By not having this information available, the effectiveness of word fitting would be reduced for viseme-based recognisers.

1.6.3 Hiding Substitution Errors

There are three types of errors that can occur when performing speech recognition insertions, deletions, and substitutions. These errors are classified by comparing the input sequence to the responses. If an additional phoneme appears in the response, it is classified as an insertion; if a phoneme is missing in the response, it is a deletion; and if a phoneme is replaced with another in the response, it is classified as a substitution error (Young et al., 2006). Table 1.5 illustrates these types of errors for the stimulus sequence "A B C D E".

Table 1.5: Examples of the three classes of recognition errors

Situation	Phoneme sequence
Correct (equal to input)	A B C D E
Substitution	A B F D E
Deletion	A B D E
Insertion	A B C F D E

Each of these error types need to be handled in some way to minimise their effect on the performance of the speech recogniser. It is the substitution errors that visemes are trying to minimise, by grouping clusters of phonemes together into classes. Visemes reclassify intra-viseme substitution errors as being "correct", as they are now within the same class. The downside is that

the fine grained substitution characteristics are lost when multiple phonemes are grouped into a single viseme.

The problem with this approach is that if the viseme groupings change, the same sequence of phonemes can be classified as either correct or incorrect at the viseme level. For example, consider the sequence "A B F D E" in Table 1.5, in which "F" has been substituted in place of "C". If "C" and "F" are in the same class (i.e. the same viseme), the substitution of "F" in place of "C" is an intra-class substitution, and is considered the correct class. In instead, "C" and "F" are in different classes, the substitution is an inter-class error, and is considered incorrect.

This demonstrates how the use of different classes can result in an error being reclassified as correct, at the class level. By constructing viseme groups that reclassify as many phoneme substitutions as possible as being correct, the accuracy metric is artificially increased, even if those some of those phonemes are visually distinguishable.

The substitution errors can be better handled at the level of the word/sentence constructor, as it can take into account the detailed characteristics in a more refined way than simply labelling multiple confused phonemes as being equivalent. If the word fitting algorithm has detailed knowledge of how the phonemes are confused visually, it can build this in to the algorithm by applying higher costs to unlikely errors, and lower costs to more likely errors. If visemes are used, this fine grained control could not be achieved, which would result in reduced effectiveness of word fitters and sentence constructors.

1.6.4 "Rogue" Phonemes

A "rogue" phoneme is one which has many false positives for many different phonemes. It appears as a substitution for many phonemes, with no real pattern or clustering evident. Whichever viseme this phoneme appears in,

this creates situations of intra-viseme substitutions for this rogue phoneme being reclassified as "correct".

This would hide the errors that are caused by the rogue phoneme being more likely to be recognised in place of any phoneme, not just the ones within the same viseme group. The result of this misclassification would be an artificial boosting of phoneme accuracy scores, even though the overall performance has not truly increased. This is due to the reclassification of many intra-viseme substitutions as being correct, when in fact they were just due to noise.

If a phoneme was often substituted in place of many other phonemes, it should not belong to any particular viseme. The typical approach of looking only for high intra-viseme substitution errors does not take into account the number of inter-viseme substitution errors. In this situation of a rogue phoneme, if it was placed in a large viseme, it would appear to have a high number of intra-viseme substitutions, but in fact would also have a high number of inter-viseme substitution errors. In this case, it would be incorrect to group it into any viseme, as it is clearly just a noisy phoneme.

1.6.5 Lack Of Standardised Viseme Groupings

Unlike phonemes, there is no universal agreement on the number of visemes, and the mapping between phonemes and visemes (Chen, 2001; Hazen, 2006; Potamianos et al., 2004). This is demonstrated by the numerous groupings in Table 1.2 (see page 18). A lot of research has gone into identifying viseme groupings, but there is always significant variability between results, even by subjects within the same study (Owens and Blazek, 1985).

It has also been found that when people are asked to visually identify both consonants in consonant-vowel-consonant sequences, the viseme groupings for the initial and final consonants were different (Chen and Rao, 1998; Fisher, 1968). Viseme groupings identified by computers are often different to those

identified by humans, with computers being able to differentiate between phonemes that humans could not (Finn and Montgomery, 1988).

The viseme groups identified by various researchers are trending towards a larger number of smaller visemes (see Table 1.2 on page 18). Another trend is to separate different forms of the same phoneme into different visemes. For example, the closure form of the /p/ and /b/ phonemes (/pcl/ and /bcl/) have been put into a different viseme than the regular /p/ and /b/ phonemes (Goldschen, Garcia, and Petajan, 1994; Hazen et al., 2004).

If visemes are truly the basic unit of mouth movements used to produce speech, it is astounding that the majority of researchers are yet to agree upon a definitive set of viseme groups. In audio based speech recognition, there are phonemes that are often confused with each other such as /m,n/ (Owens and Blazek, 1985), yet they are still treated as distinct phonemes, allowing each to be modelled and recognised independently. If this same logic were to be applied to visual speech recognition, there would still likely be a few groups of commonly confused phonemes such as /f,v/ and /p,b/, but they would still be modelled and recognised independently.

There is a precedent for determining that visemes are not optimal for use as the visual unit of speech in audio-visual speech recognition (Hazen, 2006). The work by Hazen showed that a recogniser performed better with the use of models based on phoneme classes instead of viseme classes. While this is the same conclusion that this thesis aims prove, the prior work only goes into minor detail, and does not perform a rigorous investigation of all possible viseme sets, and the reasons for the difference in performance.

These doubts need to be addressed to determine if visemes are indeed suitable for use as a basic visual unit of speech within current visual and audio-visual speech recognition systems.

Chapter 2

The Research Question: Are Visemes The Basic Visual Unit Of Speech?

Visemes have long been used in visual and audio-visual speech recognition as the basic visual unit of speech. In spite of this, there are many assumptions that have been made regarding visemes that have not been tested. While there may have originally been justification for their use, there have been many advances in the field since, and the original assumptions have not been examined to determine if they are still valid. This has raised a number of doubts regarding the suitability of visemes for visual speech recognition, and if they are in fact the basic visual unit of speech.

2.1 Research Hypothesis

Visemes are defined as a group of phonemes that are visually indistinguishable (Fisher, 1968). If phonemes within a viseme are visually indistinguishable, there are three major characteristics which will be evident when analysing

phoneme confusions. The hypothesis put forward in this thesis is that if visemes are the basic visual unit of speech, there will exist at least one viseme set that exhibits the following three characteristics:

C1. High Ratio Of Intra-Viseme To Inter-Viseme Substitutions

C2. High Confoundedness Within Visemes

C3. Non-Directionality Of Substitutions

2.1.1 C1. High Ratio Of Intra-Viseme To Inter-Viseme Substitutions

The first characteristic of a viseme should be a significantly higher number of intra-viseme substitutions than inter-viseme substitutions. As a viseme is defined as a visually indistinguishable group of phonemes, it follows that phonemes belonging to different visemes must be distinguishable from each other. As such, substitutions between phonemes belonging to different visemes must be much less frequent than the substitutions between phonemes belonging to the same viseme. Based on the methodology of existing studies (see Section 1.3), a phoneme should be recognised within the correct viseme for at least 70% of phoneme inputs for each viseme.

2.1.2 C2. High Confoundedness Within Visemes

The second characteristic that should be exhibited is a high confoundedness between all phonemes within a viseme. All phonemes within a viseme should have significant substitution errors for all other phonemes within that viseme. If phonemes within a viseme are indeed visually indistinguishable, any given phoneme should be potentially recognised as any phoneme belonging to the

same viseme. If this characteristic is not found, it indicates that the member phonemes are able to be distinguished visually.

2.1.3 C3. Non-Directionality Of Substitutions

The third characteristic that should be exhibited by a viseme is the non-directionality of intra-viseme substitutions. If phonemes within a viseme are indistinguishable, two phonemes belonging to the same viseme should be substituted with each other a proportional number of times. There should not be a directional bias, as this would indicate the two phonemes are not entirely indistinguishable. If there is a bias, the phonemes are distinguishable in at least one direction, because one of the phonemes is rarely substituted by the other phoneme.

To test this hypothesis, the phoneme substitution characteristics need to be analysed to determine if these characteristics are present in any possible viseme grouping. The remainder of this chapter outlines a visual speech recogniser that is used to generate the data, and how it is used to test this hypothesis.

2.2 Thesis Outline

In the previous chapter, a number of doubts were identified regarding the suitability of visemes as the basic visual unit of speech. A hypothesis was formed to enable the suitability of visemes for speech recognition to be determined, by examining the three characteristics, C1–C3.

To determine the suitability of visemes, a visual speech recognition system is constructed and trained to recognise phonemes. The output from this visual speech recogniser will enable the hypothesis to be tested, by examining the exhibited characteristics. In this chapter, the design of the visual speech

recogniser is discussed, and how it is used to test the three components of the hypothesis.

2.2.1 Building A Visual Speech Recogniser To Determine The Suitability Of Visemes

The visual speech recognition system will contain three stages, with the first performing lip pixel classification, the second performing feature extraction, and the final stage performing the phoneme recognition. A block diagram of this system is shown in Figure 2.1.

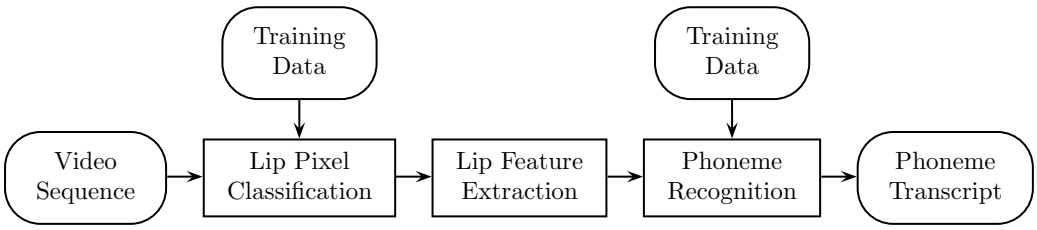


Figure 2.1: Block diagram of the visual speech recognition system

The lip pixel classification stage is fed raw colour video frames from a standard data set, and outputs a greyscale video stream, where the value of each pixel represents the likelihood of being a lip pixel. The feature extraction stage takes the greyscale lip-likeness video, and extracts the lip shape. The shape and movement of the lips are then parameterised, outputting a feature vector for each frame of video.

The phoneme recogniser is the final stage. It takes the sequences of feature vectors as an input, and uses them to calculate the most likely sequence of phonemes that could create the given sequence of feature vectors. This sequence of phonemes is then output as the recognised transcription.

The output transcript is then compared to the input sequence, to create a confusion matrix. This confusion matrix can then be analysed to determine which phonemes are distinguishable, and the nature of the substitutions,

insertions, and deletions that occur. By analysing the nature of the confusions, the hypothesis in Section 2.1 can be tested.

Chapter 3: Lip Pixel Classification

Chapter 3 details the construction of a lip pixel classifier. The task of the lip pixel classifier is to receive raw video, and identify the regions that are likely to be lips. The output is a greyscale video where the pixel intensity corresponds to the likelihood of being lips.

This task makes use of a neural network to perform the classification of each pixel. It uses colour and spatial information to determine the likelihood that a given pixel is part of the lips. This allows the lip shape to be extracted by the next stage of the recogniser.

Chapter 4: Lip Feature Extraction

Chapter 4 details the extraction of the lip feature. The task is to use the greyscale lip-likelihood video to extract the shape of the lips for each frame of video. The output from this stage is a representation of the lip shape that is suitable for use by the phoneme recognition stage.

In Chapter 4, a new active contour algorithm known as “wrapping snakes” is presented. This algorithm finds the edge of the lips, allowing the lip shape to be extracted and paramaterised. It is more robust than previous snake algorithms, and is designed to handle the specific challenges of lip shape extraction. This allows the lip shape to be found more reliably and accurately than with the traditional snake algorithm (Ramage and Lindsay, 2009).

Chapter 5: Phoneme Recognition Using Hidden Markov Models

Chapter 5 details the phoneme recogniser, which is the final stage of the visual speech recognition system. This stage takes the sequence of feature vectors containing the paramaterised lip shape for each frame of video, output by the previous stage, and outputs a phoneme transcript of the recognised speech.

The phoneme recogniser uses a standard Hidden Markov Model (HMM) design, but is configured to output at the phoneme level instead of at the word level. By comparing the phoneme output with the known input transcription, the nature of the phoneme substitutions can be examined. This will allow the hypothesis to be tested.

2.2.2 Testing The Hypothesis

To test the hypothesis, the output from the phoneme recogniser needs to be examined to determine if the characteristics in C1–C3 are present. Chapter 6 and Chapter 7 examine the output of the speech recogniser to determine if visemes are valid.

Chapter 6: Performance Of Viseme Groupings

In Chapter 6, the existing viseme groups are examined to determine if any sets exhibit the characteristics required by the hypothesis. First, the traditional viseme groups are examined in detail. Each viseme is examined to determine if it displays the required characteristics. Any phonemes which do not meet the requirements to belong to the parent viseme are split out to determine if an improved set of visemes can be created from the traditional set. Finally, existing groupings from the literature are examined to determine if they meet the required characteristics.

For a viseme grouping to satisfy the hypothesis, it must exhibit a high ratio of intra-viseme to inter-viseme substitutions, as per C1 (Section 2.1.1). It must also have a high confoundedness within each viseme, as per C2 (Section 2.1.2). The directionality characteristic is then examined in Chapter 7.

Chapter 7: Analysis Of Confusion

In Chapter 7, the confusion characteristics of individual phonemes are examined. First, the trustworthiness of each phoneme is examined. This will show the likelihood of each phoneme correctly being recognised, and which phonemes are most likely to be substituted in place of another. Once this has been done, clusters of phoneme confusion are identified. This helps determine if there is an emergent set of possible visemes that satisfy C1 and C2.

The directionality of the confusion is next to be examined. As stated in C3, phonemes belonging to a viseme should exhibit non-directional confusion, due to the visually indistinguishable nature of visemes. Finally, pairs of phonemes with both high confoundedness and low directionality of confusion are investigated. Groups of phonemes that exhibit these characteristics are ideal candidates for being grouped as a viseme.

By investigating these characteristics at the individual phoneme level, the validity of the hypothesis can be tested, not just for existing viseme groupings, but for any grouping. If the underlying characteristics are not present in the phoneme substitutions, there is no option but to reject the hypothesis.

Chapter 8: Conclusion

In Chapter 8, the results of the analysis in Chapter 6 and Chapter 7 are used to determine if visemes are truly the basic visual unit of speech, or if this hypothesis must be rejected.

The characteristics, C1–C3, are required to be evident if the hypothesis of visemes being the basic visual unit of speech is true. If these characteristics cannot be found for any grouping, the only alternative is to reject the hypothesis of visemes being groups of visually indistinguishable phonemes, forming the basic visual unit of speech.

The most important characteristic of visemes is the clustering of phonemes into multiple groups. If phonemes do not belong to groups of visemes, they will not exhibit clusters of confusion where all members are significantly confounded. The alternative is individual phonemes with different confusion characteristics for every other phoneme.

If characteristic C1 is not evident, there is no option but to reject the hypothesis of visemes being the basic visual unit of speech. The most important characteristic is the evidence of clusters of confusion between phonemes. If these clusters are not evident, the existence of visemes cannot be justified.

Even if the clustering required by C1 is not evident, C2 and C3 still need to be investigated to attempt to determine why C1 was not found. This will enable a better understanding of the visual nature of phonemes, and why visemes are not valid.

If clusters are evident, there must be confusions between all phonemes within the cluster (characteristic C2). If this is not found to be true, it would indicate some phonemes within the group can be distinguished visually from others within the same group. The absence of the characteristics in C2 would significantly reduce the strength of the justification of a set of visemes.

Finally, directional confusions within clusters (characteristic C3) would justify the rejection of visemes. Directional chains of substitutions would indicate that each phoneme has individual characteristics, and that they cannot be treated as being part of a visually indistinguishable group.

For example, consider a chain from /a/ to /b/ to /c/ that is directional in nature. In this chain, phoneme /a/ may be substituted for /b/, and /b/ may be substituted for /c/, but not vice-versa. This is not a cluster of visually indistinguishable phonemes, as the appearance of /c/ is never mis-recognised as /a/ or /b/, nor is /b/ mis-recognised as /a/.

To prove the validity of visemes, all characteristics from C1, C2, and C3 must be found to be evident for some grouping. Both existing groupings and the underlying nature of the phonemes confusion need to be examined, to determine if it is at all possible to create a set of visemes that satisfies the hypothesis. If any of these characteristics are not found, it leaves no option but to reject the hypothesis of visemes being the basic visual unit of speech.

2.3 Audio Visual Datasets Used Within This Work

The building and testing of the speech recogniser described earlier in this chapter requires large, universally available datasets for two tasks. The first is for training and testing the lip classifier and feature extractor, and the second is for training and testing the phoneme recogniser. The lip classifier needs sample data for lip and face regions, allowing the classifier to be trained to identify the lip regions within video. The phoneme recogniser needs video data of people speaking, with aligned transcriptions for each sentence.

When choosing datasets for use in visual speech recognition, there are a number of important factors that must be considered. The speech recogniser being built contains several steps with each having their own requirements to ensure adequate operation. When analysing the image for facial features, there are several factors that must be taken into account to reliably and consistently analyse an image of a given subject. These include objects obscuring the face, scale, lighting, and colour balance. Each of these factors

affects the image processing in a different way, but each must be taken into account.

In addition to these image processing requirements, there are also language content requirements for the speech recogniser. These include full coverage of all phonemes, diverse contextual occurrences for each phoneme, and transcribed labels for each sentence.

The use of standard and commonly available datasets is important, as it allows for easier comparison of results, as well as repeatability of experiments. There are many large audio data sets available for speech recognition purposes, but there are relatively few for use with visual speech recognition (Potamianos et al., 2004). By identifying the requirements, the suitability of existing datasets can be established.

2.3.1 Factors Affecting Dataset Selection

There are a number of factors that affect the image analysis that must be considered when choosing a dataset. These include obscuring objects, scale, lighting, and colour balance. Each of these factors needs to be taken into account when choosing a dataset, to ensure the dataset is suitable for use in the development and training of the lip feature extraction algorithms.

A major issue for any feature recognition algorithms is the presence of objects obscuring the target. In the case of visual speech recognition, objects obscuring the face and mouth region are of concern. For the speech recogniser being used within this research, the lip shape is used as the visual feature. If any objects obscure the lips, the recogniser may misrecognise the speech. While small occlusions may be handled by the feature extractor, large occlusions cannot feasibly be handled, as too much information is lost. The datasets are chosen to ensure that the face is visible at all times, with no objects obscuring the face.

The scale, or size, of the face within the image plays a major role in the ability of a particular technique to analyse the image for facial features. Each technique has a range of scale that it will work with, but outside this range, the results will either be unreliable or nonexistent. This is often handled by using multiple resolutions of either the image or the feature set, allowing larger and smaller scale features to be found using the one technique. Turk and Pentland (1991) suggested scaling the face images to a common size before use. In this work, datasets have been used containing images that have already been scaled to a similar size, allowing the focus to be on the suitability of viseme groups, not on designing a scale-invariant feature extractor and recogniser.

Variation in lighting conditions is one of the fundamental problems that must be dealt with for successful facial analysis. Self-shadowing of the face can result in faces appearing very different when a face is illuminated from different directions (Belhumeur, Hespanha, and Kriegman, 1997) Adini, Moses, and Ullman found that the changes induced by illumination are larger than the differences between individuals (Adini, Moses, and Ullman, 1997).

If the images within the training set are not taken under similar lighting conditions to the test set, the system may not be able to successfully locate the lips of the speaker. This issue is handled by ensuring the datasets used within this research are produced with consistent lighting. The datasets used were created in controlled studio environments, with front lighting to minimise any self shadowing.

For colour images, the colour balance can dramatically affect the appearance of the image. If a system uses colour information, it is important that a given coloured object will always appear the same colour in the image. If the colour balance is different from image to image, the system cannot reliably use the colour information in the analysis. To ensure this is not a problem, the datasets were created with consistent colour balance between each video.

These factors are problems for image processing irrespective of whether visemes are valid. As such, they have been eliminated by ensuring the data sets used within this research were filmed in controlled environments with consistent conditions. This allows the focus to be on the suitability of visemes as a basic visual unit of speech.

In addition to image analysis requirements, the speech recogniser has a number of its own requirements. The speech recogniser requires datasets exhibiting a number of characteristics to ensure adequate training and testing can be performed. To ensure the speech recogniser is trained adequately, a dataset is required that contains a phonetically balanced training set with a large number of contextually diverse samples of each phoneme. Training also requires datasets to have transcribed sentences.

The speech recognition algorithms require each phoneme to appear in a wide range of contexts. This helps ensure the phoneme models are trained on the variations caused by coarticulation. If the phonemes always appeared in the same contexts, the training process would not be able to correctly identify the transitions between consecutive phonemes, reducing the effectiveness of the training.

The varied contexts for each phoneme also allow the training to better predict the transitions between phonemes that have not been seen in that context during training. This ensures the recogniser is better able to handle new words being added to the dictionary after training has completed, and handle unstructured speech.

The training and testing processes also require transcribed, or labelled, sentences. During training, this allows the recogniser to locate the phonemes in the input stream. This is required to be able to train a model for each phoneme. During testing, the transcription is used as a reference to score the output of the recogniser to determine the accuracy of the recogniser.

In this research, two audio-visual speech datasets are used. The first is CUAVE (Patterson et al., 2002) which is used to train and develop a lip

recogniser, and the second is VidTIMIT (Sanderson and Paliwal, 2002), for use in the actual speech recognition task.

2.3.2 CUAVE

The CUAVE dataset is an audio visual speech dataset of connected and isolated digits (Patterson et al., 2002). This dataset contains 36 speakers, with an even representation of males and females, with a variety of skin tones, accents, facial hair, glasses, and hats, and was filmed with a green screen background (see Figure 2.2).

Figure 2.2: Sample frames from the CUAVE dataset, showing subjects s05f, s22m, s27m, and s36f (Sanderson and Paliwal, 2002)

The dataset is divided into two major sections, one of individual speakers, and one of pairs of speakers. For the individual section, each speaker recites the digits zero to nine, as isolated digits, five times. This is done while the speaker remains stationary, facing the camera. The speaker then recites the

digits nine to zero, as isolated digits three times. This is done while moving their face side-to-side and forward and back. The speaker then reads out a number of digits while giving the camera a profile view. The profile view is not relevant for the research in this thesis, so is not used.

The video is provided as colour MPEG2 video at a resolution of 720 x 480 pixels, at 29.97 frames per second. Manually time stamped transcriptions, at the word level, are provided for each video. Along with the video, the corpus includes one frame that has been manually labelled into face and lip regions, for each of the 36 individual speakers. These properties make this dataset well suited for use in development of the lip pixel classifier and feature extractor (see Chapter 3 and Chapter 4).

As this dataset only contains speech of the digits zero through nine, it does not provide complete coverage of all phonemes, and is not phonetically balanced. Another issue with this dataset is the sequence of digits for the frontal view video is always zero to nine, or nine to zero. This results in limited contextual variability for each phoneme, which reduces its usefulness for training a speech recogniser. Due to these limitations, this dataset was only used in the development of a lip pixel classifier and feature extractor, and not for the phoneme recogniser.

2.3.3 VidTIMIT

Another audio visual dataset is the VidTIMIT corpus (Sanderson and Paliwal, 2002). It contains 43 speakers each reciting 10 sentences from the test section of the NTIMIT corpus. The NTIMIT corpus is a large standardised audio-only speech dataset, commonly used for audio speech recognition tasks (Jankowski et al., 1990). By using sentences from the NTIMIT dataset, these sentences are known to be phonetically balanced, and ensure a good coverage and approximation to general speech. Two sentences are identical for all speakers, with the remaining 8 sentences generally being different for each speaker. In total, there are 240 different sentences, containing 1107 different words.

The video is provided as a sequence of colour JPEG frames, with a resolution of 384 x 512 pixels, recorded at a frame rate of 25 frames per second. Transcriptions are provided for each sentence, but are not time stamped.

Figure 2.3: Sample frames from the VidTIMIT dataset, showing subjects 02, 03, 26, and 31 (Sanderson, 2008)

This dataset is used to train the phoneme recogniser (see Chapter 5), as it contains a wide variety of contexts for each phoneme. This increases the ability of the recogniser to model the characteristics of each phoneme, and hence increases the performance of the recogniser.

This chapter has outlined the design of a visual speech recogniser, and how it is to be used to test the hypothesis in Section 2.1. The next three chapters detail the construction of each stage of the visual speech recogniser. This is followed by the analysis of the phoneme substitution characteristics, and finally the validity of the hypothesis is determined.

Chapter 3

Lip Pixel Classification

The first stage of a visual speech recognition system is to identify the feature being used to perform the recognition, in this case the lip shape. As such, it is first required to identify the location of the lips within the face. As the goal of this research is not to build a recogniser, but instead use the results to determine the suitability of visemes, it can be assumed that the location of the face within the video is known.

The first part of this chapter discusses the colour properties of lips, while the remainder of the chapter discusses how a neural network is used to classify lip pixels within the video frames.

3.1 Colour Properties Of Lip Pixels

To identify the lip pixels within the image, they need to be distinguished from the surrounding face pixels. This can be achieved using the colour difference between the lips and surrounding face. The task of identifying lip pixels within an image based on colour presents a significant challenge due to the colour of the lip pixels being very similar to the colour of the surrounding face pixels.

As can be seen in Figure 3.1, the lips and face pixels occupy similar regions in the RGB colour space. Each of these classes cover a large range of the red, green, and blue axis, with a generally linear relationship between each component with the red component saturating at the upper end of its range.

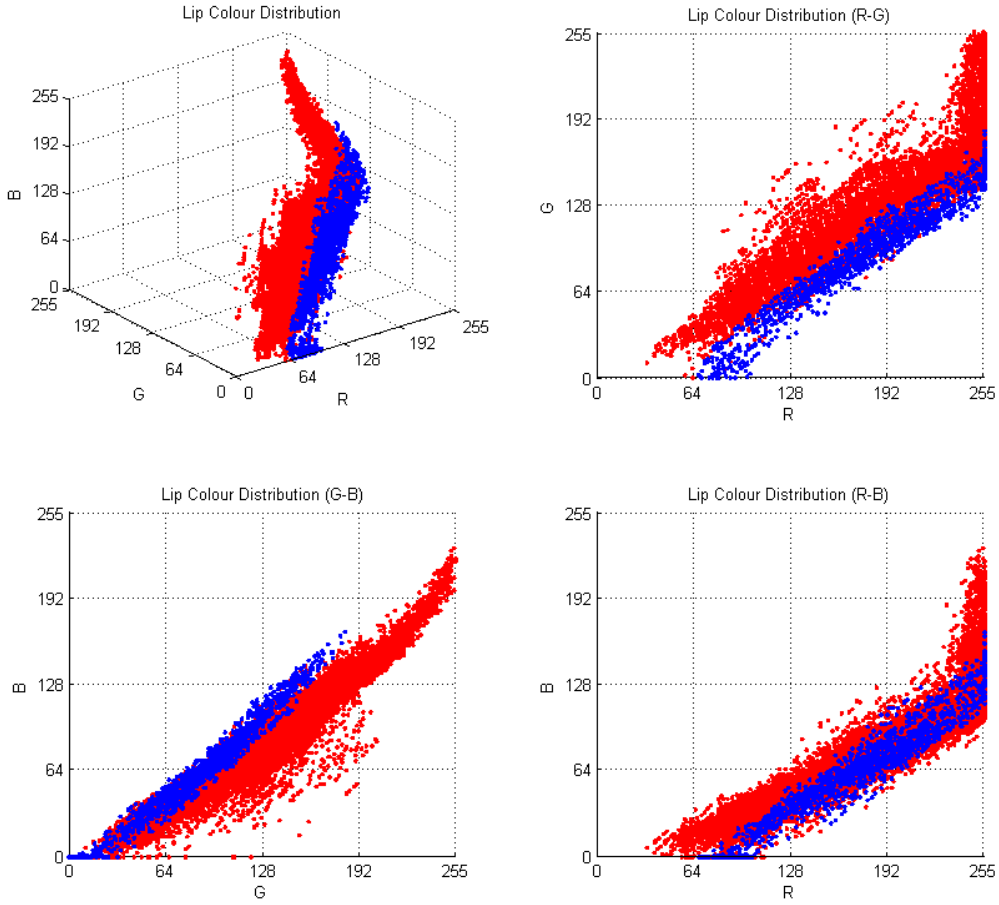


Figure 3.1: Colour distribution for lip (blue) and face (red) pixels in the RGB colour space, for subject 01 in the CUAVE dataset

As the RGB system uses red, green, and blue components, lighting variations affect all three. This means that a change in lighting conditions can significantly change the value of each component, making it very difficult to identify a particular coloured object in an image.

An alternate colour space is YCbCr. Instead of having the three axes being three colour components, this colour space uses one axis (Y) for luminance,

with the remaining two axes (Cb and Cr) containing chrominance information. As the colour information is stored independently of the luminance, changes in brightness and lighting levels do not significantly affect the chrominance.

Another advantage of using YCbCr is that it is often used as the native colour space for digital video storage (ISO/IEC 14496-10, 2004). In these cases, it removes the need to first convert the video into RGB or any other colour space. If the video is only available in the RGB colour space, the RGB values can be converted to YCbCr using Equation 3.1, where the range for R,G,B is [0, 255], the range for Y is [0, 255], and for Cb and Cr it is [14, 240] (ITU-R BT.601, 1995).

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.499 \\ 0.499 & -0.418 & -0.0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \tag{3.1}$$

As can be seen in Figure 3.2, the distribution of the lip and face pixels along the luminance axis is similar to the distribution in the RGB axes. The biggest difference between these two colour spaces is found in the chrominance plane, where a tight clustering can be seen. While the two classes (lip and face) within these distributions are clearly visible, and there is relatively tight intra-class grouping, the separation between classes is still very small, with significant overlap.

One of the main challenges in performing the classification is that the separation between classes is less than the distance that a given class can move between images. This is caused by the natural variations in skin and lip colour between people, as well as other factors as discussed in the previous section.

When the colour distribution for one subject (Figure 3.2) is compared to that of another subject (Figure 3.3), it can be seen that the colour distribution for the lips has changed slightly. This is most noticeable in the Cb-Cr (chrominance) plane. With such tight grouping of the two classes, this

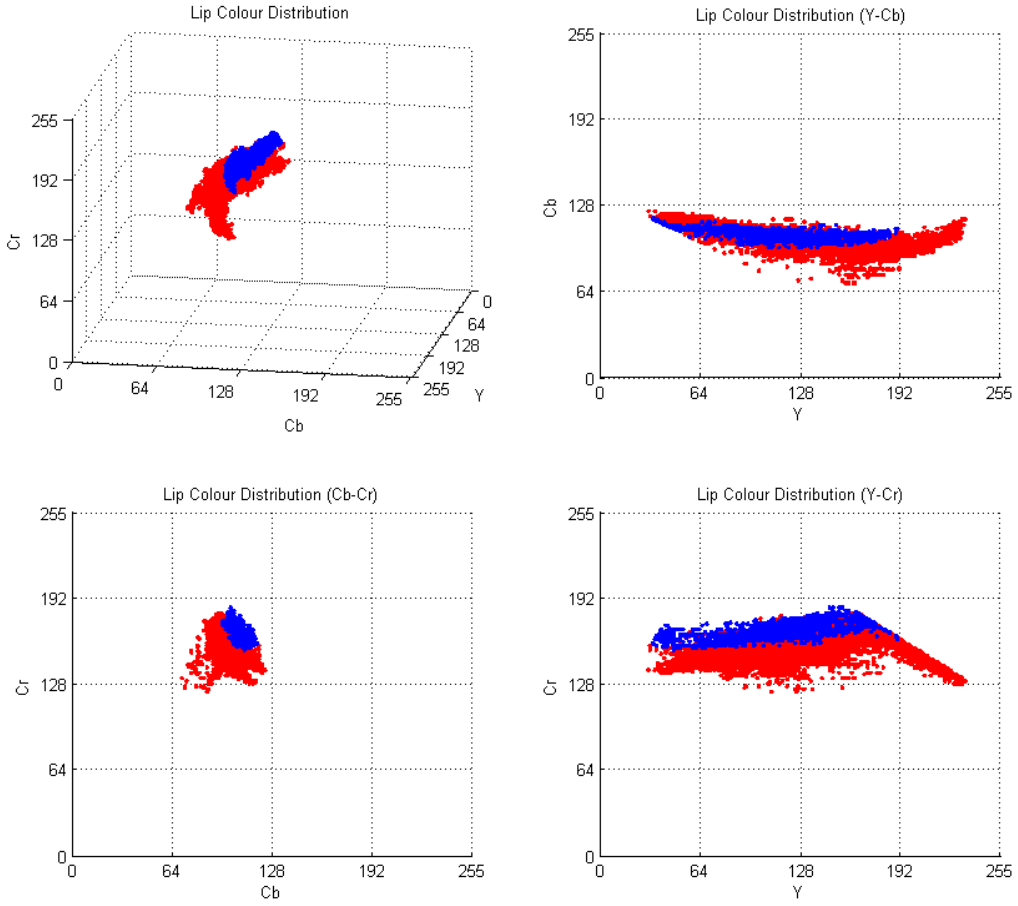


Figure 3.2: Colour distribution for lip (blue) and face (red) pixels in the YCbCr colour space, for subject 01 in the CUAVE dataset

movement of the lip class, although small, is significant when compared to the location of face class.

From these two figures, it can be seen that it is not possible to separate the two classes using a simple bisection of the YCbCr colour space. This is due to the significant overlap between the two classes, and the movement of these classes between subjects. To identify the lip pixels, a more sophisticated technique is clearly required.

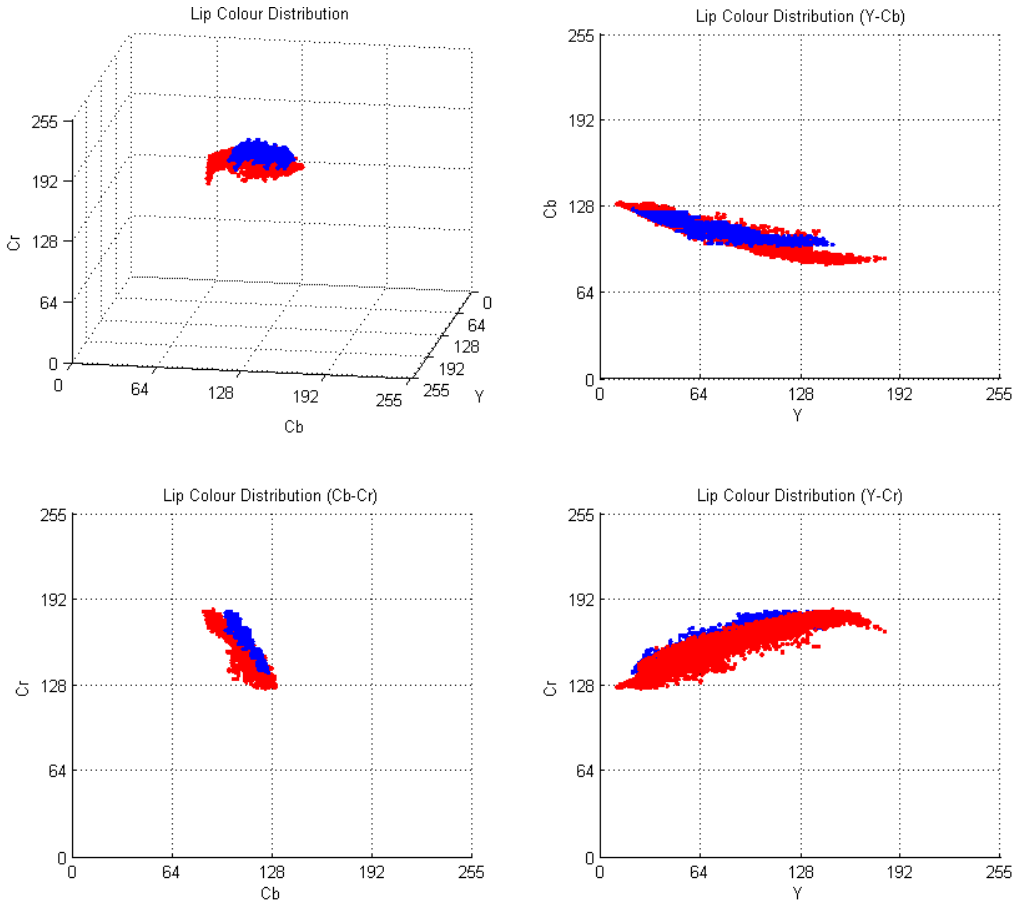


Figure 3.3: Colour distribution for lip (blue) and face (red) pixels in the YCrCb colour space, for subject 22 in the CUAVE dataset

3.2 Identifying Lip Regions Using A Neural Network

The lip regions are identified using a feed-forward neural network to determine the likelihood of each pixel belonging to the lips, using colour images. A neural network is used because it can construct nonlinear decision boundaries between different classes in a nonparametric fashion (Haykin, 1999). The power of neural networks come from their ability to learn and generalise from real data. This offers a practical solution for solving the highly complex pattern classification problem of distinguishing lip pixels from face pixels.

The type of neural network used to perform the lip classification is known as a multilayer perceptron. A multilayer perceptron is constructed of an input layer, one or more hidden layers, and an output layer, where each layer is comprised of one or more neurons.

A neuron operates by taking a weighted sum of its inputs (typically the neuron outputs from the previous layer), as well as a bias, and applying a nonlinear activation function (see Figure 3.4). By adjusting the weights of the inputs, each neuron can be trained to identify different characteristics.

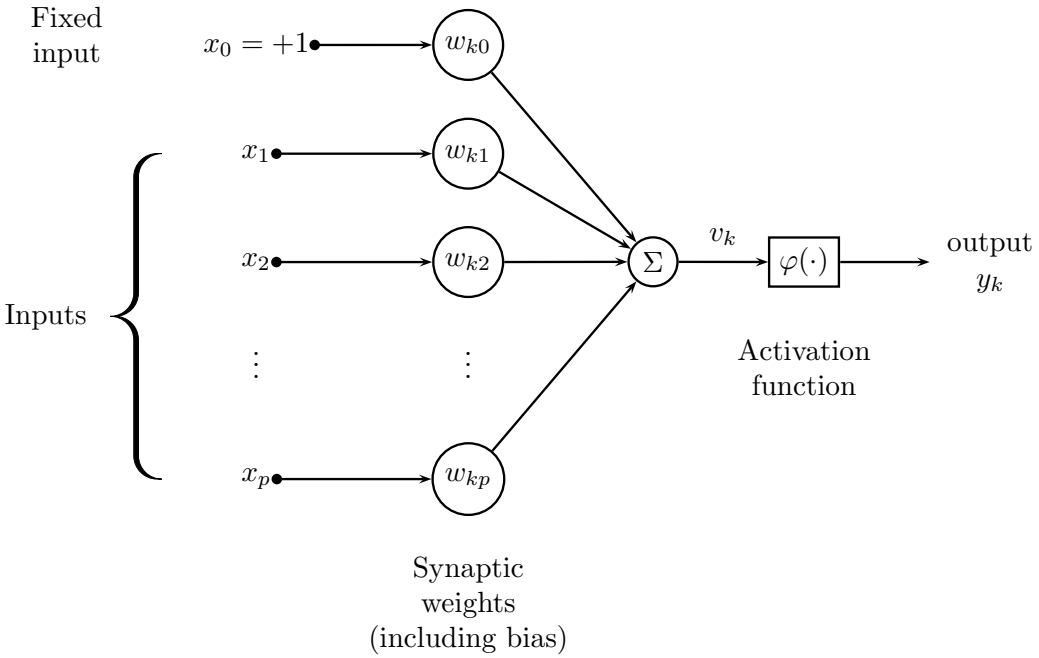


Figure 3.4: Nonlinear model of a neuron

The activation function $\varphi(\cdot)$ determines the output based on the input activity level of the neuron. It limits the amplitude of the output of the neuron to a finite range. The normalised amplitude range is typically the closed interval $[0, 1]$ or $[-1, 1]$ (Haykin, 1999). The most common form of activation function is the sigmoid function. It is defined as a strictly increasing function that exhibits smoothness (i.e. differentiable everywhere) and asymptotic properties (Haykin, 1999). An example is the tan-sigmoid function (see Figure 3.5),

defined by

$$\phi(v) = \frac{2}{1 + e^{-av}} - 1 \tag{3.2}$$

where a is the slope parameter. As the input approaches positive or negative infinity, the function becomes a threshold function, limiting the range to $[-1, 1]$.

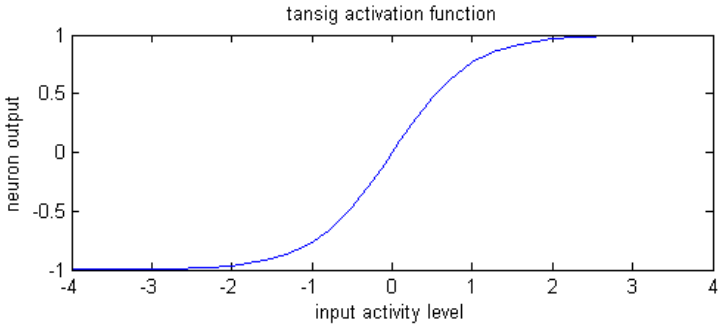


Figure 3.5: Tan-Sigmoid activation function

In the network used within this research, each neuron uses a tan-sig activation function. This is due to the training algorithm learning faster when the activation function is asymmetric than when it is nonsymmetric (Haykin, 1999). The slope parameter is given a value of 2 in Equation 3.2, resulting in the activation function being mathematically equivalent to the *tanh* function. This allows for optimised implementations of this function to be used, improving performance of the network.

The neural network used to perform the lip classification is a multilayer perceptron. The network is made of 3 tan-sigmoid layers, containing 15, 9, and 1 neuron respectively (see Figure 3.6). Each layer of neurons is fully connected with the previous layer, with each neuron obtaining an input from every neuron in the previous layer. The input layer contains the raw pixel values, and is used as the input to the first hidden layer.

As there is no definitive rule for calculating the number of neurons required for a given problem (Haykin, 1999), the number of neurons to use within each layer is determined empirically. By running tests against many different combinations, it was found that the best performing configuration consists

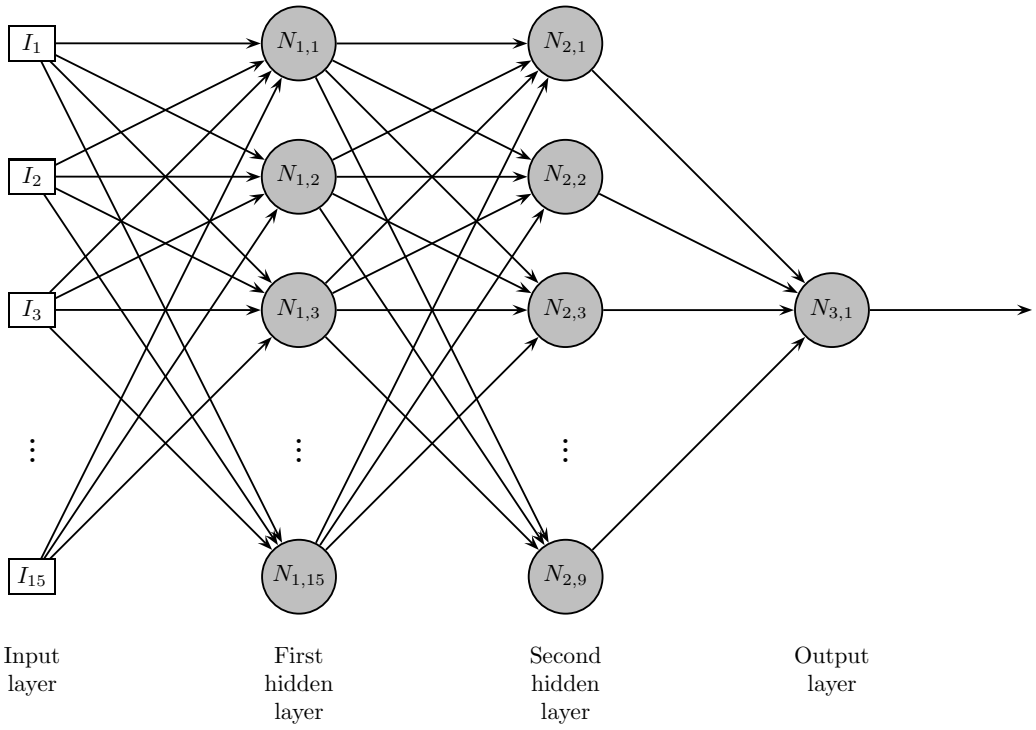


Figure 3.6: Neural network architecture consisting of 3 layers of 15, 9, and 1 neuron respectively

of two hidden layers containing 15 and 9 neurons respectively, followed by an output layer containing a single neuron. The performance data for the different numbers of neurons in each layer can be found in Appendix A.

The two hidden layers within the network act as feature detectors. Each hidden neuron can detect different localised features, with the final output layer combining these local features into a global perspective (Haykin, 1999). This allows the network to learn more complex patterns, and achieve a better performing lip pixel classifier.

The final layer, known as the output layer, contains a single neuron. The output of this neuron is the classification result for a given input. To determine the likelihood of each pixel being lips, an input vector can be constructed for each pixel location in an image. Each of these input vectors is presented to the network in turn, with the result being the likelihood of that pixel being

lips. By arranging these output values into the same geometry as the input image, a greyscale image can be formed, showing the pixels identified as likely being lips.

3.3 Input Vector

The input to the neural network is a 15 element vector. It is created by concatenating the YCbCr pixel value of the target pixel, with the YCbCr values of the four surrounding pixels being five pixels above, below, left and right of the target pixel. The YCbCr values of these additional pixels are taken after the raw frame has had a Gaussian filter applied, to ensure these pixel values represent the general colour surrounding the given pixel. This allows some spatial information to be passed to the neural network. This layout is shown in Figure 3.7.

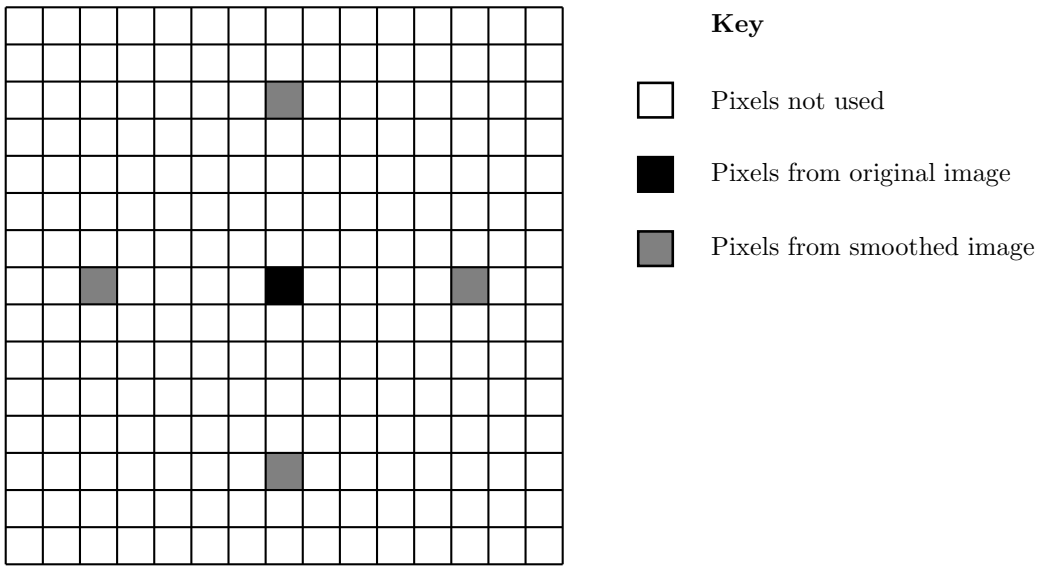


Figure 3.7: Configuration used for including spatial information in the neural network tests, showing which surrounding pixels are used to determine if the central pixel is classified as 'lips'

By including the colour data of nearby pixels from the smoothed image, the network is able to make use of spatial information in the classification process.

This allows the neural network to learn not only the colour characteristics of lip pixels themselves, but also the colour characteristics of pixels that are nearby the lips.

This additional information can allow the neural network to reject pixels that are bordered by colours that are not typically found near the lips, that is either on the face, inner mouth and teeth, or other pixels also belonging to the lips. For example, if a pixel may be the same colour as the lips, but it is bordered by a region of bright green, it is not likely to belong to the lips. This allows the neural network to better reject noise that might otherwise be classified as being a lip pixel.

As the images are provided using the standard RGB colour space, they need to be converted to YCbCr and normalised before they can be used. The RGB values are converted to YCbCr using Equation 3.1 (see 59), then dividing the result by 255. The resulting range for Y is $[0, 1]$, and for Cb and Cr it is $[0.055, 0.941]$. The data is then normalised by scaling it to the range $[-1, 1]$ for each component. This allows the data to match the output range of the tan-sig activation function.

3.4 Training The Classifier

The neural network is trained using supervised training. This requires a set of sample inputs, with corresponding target outputs. These samples must be constructed manually, by hand labelling lip and face pixels, and choosing the desired output for each. In this classifier, the lips are represented by an output of $+1$, and non-lips by an output of -1. No training data has a target output other than ± 1, as the classifier is only trained using definitive samples.

The network is only trained using lip and skin pixels, to allow it to better learn the differences between these two classes. This means that the behaviour to other inputs, such as background, inner mouth, or hair, is undefined. These

regions could generate any output, from −1 to +1, as the classifier hasn't been trained how to differentiate this data from lips or skin. This is not a problem though, as identifying the lip-face boundary in a localised region is the goal of the classifier.

It is assumed that the location of the face is known for the purposes of this classifier. As the mouth is in a known region within that face, it can be assumed that the approximate location of the mouth is known. With this in mind, most of the untrained regions will not affect the output in the localised lip region. Untrained regions, such as beards, that may affect the output in the vicinity of the lips, are expected to be handled by the lip feature extractor (see Chapter 4), and as such are not of significant concern at this stage.

The initial development of the neural network classifier used the CUAVE dataset, while the final training is performed using data from the VidTIMIT dataset (see Section 2.3), as this is the dataset used to test the speech recognition in later stages (see Chapter 5). While the CUAVE dataset does include labelled data, these have significant problems with accuracy. An example of the labelling errors is shown in Figure 3.8. The lip region in the supplied label includes the tongue and teeth, which have different colour properties to the lips. If these labels were used, the training would be less effective, as the training data itself would contain classification errors.

(a) Sample image (b) Supplied label (c) Corrected label

Figure 3.8: Example of the labelling errors in the CUAVE dataset, showing the source image (speaker 25), supplied lip label, corrected lip label

Due to these errors, the supplied labels were not used, and new labels were manually created. The new labels are available in Appendix B. Figure 3.9 shows an example of the new labels that were created. The boundaries of the new labels are much more accurate, and the number of labels has been

increased to cover 10 different categories. This allows the teeth, tongue, inner mouth and lips to each have their own label instead of being included in the lip region; it allows the neck and face to be distinguished from each other; facial hair is separated from the face region; jewellery and glasses are now separated from the face; and the eye and hair regions have also been separated from the face region.

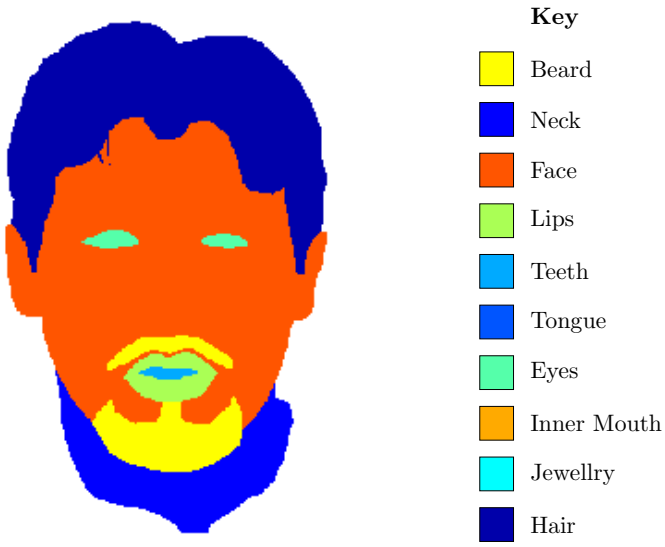


Figure 3.9: Newly created labels to replace those provided with the CUAVE dataset. Each of the individually labelled feature categories are illustrated in a different colour

These additional categories allow much more selective training of the lip classifier, as it now only needs to be trained to distinguish lips from the skin on the face. As the lips are coloured significantly differently to teeth and the inner mouth, while being similar in colour to the skin on the face, it makes sense to remove these unwanted features from the lip class.

The training used the lip region as the positive lip samples, and the skin (face and neck) regions as the negative samples. The classifier was trained using all the lip and skin data from the sample images of the first five subjects within the CUAVE dataset.

While only three of the eleven labelled regions (including "background") are used at this stage, it is possible for future work to make use of these additional regions to improve overall accuracy. For example, if the inner lip boundary was used as the lip feature, a classifier would need to be trained to distinguish lip region from the inner mouth, teeth, and tongue regions.

The network is trained offline by presenting each of the training samples to the network many times, and updating the synaptic weights using a teaching algorithm, known as resilient back-propagation, or RPROP (Riedmiller and Braun, 1993). This is a modified form of the traditional back-propagation algorithm.

The back-propagation algorithm consists of two passes through the different layers of the network - a forward pass, followed by a backward pass. During the forward pass, a sample input vector is applied, and the effect at each node is recorded layer by layer. During the backward pass, the weights are adjusted according to an error correction rule. The actual response of the network is subtracted from the target response to produce an error signal. This signal is then propagated back through the network and each synaptic weight is adjusted to minimise the error signal (Haykin, 1999).

The RPROP training algorithm modifies the standard back-propagation algorithm to better handle the nonlinearity of sigmoid activation functions. Sigmoid functions are characterised by gradients approaching zero as the input grows large in the positive or negative directions. When gradient descent is used in the standard algorithm, the very small gradients result in the weights being changed by only small amounts, even though they may be far from their optimal values. This can significantly increase the training time.

The RPROP algorithm eliminates this issue by only using the sign of the derivative to determine the direction of the weight update. The size of the change is adjusted depending on the sign of previous updates, such that successive changes in the same direction increase the size of the change, and

changes in update direction decrease the update size. This allows the training to be performed much faster than if the standard back-propagation algorithm is used, with only a minor increase in memory requirements.

The network is trained offline using a sequential approach. In this mode, a sample is presented to the network, the forward and backward calculations are performed, and the synaptic weights are adjusted. This continues until each sample within the training set has been presented once. This is known as an epoch. The process continues until the absolute rate of change of the mean squared error per epoch is sufficiently small, indicating the network has stabilised.

This is in contrast to the batch approach, where the weights are only updated after all points have been presented within the epoch. This can provide a more accurate knowledge of the error surface in weight-space, and in turn the error gradient vector that is used to calculate the updated weights, producing a faster convergence on a suitable solution. Unfortunately, this comes at the cost of requiring significantly more local storage (in the form of RAM) during training, which restricts the number of data points that can be used in training.

As a result of this major limitation in the batch mode of training, sequential training is used. By randomising the order in which the samples are presented during each epoch, the use of sequential updating of weights makes the search in weight space stochastic in nature. This in turn makes it less likely for the back-propagation algorithm to be trapped in local minima (Haykin, 1999).

Figure 3.10 illustrates the performance of the network improving during the training process. When the network is initialised with random weights at epoch zero, the mean squared error (MSE) is very high at 2.37. This is expected as the network has not undergone any training at this stage. When training commences, the MSE immediately drops to 0.12, then quickly drops to approximately 0.04 by epoch 30. The training then slows down, with the MSE slowly dropping below 0.02 after epoch 120, and slowly reducing to

0.016 by epoch 200. By this stage, minimal improvements will be made in comparison to the computational resources required. As such, training is stopped after 200 epochs. At this point, the network is ready to be used to perform the classification task.

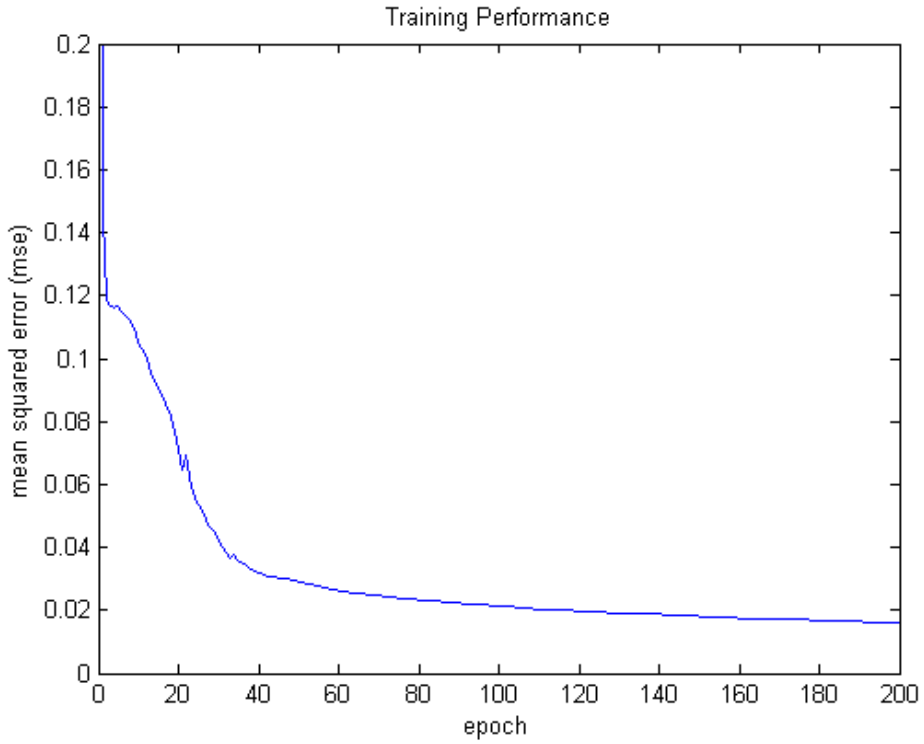


Figure 3.10: Classifier performance during training

3.5 Performance Of Classifier

With the neural network trained to recognise lip pixels, its performance can now be tested. This is done by constructing the input vector for each pixel, feeding it through the classifier, and using the output as a greyscale value for the corresponding pixel in the output image. While the output of the neural network is a value ranging from -1 to $+1$, indicating non-lips and lips respectively, this is scaled to a range from 0 to $+1$. This allows a greyscale

image to be created for each frame, with the value indicating the likeness to lip pixels.

The neural network was successful in identifying the lip regions for all of the subjects in the CUAVE dataset. Figure 3.11 to Figure 3.12 show typical results for the neural network classifier, using the sample frames from Figure 2.2. For clarity, the output images have been inverted in this thesis, so white pixels indicate areas with a value of 0 (not lips), while black pixels indicate an output of +1 (lips).

The lip region in each of these images has been successfully identified, but there are also other features that have been erroneously identified. These errors are due to issues such as clothing colour (Figure 3.12(b)) and shadowing of the skin (Figure 3.11(b) and Figure 3.12(a)) appearing similar to lip pixels. The clothing is not an issue, as the classifier is designed to differentiate between skin and lips only. As discussed previously, the location of the face can be assumed to be known, allowing the regions of the image containing clothing to be avoided.

The false positive classification errors caused by shadows are of particular interest, as they frequently occur close to the lips due to shadows beneath the nose. These are also strong false positives, which can potentially cause problems when trying to identify the lip boundary. This strong noise needs to be taken into account when extracting the lip features.

Figure 3.13 shows the classification result for the mouth region of CUAVE subject s05f, overlayed over the original image. This figure shows that the lip classifier successfully identified the majority of the lips, with minimal false positives. The lower right corner of the mouth has been incorrectly classified as not being lips, due to JPEG compression artefacts affecting the colour of these pixels. The classifier successfully identified the teeth and inner mouth as not being part of the lips.

Figure 3.14 shows the classifier output for subject s36f overlayed over the original image. The lower lip is well classified, with only a small region of

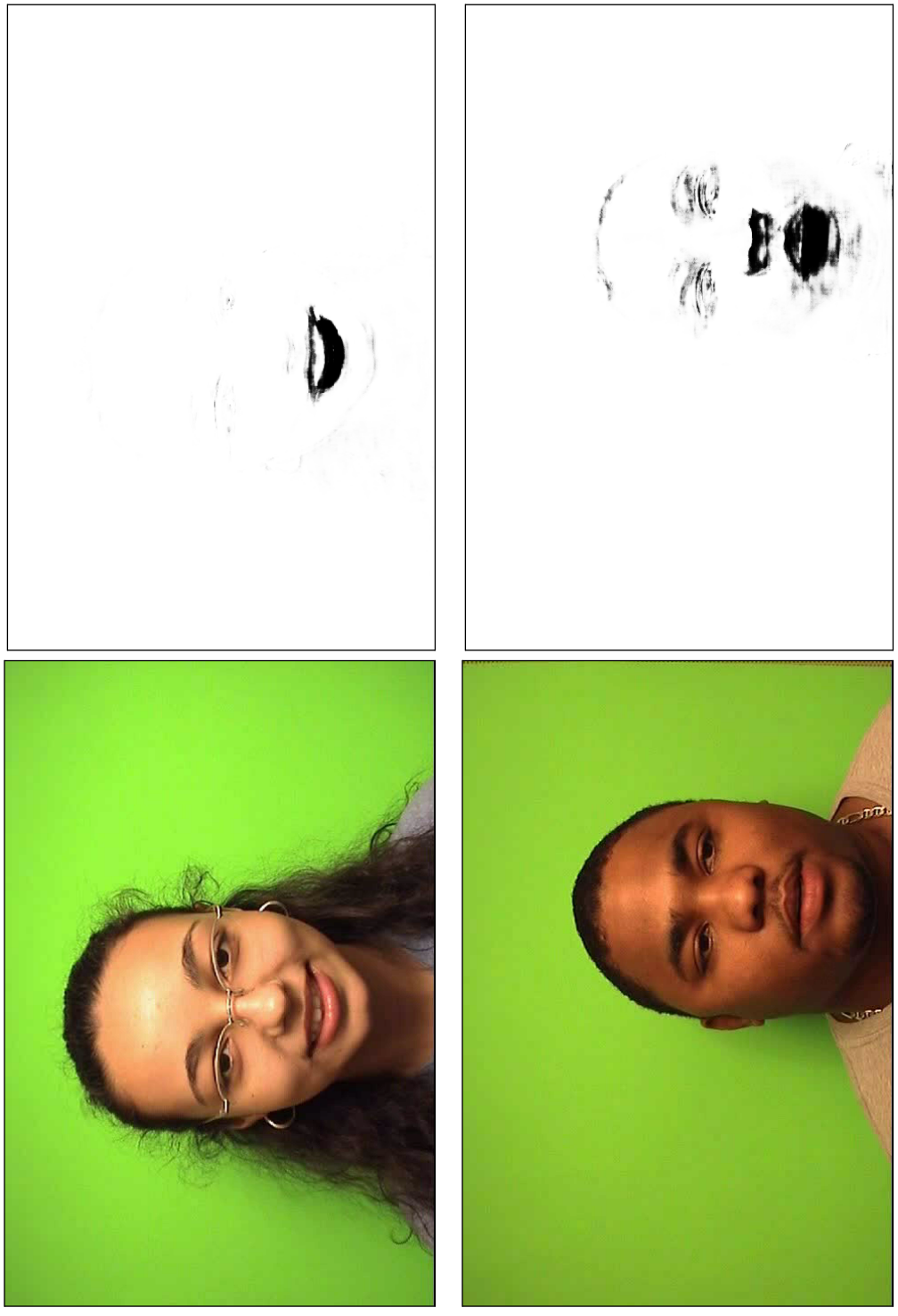


Figure 3.11: Lip classification results for CUAVE subjects (a) s05f and (b) s22m

Figure 3.12: Lip classification results for CUAVE subjects (a) s27m and (b) s36f

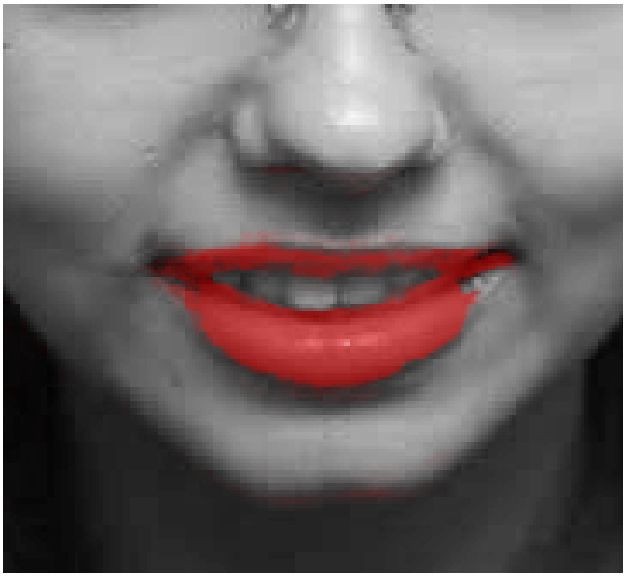


Figure 3.13: Comparing the classified lip region to the original image for CUAVE subject s05f. Areas of red indicate the pixels identified as lips

false negatives along the lower right edge of the lip. The upper lip is not as well classified, with a false positive above the centre of the upper lip (Cupid's bow), and false negatives at the outer edges of the upper lip. The false negative on the left side of the upper lip is caused by JPEG compression artefacts. While the upper lip is not as well classified as the lower lip, it is still within acceptable limits. There are very few false positives near the lips, with only a small false positive on the left nostril, and another very small region at the top of the chin.

Figure 3.15 shows the classifier output for subject s28f overlayed over the original image. The lower lip is classified very accurately, while the upper lip has a large false negative region to the right of centre. This figure illustrates the strong noise caused by shadowing, particularly beneath the nose. There are also significant false positives due to shadowing beneath the chin.

The neural network was also tested on the VidTIMIT dataset. Figure 3.16 to Figure 3.18 show a range of results obtained for the VidTIMIT dataset.

Figure 3.14: Comparing the classified lip region to the original image for CUAVE subject s36f

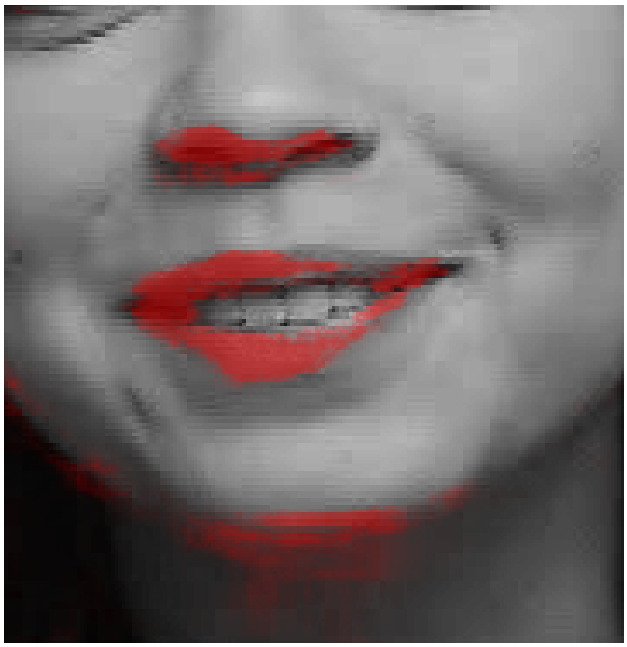


Figure 3.15: Comparing the classified lip region to the original image for CUAVE subject s28f

As demonstrated by these figures, while the lips were successfully identified for the majority of subjects, the algorithm was not always successful.

The lip region was successfully identified for the majority of the VidTIMIT subjects. It was noted that the subjects, for whom the lips were not correctly identified, were of Indian appearance. Figure 3.19 illustrates the classification results for another subject of Indian appearance (VidTIMIT subject 37). It can be seen that the small portions of the lips that were correctly classified are typically the highlights and edges. This shows how the range of colours present in this image are just outside the range identified by the neural network. It is suspected that this is due to the small number of training sample available for subjects of this appearance. As the goal of this research is to determine the suitability of visemes as a visual unit of speech, it was decided that it was acceptable to exclude these videos in the later stages.

As with the CUAVE dataset, shadows are frequently the cause of false positives near the lips for the VidTIMIT dataset. Figure 3.20 shows the identified lip regions for VidTIMIT subject 38, as output by the classifier. The upper lip is well defined with accurate edges, and the lower lip, while not as well defined, is still successfully identified. There are two significant false positive regions in this figure, both caused by the shadowing at the nostrils. While these regions are close to the lips, they are not connected to the lips themselves. These false positive regions will need to be discarded by the lip feature extractor.

Figure 3.21 shows the identified lip regions for VidTIMIT subject 09. The classifier performed very well, successfully identifying the upper and lower lips with minimal false negatives. The teeth were correctly classified as not belonging to the lips, however the inner mouth was incorrectly classified as lips. As with subject 38, the shadowing of the nostrils caused two small false positive regions.

Figure 3.22 and Figure 3.23 illustrate how facial hair affects the ability of the classifier to successfully locate the lips. In Figure 3.22, the classifier was not

Figure 3.16: Lip classification results for VidTIMIT subjects (a) 01 and (b) 09

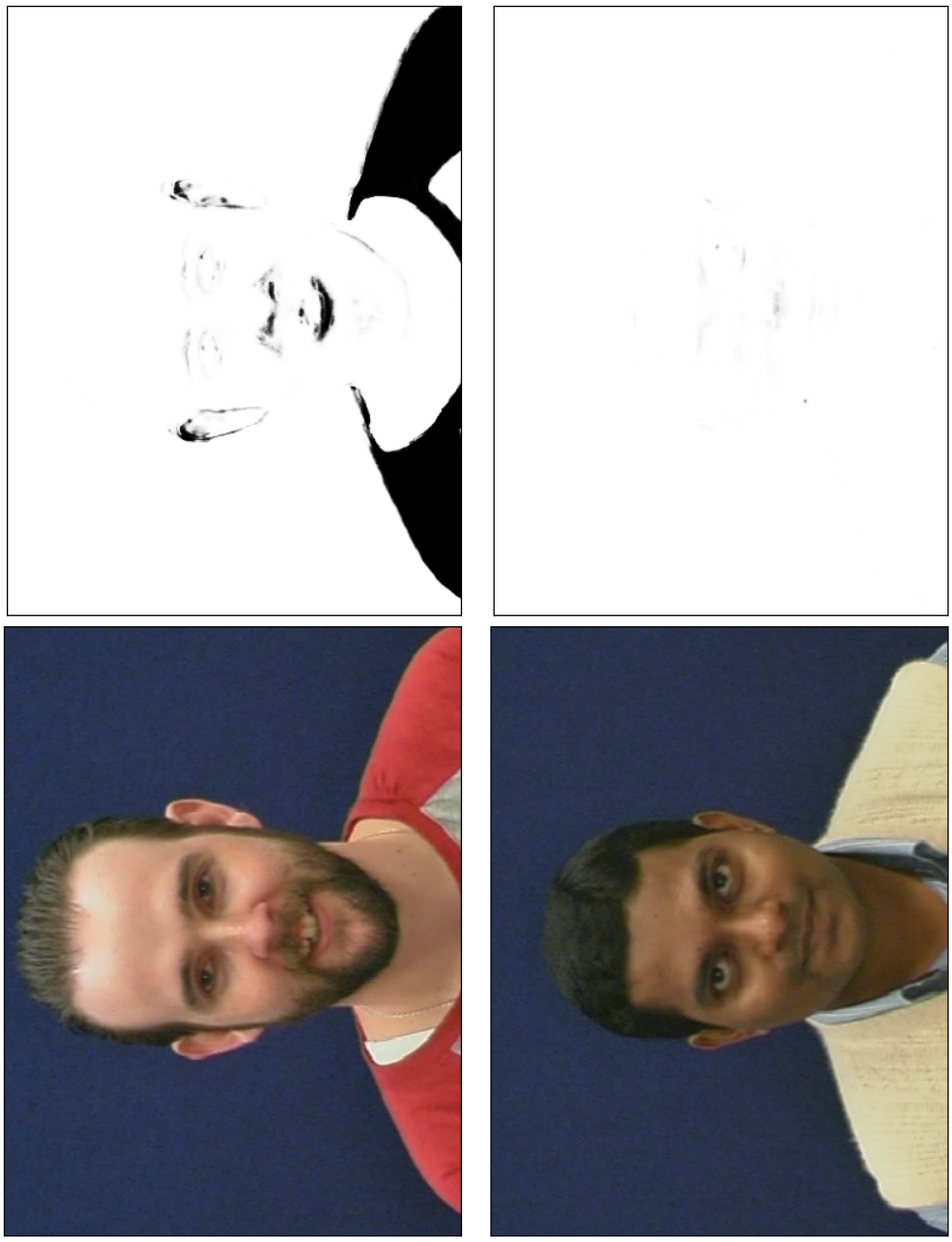


Figure 3.17: Lip classification results for VidTIMIT subjects (a) 26 and (b) 31

Figure 3.18: Lip classification results for VidTIMIT subject 38

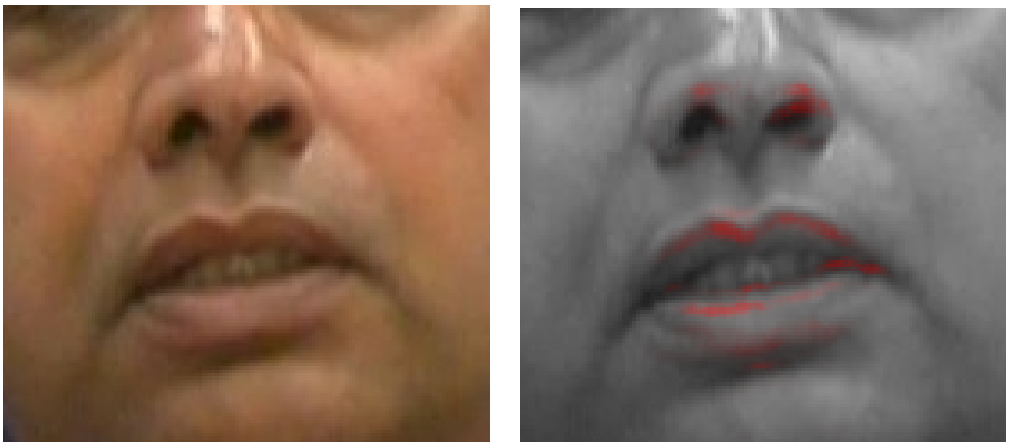


Figure 3.19: Comparing the raw frame to the failed lip classifier result, for a subject of Indian appearance (VidTIMIT subject 37)

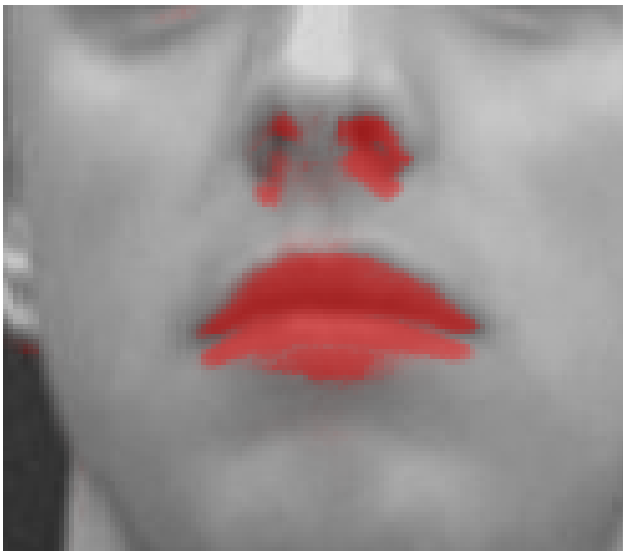


Figure 3.20: Comparing the classified lip region to the original image for VidTIMIT subject 38

able to identify the upper lip, as it is completely obscured by the subject's moustache. As the classifier is relying on the colour of the pixels, there is no way for it to successfully locate the lips in this situation. If the lip is not entirely obscured by the facial hair, it is still able to be successfully identified, as illustrated in Figure 3.23.

Figure 3.21: Comparing the classified lip region to the original image for VidTIMIT subject 09

Figure 3.22: Lip classification result when the lip is obscured by facial hair (VidTIMIT subject 26)

The neural network classifier discussed in this chapter successfully identified the lips for most subjects in the datasets. It is able to distinguish the lips from surrounding skin and inner mouth for most subjects. While it did struggle with some subjects, this is likely due to the lack of sufficient training data, with the majority of the dataset being subjects with light coloured skin.

Figure 3.23: Lip classification result when the lip is still visible near facial hair (VidTIMIT subject 35)

With a larger, more encompassing dataset, the classifier is likely able to be trained to function for subjects with these characteristics. Another potential method for handling the range of skin colours is to build several classifiers, each trained for a different range of skin colours. A suitable classifier could then be chosen dynamically, based on the general skin colour in the facial region.

While improvement would be necessary to build a general purpose visual speech recogniser, the focus of this thesis is to determine the suitability of visemes as the basic visual unit of speech. As the classifier performs sufficiently for this purpose, these improvements are not necessary.

The output of the lip pixel classifier can now by fed into the lip feature extractor, enabling the lip shape to be found for use by the speech recognition process. The lip feature extractor will need to take into account shadows near the lips causing false positives to be output by the lip pixel classifier. This is of particular importance beneath the nose, as they are particularly strong and in close proximity to the lips.

Chapter 4

Lip Feature Extraction

The underlying principle of visual speech recognition is that the mouth shape is based on the underlying structure of the word being spoken (Holden and Owens, 2000). In the previous chapter, a lip pixel classifier was described that identifies pixels that are similar in appearance to lips. Before this can be used to recognise speech, a feature needs to be extracted to represent the mouth shape.

This chapter describes a lip feature extractor that paramaterises the lip shape using an improved algorithm known as "wrapping snakes" (Ramage and Lindsay, 2009). This algorithm is based on the traditional snakes algorithm (Kass, Witkin, and Terzopoulos, 1988), but is modified to better suit the challenges of lip shape extraction. This modification increases the robustness of the algorithm, and results in a more accurate lip shape being produced.

This chapter first discusses the various feature types used for visual speech recognition. This is followed by introducing the traditional snake algorithm, and then the adapted technique, known as wrapping snakes, for finding the shape of the lips more accurately. A further improvement, through the use of pinching and cutting processes, is then discussed. Lastly is how wrapping

snakes, with pinching and cutting, are used to produced the feature vector used by the speech recogniser.

4.1 Choice Of Feature Vector

Speech recognisers require a feature vector that represents the current observation. There are many different feature types that can be used, but one has to be chosen before implementing the system. For audio data, a common feature used is mel-frequency cepstral coefficients. They are similar to the FFT frequency bins, but use a non-linear frequency scale to allow more detailed information about the lower frequencies. Other audio feature vectors used are often based on linear predictive coding (LPC) coefficients. These coefficients can be transformed in many ways to provide different feature sets. (Young, 2008)

For the visual data, there is less agreement on which feature set to use (Chen, 2001; Potamianos et al., 2004). This is due to the fact that the image and video processing fields are a lot younger than the audio processing field, and the computational requirements are much higher for video processing.

Visual feature sets can generally be grouped into three categories: video pixel (appearance) based features, shape based features, and combinations containing both appearance and shape based features (Potamianos et al., 2004). The appearance based features typically consider a region of interest (ROI) around the mouth, that include just the lip region, or may extend further to include more, or even all, of the face. Appearance features typically encode the ROI using common image transforms, often used in image compression. These include principle component analysis (PCA), discrete cosine and wavelet transforms, linear discriminant analysis (LDA), and Haar-like features (Hazen et al., 2004; Revéret, 1997; Shen and Bai, 2006; Wilson and Fernandez, 2006).

Shape based features represent the ROI using a parametric model of the lips or face. There are two types of features within this category: geometric features, and shape model based features (Potamianos et al., 2004). Geometric features use simple geometric measurements such as the width of the lips, and the height of each lip (Chen, 2001).

Shape model based features use a parametric model to represent the lip or a larger region of the face. "Snakes" are a type of active contour model (ACM) that use a mathematical model to represent detailed shapes, and have been used to represent the lip shape as a series of points on a spline (Kass, Witkin, and Terzopoulos, 1988). Active shape models (ASM) are statistical models that contain a labelled set of points to represent the object (Cootes and Taylor, 1992; McKenna et al., 1997). The models are derived from a number of sample images, and limits are placed on the parameters to restrict shapes to conform to constraints imposed by the training set.

Lastly, combination features can be used that include both appearance and shape based features. These can be simple concatenations of appearance and shape features, or they can be more complicated hybrids such as active appearance models (AAM). AAMs are an extension of ASMs but also contain the appearance information within the model itself (Potamianos et al., 2004).

For the lip feature extractor described in this chapter, a modified version of the snake algorithm has been chosen as the feature type. As the shape of the mouth has a direct relationship with the sounds being formed as speech, the decision was made to use a shape based feature. The choice of snakes over active shape models is due to snakes having a more precise representation of the lip shape when compared to ASM, which use only a small number of points to define the geometric model of the lips.

4.2 Traditional Snakes

Snakes are a series of connected points that are controlled by a mathematical model. They are a type of active contour model, which use an energy minimising spline that is guided by internal and external forces (Kass, Witkin, and Terzopoulos, 1988).

The internal forces of the snake represent the tension and rigidity of the spline. The tension force encourages the snake to contract, allowing it to enclose around features. By varying the value of the tension coefficient, the snake can be targeted towards smaller or larger features. The rigidity force is used to control how sharply the snake can bend, which allows the snake to target smoother or sharper curved features.

The external forces of the snake combine the image forces and the constraint forces. The image forces are derived from the image itself, and are chosen to track the desired features within the image. By choosing a suitable image force, the snake is encouraged towards the desired features. Constraint forces are due to any external requirements for the shape of the snake. This can allow any prior information of the target features to be used to guide the shape of the snake.

The snake energy is the combination of the energy due to internal, external, and constraint forces. The shape of the snake is determined by minimising the energy of the snake (Kass, Witkin, and Terzopoulos, 1988).

If the snake position is represented by $\mathbf{v}(s) = (x(s), y(s))$, the snake energy, E^*_{snake}, can be written as the integral of the internal, image, and constraint energy along the snake

$$\begin{aligned} E^*_{snake} &= \int_0^1 E_{snake}(\mathbf{v}(s))ds \\ &= \int_0^1 E_{int}(\mathbf{v}(s)) + E_{image}(\mathbf{v}(s)) + E_{con}(\mathbf{v}(s))ds \end{aligned} \tag{4.1}$$

where E_{int} represents the internal energy of the snake, E_{image} is derived from the image forces, and E_{con} gives rise to external constraints.

The internal energy of the snake is given by

$$E_{int} = \frac{(\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2)}{2} \tag{4.2}$$

where α and β are the tension and rigidity coefficients, and $\mathbf{v}_s$ and $\mathbf{v}_{ss}$ are the first and second derivatives of the snake position respectively.

The snake position is determined by performing energy minimisation, using an iterative process as described in Kass, Witkin, and Terzopoulos, 1988. The position of the snake at the next iteration is calculated as

$$\begin{aligned} \mathbf{x}_t &= (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{x}_{t-1} - \mathbf{f}_\mathbf{x}(x_{t-1}, y_{t-1})) \\ \mathbf{y}_t &= (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{y}_{t-1} - \mathbf{f}_\mathbf{y}(x_{t-1}, y_{t-1})) \end{aligned} \tag{4.3}$$

where $\mathbf{A}$ is a pentadiagonal banded coefficient matrix as described in (Kass, Witkin, and Terzopoulos, 1988), $\mathbf{f}_\mathbf{x}$ and $\mathbf{f}_\mathbf{y}$ are the image forces in the x and y directions respectively, and γ is the Euler step size.

4.3 Calculating The Image Force

The image forces are what allow the snake to be influenced in some way by the image itself. These are calculated as a force vector field based on the pixel values within the image, such that the snake is attracted to the desired features within the image. The simplest image forces are calculated by applying the gradient function to the pixels values, where the force is defined by the magnitude and direction of this gradient, which will attract the snake to local maxima. The gradient-derived image force is always perpendicular to the edge of the feature.

If a binary image, or a sharp greyscale image, is used to calculate the image forces, the gradient function will result in forces that have very little reach away from the features. This type of image force is only useful if the snake is initially located close to the desired final location, and does not have any concave areas that need to be expanded into, or the image has smooth transitions between high and low valued pixels. This is illustrated in Figure 4.1, where it can clearly be seen that the image forces have very little reach beyond the edge of the lip feature.

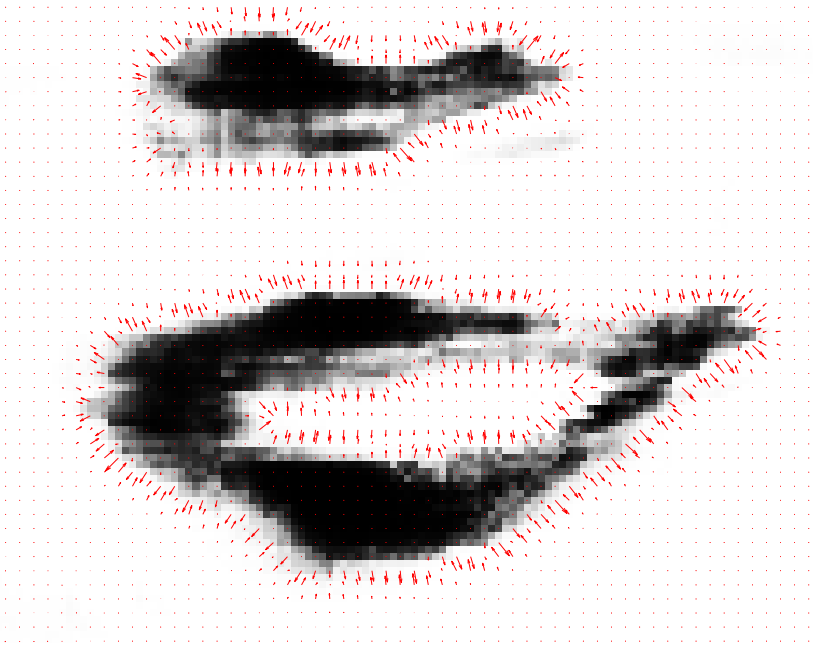


Figure 4.1: Image forces generated by the gradient function

If there are concave areas in the target feature, the gradient function will not generate image forces that push the snake into the concave feature, since the direction of the force is always perpendicular to the edge of the feature. In Figure 4.1, if a snake was initialised so the left side of the mouth was outside the initial snake position, there are no forces that would push the snake outwards towards the corner of the mouth.

One way to attempt to improve the behaviour of snakes is to choose a more sophisticated image force. This is usually done by applying a filter to the

image itself before taking the gradient to determine the image forces, giving the image forces a longer reach.

4.3.1 Gaussian Image Forces

To enable the image forces to act over greater distances, the image can be blurred by applying a Gaussian filter before taking the gradient. This will make the gradient change more gradually, allowing the snake to be attracted to features from further away. Another benefit of blurring the image is it will help allow the snake to be attracted into concave areas. A downside of blurring the image is that sharp details are lost, preventing the snake from fitting very tight corners. For the purposes of lip segmentation, the loss of sharp details is not a concern, as the lips are naturally curved. An example of the image forces generated by applying a Gaussian filter can be seen in Figure 4.2.

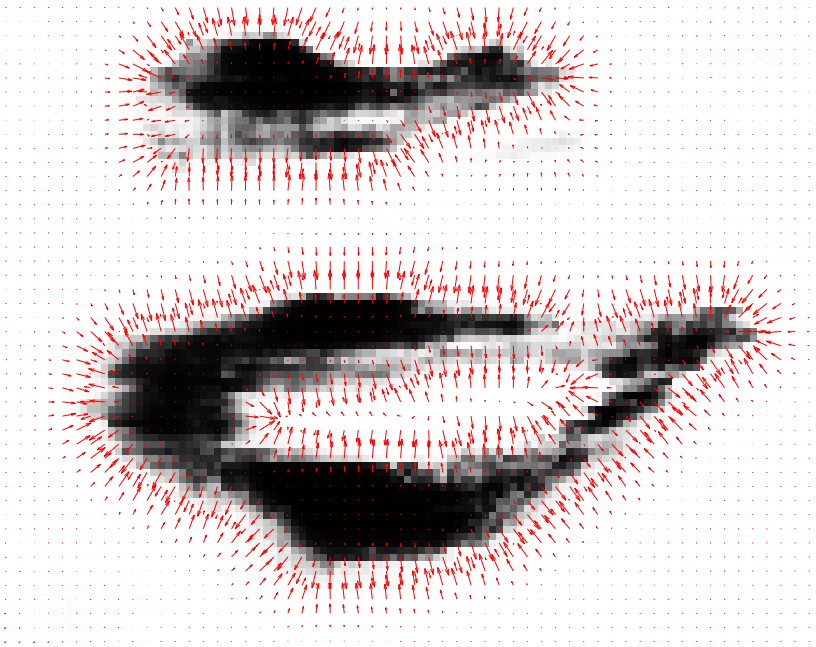


Figure 4.2: Image forces generated by applying a Gaussian filter before taking the gradient

Applying a Gaussian kernel to each frame of video is not an overly expensive operation, and is a common function in highly optimised image processing libraries. As such, the increased reach of the image forces significantly outweighs the computational cost of performing this operation.

4.3.2 Gradient Vector Flow

One method that was designed to minimise these problems is the Gradient Vector Flow (GVF) field approach for calculating the image forces (Chenyang and Prince, 1998).

The GVF field is calculated as the diffusion of the gradient vectors of an edge map. GVF image forces differ from traditional image forces in that they are not calculated as a gradient of a potential function. By removing the requirement of taking the gradient of a function, the GVF field does not have to be solely an irrotational field, it can be comprised of an irrotational component and a solenoidal component.

The main characteristic of GVF image forces is the ability to push the snake into concave areas (see Figure 4.3). Traditional gradient forces cannot do this, as they will always push towards the closest image feature. While this is a beneficial characteristic, the computational overhead of calculating the GVF field is significantly higher than other techniques. In terms of computation time, the GVF forces take approximately seven times longer to calculate than the traditional image forces (Chenyang and Prince, 1998). Even when taking into account increased processing power of modern computers, it is clear the GVF field is significantly slower than other techniques.

The GVF field is defined as the vector $\mathbf{f}_{gvf}(x, y) = [u(x, y), v(x, y)]$ that minimises the energy function

$$\epsilon = \iint \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla i|^2 |\mathbf{f}_{gvf} - \nabla i|^2 dxdy \tag{4.4}$$

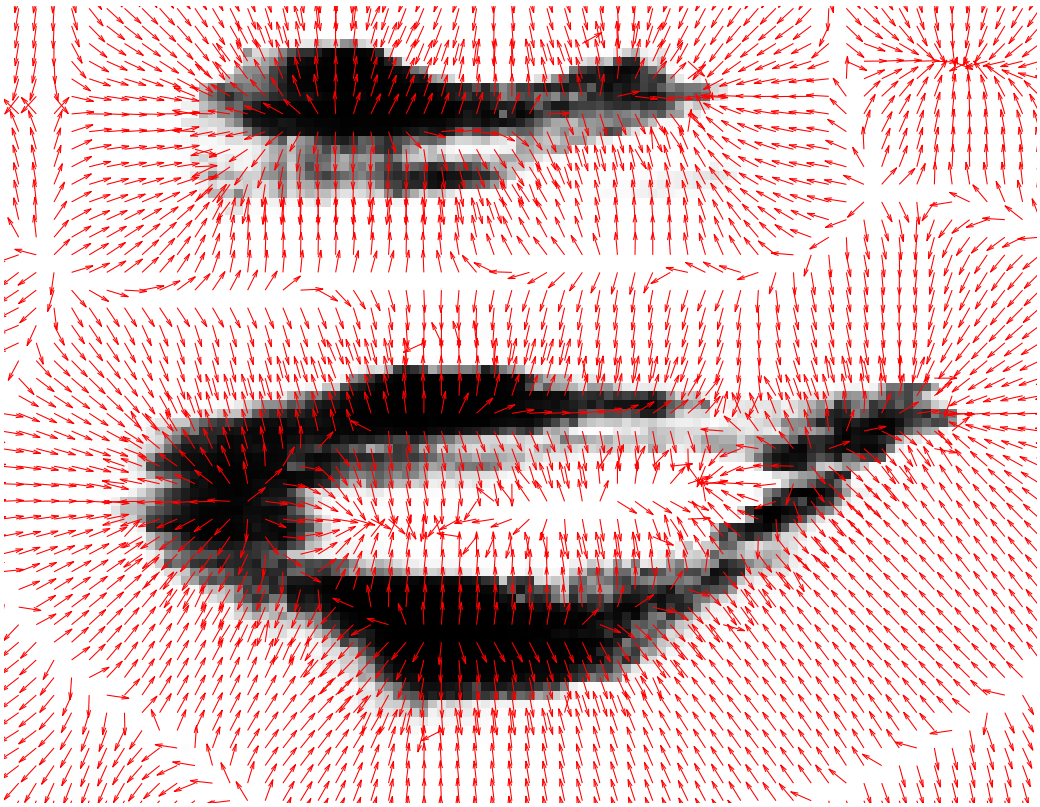


Figure 4.3: Image forces generated using Gradient Vector Flow

where i is the edge map for the given image.

In this form, it can be seen that when $|\nabla i|$ is small, the energy is dominated by the partial derivatives of the vector field, resulting in a slowly varying field. Alternately, when $|\nabla i|$ is large, the second term dominates the integrand, which is minimised by setting $\mathbf{f}_{gvf} = \nabla i$. The parameter μ controls the weighting between the first and second terms, which allows the effect of noise to be reduced when needed (Chenyang and Prince, 1998).

Even though the GVF forces have a longer reach than Gaussian forces, they are a lot more expensive to calculate in terms of CPU time. As the lips do not move very much between successive frames of video, the larger reach of the GVF forces does not outweigh the additional computational cost of calculating the GVF image forces for each frame.

4.4 Traditional Snake Behaviour

Figure 4.4 illustrates a snake successfully finding an outer lip boundary. The snake is initialised (blue line), several iterations are performed (green lines), and finally the snake converges to the lips (red line). This is an ideal situation, as there is no noise (false positives) in the image, the snake was initialised relatively close to the lip boundary and was touching the lip boundary in some places. As a result, the majority of the snake found the lip boundary on the first iteration, with the full lip boundary being found within four iterations.

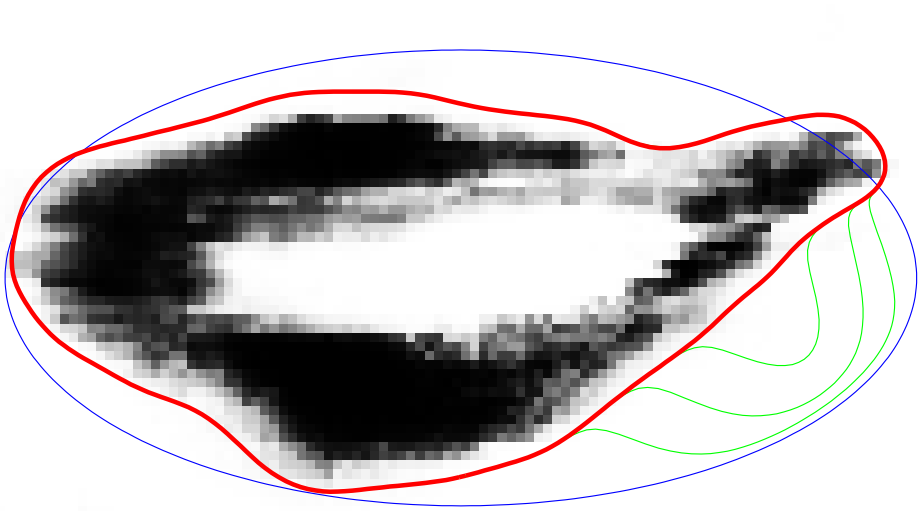


Figure 4.4: Traditional snake finding the lip boundary under ideal conditions

While this shows that snakes can be used to find the lip boundary under good conditions, it is more important to analyse the snake's performance for less than ideal conditions. This includes poor initialisation and noisy images.

In situations where the snake is initialised in a position close to noise in the image, the traditional snake cannot always overcome the image forces due to the noise to successfully find the desired feature. This is especially true if the noise is strong and the desired feature is relatively weak.

Figure 4.5 illustrates a traditional snake failing to correctly find the lip boundary due to strong noise being present just above the lip. As the initial position of the snake is in close proximity to this noise, a strong force is needed to try to counter the strong image forces pulling the snake up towards this noise feature. Of the three forces present, the internal tension is the only force that can pull the snake down towards the lips. When the tension is increased, it also acts on the snake near the corner of the mouth. The narrow shape of the lips results in only a small section at the very corner of the mouth generating image forces that pull the snake outwards towards this corner. This small outwards pulling image force is not strong enough to counter the larger tension force pulling the snake inwards, resulting in the snake falling off the corner of the mouth and failing to correctly locate the lip boundary.

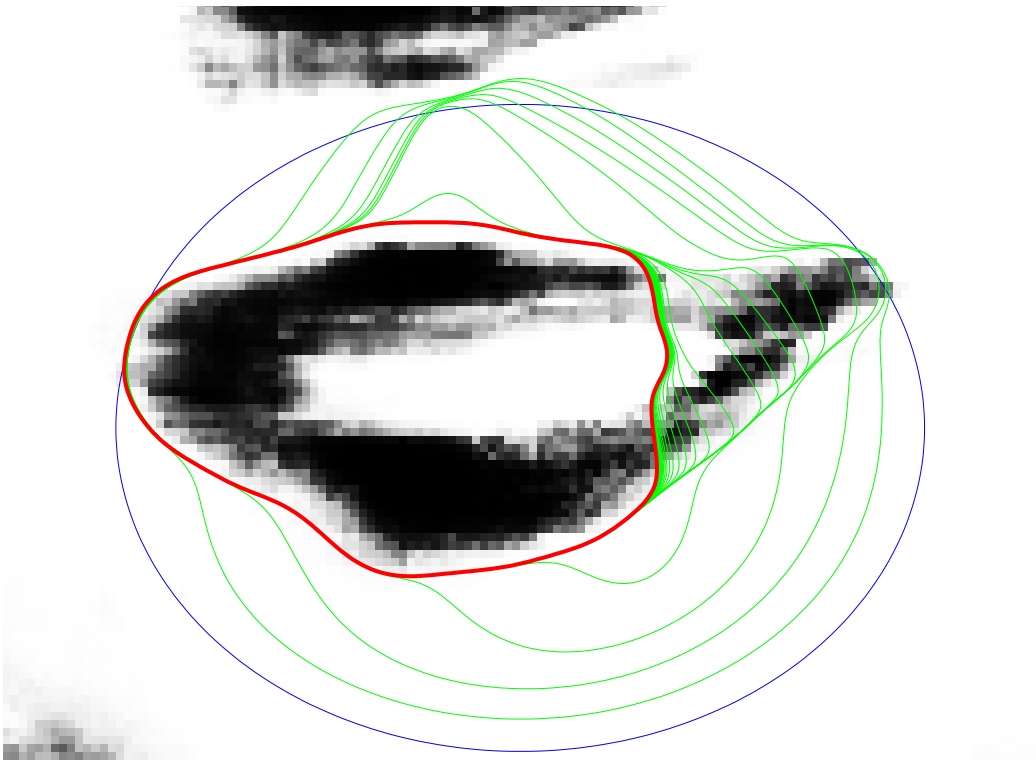


Figure 4.5: Traditional Snake with strong noise and a weak target feature

Not only is there nothing encouraging the snake to continue along features it has found, but there is also nothing discouraging the snake from retreating along these same features. As the direction of the image forces is always perpendicular to the feature, the only force parallel to the feature is the tension force. In sections of a traditional snake that are curving away from the feature, the tension of the snake will cause it to be pulled back along the feature, as illustrated in Figure 4.5.

Another situation that traditional snakes cannot handle is an initial position that encloses multiple features, such as noise features near the target feature. As can be seen in Figure 4.6, the snake fails to successfully locate either feature and ends up enclosing a single region containing the multiple features.

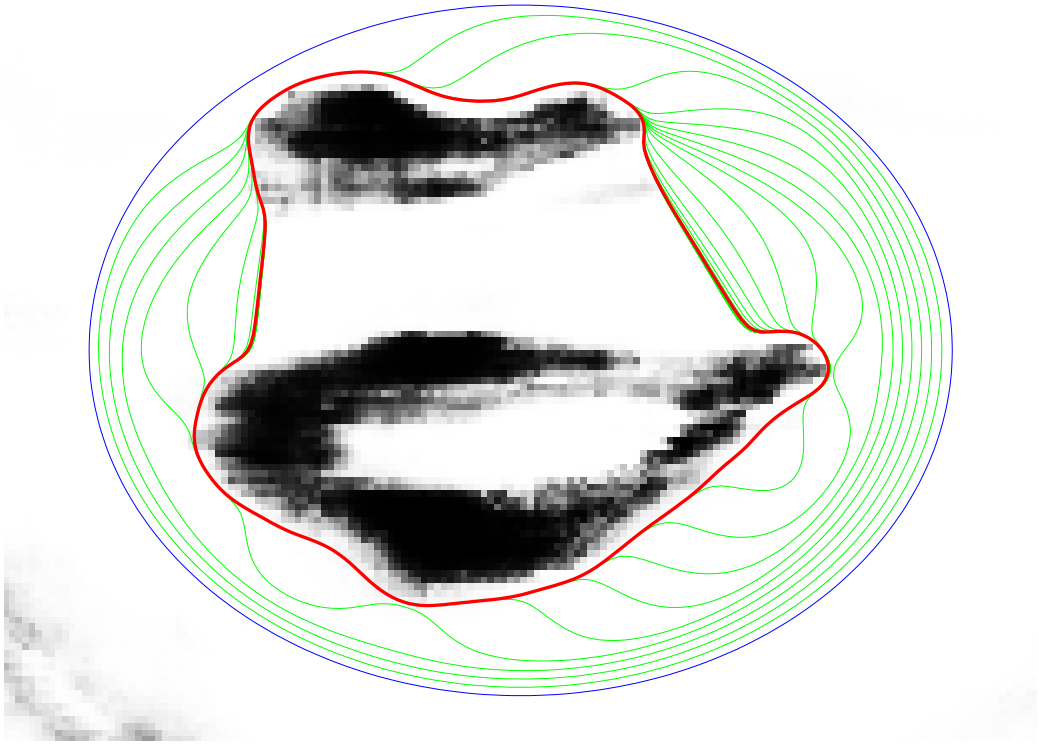


Figure 4.6: Traditional snakes fail to find either feature when enclosing multiple features

As illustrated in Figure 4.7, the forces in the region between the two enclosed features do not pull the snake into this gap. The tension force (blue) tries

to pull the snake tight between the outer extremities of the features, like an elastic band. The image forces (green) are perpendicular to the image features, resulting in vertical forces on the snake. The rigidity force discourages sharp bends in the snake, resulting in a smooth curve away from the image features, as the tension pulls the snake upwards. The balance of these forces results in a snake that cannot be pulled into the region between the enclosed features.

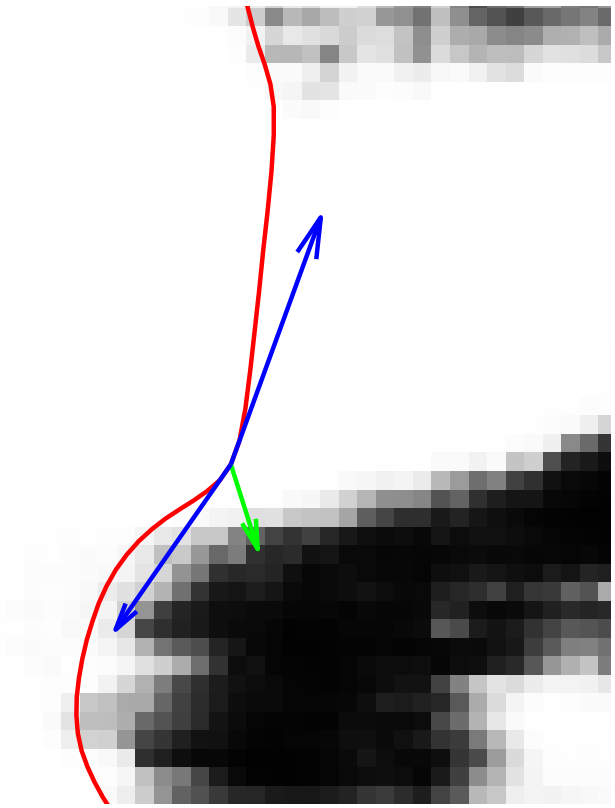


Figure 4.7: Balance of forces when traditional snakes enclose multiple regions

4.5 Wrapping Snakes: An Improved Technique

One of the main problems with traditional snakes is that they do not have any way of being encouraged to continue along features they have partially found.

While using a more sophisticated image force can partially assist with this, it does not help when the snake has partially found two conflicting features. A good example of this is Figure 4.6 (see page 96), where the snake has partially found the shadow under the nose, as well as the lip boundary. With traditional snakes, the image forces only act perpendicular to the features, and the tension pulls the snake back along partially found features. What is needed is a force that pushes the snake along a feature it has already partially found.

To overcome this problem with traditional snakes, a wrapping force is introduced as a substitute for the image force in Equation 4.3 (see page 89). The wrapping force is based on the image force, but is modified by the snakes' shape and location at each iteration. The wrapping force is simply the component of the image force that is in the direction of the normal of the snake at that point. This is illustrated in Figure 4.8.

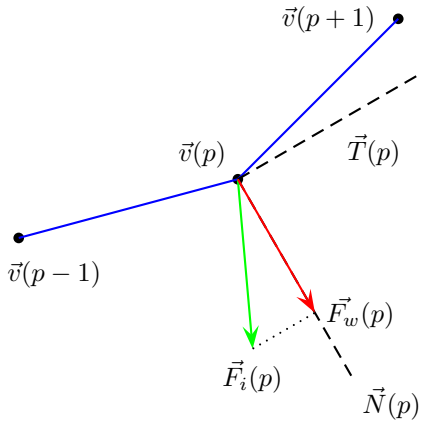
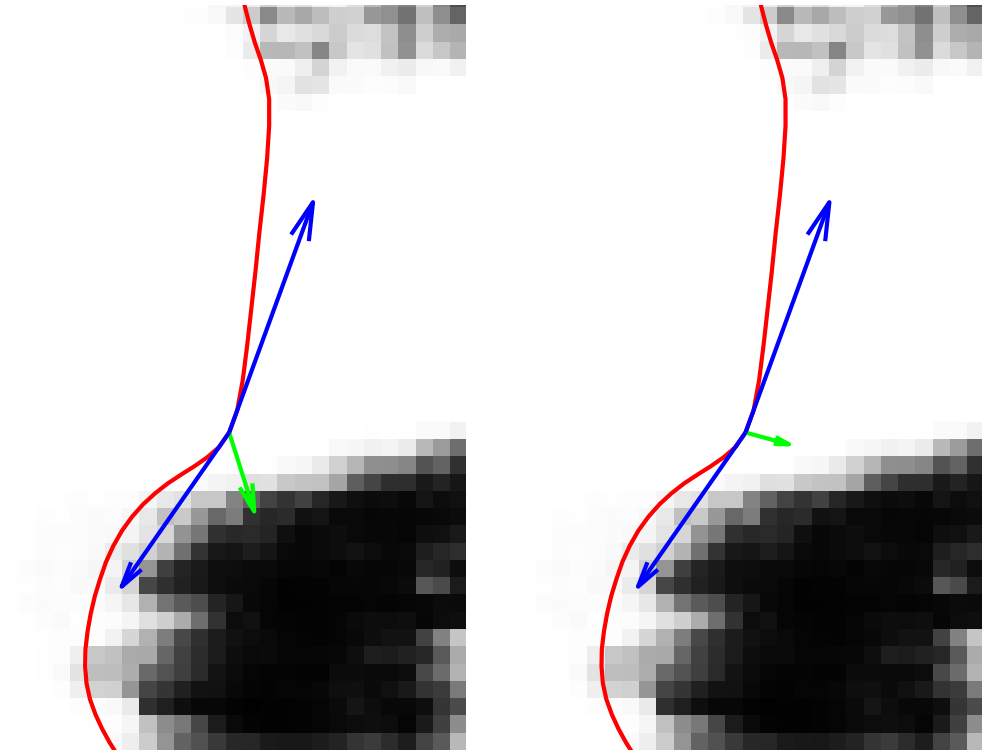


Figure 4.8: Determining the wrapping force (red), given the image force (green) and snake position (blue)

Since this new wrapping force is always perpendicular to the snake, rather than the image feature, it will encourage the snake to be pulled along features the snake is curving away from. This encourages the snake to continue along features it has already found, overcoming one of the shortcomings of traditional snakes. This is illustrated in Figure 4.9, where it can be clearly seen that the wrapping force pushes the snake along the partially

found feature. When the snake is located parallel to the image feature, the wrapping force is identical to the original image force, resulting in these sections of the snake being pulled directly towards the feature.



(a) Traditional image forces (b) Wrapping forces

Figure 4.9: Comparison between (a) traditional image forces and (b) wrapping forces

As can be seen in Equation 4.5, the wrapping force, $\mathbf{F_w}$, can be calculated as the dot product of the image force, $\mathbf{F_i}$, and the unit normal, $\hat{\mathbf{N}}$, multiplied by the negative unit normal.

$$\mathbf{F_w} = -(\hat{\mathbf{N}} \bullet \mathbf{F_i})\hat{\mathbf{N}} \tag{4.5}$$

The normal, $\mathbf{N}(p)$, is perpendicular to the tangent, $\mathbf{T}(p)$. The tangent can be calculated as

$$\mathbf{T}(p) = \frac{\mathbf{v}(p) - \mathbf{v}(p-1)}{|\mathbf{v}(p) - \mathbf{v}(p-1)|} + \frac{\mathbf{v}(p+1) - \mathbf{v}(p)}{|\mathbf{v}(p+1) - \mathbf{v}(p)|} \tag{4.6}$$

Since $\mathbf{N}(p)$ is perpendicular to $\mathbf{T}(p)$, it can be calculated by performing a 90°rotation. Therefore

$$\mathbf{N}(p) = \left(-\mathbf{T}_y\left(p\right), \mathbf{T}_x\left(p\right)\right) \tag{4.7}$$

where $\mathbf{T}_x(p)$ and $\mathbf{T}_y(p)$ are the x and y components of $\mathbf{T}(p)$ respectively. The direction of the rotation is not important, as the dot product in Equation 4.5 will correct for it, always resulting in the correct projection.

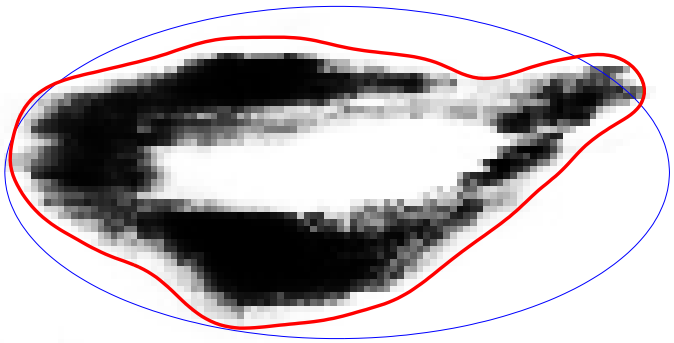
The original pair of equations for calculating the position of the snake at the next iteration (see Equation 4.3 on page 89) are also used for wrapping snakes. In the original equations, $\mathbf{f_x}$ and $\mathbf{f_y}$ represent the x and y components of the image forces, whereas for wrapping snakes they represent the x and y components of the wrapping forces. Equation 4.8 shows the new calculation for the wrapping snake position, which replaces Equation 4.3, and now includes a wrapping force coefficient, ω.

$$\begin{aligned} \mathbf{x}_t &= (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{x}_{t-1} - \omega\mathbf{f_x}(x_{t-1}, y_{t-1})) \\ \mathbf{y}_t &= (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{y}_{t-1} - \omega\mathbf{f_y}(x_{t-1}, y_{t-1})) \end{aligned} \tag{4.8}$$

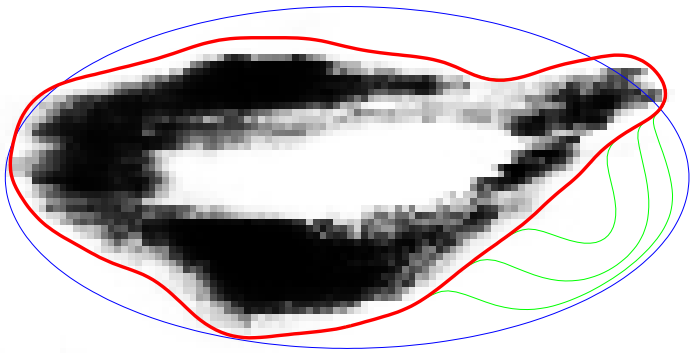
4.6 Wrapping Snake Behaviour

When the snake is perpendicular to the image forces, the wrapping forces will be equal to the image forces, as the image force is already in the same direction as the normal. This means that the snake will behave in a similar way to traditional snakes when no wrapping is required.

One advantage of using wrapping snakes, even when traditional snakes would also work, is a reduction in the number of iterations required to successfully locate the lip boundary. This is illustrated in Figure 4.10(a), where only a single iteration is required for the wrapping snake, compared to the four iterations required for the traditional snake (Figure 4.10(b)).



(a) Wrapping snake finding the lip boundary under ideal conditions



(b) Traditional snake finding the lip boundary under ideal conditions (reprint of Fig. 4.4)

Figure 4.10: Snake behaviour under ideal conditions

This reduction in the number of iterations required is due to the direction of the forces involved in the two algorithms (see Figure 4.11). Traditional snakes have to rely on the tension (blue) and rigidity (magenta) pulling the snake to within the reach of the image forces (green), which then pulls a small section perpendicular to the lip boundary. Wrapping snakes, on the

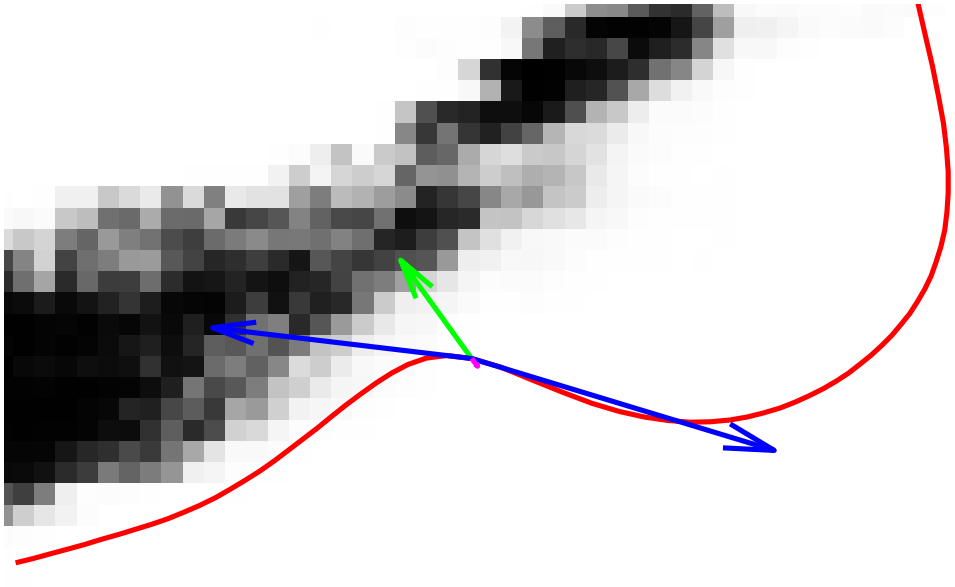
other hand, have the wrapping force (green) pushing the snake along features partially found. As a result, it quickly continues along the short section of the lip boundary that is slightly further from the initial position.

As the wrapping force encourages the snake to continue along a feature it has already found, it allows the snake to correctly find a weak target feature even when there is strong noise nearby (Figure 4.12(a)). When this is compared to the traditional snake (Figure 4.12(b)), it is clear that the wrapping and internal forces allow the snake to disregard the shadow under the nose, without falling off the side of the mouth.

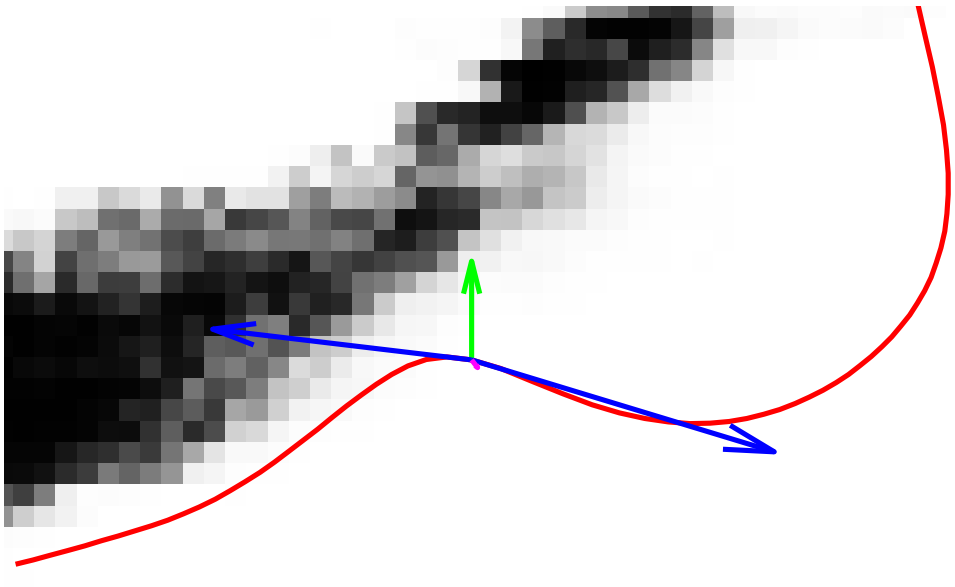
Even with very poor initialisation, wrapping snakes can still successfully locate the lip boundary. Figure 4.13 shows a snake with an initial position that is along the shadow under the jaw line and also passes through the shadow under the nose. In this situation, the wrapping snake is still able to successfully locate the lip boundary given enough iterations, whereas the traditional snake will never succeed.

As can be seen in Figure 4.14(a), by substituting the wrapping force for the image force, the snake can successfully locate the outer lip boundary even when the initial position enclosed two features. When compared with the traditional snake in Figure 4.14(b), it is seen that the wrapping forces allow the snake to continue along the features as intended.

In Figure 4.14(a), the snake has formed three closed regions. One is the lip boundary, one is the boundary of the noise feature, and the third region is an artefact of the snake algorithm itself. This artefact is caused when the wrapping forces push the two sections of the snake through the gap between the two features, but there is nothing stopping the two sections once they meet.

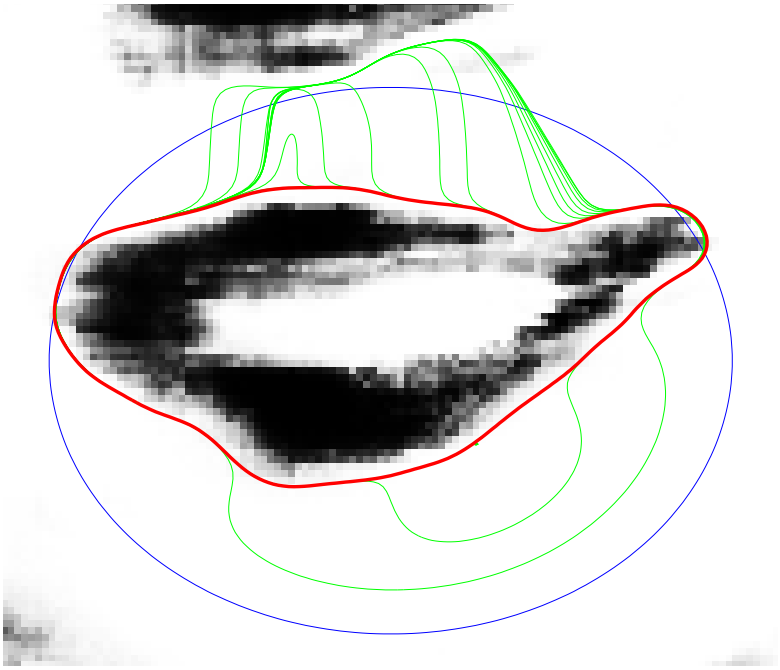


(a) Traditional image forces

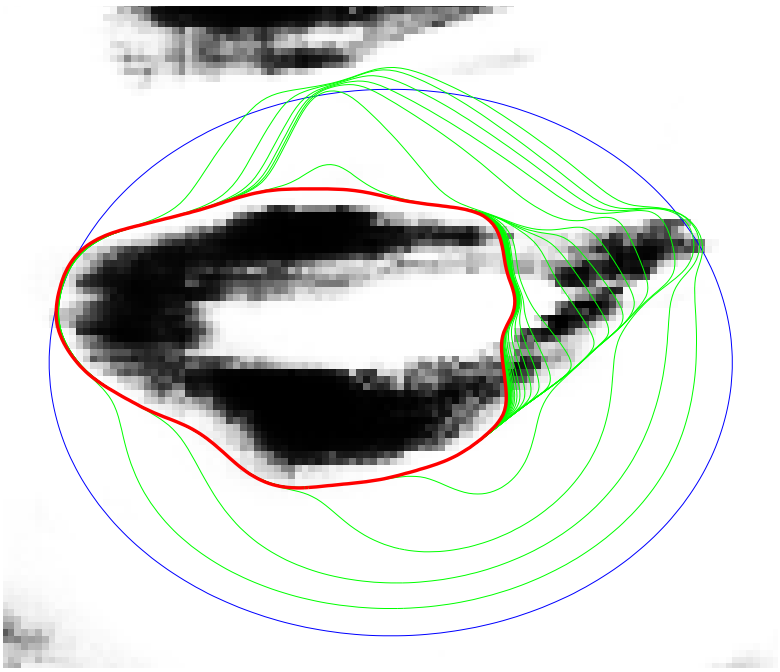


(b) Wrapping forces

Figure 4.11: Comparison between (a) traditional image forces and (b) wrapping forces



(a) Wrapping snake with weak target and strong noise features



(b) Traditional snake with strong noise and a weak target feature (reprint of Fig. 4.5)

Figure 4.12: Handling strong noise and weak target features

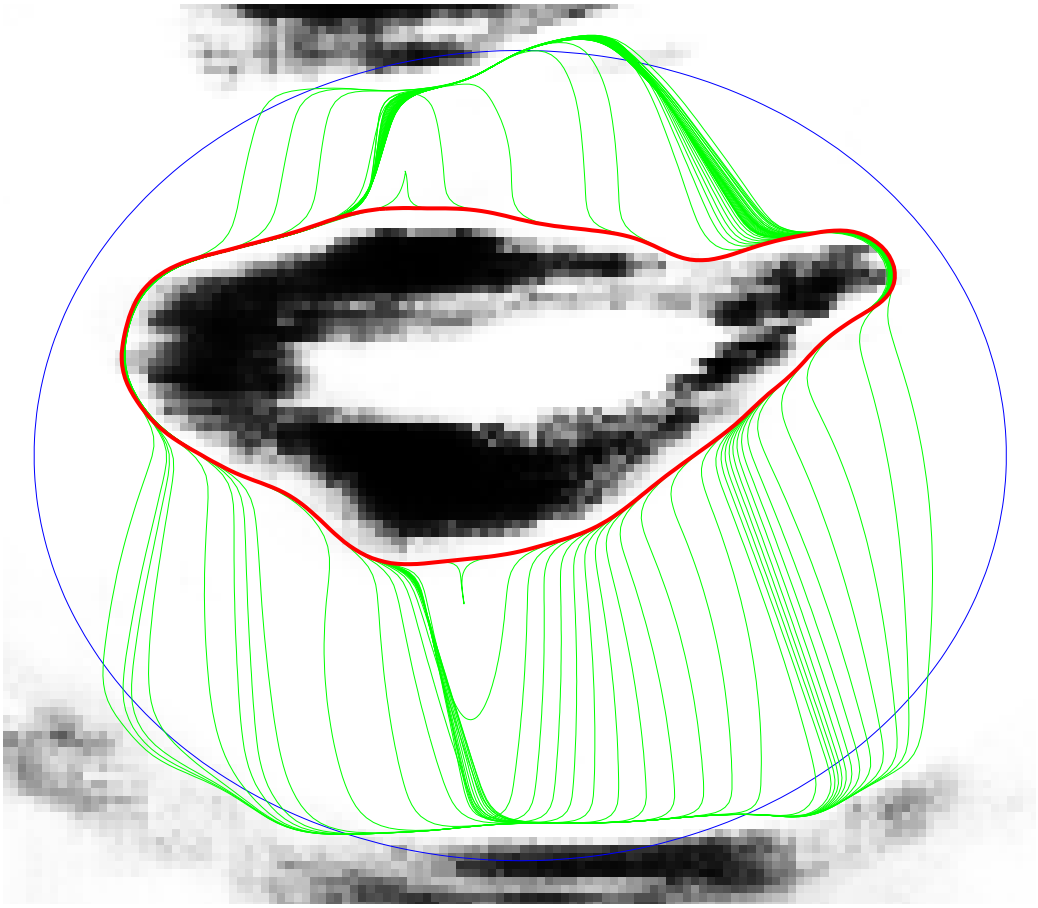
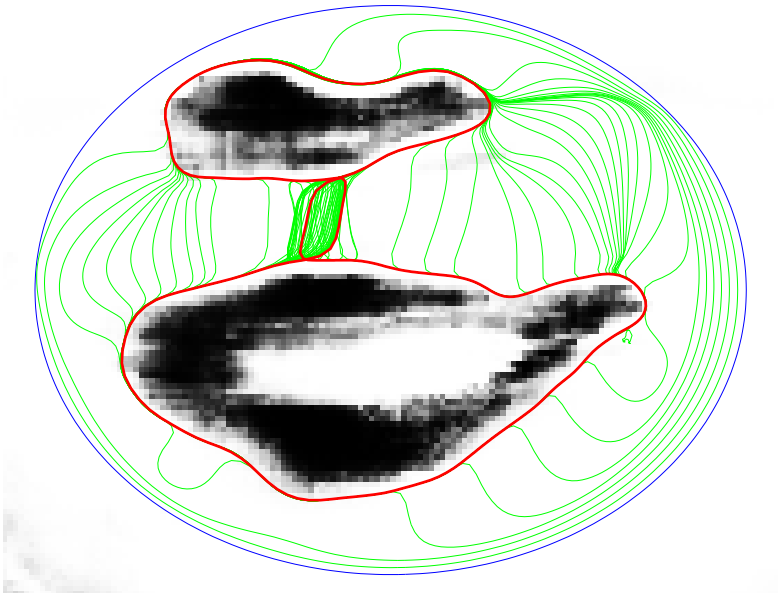


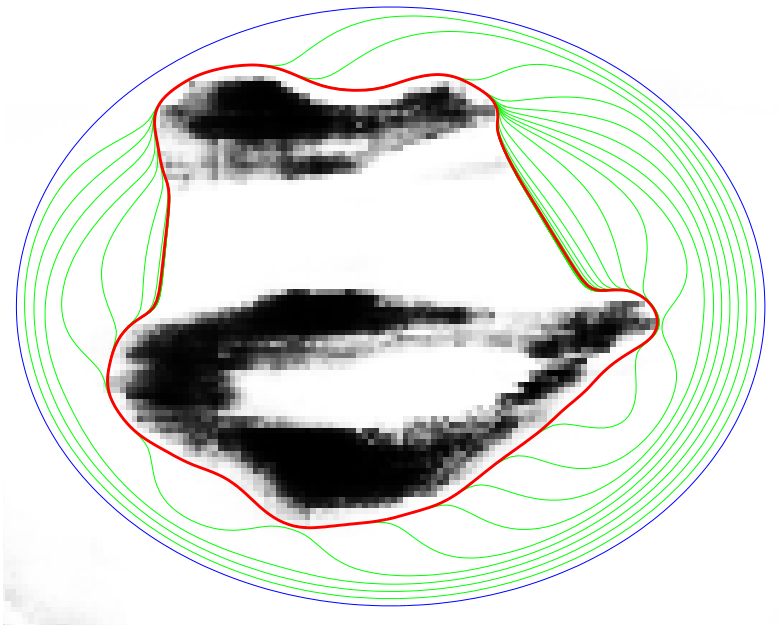
Figure 4.13: Wrapping snake successfully finding the lips, even with very poor initialisation

4.7 Pinching Force

To help the snake fully enclose multiple features, a pinching force is added to the wrapping snake algorithm. This force pulls sections of the snake towards each other, similar to the way the tension force works, but is only applied to non-adjacent sections of the snake that close to within a certain distance. This is to ensure that this additional force will not act as just another tension force, but will instead act between separate sections of the snake, such as the two sections of the snake in Figure 4.14(a) that wrap between the lip and nose features.



(a) Behaviour of wrapping snakes when initial position encloses multiple features



(b) Traditional snakes fail to find either feature when enclosing multiple features (reprint of Fig. 4.6)

Figure 4.14: Handling multiple enclosed features

As this force is to help pull non-adjacent sections of the snake towards each other, it does not need to be applied to every point within the snake. To improve processing time, a subset of the snake's points are designated "pinch points". The pinching forces are only calculated and applied between these pinch points, with the internal forces of the snake enabling the surrounding points to be affected indirectly. For a pinch force to be applied, two pinch points must be located within a given spatial distance, and be non-adjacent pinch points. If a pinch point is close enough to multiple other pinch points, each will apply a force onto this pinch point, with the resulting force being the sum of these forces.

The magnitude of the pinching force can be chosen based on the desired behaviour of the pinch. In this implementation, a simple threshold is used, such that if the pinch points are within the required distance, a fixed magnitude force is applied. Alternative approaches can use other derivations, such as inversely proportional to distance $\left(\frac{1}{r^n}\right)$, or even a logarithmic relationship.

The decision to use a simple threshold function was made as it exhibited the required behaviour, and has the lowest cost in terms of execution time. If an inverse relationship to distance was used, such as $\frac{1}{r^n}$, then the force applied as the distance drops below 1 could potentially result in significant overshoot of the two points, causing undesirable behaviour of the snake.

As illustrated in Figure 4.15, the pinching force (shown in blue) pulls the two sides of the snake towards each other as they wrap into the gap between the lips and the shadow beneath the nose. This additional pinching force ensures that the snake will completely enclose the lip feature, allowing the shape and location of the lips to be accurately determined.

As Figure 4.15 shows, there are also pinching forces in the right hand corner of the mouth. This is because the snake has closed within the required distance. Although these additional forces are trying to collapse the snake off this corner of the mouth, the rigidity and image forces are strong enough to prevent this from happening. As always, a balance is needed between each of the forces to

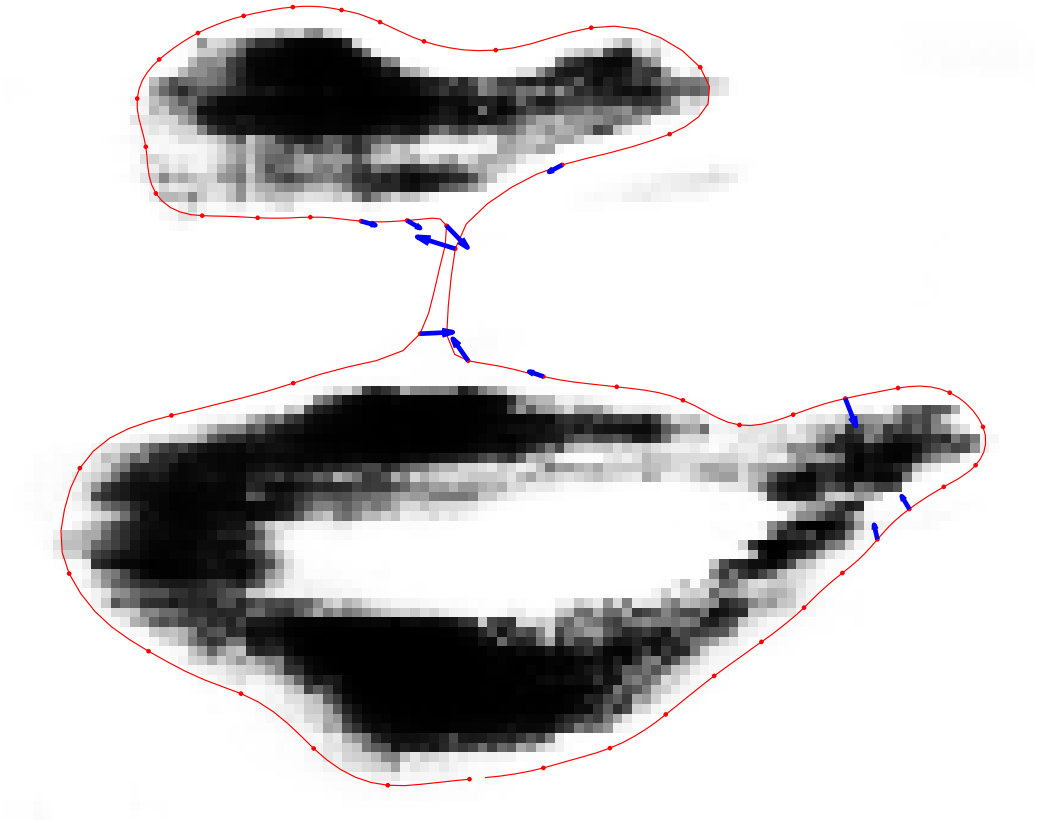


Figure 4.15: Pinching forces helping wrapping snakes "pinch off" two distinct features

ensure desirable behaviour of the snake. With this addition of the pinching force, the wrapping snake can successfully locate multiple enclosed features, without creating large "artefact regions". Now that the multiple features have been located, a process is needed to distinguish between them.

4.8 Cutting The Snake

As discussed previously, wrapping snakes do not distinguish between enclosed regions that are not needed (noise features), such as the shadow under the nose, and the actual target lip feature (see Figure 4.14(a) on page 106). As the snake wraps from both sides of the gap between features and pinches off,

another process is needed to cut the snake when the two different sections come into contact. By removing the section of snake that is surrounding the noise feature, it allows the snake to successfully find only the outer lip boundary. This would then allow the lip shape to be accurately represented, which is one of the fundamental requirements for visual speech recognition.

This cutting process is aided by the pinching force that has been added to help pull sections of the snake towards each other. Since these pinch points are strongly attracted to each other, the cutting algorithm looks for when two pinch points are located very close to each other and cutting at that point. By only cutting at pinch points, the computational requirements of the algorithm are minimised, improving the performance of the overall system. When two points are within the required distance, the snake is cut at this point.

As there are now two sections of the snake, a decision has to be made as to which has located the outer lip boundary. For simplicity, in this implementation, the shorter of the two sections of the snake is discarded. A more advanced method would be to spawn a second snake and allow each to continue independently. A decision could then be made as to which is more likely to have found the lips, using some measure of lip likeness.

Figure 4.16 illustrates the evolution of the wrapping snake, with pinching and cutting, that is initialised to enclose multiple features. When compared with traditional (Figure 4.14(b)) and wrapping-only snakes (Figure 4.14(a)), it is clear that by combining wrapping, cutting, and pinching, it is possible to successfully find the lip boundary, even when the snakes initial position encloses multiple features.

Initially, the snake locates sections of edge of the two features, and then begins to close around them. Next, the snake begins to wrap around the features, and into the gap between them. As the two sections of the snake approach each other, the pinching force pulls them towards each other, pinching off the two features (Figure 4.17(a)). As two pinch points lock together (Figure 4.17(b)),

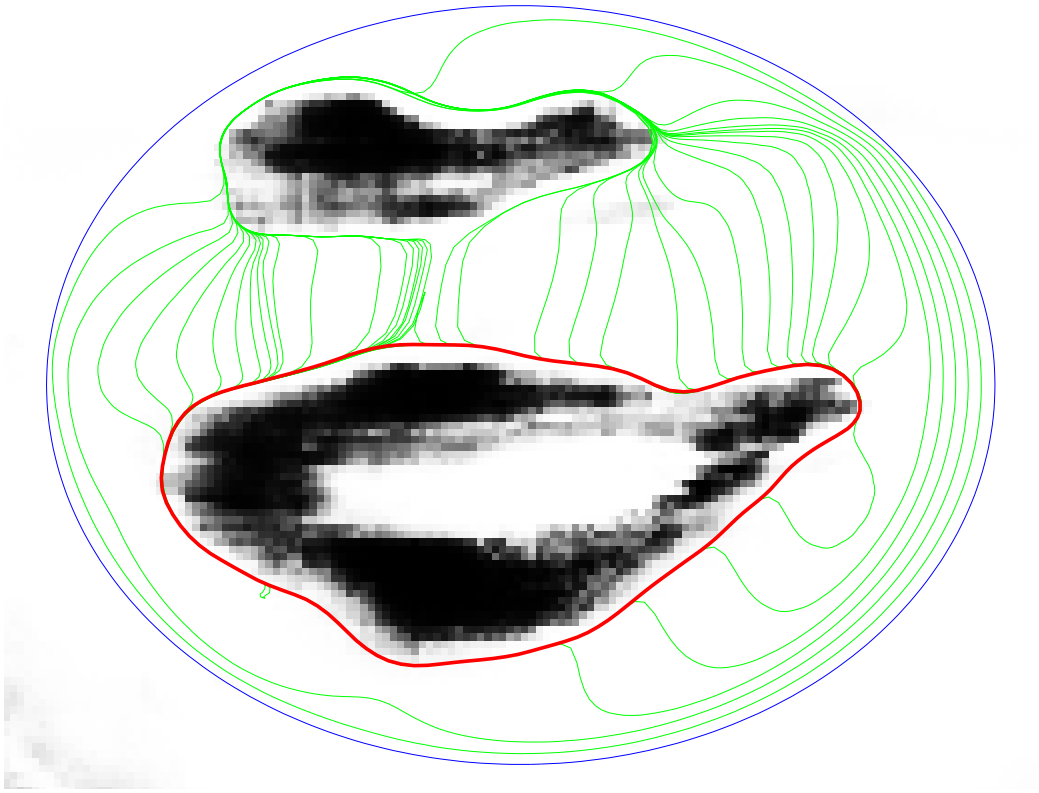
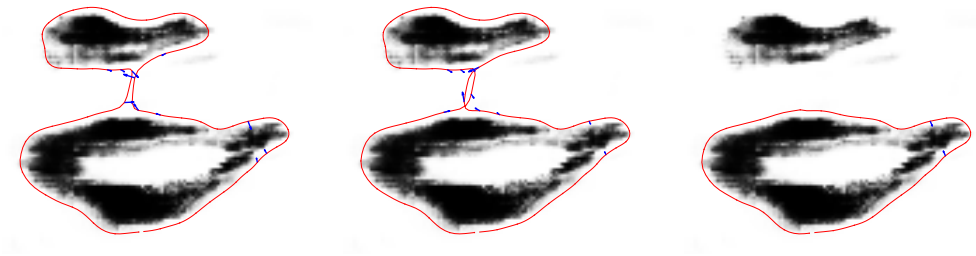


Figure 4.16: Finding the outer lip boundary using wrapping snakes with pinching and cutting

the snake is cut, spawning a separate snake for each feature. At this point the noise feature is discarded leaving just the lip boundary as the final located feature (Figure 4.17(c)).

When the snake is cut, the individual forces in this region of the snake significantly change, but this does not result in undesirable behaviour of the snake. If the snake had fully enclosed each of the features in the image before the cutting took place, the sum of resulting forces in the newly formed snake are quite similar to the sum of the original forces.

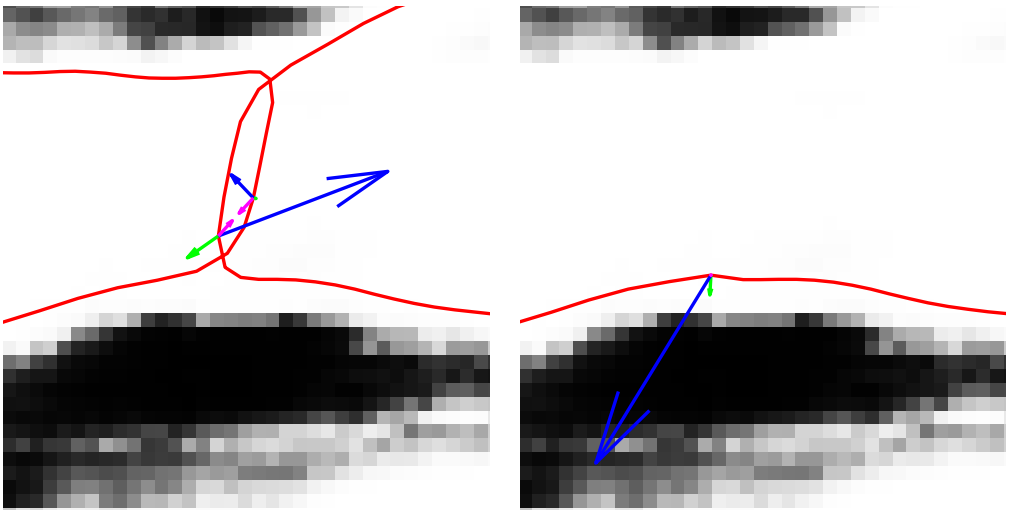
As can be seen in Figure 4.18(a), before the cut, the pinching forces (magenta) pulled the two sides of the snake laterally towards each other, the internal forces (blue) pulled the snake away from the feature, and the wrapping forces (green) pulled it back towards the feature. After the snake is cut



(a) Iteration 21 (b) Iteration 22 (c) Iteration 23

Figure 4.17: Discarding the noise by pinching off and cutting the snake

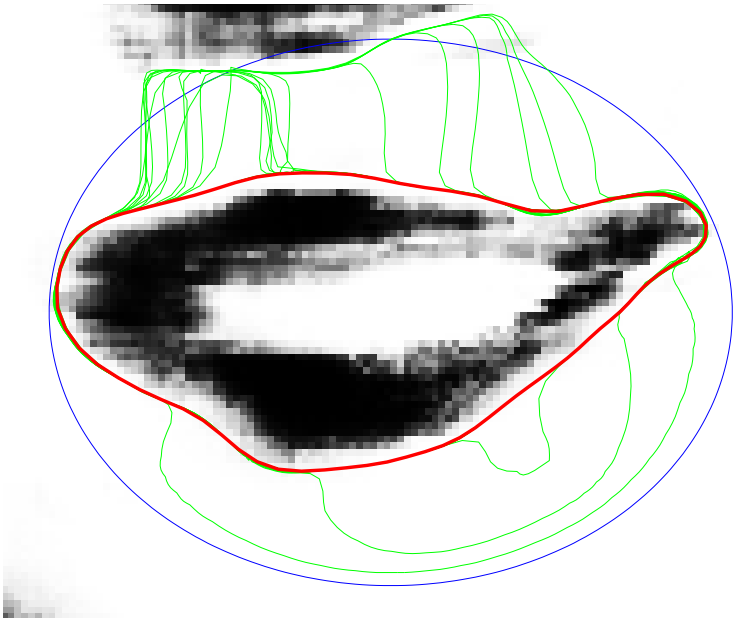
(Figure 4.18(b)), the pinching forces are now replaced by the tension force, and the rigidity and image forces pull the snake back against the edge of the feature. Both before and after the cut takes place, the shape of the snake is very similar.



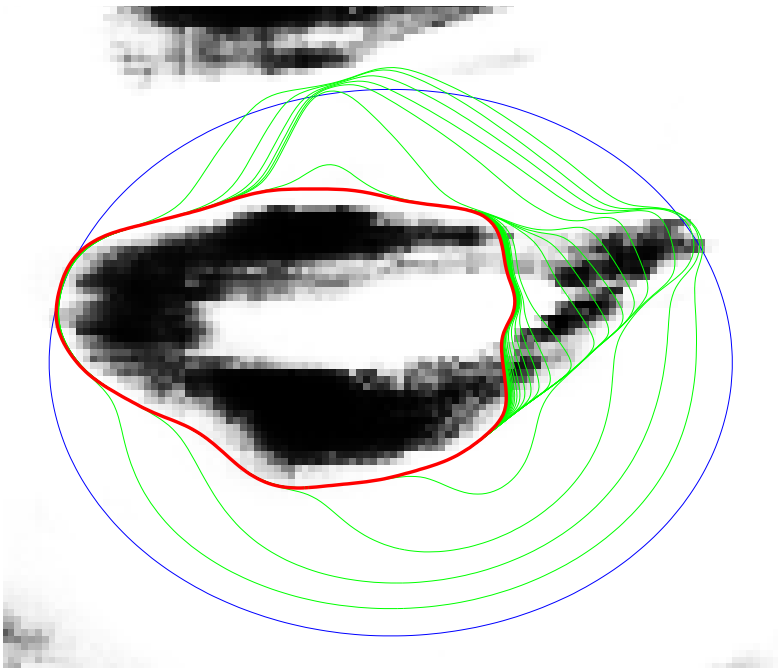
(a) Iteration 22 (b) Iteration 23

Figure 4.18: Balance of forces pre and post cut

Using wrapping snakes, with pinching and cutting, it is possible to successfully find the lip boundary when there are strong noise features and weak target features. This is illustrated in Figure 4.19(a), which can be compared to the traditional snakes in Figure 4.19(b).



(a) Wrapping snake, with pinching and cutting, with strong noise and weak target features



(b) Traditional snake with strong noise and a weak target feature (reprint of Fig. 4.5)

Figure 4.19: Comparing wrapping snakes with pinching and cutting, with traditional snakes for strong noise and weak target features

By combining wrapping snakes with pinch forces and cutting, it is possible to find the lip boundary even with an extremely poor initial position. Figure 4.20 shows a snake that's initial position was along the noise due to shadow under the jaw, as well as enclosing multiple features. In this situation, the snake was still able to successfully locate just the outer lip boundary.

In Figure 4.20, the tension is trying to pull the snake inwards, and the wrapping force near the lip boundary is stronger than near the shadow beneath the jaw. This balance of forces results in the snake beginning to wrap around the upper and lower sections of the lip boundary. Once the various sections of the snake close in towards each other, the pinching force pulls the snake sections towards each other close to the lip boundary. Once these sections meet, the snake is cut and the resulting sections of snake that extend towards the noise features are discarded.

By combining wrapping snakes with pinching and cutting processes, the lip boundary can now be accurately and reliably found. It is robust against strong noise and poor initialisation, increasing the reliability of a lip segmenter based on this algorithm. By improving the reliability and accuracy of the lip segmenter, the performance of a visual speech recognition system can increase.

4.9 Comparing The Improved Algorithm

The differences between traditional snakes, wrapping snakes, and wrapping snakes with pinching and cutting are summarised in Table 4.1. Each algorithm can handle more situations successfully than the previous, but with increased computational cost per iteration. This additional computational cost is partially offset by the reduction in the number of iterations required, as well as only performing some of the additional computations on a subset of points within the snake.

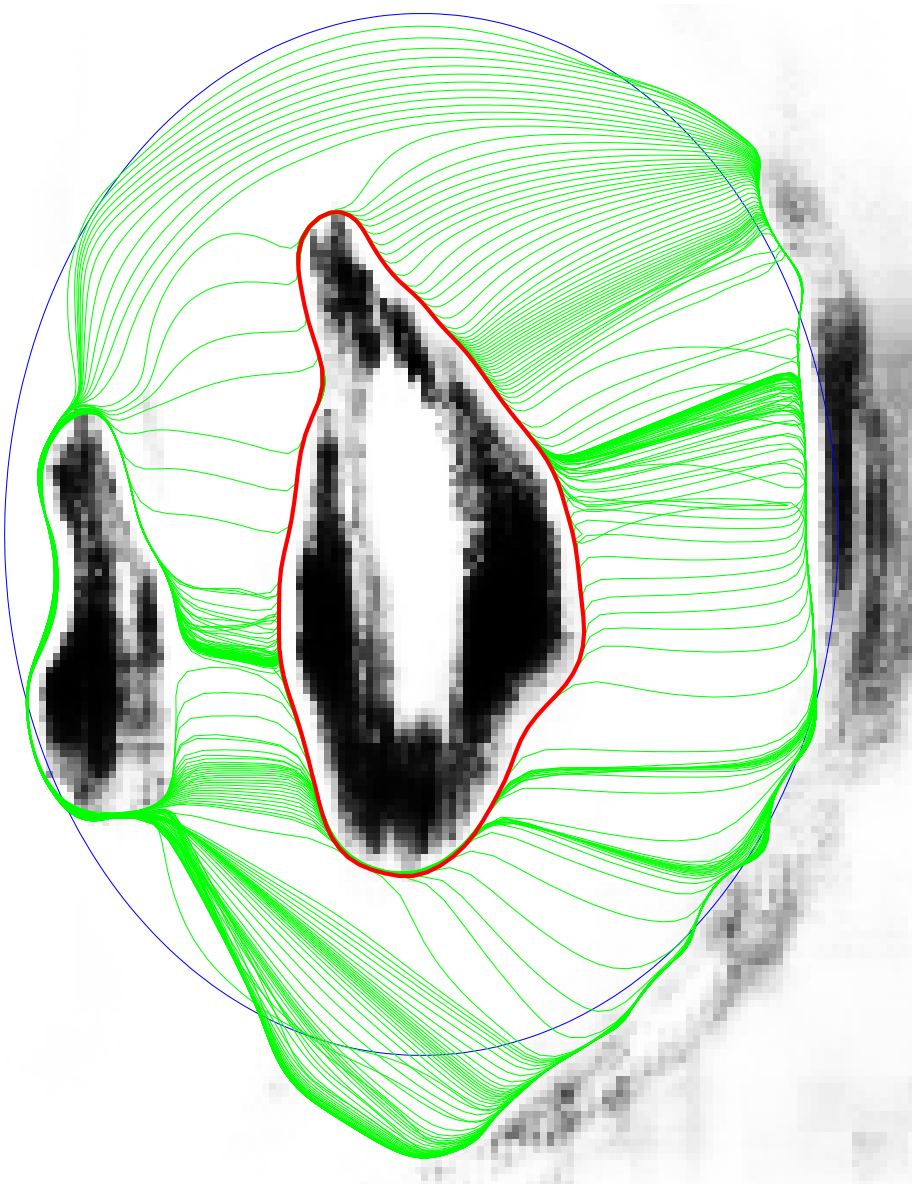


Figure 4.20: Wrapping snake, with pinching and cutting, successfully locating the lip boundary, even with very poor initialisation

Traditional snakes are suitable for ideal images with strong target features, minimal noise, and good initial position. For use as a lip segmenter, they are not very suitable due to the inherent noise present near the lips and possibility for poor initialisation. Substituting in the wrapping force improves the performance of the lip segmenter, by increasing the robustness to noise and initial position, but still cannot completely handle multiple enclosed features. By including the pinching and cutting processes, wrapping snakes can now reliably handle very poor initialisation, multiple enclosed regions, and strong noise near the lips.

Wrapping snakes, when combined with pinching and cutting forces, allow the lip boundary to be found more accurately than with traditional snakes, and are more robust to noise and poor initialisation. As visual speech recognition requires an accurate representation of the lip shape, the use of wrapping snakes can improve the performance of these systems.

By modifying the image force based on the snakes' location and orientation, the wrapping force encourages the snake to continue along features it has partially found. It has been shown that wrapping snakes allow the lip boundary to be found more accurately than with traditional snakes, and are more robust to noise and poor initialisation.

The inclusion of pinching and cutting processes allows the snake to successfully locate the lip boundary under a broader range of conditions, as it can now reliably handle multiple enclosed regions. These characteristics further improve the accuracy of the lip shape as determined by the wrapping snakes.

As visual speech recognition requires an accurate representation of the lip shape, the use of wrapping snakes can improve the performance of these systems. With the requirements of a good lip segmenter being fast operation, robust to noise and poor initialisation, and be easy to use, wrapping snakes with pinching and cutting are well suited for this purpose.

Table 4.1: Comparing traditional snakes, wrapping snakes, and wrapping, pinching, cutting snakes.

	Traditional snake	Wrapping snake	Wrapping snake with pinching and cutting
Initial conditions	Can only handle favourable initial conditions (see Figure 4.4)	Can handle poor initial conditions (see Figure 4.13)	Can handle very poor initial conditions (see Figure 4.20)
Strong noise, weak target	No (see Figure 4.5)	Most of the time (see Figure 4.12(a))	Most of the time (more often than just wrapping snakes) (see Figure 4.19(a))
Multiple enclosed regions	No (see Figure 4.6)	Partially - usually finds each, but cannot distinguish between them (see Figure 4.14(a))	Yes - each is found and identified as a separate feature (see Figure 4.16)
Computational cost (per iteration)	Lowest	Higher	Highest, but is minimised by only performing the additional calculations on a subset of points
Computational cost (number of iterations required)	Highest	Lower, and can handle more situations than traditional snake	Lowest, and can handle the most situations

4.10 Parameterising The Lip Shape Using Wrapping Snakes, With Pinching And Cutting

To produce the feature vector for use by the phoneme recogniser, the lip shape is first extracted using wrapping snakes with pinching and cutting. To eliminate the variability of the number of points due to cutting, 100 points are placed along the snake path with even spacing, with the first point being the right-most point of the mouth. This produces a 100-point list to represent the lip shape.

The point-list is then normalised by subtracting the mean, and dividing by the standard deviation for each position within the list. This produces a list of points each with zero mean and a standard deviation of one, over the length of the video. The purpose of the normalisation is to produce a scale and translation invariant feature. If the mouth takes up more or less of the frame in some videos, or if it moved within the video frame, the coordinates of the snake points would be inconsistent. This would in turn negatively affect the ability of the recogniser to learn the characteristic appearance of each phoneme. By normalising the coordinates, the scale and translation of the lips is removed from the feature vector.

For real-time usage, this normalisation process can be modified by using the mean and standard deviation of a suitable "calibration session", to eliminate the need to have the full video pre-recorded before being able to begin recognition.

The feature vector is then constructed by concatenating the following elements:

- The width of the lips
- The height of the lips

- The first N principal components of the normalised snake point-list
- The deltas and delta-deltas of each of the previous elements

The deltas (velocity) and delta-deltas (acceleration) are included to capture the dynamic movement of the lips, instead of limiting the recogniser to only using the location of lip within each frame. It has been shown that dynamic parameters contain significant speech information, and are more robust to non-speech variances (Chen, 2001; Dupont and Luettin, 2000; Potamianos et al., 2004).

The specific number of principal components, N, is determined empirically. The performance of the recogniser is tested across a range of N to determine the fewest number of principle components required to reach the peak performance. It is desirable to minimise the total length of the feature vector, to reduce the computational load of the recogniser. This will be discussed further in the following chapter.

By using wrapping snakes, with pinching and cutting, the lip shape has been successfully found and parameterised. It is robust to noise, and accurately describes the lip shape. Now that the lip shape has been parameterised, it can be used as an input into the phoneme recogniser, which will use the lip shape to determine the phonemes being spoken. This will be discussed in the next chapter, “Phoneme Recognition Using Hidden Markov Models”.

Chapter 5

Phoneme Recognition Using Hidden Markov Models

Phoneme recognition is the final stage of the speech recogniser. This involves using the sequence of feature vectors, output by the lip feature extractor in the previous chapter, to determine the sequence of phonemes being spoken. The output of the phoneme recogniser is a phonetic transcription of the speech input. This task is performed by a Hidden Markov Model (HMM) based phoneme recogniser.

This chapter discusses the architecture of the HMM-based recogniser, the training process used, and how the recogniser is used to produce the raw phoneme transcription. Finally, the output of the recogniser is briefly discussed.

5.1 Task Syntax

When building a speech recogniser, an important step is to decide which speech units will be modelled. The common choices are whole words for command-style syntax, or individual phonemes for free speech recognisers.

Command-style recognisers use a model for each word or command they need to recognise. This is useful when there are only a limited number of fixed words that are accepted, for example in a voice dialling application. The requirement of a model for each word to be recognised is a major limitation if it has to recognise free speech, as a very large number of models must be trained, and no new words can be easily added without samples for use during training.

The way to avoid this limitation is to use sub-word models, such as phonemes. There are a limited number of well defined phonemes in any given language, allowing a set of models to be trained that can be combined to form any word that has a known pronunciation. If additional words are required to be added after training, the only change is to add them to a pronunciation dictionary that is used during the recognition process. As this recogniser is being used to determine the suitability of visemes as a visual unit of speech, the free speech syntax is used.

5.2 Architecture Of Hidden Markov Model-Based Recogniser

The phoneme recogniser performs the task of identifying the most likely sequence of phonemes to have produced the sequence of lip feature vectors observed. The phoneme recogniser used is based on Hidden Markov Models (HMMs). HMMs allow an underlying sequence of states to be predicted based only on the observations, which have a hidden relationship with the underlying state.

In the context of visual speech recognition, the observations are the lip feature vectors, and the underlying states are the characteristics of the phoneme sequences. By modelling the sequence of feature vectors observed for each phoneme, in various contexts, a hidden Markov model can be constructed for each phoneme.

A Markov model is a finite state machine which changes state once every time unit and each time t that a state j is entered, a speech vector $\mathbf{o}_t$ is generated from the probability density $b_j(\mathbf{o}_t)$. Also, the transition from state i to state j is controlled by the discrete probability a_{ij} (Young et al., 2006).

Figure 5.1 shows an example of the generation of an observation sequence by a five state model as it moves through the state sequence $X = 1, 2, 2, 3, 4, 4, 5$ to generate observations $\mathbf{o}_1$ to $\mathbf{o}_6$. In this model there are three emitting states in addition to the non-emitting entry and exit states (states 1 and 5).

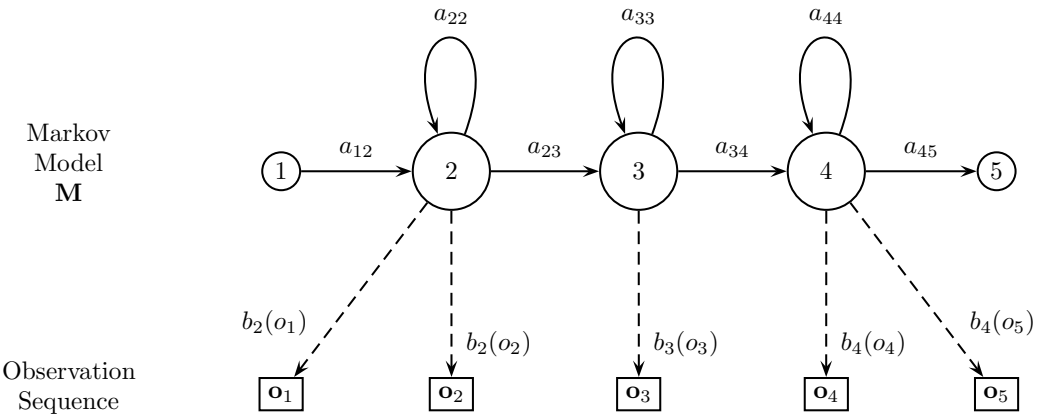


Figure 5.1: The Markov generation model

The joint probability that the observation sequence **O** is generated by the model M moving through the state sequences X is calculated as the product of the transition probabilities and the output probabilities. So for the state sequence X in Figure 5.1, this probability is calculated as shown in Equation 5.1.

$$P(\mathbf{O}, X|M) = a_{12}b_2(\mathbf{o}_1)a_{22}b_2(\mathbf{o}_2)a_{23}b_3(\mathbf{o}_3)a_{34}b_4(\mathbf{o}_4)a_{44}b_4(\mathbf{o}_5) \tag{5.1}$$

However, only the observation sequence **O** is known and the underlying state sequence X is hidden, hence the name *Hidden Markov Model*.

To determine the most likely sequence of states to generate sequence **O**, the likelihood is calculated by summing over all possible state sequences

$X = x(1), x(2), x(3), \ldots, x(T)$. This process requires the state transition probabilities a_{ij}, and the output probability distributions $b_j(\mathbf{o}_t)$ to be known (see "The HTK Book" (Young et al., 2006) for full details on the algorithms used).

The HMM Toolkit (HTK) (Young et al., 2006), used to build the recogniser used in this work, is designed to model continuous parameters using continuous density multivariate output distributions. The output distributions are represented by Gaussian Mixture Densities, and the state transition probabilities are represented by a state transition matrix.

As the goal of this research is to investigate the phoneme confusions, the HMMs are built using phoneme units. A separate HMM is constructed for each phoneme, as well as a silence model, allowing the recogniser to recognise individual phonemes. The phoneme transcript output by the recogniser can be compared to the known input phoneme sequence, enabling the nature of the phoneme confusions to be examined.

The HMMs each contain three emitting states and non-emitting entry and exit states. Each emitting state has two possible transitions. The first possible transition is back to the same state, and the second is to transition to the next state. By allowing the model to stay in the same state for several observations, variable speaking speeds can be handled. The non-emitting entry state has a single possible transition directly to the second state, as required by the HTK.

5.3 Training HMMs

Before the Hidden Markov Models can be used, they must first be trained. Using training data, this process calculates the probability values in the state transition matrix, and the Gaussian mixture densities of the outputs for each state.

Using the HMM Toolkit, a set of HMMs can be trained using sample data obtained from the lip feature extraction (see Chapter 4). The training involves several stages, including data preparation, creating monophone HMMs, and finally creating triphone HMMs.

Monophone HMMs are where each phoneme is modelled by a single HMM. A triphone is similar to a monophone, but it includes the context of the monophone, which allows a more accurate representation. Each phone model is dependent on the previous and next monophone. For example, the word "welcome" as the sequence of monophones:

```
w eh l k ah m
```

becomes:

```
w+eh w-eh+l eh-l+k l-k+ah k-ah+m ah-m
```

where "w-eh+l" represents the /eh/ monophone when preceded by /w/ and followed by the /l/ monophone.

This allows coarticulation effects to be captured by the triphones, as a separate model is created for each contextual variation. As discussed in Section 1.6.1, context can affect the appearance of phonemes. The use of triphones allows this to be better handled than with monophones, as the differences due to context are contained within different models.

5.3.1 Data Preparation

The data preparation stage involves defining the scope of the recogniser, the collection of speech samples with corresponding transcripts, and getting them into a form which is suitable for building the recogniser. The first step in this process is to define the task grammar. The grammar file defines the structure of the speech the recogniser can handle. This can be a command-style syntax,

semi structured syntax, or free form speech. The grammar file for this visual speech recogniser allows for any sequence of words by defining a word loop, as shown in Figure 5.2.

```
$phoneme = ih | iy | aa | ah | ae | ay | eh | ey | hh | ao \
 | aw | ow | oy | uh | uw | w | l | er | r | y | b | p | m \
 | n | s | z | ch | jh | sh | zh | d | dh | g | k | t | th \
 | f | v | ng ;
( SENT-START <$phoneme> SENT-END )
```

Figure 5.2: Grammar file defining a word loop containing each phoneme

The next step is to create the pronunciation dictionary. This file defines the different possible pronunciations for each word that can be recognised. For training this phoneme recogniser, the dictionary must contain the sequence of phonemes that form each word. The dictionary used was derived from the Carnegie Mellon University Pronouncing Dictionary "cmudict0.7a" (Carnegie Mellon University, 2008), as it is a widely accepted phonetic pronunciation dictionary.

This pronunciation dictionary contains over 125,000 words, and uses the ARPAbet set of phonemes (Carnegie Mellon University, 2008). These phonemes are used as they are named using regular ASCII characters suitable for use with computer processing. Table 5.1 shows the list of phonemes used, along with a sample word and pronunciation.

Next, the actual speech samples are collected and transcribed. As discussed in Section 2.3, the VidTIMIT (Sanderson and Paliwal, 2002) dataset was used to train the HMMs. The speech samples in this dataset include their corresponding transcriptions, which are required as part of the training process.

It is preferable that the majority of the speech samples do not have a restricted grammar. This is to improve the coverage of the phoneme set, and ensures each phoneme is used in multiple contexts to improve the modelling accuracy.

Table 5.1: Sample pronunciations from the CMU Pronouncing Dictionary, demonstrating each phoneme (Carnegie Mellon University, 2008)

Phoneme	Example	Translation	Phoneme	Example	Translation
AA	odd	AA D	L	lee	L IY
AE	at	AE T	M	me	M IY
AH	hut	HH AH T	N	knee	N IY
AO	ought	AO T	NG	ping	P IH NG
AW	cow	K AW	OW	oat	OW T
AY	hide	HH AY D	OY	toy	T OY
B	be	B IY	P	pee	P IY
CH	cheese	CH IY Z	R	read	R IY D
D	dee	D IY	S	sea	S IY
DH	thee	DH IY	SH	she	SH IY
EH	Ed	EH D	T	tea	T IY
ER	hurt	HH ER T	TH	theta	TH EY T AH
EY	ate	EY T	UH	hood	HH UH D
F	fee	F IY	UW	two	T UW
G	green	G R IY N	V	vee	V IY
HH	he	HH IY	W	we	W IY
IH	it	IH T	Y	yield	Y IY L D
IY	eat	IY T	Z	zee	Z IY
JH	gee	JH IY	ZH	seizure	S IY ZH ER
K	key	K IY			

The VidTIMIT dataset meets these requirements as it contains free speech covering all phonemes.

The final step of the data preparation stage is to code the data into a form suitable for use during creation of the recogniser. This involves parameterising the speech sample files into sequences of feature vectors. As discussed in Section 4.10, the feature vector contains the width and height of the lips, the top 58 principal components of the lip boundary, and the deltas and delta-deltas of each of these elements. This allows the dynamic characteristics of the speech process to be better captured by the feature vector. Through empirical testing, it was found that including the top 58 principal components obtained the highest performance of the recogniser.

5.3.2 Creating Monophone And Triphone HMMs

Now that the data have been prepared, monophone HMMs can be created. Monophone HMMs have a single model for each individual phoneme, and the silence model, without creating variations for different contexts. This stage involves defining the structure of the HMMs, creating the initial prototypes, and training them using the speech samples and corresponding transcripts. The training involves several iterations of embedded training, the silence models are then fixed, followed by several more iterations of model re-estimation. This is followed by forced alignment of the transcripts, and finally re-estimating the models several more times.

The first step in creating the monophone HMMs is to create a set of flat start monophone models. This is done by creating an identical model for each phoneme, where all means and variances are equal to the global mean and variance of the training data.

Once the prototypes have been created, embedded training is used to start to fit the models to the real data. For each training sample, the label file is loaded, and generates a composite HMM for the entire sequence. This composite HMM is created by concatenating the appropriate phoneme HMMs, according to the transcript and pronunciation dictionary. As there is no way to know which pronunciation is used if multiple are available for a given word, the first to occur in the dictionary will be used. The Forward-Backward algorithm is used to determine the most likely boundary positions for each state, which then allows the means and variances to be updated for each phoneme model (Young et al., 2006).

By this stage, there is a three state left-to-right HMM for each phoneme and the silence model "sil". Now that the models have undergone some initial training, the silence model needs to be fixed. In this step, additional transitions from state two to four and from four to two are inserted into the silence model. This will make it more robust to noise, as it allows it to absorb noise, then transition back to an earlier state within the model.

Another change that is made is the creation of a one-state short pause "sp" model. This is a so-called "tee-model", as it has a direct transition from the entry to the exit node. As a result, this model can be passed immediately through, without having to pass through the emitting state. Finally, the emitting state is tied to the centre state of the silence model, allowing it to be trained. The structure of these two silence models are shown in Figure 5.3.

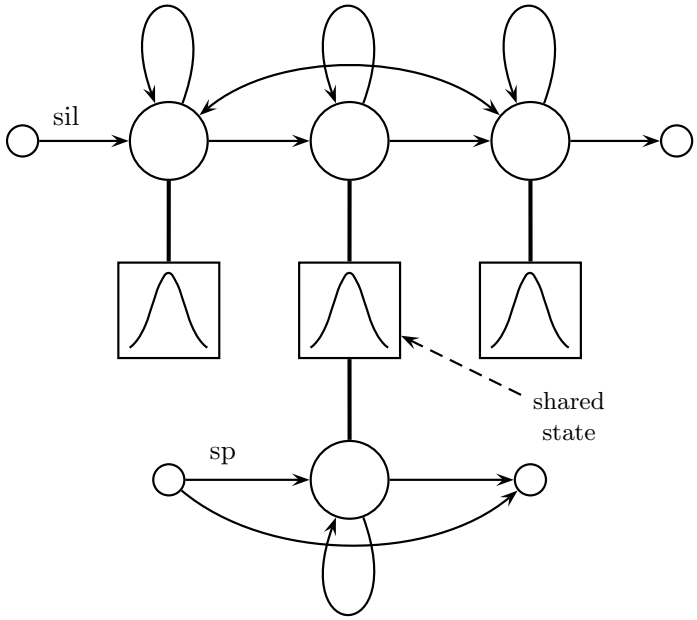


Figure 5.3: Silence Model (Young et al., 2006)

Once these silence models have been created, another two passes of training are run with the "sp" model inserted between each word in the transcription. This allows the "sp" model to be trained when present, while the direct transition from entry to exit node prevents it from interfering with the training of the other models. If this direct transition was not present, it would not allow for words to occur with no pause occurring between each. This would result in the training using incorrect data for both the "sp" model and for each phoneme adjacent within the training data, which would negatively impact the recognition capabilities of these models.

Now that the models are trained to a basic level, and the silence models are created, the correct pronunciation used in each training sample can be

determined. This step realigns the phone-level transcriptions, and chooses the most likely pronunciation for each word utterance.

The "HVite" tool from the HMM Tool Kit is used to perform the forced alignment. The tool creates a network from the word level transcription and the pronunciation dictionary, with the alternative pronunciations included in parallel. It then uses the Viterbi algorithm to find the best path through this network (Young et al., 2006). This selects the most appropriate pronunciation to use when fitting the phonemes to the word transcription. Now that the individual pronunciations have been determined, another two passes of training are run to re-estimate the HMM parameters. The result from this stage is a set of monophone HMMs.

Given a set of monophone HMMs, the next step is to build a set of triphone HMMs. To create the triphone HMMs, first the monophone transcriptions are converted to triphone transcriptions. For each triphone required, a triphone HMM is created by copying the HMM from the centre phone. For example, the "z-iy+r" triphone will be based on a copy of the "iy" phone. Once these initial triphones have been created, they can be re-estimated using the same algorithms used earlier with the monophone HMMs.

The process of creating the triphone HMMs makes use of decision tree clustering to reduce the amount of training data needed to produce properly trained HMMs. The process uses a commonly used audio-based decision tree (VoxForge, 2008), combined with a decision tree that clusters phonemes from the standard viseme set that will be discussed in Section 6.1. If this is to produce any bias in the results of the recogniser, more than what is normal for audio-based recognisers, it would be towards phonemes that can be grouped into visemes.

The basic HMM uses a single Gaussian to model the output distribution. By increasing the number of Gaussians, the HMM can better model more complex output distributions. Once reasonable tied-state triphone models have been trained, the "HHEd" tool can be used to increase the number of

Gaussian mixtures per state. It has been found that two Gaussian mixtures per state give optimum performance for this task.

5.4 Recognition Of Speech Using HMMs

The recogniser uses a recognition network to create a transcript from a provided speech sample. This network can be considered at three different levels: word, model, and state. The network is constructed by first creating a word level network. This is done using the grammar definition, which defines the sequence of words that is allowed. The model level is created by replacing each word within the network with the phoneme models according to the pronunciation dictionary. These models contain the individual states, resulting in a network that defines the allowable state sequences (see Figure 5.4).

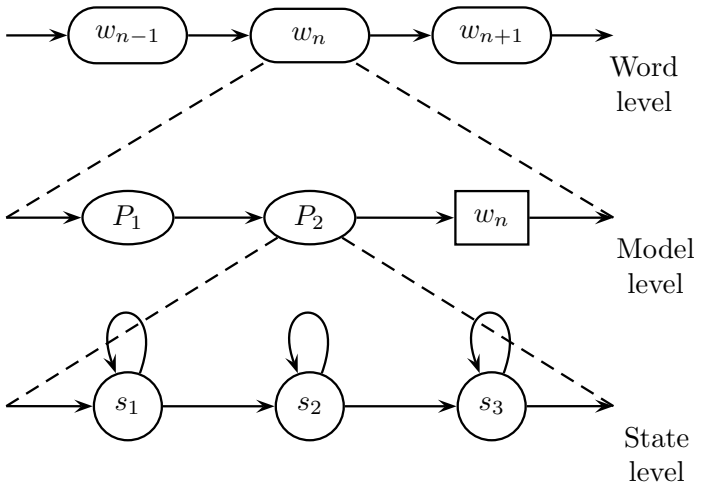


Figure 5.4: Recognition network levels

The probability of a given path through this network producing an observed output can be calculated, allowing the path with the maximum likelihood to be found. When the network is viewed at a higher level, the sequence of phonemes, and in turn words, can be seen.

The recognition process makes use of the Viterbi algorithm to determine the path through the recognition network most likely to have created a given sequence of input vectors (Young et al., 2006).

5.4.1 Balancing Parameters To Maximise Performance

There are many parameters involved in the HMMs, each of which can affect the performance of the recogniser. In particular, these include the word insertion penalty for transitioning to another word, and the number of Gaussian mixtures used to model the observations for each node. By adjusting each of these parameters, the number of insertions, deletions, substitutions, and correct matches is affected. These need to be balanced to achieve the most accurate output from the recogniser.

The word insertion penalty controls the cost applied when transitioning from one word to the next. If this cost is too low, the recogniser will be more likely to insert many short words, whereas if it is too high, it will be likely to favour fewer but longer words. This parameter can heavily influence the balance between insertion, deletions, substitutions and correctly matched phonemes. Figure 5.5 shows how the insertion penalty affects the number of insertions, deletions, substitutions, and correct phoneme matches.

As can be seen in Figure 5.5, as the insertion penalty increases, the number of phoneme deletions drops, and the number of substitutions increases significantly. The number of insertions rises gradually, while the number of correct phonemes rises slightly initially, then levels off. As a balance is needed between each of these parameters, an insertion penalty of 20 was used. This maximises the number of correct phonemes, while achieving a compromise between the other possible operations.

Another parameter that is important in determining the effectiveness of the recogniser is the number of Gaussian mixtures used to model the output

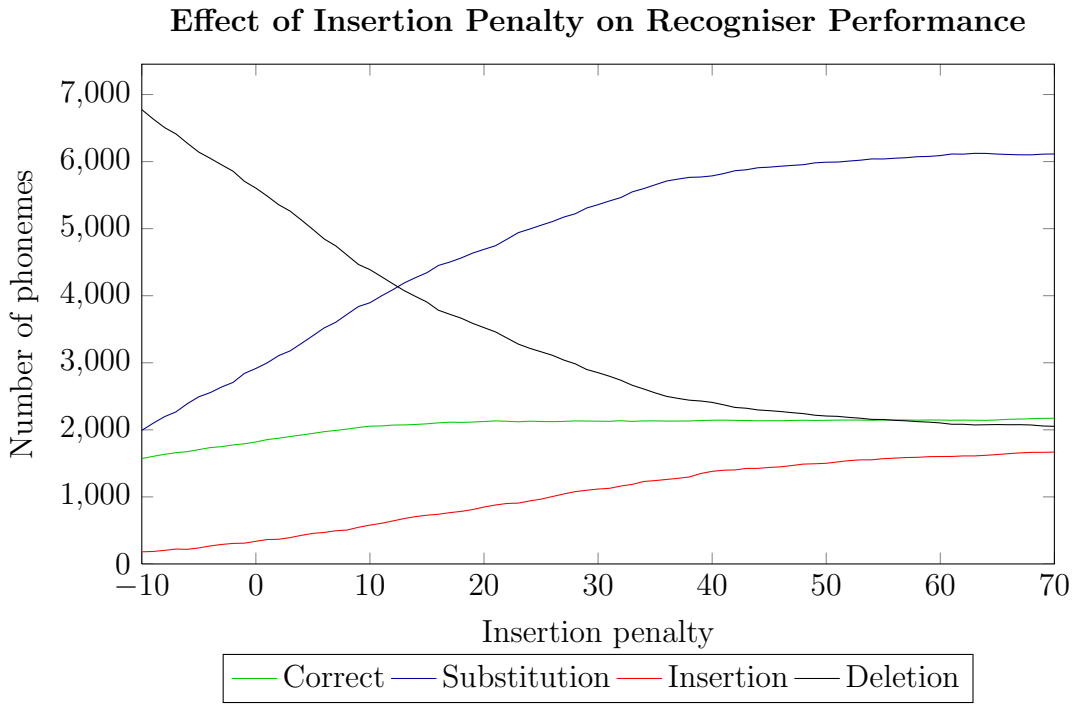


Figure 5.5: The effect of insertion penalty on the performance of the phoneme recogniser

distribution of each node within the HMMs. As the number of mixtures increases, the complexity of the system increases, as does the complexity of the output distributions that can be fitted. If too many Gaussians are used, the data may be over-fitted, reducing the generalisability of the system. This would result in a network that could identify the training data very well, but due to the over-fitting, would perform less effectively for data that was not presented during training.

The balance between the types of errors is an important choice to be made, with different mechanisms able to handle different types of errors. One way to handle insertion errors is to determine the likelihood of each phoneme being an insertion error, and using this to determine a cost to apply if this phoneme is to be deleted when fitting words to the sequence of phonemes output from the recogniser. In a similar way, deletion errors can be handled

by applying a cost related to the frequency that the phoneme is a deletion error.

The handling of deletion errors is limited in that the only information available is the frequency each phoneme is deleted, with limited ways to narrow down the choice of which phoneme, if in fact any, was deleted. The handling of insertion errors is slightly easier, as the system knows which phoneme is being considered, resulting in just a single frequency to be considered how often this phoneme appears as an insertion error. This makes insertion errors easier to handle in comparison to deletion errors.

Substitution errors are the most interesting, as they are key to testing the validity of visemes. Historically, visemes have been used to minimise the number of substitution errors (see Section 1.5). As the primary goal is to determine if the characteristics of visemes are present, they cannot be used in the recogniser being built.

With parameter values decided, the HMM-based recogniser is now able to be used to perform the phoneme recognition task. The process of performing the recognition is detailed in the following section.

5.4.2 Running The Recogniser To Produce A Phoneme Transcript

Once the recogniser has been trained, it can be used to recognise the speech in the test set. As the goal of the research is to test the validity of visemes, the recogniser is configured to output raw phonemes, without trying to fit words to the sequence. By skipping the word-fitting process, the output of the recogniser is based purely on the appearance of the phonemes, and not on any biases present within the language model, pronunciation dictionary, or the English language.

As the HMM tool kit, HTK, is designed primarily to build complete word recognisers, a pronunciation dictionary needs to be provided for the recognition process. As such, a special phoneme-only dictionary is used where the only "words" are the individual phonemes themselves, with the pronunciation simply being the phoneme followed by the optional short-pause ("sp") model (see Figure 5.6). The dictionary also contains special tokens to allow for silence at the start (`SENT-START`) and end (`SENT-END`) of each sentence. These special tokens do not output any symbol into the transcript, resulting in a pure phoneme transcript being output by the recogniser.

```
aa          [aa]    aa sp
ae          [ae]    ae sp
ah          [ah]    ah sp
ao          [ao]    ao sp
aw          [aw]    aw sp
ay          [ay]    ay sp
b           [b]     b sp
...
y           [y]     y sp
z           [z]     z sp
zh          [zh]    zh sp
SENT-END    []      sil
SENT-START  []      sil
```

Figure 5.6: Extract of pronunciation dictionary used to produce raw phoneme output

By using the tools provided within the HTK, the recogniser comprised of the triphone HMMs is used to produce a phoneme transcript for the sequence of input visual speech feature vectors. The results are briefly discussed in the following section.

5.5 Recognition Output

As discussed in the previous section, the output of the phoneme recogniser is a phonetic transcription of the speech input. This output can be compared

to the reference phonetic transcription to analyse the performance of the recogniser.

The output of the recogniser can be scored using the following two equations

$$Percentage\ Correct = \frac{H}{N} * 100\% \tag{5.2}$$

$$Accuracy = \frac{H - I}{N} * 100\% \tag{5.3}$$

where H is the number of correct labels, N is the number of labels in the reference transcription, and I is the number of insertions. These calculations are based on a dynamic programming-based string alignment procedure provided by the HTK tool "HResults" (Young et al., 2006).

Using Equation 5.2 and Equation 5.3, the recogniser is calculated to have a phoneme correctness of 26.7%, and an accuracy of 18.6%. This indicates that just over one quarter of phonemes in the input sequence were recognised correctly. The accuracy is lower than the correctness due to insertions in the output.

To further examine the results, a confusion matrix is constructed, where the reference input is compared to the output of the recogniser. This confusion matrix is shown in Table 5.2.

One feature of particular interest in this confusion matrix is the strong diagonal component. As entries on the diagonal represent correct phoneme matches, this indicates the recogniser is able to correctly recognise individual phonemes within free speech. Of the $10,339$ phonemes in the reference transcript, 26.7% were correctly recognised. Due to insertions and deletions, this equates to 36.6% of the phonemes in the output transcript being correct.

Another feature exhibited in the confusion matrix is a general vertical trend in the results. The columns represent the phoneme being recognised, which means that some phonemes are much more likely to be substituted in place of any phoneme, and not just a small number of other phonemes. The opposite is also true, with 14 of the 39 phonemes exhibiting a very low number of false

Table 5.2: Confusion matrix comparing input phonemes (rows) to output phonemes (columns) (Note: "zero" entries have been blanked for clarity)

	ih	iy	aa	ah	ae	ay	eh	ey	hh	ao	ow	w	aw	oy	uh	uw	l	er	r	y	b	p	m	n	z	s	ch	jh	sh	zh	d	k	t	dh	g	th	f	v	ng	(del)
ih	182	11	7	24	9	6	7		1	1	7	10	1				8	2	9	1			3	11	6	25					5	5	9	36						205
iy	12	207	4	26	2	5	4	4	3	5	10	20					11	1	6		2		3	10	7	21					4	4	13	35						175
aa	5	11	41	6	4	1	1				2	2					6	4	3		1	1	1	5		10					2	4	2	14				1		80
ah	17	19	3	284	8	7	13		2	5	7	11					11	3	25	1			3	7	5	14					4	4	7	35			1			205
ae	11	8	1	20	82	3	1	2	5	1	4	4					7		18		1		1	11	2	7					2	5	4	23						143
ay	5	6		7	1	51	2		1			4					4		2					4	2	10						1	2	11				1		55
eh	11	7	2	5	3	2	52			1	1	6					6	2	11	1	3		2	6	4	9					2	1	8	18			1	1		95
ey	1	3	1	9	3	2	5	32		1	1	4					2	1	5				1	4		6					2	5	2	7	2					77
hh	4	1	1	3	3		1		27		3	3	1				6		6				1	1	1	4					1	1	3	5						73
ao	10	4		14	2		3	2		52	3	3					6	3	7		2		1	8		10						3	4	15						61
ow	7	3	1	10	3		1			1	46						6		2			1		3	2	2					1	3	2	3						55
w	3	3	2	11	2	1	4		1	2	2	77					8	1	8		1		3	5	3	6						3	2	10	1					68
aw	1	2	1	1	1	1		1		1	1	4	12						5						1									2						15
oy	3	1		2						1				12			1	1					1	1		1						2	1	1						29
uh	5	3		1						1		3			11		1	2	2					1	3	5							1	1						25
uw	6	8		9	3	1	2	1			4	3				19	4	1	4	2			2	5	1	5					2	2	4	10						69
l	8	16	3	22	5	4	6	1	3	3	6	10					152	3	16				4	24	9	14					4	7	10	26						150
er	16	12	1	18	5	3	3	1	3	4	4	10					3	47	17				3	10	8	15					3	8	5	26						118
r	14	6	2	21	5	2	7	1	5	2	10	14					18	2	191	2			4	11	4	21					3	7	11	29			1	1		142
y	6	4	1	4		2			2		2	1					3	2	7	27	2		1	6	1	2					2	2	7	10						63
b	1	5		7	3	1	2			3	3	3					5	1	11		25		1	7		6						2	2	11						84
p	8	8		4	1	2	2	1	1	2	6	8					9	2	4		1	22	1	2	3	8				1	1	2	6	9			1			108
m	5	6	2	13	4	2	4	2		1	3	4					5		17			1	70	6	3	9					1	6	7	19				1		126
n	10	14	5	21	2	4	3		3	3	8	9					8	5	19	3	3		6	217	5	18					4	6	5	25				1		198
z	2	9	2	12	5	3	3		1	1		4					4	1	10				2	12	79	11					1	5	5	15						95
s	14	10		11	4		7	1	1	1	5	6					8	6	28	3			1	6	9	185					7	4	10	22						159
ch	2	1		2	1		1		1								2		3					1		1	6				2		4	2						18
jh	1	2				1			1		2	1					2		2						1	1		10				1	2	2						17
sh	7	3		2	1	1	1				2	2	1				4		5				2	3	3	5			21		2	1	4	10						69
zh				1			1					1							2		1			1						5			1	3						1
d	9	7	4	15	4	4	8	1	4	4	8	4					9		9		3		4	11	12	12					63	5	10	18						177
k	6	18	1	12	5	3	3		4	4	6	13					8	3	11	1	1	1	4	18	6	18					1	87	6	19				2		158
t	11	14	3	26	4	3	6		5	4	4	10					18	1	18	2	1	2	3	11	21	13		1			4	8	191	26				2		204
dh	8	1		7	3	1	1	2	1	1		5					5	1	6		1		1	6	3	5					1	2	5	84						69
g	5	6	3		5	1	2			1	3	8				1	3	2	2		1		1	2	3	12						4	7	16	19					60
th		3		3			2				1	1					2		3				2	1	1	2						1		2		5				23
f	7	7	2	4	2	1	1				3	1	2				7		9	3			1	2	1	5					1	3	6	14	1		22			74
v	1	1		4	5	1	4			2	2	1							5	1	1		1	2	2	5						1	3	12				29		61
ng	2	1		4		1		1		1	1	2					1		2	1					2	6						2	2	3					13	31
(ins)	61	54	13	76	24	26	30	7	7	5	24	45					58	18	46	4	2	2	12	45	26	51			2		14	25	36	110	1		3	7		

positives, resulting in mostly empty columns in the confusion matrix. This indicates that these phonemes are not likely to be substituted in place of other phonemes, and that their appearance in the output transcript is very likely to be caused by the correct recognition of the input phoneme.

There are also phonemes that are almost never recognised as a false positive in place of the correct phoneme. When combined with the general vertical trend in the matrix, this indicates that while some phonemes are far less likely to generate false positives, all phonemes are generating false negatives.

When examining the confusion matrix, there are no immediately obvious clusters that can be definitively labelled as visemes. The results will be analysed in detail in the following chapters to determine if any viseme groupings fit the data obtained from the recogniser, and if the required characteristics exist to support the use of visemes as the basic visual unit of speech.

Chapter 6

Performance Of Viseme Groupings

In the previous chapters, a visual speech recogniser has been constructed. This recogniser has been trained to recognise phonemes in video, producing a phoneme transcript. By comparing this output transcript with a reference transcript, various sets of viseme groupings can be tested to see if they exhibit the characteristics, C1–C3, required to satisfy the hypothesis.

A true viseme should exhibit significantly higher intra-viseme substitutions than inter-viseme substitutions, with at least 70% of possible responses occurring within the viseme (characteristic C1). All phonemes within a viseme should have significant substitution errors for all other phonemes belong to the same visemes (characteristic C2). Finally, the intra-viseme substitutions should be non-directional, indicating that phonemes within a viseme are indistinguishable (characteristic C3).

In this chapter, a number of viseme groups will be examined. The first is a traditional viseme grouping based on the groups used by other researchers. The next grouping is based on these traditional groups, but with the phonemes systematically regrouped based on observed characteristics. The next

grouping is based on how "noisy" each phoneme is, that is, based on how likely each phoneme produces substitutions. This grouping maximises the intra-viseme substitutions, providing an upper bound for the accuracy metric. Finally, the existing groups from the literature are examined.

Through this examination, it will be shown that none of the existing viseme groupings exhibit the characteristics required according to the hypothesis in Section 2.1, strongly suggesting the concept of visemes is flawed.

6.1 Traditional Viseme Grouping

The traditional viseme groups are derived from common groupings used by other researchers (Goldschen, Garcia, and Petajan, 1994; Hazen et al., 2004; Lucey, Martin, and Sridharan, 2004; Potamianos et al., 2004). This grouping is shown in Table 6.1. The names of the visemes derive from how each is formed. For example, the "LFr" viseme contains labiodental fricative phonemes (/f/ and /v/).

Table 6.1: Mapping phonemes to traditional visemes

Viseme	Member Phonemes
OV	/ih, iy/
BV	/aa, ah/
FV	/ae, ay, eh, ey, hh/
RV	/ao, aw, ow, oy, uh, uw, w/
L	/l/
R	/er, r/
Y	/y/
LB	/b, p/
LCl	/m/
AlCl	/n, s, z/
Pal	/ch, jh, sh, zh/
SB	/d, dh, g, k, t, th/
LFr	/f, v/
VlCl	/ng/

Table 6.2 shows the confusion matrix for the visual speech recogniser trained to recognise individual phonemes, with the traditional viseme groups labelled. This confusion matrix is obtained by taking the output from the recogniser (see Table 5.2 on page 135), and rearranging it using the viseme groups shown in Table 6.1.

As can be seen in Table 6.2, several visemes have zero intra-viseme substitutions. A good example of this is the "Pal" viseme, which contains the /sh/, /ch/, /jh/, and /zh/ phonemes. Within this viseme, there is not a single intra-viseme substitution, clearly indicating that each of the contained phonemes can be distinguished from one another in the visual domain.

When grouping the phonemes using traditional visemes, the correctness metric (see Equation 5.2 on page 134) increases from 26.7% for individual phonemes, to 29.7%. This is due to the small number of intra-viseme substitutions being reclassified as being a correct match.

A summary of the confusion matrix is shown in Table 6.3. This table shows the number of times each viseme was recognised, how many of these were correct, the insertions, deletions, intra-viseme substitutions, and both false positive and false negative inter-viseme substitutions. This table shows that each viseme has relatively few intra-viseme substitutions in comparison to inter-viseme substitutions. This data is summarised in Figure 6.1, illustrating the likelihood of each operation to cause a traditional viseme to be output by the recogniser. From Figure 6.1, it is clear that the intra-viseme substitutions are much less common when compared to the inter-viseme substitutions.

Characteristic C1 of the hypothesis states that the intra-viseme substitutions should be significantly higher than the inter-viseme substitutions, and that each viseme should be clustered with 70% of possible responses occurring within the correct viseme, that is the correct phoneme or an intra-viseme substitution. As Figure 6.1 illustrates, there are only two visemes, "Pal" and "VlCl", that achieve this 70% threshold, and neither of these contain any intra-viseme substitutions. The other component of characteristic C1 requires

Table 6.2: Confusion matrix with traditional visemes labelled. (NB: "zero" entries have been blanked for clarity)

		OV		BV		FV					RV							L	R		Y	LB		LCl	AlCl			Pal			SB						LFr		VlCl	
		ih	iy	aa	ah	ae	ay	eh	ey	hh	ao	ow	w	aw	oy	uh	uw	l	er	r	y	b	p	m	n	z	s	ch	jh	sh	zh	d	k	t	dh	g	th	f	v	ng	(del)
OV	ih	182	11	7	24	9	6	7		1	1	7	10	1				8	2	9	1			3	11	6	25					5	5	9	36						205
	iy	12	207	4	26	2	5	4	4	3	5	10	20					11	1	6		2		3	10	7	21					4	4	13	35						175
BV	aa	5	11	41	6	4	1	1				2	2					6	4	3		1	1	1	5		10					2	4	2	14				1		80
	ah	17	19	3	284	8	7	13		2	5	7	11					11	3	25	1			3	7	5	14					4	4	7	35			1			205
FV	ae	11	8	1	20	82	3	1	2	5	1	4	4					7		18		1		1	11	2	7					2	5	4	23						143
	ay	5	6		7	1	51	2		1			4					4		2					4	2	10						1	2	11				1		55
	eh	11	7	2	5	3	2	52			1	1	6					6	2	11	1	3		2	6	4	9					2	1	8	18			1	1		95
	ey	1	3	1	9	3	2	5	32		1	1	4					2	1	5				1	4		6					2	5	2	7	2					77
	hh	4	1	1	3	3		1		27		3	3	1				6		6				1	1	1	4					1	1	3	5						73
RV	ao	10	4		14	2		3	2		52	3	3					6	3	7		2		1	8		10						3	4	15						61
	ow	7	3	1	10	3		1			1	46						6		2			1		3	2	2					1	3	2	3						55
	w	3	3	2	11	2	1	4		1	2	2	77					8	1	8		1		3	5	3	6						3	2	10	1					68
	aw	1	2	1	1	1	1		1		1	1	4	12						5						1									2						15
	oy	3	1		2						1				12			1	1					1	1		1					2	1	1							29
	uh	5	3		1						1		3			11		1	2	2					1	3	5							1	1						25
	uw	6	8		9	3	1	2	1			4	3				19	4	1	4	2			2	5	1	5					2	2	4	10						69
L	l	8	16	3	22	5	4	6	1	3	3	6	10					152	3	16				4	24	9	14					4	7	10	26						150
R	er	16	12	1	18	5	3	3	1	3	4	4	10					3	47	17				3	10	8	15					3	8	5	26						118
	r	14	6	2	21	5	2	7	1	5	2	10	14					18	2	191	2			4	11	4	21					3	7	11	29			1	1		142
Y	y	6	4	1	4		2			2		2	1					3	2	7	27	2		1	6	1	2					2	2	7	10						63
LB	b	1	5		7	3	1	2			3	3	3					5	1	11		25		1	7		6						2	2	11						84
	p	8	8		4	1	2	2	1	1	2	6	8					9	2	4		1	22	1	2	3	8				1	1	2	6	9			1			108
LCl	m	5	6	2	13	4	2	4	2		1	3	4					5		17			1	70	6	3	9					1	6	7	19				1		126
AlCl	n	10	14	5	21	2	4	3		3	3	8	9					8	5	19	3	3		6	217	5	18					4	6	5	25				1		198
	z	2	9	2	12	5	3	3		1	1		4					4	1	10				2	12	79	11					1	5	5	15						95
	s	14	10		11	4		7	1	1	1	5	6					8	6	28	3			1	6	9	185					7	4	10	22						159
Pal	ch	2	1		2	1		1		1								2		3					1		1	6				2		4	2						18
	jh	1	2				1			1		2	1					2		2						1	1		10				1	2	2						17
	sh	7	3		2	1	1	1				2	2	1				4		5				2	3	3	5			21		2	1	4	10						69
	zh				1			1					1							2		1			1						5			1	3						1
SB	d	9	7	4	15	4	4	8	1	4	4	8	4					9		9		3		4	11	12	12					63	5	10	18						177
	k	6	18	1	12	5	3	3		4	4	6	13					8	3	11	1	1	1	4	18	6	18					1	87	6	19				2		158
	t	11	14	3	26	4	3	6		5	4	4	10					18	1	18	2	1	2	3	11	21	13		1			4	8	191	26				2		204
	dh	8	1		7	3	1	1	2	1	1		5					5	1	6		1		1	6	3	5					1	2	5	84						69
	g	5	6	3		5	1	2			1	3	8				1	3	2	2		1		1	2	3	12						4	7	16	19					60
	th		3		3			2				1	1					2		3				2	1	1	2						1		2		5				23
LFr	f	7	7	2	4	2	1	1				3	1	2				7		9	3			1	2	1	5					1	3	6	14	1		22			74
	v	1	1		4	5	1	4			2	2	1							5	1	1		1	2	2	5						1	3	12				29		61
VlCl	ng	2	1		4		1		1		1	1	2					1		2	1					2	6						2	2	3					13	31
(ins)		61	54	13	76	24	26	30	7	7	5	24	45					58	18	46	4	2	2	12	45	26	51			2		14	25	36	110	1		3	7		

Table 6.3: Summary of traditional viseme errors

Viseme Group	Number of Entries	Correct phoneme	Intra-viseme substitutions	Inter-viseme (FP) substitutions	Inter-viseme (FN) substitutions	Deletions	Insertions
OV	992	389	23	465	393	380	115
BV	827	325	9	404	289	285	89
FV	695	244	34	323	399	443	94
RV	685	229	29	353	350	322	74
L	421	152	0	211	204	150	58
R	671	238	19	350	362	260	64
Y	52	27	0	21	67	63	4
LB	82	47	1	30	166	192	4
LCl	146	70	0	64	121	126	12
AlCl	1285	481	61	621	401	452	122
Pal	46	42	0	2	112	105	2
SB	1548	449	135	778	603	691	186
LFr	75	51	0	14	137	135	10
VlCl	13	13	0	0	32	31	0
Total	7538	2757	311	3636	3636	3635	834

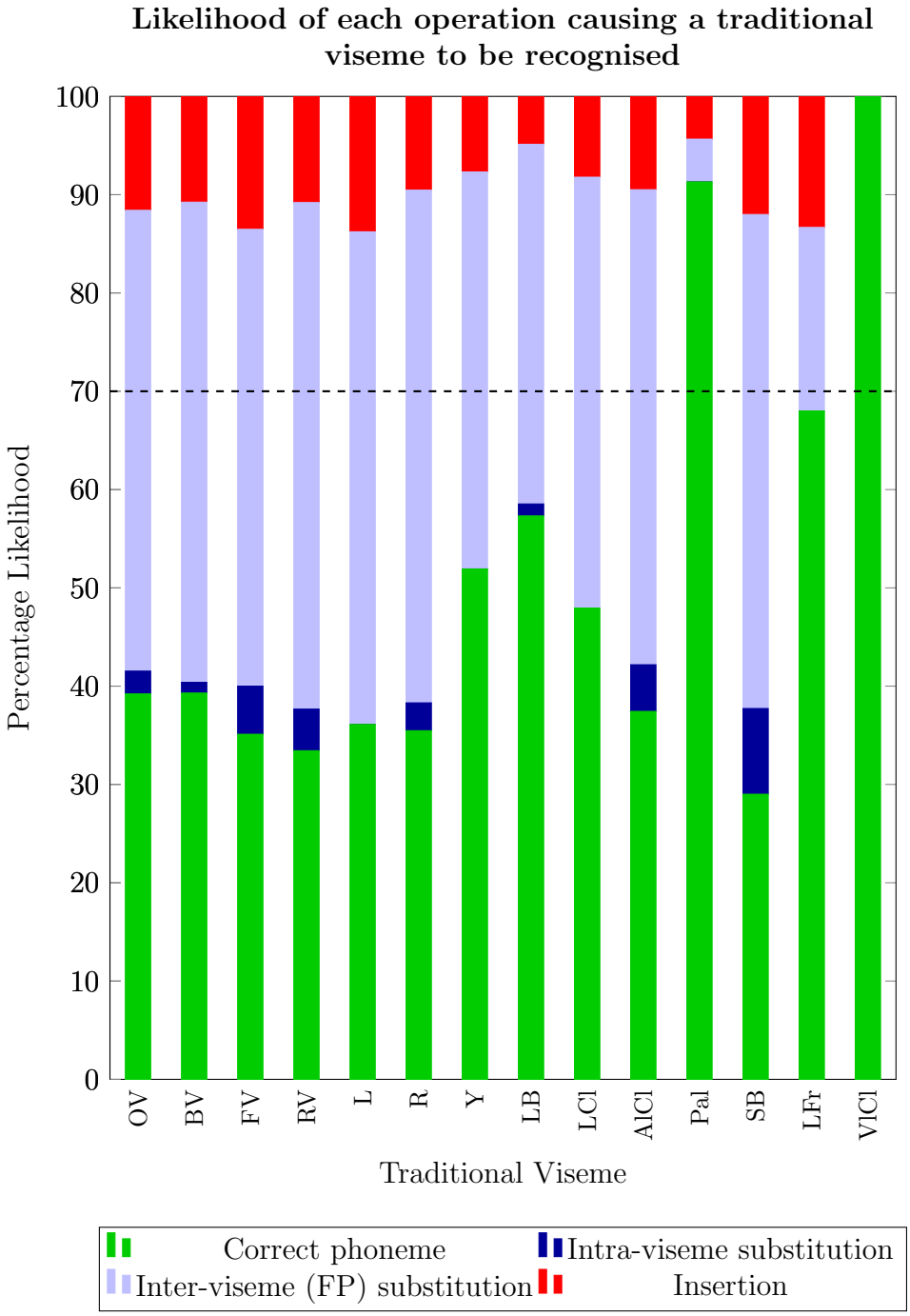


Figure 6.1: The likelihood of each operation causing a phoneme to be recognised, and cause a traditional viseme to appear in the recognition output

significantly higher intra-viseme substitutions than inter-viseme substitutions. Again, Figure 6.1 clearly illustrates that this is not achieved by the traditional viseme grouping, as this grouping produces less intra-viseme substitutions than inter-viseme substitutions.

According to characteristic C2, all phonemes within a viseme should exhibit high confoundedness with all other phonemes belonging to the same viseme. When the results in Table 6.3 are examined, it can be seen that none of the traditional visemes exhibit this high intra-viseme confoundedness. The high level of correct phoneme matches and low level of intra-viseme substitutions indicate the individual phonemes are not visually indistinguishable.

This shows that the traditional viseme groupings do not display the characteristics required of visemes according to the hypothesis in Section 2.1. While this does not disprove visemes, it requires that other groupings must be examined to determine if the hypothesis can be supported. The remainder of this chapter tests alternate groupings of phonemes into visemes, to determine if any of them are better able to satisfy the characteristics required by the hypothesis in Section 2.1.

6.2 Systematically Breaking Traditional Visemes Into More Suitable Groups

To investigate the traditional viseme groups, a confusion matrix (see Table 6.2) is used to identify what each phoneme is recognised as, and which are the most and least confounded. Using this data, the existing viseme groups can be investigated to determine how these traditional groups can be improved.

The method used to break down the traditional visemes into more suitable groups is based on the characteristics of the errors of the members of each viseme group. The intra-viseme noise is defined as the number of errors between phonemes within a viseme group, divided by the total number of

times that viseme is correctly identified. The false positive inter-viseme noise is defined as the number of times a viseme was incorrectly recognised, divided by the total number of times that viseme was correctly identified, as per Table 6.3.

The main property of a viseme is that the phonemes belonging to the viseme group are relatively indistinguishable from each other, while they are distinguishable from the other viseme groups. This would be indicated by a high ratio of intra-viseme noise to inter-viseme noise (characteristic C1). The individual phonemes would be characterised by high confoundedness between other phonemes within its viseme group, and low confoundedness between phonemes from other viseme groups (characteristic C1 and C2).

The process used to improve the viseme groups uses these characteristics to identify visemes that are not suitable, and phonemes that should not be placed within a particular viseme. First, the groups with no intra-viseme noise will be split into individual phonemes. As the main characteristic of a viseme is high intra-viseme noise and low inter-viseme noise, those groups with no intra-viseme noise are clearly not visemes, as they do not satisfy any of the characteristics required by the hypothesis. The visemes will be examined starting with the quietest (in terms of intra-viseme noise) first, and once the obvious visemes have been split, the noisiest will be examined. The intra-viseme noise, inter-viseme false-positive noise, and the number of member phonemes in each traditional viseme group can be seen in Table 6.4.

The viseme as a whole, as well as the individual phonemes within each viseme, will be examined to determine whether they exhibit the characteristics expected of a viseme. Those not exhibiting these expected characteristics will be broken out of the existing groups.

To indicate intermediate visemes created after some phonemes have been broken out, the viseme will be referred to by its name followed by a star, then a number in parenthesis, which indicates the number of members it

Table 6.4: Noise levels and number of member phonemes in each traditional viseme group

Viseme	Members	Intra-viseme noise	Inter-viseme (FP) noise
SB	6	23.1%	57.1%
FV	5	12.2%	53.7%
RV	7	11.2%	57.8%
R	2	7.4%	57.7%
OV	2	5.6%	53.0%
AlCl	3	5.4%	53.4%
BV	2	2.7%	54.7%
LB	2	2.1%	38.5%
L	1	0%	58.1%
Y	1	0%	43.7%
LCl	1	0%	47.8%
Pal	4	0%	4.6%
LFr	2	0%	32.0%
VlCl	1	0%	0%

now contains. For example, the "SB" viseme has 6 members, so if one were broken out, the resulting group would be referred to as "SB*(5)".

When looking at the individual phonemes, false positives are much more significant than false negatives. When looking at the recognition output, it is much more useful to know which phonemes are more trustworthy, and which are less trustworthy, as this allows for a word fitting algorithm to make better informed decisions when constructing sentences. As such, if a phoneme has no or very few false positives within the viseme group, it should be broken out to be treated as an individual phoneme. If a phoneme has a large number of false positives from all other phonemes outside its viseme, it does not belong to any one viseme, so should also be broken out as an individual phoneme.

For example, the confusion matrix (see Table 6.2 on page 140) shows that the /dh/ phoneme is much more likely to be a false positive than be correct. When the /dh/ phoneme is spotted in the recognition output, the recognition system should know not to trust it, and that it may have been an insertion, or that any phoneme may have been substituted. In this case, /dh/ does not

belong to a particular viseme, and should instead be treated as a "rogue". In contrast, if the /uh/ phoneme appears in the output, it should be strongly trusted, as it does not have any false positives. This would indicate that it should be treated as an individual phoneme, and not the member of a viseme, as it is not confused for any phoneme.

6.2.1 Pal Viseme

The Pal(4) viseme is comprised of the /ch/, /jh/, /sh/, and /zh/ phonemes. As can be seen in the confusion matrix excerpt below (see Table 6.5), there is no intra-viseme noise. It is clear that this group of phonemes is not a viseme, as all of the phonemes can be fully distinguished from each other. The Pal(4) viseme does not satisfy any of the required characteristics. As a result, this viseme group is to be broken into its individual phonemes.

Table 6.5: Confusion matrix for the Pal(4) viseme (excerpt of Table 6.2)

	ch	Jh	sh	Zh
ch	6	0	0	0
jh	0	10	0	0
sh	0	0	21	0
zh	0	0	0	5

6.2.2 LFr Viseme

The LFr(2) viseme is comprised of the /f/ and /v/ phonemes. Like the Pal(4) viseme, this viseme also has no intra-viseme noise (see Table 6.6). It is clear that each phoneme in this group can be fully distinguished from each other, so are also broken into its individual phonemes, as it does not satisfy any of the required characteristics.

Table 6.6: Confusion matrix for the LFr(2) viseme (excerpt of Table 6.2)

	f	v
f	22	0
v	0	29

6.2.3 LB Viseme

The next quietest viseme group is the LB(2) viseme, with an intra-viseme noise level of 2.1%. It can be seen in Table 6.7 that there is only a single error between phonemes within the LB group. If this is compared to the number of times the /b/ phoneme was mistakenly recognised in place of other phonemes (see Table 6.2 on page 140), this viseme does not demonstrate any of the characteristics required by the hypothesis. It is clear that the /b/ and /p/ phonemes do not belong to a viseme, and should be broken up into its individual phonemes.

Table 6.7: Confusion matrix for the LB(2) viseme (excerpt of Table 6.2)

	b	p
b	25	0
p	1	22

6.2.4 BV Viseme

The BV(2) viseme is comprised of the /aa/ and /ah/ phonemes, and has an intra-viseme noise level of 2.7%. When this is compared to the inter-viseme noise level of 54.7% (from Table 6.4 on page 145), it is clear that this group does not satisfy the viseme characteristic of high intra-viseme noise and low inter-viseme noise (characteristic C1), or displaying significant confusion between all member phonemes (characteristic C2), and should therefore be broken up into individual phonemes.

Table 6.8: Confusion matrix for the BV(2) viseme (excerpt of Table 6.2)

	aa	ah
aa	41	6
ah	3	284

6.2.5 SB Viseme

Now that the quietest visemes have been split, it is time to look at the noisiest viseme groups. The SB(6) viseme is the noisiest with an intra-viseme level of 23.1%, and is comprised of 6 phonemes (see Table 6.9). It is immediately clear from the confusion matrix that the /dh/ phoneme is responsible for the majority of this noise. When looking at the /dh/ phoneme in the full confusion matrix (Table 6.2 on page 140), it is clear that these /dh/ false positives are not restricted to the SB(6) viseme. This fails characteristic C1, and as such there is no reason for the /dh/ phoneme to belong in this viseme.

Table 6.9: Confusion matrix for the SB(6) viseme (excerpt of Table 6.2)

	d	dh	g	k	t	th
d	63	18	0	5	10	0
dh	1	84	0	2	5	0
g	0	16	19	4	7	0
k	1	19	0	87	6	0
t	4	26	0	8	191	0
th	0	2	0	1	0	5

When looking at the remaining phonemes, it can be seen that the /th/ and /g/ phonemes have no false positives within the SB(6) viseme group. As these fail characteristic C2, there is no reason for these phonemes to be part of the viseme. Once these three phonemes are removed, the resulting SB*(3) viseme (see Table 6.10) has an intra-viseme noise level of 9.0%.

Table 6.10: Confusion matrix for the SB*(3) viseme (excerpt of Table 6.2)

	d	k	t
d	63	5	10
k	1	87	6
t	4	8	191

6.2.6 RV Viseme

The RV(7) viseme has a intra-viseme noise level of 11.2%. From the confusion matrix (see Table 6.11), it is clear that the /aw/, /oy/, /uh/, and /uw/ phonemes have no false positives within the RV(7) viseme group, failing to satisfy characteristic C2. As such, there is no justification for these phonemes to be grouped within this viseme. This leaves an RV*(3) viseme containing the /ao/, /ow/, and /w/ phonemes (see Table 6.12), which has an intra-viseme noise level of 5.9%.

Table 6.11: Confusion matrix for the RV(7) viseme (excerpt of Table 6.2)

	ao	aw	ow	oy	uh	uw	w
ao	52	0	3	0	0	0	3
aw	1	12	1	0	0	0	4
ow	1	0	46	0	0	0	0
oy	1	0	0	12	0	0	0
uh	1	0	0	0	11	0	3
uw	0	0	4	0	0	19	3
w	2	0	2	0	0	0	77

Table 6.12: Confusion matrix for the RV*(3) viseme (excerpt of Table 6.2)

	ao	ow	w
ao	52	3	3
ow	1	46	0
w	2	2	77

6.2.7 Resulting Viseme Groups and Phonemes

After these visemes have been split, there are twenty one individual phonemes, two visemes with two members, three visemes with three members, and a single viseme with five members, resulting in twenty seven recognisable phoneme groups. The resulting confusion matrix can be seen in Table 6.13.

By systematically splitting phonemes from the traditional visemes when they did not display the required characteristics of a viseme, the correctness metric (Equation 5.2 on page 134) has dropped from 29.7% for traditional visemes, to 28.1%. Although a more meaningful grouping has been achieved, the accuracy metric has decreased. This drop is due to the nature of visemes to hide errors, and with smaller viseme groups, fewer errors are hidden.

As can be seen in Table 6.14, the maximum intra-viseme noise level is now only 12.2%, down from 23.1%. The remaining viseme groups still have a low ratio of intra-viseme substitutions to inter-viseme substitutions, which does not support the viseme concept according to characteristic C1.

6.3 Grouping The Noisiest Phonemes Together

To highlight how the phoneme to viseme mapping simply hides errors, a set of visemes can be constructed by taking the noisiest phonemes and placing them in the largest viseme, with the smallest visemes containing the quietest phonemes. By using the same size and number of visemes as the traditional groupings (Table 6.1 on page 138), a fair comparison can be made of how just the grouping affects the measurement of accuracy at the viseme level.

Table 6.15 shows this grouping as arranged by noisiest to quietest phonemes. There is one viseme containing the seven noisiest phonemes, one containing the next six phonemes, one with five, one with four, one with three, five with

Table 6.13: Rearranged confusion matrix (based on Table 6.2) with the new split up viseme groups on the left, and the separated phonemes on the right

		OV		BV		FV					RV*(3)			R		AlCl		SB*(3)			dh	s	L	Y	LCl	b	p	aw	oy	uh	uw	g	th	f	v	VlCl	ch	jh	sh	zh	
		ih	iy	aa	ah	ae	ay	eh	ey	hh	ao	ow	w	er	r	n	z	d	k	t	dh	s	l	y	m	b	p	aw	oy	uh	uw	g	th	f	v	ng	ch	jh	sh	zh	(del)
OV	ih	182	11	7	24	9	6	7		1	1	7	10	2	9	11	6	5	5	9	36	25	8	1	3			1														205
	iy	12	207	4	26	2	5	4	4	3	5	10	20	1	6	10	7	4	4	13	35	21	11		3	2															175
BV	aa	5	11	41	6	4	1	1				2	2	4	3	5		2	4	2	14	10	6		1	1	1								1						80
	ah	17	19	3	284	8	7	13		2	5	7	11	3	25	7	5	4	4	7	35	14	11	1	3									1						205
FV	ae	11	8	1	20	82	3	1	2	5	1	4	4		18	11	2	2	5	4	23	7	7		1	1															143
	ay	5	6		7	1	51	2		1			4		2	4	2		1	2	11	10	4												1						55
	eh	11	7	2	5	3	2	52			1	1	6	2	11	6	4	2	1	8	18	9	6	1	2	3								1	1						95
	ey	1	3	1	9	3	2	5	32		1	1	4	1	5	4		2	5	2	7	6	2		1							2									77
	hh	4	1	1	3	3		1		27		3	3		6	1	1	1	1	3	5	4	6		1			1													73
RV*(3)	ao	10	4		14	2		3	2		52	3	3	3	7	8			3	4	15	10	6		1	2															61
	ow	7	3	1	10	3		1			1	46			2	3	2	1	3	2	3	2	6				1														55
	w	3	3	2	11	2	1	4		1	2	2	77	1	8	5	3		3	2	10	6	8		3	1						1									68
R	er	16	12	1	18	5	3	3	1	3	4	4	10	47	17	10	8	3	8	5	26	15	3		3															118
	r	14	6	2	21	5	2	7	1	5	2	10	14	2	191	11	4	3	7	11	29	21	18	2	4									1	1					142
AlCl	n	10	14	5	21	2	4	3		3	3	8	9	5	19	217	5	4	6	5	25	18	8	3	6	3									1						198
	z	2	9	2	12	5	3	3		1	1		4	1	10	12	79	1	5	5	15	11	4		2																95
SB*(3)	d	9	7	4	15	4	4	8	1	4	4	8	4		9	11	12	63	5	10	18	12	9		4	3															177
	k	6	18	1	12	5	3	3		4	4	6	13	3	11	18	6	1	87	6	19	18	8	1	4	1	1								2						158
	t	11	14	3	26	4	3	6		5	4	4	10	1	18	11	21	4	8	191	26	13	18	2	3	1	2								2			1			204
dh	dh	8	1		7	3	1	1	2	1	1		5	1	6	6	3	1	2	5	84	5	5		1	1															69
s	s	14	10		11	4		7	1	1	1	5	6	6	28	6	9	7	4	10	22	185	8	3	1															159
L	l	8	16	3	22	5	4	6	1	3	3	6	10	3	16	24	9	4	7	10	26	14	152		4															150
Y	y	6	4	1	4		2			2		2	1	2	7	6	1	2	2	7	10	2	3	27	1	2														63
LCl	m	5	6	2	13	4	2	4	2		1	3	4		17	6	3	1	6	7	19	9	5		70		1								1						126
b	b	1	5		7	3	1	2			3	3	3	1	11	7			2	2	11	6	5		1	25															84
p	p	8	8		4	1	2	2	1	1	2	6	8	2	4	2	3	1	2	6	9	8	9		1	1	22							1					1		108
aw	aw	1	2	1	1	1	1		1		1	1	4		5		1				2							12													15
oy	oy	3	1		2						1			1		1			2	1	1	1	1		1				12												29
uh	uh	5	3		1						1		3	2	2	1	3			1	1	5	1							11											25
uw	uw	6	8		9	3	1	2	1			4	3	1	4	5	1	2	2	4	10	5	4	2	2					19											69
g	g	5	6	3		5	1	2			1	3	8	2	2	2	3		4	7	16	12	3		1	1				1	19									60
th	th		3		3			2				1	1		3	1	1		1		2	2	2		2							5									23
f	f	7	7	2	4	2	1	1				3	1		9	2	1	1	3	6	14	5	7	3	1			2				1		22							74
v	v	1	1		4	5	1	4			2	2	1		5	2	2		1	3	12	5		1	1	1									29						61
VlCl	ng	2	1		4		1		1		1	1	2		2		2		2	2	3	6	1	1												13					31
ch	ch	2	1		2	1		1		1					3	1		2		4	2	1	2														6				18
jh	jh	1	2				1			1		2	1		2		1		1	2	2	1	2															10			17
sh	sh	7	3		2	1	1	1				2	2		5	3	3	2	1	4	10	5	4		2			1											21		69
zh	zh				1			1					1		2	1				1	3					1														5	1
(ins)		61	54	13	76	24	26	30	7	7	5	24	45	18	46	45	26	14	25	36	110	51	58	4	12	2	2					1		3	7				2		

Table 6.14: Noise levels and number of member phonemes for the resulting visemes, after splitting the traditional groups

Viseme	Original Members	Members	Intra-viseme noise	Inter-viseme (FP) noise
FV	5	5	12.2%	53.7%
SB*	6	3	9.0%	46.8%
R*	2	2	7.4%	57.7%
RV*	7	3	5.9%	66.2%
OV	2	2	5.6%	53.0%
AlCl	3	3	5.4%	52.1%

two, and finally four visemes each containing a single quiet phoneme. By grouping the noisiest phonemes into the largest visemes, this should provide an upper bound on the correctness and accuracy metrics for groupings of this size and number.

Table 6.15: Viseme groups with phonemes sorted from noisiest to cleanest

Viseme number	Phoneme
1	/dh, ow, w, eh, h, s, z/
2	/r, ay, l, ih, k, ae/
3	/aa, ah, er, iy, n/
4	/d, ao, t, m/
5	/b, y, ey/
6	/v, aw/
7	/p, f/
8	/g, zh/
9	/jh, sh/
10	/uw, oy/
11	/uh/
12	/ch/
13	/th/
14	/ng/

The original phoneme confusion matrix (Table 5.2 on page 135) can be reordered using this grouping, to group the noisiest phonemes towards the top

left of the matrix, and the cleanest towards the bottom right (see Table 6.16). Using this nonsense grouping, an apparent increase in viseme accuracy is achieved due to the reclassification of errors as being correct.

From this rearranged confusion matrix, it can be seen that /oy/, /uh/, /th/, /ng/, and /ch/ are the least confounded phonemes, with no false positives. At the other extreme, the /dh/ phoneme has 655 false positives, and only 84 correctly recognised samples, giving a false positive rate of 88.6%.

It is clear that the phonemes on the left hand side of the matrix will result in similar confoundedness, irrespective of the grouping. This is simply due to the likelihood of each phoneme to be erroneously recognised for almost any other phoneme in this region. By grouping the phonemes with the highest false positive percentage into the largest viseme, the number of substitution errors that can be hidden is maximised. As a result, the overall correctness (Equation 5.2 on page 134) of this grouping is 32.8%, compared to 29.7% for the traditional grouping. As this is for the exact same phoneme output, it demonstrates how the mapping of phonemes to visemes hides substitution errors to artificially improve measurements of recognition accuracy.

Figure 6.2 shows the likelihood of each operation to cause a viseme to be output by the recogniser. From this figure it can be seen that the ratio of intra-viseme substitutions to inter-viseme substitutions is still very low, in contradiction to characteristic C1. It is also clear that the larger visemes (the lower viseme numbers in the figure) do not reach the 70% clustering threshold when combining the correct phoneme matches with the intra-viseme substitutions.

While visemes 7 to 14 do reach the 70% clustering threshold, they achieve this with correct phoneme matches alone, not requiring any of the intra-viseme substitutions to reach this threshold. Visemes 7 to 10 contain only two phonemes, and visemes 11 to 14 are single phonemes. This makes it clear that the only reason they reach the 70% threshold is because of individual

Table 6.16: Rearranged confusion matrix (based on Table 6.2), using noisiness-based viseme groups from Table 6.15

	dh	ow	w	eh	hh	s	z	r	ay	l	ih	k	ae	aa	ah	er	iy	n	d	ao	t	m	b	y	ey	v	aw	p	f	g	zh	jh	sh	uw	oy	uh	ch	th	ng	(del)
dh	84		5	1	1	5	3	6	1	5	8	2	3		7	1	1	6	1	1	5	1	1		2																69
ow	3	46		1		2	2	2		6	7	3	3	1	10		3	3	1	1	2							1													55
w	10	2	77	4	1	6	3	8	1	8	3	3	2	2	11	1	3	5		2	2	3	1							1										68
eh	18	1	6	52		9	4	11	2	6	11	1	3	2	5	2	7	6	2	1	8	2	3	1		1			1										95
hh	5	3	3	1	27	4	1	6		6	4	1	3	1	3		1	1	1		3	1					1												73
s	22	5	6	7	1	185	9	28		8	14	4	4		11	6	10	6	7	1	10	1		3	1														159
z	15		4	3	1	11	79	10	3	4	2	5	5	2	12	1	9	12	1	1	5	2																	95
r	29	10	14	7	5	21	4	191	2	18	14	7	5	2	21	2	6	11	3	2	11	4		2	1	1			1										142
ay	11		4	2	1	10	2	2	51	4	5	1	1		7		6	4			2					1													55
l	26	6	10	6	3	14	9	16	4	152	8	7	5	3	22	3	16	24	4	3	10	4			1														150
ih	36	7	10	7	1	25	6	9	6	8	182	5	9	7	24	2	11	11	5	1	9	3		1			1												205
k	19	6	13	3	4	18	6	11	3	8	6	87	5	1	12	3	18	18	1	4	6	4	1	1		2		1											158
ae	23	4	4	1	5	7	2	18	3	7	11	5	82	1	20		8	11	2	1	4	1	1		2														143
aa	14	2	2	1		10		3	1	6	5	4	4	41	6	4	11	5	2		2	1	1			1		1											80
ah	35	7	11	13	2	14	5	25	7	11	17	4	8	3	284	3	19	7	4	5	7	3		1					1										205
er	26	4	10	3	3	15	8	17	3	3	16	8	5	1	18	47	12	10	3	4	5	3			1														118
iy	35	10	20	4	3	21	7	6	5	11	12	4	2	4	26	1	207	10	4	5	13	3	2		4														175
n	25	8	9	3	3	18	5	19	4	8	10	6	2	5	21	5	14	217	4	3	5	6	3	3		1													198
d	18	8	4	8	4	12	12	9	4	9	9	5	4	4	15		7	11	63	4	10	4	3		1														177
ao	15	3	3	3		10		7		6	10	3	2		14	3	4	8		52	4	1	2		2														61
t	26	4	10	6	5	13	21	18	3	18	11	8	4	3	26	1	14	11	4	4	191	3	1	2		2		2				1							204
m	19	3	4	4		9	3	17	2	5	5	6	4	2	13		6	6	1	1	7	70			2	1		1											126
b	11	3	3	2		6		11	1	5	1	2	3		7	1	5	7		3	2	1	25																84
y	10	2	1		2	2	1	7	2	3	6	2		1	4	2	4	6	2		7	1	2	27															63
ey	7	1	4	5		6		5	2	2	1	5	3	1	9	1	3	4	2	1	2	1			32					2									77
v	12	2	1	4		5	2	5	1		1	1	5		4		1	2		2	3	1	1	1		29													61
aw	2	1	4				1	5	1		1		1	1	1		2			1					1		12													15
p	9	6	8	2	1	8	3	4	2	9	8	2	1		4	2	8	2	1	2	6	1	1		1			22	1		1								108
f	14	3	1	1		5	1	9	1	7	7	3	2	2	4		7	2	1		6	1		3			2		22	1									74
g	16	3	8	2		12	3	2	1	3	5	4	5	3		2	6	2		1	7	1	1							19				1					60
zh	3		1	1				2							1			1			1		1								5								1
jh	2	2	1		1	1	1	2	1	2	1	1					2				2											10							17
sh	10	2	2	1		5	3	5	1	4	7	1	1		2		3	3	2		4	2					1						21						69
uw	10	4	3	2		5	1	4	1	4	6	2	3		9	1	8	5	2		4	2		2	1									19					69
oy	1					1				1	3	2			2	1	1	1		1	1	1													12					29
uh	1		3			5	3	2		1	5				1	2	3	1		1	1															11				25
ch	2			1	1	1		3		2	2		1		2		1	1	2		4																6			18
th	2	1	1	2		2	1	3		2		1			3		3	1				2															5		23
ng	3	1	2			6	2	2	1	1	2	2			4		1			1	2			1	1													13	31
(ins)	110	24	45	30	7	51	26	46	26	58	61	25	24	13	76	18	54	45	14	5	36	12	2	4	7	7		2	3	1		2							

phonemes correctly being recognised. This is in strong contradiction to characteristics C1 and C2.

When compared to the results for traditional visemes (see Figure 6.1 on page 142), it can be seen that for both viseme groupings, the only visemes to reach the 70% clustering threshold achieved this with correct phoneme matches, without requiring the intra-viseme substitutions. The results for both groupings were very similar, with the likelihood of intra-viseme substitutions being lower than inter-viseme substitutions, in contradiction to characteristic C1. The intra-viseme substitutions were also significantly lower than the number of correct phoneme matches for both viseme groupings, in contradiction to characteristic C2.

As discussed previously, this grouping is designed to reach the upper bound for correctness and accuracy for groupings using the same size and number of visemes as the traditional groupings. It was designed to reclassify as many substitutions as being correct as possible. Even still, it was not able to meet the clustering threshold of 70% as required by characteristic C1, nor was it able to produce visemes with significant substitutions between all member phonemes as required by characteristic C2.

It is evident that this grouping does not support the hypothesis in Section 2.1. The following section will investigate other groupings from the literature to determine if any can be found to support the hypothesis.

6.4 Viseme Groups From The Literature

When using the viseme groups identified by other researchers (see Table 1.2 on page 18), similar recogniser performance is obtained to the traditional grouping. Table 6.17 summarises the performance of the various viseme groupings.

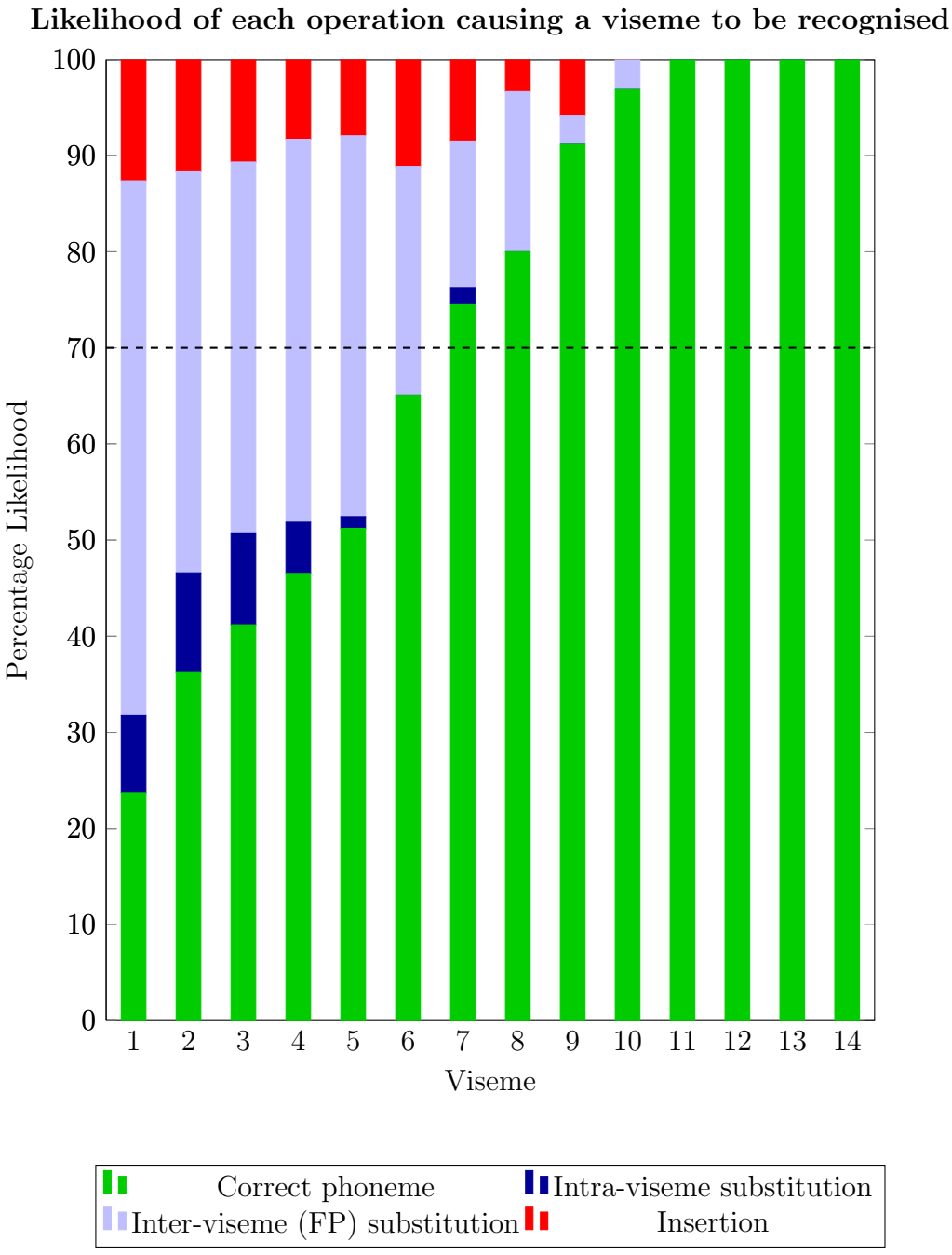


Figure 6.2: The likelihood of each operation causing a phoneme to be recognised, and cause a viseme to appear in the recognition output (uses the noise-based viseme groups from Table 6.15)

Table 6.17: Summary of performance for various viseme groupings

Grouping Method	Number Of Visemes	Responses	Correct Phonemes	Intra-viseme Substitutions	Inter-viseme Substitutions	Viseme Correctness
Phonemes only	39	7538	2757	0	3947	26.7%
Traditional visemes (Section 6.1)	14	7538	2757	311	3636	29.7%
Traditional visemes after systematic splitting (Section 6.2)	27	7538	2757	147	3800	28.1%
Noisiest phonemes in the largest visemes (Section 6.3)	14	7538	2757	632	3315	32.8%
MPEG-4 Facial Animation groups	14	7538	2757	213	3734	28.7%
(Potamianos et al., 2004)	12	7538	2757	279	3668	29.4%
(Hazen et al., 2004)	13	7538	2757	314	3633	29.7%
(Lucey, Martin, and Sridharan, 2004)	14	7538	2757	414	3533	30.7%

The accuracy of the various groupings are all within 10%, with individual phonemes achieving 26.7% correctness, and grouping the noisiest phonemes into the largest visemes achieving 32.8%, with the other groupings typically around 29-30%.

The grouping with the highest correctness was the one in which the nature of the sounds and mouth shapes were completely ignored, and simply grouped the noisiest phonemes together into the largest viseme groups. This demonstrates the way in which mapping phonemes to visemes arbitrarily hides substitution errors, by relabelling them as being correct. These apparent increases in correctness are due to this hiding of substitution errors, not a true increase in performance.

This is demonstrated by the accuracy of the systematically split up visemes (see Section 6.2.7) which have a correctness of 28.1%, lower than the traditional groups themselves. These split up groups were created by methodically analysing each viseme group and determining which phonemes should be split out of that viseme, forming more suitable groups. This shows that the viseme correctness is not a useful measurement, in itself, as it is simply hiding substitution errors. The accuracy measurement encourages larger and fewer viseme groups, as this mislabels more substitution errors as being a correct match.

The various viseme groupings in Table 6.17 all achieved similar results. It is clear that none of the groupings reached the 70% threshold required by characteristic C1. For the hypothesis to be true, a grouping must be found to satisfy characteristic C1. The grouping of the noisiest phonemes into the largest visemes (see Section 6.3) should provide an upper bound on the intra-viseme substitutions. While this grouping displayed the highest "viseme correctness", it still fell well short of the 70% threshold, reaching only 45%.

All the existing groupings also fail to satisfy characteristic C2, which requires significantly higher intra-viseme substitutions than inter-viseme substitutions. All the groupings in Table 6.17 have at least 5 times as many inter-viseme

substitutions than intra-viseme substitutions, with many of the groupings exhibiting more than 10 times the intra-viseme substitutions. This clearly demonstrates that none of these groups satisfy characteristic C2.

In this chapter, it has been shown that all of the existing groupings fail to satisfy the hypothesis of visemes being the basic visual unit of speech. This leaves the hypothesis, of visemes being the basic visual unit of speech, in serious doubt.

In Chapter 7, the underlying phoneme confusion will be examined to determine why the existing groupings failed to satisfy the hypothesis, and if it is at all possible to construct a set of viseme groupings that will prove the hypothesis.

Chapter 7

Analysis Of Confusion

As discussed in the previous chapter, no existing set of visemes has exhibited all of the characteristics C1–C3 required by the hypothesis. The nature of the phoneme confusion needs to be analysed to determine whether any of the characteristics of visemes are present.

In this chapter, three aspects of the confusion are investigated: the phoneme trustworthiness; the degree of confoundedness between pairs of phonemes; and finally the directionality of the confusion between phonemes.

7.1 Phoneme Trustworthiness

The trustworthiness of a phoneme is the likelihood of that phoneme appearing in the output due to being correctly recognised, as opposed to it appearing due to an insertion or substitution error. Trustworthiness does not take into account how likely it is for a deletion error to occur preventing that phoneme appearing in the output stream, as the metric is only a measure of how much trust can be put in a phoneme that does appear in the output stream. By calculating the trustworthiness of each phoneme, the required viseme characteristics can be examined.

Phoneme trustworthiness is useful to word fitting algorithms. By outputting phonemes directly, a word fitting algorithm can use the phoneme trustworthiness when deciding which sequence of words best fit a phoneme sequence. If a phoneme is never a false positive, then its appearance in the output stream indicates a phoneme that must be correctly placed into a word. On the other hand, if a phoneme is extremely untrustworthy, it can be allowed to be substituted or deleted when trying to fit words to the phoneme sequence.

Characteristic C1 requires 70% of samples for a viseme to be recognised within the correct viseme. This would require medium to high trustworthiness phonemes to ensure 70% of recognised phonemes occur within the correct viseme group. Characteristic C2 requires all member phonemes to have high confoundedness for all other member phonemes. This would require phonemes with low to medium trustworthiness, to ensure sufficient intra-viseme substitutions occur.

Phonemes with high trustworthiness will help satisfy C1, as they will ensure the phonemes are correctly recognised within the parent viseme. On the other hand, high trustworthiness phonemes work against C2, as the high number of correct phoneme matches results in few intra-viseme substitutions. Trustworthy phonemes do not demonstrate the characteristics required to be grouped into a viseme, as they are recognisable in their own right. This would result in highly trustworthy phonemes being left as individual phonemes.

Phonemes with low trustworthiness do not help satisfy C1, as they result in substitutions for many other phonemes, reducing the number of responses within the parent viseme. Conversely, they help satisfy C2, as these substitutions result in significant intra-viseme substitutions.

For visemes to be valid, the trustworthiness of each phoneme must be balanced between being too trustworthy, and too untrustworthy. The phonemes need to demonstrate a high trustworthiness outside the viseme, but a low to medium trustworthiness within the viseme group. This would indicate phonemes that

are easily confused within a cluster of phonemes (i.e. a viseme), but not confused outside that cluster.

Figure 7.1 (based on the data from Section 5.5) shows a summary of the phoneme trustworthiness, indicating the likelihood of each operation (i.e. due to a correctly recognised phoneme, a substitution, or an insertion) causing a phoneme to appear in the output. In this figure, the two most likely substitutions for each phoneme have been extracted and shown separately from all other substitutions.

As can be seen in Figure 7.1, the likelihood of a phoneme that appears in the output being correct varies significantly between phonemes, with five phonemes having 100% likelihood of being correct, and others as low as 11.3%. The median likelihood of being correct is 45.6%. It is clear that the most likely scenario to cause a phoneme to appear is due to being a correct match, with only /dh/ being less likely to be correct than to be an insertion error.

The likelihood of being a substitution ranges from 0% to 73.7%, with a median of 44.6%. When considering any specific substitution, the likelihood of this ranges from 0% to 16.7%, with a median of 0.2%.

When the 70% threshold of characteristic C1, is combined with the requirement of characteristic C2 for all phonemes within a viseme to be significantly confounded, it is difficult to justify a set of visemes using the data shown in Figure 7.1.

To satisfy the requirement, as per characteristic C2, for all phonemes within a viseme to be significantly confounded, a phoneme would have to have a significant proportion of substitutions for at least one or two other phonemes. As can be seen in Figure 7.1, most of the substitutions for each phoneme have a very low likelihood of occurring. It should be remembered that each of the individual substitutions grouped into the light blue region must, by definition, be smaller than the first and second most likely substitutions (indicated in darker blue, labelled “Substitution 1” and “Substitution 2” respectively).

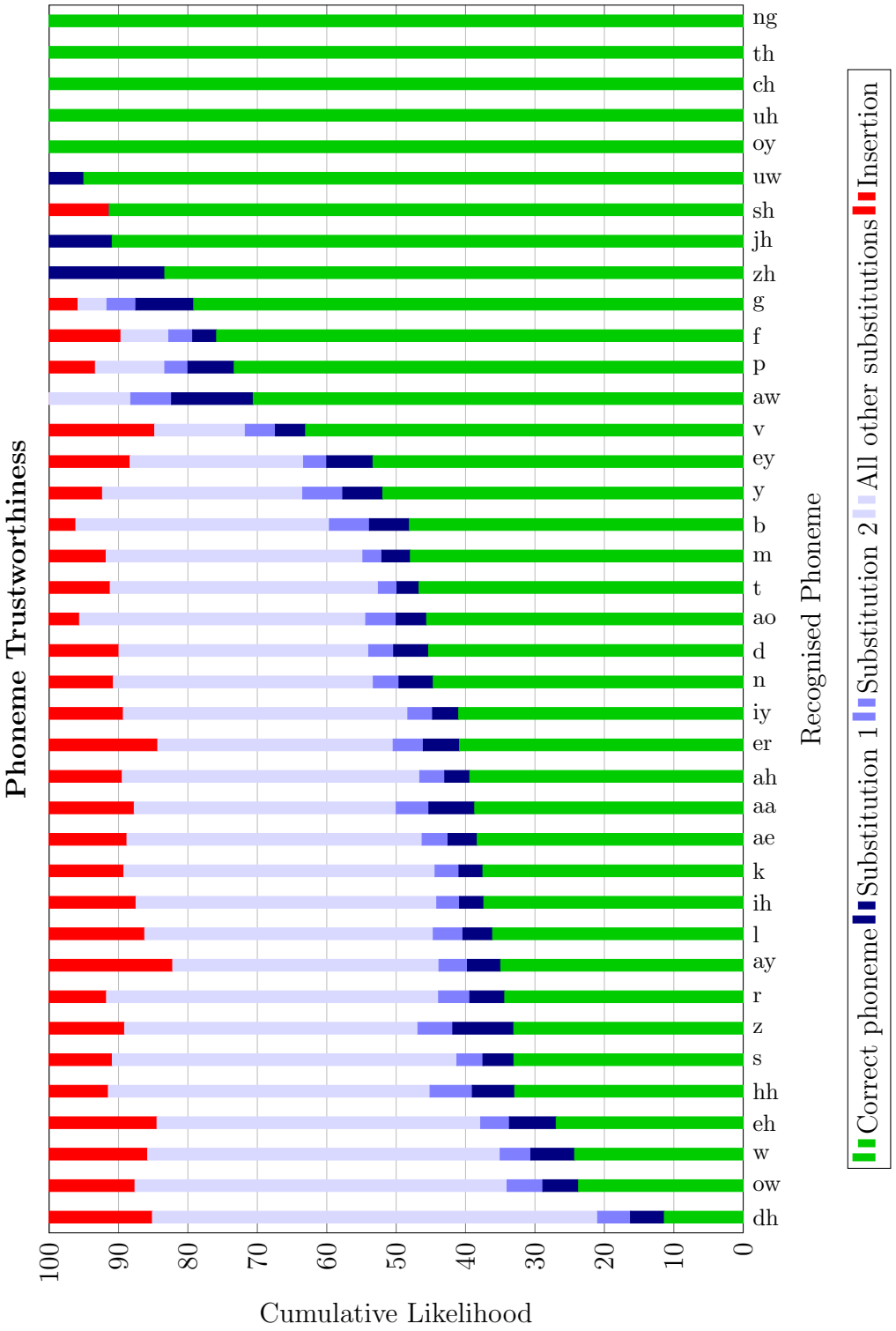


Figure 7.1: Phoneme trustworthiness – the likelihood and operation causing each phoneme to appear in the output stream

If the two most likely substitutions for each phoneme are examined, it can be seen that the combined likelihood of either occurring is generally below 10%, with a median likelihood of 8.7%. As the highest likelihood for any second-most-likely substitution is 6.1%, all remaining substitutions must be less likely than this. When this is compared to the median likelihood of being correct at 45.6%, the existence of a set of viseme candidates is difficult to justify.

As most of the phonemes have reasonably high percentage of correct matches, the only possible candidates for grouping into visemes must have a low percentage of being correct to have any chance of achieving significant clustering. As illustrated in Figure 7.1, the phonemes with low likelihood of correctness also have low likelihoods for each individual substitution. It is clear that these phonemes would not be able to satisfy the 70% clustering threshold using just a handful of substitutions.

Table 7.1 shows the three most likely causes for each phoneme to appear, and the percentage likelihood for each to have caused a phoneme to appear in the recognition output. The full table of trustworthiness data is attached as Appendix C.

As can be seen in Table 7.1, for all but one phoneme, the most likely cause of the phoneme appearing in the recognition output is it appearing in the input stream. The /dh/ phoneme is different from the other phonemes in that it is 30% more likely to occur due to an insertion, than from being a correct match. As per Section 6.2.5, this indicates that /dh/ is a “rogue” phoneme, as it is an extremely common substitution for many phonemes, while it is not often substituted by other phonemes.

The second most likely cause for a phoneme to appear is due to an insertion. This indicates the phoneme recogniser has a significant number of insertions. As discussed in Section 5.4.1, this is due to having to balance between insertion, deletion, and substitution errors.

Table 7.1: Trustworthiness of each phoneme – the three most likely causes of each phoneme appearing in the recognition output, and the likelihood of it occurring

Recognised Phoneme	1st Choice	(%)	2nd Choice	(%)	3rd Choice	(%)
dh	(ins)	14.88	dh	11.37	ih	4.87
ow	ow	23.71	(ins)	12.37	iy	5.15
w	w	24.29	(ins)	14.20	iy	6.31
eh	eh	26.94	(ins)	15.54	ah	6.74
hh	hh	32.93	(ins)	8.54	ae	6.10
s	s	33.04	(ins)	9.11	ih	4.46
z	z	33.05	(ins)	10.88	t	8.79
r	r	34.35	(ins)	8.27	s	5.04
ay	ay	34.93	(ins)	17.81	ah	4.79
l	l	36.10	(ins)	13.78	r	4.28
ih	ih	37.37	(ins)	12.53	ah	3.49
k	k	37.50	(ins)	10.78	er	3.45
ae	ae	38.32	(ins)	11.21	ih	4.21
aa	aa	38.68	(ins)	12.26	ih	6.60
ah	ah	39.39	(ins)	10.54	iy	3.61
er	er	40.87	(ins)	15.65	s	5.22
iy	iy	40.99	(ins)	10.69	ah	3.76
n	n	44.65	(ins)	9.26	l	4.94
d	d	45.32	(ins)	10.07	s	5.04
ao	ao	45.61	iy	4.39	ah	4.39
t	t	46.70	(ins)	8.80	iy	3.18
m	m	47.95	(ins)	8.22	n	4.11
b	b	48.08	eh	5.77	n	5.77
y	y	51.92	(ins)	7.69	n	5.77
ey	ey	53.33	(ins)	11.67	iy	6.67
v	v	63.04	(ins)	15.22	k	4.35
aw	aw	70.59	f	11.76	ih	5.88

Table 7.1: (Continued) Trustworthiness of each phoneme – the three most likely causes of each phoneme appearing in the recognition output, and the likelihood of it occurring

Recognised Phoneme	1st Choice	(%)	2nd Choice	(%)	3rd Choice	(%)
p	p	73.33	t	6.67	(ins)	6.67
f	f	75.86	(ins)	10.34	ah	3.45
g	g	79.17	ey	8.33	w	4.17
zh	zh	83.33	p	16.67		
jh	jh	90.91	t	9.09		
sh	sh	91.30	(ins)	8.70		
uw	uw	95.00	g	5.00		
oy	oy	100.00				
uh	uh	100.00				
ch	ch	100.00				
th	th	100.00				
ng	ng	100.00				

As can be seen in Table 7.1, there are five phonemes (/oy/, /uh/, /ch/, /th/, and /ng/) that have a trustworthiness of 100%. This means that if they appear in the output, they must have appeared in that location in the input. These phonemes never appear due to being substituted in place of another phoneme. According to characteristic C2 in Section 2.1, these phonemes cannot belong to any viseme, as no phoneme is ever confused for these phonemes.

If these trustworthy phonemes occur in the recogniser output, these phonemes must be correct when fitting words. These trustworthy phonemes appear 174 times in 145 words from the "cmudict" dictionary (see Section 5.3.1), and occur in 305 words from the VidTIMIT corpus. By using this knowledge, a

word fitting algorithm can ensure these trusted phonemes are handled with priority.

While the most likely scenario is that the phoneme is correctly recognised, with the likelihood for a particular substitution being considerably lower, there are still significant substitution errors that need to be examined in more depth. The remainder of this chapter examines the nature of this confusion between phonemes.

7.2 Measuring Confoundedness Of Phonemes

The clustering of phonemes needs to be based on how confounded the phonemes are. By looking at how often a phoneme is a false positive for each of the other phonemes, the most confounded phonemes can be identified to see if they can be clustered into a viseme.

The confoundedness of a phoneme pair is a measure of how likely it is for the two phonemes to be mistakenly recognised as one another. To normalise for the varying numbers of occurrence for each phoneme, each column in the raw confusion matrix (see Table 5.2 on page 135) is divided by the number of times that phoneme correctly appears in the recognition output, and multiplied by 50. Then, for each pair of phonemes, the two normalised values are added together (i.e. the value for A mapping to B, plus the value for B mapping to A), resulting in a score out of 100. The resultant triangular matrix of confoundedness is shown in Table 7.2, with highly confounded (top 25%) pairs highlighted, and pairs with zero confoundedness blanked for clarity. The 52 most confounded pairs of the 520 pairs of phonemes that were confounded (i.e. the top 10%), are listed in Table 7.3.

As Table 7.3 shows, only six of the top 10% most confounded phonemes pairs both belong to the same traditional viseme group (indicated in bold).

Table 7.2: Confoundedness between phonemes. The scores represent how often a phoneme is confounded with another phoneme

		ih	iy	aa	ah	ae	ay	eh	ey	hh	ao	ow	w	aw	oy	uh	uw	l	er	r	y	b	p	m	n	z	s	ch	jh	sh	zh	d	k	t	dh	g	th	f	v	ng
		OV	OV	BV	BV	FV	FV	FV	FV	FV	RV	RV	RV	RV	RV	RV	RV	L	R	R	Y	LB	LB	LCl	AlCl	AlCl	AlCl	Pal	Pal	Pal	Pal	SB	SB	SB	SB	SB	SB	LFr	LFr	VlCl
OV	ih																																						
OV	iy	6.0																																					
BV	aa	9.9	7.5																																				
BV	ah	8.9	9.2	4.7																																			
FV	ae	8.5	3.2	3.7	8.4																																		
FV	ay	7.3	6.4	1.0	8.1	3.6																																	
FV	eh	9.8	5.5	3.4	13.4	2.8	3.9																																
FV	ey	0.3	7.0	1.2	1.6	5.0	2.0	4.8																															
FV	hh	3.0	5.8	1.2	4.2	11.1	1.9	1.0																															
RV	ao	3.7	5.8		7.3	2.2		3.8	4.1																														
RV	ow	9.5	11.6	3.4	9.4	6.2		2.0	1.1	3.3	4.2																												
RV	w	7.3	13.7	3.7	9.1	3.8	3.6	7.7	2.6	3.8	3.9	2.2																											
RV	aw	4.4	0.5	1.2	0.2	0.6	1.0		1.6	4.2	1.0	1.1	2.6																										
RV	oy	0.8	0.2		0.4						1.0																												
RV	uh	1.4	0.7		0.2						1.0		1.9																										
RV	uw	1.6	1.9		1.6	1.8	1.0	1.9	1.6			4.3	1.9																										
L	l	4.8	7.5	5.6	7.5	5.4	5.2	7.7	2.2	7.5	4.9	8.5	9.1		0.3	0.3	1.3																						
R	er	6.5	4.0	5.5	6.4	3.0	2.9	5.0	2.6	5.6	7.0	4.3	7.6		1.1	2.1	1.1	4.2																				
R	r	6.2	3.0	3.2	10.2	7.8	2.5	9.6	2.9	10.8	3.8	11.4	11.2	1.3		0.5	1.0	10.1	6.6																			
Y	y	3.5	1.0	1.2	2.6		2.0	1.9		3.7		2.2	0.6				3.7	1.0	2.1	5.5																		
LB	b	0.3	5.2	2.0	1.2	3.8	1.0	7.9			6.9	3.3	3.9					1.6	1.1	2.9	4.0																	
LB	p	2.2	1.9	2.3	0.7	0.6	2.0	1.9	1.6	1.9	1.9	8.8	5.2					3.0	2.1	1.0		2.0																
LCl	m	3.5	3.6	3.2	4.4	3.2	2.0	5.3	3.8	0.7	1.7	3.3	4.7		0.7		1.4	4.5	2.1	7.3	0.7	0.7	3.0														
AlCl	n	5.3	5.7	7.2	5.3	3.8	4.8	4.3	0.9	5.8	4.7	9.4	7.0		0.2	0.2	1.2	8.2	7.6	7.5	6.9	7.6	0.5	5.7													
AlCl	z	4.3	6.6	2.4	5.3	4.3	4.2	5.4		2.5	1.0	1.3	4.5	0.6		1.9	0.6	7.0	6.1	5.1	0.6		1.9	3.3	5.9												
AlCl	s	10.6	8.1	2.7	5.7	4.3	2.7	9.2	3.2	2.9	3.7	6.0	5.5		0.3	1.4	1.4	6.4	10.4	13.0	6.1	1.6	2.2	3.1	6.2	8.7											
Pal	ch	0.5	0.2		0.4	0.6		1.0		1.9								0.7		0.8					0.2		0.3										
Pal	jh	0.3	0.5				1.0			1.9		2.2	0.6					0.7		0.5						0.6	0.3										
Pal	sh	1.9	0.7		0.4	0.6	1.0	1.0				2.2	1.3	4.2				1.3		1.3				1.4	0.7	1.9	1.4										
Pal	zh				0.2			1.0					0.6							0.5		2.0	10.0		0.2												
SB	d	6.4	4.9	6.5	5.8	4.0	3.9	9.3	3.1	8.2	3.8	9.5	2.6				1.6	6.1	2.4	4.7	1.6	6.0	0.8	3.7	5.7	8.4	8.8	1.6		1.6							
SB	k	4.5	6.6	3.5	4.4	5.9	3.5	3.5	2.9	8.0	5.6	8.2	10.2		1.1		1.1	6.7	7.8	6.9	3.0	3.1	3.4	6.3	7.6	6.7	7.2		0.6	0.6		3.7					
SB	t	5.4	6.8	4.2	6.4	3.5	3.5	7.9	0.5	10.0	4.9	4.9	7.0		0.3	0.3	1.0	8.5	2.4	7.6	5.5	2.5	6.1	4.0	3.8	14.6	6.1	1.0	5.5	1.0	0.3	5.8	6.2				
SB	dh	23.6	21.1	8.3	22.1	15.5	7.5	11.7	7.3	4.8	9.9	1.8	9.2	1.2	0.6	0.6	6.0	17.1	16.5	18.8	6.0	8.5	5.4	12.0	16.3	10.8	14.4	1.2	1.2	6.0	1.8	11.5	12.5	16.8			
SB	g	1.4	1.4	3.7		3.0	1.0	1.9	5.3		1.0	3.3	7.8				2.6	1.0	2.1	0.5		2.0		0.7	0.5	1.9	3.2						2.3	1.8	9.5			
SB	th		0.7		0.5			1.9				1.1	0.6					0.7		0.8				1.4	0.2	0.6	0.5						0.6		1.2			
LFr	f	1.9	1.7	2.4	3.0	1.2	1.0	3.2				3.3	0.6	8.3				2.3		4.6	5.6		2.3	0.7	0.5	0.6	1.4					0.8	1.7	1.6	8.3	2.6			
LFr	v	0.3	0.2	1.7	0.7	3.0	2.7	5.6			1.9	2.2	0.6							3.0	1.9	2.0		2.4	2.2	1.3	1.4						4.0	4.2	7.1			
VlCl	ng	0.5	0.2		0.7		1.0		1.6		1.0	1.1	1.3					0.3		0.5	1.9					1.3	1.6						1.1	0.5	1.8			

Key
top 5%
top 10%
top 15%
top 20%
top 25%

Table 7.3: The 52 most confounded phoneme pairs (top 10% of all confounded pairs)

Phoneme Pair	Viseme Groups	Confusion (%)	Phoneme Pair	Viseme Groups	Confusion (%)
(dh,ih)	(SB,OV)	23.6	(r,ah)	(R,BV)	10.2
(dh,ah)	(SB,BV)	22.1	(k,w)	(SB,RV)	10.2
(dh,iy)	(SB,OV)	21.1	(r,l)	(R,L)	10.1
(dh,r)	(SB,R)	18.8	(t,hh)	(SB,FV)	10
(dh,l)	(SB,L)	17.1	(zh,p)	(Pal,LB)	10
(dh,t)	(SB,SB)	16.8	(aa,ih)	(BV,OV)	9.9
(dh,er)	(SB,R)	16.5	(dh,ao)	(SB,RV)	9.9
(dh,n)	(SB,AlCl)	16.3	(eh,ih)	(FV,OV)	9.8
(dh,ae)	(SB,FV)	15.5	(r,eh)	(R,FV)	9.6
(t,z)	(SB,AlCl)	14.6	(ow,ih)	(RV,OV)	9.5
(dh,s)	(SB,AlCl)	14.4	(g,dh)	(SB,SB)	9.5
(w,iy)	(RV,OV)	13.7	(d,ow)	(SB,RV)	9.5
(eh,ah)	(FV,BV)	13.4	(n,ow)	(AlCl,RV)	9.4
(s,r)	(AlCl,R)	13	(ow,ah)	(RV,BV)	9.4
(dh,k)	(SB,SB)	12.5	(d,eh)	(SB,FV)	9.3
(dh,m)	(SB,LCl)	12	(dh,w)	(SB,RV)	9.2
(dh,eh)	(SB,FV)	11.7	(ah,iy)	(BV,OV)	9.2
(ow,iy)	(RV,OV)	11.6	(s,eh)	(AlCl,FV)	9.2
(dh,d)	(SB,SB)	11.5	(l,w)	(L,RV)	9.1
(r,ow)	(R,RV)	11.4	(w,ah)	(RV,BV)	9.1
(r,w)	(R,RV)	11.2	(ah,ih)	(BV,OV)	8.9
(hh,ae)	(FV,FV)	11.1	(d,s)	(SB,AlCl)	8.8
(r,hh)	(R,FV)	10.8	(p,ow)	(LB,RV)	8.8
(dh,z)	(SB,AlCl)	10.8	(s,z)	(AlCl,AlCl)	8.7
(s,ih)	(AlCl,OV)	10.6	(dh,b)	(SB,LB)	8.5
(s,er)	(AlCl,R)	10.4	(t,l)	(SB,L)	8.5

These phoneme pairs are (dh,t), (dh,k), (dh,d), (hh,ae), (g,dh), and (s,z). If /dh/ is excluded, as it is a very common false positive for all phonemes (see Section 6.2.5 and Section 7.1), the highest confoundedness for any two phonemes within the same traditional viseme is only 11.1% for the /hh/ and /ae/ phoneme pair. A confoundedness of 11.1% is a low value, considering phonemes within the same viseme group are supposed to be visually indistinguishable. In spite of this, these phonemes are only confused for each other once for every eight times they are correctly recognised, after normalisation. This strongly suggests that these two phonemes are in fact visually distinguishable, and as a result they do not belong to the same viseme.

The only other pair from the same traditional viseme to display any significant confoundedness is the /s/ and /z/ phoneme pair, with a value of 8.7%. This is also a low value considering these phonemes are supposed to be visually indistinguishable. All other pairs of phonemes belonging to the same visemes are confounded even less often than this.

If /dh/ is excluded, as it appears to simply be a rogue phoneme, the most confounded pair of phonemes is /t/ and /z/, which are assigned to the SB and AlCl traditional visemes. This pair of phonemes has a confoundedness of 14.6%. This means that the most confounded, non-rogue phoneme pair is still correctly recognised six times as often (after normalisation) as they are confused with each other.

Even if /dh/ is not excluded, the most confounded pair, /dh/ and /ih/, still only has a confoundedness of 23.6% after normalisation, indicating that these phonemes are approximately four times more likely to be correctly recognised, than to be confused for each other.

This strongly suggests that the concept of visemes is fundamentally flawed, as a defining characteristic of a viseme, according to characteristic C2 in Section 2.1, is that they have high intra-viseme errors, which would be indicated by a high confoundedness between phonemes within a viseme group.

This is clearly not present in any of the traditional visemes, nor between any other pair of phonemes.

The analysis of the phoneme confoundedness does not support the hypothesis for the existence of a set of visemes. This is further undermined by investigating the directionality of confusion.

7.3 Directionality Of Confusion

When analysing the confusion matrix, an important characteristic to consider is the directionality of the confusion. By looking at not just the confoundedness, but also the directionality, a better understanding of how phonemes are confused can be gained. According to characteristic C3 (see Section 2.1), intra-viseme confusion should display low directionality, indicative of the phonemes within a viseme being visually indistinguishable.

Phoneme pairs with high directionality are those where one phoneme (phoneme "A") is often confused for another (phoneme "B"), but where this other phoneme is rarely confused for the first. The fact that phoneme "B" is rarely confused for "A" is significant, in that it indicates that it is able to be distinguished visually to some degree. While "A" is still confused for the latter, it is still important to know that "B" is not likely to cause phoneme "A" to appear in the output of the recogniser. Knowing which phonemes are likely to have caused a phoneme to appear in the output of the recogniser can be made use of by the word fitter/sentence constructor.

The directionality value is calculated as the normalised difference between the confoundedness between two phonemes in each direction. That is, for the pair (A, B), it is the difference between the normalised likelihood of "A" being substituted for "B", and "B" being substituted for "A", divided by the sum of these likelihoods (see Equation 7.1). This value will indicate how directional the substitutions are for a pair of phonemes. A directionality

value of +25.0 indicates that confoundedness of "A" for "B" is more than the confoundedness of "B" for "A", by 25% of the combined confoundedness.

$$directionality = \frac{(A \rightarrow B) - (B \rightarrow A)}{(A \rightarrow B) + (B \rightarrow A)} * 100 \tag{7.1}$$

Table 7.4 shows the directionality for each pair of phonemes. Values close to zero indicate no or minimal directionality (highlighted in Table 7.4), while a large positive or negative number indicates a highly directional relationship. If the confusion for a phoneme pair is 100% directional, this indicates that all of the substitutions were in the same direction (that is, "A" is substituted for "B", but "B" is never substituted for "A"). Blank entries (within the lower triangular matrix) indicate no substitutions occurred, so they have been removed for clarity. Positive values indicate the phoneme row is more likely to be substituted in place of the phoneme column than the other way around.

The 75 least directional phoneme pairs of the 520 confused pairs (i.e. the bottom 15%) are shown in Table 7.5, with phoneme pairs belonging to the same traditional viseme shown in bold. As can be seen in Table 7.5, there are only five pairs of phonemes, in the 15% least directional, that both belong to the same traditional viseme group. These pairs are (w,ao), (eh,ay), (z,n), (t,d), and (iy,ih). In fact, there are only six pairs of phonemes with a directionality of less than 30%, where both phonemes belong to the same traditional viseme. This indicates that the majority of low-directionality confusion is not between phonemes within the same viseme group.

Of the 75 pairs listed in Table 7.5, 18 are grouped in the same viseme in at least one of the studies by various researchers listed in Table 1.2 (see page 18). As can be seen in Table 7.6, of these 18 pairs, some were grouped into a viseme in only one or two studies, while others were grouped into visemes in almost every study containing both phonemes.

Table 7.4: Directionality of phoneme confusion. Highly non directional entries have been highlighted

OV	ih																																							
OV	iy	10.7																																						
BV	aa	-72.3	-29.5																																					
BV	ah	5.0	0.1	55.2																																				
FV	ae	-29.0	22.6	-33.3	-16.2																																			
FV	ay	-62.1	-54.4	-100.0	-69.6	-65.7																																		
FV	eh	-38.0	-38.9	43.4	-86.8	31.1	1.0																																	
FV	ey	100.0	-79.2	100.0	100.0	-26.2	100.0	100.0																																
FV	hh	-25.5	-91.7	100.0	-75.0	-67.0	-100.0	100.0																																
RV	ao	48.1	-66.5		-32.2	11.8		50.0	52.9																															
RV	ow	-59.6	-87.5	-28.1	-62.4	-40.8		-6.1	-100.0	-100.0	-54.5																													
RV	w	-77.5	-89.4	30.5	-57.3	-36.1	-45.2	-0.6	-100.0	-2.5	-0.6	100.0																												
RV	aw	-87.6	100.0	100.0	100.0	100.0	100.0		100.0	-100.0	100.0	100.0	100.0																											
RV	oy	100.0	100.0		100.0						100.0																													
RV	uh	100.0	100.0		100.0						100.0		100.0																											
RV	uw	100.0	100.0		100.0	100.0	100.0	100.0	100.0			100.0	100.0																											
L	l	-9.0	3.3	29.9	3.4	13.9	49.8	49.0	40.7	47.6	18.8	53.5	42.3		-100.0	-100.0	-100.0																							
R	er	34.8	46.3	-55.5	-0.4	100.0	100.0	15.1	19.0	100.0	9.3	100.0	71.8		-100.0	-100.0	-100.0	-52.8																						
R	r	24.0	-4.0	51.3	-27.8	-21.4	57.9	40.1	8.8	71.0	2.4	90.8	62.6	-100.0		-100.0	-100.0	17.1	-35.3																					
Y	y	-5.8	100.0	100.0	-44.9		100.0	-100.0		100.0		100.0	100.0				-100.0	100.0	100.0	-33.8																				
LB	b	100.0	-53.6	-100.0	100.0	-4.5	100.0	-51.5			-16.2	100.0	-1.3					100.0	100.0	100.0	-100.0																			
LB	p	100.0	100.0	-100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	48.3	100.0					100.0	100.0	100.0		100.0																		
LCl	m	-21.9	-19.3	54.7	3.3	54.7	100.0	45.8	62.8	-100.0	14.8	100.0	9.6		-100.0		-100.0	-26.9	-100.0	21.8	-100.0	-100.0	52.2																	
AlCl	n	4.0	19.0	68.2	39.3	-35.0	61.9	35.2	-100.0	92.0	22.0	85.3	67.1		-100.0	-100.0	-100.0	-35.5	39.5	32.5	60.1	57.6	-100.0	51.2																
AlCl	z	-74.7	-34.2	100.0	-19.9	41.3	39.8	6.5		49.1	100.0	-100.0	15.5	-100.0		-100.0	-100.0	-62.5	-65.3	1.7	-100.0		-100.0	-14.1	-6.7															
AlCl	s	-27.5	-40.3	-100.0	-32.3	12.6	-100.0	46.9	-1.9	26.3	-47.5	81.9	41.2		-100.0	-100.0	-100.0	-18.0	22.3	12.7	82.3	-100.0	-100.0	-54.6	-55.7	31.4														
Pal	ch	100.0	100.0		100.0	100.0		100.0		100.0								100.0		100.0					100.0		100.0													
Pal	jh	100.0	100.0				100.0			100.0		100.0	100.0					100.0		100.0						100.0	100.0													
Pal	sh	100.0	100.0		100.0	100.0	100.0	100.0				100.0	100.0	100.0				100.0		100.0				100.0	100.0	100.0	100.0													
Pal	zh				100.0			100.0					100.0							100.0		100.0	-100.0		100.0															
SB	d	-23.2	-30.5	50.9	-9.2	21.2	100.0	65.8	-0.8	80.6	100.0	83.3	100.0				-100.0	-3.5	-100.0	-0.5	-100.0	100.0	-100.0	56.5	-11.2	81.1	-26.3	-100.0		-100.0										
SB	k	-27.1	30.8	-30.7	-4.2	3.0	67.3	66.8	-100.0	85.6	38.1	58.2	66.1		-100.0		-100.0	-20.9	-18.1	-16.6	23.4	27.0	32.8	-9.4	9.2	13.8	35.8		-100.0	-100.0		-56.7								
SB	t	12.4	-0.3	75.0	42.8	39.9	69.8	46.7	-100.0	84.4	57.2	78.5	85.1		-100.0	-100.0	-100.0	38.7	-10.3	24.1	33.8	58.5	48.6	7.8	31.9	82.1	14.6	-100.0	81.0	-100.0	-100.0	9.6	49.1							
SB	dh	-81.4	-97.7	-100.0	-88.8	-76.4	-74.0	-83.5	-14.3	-23.3	-80.6	-100.0	-29.4	-100.0	-100.0	-100.0	-100.0	-80.8	-87.1	-83.3	-100.0	-53.2	-100.0	-88.1	-83.0	-64.9	-81.3	-100.0	-100.0	-100.0	-100.0	-86.2	-81.5	-84.4						
SB	g	100.0	100.0	100.0		100.0	100.0	100.0	-100.0		100.0	100.0	32.8				100.0	100.0	100.0	100.0		100.0		100.0	100.0	100.0	100.0						100.0	100.0	100.0					
SB	th		100.0		100.0			100.0				100.0	100.0					100.0		100.0				100.0	100.0	100.0	100.0						100.0		100.0					
LFr	f	100.0	100.0	100.0	-52.7	100.0	100.0	-40.5				100.0	100.0	100.0				100.0		1.8	100.0		-100.0	100.0	100.0	100.0	100.0					100.0	100.0	100.0	100.0	100.0				
LFr	v	100.0	100.0	-100.0	100.0	100.0	-27.5	38.1			100.0	100.0	100.0							-13.7	100.0	100.0		-41.4	-57.8	100.0	100.0						-71.4	-62.9	100.0					
VlCl	ng	100.0	100.0		100.0		100.0		100.0		100.0	100.0	100.0					100.0		100.0	100.0					100.0	100.0						100.0	100.0	100.0					
		ih	iy	aa	ah	ae	ay	eh	ey	hh	ao	ow	w	aw	oy	uh	uw	l	er	r	y	b	p	m	n	z	s	ch	jh	sh	zh	d	k	t	dh	g	th	f	v	ng
		OV	OV	BV	BV	FV	FV	FV	FV	FV	RV	RV	RV	RV	RV	RV	RV	L	R	R	Y	LB	LB	LCl	AlCl	AlCl	AlCl	Pal	Pal	Pal	Pal	SB	SB	SB	SB	SB	SB	LFr	LFr	VlCl

Key
- < ±5
- < ±10
- < ±15
- < ±20

Table 7.5: Least directional phoneme pairs (bottom 15%)

Phoneme pair	Viseme groups	Direction-ality (%)	Phoneme Pair	Viseme Groups	Direction-ality (%)
(ah,iy)	(BV,OV)	0.1	**(iy,ih)**	(OV,OV)	10.7
(t,iy)	(SB,OV)	-0.3	(d,n)	(SB,AlCl)	-11.2
(er,ah)	(R,BV)	-0.4	(ao,ae)	(RV,FV)	11.8
(d,r)	(SB,R)	-0.5	(t,ih)	(SB,OV)	12.4
(w,eh)	(RV,FV)	-0.6	(s,ae)	(AlCl,FV)	12.6
(w,ao)	(RV,RV)	-0.6	(s,r)	(AlCl,R)	12.7
(d,ey)	(SB,FV)	-0.8	(v,r)	(LFr,R)	-13.7
(eh,ay)	(FV,FV)	1.0	(k,z)	(SB,AlCl)	13.8
(b,w)	(LB,RV)	-1.3	(l,ae)	(L,FV)	13.9
(z,r)	(AlCl,R)	1.7	(z,m)	(AlCl,LCl)	-14.1
(f,r)	(LFr,R)	1.8	(dh,ey)	(SB,FV)	-14.3
(s,ey)	(AlCl,FV)	-1.9	(t,s)	(SB,AlCl)	14.6
(r,ao)	(R,RV)	2.4	(m,ao)	(LCl,RV)	14.8
(w,hh)	(RV,FV)	-2.5	(er,eh)	(R,FV)	15.1
(k,ae)	(SB,FV)	3.0	(z,w)	(AlCl,RV)	15.5
(l,iy)	(L,OV)	3.3	(ae,ah)	(FV,BV)	-16.2
(m,ah)	(LCl,BV)	3.3	(b,ao)	(LB,RV)	-16.2
(l,ah)	(L,BV)	3.4	(k,r)	(SB,R)	-16.6
(d,l)	(SB,L)	-3.5	(r,l)	(R,L)	17.1
(r,iy)	(R,OV)	-4.0	(s,l)	(AlCl,L)	-18.0
(n,ih)	(AlCl,OV)	4.0	(k,er)	(SB,R)	-18.1
(k,ah)	(SB,BV)	-4.2	(l,ao)	(L,RV)	18.8
(b,ae)	(LB,FV)	-4.5	(n,iy)	(AlCl,OV)	19.0
(ah,ih)	(BV,OV)	5.0	(er,ey)	(R,FV)	19.0
(y,ih)	(Y,OV)	-5.8	(m,iy)	(LCl,OV)	-19.3
(ow,eh)	(RV,FV)	-6.1	(z,ah)	(AlCl,BV)	-19.9
(z,eh)	(AlCl,FV)	6.5	(k,l)	(SB,L)	-20.9
(z,n)	(AlCl,AlCl)	-6.7	(d,ae)	(SB,FV)	21.2
(t,m)	(SB,LCl)	7.8	(r,ae)	(R,FV)	-21.4
(r,ey)	(R,FV)	8.8	(m,r)	(LCl,R)	21.8
(l,ih)	(L,OV)	-9.0	(m,ih)	(LCl,OV)	-21.9
(d,ah)	(SB,BV)	-9.2	(n,ao)	(AlCl,RV)	22.0
(k,n)	(SB,AlCl)	9.2	(s,er)	(AlCl,R)	22.3
(er,ao)	(R,RV)	9.3	(ae,iy)	(FV,OV)	22.6
(k,m)	(SB,LCl)	-9.4	(d,ih)	(SB,OV)	-23.2
(m,w)	(LCl,RV)	9.6	(dh,hh)	(SB,FV)	-23.3
(t,d)	(SB,SB)	9.6	(hh,ih)	(FV,OV)	-25.5
(t,er)	(SB,R)	-10.3			

The most commonly grouped phoneme pair is (t,d), which was grouped into the same viseme in all 10 studies that contained both of these phonemes. Another common grouping was (d,n) which was grouped in all seven of the studies that included both phonemes. While these two pairs have low directionality, and are often grouped within a viseme, neither demonstrate the high confoundedness required by characteristic C2, with only 5.8% and 5.7% respectively (see Table 7.2 on page 169).

Of the 13 consonant pairs listed in Table 7.6, all but one these pairs occur in a "super group" in at least 70% of the studies that group these pairs into the same visemes. The (r,l) pair is only grouped in two studies, one of which was in a super group. This indicates that almost all non-directional pairs are typically included in a viseme due to the presence of a super group, which contains many phonemes. This does not support the existence of a set of visemes, as a super group is typically created by taking all the phonemes not yet placed within any other viseme, and creating a group from these phonemes.

All three studies containing vowels grouped the pair (iy, ih) into a viseme. The (eh,ah) pair was grouped in two out of three studies of vowels, and the (er,ah), (er,oa), and (w,ao) were each grouped in only one of the three studies containing vowels. Since only three studies included vowels, it is difficult to draw strong conclusions from these pairs. Instead, the characteristics of the directionality, across all pairs, needs to be examined further.

Figure 7.2 shows a histogram of the magnitude of the directionality of phoneme confusion. This figure excludes the 221 phoneme pairs (29.8% of the total number of pairs) with no confusion, as the directionality would be undefined in these cases. From a total of 520 pairs that display non-zero confusion, there are 275 pairs where the directionality is exactly 100%. This shows that the majority of confusions are uni-directional, which is a strong argument against the traditional viseme concept according to characteristic C3.

Table 7.6: Least directional phonemes grouped together in the literature

Phoneme pair	Directionality (%)	Number of times grouped together	Number of studies including both phonemes	Number of studies with pair in a super group
(t,d)	9.6	10	10	7
(d,n)	-11.2	7	7	5
(t,s)	14.6	4	5	3
(z,n)	-6.7	3	4	3
(k,n)	9.2	5	7	5
(s,l)	-18.0	2	3	2
(k,z)	13.8	2	4	2
(d,l)	-3.5	3	8	3
(k,l)	-20.9	3	8	3
(z,r)	1.7	1	4	1
(s,r)	12.1	1	4	1
(r,l)	17.1	2	9	1
(d,r)	-0.5	1	10	1
(iy,ih)	10.7	3	3	N/A
(eh,ay)	1.0	2	3	N/A
(er,ah)	-0.4	1	3	N/A
(w,ao)	-0.6	1	3	N/A
(er,ao)	9.3	1	3	N/A

When pairs have highly directional confusion, it indicates that one of the phonemes within the pair is rarely confused for the other phoneme. This disagrees with the concept of a viseme containing visually indistinguishable phonemes, as some of the phonemes can in fact be visually distinguished from the others.

If only the pairs that each belong to the same traditional viseme group are analysed, it is clear that the majority of confusion is still highly directional (see Figure 7.3). Of the 60 pairs where each phoneme belongs to the same traditional viseme, there are only 38 pairs (63.3%) that have any level of confusion, 19 of which are 100% directional.

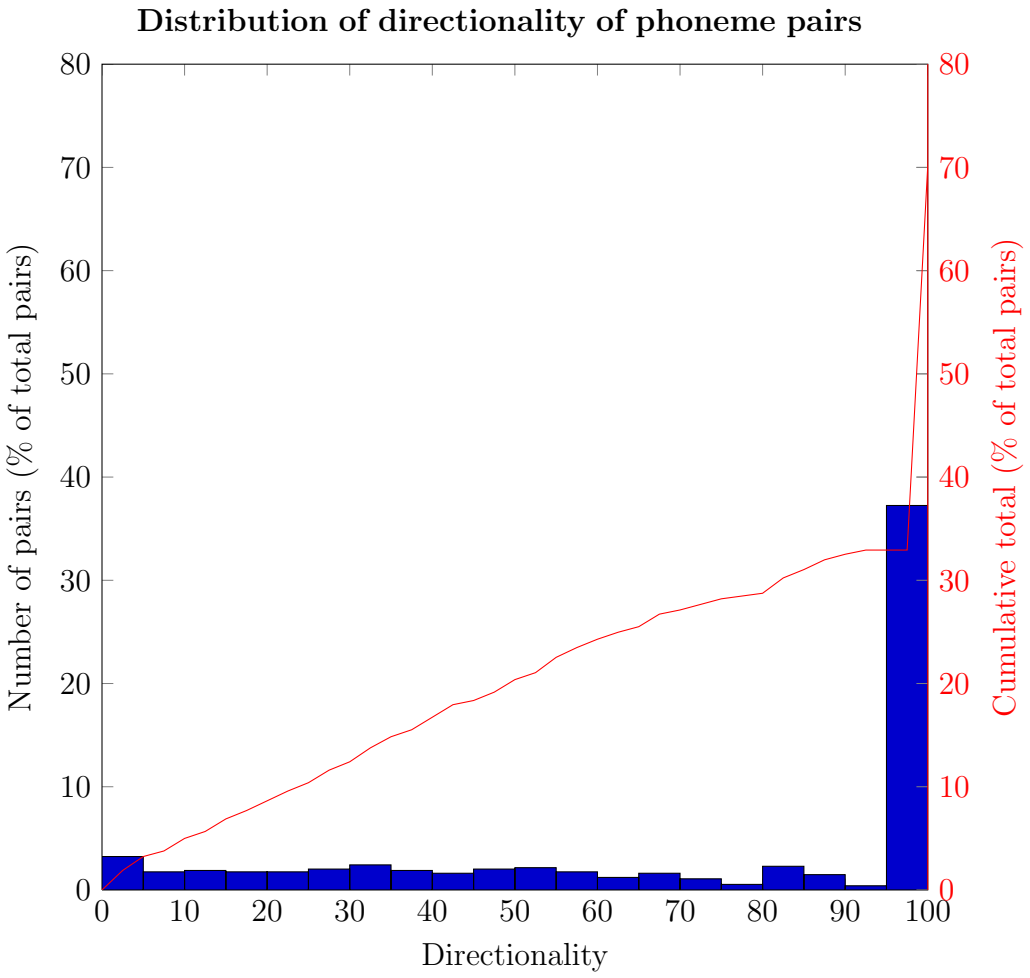


Figure 7.2: Histogram showing distribution of directionality of phoneme confusion

These findings strongly disagree with the concept of a viseme containing visually indistinguishable phonemes, as the large majority of them have highly directional confusion. This indicates that phonemes are visually distinguishable to some degree. As discussed in Section 1.3, viseme groups have traditionally been based on confusion matrices of human responses. This suggests that while humans may not be able to visually distinguish between phonemes within each viseme group, the algorithms used within this research can to some degree.

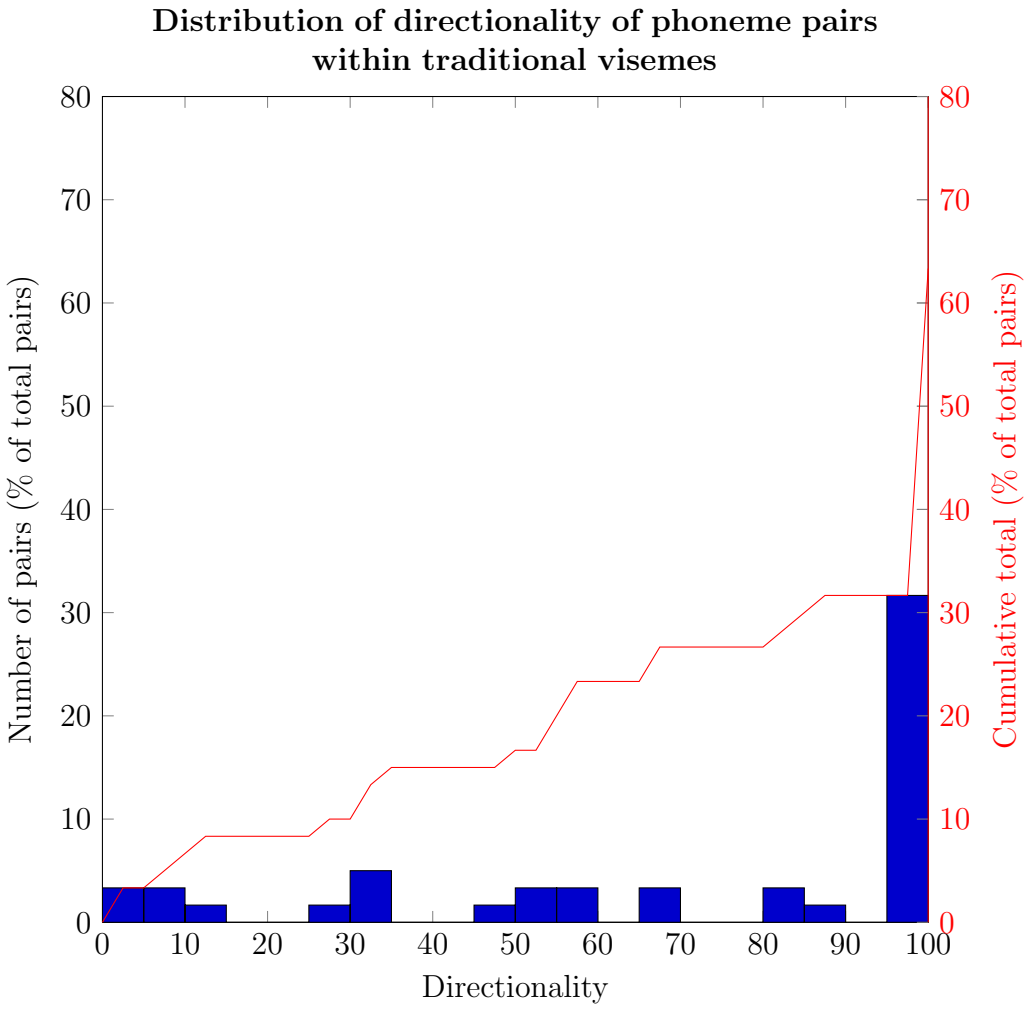


Figure 7.3: Histogram showing distribution of directionality of phoneme confusion, for pairs within traditional visemes only

7.4 Phoneme Pairs With High Confoundedness And Low Directionality

According to characteristics C2 and C3 of the hypothesis, phonemes within each viseme must have high confoundedness and low directionality. Table 7.7 lists the five phoneme pairs that have confoundedness greater than 10%, and less than 50% directionality.

Table 7.7: Phoneme pairs with confoundedness greater than 10%, and directionality less than 50%

Phoneme Pair	Traditional Viseme Groups	Confoundedness	Directionality
(s,r)	(AlCl,R)	13.0	12.7
(s,ih)	(AlCl,OV)	10.6	-27.5
(s,er)	(AlCl,R)	10.4	22.3
(r,ah)	(R,BV)	10.2	-27.8
(r,l)	(R,L)	10.1	17.1

The five phoneme pairs listed are the closest candidates for having high confoundedness and low directionality, yet even these pairs do not demonstrate significantly high confoundedness, with the maximum being 13.0% for the (s,r) phoneme pair. This indicates that the most confounded pair of phonemes which also had low directionality still had seven correct matches for every substitution that occurred between this pair. This clearly shows that of the pairs with low directionality, none demonstrate high confoundedness. This is clearly in contradiction with characteristics C2 and C3.

It should also be noted that of these five pairs, (r,l) is grouped in only two studies (one of which was in a super group), and (s,r) is grouped in only one study which was due to a directional grouping only. The other three pairs were not grouped in any of the studies.

As an alternative approach, the most confounded (top 15%) and least direction (bottom 15%) pairs are considered. As shown in Table 7.8, there are only 11 pairs meeting these criteria. It stands out that none of these pairs have both phonemes belonging to the same traditional viseme group.

If the phoneme pairs listed in Table 7.8 are compared to the viseme groups used by other researchers (see Table 1.2 on page 18), it can be seen that two of the eleven pairs occur within a proper viseme, as determined by at least one study, and one additional pair occurring in a study if directional inclusions are considered.

Table 7.8: Phoneme pairs in the top 15% for confoundedness and bottom 15% for directionality

Phoneme Pair	Traditional Viseme Groups	Confoundedness	Directionality
(s,r)	(AlCl,R)	13.0	12.7
(s,er)	(AlCl,R)	10.4	22.3
(r,l)	(R,L)	10.1	17.1
(ah,iy)	(BV,OV)	9.2	0.1
(ah,ih)	(BV,OV)	8.9	5.0
(ae,ah)	(FV,BV)	8.4	-16.2
(k,er)	(SB,R)	7.8	-18.1
(r,ae)	(R,FV)	7.8	-21.4
(w,eh)	(RV,FV)	7.7	-0.6
(k,n)	(SB,AlCl)	7.6	9.2
(t,r)	(SB,R)	7.6	24.1

Only the (k,n) phoneme pair occurs in the same viseme in multiple studies, occurring in six of the fourteen studies. The (r,l) pair occurs in a viseme in one study, and as a directional inclusion to a viseme in another study. The (s,r) pair only occurs in a single study, and still only as a direction inclusion to a viseme.

When the confoundedness of each of these pairs is examined, it is clear that these phonemes are not actually significantly confounded with each other. For the (k,n) pair, which was the only pair that appeared within a viseme in several studies, the two phonemes are still only likely to be confounded once for every twelve times they are correctly recognised.

The (r,l) pair has a confoundedness of 10.1%, which means that these two phonemes are confounded once for every nine correctly recognised occurrences. The (s,r) pair, which is the most confounded pair that also has directionality in the bottom 15%, is still only confounded once for every eight times they are correctly recognised. This shows that the phonemes with high confoundedness and low directionality are still not significantly confounded,

as they are clearly able to be correctly identified many times more often than they are confounded.

According to characteristics C2 and C3, phonemes within a viseme should demonstrate significant substitution errors for all other phonemes within that viseme, and the substitutions should be non-directional. If there was a set of visemes that had not yet been identified, there would be a significant number of phonemes pairs that display both high confoundedness and low directionality. As shown in this section, this is clearly not emergent in the data.

The data indicate that there is not a set of visemes that satisfy the characteristics, C1–C3, as required by the hypothesis in Section 2.1. This strongly suggests that there are not groups of phonemes that are mostly visually indistinguishable from each other, and hence that visemes are not suitable for use as the basic visual unit of speech.

Chapter 8

Conclusion

The hypothesis put forward in Chapter 2 states that if visemes are the basic visual unit of speech, there exists a viseme set such that the output of the phoneme recogniser will exhibit the following three characteristics:

C1. High Ratio Of Intra-Viseme To Inter-Viseme Substitutions

C2. High Confoundedness Within Visemes

C3. Non-Directionality Of Substitutions

To test this hypothesis, a visual speech recogniser has been constructed, and test data captured. In this conclusion, each of the three required characteristics will be examined in turn to show that none are supported by the data. It is shown that none of the new or existing viseme groupings demonstrate the characteristics, and it is not possible to construct any alternative viseme grouping that can, leaving no choice but to reject the hypothesis.

8.1 C1. High Ratio Of Intra-Viseme To Inter-Viseme Substitutions

Characteristic C1 states that the intra-viseme substitutions must be significantly higher than the inter-viseme substitutions, and that 70% of inputs must be recognised within the correct viseme group. This characteristic is fundamental to the idea of visemes being groups of visually indistinguishable phonemes.

In Chapter 6, a number of existing viseme groupings from the literature were examined, as well as some new groupings. The existing groupings from the literature were examined in Section 6.4, and none were found to satisfy characteristic C1, with the intra-viseme substitutions being significantly lower than the inter-viseme substitutions. When the traditional viseme groups were examined in Section 6.1, characteristic C1 was again not evident, with 11 times as many inter-viseme substitutions (3636) as intra-viseme substitutions (311).

A systematic process was applied to the traditional viseme groups to create a more suitable set of groupings (see Section 6.2), but this set also fails to satisfy characteristic C1 with significantly fewer intra-viseme substitutions (311) than inter-viseme substitutions (3696). Next, a set of visemes was constructed by grouping the noisiest phonemes into the largest viseme groups, maximising the number of phonemes to be recognised within the correct viseme group (see Section 6.3). This grouping was designed to provide an upper bound on recognition of the correct viseme. While it had the highest ratio of intra- to inter-viseme substitutions, it still had more than five times as many inter-viseme substitutions (3315) as intra-viseme substitutions (632). This clearly fails to satisfy characteristic C1.

This characteristic is the most fundamental to visemes, as it is the one that establishes the clusters of visually indistinguishable phonemes. With these

clusters not evident in any of the existing or new groupings, it leaves the hypothesis in strong doubt.

To determine if it is possible to create any viseme grouping that can satisfy C1, the underlying phoneme confusion was examined in Section 7.1: "Phoneme Trustworthiness". It was found that the only phonemes that could successfully create a viseme to satisfy the 70% threshold of C1, were those trustworthy phonemes that were recognised correctly at the phoneme level the majority of the time.

While these phonemes are able to be grouped to reach the 70% threshold, they do not fully satisfy C1, as they do not have a high ratio of intra-viseme to inter-viseme substitutions. These phonemes have very low numbers of substitutions, as they are mostly recognised as the correct phonemes. This does not support the existence of a set of visemes; instead it points to the individual phonemes being visually distinguishable in their own right.

With such low numbers of substitutions for these phonemes, it is not possible to reliably calculate a intra-viseme to inter-viseme substitution ratio. To definitively determine the validity of the hypothesis, characteristics C2 and C3 must therefore be examined.

8.2 C2. High Confoundedness Within Visemes

Characteristic C2 requires high confoundedness between all phonemes belonging to a viseme. As visemes require member phonemes to be visually indistinguishable, the input of a member phoneme should be recognised as all other member phonemes at some point.

After examining characteristic C1 in the previous section, the hypothesis is in strong doubt, with only a number of high trustworthiness phonemes able to be grouped to potentially satisfy the 70% threshold of C1. The phonemes

within this grouping however, have very few substitutions. This contradicts C2 which requires all phonemes within a viseme group to have significant confoundedness with all other member phonemes.

When the counfoundedness between phonemes is considered among the full set of phonemes, it is clear that there is no grouping that can demonstrate high confoundedness between all member phonemes. This is best illustrated by Figure 7.1 on page 164. When examining this figure, it is clear that the likelihood for each substitution is always significantly less than the likelihood for a correct phoneme match. The median likelihood for the most likely substitution for each phoneme is only 8.7%, and the second most likely is only 6.1%. These are all significantly lower than the median likelihood for being a correct phoneme, at 45.6%.

Section 7.2 examines the level of confoundedness between all possible pairs of phonemes. For C2 to be potentially satisfied, there would have to be significant confoundedness within a number of phoneme pairs. Of the 771 possible phoneme pairs, 221 (29.8%) did not display any confusion (see Section 7.3).

As can be seen in Table 7.3 (see page 170), the most confounded pair is (dh,ih) at 23.6% confusion. While this may sound significant, it still means that this pair is correctly recognised almost four times as often as it is confounded. It must also be noted that /dh/ is a common substitution in place of almost every other phoneme. The /dh/ phoneme does not demonstrate any clustering (as required by both C1 and C2), but instead is simply a rogue phoneme (see Section 6.2.5 and Section 7.1). With /dh/ excluded, the most confounded pair is now (t,z), with 14.6% confusion. This means that the most confounded non-rogue phoneme pair are still correctly recognised six times as often as they are confused with each other.

With the highest level of confoundedness being 23.6%, or 14.6% for non-rogue phonemes, it is clear that there is not a high level of confoundedness between pairs of phonemes. The confoundedness within a viseme is bounded above

by the highest level of confoundedness of the contained phoneme pairs. All other pairs will be even less confounded. As the most confounded pair still has a relatively low level of confoundedness, it is clear that no viseme can be constructed to achieve a high level of confoundedness among all member phonemes. This shows that the substitutions for each phoneme are spread among many other phonemes, not appearing within a cluster. This is in contradiction with the concept of visemes, as there are no evident clusters.

As the individual phonemes do not demonstrate any significant clustering, no visemes can be constructed to satisfy characteristic C2. Before rejecting the hypothesis, the directionality characteristic C3 must also be examined.

8.3 C3. Non-Directionality Of Substitutions

Characteristic C3 states that visemes should exhibit non-directional substitutions between member phonemes. This requires that for two phonemes within a viseme, they should be substituted in place with each other a proportional number of times. If these phonemes demonstrate a bias by exhibiting directional substitutions, it indicates that they are in fact visually distinguishable to some degree, as one is less likely to be confused for the other.

Section 7.3 examined the directionality of all possible phoneme pairs. Of the 771 possible phoneme pairs, 251 (32.5%) did not display any confusion and 275 (35.7%) displayed 100% directionality, leaving just 245 (31.8%) that displayed less than 100% directionality (see Figure 7.2 on page 178).

Table 7.6 (see page 177) shows the 18 pairs that are the least directional and occur within a viseme group in at least one study in the literature. Of the 13 consonant pairs listed in this table, all but one occur in a “super group” in at least 70% of the studies that grouped these pairs into the same viseme.

The presence within a super group must be considered with caution, as super groups are typically created by taking all phonemes not yet placed within any

viseme, and creating a group from these phonemes. This does not support the hypothesis, as these groups are not created based on the phonemes being visually indistinguishable, but instead simply because they were left over after grouping other phonemes.

Section 7.4 examines these non-directional phoneme pairs, to determine if any also exhibit high confoundedness. There are only five phoneme pairs that exhibited less than 50% directionality, and greater than 10% confoundedness. These pairs are the closest candidates for grouping into visemes that could be found, yet they still exhibit very low confoundedness.

The most confounded of these five pairs is (s,r), which has a confoundedness of 13.0%, and a directionality of 12.7%. This shows that the best candidate pair was still recognised as the correct phoneme seven times for every substitution that occurred between this pair. This clearly does not satisfy C2.

Characteristic C1 is required for visemes to be valid, yet not one grouping exhibited this characteristic. The underlying phoneme characteristics show that it is not possible to construct a set that will satisfy characteristic C1, and C2. It was also found that the majority of phoneme pairs exhibited either 100% directionality, or no confoundedness at all, failing to satisfy C3. Of the phoneme pairs that were able to satisfy characteristic C2, none were able to satisfy either C1 or C3.

After examining the new and existing viseme groupings, and the underlying phoneme characteristics, there is no choice but to reject the hypothesis of visemes being the basic visual unit of speech.

8.4 Recommendations

It has been shown that no possible set of visemes can satisfy the requirements in the hypothesis. This shows that phonemes, not visemes, are the basic

visual unit of speech. Further, visemes are in fact detrimental to visual speech recognition.

By using visemes, useful information is lost that could otherwise be used to construct a more effective word or sentence constructor. Visemes discard the information contained within the specific phoneme substitutions that should be used to improve the effectiveness of a word fitting algorithm.

By outputting phonemes from a visual speech recogniser, these detailed characteristics should be used to determine the trustworthiness of the specific phoneme, and the most likely cause of each phoneme appearing in the output.

For example, there are five phonemes (/oy/, /uh/, /ch/, /th/, and /ng/) which exhibited 100% trustworthiness. If any of these phonemes appears in the output, a word fitting algorithm should take this into account, and apply a very high cost to the substitution or deletion of these phonemes. On the other hand, the /dh/ phoneme exhibits a very high likelihood of being an insertion or substitution error. This allows the algorithm to apply a low cost to the substitution or deletion of this phoneme when trying to fit a word to the phoneme sequence.

By adjusting these costs based on the likelihood of each operation causing the phoneme to appear, the accuracy of word fitting algorithms can be improved. These improvements are not possible if visemes are used as the basic visual unit of speech, and as such, phonemes should be used as the basic visual unit of speech.

8.5 Contribution of Thesis

There are two major contributions in this thesis. The first is the development of an improved lip segmentation algorithm, known as “wrapping snakes”. The second, is disproving the validity of visemes for visual speech recognition.

The wrapping snakes algorithm improves the robustness of the traditional snake algorithm. By modifying the external force to only act perpendicular to the snake, it allows the snake to expand along partially located features. When this is combined with the new pinching force and cutting process, the snake is able to successfully locate features even in the presence of significant noise. These behaviours are particularly useful for lip segmentation, as there is often strong noise in close proximity to the lips, typically due to shadowing below the nose and jaw. By improving the robustness of the lip segmentation process, lip shapes can be determined with higher accuracy.

The second, and most important contribution, is disproving the validity of visemes for visual speech recognition. Proving that visemes are not the basic visual unit of speech is a significant step for visual speech recognition. The validity of using visemes for visual speech recognition has not previously been studied, yet they are commonly used as the visual unit of speech. Proving they are not valid allows the future research to instead focus on the use of phonemes as the visual unit of speech.

References

Adini, Y., Y. Moses, and S. Ullman (1997). "Face Recognition: The Problem of Compensating for Changes in Illumination Direction". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.7, pp. 721–732. DOI: 10.1109/34.598229.

Aleksic, P. S., G. Potamianos, and A. K. Katsaggelos (2005). "Exploiting Visual Information in Automatic Speech Processing". In: *Handbook of Image and Video Processing*. Ed. by A. Bovik. 2nd. Burlington: Academic Press, pp. 1263–1289. DOI: 10.1016/B978-012119792-6/50134-0.

Belhumeur, P. N., J. P. Hespanha, and D. J. Kriegman (1997). "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.7, pp. 711–720. DOI: 10.1109/34.598228.

Benguerel, A. P. and M. K. Pichora-Fuller (1982). "Coarticulation Effects in Lipreading". In: *Journal of Speech and Hearing Research* 25.4, p. 600. DOI: 10.1109/34.598228.

Bernstein, L. E. and C. Benoit (1996). "For Speech Perception By Humans Or Machines, Three Senses Are Better Than One". In: *Proceedings of the Fourth International Conference on Spoken Language (ICSLP 96)*. Vol. 3, pp. 1477–1480. DOI: 10.1109/ICSLP.1996.607895.

Binnie, C. A., P. L. Jackson, and A. A. Montgomery (1976). "Visual Intelligibility of Consonants: A Lipreading Screening Test With Implications for Aural Rehabilitation". In: *Journal of Speech and Hearing Disorders* 41.4, p. 530. PMID: 994484.

Binnie, C. A., A. A. Montgomery, and P. L. Jackson (1974). "Auditory and Visual Contributions to the Perception of Consonants". In: *Journal of Speech and Hearing Research* 17.4, p. 619. PMID: 4444283.

Bregler, C., H. Hild, S. Manke, and A. Waibel (1993). "Improving Connected Letter Recognition by Lipreading". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 1, pp. 557–560. DOI: 10.1109/ICASSP.1993.319179.

Campbell, R. (2008). "The Processing of Audio-Visual Speech: Empirical and Neural Bases". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 363.1493, pp. 1001–1010. DOI: 10.1098/rstb.2007.2155.

Carnegie Mellon University (2008). *CMU Pronouncing Dictionary CMUdict 0.7a*. URL: http://www.speech.cs.cmu.edu/cgi-bin/cmudict.

Cassell, J., H. H. Vilhjálmsson, and T. Bickmore (2001). "BEAT: the Behavior Expression Animation Toolkit". In: *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. New York, NY, USA: ACM, pp. 477–486. DOI: 10.1145/383259.383315.

Chen, T. (2001). "Audiovisual Speech Processing". In: *IEEE Signal Processing Magazine* 18.1, pp. 9–21. DOI: 10.1109/79.911195.

Chen, T. and R. R. Rao (1998). "Audio-Visual Integration in Multimodal Communication". In: *Proceedings of the IEEE* 86.5, pp. 837–852. DOI: 10.1109/5.664274.

Chenyang, X. and J. L. Prince (1998). "Snakes, Shapes, and Gradient Vector Flow". In: *IEEE Transactions on Image Processing* 7.3, pp. 359–369. DOI: 10.1109/83.661186.

Cootes, T. F. and C. J. Taylor (1992). "Active Shape Models - Smart Snakes". In: *Proceedings of the British Machine Vision Conference*.

Danihelka, J. and L. Kencl (2010). "Reduction of Animated Models for Embedded Devices". In: *International Conference on Computer Graphics, Visualization and Computer Vision*.

Davis, K. H., R. Biddulph, and S. Balashek (1952). "Automatic Recognition of Spoken Digits". In: *The Journal of the Acoustical Society of America* 24.6, pp. 637–642. DOI: 10.1121/1.1906946.

Dupont, S. and J. Luettin (2000). "Audio-Visual Speech Modeling for Continuous Speech Recognition". In: *IEEE Transactions on Multimedia* 2.3, p. 141. DOI: 10.1109/6046.865479.

Ezzat, T. and T. Poggio (1998). "MikeTalk: A Talking Facial Display Based on Morphing Visemes". In: *Computer Animation 98. Proceedings*, pp. 96–102. DOI: 10.1109/CA.1998.681913.

Finn, K. E. and A. A. Montgomery (1988). "Automatic Optically-Based Recognition of Speech". In: *Pattern Recognition Letters* 8.3, pp. 159–164. DOI: 10.1016/0167-8655(88)90094-3.

Fisher, C. G. (1968). "Confusions Among Visually Perceived Consonants". In: *Journal of Speech and Hearing Research* 11.4, p. 796. PMID: 5719234.

Forgie, J. W. and C. D. Forgie (1959). "Results Obtained from a Vowel Recognition Computer Program". In: *The Journal of the Acoustical Society of America* 31.11, pp. 1480–1489. DOI: 10.1121/1.1936151.

Gailey, L. (1987). "Psychological Parameters of Lip-Reading Skill". In: *Hearing by Eye: The Psychology of Lip-Reading*. Erlbaum, pp. 115–141.

Goldschen, A. J., O. N. Garcia, and E. Petajan (1994). "Continuous Optical Automatic Speech Recognition by Lipreading". In: *Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*. Vol. 1, pp. 572–577. DOI: 10.1109/ACSSC.1994.471517.

Haykin, S. S. (1999). *Neural Networks: A Comprehensive Foundation*. 2nd. Upper Saddle River, NJ: Prentice Hall.

Hazen, T. J. (2006). "Visual Model Structures and Synchrony Constraints for Audio-Visual Speech Recognition". In: *IEEE Transactions on Audio, Speech, and Language Processing* 14.3, pp. 1082–1089. DOI: 10.1109/TSA.2005.857572.

Hazen, T. J., K. Saenko, C. H. La, and J. R. Glass (2004). "A Segment-Based Audio-Visual Speech Recognizer: Data Collection, Development, and Initial Experiments". In: *Proceedings of the 6th International Conference on Multimodal Interfaces*, pp. 235–242. DOI: 10.1145/1027933.1027972.

Heider, F. and G. Heider (1940). "An Experimental Investigation of Lip Reading". In: *Psychological Monographs* 52, pp. 1–153.

Hilder, S., B.-J. Theobald, and R. Harvey (2010). *In Pursuit of Visemes*.

Holden, E. J. and R. Owens (2000). "Visual Speech Recognition using Cepstral Images". In: *Proceedings of IASTED International conference on Signal and Image Processing*, 331–336.

Hyde, S. R. (1972). "Automatic Speech Recognition: A Critical Survey and Discussion of the Literature". In: *Human communication: a unified view*. McGraw-Hill, 399–438.

International Phonetic Association (2005). *The International Phonetic Alphabet*.

International Phonetics Association (1999). *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press.

ISO/IEC 14496-10 (2004). *Information technology - Coding of Audio-Visual Objects - Part 10: Advanced Video Coding*. ISO, Geneva, Switzerland.

ISO/IEC 14496-2 (2001). *Information technology - Coding of Audio-Visual Objects - Part 2: Visual*. ISO, Geneva, Switzerland.

ITU-R BT.601 (1995). *Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide-Screen 16:9 Aspect Ratios*. ITU, Geneva, Switzerland.

Jankowski, C., A. Kalyanswamy, S. Basson, and J. Spitz (1990). "NTIMIT: A Phonetically Balanced, Continuous Speech, Telephone Bandwidth Speech Database". In: *International Conference on Acoustics, Speech, and Signal Processing (ICASSP90)*. DOI: 10.1109/ICASSP.1990.115550.

Kass, M., A. Witkin, and D. Terzopoulos (1988). "Snakes: Active Contour Models". In: *International Journal of Computer Vision* 1.4, pp. 321–331. DOI: 10.1007/BF00133570.

Kumar, K., T. Chen, and R. M. Stern (2007). "Profile View Lip Reading". In: *IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*. DOI: 10.1109/ICASSP.2007.366941.

Lucey, P., T. Martin, and S. Sridharan (2004). "Confusability of Phonemes Grouped According to their Viseme Classes in Noisy Environments". In: *Tenth Australian International Conference on Speech Science & Technology*. Macquarie University, Sydney.

McGurk, H. and J. MacDonald (1976). "Hearing Lips and Seeing Voices". In: *Nature* 264.5588, p. 746. DOI: 10.1038/264746a0.

McKenna, S., S. Gong, R. Würtz, J. Tanner, and D. Banin (1997). "Tracking Facial Feature Points with Gabor Wavelets and Shape Models". In: *Proceedings of the First International Conference on Audio and Video-Based Biometric Person Authentication*. DOI: 10.1007/BFb0015977.

Neti, C., G. Potamianos, J. Luettin, I. Matthews, H. Glotin, D. Vergyri, J. Sison, A. Mashari, and J. Zhou (2000). "Audio-Visual Speech Recognition". In: Final Workshop 2000 Report.

Niswar, A., E. P. Ong, H. T. Nguyen, and Z. Huang (2009). "Real-Time 3D Talking Head from a Synthetic Viseme Dataset". In: *Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry*. VRCAI '09. Yokohama, Japan: ACM, pp. 29–33. DOI: 10.1145/1670252.1670260.

Ostermann, J. (1998). "Animation of synthetic faces in MPEG-4". In: *Computer Animation 98. Proceedings*, pp. 49–55. DOI: 10.1109/CA.1998.681907.

Ostermann, J. (2002). "Face Animation in MPEG-4". In: *MPEG-4 Facial Animation: The Standard, Implementation and Applications*. John Wiley and Sons, pp. 17–55. DOI: 10.1002/0470854626.ch2.

Owens, E. and B. Blazek (1985). "Visemes Observed by Hearing-Impaired and Normal-Hearing Adult Viewers". In: *Journal of Speech, Language, and Hearing Research* 28.3, pp. 381–393. PMID: 4046579.

Pallett, D. S. (2003). "A Look at NIST's Benchmark ASR Tests: Past, Present, and Future". In: *IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 483–488. DOI: 10.1109/ASRU.2003.1318488.

Patterson, E., S. Gurbuz, Z. Tufekci, and J. N. Gowdy (2002). "CUAVE: A New Audio-Visual Database for Multimodal Human-Computer Interface Research". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 2. ICASSP '02, pp. 2017–2020. DOI: 10.1109/ICASSP.2002.5745028.

Potamianos, G., C. Neti, G. Iyengar, A. W. Senior, and A. Verma (2001). "A Cascade Visual Front End for Speaker Independent Automatic Speechread-

ing". In: *International Journal of Speech Technology* 4.3, pp. 193–208. DOI: 10.1023/A:1011352422845.

Potamianos, G., C. Neti, G. Gravier, A. Garg, and A. W. Senior (2003). "Recent Advances in the Automatic Recognition of Audiovisual Speech". In: *Proceedings of the IEEE* 91.9, pp. 1306–1326. DOI: 10.1109/JPROC.2003.817150.

Potamianos, G., C. Neti, J. Luettin, and I. Matthews (2004). "Audio-Visual Automatic Speech Recognition: An Overview". In: *Issues in Visual and Audio-Visual Speech Processing*. MIT Press.

Potter, R. K., G. A. Kopp, and H. C. G. Kopp (1966). *Visible Speech*. Dover Publications Inc.

Rabiner, L. and B.-H. Juang (2008). "Historical Perspective of the Field of ASR/NLU". In: *Springer Handbook of Speech Processing*. Ed. by J. Benesty, M. M. Sondhi, and Y. Huang. Springer Berlin Heidelberg, pp. 521–538. DOI: 10.1007/978-3-540-49127-9_26.

Ramage, M. and E. Lindsay (2009). "Wrapping Snakes for Improved Lip Segmentation". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)*, pp. 1205–1208. DOI: 10.1109/ICASSP.2009.4959806.

Reisberg, D., J. McLean, and A. Goldfield (1987). "Easy to Hear but Hard to Understand: A Lip-Reading Advantage with Intact Auditory Stimuli". In: *Hearing by eye: The psychology of lip-reading*. Ed. by B. Dodd and R. Campbell. Lawrence Erlbaum Associates, pp. 97–114.

Revéret, L. (1997). "From Raw Images of the Lips to Articulatory Parameters: A Viseme-Based Prediction". In: *Fifth European Conference on Speech Communication and Technology*. ISCA.

Revéret, L. and C. Benot (1997). "A Viseme-Based Approach to Labiometrics for Automatic Lipreading". In: *Audio-and Video-based Biometric Person Authentication*. Springer, pp. 335–342. DOI: 10.1007/BFb0016013.

Riedmiller, M. and H. Braun (1993). "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm". In: *IEEE International Conference on Neural Networks*. Vol. 1, pp. 586–591. DOI: 10.1109/ICNN.1993.298623.

Rogozan, A. (1999). "Discriminative Learning of Visual Data for Audiovisual Speech Recognition". In: *International Journal on Artificial Intelligence Tools* 8.1, pp. 43–52. DOI: 10.1142/S021821309900004X.

Sanderson, C. (2008). *Biometric Person Recognition: Face, Speech and Fusion*. VDM-Verlag.

Sanderson, C. and K. Paliwal (2002). "Polynomial Features for Robust Face Authentication". In: *International Conference on Image Processing*. Vol. 3, pp. 997–1000. DOI: 10.1109/ICIP.2002.1039143.

Shen, L. and L. Bai (2006). "A Review on Gabor Wavelets for Face Recognition". In: *Pattern Analysis and Applications* 9.2, pp. 273–292. DOI: 10.1007/s10044-006-0033-y.

Silsbee, P. L. (1994). "Sensory Integration in Audiovisual Automatic Speech Recognition". In: *Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*. Vol. 1, pp. 561–565. DOI: 10.1109/ACSSC.1994.471515.

Sumby, W. H. and I. Pollack (1954). "Visual Contribution to Speech Intelligibility in Noise". In: *Journal of the Acoustical Society of America* 26.2, pp. 212–215. DOI: 10.1121/1.1907309.

Summerfield, Q. (1987). "Some Preliminaries to a Comprehensive Account of Audio-Visual Speech Perception". In: *Hearing By Eye: The Psychology of Lip-Reading*. Ed. by B. Dodd and R. Campbell. London: Lawrence Erlbaum Associates.

Terry, L. and A. Katsaggelos (2008). "A Phone-Viseme Dynamic Bayesian Network for Audio-Visual Automatic Speech Recognition". In: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1–4. DOI: 10.1109/ICPR.2008.4761927.

Turk, M. A. and A. P. Pentland (1991). "Face Recognition using Eigenfaces". In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. CVPR '91, pp. 586–591. DOI: 10.1109/CVPR.1991.139758.

VoxForge (2008). *VoxForge Decision Tree*. URL: http://web.archive.org/web/20080829231414/http://www.voxforge.org/uploads/uD/LL/uDLLyMlja9EcLstgwFSMsA/tree.hed (visited on 05/31/2010).

Walden, B. E., R. A. Prosek, A. A. Montgomery, C. K. Scherr, and C. J. Jones (1977). "Effects of Training on the Visual Recognition of Consonants". In: *Journal of Speech and Hearing Research* 20.1, p. 130. PMID: 846196.

Walden, B. E., S. A. Erdman, A. A. Montgomery, D. M. Schwartz, and R. A. Prosek (1981). "Some Effects of Training on Speech Recognition by Hearing-Impaired Adults". In: *Journal of Speech, Language, and Hearing Research* 24.2, pp. 207–216. PMID: 7265936.

Williams, J. J., J. C. Rutledge, A. K. Katsaggelos, and D. C. Garstecki (1998). "Frame Rate and Viseme Analysis for Multimedia Applications to Assist Speechreading". In: *The Journal of VLSI Signal Processing* 20.1, pp. 7–23. DOI: 10.1023/A:1008062122135.

Wilson, P. I. and J. Fernandez (2006). "Facial Feature Detection using Haar Classifiers". In: *Journal of Computing Sciences in Colleges* 21.4, pp. 127–133.

Woodward, M. F. and C. G. Barber (1960). "Phoneme Perception in Lipreading". In: *Journal of Speech and Hearing Research* 3, p. 212. PMID: 13845910.

Young, S (2008). "HMMs and Related Speech Recognition Technologies". In: *Springer Handbook of Speech Processing*. Ed. by J. Benesty, M. M. Sondhi, and Y. Huang. Springer Berlin Heidelberg, pp. 539–558. DOI: 10.1007/978-3-540-49127-9_27.

Young, S, G Evermann, M Gales, T Hain, D Kershaw, G Moore, J Odell, D Ollason, D Povey, and V Valtchev (2005). *The HTK Book (for HTK Version 3.3)*. Cambridge University Engineering Department.

Young, S, G Evermann, M Gales, T Hain, D Kershaw, X Liu, G Moore, J Odell, D Ollason, and D Povey (2006). *The HTK Book (for HTK version 3.4)*. Cambridge University Engineering Department.

Appendix A

Neural Network Performance For Various Network Configurations

The neural network (see Chapter 3) performance was evaluated for a number of network configurations. The number of neurons in the first layer was tested for the range of 6 to 20 neurons. The second layer was tested for the range of 6 to 15 neurons. The third layer was fixed with a single output neuron, as network only requires a single output.

The network performance was evaluated for all combinations of neuron numbers. This was done using one frame for each subject in the CUAVE dataset. The performance was measured by calculating the mean squared error of lip and skin regions of the output only, with the manual labels used as the reference.

The results can be found on the attached CD in the file named: "`./Appendix A - neural network performance/mse-cuave.csv`". The file lists the MSE for each of the 36 subjects, for all 150 network configurations.

This data is used in Chapter 3: "Lip Pixel Classification", to empirically determine the best network configuration.

Appendix B

Corrected Labels For The CUAVE Dataset

The full set of corrected labels for the CUAVE dataset can be found on the attached CD, in the folder "`./Appendix B - corrected CUAVE labels/`". This data is used in Chapter 3: "Lip Pixel Classification".

Appendix C

Phoneme Trustworthiness

The tables in this appendix list the likelihood of each operation causing a phoneme to appear in the output. They also include the cumulative likelihood for any of the preceding operations causing the particular phoneme to appear in the output of the recogniser.

For example, the /dh/ phoneme lists "(ins)" as 14.9%, and /dh/ as 11.4%. This means there is a 14.9% chance that an insertion caused /dh/ to appear, and a 11.4% chance of /dh/ in the input causing /dh/ to appear in the output. The cumulative column shows there is a 26.3% chance of either /dh/ or an insertion causing /dh/ to appear in the output.

The full spreadsheet of phoneme trustworthiness can also be found on the attached CD, in the "`summary (rearranged)`" worksheet in the file: "`./Appendix C - phoneme trustworthiness/trustworthiness.xlsx`".

This data is used in Section 7.1: "Phoneme Trustworthiness".

Table C.1: Trustworthiness of the /dh/ and /ow/ phonemes

Cause of /dh/	Likelihood (%)	Cumulative (%)	Cause of /ow/	Likelihood (%)	Cumulative (%)
(ins)	14.9	14.9	ow	23.7	23.7
dh	11.4	26.3	(ins)	12.4	36.1
ih	4.9	31.1	iy	5.2	41.2
iy	4.7	35.9	r	5.2	46.4
ah	4.7	40.6	n	4.1	50.5
r	3.9	44.5	d	4.1	54.6
l	3.5	48.0	ih	3.6	58.2
er	3.5	51.6	ah	3.6	61.9
t	3.5	55.1	l	3.1	64.9
n	3.4	58.5	p	3.1	68.0
ae	3.1	61.6	k	3.1	71.1
s	3.0	64.5	s	2.6	73.7
m	2.6	67.1	ae	2.1	75.8
k	2.6	69.7	uw	2.1	77.8
eh	2.4	72.1	er	2.1	79.9
d	2.4	74.6	t	2.1	82.0
g	2.2	76.7	hh	1.5	83.5
ao	2.0	78.8	ao	1.5	85.1
z	2.0	80.8	b	1.5	86.6
aa	1.9	82.7	m	1.5	88.1
f	1.9	84.6	g	1.5	89.7
v	1.6	86.2	f	1.5	91.2
ay	1.5	87.7	aa	1.0	92.3
b	1.5	89.2	w	1.0	93.3
w	1.4	90.5	y	1.0	94.3
uw	1.4	91.9	jh	1.0	95.4
y	1.4	93.2	sh	1.0	96.4
sh	1.4	94.6	v	1.0	97.4
p	1.2	95.8	eh	0.5	97.9
ey	0.9	96.8	ey	0.5	98.5
hh	0.7	97.4	aw	0.5	99.0
ow	0.4	97.8	th	0.5	99.5
zh	0.4	98.2	ng	0.5	100.0
ng	0.4	98.6	ay	0.0	100.0
aw	0.3	98.9	oy	0.0	100.0
ch	0.3	99.2	uh	0.0	100.0
jh	0.3	99.5	z	0.0	100.0
th	0.3	99.7	ch	0.0	100.0
oy	0.1	99.9	zh	0.0	100.0
uh	0.1	100.0	dh	0.0	100.0

Table C.2: Trustworthiness of the /w/ and /eh/ phonemes

Cause of /w/	Likelihood (%)	Cumulative (%)	Cause of /eh/	Likelihood (%)	Cumulative (%)
w	24.3	24.3	eh	26.9	26.9
(ins)	14.2	38.5	(ins)	15.5	42.5
iy	6.3	44.8	ah	6.7	49.2
r	4.4	49.2	d	4.1	53.4
k	4.1	53.3	ih	3.6	57.0
ah	3.5	56.8	r	3.6	60.6
ih	3.2	59.9	s	3.6	64.2
l	3.2	63.1	l	3.1	67.4
er	3.2	66.2	t	3.1	70.5
t	3.2	69.4	ey	2.6	73.1
n	2.8	72.2	iy	2.1	75.1
p	2.5	74.8	w	2.1	77.2
g	2.5	77.3	m	2.1	79.3
eh	1.9	79.2	v	2.1	81.3
s	1.9	81.1	ao	1.6	82.9
dh	1.6	82.6	er	1.6	84.5
ae	1.3	83.9	n	1.6	86.0
ay	1.3	85.2	z	1.6	87.6
ey	1.3	86.4	k	1.6	89.1
aw	1.3	87.7	ay	1.0	90.2
m	1.3	89.0	uw	1.0	91.2
z	1.3	90.2	b	1.0	92.2
d	1.3	91.5	p	1.0	93.3
hh	0.9	92.4	g	1.0	94.3
ao	0.9	93.4	th	1.0	95.3
uh	0.9	94.3	aa	0.5	95.9
uw	0.9	95.3	ae	0.5	96.4
b	0.9	96.2	hh	0.5	96.9
aa	0.6	96.8	ow	0.5	97.4
sh	0.6	97.5	ch	0.5	97.9
ng	0.6	98.1	sh	0.5	98.4
y	0.3	98.4	zh	0.5	99.0
jh	0.3	98.7	dh	0.5	99.5
zh	0.3	99.1	f	0.5	100.0
th	0.3	99.4	aw	0.0	100.0
f	0.3	99.7	oy	0.0	100.0
v	0.3	100.0	uh	0.0	100.0
ow	0.0	100.0	y	0.0	100.0
oy	0.0	100.0	jh	0.0	100.0
ch	0.0	100.0	ng	0.0	100.0

Table C.3: Trustworthiness of the /hh/ and /s/ phonemes

Cause of /hh/	Likelihood (%)	Cumulative (%)	Cause of /s/	Likelihood (%)	Cumulative (%)
hh	32.9	32.9	s	33.0	33.0
(ins)	8.5	41.5	(ins)	9.1	42.1
ae	6.1	47.6	ih	4.5	46.6
r	6.1	53.7	iy	3.8	50.4
t	6.1	59.8	r	3.8	54.1
d	4.9	64.6	n	3.2	57.3
k	4.9	69.5	k	3.2	60.5
iy	3.7	73.2	er	2.7	63.2
l	3.7	76.8	ah	2.5	65.7
er	3.7	80.5	l	2.5	68.2
n	3.7	84.1	t	2.3	70.5
ah	2.4	86.6	d	2.1	72.7
y	2.4	89.0	g	2.1	74.8
ih	1.2	90.2	z	2.0	76.8
ay	1.2	91.5	aa	1.8	78.6
w	1.2	92.7	ay	1.8	80.4
p	1.2	93.9	ao	1.8	82.1
z	1.2	95.1	eh	1.6	83.8
s	1.2	96.3	m	1.6	85.4
ch	1.2	97.6	p	1.4	86.8
jh	1.2	98.8	ae	1.3	88.0
dh	1.2	100.0	ey	1.1	89.1
aa	0.0	100.0	w	1.1	90.2
eh	0.0	100.0	b	1.1	91.3
ey	0.0	100.0	ng	1.1	92.3
ao	0.0	100.0	uh	0.9	93.2
ow	0.0	100.0	uw	0.9	94.1
aw	0.0	100.0	sh	0.9	95.0
oy	0.0	100.0	dh	0.9	95.9
uh	0.0	100.0	f	0.9	96.8
uw	0.0	100.0	v	0.9	97.7
b	0.0	100.0	hh	0.7	98.4
m	0.0	100.0	ow	0.4	98.8
sh	0.0	100.0	y	0.4	99.1
zh	0.0	100.0	th	0.4	99.5
g	0.0	100.0	oy	0.2	99.6
th	0.0	100.0	ch	0.2	99.8
f	0.0	100.0	jh	0.2	100.0
v	0.0	100.0	aw	0.0	100.0
ng	0.0	100.0	zh	0.0	100.0

Table C.4: Trustworthiness of the /z/ and /r/ phonemes

Cause of /z/	Likelihood (%)	Cumulative (%)	Cause of /r/	Likelihood (%)	Cumulative (%)
z	33.1	33.1	r	34.4	34.4
(ins)	10.9	43.9	(ins)	8.3	42.6
t	8.8	52.7	s	5.0	47.7
d	5.0	57.7	ah	4.5	52.2
l	3.8	61.5	n	3.4	55.6
s	3.8	65.3	ae	3.2	58.8
er	3.3	68.6	t	3.2	62.1
iy	2.9	71.5	er	3.1	65.1
ih	2.5	74.1	m	3.1	68.2
k	2.5	76.6	l	2.9	71.0
ah	2.1	78.7	eh	2.0	73.0
n	2.1	80.8	b	2.0	75.0
eh	1.7	82.4	k	2.0	77.0
r	1.7	84.1	z	1.8	78.8
w	1.3	85.4	ih	1.6	80.4
uh	1.3	86.6	d	1.6	82.0
p	1.3	87.9	f	1.6	83.6
m	1.3	89.1	w	1.4	85.1
sh	1.3	90.4	ao	1.3	86.3
dh	1.3	91.6	y	1.3	87.6
g	1.3	92.9	iy	1.1	88.7
ae	0.8	93.7	hh	1.1	89.7
ay	0.8	94.6	dh	1.1	90.8
ow	0.8	95.4	ey	0.9	91.7
v	0.8	96.2	aw	0.9	92.6
ng	0.8	97.1	sh	0.9	93.5
hh	0.4	97.5	v	0.9	94.4
aw	0.4	97.9	uw	0.7	95.1
uw	0.4	98.3	p	0.7	95.9
y	0.4	98.7	aa	0.5	96.4
jh	0.4	99.2	ch	0.5	96.9
th	0.4	99.6	th	0.5	97.5
f	0.4	100.0	ay	0.4	97.8
aa	0.0	100.0	ow	0.4	98.2
ey	0.0	100.0	uh	0.4	98.6
ao	0.0	100.0	jh	0.4	98.9
oy	0.0	100.0	zh	0.4	99.3
b	0.0	100.0	g	0.4	99.6
ch	0.0	100.0	ng	0.4	100.0
zh	0.0	100.0	oy	0.0	100.0

Table C.5: Trustworthiness of the /ay/ and /l/ phonemes

Cause of /ay/	Likelihood (%)	Cumulative (%)	Cause of /l/	Likelihood (%)	Cumulative (%)
ay	34.9	34.9	l	36.1	36.1
(ins)	17.8	52.7	(ins)	13.8	49.9
ah	4.8	57.5	r	4.3	54.2
ih	4.1	61.6	t	4.3	58.4
iy	3.4	65.1	iy	2.6	61.0
l	2.7	67.8	ah	2.6	63.7
n	2.7	70.5	p	2.1	65.8
d	2.7	73.3	d	2.1	67.9
ae	2.1	75.3	ih	1.9	69.8
er	2.1	77.4	w	1.9	71.7
z	2.1	79.5	n	1.9	73.6
k	2.1	81.5	s	1.9	75.5
t	2.1	83.6	k	1.9	77.4
eh	1.4	84.9	ae	1.7	79.1
ey	1.4	86.3	f	1.7	80.8
r	1.4	87.7	aa	1.4	82.2
y	1.4	89.0	eh	1.4	83.6
p	1.4	90.4	hh	1.4	85.0
m	1.4	91.8	ao	1.4	86.5
aa	0.7	92.5	ow	1.4	87.9
w	0.7	93.2	b	1.2	89.1
aw	0.7	93.8	m	1.2	90.3
uw	0.7	94.5	dh	1.2	91.4
b	0.7	95.2	ay	1.0	92.4
jh	0.7	95.9	uw	1.0	93.3
sh	0.7	96.6	z	1.0	94.3
dh	0.7	97.3	sh	1.0	95.2
g	0.7	97.9	er	0.7	96.0
f	0.7	98.6	y	0.7	96.7
v	0.7	99.3	g	0.7	97.4
ng	0.7	100.0	ey	0.5	97.9
hh	0.0	100.0	ch	0.5	98.3
ao	0.0	100.0	jh	0.5	98.8
ow	0.0	100.0	th	0.5	99.3
oy	0.0	100.0	oy	0.2	99.5
uh	0.0	100.0	uh	0.2	99.8
s	0.0	100.0	ng	0.2	100.0
ch	0.0	100.0	aw	0.0	100.0
zh	0.0	100.0	zh	0.0	100.0
th	0.0	100.0	v	0.0	100.0

Table C.6: Trustworthiness of the /ih/ and /k/ phonemes

Cause of /ih/	Likelihood (%)	Cumulative (%)	Cause of /k/	Likelihood (%)	Cumulative (%)
ih	37.4	37.4	k	37.5	37.5
(ins)	12.5	49.9	(ins)	10.8	48.3
ah	3.5	53.4	er	3.4	51.7
er	3.3	56.7	t	3.4	55.2
r	2.9	59.5	l	3.0	58.2
s	2.9	62.4	r	3.0	61.2
iy	2.5	64.9	m	2.6	63.8
ae	2.3	67.1	n	2.6	66.4
eh	2.3	69.4	ih	2.2	68.5
t	2.3	71.7	ae	2.2	70.7
ao	2.1	73.7	ey	2.2	72.8
n	2.1	75.8	z	2.2	75.0
d	1.8	77.6	d	2.2	77.2
l	1.6	79.3	iy	1.7	78.9
p	1.6	80.9	aa	1.7	80.6
dh	1.6	82.5	ah	1.7	82.3
ow	1.4	84.0	s	1.7	84.1
sh	1.4	85.4	g	1.7	85.8
f	1.4	86.9	ao	1.3	87.1
uw	1.2	88.1	ow	1.3	88.4
y	1.2	89.3	w	1.3	89.7
k	1.2	90.6	f	1.3	90.9
aa	1.0	91.6	oy	0.9	91.8
ay	1.0	92.6	uw	0.9	92.7
uh	1.0	93.6	y	0.9	93.5
m	1.0	94.7	b	0.9	94.4
g	1.0	95.7	p	0.9	95.3
hh	0.8	96.5	dh	0.9	96.1
w	0.6	97.1	ng	0.9	97.0
oy	0.6	97.7	ay	0.4	97.4
z	0.4	98.2	eh	0.4	97.8
ch	0.4	98.6	hh	0.4	98.3
ng	0.4	99.0	jh	0.4	98.7
ey	0.2	99.2	sh	0.4	99.1
aw	0.2	99.4	th	0.4	99.6
b	0.2	99.6	v	0.4	100.0
jh	0.2	99.8	aw	0.0	100.0
v	0.2	100.0	uh	0.0	100.0
zh	0.0	100.0	ch	0.0	100.0
th	0.0	100.0	zh	0.0	100.0

Table C.7: Trustworthiness of the /ae/ and /aa/ phonemes

Cause of /ae/	Likelihood (%)	Cumulative (%)	Cause of /aa/	Likelihood (%)	Cumulative (%)
ae	38.3	38.3	aa	38.7	38.7
(ins)	11.2	49.5	(ins)	12.3	50.9
ih	4.2	53.7	ih	6.6	57.5
ah	3.7	57.5	n	4.7	62.3
l	2.3	59.8	iy	3.8	66.0
er	2.3	62.1	d	3.8	69.8
r	2.3	64.5	ah	2.8	72.6
z	2.3	66.8	l	2.8	75.5
k	2.3	69.2	t	2.8	78.3
g	2.3	71.5	g	2.8	81.1
v	2.3	73.8	eh	1.9	83.0
aa	1.9	75.7	w	1.9	84.9
m	1.9	77.6	r	1.9	86.8
s	1.9	79.4	m	1.9	88.7
d	1.9	81.3	z	1.9	90.6
t	1.9	83.2	f	1.9	92.5
eh	1.4	84.6	ae	0.9	93.4
ey	1.4	86.0	ey	0.9	94.3
hh	1.4	87.4	hh	0.9	95.3
ow	1.4	88.8	ow	0.9	96.2
uw	1.4	90.2	aw	0.9	97.2
b	1.4	91.6	er	0.9	98.1
dh	1.4	93.0	y	0.9	99.1
iy	0.9	93.9	k	0.9	100.0
ao	0.9	94.9	ay	0.0	100.0
w	0.9	95.8	ao	0.0	100.0
n	0.9	96.7	oy	0.0	100.0
f	0.9	97.7	uh	0.0	100.0
ay	0.5	98.1	uw	0.0	100.0
aw	0.5	98.6	b	0.0	100.0
p	0.5	99.1	p	0.0	100.0
ch	0.5	99.5	s	0.0	100.0
sh	0.5	100.0	ch	0.0	100.0
oy	0.0	100.0	jh	0.0	100.0
uh	0.0	100.0	sh	0.0	100.0
y	0.0	100.0	zh	0.0	100.0
jh	0.0	100.0	dh	0.0	100.0
zh	0.0	100.0	th	0.0	100.0
th	0.0	100.0	v	0.0	100.0
ng	0.0	100.0	ng	0.0	100.0

Table C.8: Trustworthiness of the /ah/ and /er/ phonemes

Cause of /ah/	Likelihood (%)	Cumulative (%)	Cause of /er/	Likelihood (%)	Cumulative (%)
ah	39.4	39.4	er	40.9	40.9
(ins)	10.5	49.9	(ins)	15.7	56.5
iy	3.6	53.5	s	5.2	61.7
t	3.6	57.1	n	4.3	66.1
ih	3.3	60.5	aa	3.5	69.6
l	3.1	63.5	ah	2.6	72.2
r	2.9	66.4	ao	2.6	74.8
n	2.9	69.3	l	2.6	77.4
ae	2.8	72.1	k	2.6	80.0
er	2.5	74.6	ih	1.7	81.7
d	2.1	76.7	eh	1.7	83.5
ao	1.9	78.6	uh	1.7	85.2
m	1.8	80.4	r	1.7	87.0
z	1.7	82.1	y	1.7	88.7
k	1.7	83.8	p	1.7	90.4
w	1.5	85.3	g	1.7	92.2
s	1.5	86.8	iy	0.9	93.0
ow	1.4	88.2	ey	0.9	93.9
ey	1.2	89.5	w	0.9	94.8
uw	1.2	90.7	oy	0.9	95.7
ay	1.0	91.7	uw	0.9	96.5
b	1.0	92.6	b	0.9	97.4
dh	1.0	93.6	z	0.9	98.3
aa	0.8	94.5	t	0.9	99.1
eh	0.7	95.1	dh	0.9	100.0
y	0.6	95.7	ae	0.0	100.0
p	0.6	96.3	ay	0.0	100.0
f	0.6	96.8	hh	0.0	100.0
v	0.6	97.4	ow	0.0	100.0
ng	0.6	97.9	aw	0.0	100.0
hh	0.4	98.3	m	0.0	100.0
th	0.4	98.8	ch	0.0	100.0
oy	0.3	99.0	jh	0.0	100.0
ch	0.3	99.3	sh	0.0	100.0
sh	0.3	99.6	zh	0.0	100.0
aw	0.1	99.7	d	0.0	100.0
uh	0.1	99.9	th	0.0	100.0
zh	0.1	100.0	f	0.0	100.0
jh	0.0	100.0	v	0.0	100.0
g	0.0	100.0	ng	0.0	100.0

Table C.9: Trustworthiness of the /iy/ and /n/ phonemes

Cause of /iy/	Likelihood (%)	Cumulative (%)	Cause of /n/	Likelihood (%)	Cumulative (%)
iy	41.0	41.0	n	44.7	44.7
(ins)	10.7	51.7	(ins)	9.3	53.9
ah	3.8	55.4	l	4.9	58.8
k	3.6	59.0	k	3.7	62.6
l	3.2	62.2	z	2.5	65.0
n	2.8	65.0	ih	2.3	67.3
t	2.8	67.7	ae	2.3	69.5
er	2.4	70.1	r	2.3	71.8
ih	2.2	72.3	d	2.3	74.1
aa	2.2	74.5	t	2.3	76.3
s	2.0	76.4	iy	2.1	78.4
z	1.8	78.2	er	2.1	80.5
ae	1.6	79.8	ao	1.6	82.1
uw	1.6	81.4	ah	1.4	83.5
p	1.6	83.0	b	1.4	85.0
eh	1.4	84.4	eh	1.2	86.2
d	1.4	85.7	y	1.2	87.4
f	1.4	87.1	m	1.2	88.7
ay	1.2	88.3	s	1.2	89.9
r	1.2	89.5	dh	1.2	91.2
m	1.2	90.7	aa	1.0	92.2
g	1.2	91.9	w	1.0	93.2
b	1.0	92.9	uw	1.0	94.2
ao	0.8	93.7	ay	0.8	95.1
y	0.8	94.5	ey	0.8	95.9
ey	0.6	95.0	ow	0.6	96.5
ow	0.6	95.6	sh	0.6	97.1
w	0.6	96.2	p	0.4	97.5
uh	0.6	96.8	g	0.4	97.9
sh	0.6	97.4	f	0.4	98.4
th	0.6	98.0	v	0.4	98.8
aw	0.4	98.4	hh	0.2	99.0
jh	0.4	98.8	oy	0.2	99.2
hh	0.2	99.0	uh	0.2	99.4
oy	0.2	99.2	ch	0.2	99.6
ch	0.2	99.4	zh	0.2	99.8
dh	0.2	99.6	th	0.2	100.0
v	0.2	99.8	aw	0.0	100.0
ng	0.2	100.0	jh	0.0	100.0
zh	0.0	100.0	ng	0.0	100.0

Table C.10: Trustworthiness of the /d/ and /ao/ phonemes

Cause of /d/	Likelihood (%)	Cumulative (%)	Cause of /ao/	Likelihood (%)	Cumulative (%)
d	45.3	45.3	ao	45.6	45.6
(ins)	10.1	55.4	iy	4.4	50.0
s	5.0	60.4	ah	4.4	54.4
ih	3.6	64.0	(ins)	4.4	58.8
iy	2.9	66.9	er	3.5	62.3
ah	2.9	69.8	d	3.5	65.8
l	2.9	72.7	k	3.5	69.3
n	2.9	75.5	t	3.5	72.8
t	2.9	78.4	l	2.6	75.4
er	2.2	80.6	b	2.6	78.1
r	2.2	82.7	n	2.6	80.7
aa	1.4	84.2	w	1.8	82.5
ae	1.4	85.6	r	1.8	84.2
eh	1.4	87.1	p	1.8	86.0
ey	1.4	88.5	v	1.8	87.7
uw	1.4	89.9	ih	0.9	88.6
y	1.4	91.4	ae	0.9	89.5
ch	1.4	92.8	eh	0.9	90.4
sh	1.4	94.2	ey	0.9	91.2
hh	0.7	95.0	ow	0.9	92.1
ow	0.7	95.7	aw	0.9	93.0
p	0.7	96.4	oy	0.9	93.9
m	0.7	97.1	uh	0.9	94.7
z	0.7	97.8	m	0.9	95.6
k	0.7	98.6	z	0.9	96.5
dh	0.7	99.3	s	0.9	97.4
f	0.7	100.0	dh	0.9	98.2
ay	0.0	100.0	g	0.9	99.1
ao	0.0	100.0	ng	0.9	100.0
w	0.0	100.0	aa	0.0	100.0
aw	0.0	100.0	ay	0.0	100.0
oy	0.0	100.0	hh	0.0	100.0
uh	0.0	100.0	uw	0.0	100.0
b	0.0	100.0	y	0.0	100.0
jh	0.0	100.0	ch	0.0	100.0
zh	0.0	100.0	jh	0.0	100.0
g	0.0	100.0	sh	0.0	100.0
th	0.0	100.0	zh	0.0	100.0
v	0.0	100.0	th	0.0	100.0
ng	0.0	100.0	f	0.0	100.0

Table C.11: Trustworthiness of the /t/ and /m/ phonemes

Cause of /t/	Likelihood (%)	Cumulative (%)	Cause of /m/	Likelihood (%)	Cumulative (%)
t	46.7	46.7	m	47.9	47.9
(ins)	8.8	55.5	(ins)	8.2	56.2
iy	3.2	58.7	n	4.1	60.3
r	2.7	61.4	l	2.7	63.0
l	2.4	63.8	r	2.7	65.8
s	2.4	66.3	d	2.7	68.5
d	2.4	68.7	k	2.7	71.2
ih	2.2	70.9	ih	2.1	73.3
eh	2.0	72.9	iy	2.1	75.3
ah	1.7	74.6	ah	2.1	77.4
y	1.7	76.3	w	2.1	79.5
m	1.7	78.0	er	2.1	81.5
g	1.7	79.7	t	2.1	83.6
p	1.5	81.2	eh	1.4	84.9
k	1.5	82.6	uw	1.4	86.3
f	1.5	84.1	z	1.4	87.7
er	1.2	85.3	sh	1.4	89.0
n	1.2	86.6	th	1.4	90.4
z	1.2	87.8	aa	0.7	91.1
dh	1.2	89.0	ae	0.7	91.8
ae	1.0	90.0	ey	0.7	92.5
ao	1.0	91.0	hh	0.7	93.2
uw	1.0	91.9	ao	0.7	93.8
ch	1.0	92.9	oy	0.7	94.5
sh	1.0	93.9	y	0.7	95.2
hh	0.7	94.6	b	0.7	95.9
v	0.7	95.4	p	0.7	96.6
aa	0.5	95.8	s	0.7	97.3
ay	0.5	96.3	dh	0.7	97.9
ey	0.5	96.8	g	0.7	98.6
ow	0.5	97.3	f	0.7	99.3
w	0.5	97.8	v	0.7	100.0
b	0.5	98.3	ay	0.0	100.0
jh	0.5	98.8	ow	0.0	100.0
ng	0.5	99.3	aw	0.0	100.0
oy	0.2	99.5	uh	0.0	100.0
uh	0.2	99.8	ch	0.0	100.0
zh	0.2	100.0	jh	0.0	100.0
aw	0.0	100.0	zh	0.0	100.0
th	0.0	100.0	ng	0.0	100.0

Table C.12: Trustworthiness of the /b/ and /y/ phonemes

Cause of /b/	Likelihood (%)	Cumulative (%)	Cause of /y/	Likelihood (%)	Cumulative (%)
b	48.1	48.1	y	51.9	51.9
eh	5.8	53.8	(ins)	7.7	59.6
n	5.8	59.6	n	5.8	65.4
d	5.8	65.4	s	5.8	71.2
iy	3.8	69.2	f	5.8	76.9
ao	3.8	73.1	uw	3.8	80.8
y	3.8	76.9	r	3.8	84.6
(ins)	3.8	80.8	t	3.8	88.5
aa	1.9	82.7	ih	1.9	90.4
ae	1.9	84.6	ah	1.9	92.3
w	1.9	86.5	eh	1.9	94.2
p	1.9	88.5	k	1.9	96.2
zh	1.9	90.4	v	1.9	98.1
k	1.9	92.3	ng	1.9	100.0
t	1.9	94.2	iy	0.0	100.0
dh	1.9	96.2	aa	0.0	100.0
g	1.9	98.1	ae	0.0	100.0
v	1.9	100.0	ay	0.0	100.0
ih	0.0	100.0	ey	0.0	100.0
ah	0.0	100.0	hh	0.0	100.0
ay	0.0	100.0	ao	0.0	100.0
ey	0.0	100.0	ow	0.0	100.0
hh	0.0	100.0	w	0.0	100.0
ow	0.0	100.0	aw	0.0	100.0
aw	0.0	100.0	oy	0.0	100.0
oy	0.0	100.0	uh	0.0	100.0
uh	0.0	100.0	l	0.0	100.0
uw	0.0	100.0	er	0.0	100.0
l	0.0	100.0	b	0.0	100.0
er	0.0	100.0	p	0.0	100.0
r	0.0	100.0	m	0.0	100.0
m	0.0	100.0	z	0.0	100.0
z	0.0	100.0	ch	0.0	100.0
s	0.0	100.0	jh	0.0	100.0
ch	0.0	100.0	sh	0.0	100.0
jh	0.0	100.0	zh	0.0	100.0
sh	0.0	100.0	d	0.0	100.0
th	0.0	100.0	dh	0.0	100.0
f	0.0	100.0	g	0.0	100.0
ng	0.0	100.0	th	0.0	100.0

Table C.13: Trustworthiness of the /ey/ and /v/ phonemes

Cause of /ey/	Likelihood (%)	Cumulative (%)	Cause of /v/	Likelihood (%)	Cumulative (%)
ey	53.3	53.3	v	63.0	63.0
(ins)	11.7	65.0	(ins)	15.2	78.3
iy	6.7	71.7	k	4.3	82.6
ae	3.3	75.0	t	4.3	87.0
ao	3.3	78.3	aa	2.2	89.1
m	3.3	81.7	ay	2.2	91.3
dh	3.3	85.0	eh	2.2	93.5
aw	1.7	86.7	r	2.2	95.7
uw	1.7	88.3	m	2.2	97.8
l	1.7	90.0	n	2.2	100.0
er	1.7	91.7	ih	0.0	100.0
r	1.7	93.3	iy	0.0	100.0
p	1.7	95.0	ah	0.0	100.0
s	1.7	96.7	ae	0.0	100.0
d	1.7	98.3	ey	0.0	100.0
ng	1.7	100.0	hh	0.0	100.0
ih	0.0	100.0	ao	0.0	100.0
aa	0.0	100.0	ow	0.0	100.0
ah	0.0	100.0	w	0.0	100.0
ay	0.0	100.0	aw	0.0	100.0
eh	0.0	100.0	oy	0.0	100.0
hh	0.0	100.0	uh	0.0	100.0
ow	0.0	100.0	uw	0.0	100.0
w	0.0	100.0	l	0.0	100.0
oy	0.0	100.0	er	0.0	100.0
uh	0.0	100.0	y	0.0	100.0
y	0.0	100.0	b	0.0	100.0
b	0.0	100.0	p	0.0	100.0
n	0.0	100.0	z	0.0	100.0
z	0.0	100.0	s	0.0	100.0
ch	0.0	100.0	ch	0.0	100.0
jh	0.0	100.0	jh	0.0	100.0
sh	0.0	100.0	sh	0.0	100.0
zh	0.0	100.0	zh	0.0	100.0
k	0.0	100.0	d	0.0	100.0
t	0.0	100.0	dh	0.0	100.0
g	0.0	100.0	g	0.0	100.0
th	0.0	100.0	th	0.0	100.0
f	0.0	100.0	f	0.0	100.0
v	0.0	100.0	ng	0.0	100.0

Table C.14: Trustworthiness of the /aw/ and /p/ phonemes

Cause of /aw/	Likelihood (%)	Cumulative (%)	Cause of /p/	Likelihood (%)	Cumulative (%)
aw	70.6	70.6	p	73.3	73.3
f	11.8	82.4	t	6.7	80.0
ih	5.9	88.2	(ins)	6.7	86.7
hh	5.9	94.1	aa	3.3	90.0
sh	5.9	100.0	ow	3.3	93.3
iy	0.0	100.0	m	3.3	96.7
aa	0.0	100.0	k	3.3	100.0
ah	0.0	100.0	ih	0.0	100.0
ae	0.0	100.0	iy	0.0	100.0
ay	0.0	100.0	ah	0.0	100.0
eh	0.0	100.0	ae	0.0	100.0
ey	0.0	100.0	ay	0.0	100.0
ao	0.0	100.0	eh	0.0	100.0
ow	0.0	100.0	ey	0.0	100.0
w	0.0	100.0	hh	0.0	100.0
oy	0.0	100.0	ao	0.0	100.0
uh	0.0	100.0	w	0.0	100.0
uw	0.0	100.0	aw	0.0	100.0
l	0.0	100.0	oy	0.0	100.0
er	0.0	100.0	uh	0.0	100.0
r	0.0	100.0	uw	0.0	100.0
y	0.0	100.0	l	0.0	100.0
b	0.0	100.0	er	0.0	100.0
p	0.0	100.0	r	0.0	100.0
m	0.0	100.0	y	0.0	100.0
n	0.0	100.0	b	0.0	100.0
z	0.0	100.0	n	0.0	100.0
s	0.0	100.0	z	0.0	100.0
ch	0.0	100.0	s	0.0	100.0
jh	0.0	100.0	ch	0.0	100.0
zh	0.0	100.0	jh	0.0	100.0
d	0.0	100.0	sh	0.0	100.0
k	0.0	100.0	zh	0.0	100.0
t	0.0	100.0	d	0.0	100.0
dh	0.0	100.0	dh	0.0	100.0
g	0.0	100.0	g	0.0	100.0
th	0.0	100.0	th	0.0	100.0
v	0.0	100.0	f	0.0	100.0
ng	0.0	100.0	v	0.0	100.0
(ins)	0.0	100.0	ng	0.0	100.0

Table C.15: Trustworthiness of the /f/ and /g/ phonemes

Cause of /f/	Likelihood (%)	Cumulative (%)	Cause of /g/	Likelihood (%)	Cumulative (%)
f	75.9	75.9	g	79.2	79.2
(ins)	10.3	86.2	ey	8.3	87.5
ah	3.4	89.7	w	4.2	91.7
eh	3.4	93.1	f	4.2	95.8
r	3.4	96.6	(ins)	4.2	100.0
p	3.4	100.0	ih	0.0	100.0
ih	0.0	100.0	iy	0.0	100.0
iy	0.0	100.0	aa	0.0	100.0
aa	0.0	100.0	ah	0.0	100.0
ae	0.0	100.0	ae	0.0	100.0
ay	0.0	100.0	ay	0.0	100.0
ey	0.0	100.0	eh	0.0	100.0
hh	0.0	100.0	hh	0.0	100.0
ao	0.0	100.0	ao	0.0	100.0
ow	0.0	100.0	ow	0.0	100.0
w	0.0	100.0	aw	0.0	100.0
aw	0.0	100.0	oy	0.0	100.0
oy	0.0	100.0	uh	0.0	100.0
uh	0.0	100.0	uw	0.0	100.0
uw	0.0	100.0	l	0.0	100.0
l	0.0	100.0	er	0.0	100.0
er	0.0	100.0	r	0.0	100.0
y	0.0	100.0	y	0.0	100.0
b	0.0	100.0	b	0.0	100.0
m	0.0	100.0	p	0.0	100.0
n	0.0	100.0	m	0.0	100.0
z	0.0	100.0	n	0.0	100.0
s	0.0	100.0	z	0.0	100.0
ch	0.0	100.0	s	0.0	100.0
jh	0.0	100.0	ch	0.0	100.0
sh	0.0	100.0	jh	0.0	100.0
zh	0.0	100.0	sh	0.0	100.0
d	0.0	100.0	zh	0.0	100.0
k	0.0	100.0	d	0.0	100.0
t	0.0	100.0	k	0.0	100.0
dh	0.0	100.0	t	0.0	100.0
g	0.0	100.0	dh	0.0	100.0
th	0.0	100.0	th	0.0	100.0
v	0.0	100.0	v	0.0	100.0
ng	0.0	100.0	ng	0.0	100.0

Table C.16: Trustworthiness of the /zh/ and /jh/ phonemes

Cause of /zh/	Likelihood (%)	Cumulative (%)	Cause of /jh/	Likelihood (%)	Cumulative (%)
zh	83.3	83.3	jh	90.9	90.9
p	16.7	100.0	t	9.1	100.0
ih	0.0	100.0	ih	0.0	100.0
iy	0.0	100.0	iy	0.0	100.0
aa	0.0	100.0	aa	0.0	100.0
ah	0.0	100.0	ah	0.0	100.0
ae	0.0	100.0	ae	0.0	100.0
ay	0.0	100.0	ay	0.0	100.0
eh	0.0	100.0	eh	0.0	100.0
ey	0.0	100.0	ey	0.0	100.0
hh	0.0	100.0	hh	0.0	100.0
ao	0.0	100.0	ao	0.0	100.0
ow	0.0	100.0	ow	0.0	100.0
w	0.0	100.0	w	0.0	100.0
aw	0.0	100.0	aw	0.0	100.0
oy	0.0	100.0	oy	0.0	100.0
uh	0.0	100.0	uh	0.0	100.0
uw	0.0	100.0	uw	0.0	100.0
l	0.0	100.0	l	0.0	100.0
er	0.0	100.0	er	0.0	100.0
r	0.0	100.0	r	0.0	100.0
y	0.0	100.0	y	0.0	100.0
b	0.0	100.0	b	0.0	100.0
m	0.0	100.0	p	0.0	100.0
n	0.0	100.0	m	0.0	100.0
z	0.0	100.0	n	0.0	100.0
s	0.0	100.0	z	0.0	100.0
ch	0.0	100.0	s	0.0	100.0
jh	0.0	100.0	ch	0.0	100.0
sh	0.0	100.0	sh	0.0	100.0
d	0.0	100.0	zh	0.0	100.0
k	0.0	100.0	d	0.0	100.0
t	0.0	100.0	k	0.0	100.0
dh	0.0	100.0	dh	0.0	100.0
g	0.0	100.0	g	0.0	100.0
th	0.0	100.0	th	0.0	100.0
f	0.0	100.0	f	0.0	100.0
v	0.0	100.0	v	0.0	100.0
ng	0.0	100.0	ng	0.0	100.0
(ins)	0.0	100.0	(ins)	0.0	100.0

Table C.17: Trustworthiness of the /sh/ and /uw/ phonemes

Cause of /sh/	Likelihood (%)	Cumulative (%)	Cause of /uw/	Likelihood (%)	Cumulative (%)
sh	91.3	91.3	uw	95.0	95.0
(ins)	8.7	100.0	g	5.0	100.0
ih	0.0	100.0	ih	0.0	100.0
iy	0.0	100.0	iy	0.0	100.0
aa	0.0	100.0	aa	0.0	100.0
ah	0.0	100.0	ah	0.0	100.0
ae	0.0	100.0	ae	0.0	100.0
ay	0.0	100.0	ay	0.0	100.0
eh	0.0	100.0	eh	0.0	100.0
ey	0.0	100.0	ey	0.0	100.0
hh	0.0	100.0	hh	0.0	100.0
ao	0.0	100.0	ao	0.0	100.0
ow	0.0	100.0	ow	0.0	100.0
w	0.0	100.0	w	0.0	100.0
aw	0.0	100.0	aw	0.0	100.0
oy	0.0	100.0	oy	0.0	100.0
uh	0.0	100.0	uh	0.0	100.0
uw	0.0	100.0	l	0.0	100.0
l	0.0	100.0	er	0.0	100.0
er	0.0	100.0	r	0.0	100.0
r	0.0	100.0	y	0.0	100.0
y	0.0	100.0	b	0.0	100.0
b	0.0	100.0	p	0.0	100.0
p	0.0	100.0	m	0.0	100.0
m	0.0	100.0	n	0.0	100.0
n	0.0	100.0	z	0.0	100.0
z	0.0	100.0	s	0.0	100.0
s	0.0	100.0	ch	0.0	100.0
ch	0.0	100.0	jh	0.0	100.0
jh	0.0	100.0	sh	0.0	100.0
zh	0.0	100.0	zh	0.0	100.0
d	0.0	100.0	d	0.0	100.0
k	0.0	100.0	k	0.0	100.0
t	0.0	100.0	t	0.0	100.0
dh	0.0	100.0	dh	0.0	100.0
g	0.0	100.0	th	0.0	100.0
th	0.0	100.0	f	0.0	100.0
f	0.0	100.0	v	0.0	100.0
v	0.0	100.0	ng	0.0	100.0
ng	0.0	100.0	(ins)	0.0	100.0

Table C.18: Trustworthiness of the /oy/ and /uh/ phonemes

Cause of /oy/	Likelihood (%)	Cumulative (%)	Cause of /uh/	Likelihood (%)	Cumulative (%)
oy	100.0	100.0	uh	100.0	100.0
ih	0.0	100.0	ih	0.0	100.0
iy	0.0	100.0	iy	0.0	100.0
aa	0.0	100.0	aa	0.0	100.0
ah	0.0	100.0	ah	0.0	100.0
ae	0.0	100.0	ae	0.0	100.0
ay	0.0	100.0	ay	0.0	100.0
eh	0.0	100.0	eh	0.0	100.0
ey	0.0	100.0	ey	0.0	100.0
hh	0.0	100.0	hh	0.0	100.0
ao	0.0	100.0	ao	0.0	100.0
ow	0.0	100.0	ow	0.0	100.0
w	0.0	100.0	w	0.0	100.0
aw	0.0	100.0	aw	0.0	100.0
uh	0.0	100.0	oy	0.0	100.0
uw	0.0	100.0	uw	0.0	100.0
l	0.0	100.0	l	0.0	100.0
er	0.0	100.0	er	0.0	100.0
r	0.0	100.0	r	0.0	100.0
y	0.0	100.0	y	0.0	100.0
b	0.0	100.0	b	0.0	100.0
p	0.0	100.0	p	0.0	100.0
m	0.0	100.0	m	0.0	100.0
n	0.0	100.0	n	0.0	100.0
z	0.0	100.0	z	0.0	100.0
s	0.0	100.0	s	0.0	100.0
ch	0.0	100.0	ch	0.0	100.0
jh	0.0	100.0	jh	0.0	100.0
sh	0.0	100.0	sh	0.0	100.0
zh	0.0	100.0	zh	0.0	100.0
d	0.0	100.0	d	0.0	100.0
k	0.0	100.0	k	0.0	100.0
t	0.0	100.0	t	0.0	100.0
dh	0.0	100.0	dh	0.0	100.0
g	0.0	100.0	g	0.0	100.0
th	0.0	100.0	th	0.0	100.0
f	0.0	100.0	f	0.0	100.0
v	0.0	100.0	v	0.0	100.0
ng	0.0	100.0	ng	0.0	100.0
(ins)	0.0	100.0	(ins)	0.0	100.0

Table C.19: Trustworthiness of the /ch/ and /th/ phonemes

Cause of /ch/	Likelihood (%)	Cumulative (%)	Cause of /th/	Likelihood (%)	Cumulative (%)
ch	100.0	100.0	th	100.0	100.0
ih	0.0	100.0	ih	0.0	100.0
iy	0.0	100.0	iy	0.0	100.0
aa	0.0	100.0	aa	0.0	100.0
ah	0.0	100.0	ah	0.0	100.0
ae	0.0	100.0	ae	0.0	100.0
ay	0.0	100.0	ay	0.0	100.0
eh	0.0	100.0	eh	0.0	100.0
ey	0.0	100.0	ey	0.0	100.0
hh	0.0	100.0	hh	0.0	100.0
ao	0.0	100.0	ao	0.0	100.0
ow	0.0	100.0	ow	0.0	100.0
w	0.0	100.0	w	0.0	100.0
aw	0.0	100.0	aw	0.0	100.0
oy	0.0	100.0	oy	0.0	100.0
uh	0.0	100.0	uh	0.0	100.0
uw	0.0	100.0	uw	0.0	100.0
l	0.0	100.0	l	0.0	100.0
er	0.0	100.0	er	0.0	100.0
r	0.0	100.0	r	0.0	100.0
y	0.0	100.0	y	0.0	100.0
b	0.0	100.0	b	0.0	100.0
p	0.0	100.0	p	0.0	100.0
m	0.0	100.0	m	0.0	100.0
n	0.0	100.0	n	0.0	100.0
z	0.0	100.0	z	0.0	100.0
s	0.0	100.0	s	0.0	100.0
jh	0.0	100.0	ch	0.0	100.0
sh	0.0	100.0	jh	0.0	100.0
zh	0.0	100.0	sh	0.0	100.0
d	0.0	100.0	zh	0.0	100.0
k	0.0	100.0	d	0.0	100.0
t	0.0	100.0	k	0.0	100.0
dh	0.0	100.0	t	0.0	100.0
g	0.0	100.0	dh	0.0	100.0
th	0.0	100.0	g	0.0	100.0
f	0.0	100.0	f	0.0	100.0
v	0.0	100.0	v	0.0	100.0
ng	0.0	100.0	ng	0.0	100.0
(ins)	0.0	100.0	(ins)	0.0	100.0

Table C.20: Trustworthiness of the /ng/ phoneme

Cause of /ng/	Likelihood (%)	Cumulative (%)
ng	100.0	100.0
ih	0.0	100.0
iy	0.0	100.0
aa	0.0	100.0
ah	0.0	100.0
ae	0.0	100.0
ay	0.0	100.0
eh	0.0	100.0
ey	0.0	100.0
hh	0.0	100.0
ao	0.0	100.0
ow	0.0	100.0
w	0.0	100.0
aw	0.0	100.0
oy	0.0	100.0
uh	0.0	100.0
uw	0.0	100.0
l	0.0	100.0
er	0.0	100.0
r	0.0	100.0
y	0.0	100.0
b	0.0	100.0
p	0.0	100.0
m	0.0	100.0
n	0.0	100.0
z	0.0	100.0
s	0.0	100.0
ch	0.0	100.0
jh	0.0	100.0
sh	0.0	100.0
zh	0.0	100.0
d	0.0	100.0
k	0.0	100.0
t	0.0	100.0
dh	0.0	100.0
g	0.0	100.0
th	0.0	100.0
f	0.0	100.0
v	0.0	100.0
(ins)	0.0	100.0

An electronic copy of this thesis, and all supplementary material for the appendices, can be found on the attached CD.

They are also available online at www.mramage.id.au/phd