Features for Audio-Visual Speech Recognition

Iain Matthews

School of Information Systems
University of East Anglia

September 1998

Double-sided final version.

Abstract

Human speech perception considers both the auditory and visual nature of speech. Speech is more intelligible if the face of the talker can be seen and this is especially so in noisy conditions. More robust automatic speech recognition is possible if visual speech cues can be integrated with traditional acoustic systems.

This thesis discusses the problems of visual speech parameterisation from mouth image sequences for use in audio-visual speech recognition. Five new lipreading techniques are evaluated using a hidden Markov model based visual-only recognition task and compared with an enhanced implementation of a previous lip contour tracker.

The best methods are tested on two different multi-talker audio-visual databases to compare performance across different tasks. Combined audio-visual performance is tested using both early and late integration schemes.

The addition of visual information to automatic speech recognition is found to improve accuracy and this is most pronounced in acoustically noisy conditions.

Real-time implementations of two of the proposed methods demonstrate that the extension to audio-visual speech recognition is not impractical using current desktop technology.

Contents

List of Figures

List of Tables

Statement of Originality

Unless otherwise noted or referenced in the text the work described in this thesis is that of the author. Two sections in particular should be highlighted,

1. Section 5.2, on the use of Active Appearance Models (AAM's) for lipreading, is the result of collaborative work with Dr T. Cootes at the Wolfson Image Analysis Unit, University of Manchester.

2. Section 8.4, on the late integration of audio and visual features using an entropy derived confidence measure, was conducted by my supervisor Dr S. Cox using the visual features derived in this thesis.

Acknowledgements

Many thanks to my supervisors, Prof Andrew Bangham and Dr Stephen Cox. Firstly, for an interesting and challenging thesis topic, and secondly, for the guidance, facilities and finance they provided. Special thanks to Dr Richard Harvey for the many discussions, ideas and solutions.

I am especially indebted to the ten subjects of the AVletters database who all gave their time freely, thank you; Anya, Bill, Faye, John, Kate, Nicola, Stephen, Steve, Verity and Yi. Thanks also to the staff at the audio-visual centre for their help during and after the recording session.

Thank you to Shaun McCullagh and the rest of the SYS support staff for, despite my best efforts, keeping it all running.

Thanks to Scott and Emma for enduring the restrained recording session and to John Stocking for constructing it, and several other devices of humiliation.

Kudos to the cast and crew of S2.27 and S2.28 past and present.

This work was funded by the Engineering and Physical Sciences Research Council.

Glossary

AAM	active appearance model
ASM	active shape model
ASR	automatic speech recognition
AVSR	auditory-visual speech recognition
DTW	dynamic time warping
GLDM	grey level profile distribution model
fps	frames per second
HTK	a hidden Markov model toolkit, distributed by Entropic
HMM	hidden Markov model
IPA	International Phonetics Association
MFCC	Mel frequency cepstral coefficient
MSA	multiscale spatial analysis
PCA	principal component analysis
pdf	probability density function
PDM	point distribution model
SNR	signal to noise ratio
RP	received pronunciation

They went to sea in a Sieve, they did,
In a Sieve they went to sea:

In spite of all their friends could say,
On a winter's morn, on a stormy day,
In a Sieve they went to sea!

And when the Sieve turned round and round,
And every one cried, 'You'll all be drowned!'
They called aloud, 'Our Sieve ain't big,
But we don't care a button! We don't care a fig!
In a Sieve we'll go to sea!'

Far and few, far and few,
Are the lands where the Jumblies live;
Their heads are green, and their hands are blue,
And they went to sea in a Sieve.

The Jumblies — Edward Lear

Chapter 1

Introduction

Many people with a hearing impairment can understand fluent speech by *lipreading*, a fact that demonstrates that linguistic information can be conveyed by vision. In the 1950's it was shown that vision has a more fundamental role in speech perception. It was found that even normal-hearing individuals are able to make use of the visual speech cues available. The presentation of bimodal (auditory and visual) speech was found to be more intelligible than speech sound alone. The improvement obtained with the addition of vision is most distinct when the auditory signal is degraded with, for example, background noise or speech from nearby talkers. In these conditions speech intelligibility is significantly degraded if the face of the talker is hidden. An example is the 'cocktail party' effect: in a room full of people talking, it is easier to understand the person you are looking at. Psychological studies indicate that speech perception is a fundamentally bimodal process that integrates auditory and visual cues.

A distinction needs to be made between auditory, visual and auditory-visual speech. Summerfield attempts to clarify this in [184],

> *Lipreading* is the perception of speech purely visually by observing the talker articulatory gestures. *Audio-visual* speech perception is the perception of speech by combining lipreading with audition. *Speechreading* embraces a larger set of activities. It is the understanding of speech by observing the talker's articulation and facial and manual gestures, and may also include audition.

The psychological findings suggest that automatic speech recognition might also be made more robust if visual speech information could be incorporated. Robust automatic speech recognition has been an engineering goal for a long time. Statistical modelling methods have produced systems with useable performance in constrained tasks, such as using restricted vocabularies or limiting the number of talkers. A significant problem is maintaining recognition accuracy in unpredictable and noisy environments. Current commercial dictation systems (from companies such as Dragon, IBM, Phillips and Kurzweil) employ close fitting noise-reducing microphones. All of these systems incorporate elementary grammar, using basic language rules to resolve illegal utterances caused by misclassification at the phoneme or word level. However, they all ignore the auditory-visual nature of speech. A step closer to human speech perception can be made by adding vision to speech recognition. This provides an additional, and often complementary source of speech information, for example [m] and [n] sound similar but look different.

Motivated by the possibility of more robust automatic speech recognition there have been several machine speechreading systems that combine audio and visual speech features. Indeed, the idea of lipreading computers is not new. Arthur C. Clarke's 1968 novel 2001: A

Space Odyssey famously includes the lipreading computer HAL 9000[1]. For all such systems, the goal of audio-visual speech recognition is to improve recognition accuracy, especially in noisy conditions. However, the two problems of visual feature extraction and audio-visual integration are non-trivial. Audio-visual speech recognition is a task that combines the disciplines of speech recognition, computer vision and data fusion.

1.1 Visual Speech Analysis

This thesis is mainly concerned with visual feature extraction. There have been two main approaches to this problem; a low-level analysis of image sequence data which does not attempt to incorporate much prior knowledge; or a high-level approach that imposes a model on the data using prior knowledge. Typically high-level analysis has used lip tracking to extract lip shape information alone. Low-level analysis directly processes image pixels and is implicitly able to retrieve additional features, that may be difficult to track, such as the teeth and tongue.

This thesis considers five new lipreading methods. Four are strictly low-level analyses that form features using a scale-based transformation of each mouth image. The fifth combines a high-level model-based lip shape analysis with pixel-based analysis of appearance. All of these are compared with an new implementation of a high-level lip contour tracker that has previously been used for lipreading [107].

Although many methods have been used for visual feature extraction, they are not usually compared on the same task. This thesis compares all six different methods on the same recognition task. The most successful are further tested using an additional isolated digits databases to evaluate robustness to the application.

All of the methods described extract information from the mouth area only. Although there is evidence that speech information is available from other areas of the face, and expression and gesture are likely to be useful in some situations, the problem of finding and locating the face and mouth area is not covered. Automatic face location and tracking have already been extensively discussed in, for example [15, 44, 49, 54, 102, 137], and many of the schemes reported in this literature could successfully serve as the front end to the analysis described here.

All visual features were evaluated according to the best recognition accuracy they obtained. The recognition experiments were all implemented using hidden Markov models (HMM's) to statistically model either visual, audio or audio-visual speech. The use of statistical models requires a representative database of examples on which they can be trained and tested.

1.2 Audio-Visual Databases

A significant problem in the evaluation of audio-visual speech recognition systems is obtaining an aligned audio-visual database of suitable size. The acoustic speech recognition community has recorded and distributes many large databases for comparative evaluation. It much is harder to obtain and distribute an audio-visual database because of the very high data rate of the visual channel.

A significant amount of time during this PhD project was allocated to the recording, digitising and segmentation of a database of the isolated letters A to Z by ten talkers. This

[1] A tribute to the vision of Clarke and Stanley Kubrick that reviews the progress toward such a system is given in a recent book edited by Stork [180].

was a technologically difficult task in 1995, but rapid speed and bandwidth increases since make this less of an issue in 1998.

A smaller database of the digits 1 to 4 was made available around the same time by Movellan [141]. His database is used, with permission, for a comparison of visual speech features over two different tasks.

1.3 Audio-Visual Integration

The final evaluation of a proposed lipreading feature must be how much benefit is obtained when it is used in conjunction with acoustic speech. However, the best method for the fusion of auditory and visual information is currently a subject for research. Psychologists have proposed several conflicting models of bimodal human speech perception. In some cases it is not clear how useful these models are for engineering solutions to the audio-visual integration problem.

In signal processing terms, the problem is that of data-fusion and occurs in other fields where multiple streams of data must be merged. In this thesis, two different integration strategies are compared (early and late) for the three most successful visual features considered.

1.4 Thesis Overview

Chapter 2 begins by reviewing the relevant findings from psychological studies of human speechreading. This provides the motivation for building audio-visual speech recognition systems. Previous work on visual and audio-visual recognition is then reviewed and classified according to the visual feature extraction approach used.

The audio-visual databases used to evaluate the proposed lipreading features are described and discussed in Chapter 3.

The next two chapters introduce the computer vision techniques used to extract visual features. Chapter 4 describes *sieves* and how they can be used in low-level, pixel-based image analysis. Both two-dimensional and one-dimensional image segmentations are discussed and potential lipreading features identified. Chapter 5 introduces statistical modelling of lip shape and describes how this can be used to constrain an iterative lip contour tracker. An extension to this technique models the appearance of the lips as well as the shape and uses a learned model to fit to example images.

Hidden Markov models are reviewed in Chapter 6. All proposed visual features are compared using similar visual-only recognition tasks with HMM's in Chapter 7.

The different approaches to audio-visual integration are described in Chapter 8. For the most successful visual features, results are presented comparing two different integration strategies.

The results and findings of this thesis are summarised in Chapter 9. Proposals for future extensions to the work are discussed and real-time implementations of two of the visual feature extraction methods proposed are described.

Chapter 2

Speechreading Background

2.1 Human Speechreading

The human capacity to perceive speech through visual cues, the ability to lipread, is well known. At least since the 16th century, it has been known that hearing-impaired people can be taught to read, write and talk. An historical review is given in French-St. George [69], and a 17th century example describing speechreading is Bulwer [35]. Today, it is clear that it is not only hearing-impaired people that benefit from, and use, the visual aspects of speech. Even those with normal hearing are better able to understand speech if they can see the talker, Reisberg [162]. Visual enhancement is even more pronounced when the auditory signal is degraded and noisy. For example, Sumby and Pollack [182] found that viewing the talker's face was effectively equivalent to an additional 15dB acoustic signal to noise ratio. Neely [145] found an increase in intelligibility of 20% with the addition of vision. Others have reported similar findings [14, 66, 148].

A powerful demonstration of the auditory-visual nature of speech comes from the McGurk effect [117, 135, 185]. When presented with conflicting auditory and visual symbols, under certain conditions, the perceiver will 'hear' something other than the spoken sound. An example fusion illusion occurs when an auditory /ba/ is synchronised with a visual /ga/. This is usually perceived as /da/, different from either the visual or auditory stimuli. This occurs even when the perceiver is aware of the illusion, and has been demonstrated in four month old infants [36] as well as adults.

Another demonstration comes from experiments with auditory-visual asynchrony. McGrath [133] found an audio lead of less than 80ms or lag of less than 140ms could not be detected during speech. However, if the audio was delayed by more than 160ms it no longer contributed useful information and the subjects reported identification accuracy down to the visual-only level. They concluded that, in practice, delays up to 40ms are acceptable. It is worth noting that in television broadcasting terms this represents a single PAL frame of asynchrony. Evidently any audio-visual speech recording must take care to maintain audio-visual synchronisation. A study by Kuhl [100] found that even five month old infants prefer the face of a talker with synchronised audio to one without.

More comprehensive reviews of the psychological aspects of human speechreading may be found in [37, 41, 61, 184]. It is clear however, that even normal hearing humans are able to make good use of visual speech cues and that this is particularly important under noisy conditions. The McGurk effect shows that auditory and visual speech perception are tightly bound and appear to be so from an early age. Precisely how and where this fusion occurs remains the topic of much research.

2.1.1 Speech Production

A simplified diagram of the speech articulators is shown in Figure 2.1. Speech in most languages is produced by the lungs forcing air through the vocal chords located in the larynx. The vocal chords are two muscular folds that are usually apart for breathing, but can be brought close together and then vibrate in the airstream from the lungs. The vibration is controlled by the tension of the chords and modulates the airstream. This is the process of phonation, and the sounds produced are *voiced*. Sounds made using an unrestricted, unmodulated airstream are *unvoiced*.

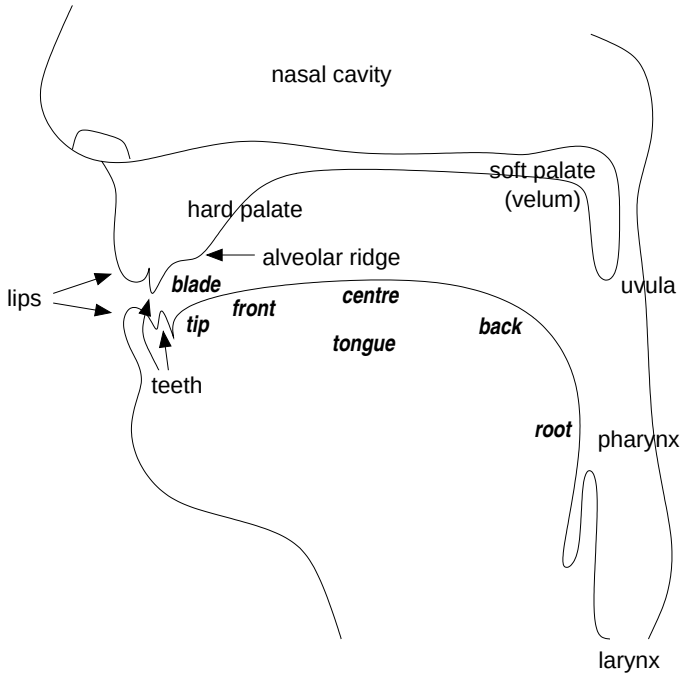


Figure 2.1: Principal features of the vocal tract. Due to [101].

Above the larynx is the vocal tract, the first stage of which is the pharynx (back of the throat) which can be tightened to change speech sounds, but is not often used in English. From the pharynx the airflow may be redirected either into the nose and mouth, or just the mouth by closing the velum (soft palate). Sounds made with the velum open are *nasal* and with the velum closed *oral*. The shape and configuration of the vocal tract further filter the speech sound. The sounds produced can be classified according the the place and manner of their articulation.

The manner of articulation describes the degree of occlusion used to create the sound. For example, a complete closure and hold of the articulators halts the airstream and is called a *stop*, e.g. the first and last sounds in 'pop'. A *fricative* occurs when the articulators are brought close enough together to cause a turbulent airflow, e.g. 'zoo', and an *approximant* is when the articulators are close, but not enough to cause a fricative, e.g. 'we'. Finer classifications of the manner of articulation can be made and are described in [47, 101].

The place of articulation describes which articulators are used, and is classified as one of,

Bilabial between both lips. For example, 'pie'.

Labiodental between lower lip and upper front teeth. For example, 'fie'.

Dental between tongue tip or blade and upper front teeth. For example 'thigh'.

Alveolar between the tongue tip or blade and alveolar ridge. For example, 'tie'.

Retroflex tongue tip and back of the alveolar ridge. Not used in English.

Palato-alveolar tongue blade and back of the alveolar ridge. For example, 'shy'.

Palatal between the front of the tongue and the hard palate. For example, 'you'.

Velar back of the tongue and the soft palate. For example, the end of 'hang'.

If there is no contact between articulators the sound is a *vocoid* (as opposed to a *contoid*) and there is no identifiable place of articulation. These are classified, using tongue position in the cardinal vowel space, as front, centre or back, and low, mid or high. Additionally, lip shape may be classified as *rounded* or *spread*. The vowels used for received pronunciation (RP) English transcription are shown in their positions in the cardinal vowel space in Figure 2.2.

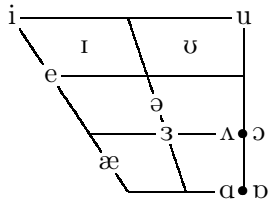


Figure 2.2: Cardinal vowel-space showing vowels used for received pronunciation (RP) English. Front is to the left.

2.1.2 Phonemes and Visemes

When two different speech sounds can be used to change the meaning of a word they are different *phonemes*. A phoneme is an abstract unit of speech sound that is useful for transcribing speech in an unambiguous manner. Because there are more phonemes than letters in most alphabets, the International Phonetic Association (IPA) define a phonetic alphabet for transcription that is common across all languages. The RP vowels are identified by position and their IPA symbol in Figure 2.2. A list of the 45 phonemes used for RP English transcription is given in Table 2.1. Some vowels are not stationary but involve transition from one position to another. These are termed *diphthongs* and are also listed in Table 2.1.

The equivalent unit of meaningful visual speech would be the *viseme*. From the brief description of human speech production given, and the locations of the speech articulators, Figure 2.1, it is clear that only a small part of the articulation process may be seen. Typically only the lips, teeth, tongue tip and blade are clearly visible. It is therefore not surprising to find that there are fewer visemes than phonemes, and that typically many phonemes map to a single viseme. However, there is little agreement on the number of visemes, or how phonemes are clustered to form them. Nitchie [147] found consonants clustered into 12 visemes. Walden [192] found that, without training, humans could identify only five consonant visemes, but that this increased to nine with training. Finn [67] used a computational analysis of twelve dots placed around the lips of a single talker and formed ten consonant viseme clusters. The nine consonant viseme clusters found by Walden were /θð, fv, ʃʒ, sz, pbm, tdnkgj, w, r, l/.

There is also evidence that visual intelligibility varies between talkers. For example, Kricos [97] evaluated six different talkers using the Utley lipreading test [187]. It was found

	IPA symbol	Example		IPA symbol	Example
Vowels	i	lead	Consonants	p	pin
	ɪ	lid		b	bin
	e	led		t	tin
	æ	lad		d	din
	u	food		k	could
	ʊ	good		g	good
	ʌ	cut		f	fan
	ɒ	cot		v	van
	ə	about		θ	thin
	ɜ	church		ð	that
	ɔ	caught		s	sue
	ɑ	lard		z	zoo
Diphthongs	eɪ	bay		ʃ	shoe
	aɪ	buy		ʒ	vision
	ɔɪ	boy		tʃ	chew
	ɑʊ	loud		dʒ	gin
	əʊ	load		m	my
	ɪə	pier		n	no
	ɛə	pear		ŋ	sing
	ɔə	pore		l	lie
	ʊə	poor		r	rye
				j	yes
				w	wet
				h	hat

Table 2.1: IPA symbols for the phonemes used in RP English transcription with examples of their use.

that the talkers that were easiest to lipread also articulated the most visemes. The talkers had between four and eight visemes and they did not cluster in the same way for all talkers.

2.1.3 Visual Features

Given that vision has been shown to be important in human speech perception, then what are the visual cues that best provide speech information?

Benoît [14] compared speech intelligibility under conditions of audio-only, audio and lips and audio and face for natural and synthetic image displays. The lips gave improved intelligibility but this was further improved by the addition of the rest of the face. Similar results were earlier presented by McGrath [134], who found that being able to see the teeth improved performance compared to just the lips. Jackson and Montgomery [88, 139] used statistical analysis of physical lip measurements in a vowel task. They found the strongest dimensions were horizontal and vertical lip opening. A similar finding is seen in the statistical shape analysis used for lipreading in Chapter 5.

There is some evidence that it is the dynamic motion of the visible articulators that is most important. Rosenblum [165,166] found a kinematic point light display of the lower face could be used to induce McGurk effects. The only visual features were dots placed on the talkers face. However, neuropsychological studies of brain damaged patients by Campbell [38, 39] finds conflicting examples. A patient unable to recognise static faces or read facial expression was found to have normal speechreading ability and responded to McGurk illusions. Another patient, that could see motion but not form, was unable to speechread. She concludes that both visual form and movement are required for speechreading [40].

Munhall and Bateson [144, 188, 189] describe a combined optical point tracking and electromyographic analysis of facial dynamics during speech. They found visual speech information was distributed over the entire face and not just the lips. They also found that even in high noise conditions the listener did not look at the lips of the talker more than 50% of the time. This suggests humans are able to perceive visual speech using the low spatial resolution off-fovea parts of the retina. Experiments by Jordan [90] have shown that speechreading is robust over large variations in image size.

In summary, while most visible speech information is, unsurprisingly, seen around the lips, it is also distributed over the entire face. Whenever a component is removed, for example the face or teeth, the intelligibility falls. This cautions that visual speech analysis should take care when discarding some of the many possible facial features which are available.

2.1.4 Multimodal Integration

Section 2.1 described research that has clearly shown that human speech perception is able to use visual cues to improve intelligibility, especially in noisy conditions. However, exactly how and where humans integrate the auditory and visual stimuli is still unknown. Broadly, the various integration models can be classified as either *early* or *late*. For early integration, the auditory and visual stimuli are processed together as a single feature. Late integration considers the two modalities separately and forms a fused result only after they have been independently classified. According to these definitions, early integration is able to make use of the temporal correlations between each modality, and late integration cannot.

The extra speech information made available through vision was highlighted by Summerfield [183]. First, the identity and position of the talker is known and attention can be directed to them, vision also determines *when* the talker is speaking. Second, vision can provide information common to the acoustic source, allowing audio-visual synchronisation to be maintained and providing a useful additional source of information. Finally, vision

presents complementary information. Summerfield found that the clearest visual cues, place of articulation, are the least robust acoustic features in noise. He also proposed five different integration models, considering audio-visual fusion using the following representations:

1. phonetic features. Fused using a vision-place, audition-manner rule;
2. filter function of the vocal tract. An estimate of the vocal tract is made from the visual signal and fused with auditory;
3. direct concatenation of acoustic visual features;
4. vocal tract configurations. Full 3D vocal tract representations, the front from the visual analysis and the internal from acoustic analysis.
5. articulatory dynamics. Transform both auditory and visual features to a modality-free domain, such as kinematic parameters that describe the articulatory dynamics.

Summerfield dismisses only the vision-place, audition manner (VPAM) model as an inaccurate representation of human integration and suggests only that integration occurs before categorisation [184]. Early integration allows more complex interaction between acoustic and visual speech before categorisation into phonetic features. Green [74,75] also agrees with this conclusion.

By contrast, Massaro [122–125] proposes a late integration scheme using the fuzzy logical model of perception (FLMP). Here, independently calculated truth values for each modality are fused using a multiplicative rule.

A comparison by Braida [22] found that, for the identification of consonants, the early integration model more consistently predicted accuracy than the FLMP or an alternative late integration model. A comparison of several models of integration, similar to those suggested by Summerfield [183], is Robert-Ribes [163] and Schwartz [169]. Their proposed models are considered in more detail in Chapter 8.

2.2 Previous Speechreading Systems

The brief overview of human speechreading in the previous section suggests that significant gains may be made by integrating auditory and visual cues for automatic speech recognition. This section briefly describes the previous work in automatic visual and audio-visual speech recognition. Recent reviews can be found in Henneke [82], Goldschen [72] and Chen [45].

There are two major problems for an audio-visual speech recogniser. The first problem is robust visual feature extraction. What are the salient visual speech features? Compared to acoustic speech recognition, this problem is even more difficult due to the much higher visual data rates. For example, full frame, full rate PAL video at 16-bits per sample requires a data rate of 22.5MB/s.

Two clear paradigms have emerged for visual feature extraction. The high-level, or model-based, approach asserts prior decisions about what visual features are important. The low-level, or pixel-based, approach ignores all prior knowledge (such as there is, Section 2.1.3) and applies a direct statistical analysis to the data. Provided the correct model is used (and known) then a model-based system that focuses on specific image features, that are considered to be important, might be expected to be more robust.

The second problem is how and when to integrate the audio and visual speech features. Similar to the discussion of human integration, Section 2.1.4, these can be described as either

early (pre-classification) or late (post-classification). More detailed discussion on integration for audio-visual speech recognition is deferred to Chapter 8.

Between the extremes of model-based vs. pixel-based and early integration vs. late integration is a continuum of implementational possibilities [72]. This is further expanded by considering the complexity of the speech recognition task; how many talkers, how large a vocabulary, and discrete or continuous speech dictation etc. The following sections briefly describe some of the systems previously reported, classified by the type of visual analysis and in chronological order.

2.2.1 Low-level Analysis

The first audio-visual speech recognition system was built by Petajan [151] in 1984. He used custom image processing hardware to extract mouth features in real-time and combined them, using a heuristic rule-based late integration, with audio features from a commercial acoustic speech recogniser. Face images were thresholded, smoothed and the nostrils located. These identified the binarised mouth area, and mouth height, area, width and perimeter length were measured. Using a single talker (himself), he showed that adding visual information improved recognition accuracy on digits, letters and a 100 word vocabulary test. Visual recognition was implemented using a direct distance measure to example templates.

An improved version of Petajan's first system [152] used vector quantisation of the mouth images and added dynamic time warping to better align utterances for the template matching.

An audio-visual vowel recognition task was described by Yuhas [195, 196]. Static 20×25 images from a single talker were mapped to a power spectra estimates using a neural network. This was integrated with the acoustic power spectra using a weighted average and classified using another neural network. Adding visual information from static images still improved accuracy in noisy conditions.

An optical flow analysis around the mouth was used by Mase [121] in a continuous digit recognition task. Four windows were located around the mouth and the horizontal and vertical velocity computed in each. A principal component analysis (PCA) of the velocities identified lip opening and elongation as the most significant modes and these measures formed the final visual features. Linear time warped template matching was used for a purely visual recognition task and, for a very small test set (15 utterances from talker 1, 2 from talker 2 and 4 from talker 3) that did not include examples of all digits, 70–100% accuracy was achieved.

Some early work by Bregler [23] used normalised greylevels from a 24×16 mouth image, or FFT coefficients from a 64×64 image, as features for a time-delay neural network. Results from a continuous German letter recognition task for two talkers were improved when visual features were integrated using a combination layer in the neural network.

Following from Petajan's work, and extending his original hardware, Goldschen [70–72] used hidden Markov models (HMM's, Chapter 6) in a continuous visual-only recognition task on 450 sentences from the TIMIT database for a single talker. He extracted seven features from the binarised mouth images of Petajans hardware and calculated the first and second derivatives and magnitudes for all. Using PCA only 13 measures were retained, ten of which were derivative based. They considered viseme and triseme (three visemes in context) HMM's but obtained a best result, by clustering triseme models, of 25%. They also performed a phoneme to viseme clustering based on the hidden Markov models and found close agreement with those found by Walden in Section 2.1.2.

Silsbee [175, 178, 179] also used HMM's for a speaker dependent 500 word vocabulary audio-visual recognition task. Weighted late integration was used to combine the scores of

the audio and visual HMM's. All images were classified as one of 32 codebook images and the codebook label formed the visual feature. This system was also tested on vowel and consonant discrimination tasks.

A direct PCA analysis of subsampled mouth images was used by Brooke [29–34, 186] for a continuous digit recognition task. A single talker was restrained, to remove head motion, and recorded an audio-visual database of digit triples. A 6×10 pixel window of the the lips was obtained after subsampling the 64×64 mouth images to 16×16. Visual features were formed using the top ten components of a PCA analysis of these images. For classification hidden Markov models were used with early integration (concatenation) of the audio and visual features. Improved accuracy was obtained after noise compensation was used on the audio features. A further enhancement to this system was a cross-product HMM topology [186] that allows temporal variability between the audio and visual features during articulation.

Lip motion has also been found to improve speech segmentation, especially in noise, Mak [119, 120]. Motion vectors were extracted from lip images using morphological image processing (Chapter 4) in a similar method to the optical flow used by Mase [121]. The resultant velocity of the lips was estimated by summing the velocities from four windows on the image; left, right, top and bottom. A weighted late integration combined the visual and acoustic segmentation boundaries.

Duchnowski [63, 64] extended Bregler's [23] previous German letter recognition task by adding visual features derived using PCA analysis and linear discriminant analysis (LDA) of the mouth images. They also tried varying the position in the neural network where audio and visual integration occurs and investigated an automatic, neural network based, lip position estimator. They found slightly better performance when integrating later in the network, Meier [136]. A recent enhancement to this system is an automatic face and lip finding system [137].

Movellan [142, 143] used lip images directly as visual features for HMM's. A subsampled 20×15 image was formed so that horizontally, half was the lip image and half the difference between the previous lip image. A comparison was made between early and late integration that slightly favoured a sum of log-likelihoods late integration for a multi-talker isolated digits (1–4) task. Movellan's database is described more fully in Section 3.2 and results are obtained using it in Chapter 7.

Some preliminary results using 'eigen-sequences' have recently been reported by Li [104]. Principal component analysis was applied to entire subsampled image sequences, warped to a standard length. A continuous recognition task of the letters 'A'–'J' for a single talker gave 90% accuracy using only visual information.

2.2.2 High-level Analysis

Current high-level, model-based, methods either explicitly or implicitly measure lip shape using tracking techniques from computer vision. All of these fit some sort of 'cartoon' model to the lips and extract features that describe the shape of this model.

The original implementation of active contours, or *snakes* [91] included a demonstration of lip-tracking. A snake is a dynamic elastic contour controlled by its internal stiffness parameters and attraction to image features, such as strong edges. These are typically used to track image sequences using a Kalman filter framework and can be defined to include shape constraints and learned dynamics [16, 17].

Deformable templates [18, 197], differ from elastic snakes by explicitly defining parameterised geometric shape templates. These were also first applied to locating and tracking facial features, including lips.

A further shape constrained method is the Active Shape Model (ASM), or 'smart snake' [50–55]. These are described more fully in Section 5.1 as they are one of the lipreading methods applied in this thesis. They differ from deformable templates by learning their shape constraints using a statistical analysis of a set of hand labelled training examples.

For lipreading, Hennecke [83] used a deformable template of the lips and found that the poor contrast of the lips to the surrounding face posed significant problems. In particular the lower lip contour tended to settle into the valley between the lips and chin. Later work [82,84, 85] included automatic location of the face and lips and HMM based early integrated audio-visual recognition. For a visually distinct task of /da/, /fa/, /la/ and /ma/ they achieved 38% visual accuracy across ten talkers and found a 16% improvement when combining audio and visual features. For a harder eleven word train-station name task, they found early integration degraded audio-only performance, and late integration gave little improvement.

Silsbee [176, 177] and Chandramohan [42] used a simpler deformable template to track only the inner lip contour for the consonant task described for Silsbee in the previous section.

Although not strictly a high-level, model-based tracking method, the blue lipstick chroma-key extraction used by Adjoudani [1] can be considered, for the classification used here, as a real-life, real-time snake. A single talker wore blue lipstick and the lips were extracted using chroma filtering in real-time. Six lip parameters were extracted; internal and external lip width and height, inner area and total lip area. Using HMM classifiers on a 54 nonsense word vocabulary early and late integration was compared. Best audio-visual results were obtained using a late integration scheme that allowed the relative audio and visual weighting to varied as a function of acoustic signal to noise ratio (SNR).

Coianiz [48] used a simplified version of the deformable template used by Henneke [83] with colour filtering to enhance lip colours and better locate the lip contours. A spline based lip tracker that also used colour filtering has been described by Sánchez [168].

A different deformable template model, constructed using curve sections and node points with associated springs was implemented by Vogt [190, 191]. This also used colour filtering to highlight the lips and improved audio-visual performance was obtained for a 40 word recognition task using time delay neural network classification.

Dynamic contours were used for lip tracking of a single talker by Kaucic [58, 93, 94]. He found that greylevel images were inadequate for tracking, unless lipstick was used. The use of learned shape and motion models allowed the tracker to run in real-time on an SGI workstation. For a side-view lip profile tracker (with bright background) [93] dynamic time warping and early integration gave some improvement on a digit recognition task. Improvement was also seen for a 40 word recognition task using lipstick highlighted frontal view tracking. Later work showed that colour filtering could be used to highlight the lips sufficiently to dispense with the lipstick [92].

Luettin [106–116] used active shape models (ASM's) [50–55] on the same subset of digits task and database as Movellan [142, 143]. A greylevel profile model of the lip contours was trained from the same example images used to define the statistical shape model used in ASM fitting. Using both shape and greylevel information gave the best visual-only performance using HMM classifiers. This approach was also successfully used in person identification [113] and on a larger digits database [106]. A similar implementation was used for the lipreading experiments in this thesis, and is described in Section 5.1.

Recently a three-dimensional model of lips has been developed for a single talker by Basu [9–11]. This has also been used for 3D lip tracking, aided using colour modelling of lips.

2.2.3 Combined High- and Low-level Analysis

In an extension to earlier low-level analysis, [23], Bregler [24–28] used lip tracking to guide a pixel-based analysis of the mouth. A nonlinear analysis of the shape of the outer lip contour was used to constrain the tracker. This is similar to the linear analysis of shape used for active shape models, and can also be used to interpolate between lip images by using the nonlinear manifold in shape space to define only valid lip contours.

However, it was found that features obtained using the lip tracker were not good enough for visual or audio-visual speech recognition using a hybrid HMM where the probability distributions (Chapter 6) are modelled using a multi-layer perceptron. Instead, the tracker was used only to locate the mouth area which was subsampled to 24×16 pixels before PCA analysis to form the final visual features. To interpolate lip images (for early integration the audio and visual rate must match) nonlinear analysis of the lip images was used. A relative improvement in recognition accuracy of between 15% and 32% was found by adding visual information for a continuous letter spelling recognition task for 6 talkers.

Chapter 3

Audio-Visual Databases

This chapter describes the two aligned audio-visual databases used to evaluate the different types of visual speech features considered in this thesis.

Statistical models for classification require a representative database of examples if they are to be reliably trained and tested. In the audio-visual speech community there is currently a need for a large standard database on which statistically well trained models and comparative results can be made. This is now being addressed by, for example, BT Labs [46] and AT&T [155] but the problems of storing and distributing large audio-visual databases are still significant.

In the absence of a standard database each research group has collected their own. When the work described in this thesis was started there were no audio-visual speech databases available and we recorded our own aligned audio-visual database of isolated letters called *AVletters*. Subsequently the *Tulips* database [141] of a subset of isolated digits was made freely available from the University of California at San Diego. To compare the methods presented in this thesis with those of other researchers this digits database has been used in conjunction with our letters database. Also, using two databases usefully allows the comparison of different visual speech analysis methods over different sized tasks.

Section 3.1 describes the AVletters database and Section 3.2 the Tulips database. Finally, a comparison between the two is made in Section 3.3.

3.1 AVletters Database

The AVletters database consists of three repetitions by each of ten talkers, five male (two with moustaches) and five female (none with moustaches), of the isolated letters A–Z, a total of 780 utterances. All talkers were volunteers and all recording had to be completed in one hour, allowing few re-takes to be made. All the females were undergraduate students, one male was staff, two were postgraduate and the remaining two males undergraduates.

Recording took place in the campus TV studio under normal studio overhead lighting conditions. All recording was of the full face using a JVC KY950B video camera and stored on SVHS videotape using a Panasonic AG-MD830 recorder. The output of an SONY ECM55 tie-clip microphone was adjusted for each talker through a Soundtech Series A audio mixing desk and fed to the video recorder. Talkers were prompted using an autocue that presented each of three repetitions of the alphabet in the non-sequential, non-repeating order shown in Table 3.1. Each talker was requested to begin and end each letter utterance with their mouth in the closed position. No head restraint was used but talkers were provided with a close-up view of their mouth and asked not to move out of frame.

Repetition	Letter sequence
First	NEIYSFMDCKHJZTQUVXPWRAGBLO
Second	DGMOKWLJBENSCUIFHXAQRTYVZP
Third	GLUCMDYPZTJNOSQEHRKAWVXIBF

Table 3.1: Recording sequence for the AVletters database.

Each utterance was digitised at quarter frame PAL resolution (376×288 at 25fps) using a Macintosh Quadra 660AV in 8-bit 'greyscale' mode. In practice the greyscale data obtained is simply the luma information stripped from the 16-bit colour component (YUV 4:2:2) video digitiser. This standard digital video sampling method reflects the undersampling of the chroma information, analogous to the narrower bandwidth used for chroma in analogue video. The colour information is shared between two pixels so that each has an 8-bit luma value but only one 8-bit colour component. The second 8-bit colour component is stored with the neighbouring pixels luma. This sharing of the colour information allows two colour pixels to be packed into 32 bits. Table 3.2 shows how this is stored in memory on a typical workstation [153].

Pixels 1–2			
Byte 1	Byte 2	Byte 3	Byte 4
uuuuuuuu	yyyyyyyy	vvvvvvvv	yyyyyyyy
left			right

Table 3.2: YUV 4:2:2 digital video sampling.

As defined by the international standard for component digital video, ITU-R BT.601-4, luma is coded using headroom range [156]. Headroom range is defined for 8-bit luma data as having the range [16–235]. Black is defined as 16 and white 235. This extra range is defined to allow for signal processing of the signal in studio environments, for example some under- and over-shoot introduced by ringing of a digital filter. For image processing it simply means the data does not use the full 8-bit dynamic range.

Audio was simultaneously recorded at 22.05kHz, 16-bit resolution. All recording was done using a QuickTime capture application written specifically for this task. Digitisation was direct to RAM (maximum amount, 68MB) as even quarter PAL frame greyscale video was beyond the disk bandwidth of a 1993 Macintosh. Recording direct to memory ensured no frames were dropped and using the QuickTime recording libraries maintained audio-visual time alignment. Digitisation was performed in blocks of three letters due to memory (recording time) limitations. Sufficient space was left before the first letter to allow the SVHS video recorder to settle and the video digitiser to accurately synchronise to it; this gave reliable 25fps recording.

The full face images (Figure 3.1 shows an example), were further cropped to a region of 80×60 pixels after manually locating the centre of the mouth in the middle frame of each utterance.

Each utterance was temporally segmented by hand using the visual data so that each utterance began and ended with the talkers mouth in the closed position. The audio signal was further manually segmented into periods of silence-utterance-silence. Figure 3.2 shows example frames from each of the ten talkers. An example utterance sequence is shown in Figure 3.3 for the letter 'M'.

This database has been made freely available for research on CDROM and has been

Figure 3.1: Example full frame image from the AVletters database.

Figure 3.2: Example frame from each of the ten talkers in the AVletters database.

Figure 3.3: Example sequence from the AVletters database of the letter 'M'. First frame is top-left and the sequence proceeds in a left to right, top to bottom order.

distributed to several non-UK research groups by request.

3.2 Tulips1 Database

The Tulips database [141] was recorded by Javier Movellan of the Department of Cognitive Science at the University of California at San Diego. It contains two repetitions of the isolated digits 1–4 by each of 12 talkers, 9 male and 3 female, a total of 96 utterances. This was recorded using office ceiling lights with an additional incandescent lamp at the side to simulate office working conditions. Talkers were not restrained but could view their mouths and were asked not to move out of shot. All talkers were students of the cognitive science school at the time of recording and are of mixed ethnic origin.

The database was digitised at 100×75 resolution at 30fps (NTSC frame rate) using a Macintosh Quadra 840AV in ITU-R BT.601-4 8-bit headroom greyscale. Audio was simultaneously recorded at 11kHz, 8-bit resolution. This database is available to download from `http://cogsci.ucsd.edu/~movellan/`. Each utterance was hand segmented so that the video and audio channels extended to one frame either side of an interval containing the significant audio energy. If the lips were clearly moving before or after this time up to an additional three extra frames were included.

This database has been extensively studied and recognition results obtained using several feature extraction methods have been published by Movellan et. al [73, 141–143] and Luettin [106, 107, 109, 110].

An example frame from each of the twelve talkers is shown in Figure 3.4. The side incandescent lamp more clearly highlights the lip contour in comparison to the ceiling lit AVletters, Figure 3.2. An example utterance sequence of the digit '1' is shown in Figure 3.5.

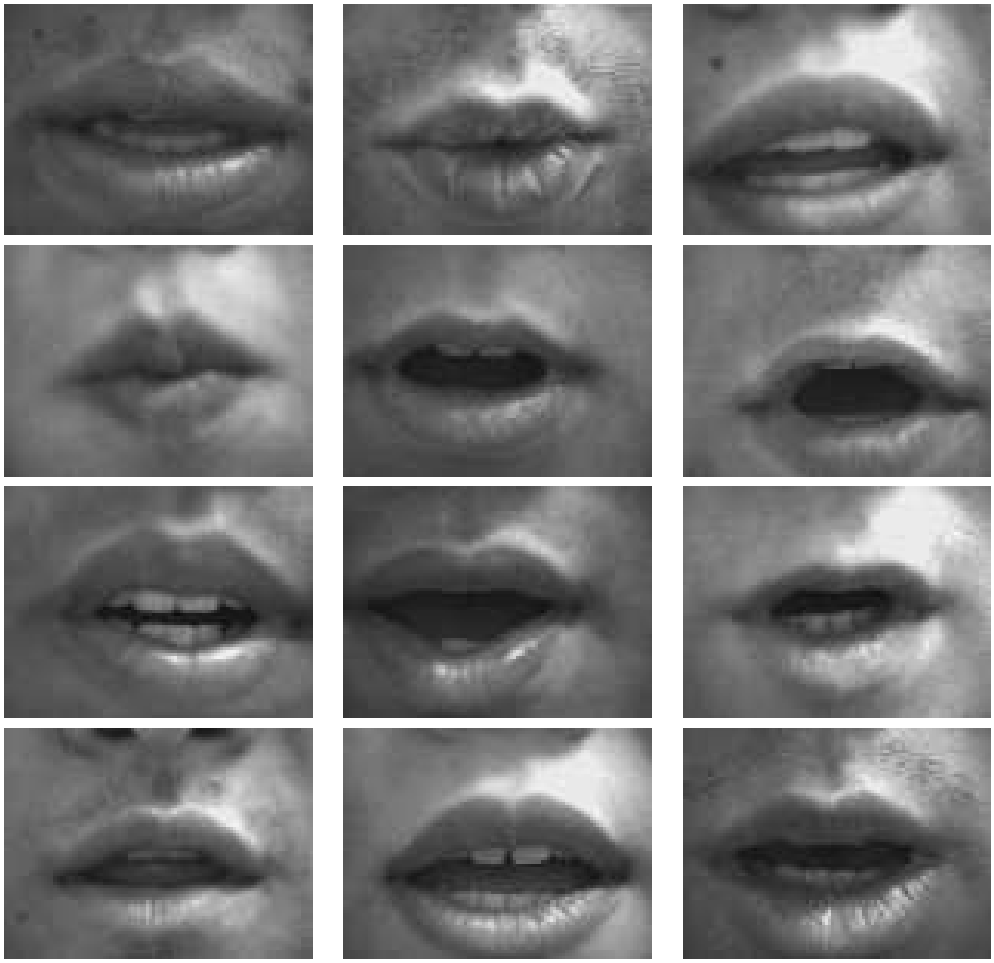


Figure 3.4: Example frames from each of the twelve talkers in the Tulips database.

Figure 3.5: Example sequence from the Tulips database of the digit '1'. First frame is top-left and the sequence proceeds in a left to right, top to bottom order.

3.3 Comparison

The most obvious difference between the AVletters and Tulips database is size. For statistical analysis the 780 utterances of the AVletters database is really rather small. In comparison, audio databases typically have hundreds of thousands of utterances. The WSJCAM0 database [68] for example has 8280 training *sentences* and 3840 test sentences with a 64,000 word vocabulary. The Tulips database is only an eighth as large as AVletters but offsets this to an extent by covering the much simpler four word vocabulary. A comparison of the database tasks and conditions is given in Table 3.3.

Attribute	AVletters	Tulips
Task	'A'–'Z'	'1'–'4'
Repetitions	3	2
No. talkers	10	12
Utterances	780	96
Frames	18,562	934
Frame rate	25Hz	30Hz
Image size	80×60	100×75
Lighting	ceiling	ceiling + side
Audio	16-bit	8-bit
Audio rate	22.05kHz	11.127kHz

Table 3.3: Comparison of AVletters and Tulips databases.

The second clear difference is the average utterance length. Despite the faster 30Hz NTSC standard frame rate of the Tulips database the more zealous cropping by audio energy of the utterances produces a mean of only 9.7 frames. The AVletters utterances where cropped using lip motion and audio was segmented independently so the mean utterance has 23.8 frames. There is also a greater range of utterance length in AVletters. Table 3.4 shows the min, max, mean and standard deviation of the utterance lengths for both databases.

Stat	Unit	AVletters	Tulips
Min	frames	12	6
	seconds	0.48	0.20
Max	frames	40	16
	seconds	1.60	0.53
Mean	frames	23.80	9.73
	seconds	0.95	0.32
Std	frames	4.48	2.24
	seconds	0.18	0.07

Table 3.4: Statistics of AVletters and Tulips databases.

Both databases have considerable variability in image quality, the quality of focus often differs between talkers. Talkers in the Tulips database sometimes move their lips out of the frame, this can be seen in the example sequence in Figure 3.5. Both databases attempted to minimise motion of the talker by providing a feedback framed shot of the mouth during recording and in both cases this is not always successful. Gray [73] found recognition accuracy based on pixel analysis improved after the Tulips images were aligned for translation, rotation and scale using the tracking results of Luettin [116].

For a database as small as Tulips it is practical to hand align many of the images. Luettin hand labelled training points on 250 images for his active shape model lip tracker, almost 27% of the data. This means for a large portion of the database the true model alignment was known. For a more representative larger database this is unrealistic. In this thesis all pixel-based analysis methods are applied to these databases without any pre-processing.

Chapter 4

Low-Level Visual Speech Analysis

The goal of visual feature extraction is to obtain discriminatory lipreading information from image sequences that is robust to varying imaging conditions and talkers. This chapter describes four low-level, pixel-based methods that use a multiscale spatial analysis (MSA) technique based on sieves [3–5, 7, 8]. Low-level methods avoid enforcing prior knowledge of what the salient image features for lipreading are. Instead, direct analysis of the image pixel values is used and, typically, statistical methods are used to determine features.

Multiscale spatial analysis differs from most low-level methods that use principal component analysis (Appendix A) to extract 'eigenlip' features from the image greylevels [24,27,32, 34,63,104,136]. Instead, visual features are derived using one- or two-dimensional scale-space analysis. This disassociates pixel intensities from the low-level image structure and allows independent analysis of the structure within a well defined mathematical framework.

This chapter describes three methods that use a 2D image analysis to derive area-based features, and one 1D analysis that measures vertical lengths. When further data reduction is required PCA is used to find the most significant components.

First, sieves and the morphological operations required are defined and their properties discussed. The four proposed lipreading feature extraction methods are then described and demonstrated using example sequences.

4.1 Multiscale Spatial Analysis

Multiscale spatial analysis is a low-level, pixel based, method of image analysis. An image is decomposed using a nonlinear scale-space decomposition algorithm called a *sieve* [3–5, 7, 8]. This is a mathematical morphology serial filter structure that progressively removes features from the input signal by increasing scale, Figure 4.1 shows this structure. At each stage the filtering element ϕ removes extrema of only that scale. The first stage, ϕ_1, removes extrema of scale 1, ϕ_2 removes extrema of scale 2 and so on until the maximum scale m. The extrema removed are called *granules* and a decomposition into a *granularity* domain is invertible and preserves scale-space causality [79].

There are a number of possibilities for the filtering element, ϕ. Section 4.2 covers some mathematical morphology operators and describes the properties required for using them in a sieve.

Section 4.3 gives a mathematical description of a sieve and shows how it decomposes a signal. Section 4.4 describes how lip features can be obtained using this decomposition.

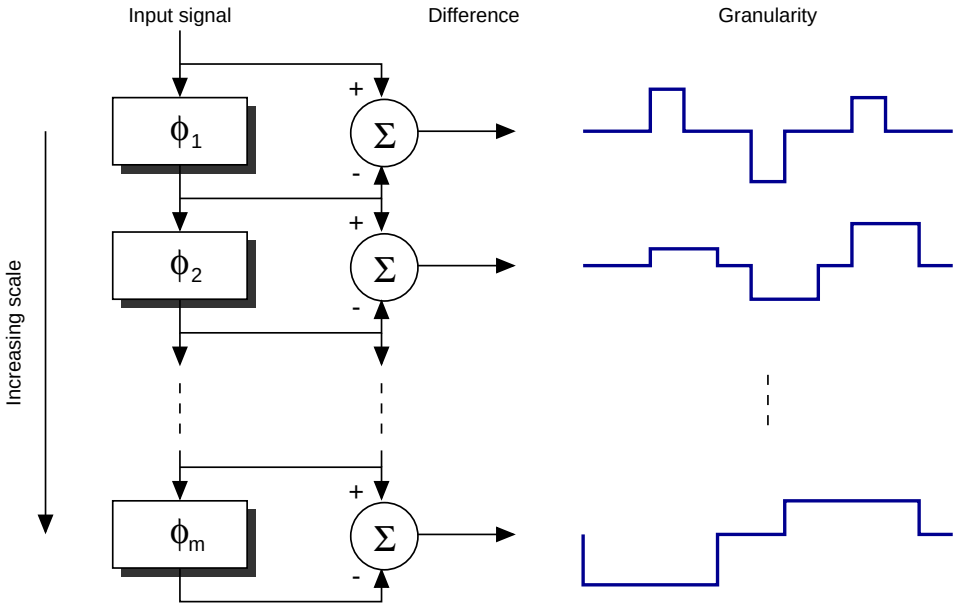


Figure 4.1: Sieve structure.

4.2 Morphological Processors

Mathematical morphology is the nonlinear analysis of signals, often images, by shape. The original work by Matheron in 1975 [126] was extended by Serra [170–172]. This section is a basic introduction to the morphological processors as applied within a sieve. Fuller expositions include the above references and introductions by [62, 78, 154].

The basic morphological operation is a probing of the signal, X, using some *structuring element*, B. When applied within a sieve there is no longer the need for the concept of a rigid structuring element, it is retained here to make the explanation simpler and compatible with the standard definitions.

4.2.1 Erosion

Binary morphological erosion is defined as the set of all points, y, from the output of the structuring element, B, where it completely fits inside the set of signal points, X,

$$\epsilon_B(X) = \{y,\ \forall b \in B,\ y + b \in X\} \tag{4.1}$$

$$X \ominus B = \bigcap_{b \in B} (X + b) \tag{4.2}$$

where $\epsilon_B(X)$ and $X \ominus B$ are alternative notations for the erosion of X by B.

Equation (4.2) shows that binary erosion can also be considered as the intersection of the signal and the structuring element. In the binary case this is a logical AND'ing of the signal with the structuring element. This can be extended to greyscale, i.e. the processing of a real valued function, $f(x)$, with a real valued structuring element, $h(x)$, that need not be flat. Greyscale erosion is then defined as,

$$\epsilon_{h(x)} f(x) = \inf_{y \in H} [f(x + y) - h(y)] \tag{4.3}$$

If the space is discrete the infimum is replaced by the minimum operation,

$$\epsilon_{h(x)} f(x) = \min_{y \in H} [f(x+y) - h(y)] \tag{4.4}$$

A useful analogy is that of an upwardly sprung stylus, the structuring element, moving underneath the signal. In Figure 4.2 the structuring element passes along the underside of the signal shown in blue. The red signal is obtained for a flat structuring element of width three that outputs in the centre.

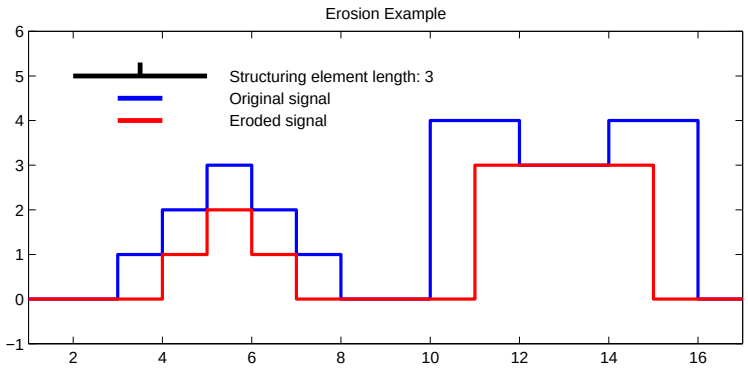


Figure 4.2: Example of greyscale erosion.

4.2.2 Dilation

Binary dilation is the complement of erosion and defines the set of all points, y, from the output of the structuring element, B, where it touches the set, X,

$$\delta_B(X) = \{y + b,\ y \in X,\ b \in B\} \tag{4.5}$$

$$X \oplus B = \bigcup_{b \in B} (X + b) \tag{4.6}$$

again $\delta_B(X)$ and $X \oplus B$ are alternatives for the dilation of X by B.

Dilation is the union of the signal and the structuring element. For binary signals this is a logical OR'ing of the structuring element with the signal. This can also be extended to greyscale processing of the function, $f(x)$, and structuring element, $h(x)$,

$$\delta_{h(x)} f(x) = \sup_{y \in H} [f(x-y) + h(y)] \tag{4.7}$$

In discrete space the supremum is replaced by the maximum operation,

$$\delta_{h(x)} f(x) = \max_{y \in H} [f(x-y) + h(y)] \tag{4.8}$$

Figure 4.3 shows the same signal as in Figure 4.2 but with the structuring element passed over the top of the signal. The red signal is the dilation of the blue signal with a flat, centre outputting, structuring element of width three.

4.2.3 Opening and Closing

Erosion and dilation are the most basic morphological operators but they introduce a bias into the filtered signal through either the max or min operation. This can be reduced by

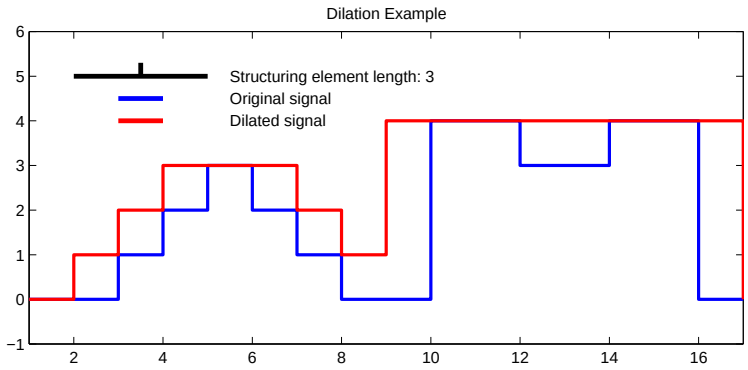


Figure 4.3: Example of greyscale dilation.

following an erosion by a dilation, or following an dilation by an erosion—an opening or closing.

The opening operation is obtained when the signal that is first eroded then dilated (min then max),

$$\psi_{h(x)} f(x) = \delta_{h(x)}(\epsilon_{h(x)} f(x)) \tag{4.9}$$

This removes the positive extrema smaller that the structuring element size. Figure 4.4 shows the example signal opened with a flat structuring element of length three that outputs in the centre, the local maxima have been removed.

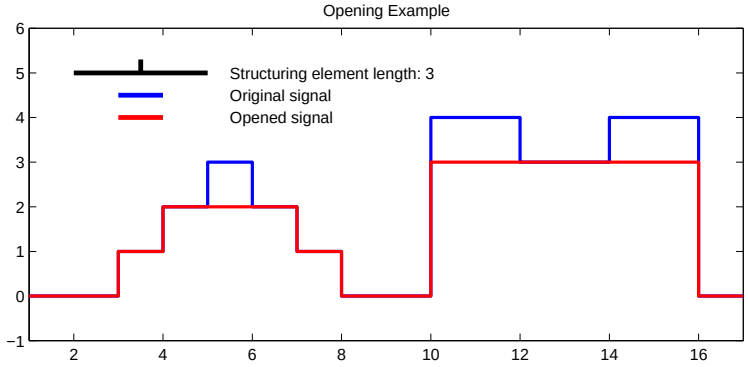


Figure 4.4: Example of greyscale opening.

The opposite to opening is closing which is defined as a dilation followed by an erosion (max then min),

$$\gamma_{h(x)} f(x) = \epsilon_{h(x)}(\delta_{h(x)} f(x)) \tag{4.10}$$

This removes the negative extrema smaller than the structuring element size. Figure 4.5 shows the example signal closed with a flat structuring element of length three that outputs in the centre.

4.2.4 Alternating Sequential Filters

A further extension is to apply alternating openings and closings [172]. Both openings and closings have a bias in the sign of the extrema they remove: openings remove positive extrema, closings remove negative extrema. By applying one after the other this bias can also be reduced, again noting that the order of application affects the resulting signal.

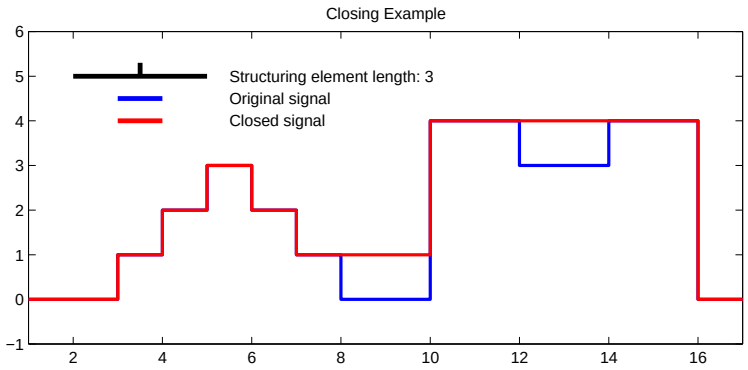


Figure 4.5: Example of greyscale closing.

The $\mathcal{M}$-filter is defined as an opening followed by a closing (e.g. min, max, max, min),

$$\mathcal{M}_{h(x)} = \gamma_{h(x)}(\psi_{h(x)} f(x)) \tag{4.11}$$

This removes both positive and negative extrema smaller than the structuring element. Figure 4.6 shows an example.

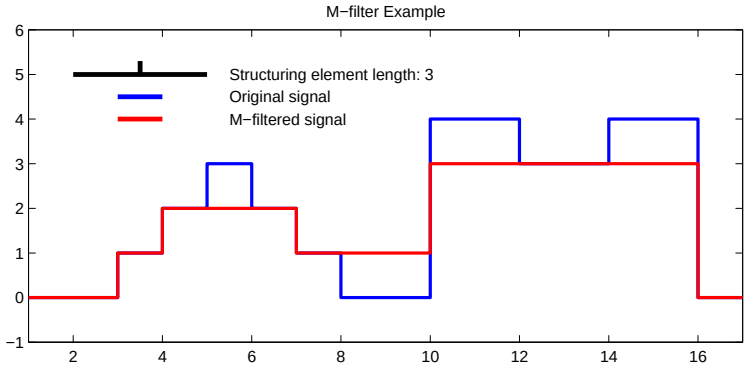


Figure 4.6: Example of an $\mathcal{M}$-filter.

The opposite $\mathcal{N}$-filter is defined as a closing followed by an opening (e.g. max, min, min, max),

$$\mathcal{N}_{h(x)} = \psi_{h(x)}(\gamma_{h(x)} f(x)) \tag{4.12}$$

This differs from the $\mathcal{M}$-filter as it removes negative extrema before positive extrema. The difference can be seen in Figure 4.7 between positions 10–16, where the $\mathcal{M}$-filter removed the positive extrema first and so filled in at the lower level the $\mathcal{N}$-filter removes negative extrema first and fills in at the higher level.

4.2.5 Morphological Properties

This section briefly describes the main properties of the morphological operations covered. To define the properties it is useful to introduce the symbol $\mathcal{Q}$ to be any of the morphological operators defined previously. In all cases a flat structuring element is assumed.

- Given an operator $\mathcal{Q}$ the *dual* is $\mathcal{Q}^*(f) = -\mathcal{Q}(-f)$ which is the inverse of the dual operator applied to the inverse of the signal. Erosions, dilations, openings, closings and $\mathcal{M}$- and $\mathcal{N}$-filters are duals.

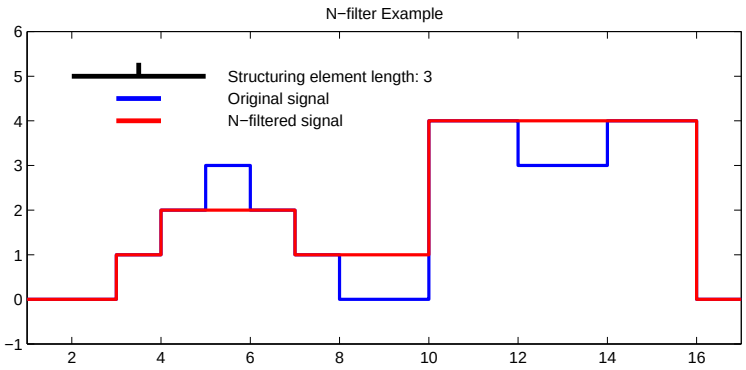


Figure 4.7: Example of an $\mathcal{N}$-filter.

- The operator $\mathcal{Q}$ is monotonically *increasing* if given $g \leq h$, then $\mathcal{Q}(g) \leq \mathcal{Q}(h)$. This means given a signal that is a subset of of another the filtered version of the subset is a subset of the filtered version of the signal. Erosion, dilation, opening, closing $\mathcal{M}$- and $\mathcal{N}$-filters are all increasing.

- The operator $\mathcal{Q}$ is *anti-extensive* if $\mathcal{Q}(f) \leq f$ and *extensive* if $f \leq \mathcal{Q}(f)$. This describes operators that are contained by the original signal and operators that contain the original signal. Opening is anti-extensive and closing is extensive.

- The operator $\mathcal{Q}$ is *idempotent* if $\mathcal{Q}(\mathcal{Q}(f)) = \mathcal{Q}(f)$. This property means that once applied, repeated applications of the same filter have no effect. Opening, closing, $\mathcal{M}$- and $\mathcal{N}$-filters are idempotent.

For use in a sieve the idempotent property is required. This ensures that at each scale the filter, in a single application, has removed everything up to the current scale. No smaller scale features are allowed to exist after the current scale filter has been applied. This means morphological opening, closing, $\mathcal{M}$- and $\mathcal{N}$-filters can be used in a sieve.

A significant difference between nonlinear morphological filters and linear filters is in edge preservation. Low pass linear filters remove the high frequency components needed to represent a sharp edge. Nonlinear filters use rank order operations (so far only mins and maxes have been defined) and have a different effect on sharp edges. Using flat centre outputting structuring elements leaves edges intact and in place.

4.3 Sieve Decomposition

A sieve can be defined in any number of dimensions by considering an image to be a set of connected pixels with their connectivity represented as a graph, $G = (V, E)$ where the set of vertices, V, are the pixel labels and the set of edges, E, represent the adjacencies. Figure 4.8 shows an example 5×5 image and the four-connected edges. For this example the image is the graph $G = (V, E)$ with $V = \{1, 2, 3, \ldots, 25\}$ and $E = \{(1, 2), (1, 6), (2, 3), (2, 7), \ldots\}$.

If the set of connected subsets of G containing m elements is $C_r(G)$ then the set of connected subsets of m elements containing the vertex x can be defined as,

$$C_r(G, x) = \{\xi \in C_r(G) \mid x \in \xi\} \tag{4.13}$$

For the example four-connected graph in Figure 4.8 the set of connected subsets with 2

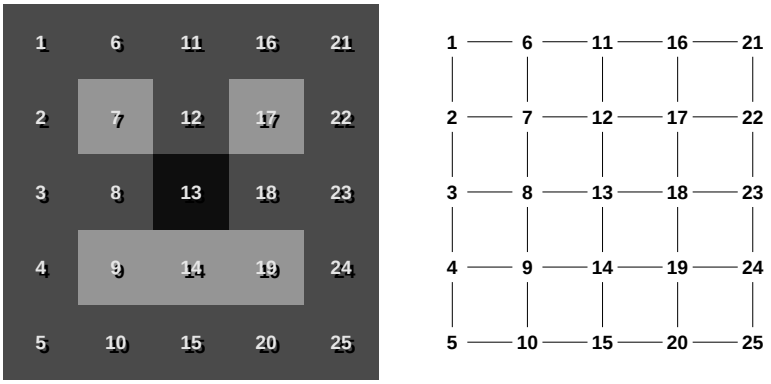


Figure 4.8: Image represented as a four-connected graph.

elements containing vertex 7 ($r = 2$, $x = 7$) is,

$$C_2(G, 7) = \{(2, 7), (6, 7), (7, 12), (7, 8)\} \tag{4.14}$$

The morphological operators defined in section 4.2 can be re-defined using this graph notation to operate on connected regions. There is no longer a rigid structuring element operating on the image because the image is now considered as a series of connected subsets and the filters are defined to operate on these. Effectively a flat structuring element is used, but it has no fixed shape as it operates over an arbitrarily connected region. The properties of filtering using a flat structuring element are retained.

In any number of dimensions, for each integer $r \geq 1$, an operator $\mathcal{Q}_r : \mathbb{Z}^V \mapsto \mathbb{Z}^V$ can be defined over the graph where $\mathcal{Q}_r$ is one of,

$$\psi_r f(x) = \max_{\xi \in C_r(G,x)} \min_{u \in \xi} f(u) \tag{4.15}$$

$$\gamma_r f(x) = \min_{\xi \in C_r(G,x)} \max_{u \in \xi} f(u) \tag{4.16}$$

$$\mathcal{M}_r f(x) = \gamma_r(\psi_r f(x)) \tag{4.17}$$

$$\mathcal{N}_r f(x) = \psi_r(\gamma_r f(x)) \tag{4.18}$$

For example, ψ_2 is an opening of scale one (ψ_1 operates on individual pixels so has no effect on the signal) and removes all maxima of length one in 1D, area one in 2D and so on for higher dimensional signals. Likewise for closing, γ, and $\mathcal{M}$- and $\mathcal{N}$-filters. Applying ψ_3 to $\psi_2(f(x))$ would further remove all maxima of scale two; length two for 1D, area two for 2D. This is the serial structure of a sieve shown in Figure 4.1. Each stage removes the extrema (maxima and/or minima) of a particular scale. The output at a scale m, f_r, is the current scale filtered version of the previous signal,

$$f_{r+1} = \mathcal{Q}_{r+1} f_r \tag{4.19}$$

where the initial signal (unaffected by an $r = 1$ morphological filter) is,

$$f_1 = \mathcal{Q}_1 f = f \tag{4.20}$$

The differences between successive stages are the *granule functions*,

$$d_r = f_r - f_{r+1} \tag{4.21}$$

the non-zero regions of which are the *granules* of only that scale.

The sieves defined using these functions are summarised in Table 4.1.

Filter	Symbol	Sieve	Extrema Processing
opening	ψ	o-sieve	maxima
closing	γ	c-sieve	minima
$\mathcal{M}$-filter	$\mathcal{M}$	M-sieve	bipolar $\pm$
$\mathcal{N}$-filter	$\mathcal{N}$	N-sieve	bipolar $\mp$

Table 4.1: Overview of sieve types.

In the 1D case Equation (4.13) becomes the set of intervals containing m elements,

$$C_r(x) = \{[x,\ x + r - 1] \mid x \in \mathbb{Z}\} \qquad r \geq 1 \tag{4.22}$$

which is identical to filtering using a flat structuring element.

4.3.1 Recursive Median Filter

In 1D a further variant on the choice of filter is a recursive median filter. The morphological filters discussed in Section 4.2 process extrema differently. An opening is a single pass operation and removes only positive extrema, closing only negative extrema. An $\mathcal{M}$-filter is a two pass operation and removes positive then negative extrema and an $\mathcal{N}$-filter removes negative extrema before positive. The recursive median filter is a one-pass approximation to the root median filter[1] in which maxima and minima are removed in the order they occur in the signal. A recursive median filter can be defined as,

$$\rho_s f(x) = \begin{cases} \text{med}(\rho_s f(x-s+1), \ldots, \rho_s f(x-1), & x \geq 0 \\ \quad f(x), \ldots, f(x+s-1)) & \\ 0 & x < 0 \end{cases} \tag{4.23}$$

$$r = (s-1)/2 \tag{4.24}$$

This describes the sliding of a centre outputting median filter of s samples over the signal. Figure 4.9 shows an example signal recursive median filtered with $s = 3$.

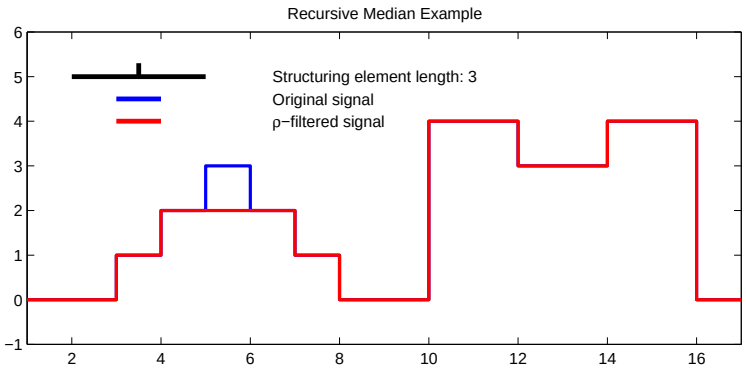


Figure 4.9: Example of a recursive median filter width three.

Unlike the previous morphological filters, a recursive median over three samples only removes extrema of width one, not two. To remove extrema of width two the signal must be

[1] A median filter applied repeatedly until no further change occurs.

filtered over five samples, Figure 4.10 shows the example at $s = 5$ and is analogous to the previous example figures. In this case it is identical to the $\mathcal{M}$-filter. In general a recursive median filter over s-samples removes extrema of scale $(s-1)/2$ where s must be odd. The definition of r in Equation (4.24) takes this into account so that r is equivalent to the other morphological filters.

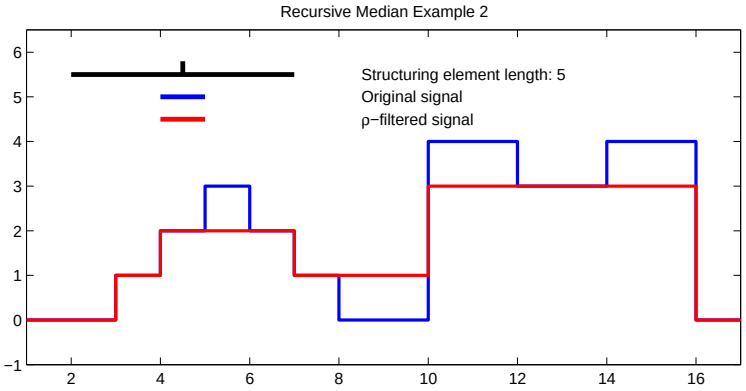


Figure 4.10: Example of a recursive median filter width five.

The recursive median, like the multiple pass root median filter, is idempotent [8]. It also inherits the robust noise rejection properties of median filters and so, in 1D, is a valid candidate for use in a sieve. In addition to the sieves defined in Table 4.1 the recursive median sieve can also be defined, Table 4.2.

Filter	Symbol	Sieve	Extrema Processing
recursive median	ρ	m-sieve	bipolar random

Table 4.2: Recursive median sieve.

One dimensional signal extrema have an implicit order, e.g. left to right, and the mathematical properties of a recursive median filter are well defined [8]. For higher dimensional signals there is no natural ordering so one must be imposed. In practice, a left-to-right, top-to-bottom scan, for a 2D example, can give viable filters that differ little from $\mathcal{M}$- and $\mathcal{N}$-filters and can be used in a 2D m-sieve.

4.3.2 Example Decomposition

Figure 4.11 shows a sieve decomposition of the 'face' image in Figure 4.8 using a 2D m-sieve. The left column shows the image successively decomposed starting from the original image at the top. The right column shows the difference image at each scale, the non-zero parts of which are the granules at that scale. At scale one the area one extrema, the 'nose' and 'eyes', are removed leaving only the 'mouth'. Nothing is removed at scale two and the 'mouth' is removed at scale three. There are no extrema greater than area three in the image so it is fully decomposed at scale three. There are no further changes from scale four through to the maximum scale of the image, $5 \times 5 = 25$.

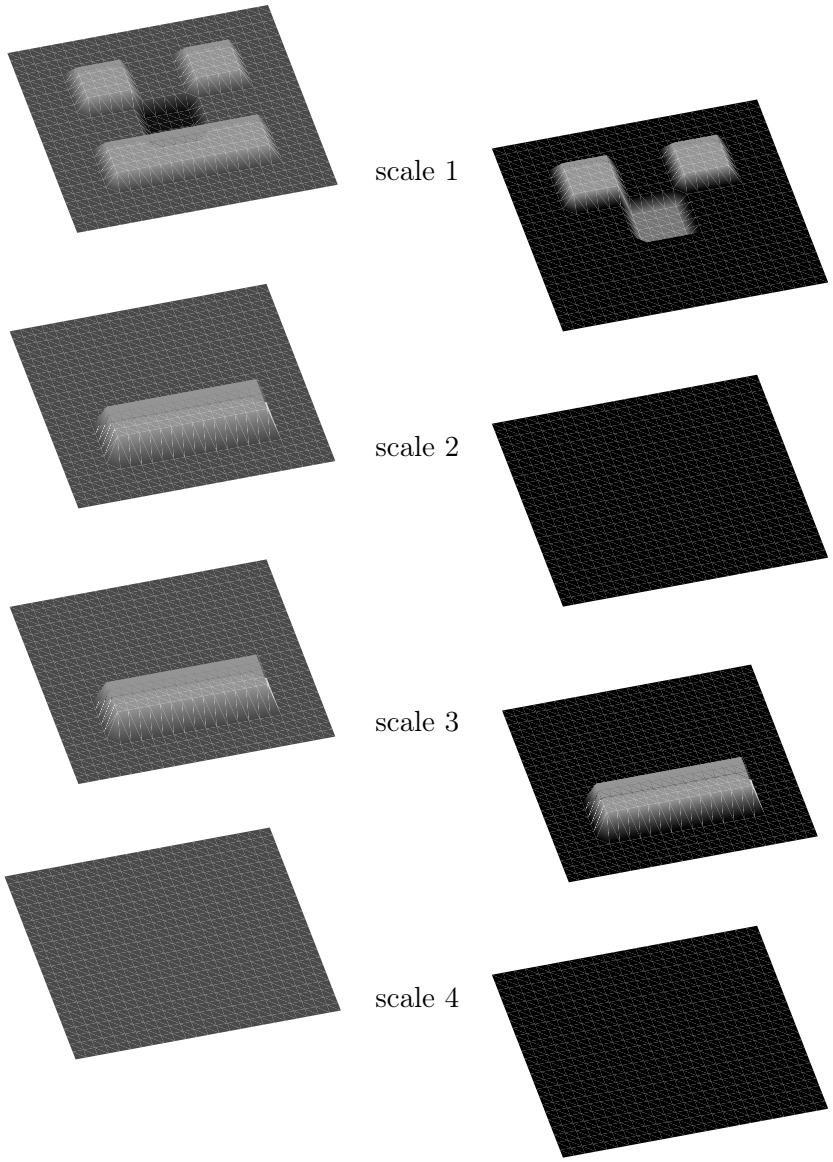


Figure 4.11: Sieve decomposition. Left column shows the successively filtered images. The right column shows the differences, containing the granules of that scale.

4.3.3 Invertibility

This simple example demonstrates the invertibility property of sieves [4,8], the original signal can be reconstructed by summing the granules, Equation (4.21), over all scales,

$$f = d_1 + d_2 + \ldots + d_m \tag{4.25}$$

where m is the maximum scale of the signal.

Adding up all the images of the right hand column of Figure 4.11 would result in an image with eyes, nose and mouth the same as the original. However, the baseline or maximum scale value, analogous to the DC component of a Fourier transform, of the image would not be preserved. To preserve the maximum scale value the signal can be padded with zeros, effectively adding a 'moat' around the signal. In 1D this is simply adding a zero to the start and end of the signal, in 2D a border of zeros around the image with area greater than the image is required. This extends the signal, without adding any new extrema, so that when the padded signal is fully decomposed the true, unpadded, maximum scale extrema can be removed. When the resulting granules are summed the original signal can be reconstructed exactly. If the signal is not zero padded there may be a offset in the reconstructed signal.

Figure 4.12 demonstrates the effect of zero padding a 1D signal using an m-sieve. The left hand column shows the original signal, its granularity decomposition and the reconstruction. Two granules are found: a negative amplitude at scale one and a positive at scale two. When summed, these reconstruct the signal with the baseline moved to zero. The right hand side shows the zero-padded signal, granularity and reconstruction. This example also demonstrates the in-order extrema processing of an m-sieve, this time the first extremum is a large positive scale one. The rest of the signal is then decomposed as two positive extrema at scale two and four with the baseline offset at scale six as expected. The reconstructed signal is identical to the original, zero-padded signal.

4.3.4 Scale Space

A sieve is a method for scale-space analysis of a signal. Scale-space extends and formalises multi-resolution analysis by defining scale as a continuous parameter, Witkin [193]. In a conventional multi-resolution scheme coarse scale is associated with coarse resolution and analysis takes place at a set of fixed scales. A scale-space representation of a signal allows analysis over all scale. Figure 4.13 shows the multiscale, scale-space representation of an image.

Koenderink [96] formalised the *causality* requirement which demands that a scale-space processor must progressively simplify the signal as the scale parameter is increased. He required that no new level surfaces may be introduced by the filtering, all features must be due to features seen at a lower scale—the process must be causal.

In addition to causality he introduced the further constraints of *homogeneity* and *isotropy*. These require all spatial points and all scales to be treated in the same manner. Under these conditions Koenderink showed that the scale-space representation of an image must satisfy the diffusion equation,

$$\nabla \left(c \nabla f \right) = \frac{\partial f}{\partial s} \tag{4.26}$$

where s is scale. If the diffusivity is a suitable constant this becomes the linear diffusion equation,

$$\nabla^2 f = f_s \tag{4.27}$$

which can be implemented by convolving with the Green's function, the Gaussian kernel. This

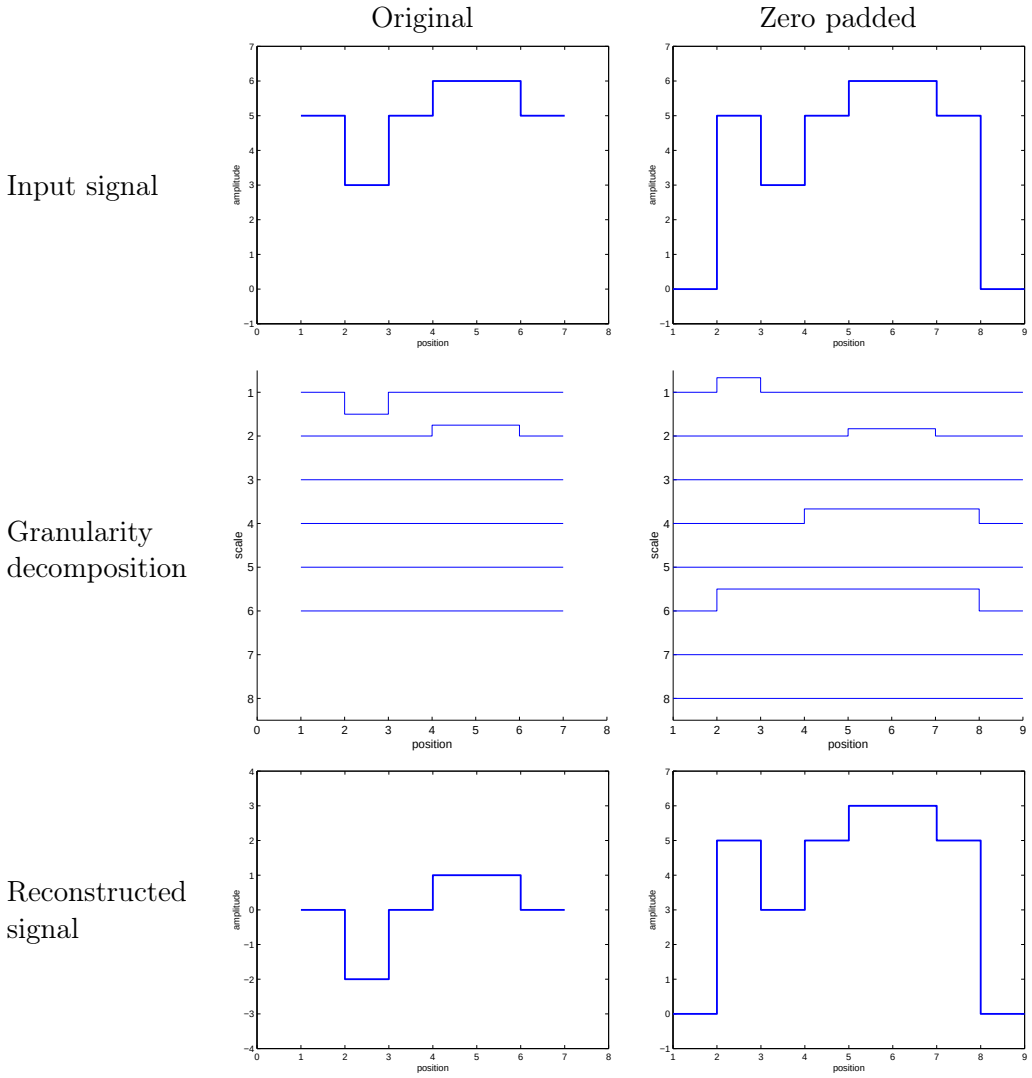


Figure 4.12: Demonstration of zero padding to preserve the baseline signal value. Decomposition using an m-sieve processes extrema in order so the zero-padded right hand side is decomposed differently. Note positive extremum at scale one instead of negative. The reconstructed zero-padded signal is identical to the original zero-padded signal. The reconstructed non-padded signal has the baseline value removed.

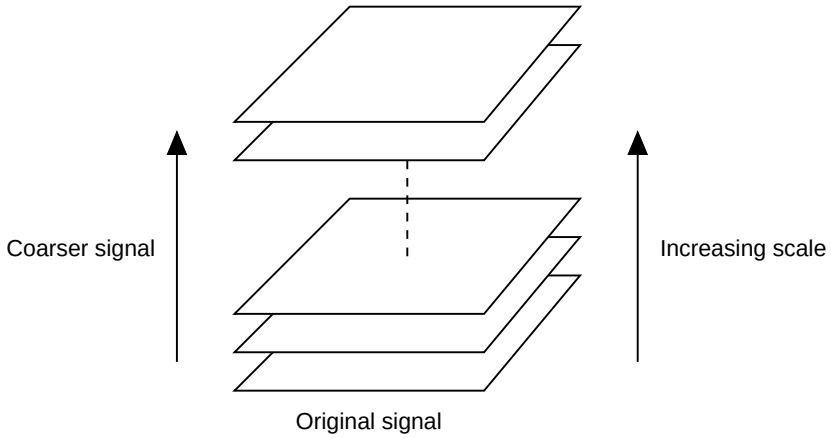


Figure 4.13: Scale-space filtering of a signal. Due to [105].

is the unique kernel for scale-space filtering under these definitions, a result that extends to arbitrary dimensional signals if causality is carefully defined [105]. However, the resulting scale-space has several problems:

1. Edges become blurred at increasing scale;
2. Extrema may be introduced;
3. An image at scale s may contain features at many scales;
4. Multiple convolutions require significant computation.

If the homogeneity and isotropy restraints are relaxed alternative methods of scale-space analysis are allowed. Perona and Malik allow the conduction coefficient, c, in Equation (4.26) to vary as a function of the image gradient [149]. The resulting anisotropic diffusion is computationally even more expensive than linear diffusion but preserves sharp edged structures in the signal.

A sieve represents an alternative nonlinear method of scale-space filtering. Figure 4.13 is clearly similar to the sieve structure in Figure 4.1. Furthermore, a sieved signal at a given scale contains *only* extrema at that scale, a result that leads to the invertibility property described in Section 4.3.3. Also, sharp edged features are preserved in scale and position, extrema are never introduced ($\mathcal{M}$- and $\mathcal{N}$-sieves [4]) and a sieve can be computed very efficiently [6].

An example of linear diffusion (Gaussian kernel), anisotropic diffusion and $\mathcal{M}$-sieve decomposition is shown in Figure 4.14 for various scales. The original image[2] is shown increasingly simplified, the edge blurring of linear diffusion is greatly reduced by using anisotropic diffusion and eliminated in the sieve decomposition. A comparison of several scale space processors can be found in [21].

4.4 Feature Extraction

A full sieve decomposition of a mouth image retains all of the information in the original image. To extract useful features from images for lipreading some way of condensing this information into a relevant feature space must be found. The use of a scale based decomposition allows this feature space to be defined using scale information rather than pixel intensity

[2]Crescent wing of the Sainsbury centre for visual arts, University of East Anglia.

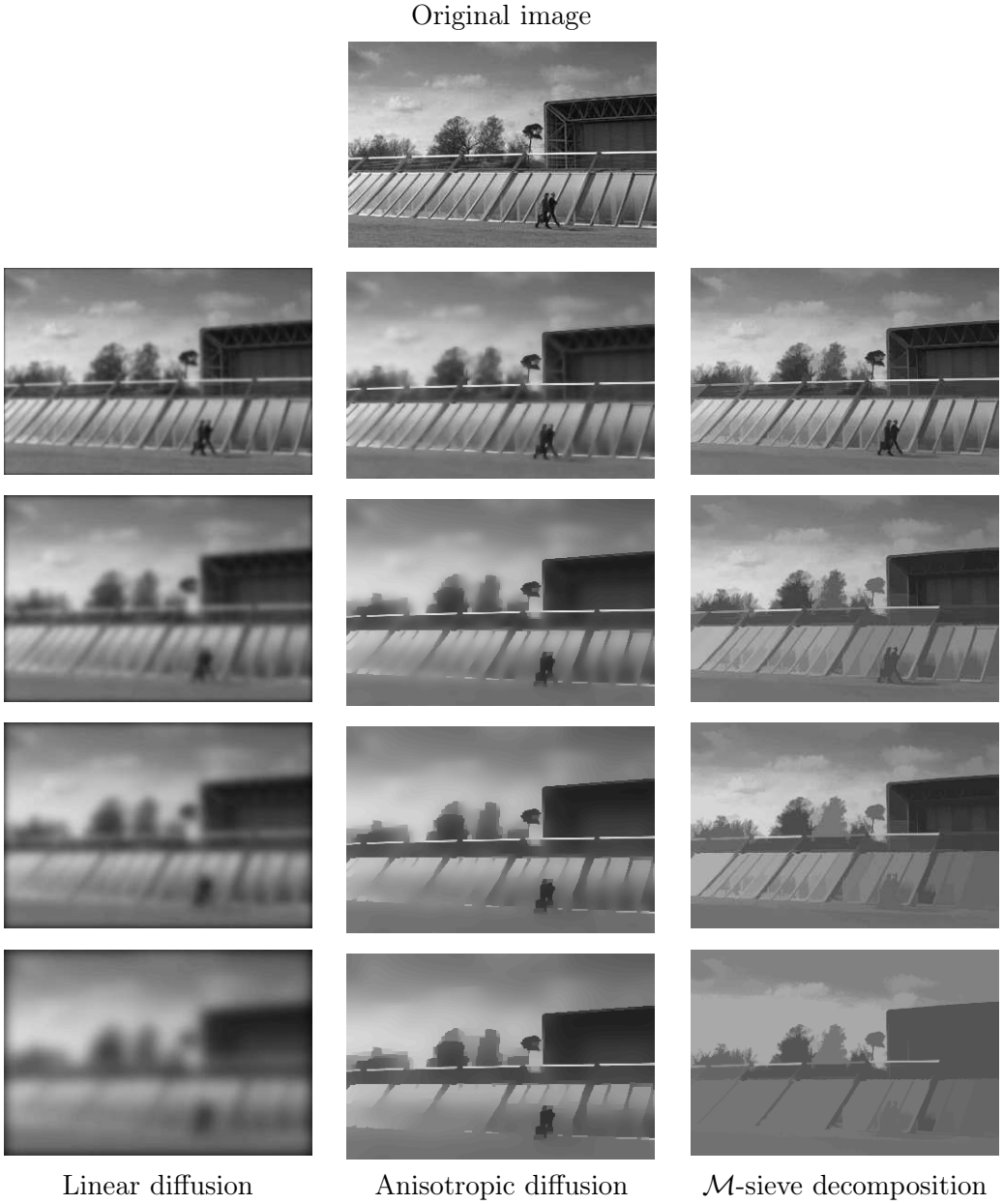


Figure 4.14: Comparison of scale-space processors. The original image is progressively simplified using: left, Gaussian filtering (linear diffusion); middle, edge preserving anisotropic diffusion and; right, sieve decomposition. From [20].

values directly. This makes the assumption that useful visual speech information may be found by examining the temporal scale structure of mouth image sequences.

All of the work presented in this section concerns feature extraction from the roughly hand-located mouth images of the two databases described in Chapter 3. The problem of automatically finding the region of interest containing the mouth is not discussed here but face location systems for lipreading have been demonstrated elsewhere [137,150]. The mouth image sequences often show significant motion as they were not accurately tracked and the talkers were not restrained. It should not be unreasonable to demand equivalent performance from an automatic system used as a front end for any of the methods discussed.

The first three proposed methods, based on 2D area-sieve analysis of the mouth images, were first published in [80]. The final method applies a 1D length-sieve to the 2D images and has been published as [80, 127–132].

4.4.1 Area Tracking

An obvious sieve-based approach is to use a 2D sieve to decompose each mouth image by area and locate the granule associated with the mouth cavity. This is similar to the blob extraction approach originally used by Petajan [151]. An example area decomposition is shown in Figure 4.15.

One problem when decomposing two-dimensional images is the large number of resulting granules; for example, an $N \times N$ square image, where $N = 2^M$, that has a checkerboard pattern at area one with another checkerboard overlayed at area four and so on would potentially have,

$$\text{No. granules} = \sum_{s=0}^{\log_2 N} \left(\frac{N}{2^s}\right)^2 \tag{4.28}$$

which is a geometric series with sum,

$$\text{No. granules} = \frac{4N^2 - 1}{3} \tag{4.29}$$

In practice there will be less granules as there are fewer greylevels than pixel values in many images. In general for an $M \times N$ image there may be granules at any or all of the $M \times N$ possible scales. This is often too fine a decomposition and for the example in Figure 4.15 the granules are grouped into channels increasing in scale bandwidth by a power of two each time, scales 1–2, 3–4, 5–16, 17–32 and so on until the maximum scale of the image, thirteen channels for this 80×60 mouth image.

The first twelve channels are shown in Figure 4.15 resulting from an $\mathcal{M}$-sieve decomposition. Here, positive granules are red and negative blue. The mouth cavity is clearly seen as a large negative granule in channels nine and ten (granules of area 257 to 512 and 513 to 1024). By channel eleven, the dark patch under the lower lip is included also. There are no granules greater than area 2048 pixels so channels twelve and thirteen are empty.

One method of obtaining lipreading information using this decomposition is to bandpass 2D sieve each mouth image and extract only the large negative granules associated with the mouth cavity. The area of these granules is a measure of the amount of mouth opening. Figure 4.16 shows an example image bandpass sieved. A threshold is applied to find only large negative granules (only granules with intensity < -25) which forms a mask covering only the mouth cavity. This is summarised in Algorithm 4.1.

The area of this cavity can be plotted over time, Figure 4.17, for an example utterance of the isolated letters D, G, M. The audio waveform is aligned and plotted underneath as

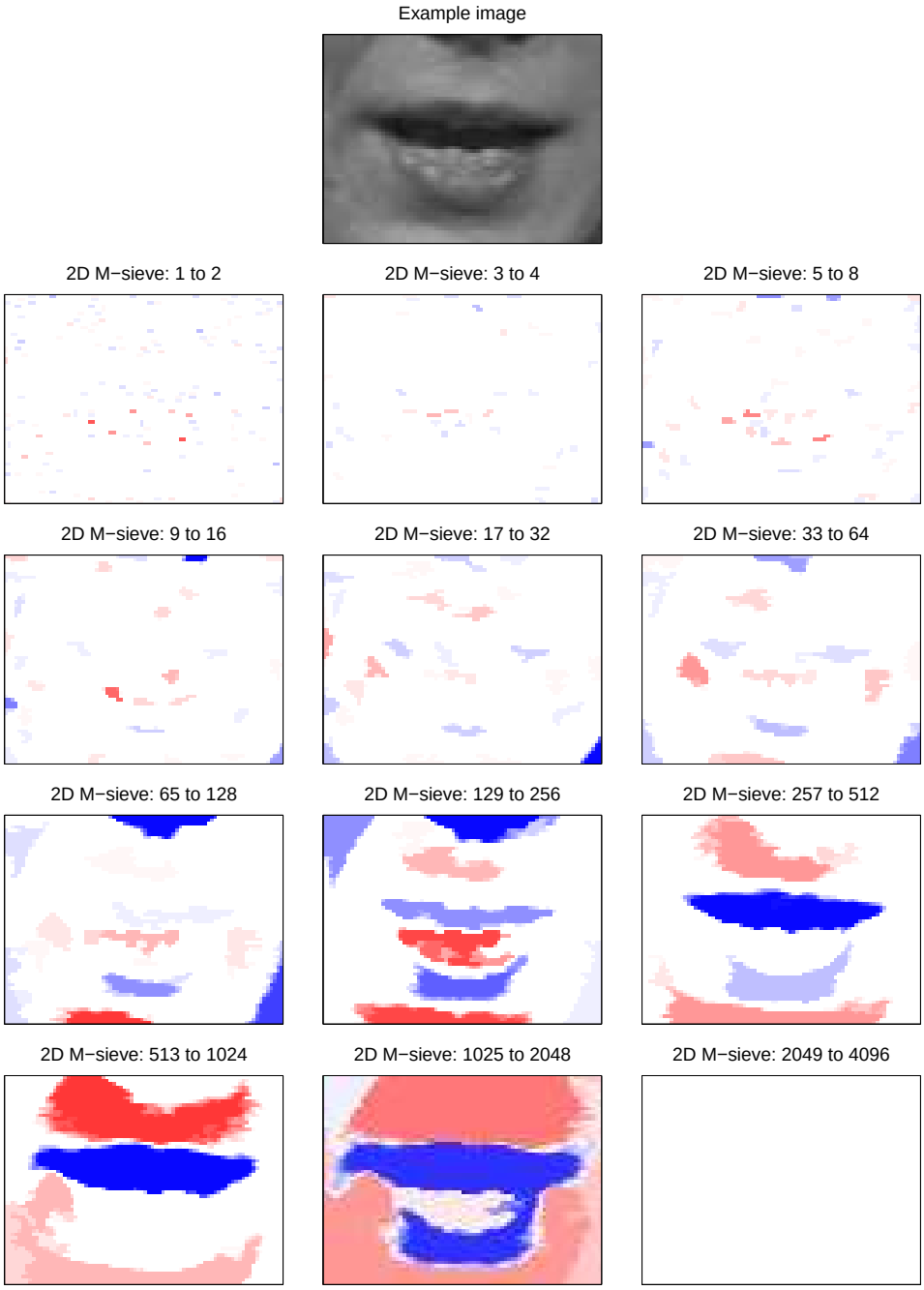
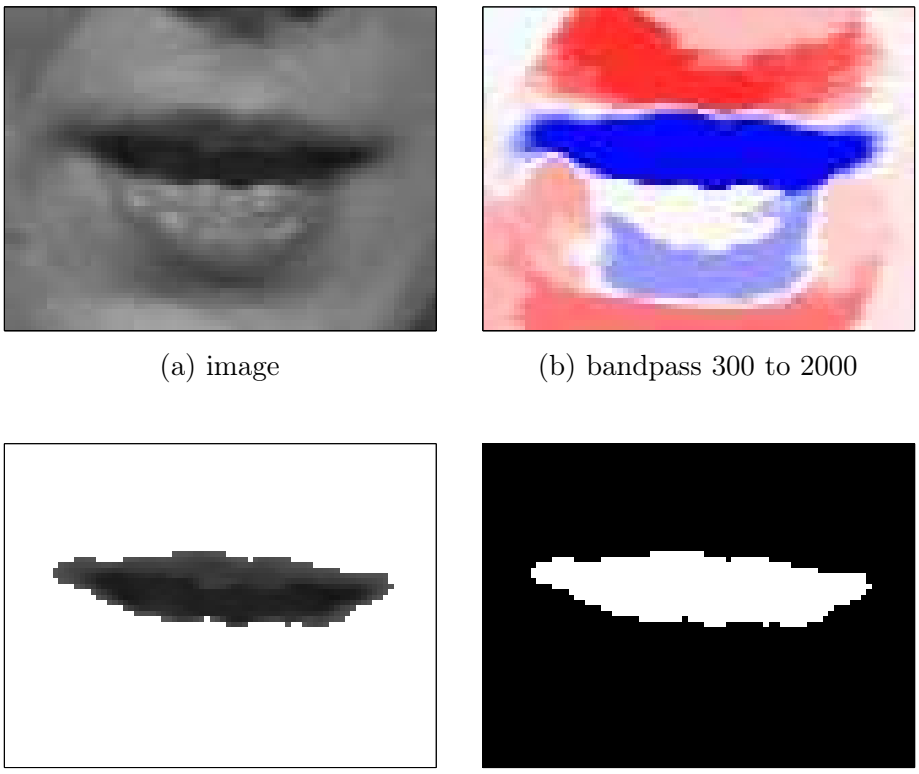


Figure 4.15: Area decomposition of a mouth image. The full 2D sieve decomposition is grouped into *channels* of granularity increasing in bandwidth by a power of two each time. Positive granules are shown in red, negative in blue.

```
1: minScale = 300
2: maxScale = 2000
3: threshold = −25
4: for all images in sequence do
5:    bpimg = bandpassSieve2D(minScale, maxScale)
6:    area = 0
7:    for all pixel values ≤ threshold do
8:       area = area + 1
9:    end for
10:   save area
11: end for
```

Algorithm 4.1: Area tracking.



(a) image (b) bandpass 300 to 2000

(d) masked image (c) ≤ -25 threshold mask

Figure 4.16: Area tracking. The image (a) is bandpass 2D sieved (b) to keep only granules of area between 300 and 200 pixels. A threshold is applied to find extrema of amplitude ≤ -25 and form a mask (c). The area of this mask forms the measurement. The actual image pixels this represents (d) are mainly the mouth cavity.

an aid to interpreting the visual data. The mouth area is generally zero between utterances, as expected. A couple of noise points between G and M are the exception. During each utterance the area rises and falls as the mouth opens and closes. For the utterance M, there is a lip smack before a release and final closing, Figure 3.3 shows the images of the 'M' of this sequence, and this can be seen in the area plot.

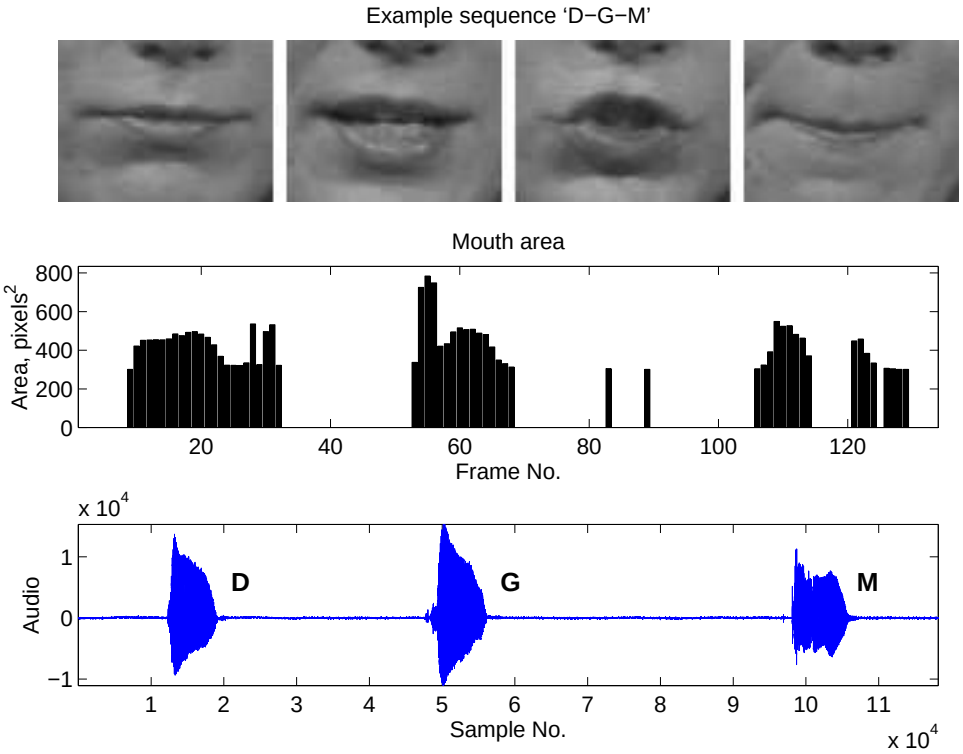


Figure 4.17: Area tracked sequence. Top row shows example images from the sequence of isolated letters, 'D-G-M'. Middle row plots the area of the thresholded granularity assumed to be the mouth cavity. Bottom row is the time aligned audio waveform.

4.4.2 Height and Width Tracking

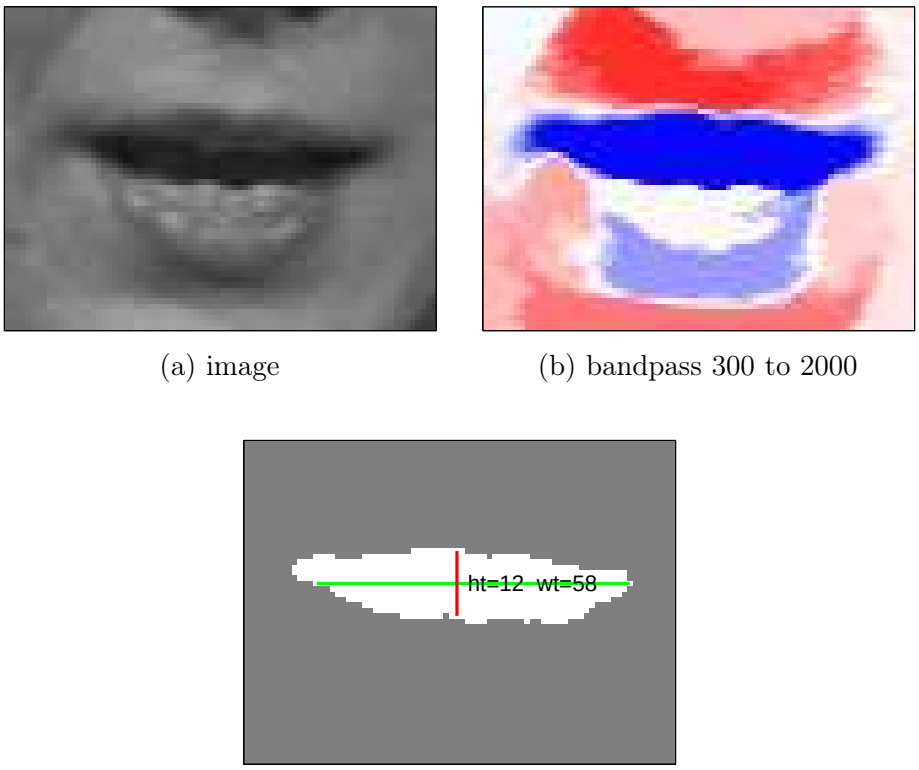
An extension to simply finding the area of the 2D sieve extracted mouth blob described in the previous section is to measure its shape. Exactly the same image processing is used to obtain the mask, for which the granularity bandpass values and threshold have been chosen so that it is associated with the mouth cavity. New features are then formed by measuring the horizontal width and vertical height of this mask.

Section 5.1 will discuss Active Shape Model (ASM) lip-tracking and, as part of the statistical analysis of lip shape, vertical and horizontal mouth opening are found to be the major modes of shape variation. This approach is similar, but much cruder, to the chroma-key lip extraction and measurement system used by the group at the Institut de la Communication Parlée, Grenoble, for example [1], and was also used in Petajan's first computer lipreading system [151].

A very simple algorithm was used but the results obtained appeared acceptable on the data sets used. However, no hand measured data exists to allow a numerical analysis of the accuracy of this method. In particular the 'flood fill' like operation of an area sieve (locating connected regions in 2D is similar to finding the level zones using a flood fill) cannot be

guaranteed to always extract just the mouth cavity. The example in Figure 4.16 shows that sometimes the mask includes part of the upper lip. The stability of this method depends on the shading of the inner lip contour and can vary between talkers and utterances. In this case the extracted mask does not extend to the lower lip but is bound by the bright contour of the lower teeth.

The height is measured by finding the vertical length of the mask through the mid-point of the image, shown in red in Figure 4.18. The width is measured as the horizontal extent of the mask mid-way along the vertical height line and shown in green. As there is little rotation in the databases this tends to extract visually acceptable measures of height and width.



(a) image (b) bandpass 300 to 2000

(c) measured height and width

Figure 4.18: Height and width tracking. The image (a) is bandpass 2D sieved (b) to keep only granules of area between 300 and 2000 pixels. A threshold is applied to find extrema of amplitude ≤ -25 and form a mask (c). The vertical length of the mask at the midpoint of the image and the horizontal width at the midpoint of the height form the mouth height and width measurements.

The height and width measured using this method are plotted for the D-G-M sequence in Figure 4.19. Height is plotted in red, width in green. As the height and width measurements are based on the same image processing used for area tracking the errors between G and M are also seen on this plot. The lip rounding at the start of the G utterance can be seen as the increase in height and decrease in width after the initial mouth opening. The utterance D has no elongation for this talker so the opposite effect is not seen there.

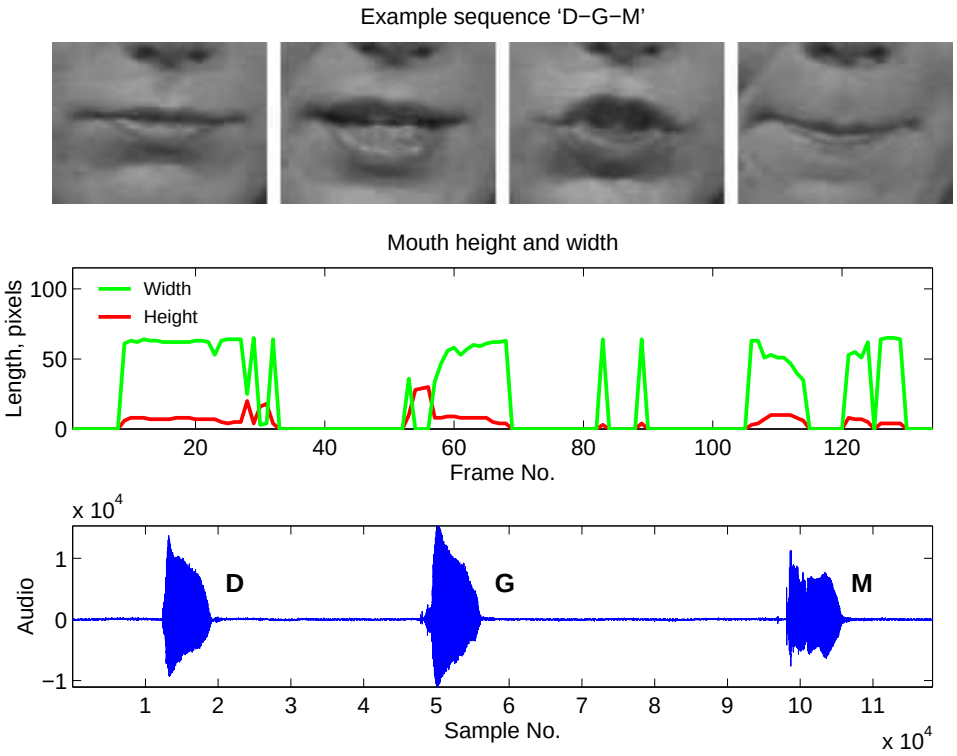
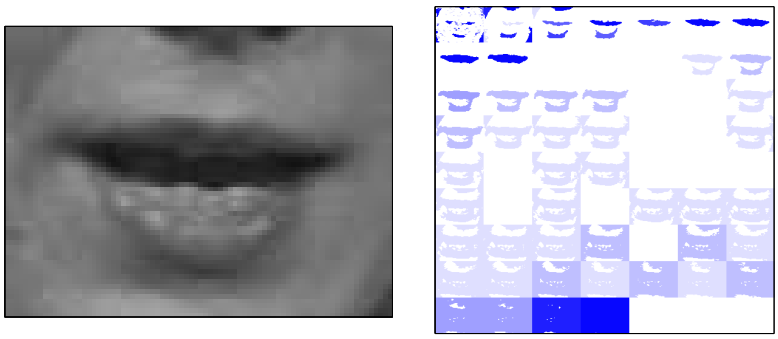


Figure 4.19: Height and width tracked sequence. Top row shows example images from the sequence of isolated letters, 'D-G-M'. Middle row plots the height (red) and width (green) of the thresholded granularity assumed to be the mouth cavity. Bottom row is the time aligned audio waveform.

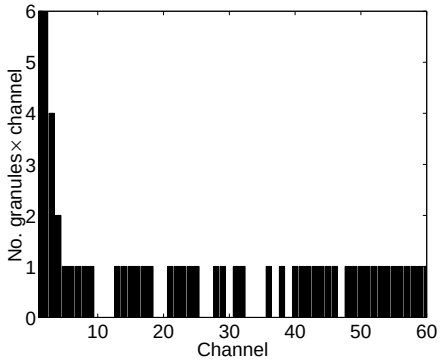
4.4.3 Area Histogram

The problems of reliably using a 2D sieve to extract the mouth cavity and perform measurements discussed in the previous two sections can be avoided by using all of the scale information rather than applying thresholds and fixing bandpass limits by hand.

An example decomposition is shown in Figure 4.20 using a closing c-sieve. As mentioned previously, a full 2D decomposition covers potentially $M \times N$ scales but can be compressed into more manageable channels of granularity. A simple measure of the distribution of scale (area) in the image can be obtained by plotting the number of granules found at each scale, or more practically each channel. In this example 60 channels are evenly spaced along the $1, 2, \ldots, 4800$ possible scales of the 80×60 mouth image from the AVletters database. This is also plotted for the D-G-M sequence in Figure 4.21 and the number of granules per channel clearly changes during articulation and remains fairly stationary at other times.



(a) image (b) 2D sieve decomposition



(c) area histogram

Figure 4.20: Area histogram. The image (a) is fully decomposed using a c-sieve to process only negative extrema. The decomposition in grouped into 60 linearly space channels (b) and the number of granules found at each scale plotted (c).

Other choices for grouping the granularity decomposition into channels are possible. If, for example, the mouth opening is considered as roughly circular then the area increases as the square of vertical opening. In this case channels spaced so they increase by a squared relationship might be more appropriate. It may be that only coarse large scale granules are relevant as any noise in the image tends to be shown at small scale, in this case an exponential channel spacing might be effective. Figure 4.22 plots the linear, squared and exponentially spaced channel boundaries used.

An example decomposition spaced using the squared curve is shown in Figure 4.23 with

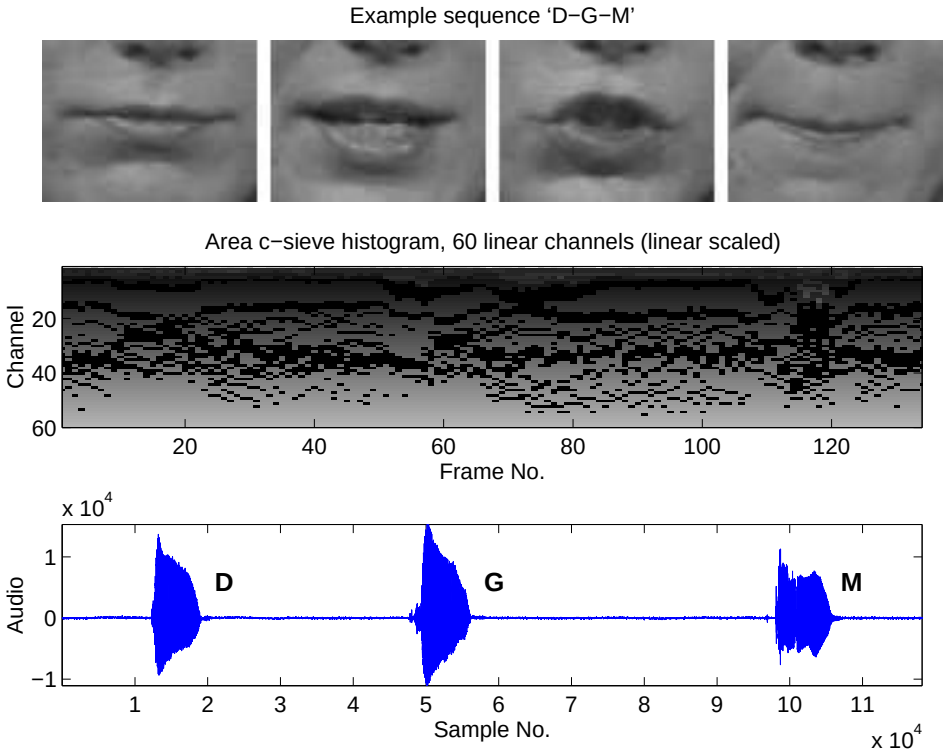


Figure 4.21: Area histogram sequence. Top row shows example images from the sequence of isolated letters, 'D-G-M'. Middle row plots the area histogram, the number of granules found at each of 60 linearly space channels in the granularity decomposition. Bottom row is the time aligned audio waveform.

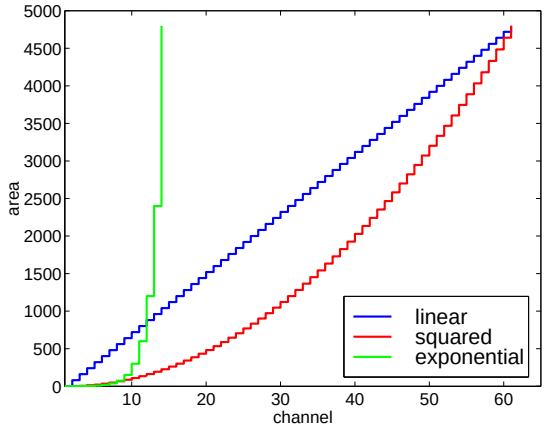
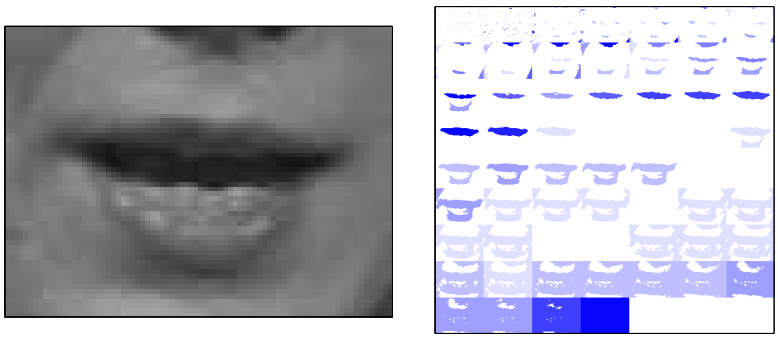
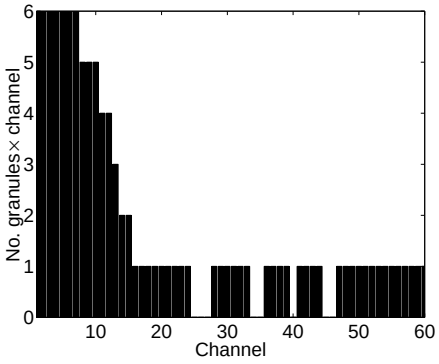


Figure 4.22: Channel spacing. Comparison between linearly spaced channels and channels spaced evenly on a square curve or exponential.

the matching D-G-M sequence in Figure 4.24. The result of used this spacing is that more channels are devoted to the smaller scales and finer information can be seen in the lower channels of Figure 4.24.



(a) image (b) 2D sieve decomposition



(c) area histogram

Figure 4.23: Area histogram. The image (a) is fully decomposed using a *c*-sieve to process only negative extrema. The decomposition in grouped into 60 channels increasingly spaced on a squared axis (b) and the number of granules found at each scale plotted (c).

As well as choosing the channel spacing and number of channels the type of sieve used for the decomposition is important. Figures 4.20 and 4.23 both show decompositions using a closing, *c*-sieve, that only processes negative extrema. The mouth cavity is usually darker than the face and a closing sieve constrains the analysis to decompose the image using only these features. However, any of the types of sieve discussed in section 4.3 can be used.

The histogram of an area decomposition of an image is relatively insensitive to displacement. If no new image features were introduced by, for example a horizontal shift of the mouth, then the area histogram would not change.

4.4.4 Scale Histogram

The previous sieve-based analysis has all been derived using a 2D area-sieve. This gives a very large number of possible scales and can suffer from a lack of stability in the poor contrast regions that are often found around the lips. The resulting features are rather coarse measurements either based on empirically chosen values to preprocess the images or on the distribution of area.

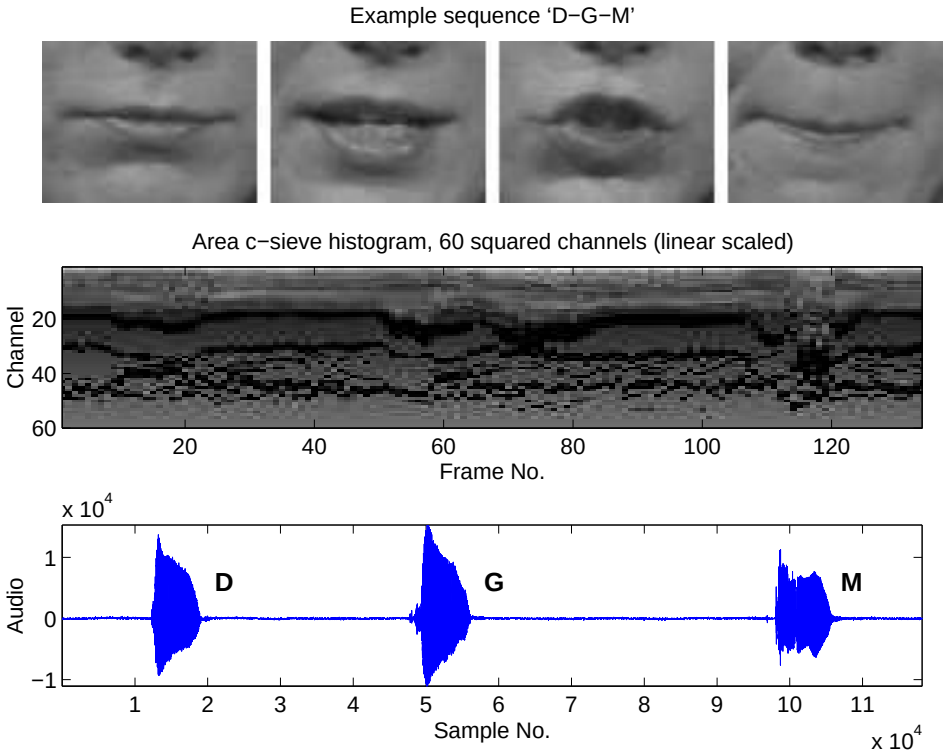


Figure 4.24: Area histogram sequence using channels spaced using a squared curve. Top row shows example images from the sequence of isolated letters, 'D-G-M'. Middle row plots the area histogram, the number of granules found at each of 60 channels, spaced on a squared axis, in the granularity decomposition. Bottom row is the time aligned audio waveform.

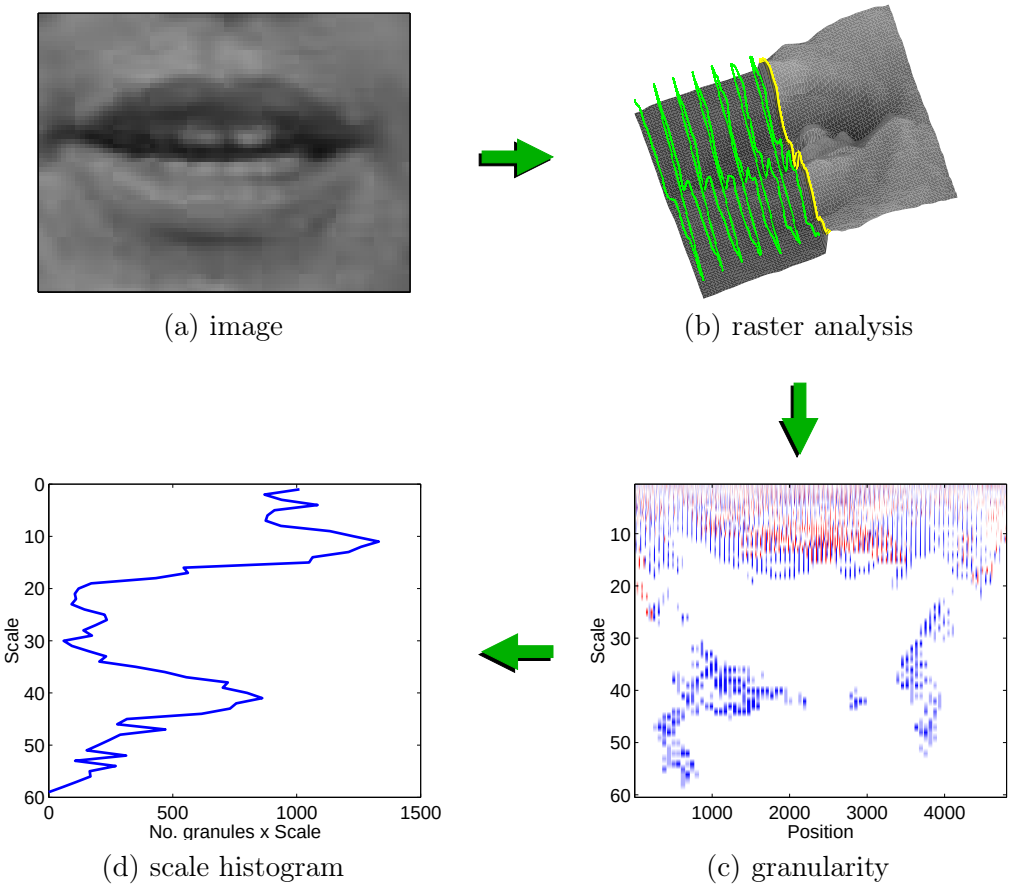
An alternative approach is to apply a 1D length-sieve to the image. A 1D sieve is more stable in smooth low contrast signal regions than its 2D equivalent because 1D connected regions are less able to merge or split due to pixel noise. A 1D analysis of a 2D image can be obtained by raster scanning, i.e. for a vertical scan start at the top left and scan down the first column then repeat starting at the top of the second column and so on. The resulting granularity describes the image in terms of granules that measure amplitude, scale and position relative to the raster scan. All of the sieve properties are maintained, and the image is simply treated as a series of one dimensional signals. An example image is shown in Figure 4.25(a) and a vertical scan line highlighted in yellow in Figure 4.25(b) with the preceding scan lines amplitude compressed to demonstrate the piecewise 1D nature of this decomposition. The granularity is shown in Figure 4.25(c) plotting scale on the vertical axis and vertical raster position relative to the top left of the image along the horizontal. Granules for this bipolar recursive median m-sieve decomposition are plotted in red for positive amplitude and blue for negative. The maximum scale is determined by the longest raster line. For this example, the maximum scale is the height of the image, 60 pixels. A 1D length decomposition is significantly more compact over scale than a 2D area decomposition.

A full 1D decomposition maintains all of the information in the image. The invertibility property, Equation (4.25), means that the original image can be reconstructed by summing the granules plotted in Figure 4.25(c). The 1D decomposition separates the image features out according to length, in this case vertical length. Other 1D decompositions are possible by changing the direction of the raster scanning, for example horizontal or at an arbitrary angle. For lipreading, where most mouth movement is up and down, it is preferable to scan the image vertically.

In the same way as area-histograms were formed by counting the number of granules in each 2D channel, *length-histograms* measuring the distribution of vertical length in the image can be formed by summing the number of granules present at each scale. Normally, length-histograms will be referred to as *scale-histograms* in this thesis. The relatively small number of scales of the 1D decomposition means the grouping into channels required to make the 2D analysis manageable is not necessary. A scale histogram is shown in Figure 4.25(d), plotting scale on the vertical axis and number of granules along the horizontal. This can be visualised as summing along the horizontal, position, axis of Figure 4.25(c). A scale histogram formed in this way is a low dimensional representation of the overall shape of the mouth in the image. It is biased in the direction of most expected motion to be most sensitive to vertical changes in the image.

If amplitude information is ignored by simply counting the number of granules at each scale then the scale histogram is relatively insensitive to lighting conditions. An overall brightening or dimming of the image is unlikely to significantly change the granularity decomposition because it is the relative amplitudes of the pixels that define the signal extrema. Until greylevel quantisation or clipping effects are seen, a scale histogram is in practice very stable to varying lighting conditions. However, significant information is described by the amplitude of granules, for example when the granularity is inverted the low amplitude granules have very little observable effect in the image. An obvious extension to a scale count histogram (shortened to sh) is summing the amplitudes at each scale (a). As amplitudes are generally bipolar, further alternatives are to sum the absolute ($|a|$) or squared amplitudes (a^2). Figure 4.26 compares each method.

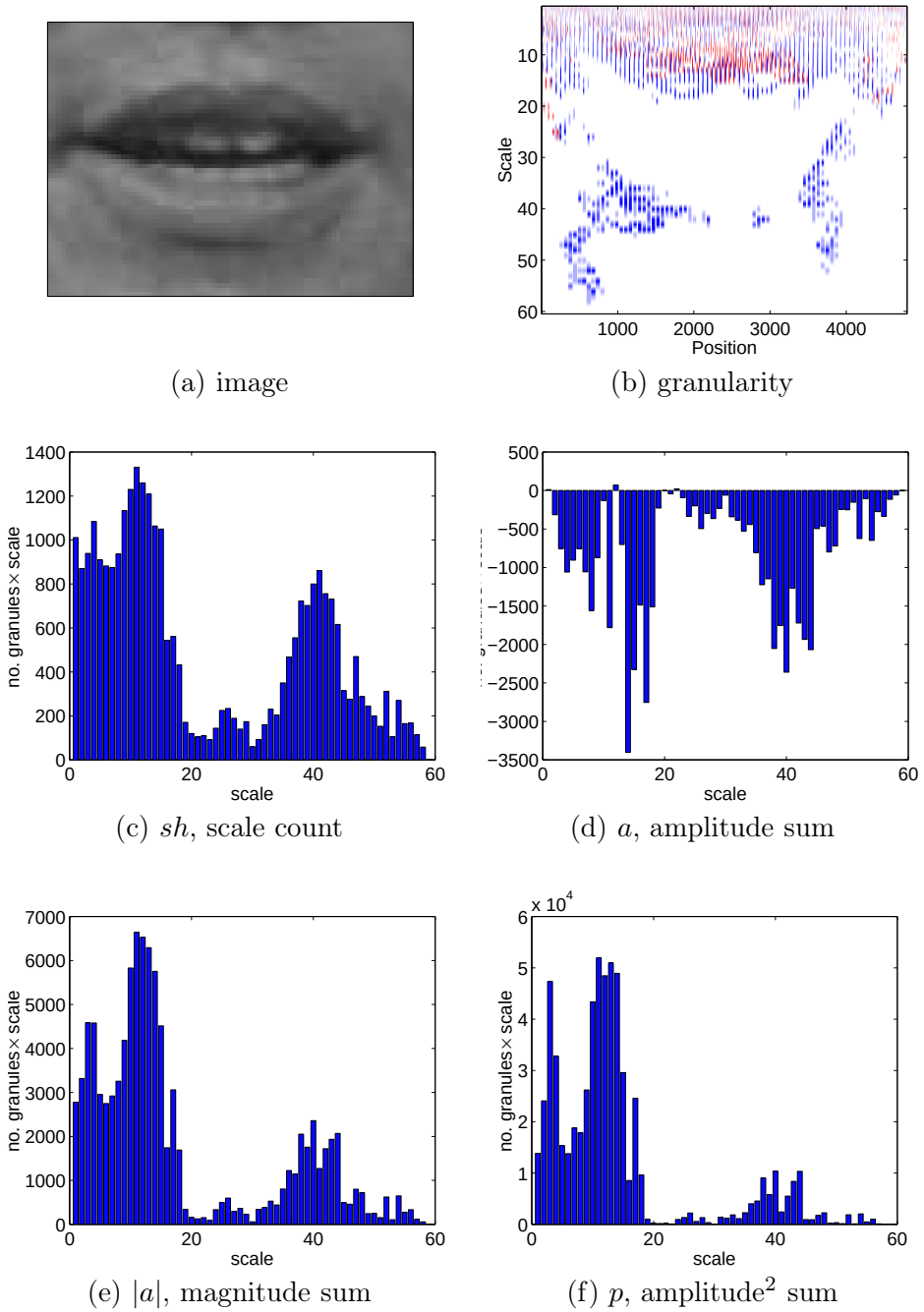
By summing or counting the number of granules over position, a scale-histogram is relatively insensitive to image translations that do not introduce significant new image features, more so in the less sensitive horizontal direction. The most significant problem in practice is due to image scaling. Both scale and area-histograms are a measure of the scale distribution



(a) image (b) raster analysis

(d) scale histogram (c) granularity

Figure 4.25: Multiscale spatial analysis. The image (a) is vertically raster scanned. An example cut away (b) highlights a single slice in yellow and some previous scanlines in green. The entire image is decomposed by scale (vertical length) into granularity (c) using an m-sieve. Positive granules are plotted in red, negative in blue. A scale histogram formed by summing or counting over position at each scale (d).



(a) image (b) granularity

(c) sh, scale count (d) a, amplitude sum

(e) $|a|$, magnitude sum (f) p, amplitude^2 sum

Figure 4.26: Scale histogram types. The image (a) is decomposed using and m-sieve, (b). The scale count histogram, sh, is plotted in (c), the amplitude sum histogram, a, in (d), the magnitude sum histogram, $|a|$, in (e) and the amplitude^2 sum histogram, a^2, in (f).

in the image. Motion in the z-plane, toward or away from the camera, shifts both 1D and 2D decompositions through scale.

Any type of sieve can be used for a 1D rasterised image decomposition. As discussed in the previous section a closing c-sieve is biased to only process negative extrema which are often associated with the dark mouth cavity. A recursive median, m-sieve, is statistically more robust and, by processing bipolar extrema, may be less sensitive to variations between talkers appearance. Figure 4.27 shows example granularity decompositions and scale count (sh) histograms for m- o- and c-sieves.

Extracting lipreading features in this way is abbreviated to Multiscale Spatial Analysis (MSA) to highlight the low-level scale based analysis used to derive lipreading features. The scale-histogram obtained by counting the number of granules at each scale from an m-sieve is shown in Figure 4.28 for the D-G-M sequence.

4.5 Principal Component Analysis

The 2D sieve derived area-histograms and 1D (length) scale-histograms discussed in Section 4.4 extract feature vectors from mouth images that are rather large. Both examples shown form 60 dimensional vectors from the 80×60 mouth images of the AVletters database. For the larger 100×75 images of the Tulips database vertical scale-histograms are 100 dimensional vectors.

For recognition a smaller feature space is desired as this allows less complex statistical models that are easier to train. Principal component analysis (PCA) is a statistical method for identifying orthogonal directions ordered by their relative variance contribution to the multidimensional data. PCA is a linear transform and each direction is a linear combination of the original data. The result is an orthogonal rotation of the axes to align the first in the direction of most variance and so on by decreasing variance contribution. If the data is correlated this transforms into a linearly decorrelated space. If there is no correlation PCA will simply order the axes by variance. The values of the original data transformed along the top N directions can be used as decorrelated features in a reduced N-dimensional transformed feature space. Appendix A has the full mathematical details of PCA.

A problem with using PCA on scale-histograms is that although the coefficients are all measures of scale they do not have similar variance. There are typically many more small scale objects in images than large and these often directly represent pixel-level noise. These will be identified as the major axes of variance and any correlation between small and large scale coefficients is lost. This is an example of the PCA scaling problem discussed in Section A.4. The usual solution is to assume all variables are equally significant and normalise for variance by calculating PCA using the correlation matrix rather than the covariance matrix. However, if the variables are not equally important then this is not recommended [43]. As the relative importance of each scale for lipreading is unknown both methods were used to derive PCA features from scale-histograms.

An example transformation calculated using the covariance matrix and taking the top twenty rotated directions is shown in Figure 4.29 for a concatenated letter sequence.

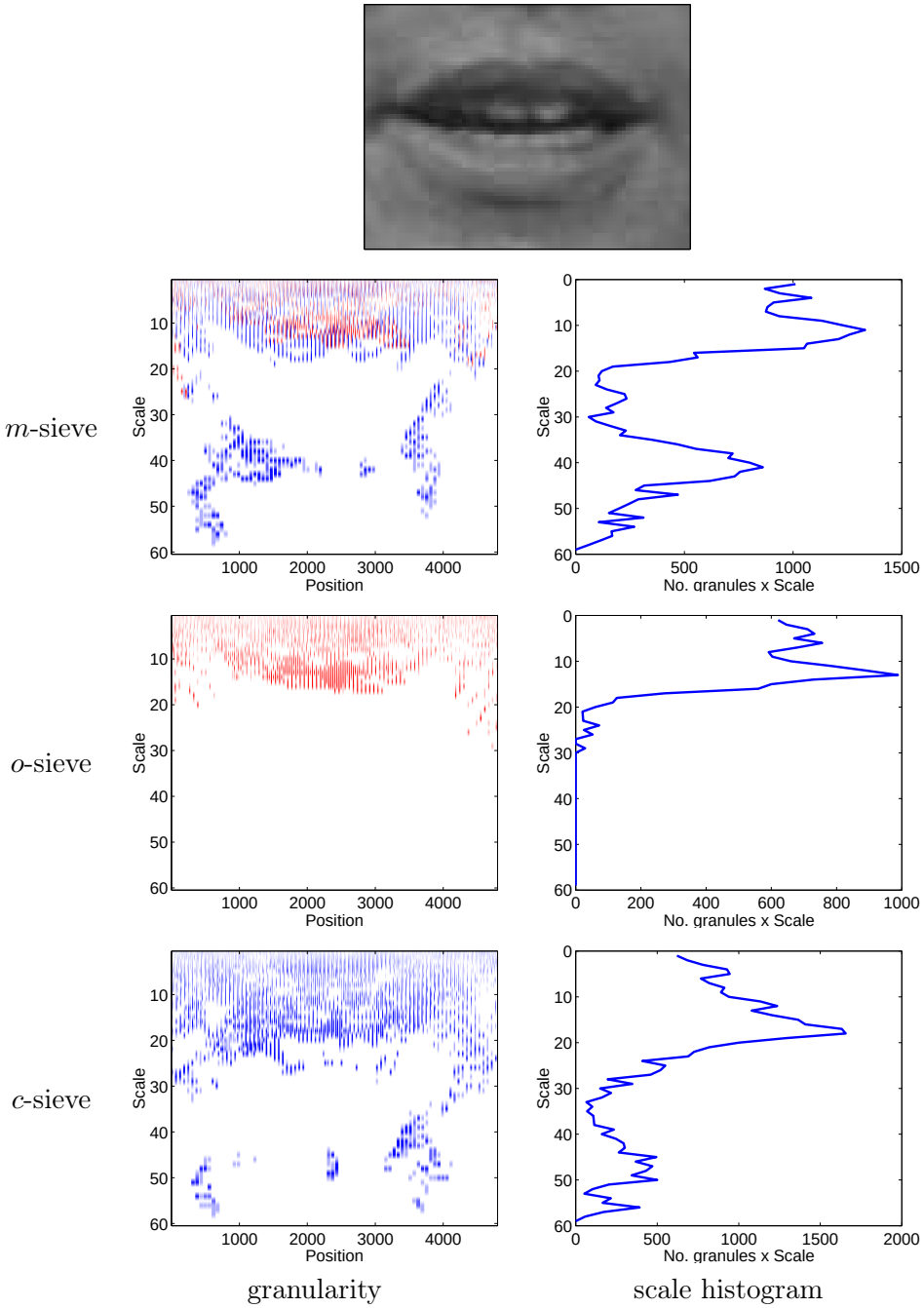


Figure 4.27: Scale histogram sieve types. The full granularity of the image is plotted for recursive median, m, opening, o, and closing, c, sieves. Granules are red for positive and blue for negative amplitude. The number of granules at each scale are plotted on the scale-histograms.

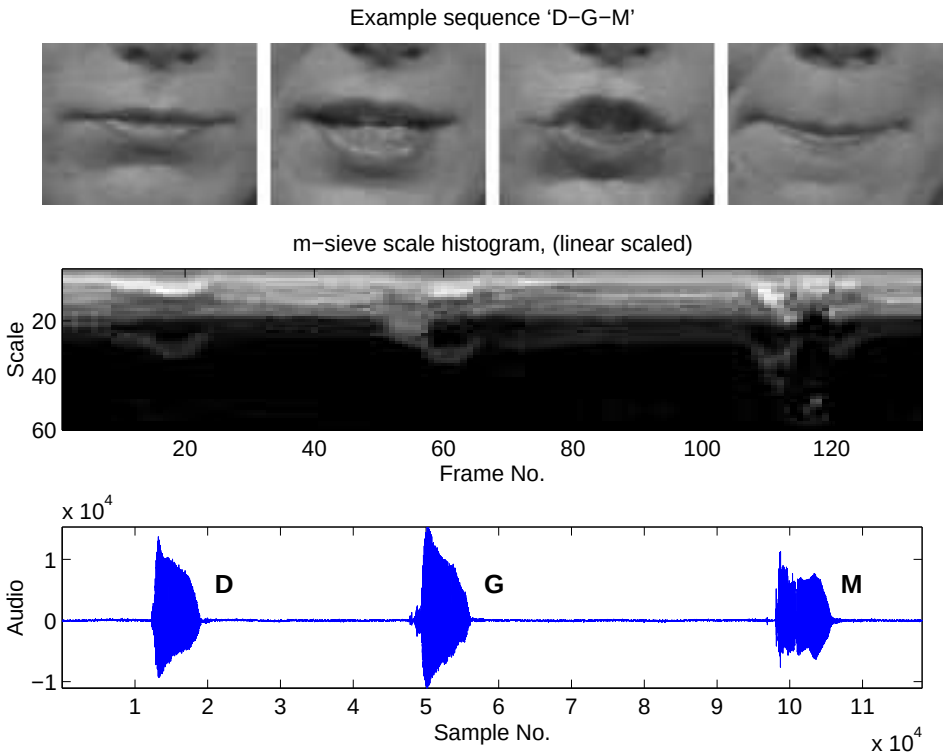
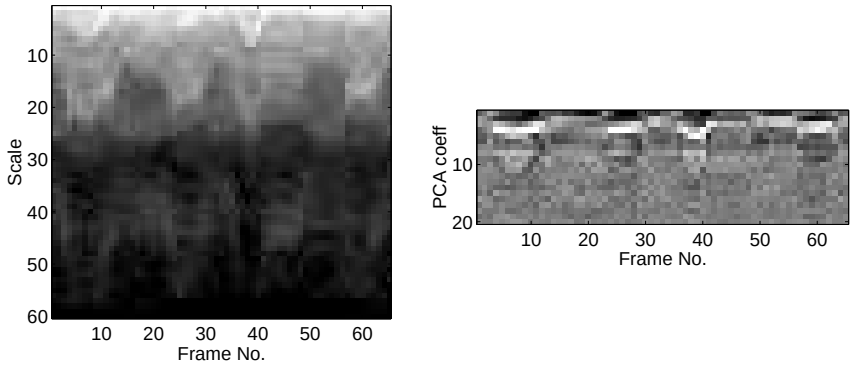


Figure 4.28: Scale count histogram sequence. Top row shows example images from the sequence of isolated letters, 'D-G-M'. Middle row plots the scale count histogram (sh), the number of granules found at each of the 60 scales of the m-sieve vertical decomposition of the 80×60 images. Bottom row is the time aligned audio waveform.



(a) scale-histogram sequence (b) PCA transformed sequence

Figure 4.29: Example PCA transformed scale-histogram sequence. The 60 dimensional scale count histogram (sh) from an m-sieve decomposition for concatenated isolated letters 'A-V-S-P', (a), is transformed using the top 20 directions of PCA analysis using the covariance matrix calculated over the entire database, (b).

Chapter 5

High-Level Visual Speech Analysis

This chapter contrasts the previous low-level analysis chapter and describes two related high-level, model-based methods. The model-based approach asserts some prior decision of what the important visual speech features are. Typically the shape of the lip contours is used because they are the most prominent features [1,9,48,50,54,58,82,91,92,107,109,168,190,197].

A model-based system can only extract information about itself. Typically the model is imposed onto the data even if it does not support it and it is not difficult for a lip contour model to converge into a sharply defined nostril cavity or prefer the teeth or tongue boundary to the real lip/face contour. Even if the model normally extracts relevant features, under such failure conditions little useful information is obtained. However, if a good model is chosen, and it can be reliably applied to the data, the high-level method should be more robust. By definition it should ignore all the (generally assumed) irrelevant information.

The first method described uses Active Shape Models (ASM's) to track the inner and outer lip contour. These were first formulated by Cootes [50, 54] and applied to lipreading by Luettin [107, 109]. Their application here extends that of Luettin with the addition of a multi-resolution search and application to a more complex task. The second method is a recent extension of ASM's by Cootes [49] that combines statistical shape and greylevel appearance in a single unified Active Appearance Model (AAM).

5.1 Active Shape Model Lip Tracking

Active shape models are a high-level, model based, method of extracting lip shape information from image sequences. An active shape model (ASM) is a shape constrained iterative fitting process. The shape constraint comes from the use of a statistical shape model, also known as a point distribution model (PDM), that is defined from the statistics of hand labelled training data. The PDM describes the space of valid lip shapes, in the sense of the training data, and points in this reduced space are statistically compact representations of lip shape that can be directly used for lipreading. Section 5.1.1 describes how PDM's were built from training data of both AVletters and Tulips databases.

In order to iteratively fit a PDM to an example image, a cost function is required that can be evaluated to determine the current goodness of fit. In keeping with Luettin [107] a model of the concatenated gray level profiles of the normals of each point of a shape model is used. This model is formed in the same way as the PDM and called a grey level profile distribution model (GLDM). Section 5.1.2 describes how these can be built.

A downhill simplex minimisation is used to iteratively fit a PDM and associated GLDM to an example image (this method is also used by Luettin). This finds the local minima of the cost function in terms of the pose and shape parameters (PDM weights) of the current

estimate of lip shape and position. Section 5.1.3 describes this process and Section 5.1.5 extends it using a coarse to fine multi-resolution image search.

A narrowing of the search space can be achieved by building models independently for each talker so the tracker is constrained over a smaller shape space. The shape parameters obtained can then be transformed into the global shape space to allow multi-talker recognition. Section 5.1.6 describes this further.

5.1.1 Point Distribution Model

A point distribution model (PDM) is a statistical model of shape calculated from a set of training images in which landmark points have been located. Here, landmark points must be hand located but schemes to automate this process have been considered [86]. Each example shape model is represented by the (x, y) coordinates of its landmark points, which must have the same meaning in all training examples, i.e. all images have the landmark points labelled in the same order on the same image features. The inner and outer lip contour model used is shown in Figure 5.1 and consists of 44 (x, y) points. The outer contour is made up of 24 points and the inner 20. The model used is arbitrary but should be meaningful to the operator that will label the training images so that landmark points can be reliably placed.

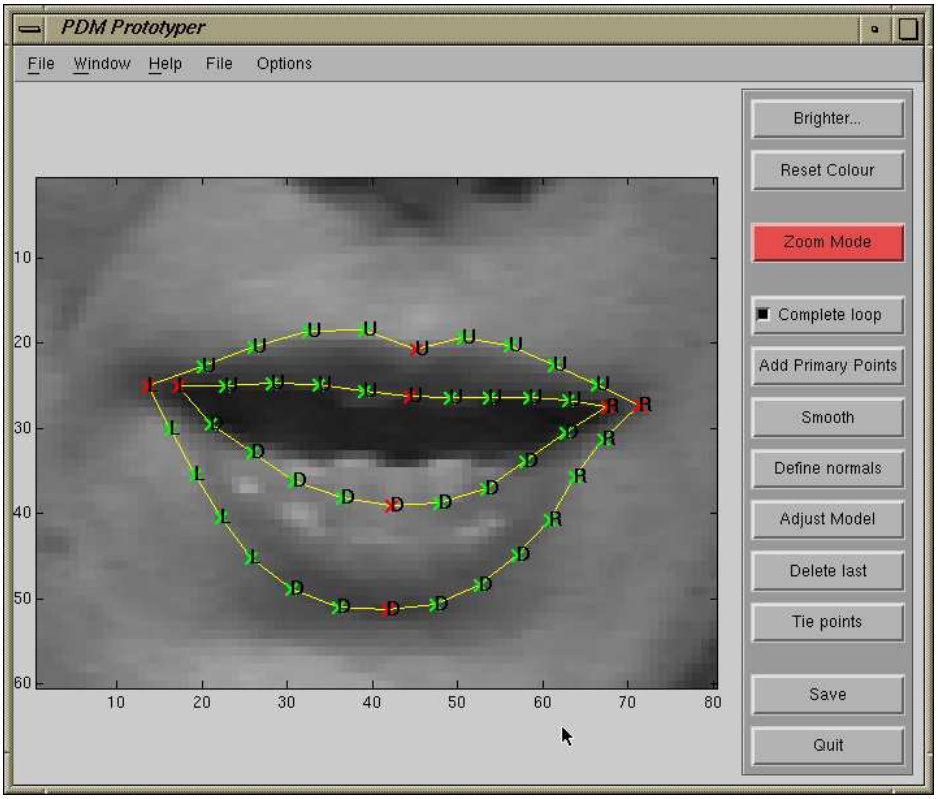


Figure 5.1: Inner and outer lip contour model definition. Red crosses are primary landmark points, secondaries are green. The letter indicates the direction of a normal calculated through that point.

The landmark points are split into two groups, primary points, shown in red in Figure 5.1, and secondary points, shown in green. The primary points are those that the operator feels can be reliably positioned in all training images. The secondary points are spaced equidistant between between primary points to define the overall shape. Because the outer contour is

longer than the inner more secondary points are used, the extra used in the longer lower contour.

It is imperative that points are consistently placed during training otherwise the statistical shape analysis will identify labelling errors as sources of significant variance. To reduce this problem spline interpolation is used to fit a curve through the secondary points between pairs of primaries, and the secondaries are repositioned evenly along this curve. This is a simple method of reducing the labelling error caused by not evenly distributing the secondary points—something extremely hard for a human operator to do. During this smoothing process no primary points are moved as by definition these are points that can be reliably positioned.

All model placing is done interactively using a GUI written in MATLAB and shown in Figure 5.2. The lip shape model template shown in Figure 5.1 is initialised for each new image. This puts all the points that must be placed on to the image. These can then be dragged into the desired position with special care taken of the eight primary points. Spline curve smoothing may be used to more evenly distribute the secondary points and all points can be moved and smoothed again as necessary. The resulting 44 (x, y) points are saved for each training image.

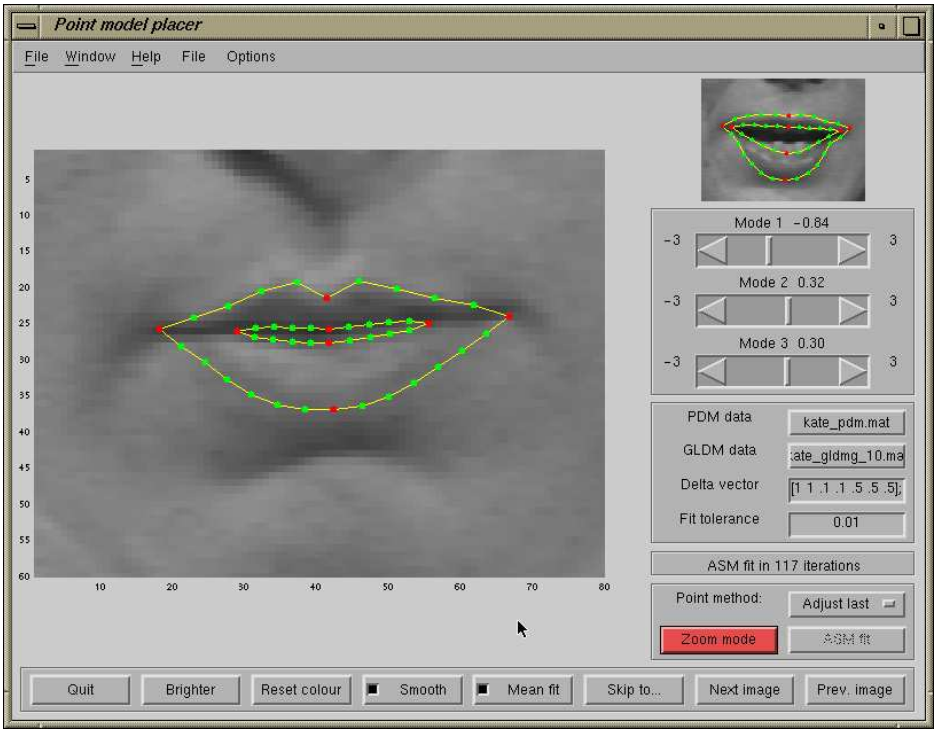


Figure 5.2: Landmark point placing. The lip model defined in Figure 5.1 is initialised onto a new image and dragged into the correct position. An estimate of the correct positions can be made using models calculated from the previously placed landmark points.

The extra information seen on the right cluster of Figure 5.2 is a semi-automatic point placing system. When several training examples have been placed shape and greylevel models can calculated and the iterative fitting process that will be described in Section 5.1.3 can be used to attempt to fit the points to the image. These can then be hand edited from often nearly accurate positions and this greatly speeds the training process.

It is not always possible to correctly label the training images. The Tulips database often has images where the lower lip contour extends out of the image and in these cases landmark

points were placed at the edge of the image. Some talkers in the AVletters database, that does not have so direct lighting, have extremely poor lower lip contours. In these cases point placement is unavoidably somewhat subjective. Where possible the primary points are placed accurately in the corners and midpoints of the inner and outer lip contours and the secondaries evenly along the lip contours.

Given a set of labelled images the statistics describing shape variation can be calculated. To consider only shape variation all landmark point models must first be aligned to the same axes so pose variation (translation, rotation and scaling) in the training set is removed. The following is identical to that described by Cootes *et. al* [54]. The ith shape model is,

$$\mathbf{x}_i = (x_{i1}, y_{i1}, x_{i2}, y_{i2}, \ldots, x_{iN}, y_{iN})^T \tag{5.1}$$

where N is 44 for the inner and outer lip contour model used here. Two similar point models $\mathbf{x}_1$ and $\mathbf{x}_2$ are aligned by minimising,

$$E = (\mathbf{x}_1 - M(s,\theta)[\mathbf{x}_2] - \mathbf{t})^T \mathbf{W} (\mathbf{x}_1 - M(s,\theta)[\mathbf{x}_2] - \mathbf{t}) \tag{5.2}$$

where the pose transform for scale, s, rotation, θ, and translation (t_x, t_y) is,

$$M(s,\theta) \begin{bmatrix} x_{jk} \\ y_{jk} \end{bmatrix} = \begin{pmatrix} (s\cos\theta)x_{jk} - (s\sin\theta)y_{jk} \\ (s\sin\theta)x_{jk} + (s\cos\theta)y_{jk} \end{pmatrix} \tag{5.3}$$

$$\mathbf{t} = (t_{x1}, t_{y1}, \ldots, t_{xN}, t_{yN}) \tag{5.4}$$

and $\mathbf{W}$ is a diagonal weight matrix for each point. The weights are chosen so that points that have the most variance are weighted least and stable, low variance, points have greater contribution. If R_{kl} is the distance between points k and l and $V_{R_{kl}}$ the variance of R_{kl} then the kth weight is,

$$w_k = \left(\sum_{l=1}^{N} V_{R_{kl}} \right)^{-1} \tag{5.5}$$

To align the set of training models an iterative algorithm is used. First all models are translated, rotated and scaled to align with the first model. The mean shape is calculated and translated, rotated and scaled back toward the first shape. All point models are realigned to the transformed mean and the mean recalculated. This repeats until there is no significant change in the mean shape model. This is different from normalising each individual model, which would artificially scale all examples to some fixed value.

Given the set of M aligned shape models the mean shape can be calculated,

$$\overline{\mathbf{x}} = \frac{1}{M} \sum_{i=1}^{M} \mathbf{x}_i \tag{5.6}$$

The axes that describe most variance about the mean shape can be calculated using a principal component analysis (PCA), Appendix A, on the aligned shape models. This identifies the major axes of shape variation, or modes of variation. Any valid shape, in the sense of the training data, can be approximated by adding a reduced subset, t, of these modes to the mean shape,

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s \tag{5.7}$$

where $\mathbf{P}_s$ is the matrix of the first t eigenvectors, $\mathbf{P}_s = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_t)$ and $\mathbf{b}_s$ is a vector of t weights, $\mathbf{b}_s = (b_1, b_2, \ldots, b_t)^T$. As the eigenvectors are orthogonal, $\mathbf{P}_s^T \mathbf{P}_s = \mathbf{I}$ and the shape

parameters can be calculated given an example set of points, $\mathbf{x}$,

$$\mathbf{b}_s = \mathbf{P}_s^T(\mathbf{x} - \overline{\mathbf{x}}) \tag{5.8}$$

This allows valid lip shapes to be defined in a compact statistically derived shape space. The number of modes of variation is much less than the number of landmark points used as the number of points is chosen to clearly define lip shape and they are highly correlated. There are no PCA scaling problems such as the ones discussed in Section 4.4 as all variables are either x or y values in square image coordinate axes. The order of the PDM is chosen so that the first t eigenvalues of the covariance matrix describe 95% of the total variance,

$$\frac{\sum_{i=1}^{t} \lambda_i}{\sum_{i=1}^{2N} \lambda_i} \geq 0.95 \tag{5.9}$$

where λ_i is the ith largest eigenvalue of the covariance matrix,

$$S = \frac{1}{M} \sum_{i=1}^{M} (\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T \tag{5.10}$$

The values of the shape parameter vector $\mathbf{b}_s$ are constrained to lie within three standard deviations of the mean for each coefficient,

$$-3\sqrt{\lambda_k} \leq b_k \leq 3\sqrt{\lambda_k} \tag{5.11}$$

This allows lip shapes to be synthesised by choosing values up to plus or minus three standard deviations for any of the shape coefficients. A MATLAB GUI that allows the operator to move a slider to alter the value of the shape parameter for any mode of variation is shown in Figure 5.3. This also displays preset 'walks' around the shape space by moving each slider in turn from one extent to the other.

The PDM calculated from 1,144 labelled training images of the AVletters database is shown in Figure 5.4. All frames of the first utterances of A, E, F, M and O for all ten talkers were used as training data. Each mode is plotted at plus and minus two standard deviations from the mean on the same axes. Seven modes of variation were required to capture 95% of the variance.

The PDM calculated from 223 labelled training images of the Tulips database is shown in Figure 5.5. The first utterances of 1 and 3 for all twelve talkers formed the training set. Each mode is again plotted at plus and minus two standard deviations from the mean on the same axes. Like the AVletters database PDM seven modes of variation were required to capture 95% of the variance. This differs from Luettin's [107] ASM implementation which required ten modes to describe 84% of variation in 250 images hand labelled with a 38 point inner and outer lip contour model. The more compact model presented here is probably due to the interactive spline smoothed placing of the secondary points. This was found to remove much of the operator point placement error and so produce much more compact models.

Although in general no direct physical meaning can be attributed to the eigenvectors obtained using PCA, the first two modes of both PDM's clearly describe the degree of vertical and horizontal mouth opening. The third mode in both cases models the 'smile' as the mouth corners move and the remaining modes account for other pose and talker variations and lip

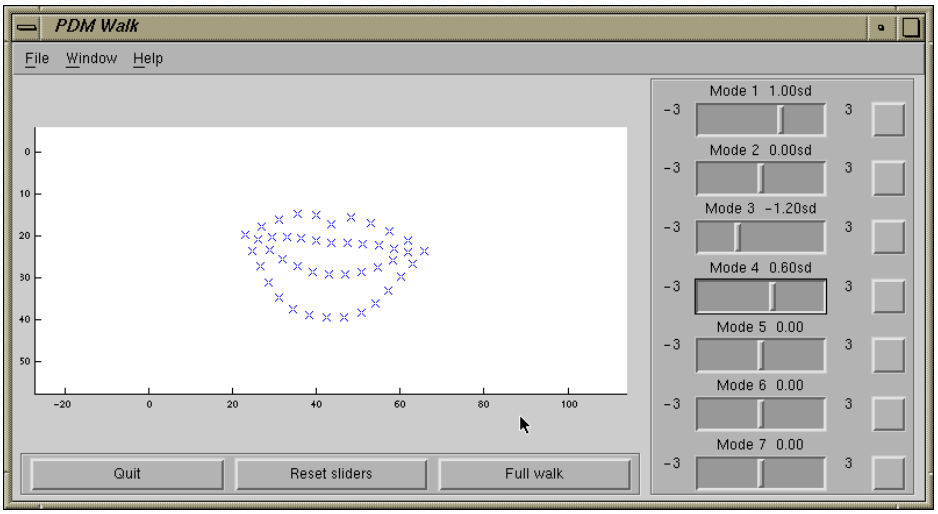


Figure 5.3: Point distribution model viewer. Valid lip shapes can be chosen by selecting the shape parameter values between ± 3 standard deviations using the sliders.

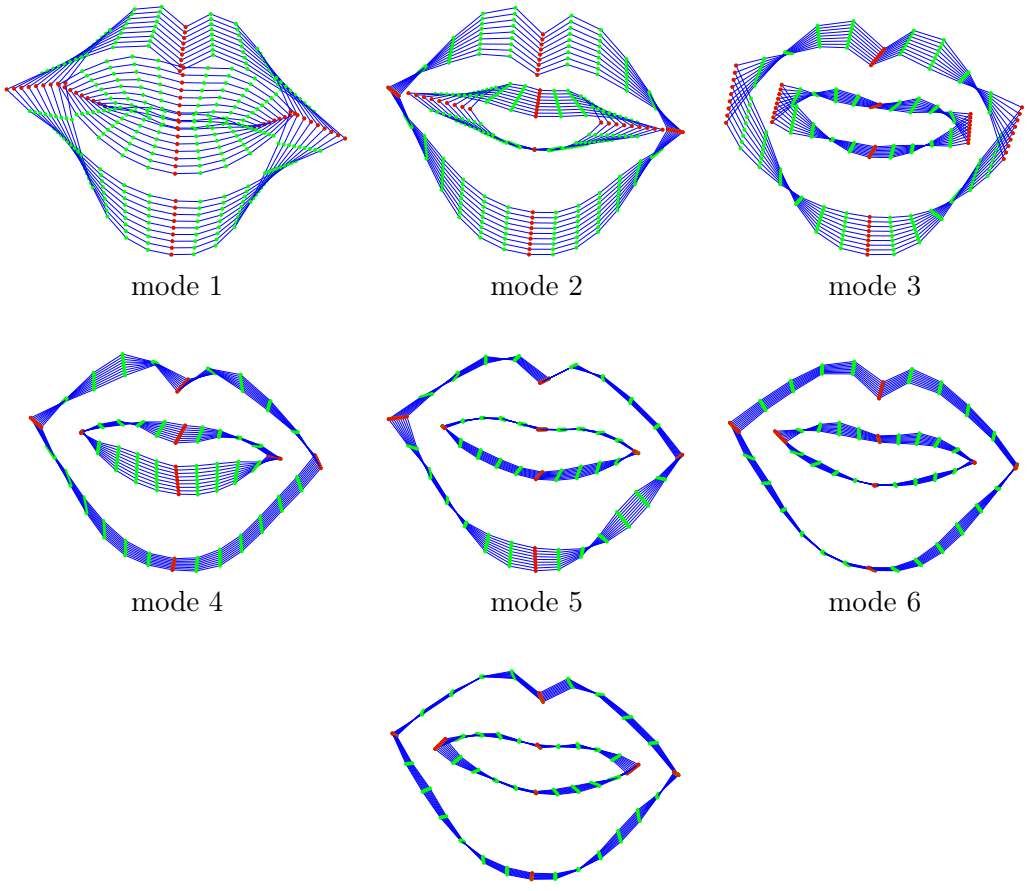


Figure 5.4: PDM for the AVletters database. Each mode is plotted at $\pm 2\sigma$ about the mean. The seven modes of variation describe 95% of the variance of the training set of letters A, E, F, M and O for all ten talkers.

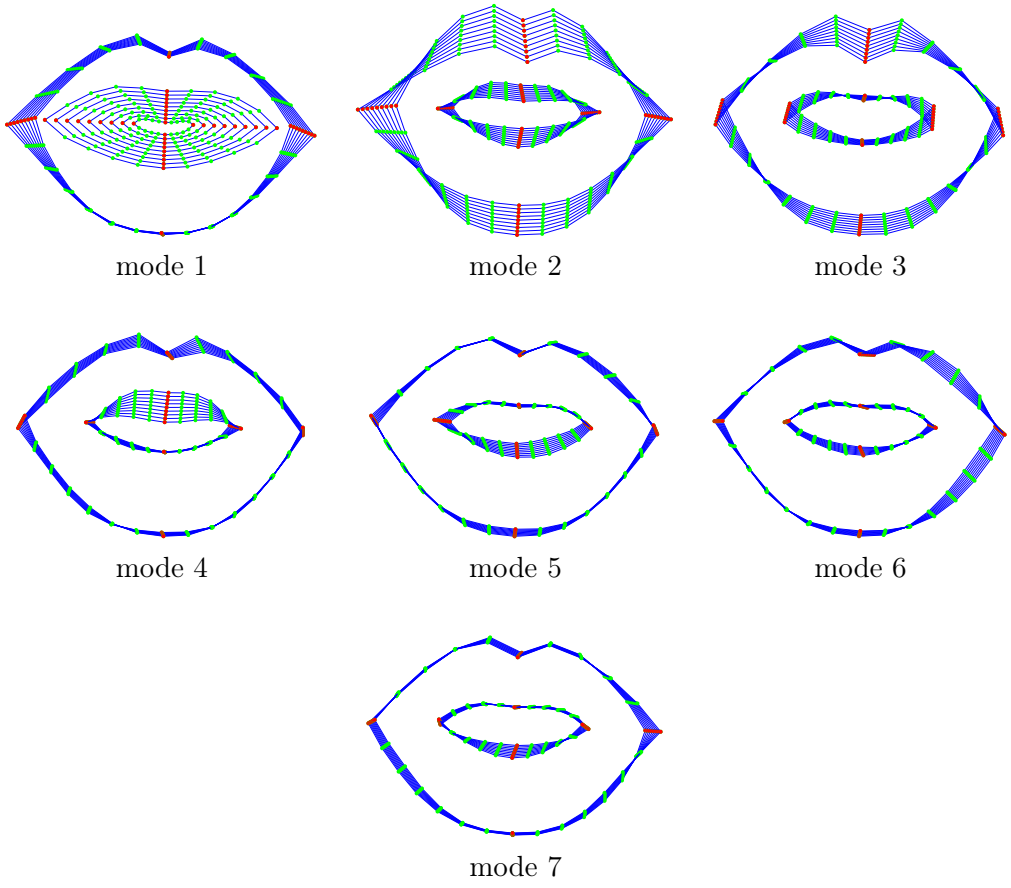


Figure 5.5: PDM for the Tulips database. Each mode is plotted at $\pm 2\sigma$ about the mean. The seven modes of variation describe 95% of the variance of the training set of digits 1 and 3 for all twelve talkers.

shape asymmetry. It is reassuring that the number and effect of the modes of variation are essentially the same when trained independently on two distinct databases.

The PDM for the AVletters database appears rotated because the pose alignment is initialised on the first image, see Figure 3.2. The alignment was not forced to have zero rotation.

5.1.2 Grey Level Profile Distribution Model

To fit a PDM to an image a cost function is required that can be evaluated at each stage of an iterative minimisation to determine the current goodness of fit. The iterative fitting of a PDM is an Active Shape Model (ASM). The original implementation [54] moved individual points towards strong edges and then imposed the PDM shape constraints until the model converged. Alternatively the statistics of the greylevel profiles at the normals of the model points can be used [50] or a statistical model, analogous to a PDM, can be calculated for each profile [52].

Following [81, 107, 109], the greylevel profiles are modelled by concatenating the profile at the normal of each model point into a single vector. This allows the use of PCA to calculate a single statistical model of the greylevel normals of the shape model and so implicitly take into account any correlation between the greylevels at different points. Figure 5.6 plots eleven pixel length normals about each model point. Care must be taken that normal vectors are always calculated in the same direction: the cyan dots indicate this direction.

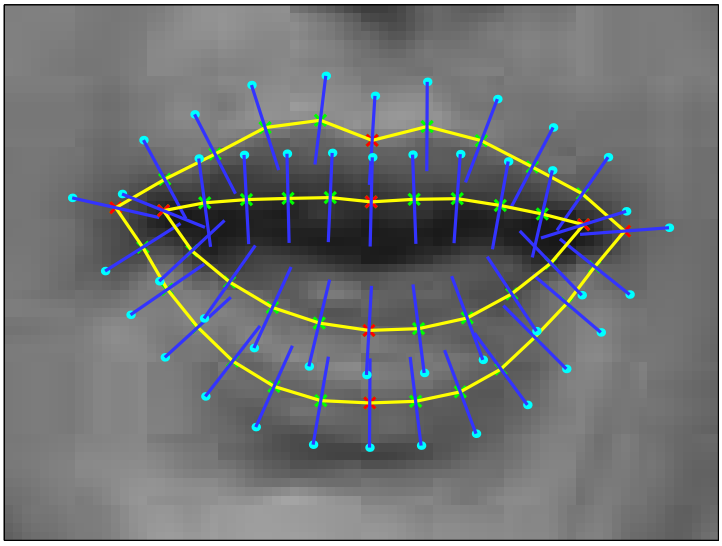


Figure 5.6: Collecting greylevel normal profiles. Cyan dots indicate normal direction.

The concatenated greylevel profiles in this example form a $44 \times 11 = 484$ length vector. In the same way that PCA was used to calculate the PDM, a greylevel normal profile distribution model (GLDM) can be calculated,

$$\mathbf{x}_p = \overline{\mathbf{x}}_p + \mathbf{P}_p \mathbf{b}_p \tag{5.12}$$

The order of the GLDM is chosen such that t_p modes describe 95% of the variance. The resulting GLDM's have 71 modes for the AVletters database and 59 modes for the smaller Tulips database. As with the PDM the GLDM weights are constrained to lie within ± 3

standard deviations of the mean.

The first three modes of the GLDM for the AVletters database are shown in Figure 5.7 at ± 2 standard deviations about the mean. For easier viewing, the profiles have been widened and the image smoothed to give the appearance of a full greylevel image, the GLDM only models single pixel width normals at each point.

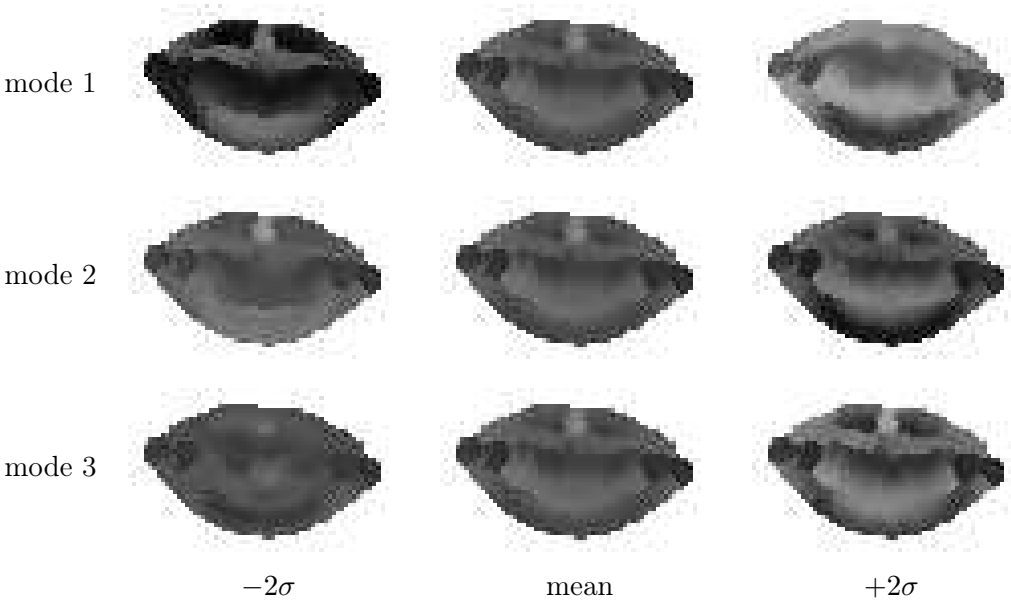


Figure 5.7: First three modes of the GLDM for the AVletters database. Each mode is plotted at ± 2 standard deviations about the mean.

As with the PDM, Equation (5.8), the model weights vector for a given concatenated greylevel profile vector can be calculated using,

$$\mathbf{b}_p = \mathbf{P}_p^T(\mathbf{x}_p - \overline{\mathbf{x}}_p) \tag{5.13}$$

5.1.3 Simplex Fitting

Cootes [54] models the greylevels for each individual landmark point and calculates the model update on a point-wise basis. The PDM shape constraint is applied after the new pose is calculated from the point-wise suggested update. A simpler fitting algorithm was used by Luettin [107] who used the pose and shape parameters as parameters for a downhill simplex function minimisation. The simplex algorithm [146] does not require calculation of the gradient of the error surface but may require many iterations to converge to a local minimum.

The simplex is a geometric construct formed in N dimensions as $N + 1$ vertices with interconnecting lines. For a 2D problem the simplex is a triangle, for 3D a square and so on for higher dimensional problems. The simplex is initialised with $N + 1$ points that represent the initial perturbations about the starting point. At each iteration a cost function is evaluated for all points in the simplex and it is either; reflected away from the highest error point, reflected and expanded away from the high point, contracted in one dimension toward the low point or contracted in all dimensions toward the low point. These operations allow the algorithm to find the local minimum by 'oozing' over the high dimensional error surface until the simplex has collapsed into a minimum, or a maximum number of iterations

is reached. The simplex algorithms used were initially the standard MATLAB `fmins` function and later, for the speed increase obtained using compiled C code, the Numerical Recipes in C function `amoeba` [158].

For iterative ASM fitting a simplex is formed in a combined PDM pose and shape parameter space, $(t_x, t_y, s, \theta, b_1, b_2, \ldots, b_t)$. The cost function is evaluated for each point in the simplex at each iteration and ideally has a minimum only at the correct model shape and position. The cost function used calculates sum of squares error of the GLDM and is a measure of how well the greylevel profiles about the current model points match those seen in the training set of hand located points.

The weight parameters $\mathbf{b}_p$ can be calculated for an example concatenated profile vector $\mathbf{x}_p$ using Equation (5.13) to find the best approximation to the current concatenated greylevel profile given the GLDM, $\hat{\mathbf{x}}_p$,

$$\hat{\mathbf{x}}_p = \overline{\mathbf{x}}_p + \mathbf{P}_p\mathbf{b}_p \tag{5.14}$$

There is some error introduced due to the reduced t_p modes of the GLDM,

$$\begin{aligned} \mathbf{e} &= \mathbf{x}_p - \hat{\mathbf{x}_p} \\ &= (\mathbf{x}_p - \overline{\mathbf{x}}_p) - \mathbf{P}_p\mathbf{b}_p \end{aligned} \tag{5.15}$$

The sum of squares error between the model and the profile, E^2, can calculated noting that, $\mathbf{e}^T = (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T - \mathbf{b}_p^T\mathbf{P}_p^T$, $\mathbf{b}_p^T = (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T\mathbf{P}_p$ and $\mathbf{P}_p^T\mathbf{P}_p = \mathbf{I}$,

$$\begin{aligned} E^2 &= \mathbf{e}^T\mathbf{e} \\ &= ((\mathbf{x}_p - \overline{\mathbf{x}}_p)^T - \mathbf{b}_p^T\mathbf{P}_p^T)((\mathbf{x}_p - \overline{\mathbf{x}}_p) - \mathbf{P}_p\mathbf{b}_p) \\ &= (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T(\mathbf{x}_p - \overline{\mathbf{x}}_p) - (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T\mathbf{P}_p\mathbf{b}_p - \mathbf{b}_p^T\mathbf{P}_p^T(\mathbf{x}_p - \overline{\mathbf{x}}_p) + \mathbf{b}_p^T\mathbf{P}_p^T\mathbf{P}_p\mathbf{b}_p \\ &= (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T(\mathbf{x}_p - \overline{\mathbf{x}}_p) - \mathbf{b}_p^T\mathbf{b}_p - \mathbf{b}_p^T\mathbf{b}_p + \mathbf{b}_p^T\mathbf{b}_p \\ &= (\mathbf{x}_p - \overline{\mathbf{x}}_p)^T(\mathbf{x}_p - \overline{\mathbf{x}}_p) - \mathbf{b}_p^T\mathbf{b}_p \end{aligned} \tag{5.16}$$

The fit measure proposed by Cootes [52] is based on the Mahalonobis distance [118] and also considers error introduced by only using the top t_p modes (the second term) by assuming $\lambda_j = 0.5\lambda_{t_p}$ for $j > t_p$,

$$F = \sum_{j=1}^{t_p} \frac{b_{p_j}^2}{\lambda_j} + \frac{2E^2}{\lambda_{t_p}} \tag{5.17}$$

Luettin [107] rejects this measure for its penalisation of values far from the mean and argues that for lipreading, the greylevel profiles vary greatly during speech and between talkers. He proposes that the sum of squares error, Equation (5.16), is used as the cost function, with the values of $\mathbf{b}_p$ clipped to lie within ± 3 standard deviations, accounting for 99% of greylevel profile variation.

Figure 5.8 shows the result of using E^2 as the cost function for a simplex minimisation. The fit process was initialised using the mean shape in the centre of the image with zero rotation and unity scale. The simplex was initialised to perturb from this position by a translation of five pixels in both x and y directions, rotationally by 0.1 radians, with a 10% scale increase and by 0.5 of a standard deviation for each of the seven modes of variation of the PDM. The model is plotted in yellow for each of the 224 cost function evaluations required for convergence. The final fit is shown in blue. Convergence is obtained when the ratio of the cost function at the maximum and minimum points in the simplex is less than 0.01.

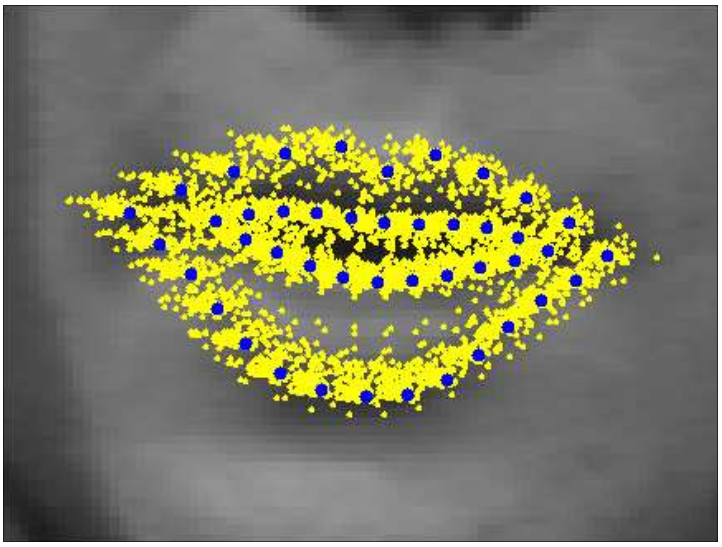


Figure 5.8: Simplex fit using E^2 as the cost function. The lip model is plotted in yellow for each of the 224 cost function evaluations, the final fit is in blue.

Only the shape parameters of the simplex minimised pose and shape vector are used as lipreading features. The pose is unlikely to be useful in the context of either of the databases considered and is simply part of the tracking process. Figure 5.9 plots the directions in each of the seven modes of variation of the PDM for the tracking results on the D-G-M sequence.

5.1.4 Per Talker Greylevel Modelling

The large variation between the appearance of the talkers in both AVletters, Figure 3.2 and Tulips databases, Figure 3.4, means the GLDM's trained over either entire database have a great many modes. The cost function, Equation (5.16), evaluated in such a high dimensional space is unlikely to have a clear minimum and the simplex search will be unable to find the correct pose and shape of the talkers lips.

One solution is to build separate GLDM's for each talker. These have to model only the variance during speech of a single talker and are much more compact. A set of GLDM's are built, one for each talker, k,

$$\mathbf{x}_p^k = \overline{\mathbf{x}}_p^k + \mathbf{P}_p^k \mathbf{b}_p^k \tag{5.18}$$

When fitting to an image the correct GLDM is chosen for the talker, which requires a priori knowledge of the identity of the talker. In practice it might be possible to automatically select a GLDM by evaluating several and finding which has the lowest cost function. For all experiments using these *local* GLDM's the identity of the talker was known. The whole database GLDM is referred to as the *global* GLDM.

5.1.5 Multi-resolution Simplex Fitting

The use of a coarse to fine multi-resolution image pyramid to improve the performance of ASM's was demonstrated by Cootes [55] for point-wise fitting. The image is Gaussian filtered and subsampled by two at each stage of the pyramid to form smaller versions of the original image, this is illustrated in Figure 5.10.

For each stage of the pyramid a new model must be built to learn the greylevel profiles about the model points. This is because length of the profiles is usually kept the same

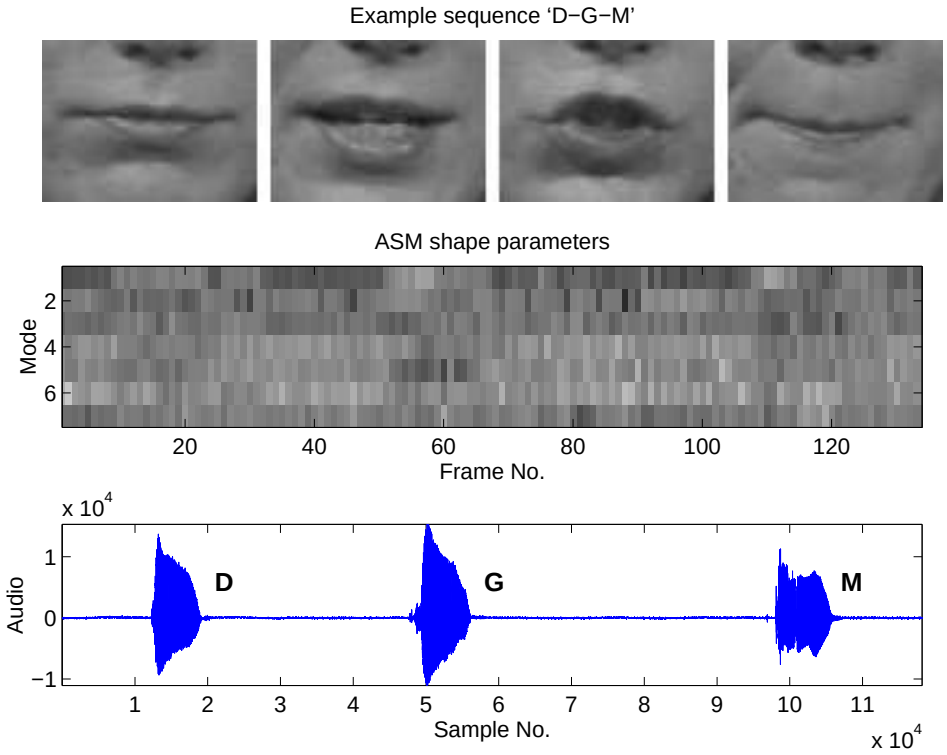


Figure 5.9: Simplex ASM tracked sequence. The shape parameters, the directions in each of the seven modes of the PDM, are plotted for the sequence of isolated letters, 'D-G-M'. Top row shows example images, bottom row is the aligned audio waveform.

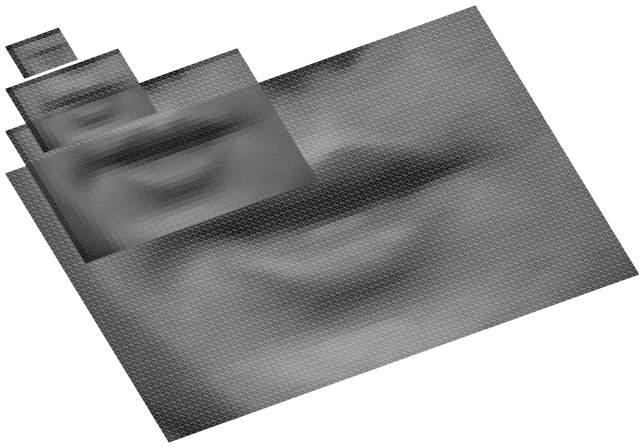


Figure 5.10: Multi-resolution Gaussian pyramid of a mouth image.

throughout but, as the image is half the size at each stage, the profiles extend twice as far each time and must model different image features.

For multi-resolution fitting each image is subsampled to form a pyramid. Because the mouth images of the AVletters and Tulips databases are small usually only two resolutions are used—the original and the half sized image. The search begins in the most coarse resolution image by scaling the mean model by the correct amount (by half for a two stage pyramid) and selecting the GLDM calculated at that resolution. Simplex minimisation is used to find the local minimum and when converged the final fit is scaled up to the next level. The next level GLDM is selected and the new simplex is run to convergence and so on. This extends the single resolution simplex fitting of Luettin [107] using the multi-resolution approach of Cootes [55]. To achieve this a set of GLDM's, one for each resolution, r, of the search is required,

$$\mathbf{x}_{p_r} = \overline{\mathbf{x}}_{p_r} + \mathbf{P}_{p_r}\mathbf{b}_{p_r} \tag{5.19}$$

Multi-resolution fitting allows much greater tolerance in the initial parameters. For example, at a coarse resolution, a displacement of five pixels is much more significant than at fine resolution. The GLDM's are more complex in the coarse resolution images because the same profile length covers more image features, there are 81 modes for AVletters and 61 for the Tulips coarse resolution GLDM's. The net effect is that a better fit can be obtained with less exact starting values over a wider range of images. The penalty is more minimisation iterations and the time taken to form the multi-resolution pyramid for each image. Figure 5.11 plots the multi-resolution fitting of the same image as Figure 5.8 with the same initial parameters. The coarse resolution cost function evaluations are plotted unscaled on the fine resolution image. There are fewer evaluations at the fine resolution as a rough fit had already been obtained at the coarse resolution, but in total there are 492 cost function evaluations versus 224 for the single scale search. The final fit is plotted in blue.

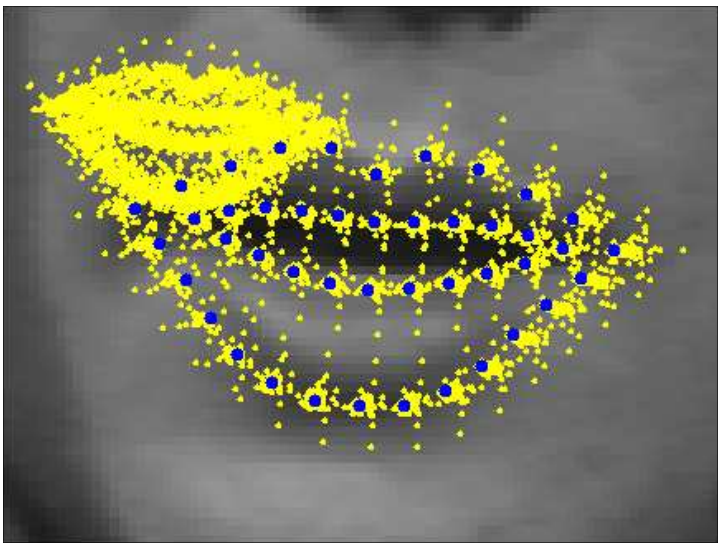


Figure 5.11: Multi-resolution simplex fit using E^2 as the cost function. The lip model is plotted in yellow for each of a total of 492 cost function evaluations at both coarse and fine resolutions on the same axes. The final fit is in blue.

The shape parameters obtained using multi-resolution fitting on the D-G-M sequence are shown in Figure 5.12. These are plotted on the same scale as the single resolution fit parameters shown in Figure 5.9. By finding a rough fit a the coarse resolution a better fit

can be obtained at the fine resolution. This can be seen as the slightly better defined parts of the sequence during speech. The largest shape parameter deviations from the mean for the single resolution search of this sequence were -1.96 and 1.66 which increased to -2.10 and 2.17 using the multi-resolution search.

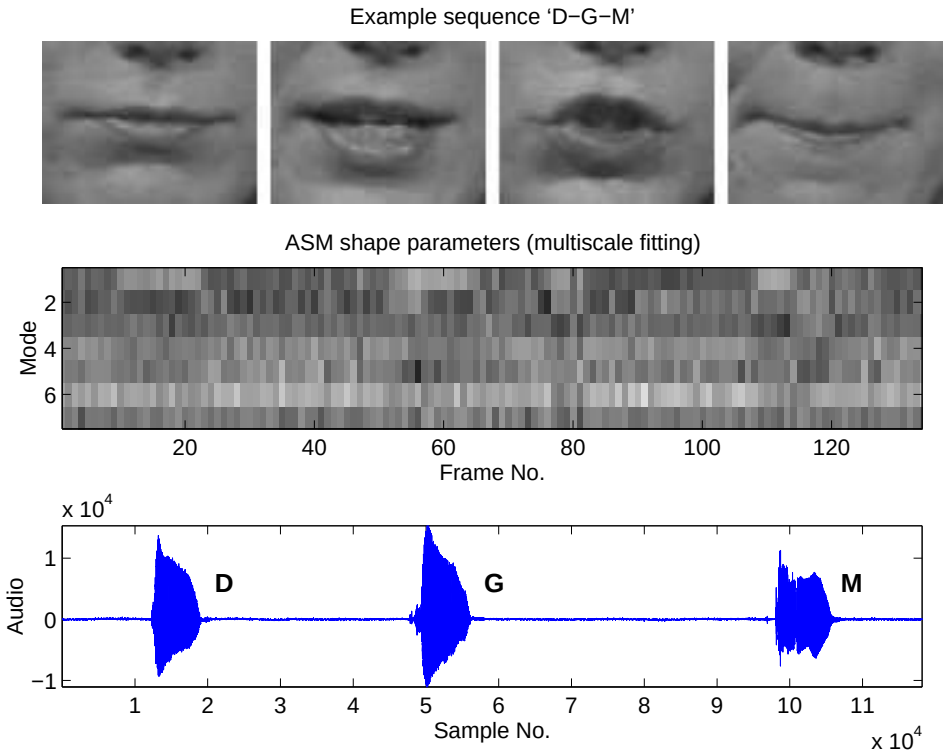


Figure 5.12: Multi-resolution simplex ASM tracked sequence. The shape parameters, the directions in each of the seven modes of the PDM, are plotted for the sequence of isolated letters, 'D-G-M'. Top row shows example images, bottom row is the aligned audio waveform.

Local, multi-resolution, GLDM's can further improve the quality of fit by modelling the greylevel profiles for each individual talker at each resolution. There are then a set of GLDM's for each talker, k, and resolution, r,

$$\mathbf{x}_{p_r}^k = \overline{\mathbf{x}}_{p_r}^k + \mathbf{P}_{p_r}^k \mathbf{b}_{p_r}^k \tag{5.20}$$

The following four Figures show some example well- and poorly-fitted sequences from both the AVletters and Tulips databases. All tracking results were obtained using a dual resolution search with a local GLDM. Figure 5.13 shows a good example 'A' from AVletters and demonstrates the ASM's ability to cope with talkers with moustaches. Figure 5.14 show a poorly fitted AVletters sequence for a different talker saying the letter 'M'. This shows the most common cause of failure—a mouth corner point has been lost. Without the corner point to clearly define the horizontal extent of the lip model it tends to collapse toward the other corner. Unless the simplex is set up to perturb over a large enough scale and translation the local minima will not include the correct position. By searching over a greater space, multi-resolution tracking reduces this problem. When interacting with a real-time implementation of the single resolution ASM tracker (Section 9.2) this failure is common and often requires manual reinitialisation of the model. Figure 5.15 shows a good example from the Tulips

database for the digit '3' and Figure 5.16 a bad example, also of the digit '3', for a different talker. In this example the inner contour remains attached to the strong edge defined by the upper teeth. This is such a strong feature that the local minimum of the cost function does not require the lower contour to follow the lower lip.

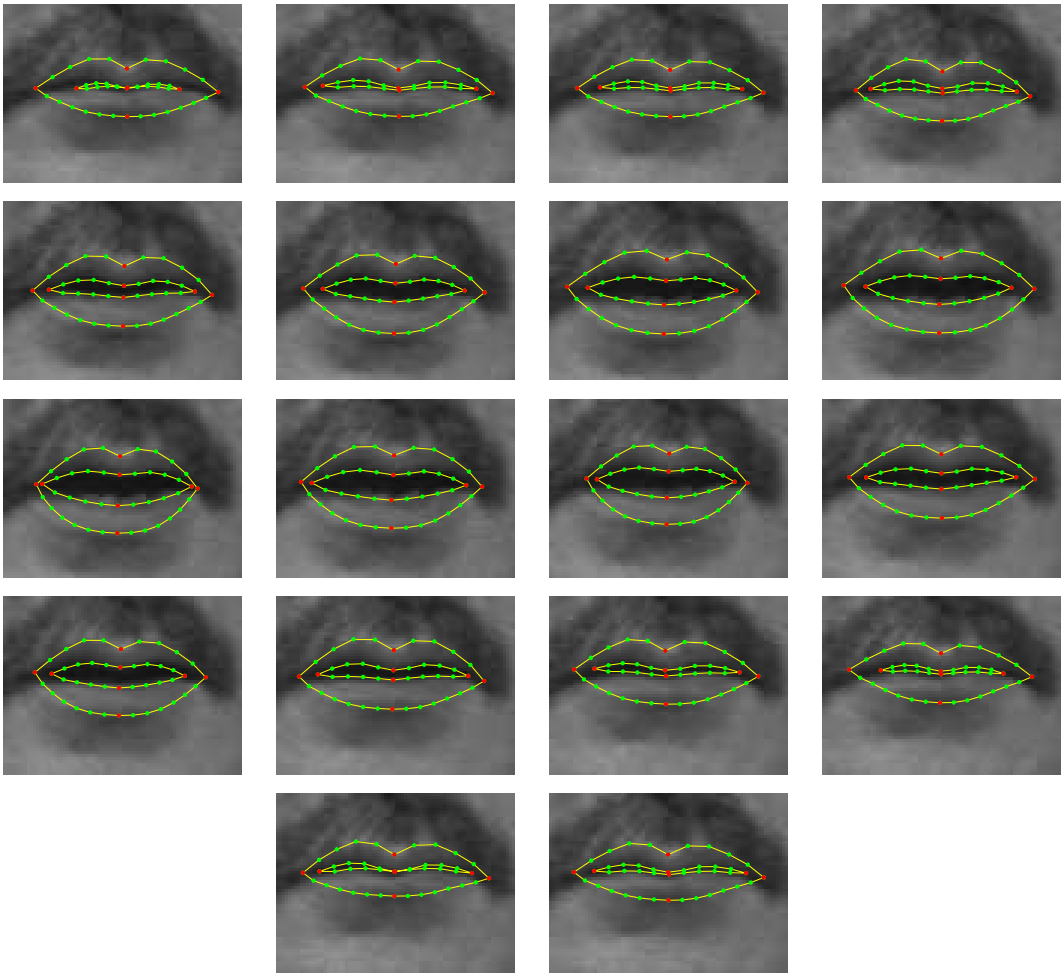


Figure 5.13: Well tracked example from the AVletters database for the letter 'A'. Moustaches are not a problem when using learned greylevel profile distribution models.

5.1.6 Per Talker Shape Modelling

The active shape model search described in the previous sections fits a PDM to an image by minimising a greylevel cost function over a combined shape and pose space $(t_x, t_y, s, \theta, b_1, b_2, \ldots, b_t)$. This search can be simplified by reducing the dimensionality of the space, i.e. reducing t, the number of modes of variation of the PDM. However, this would result in poorer fit as the shape model would represent less of the variance seen in the training set. The number of modes can be reduced without sacrificing variability if a new PDM is built for each talker. By removing the inter-talker variation the per-talker models are, in all cases but one, smaller than the global PDM. There are now a set of PDM's, one for each talker, k, in the database,

$$\mathbf{x}^k = \overline{\mathbf{x}}^k + \mathbf{P}_s^k \mathbf{b}_s^k \tag{5.21}$$

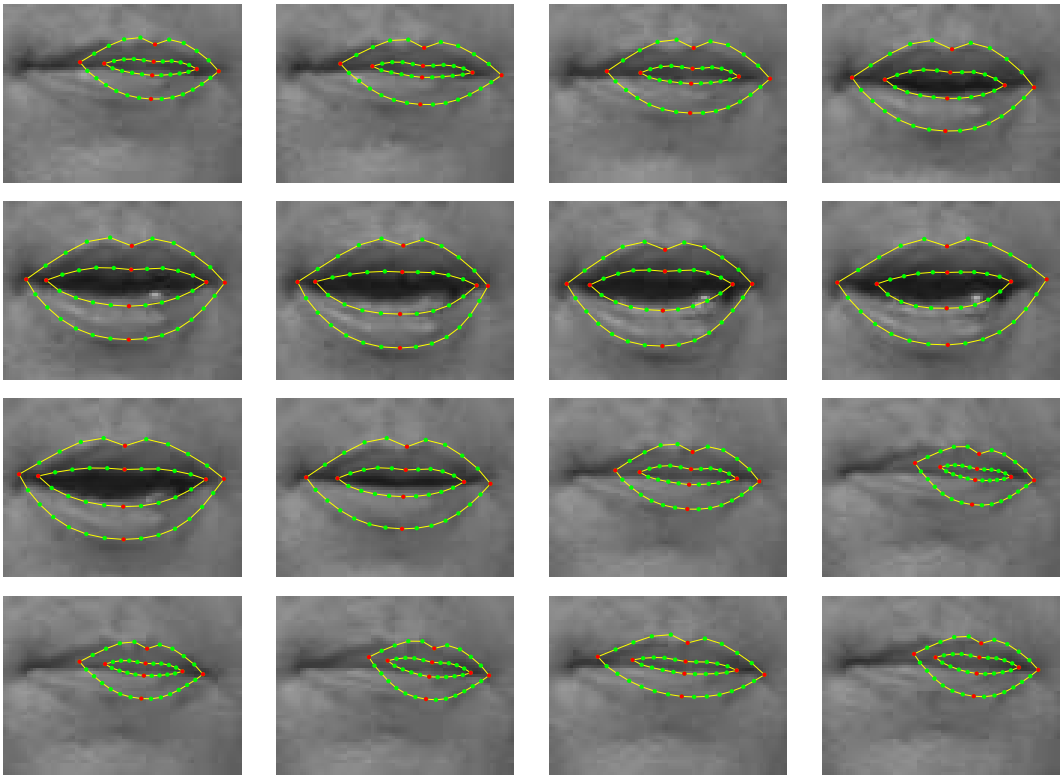


Figure 5.14: Poorly tracked example from the AVletters database for the letter 'M'. The most common tracking failure occurs when a mouth corner point is lost.

Figure 5.17 shows the PDM modes at ± 2 standard deviations about the mean for each talker of the AVletters database plotted on the same scale axes. There are clearly large scale and mean shape differences between talkers. In most cases the per-talker PDM has significantly fewer modes than the seven of the whole database. Only talkers two and seven have more than three modes. These are the two talkers with moustaches and this may be due in part to the difficulty that poses when hand placing the landmark points. Figure 5.18 shows the modes for each talker of the Tulips database and there is clearly less scale and mean shape variation between talkers than seen in the AVletters database. Talker nine has the maximum of four modes.

The fit parameters obtained by running an ASM with a *local* PDM cannot be related to the parameters obtained by fitting on any other talker. For multi-talker speech recognition (trained and tested using examples from all talkers) to avoid training a separate hidden Markov model for each talker (which would be difficult either of the small databases) the fit parameters must be mapped into a talker independent shape space. This is possible by transforming the fit parameters through the 88 point image coordinate space and into talker independent shape space using the talker independent, global PDM. First the translation, rotation and scaling pose differences between the mean shapes of the talker dependent and talker independent models must be removed. This aligns the mean 88 landmark points of both models as closely as possible, the remaining difference is described by the shape parameters in talker independent shape space, using Equation (5.8), the fit parameters required.

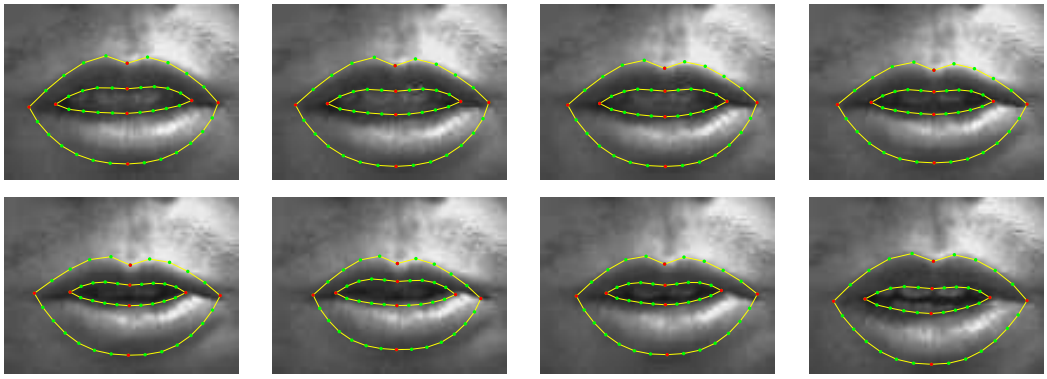


Figure 5.15: Well tracked example from the Tulips database for the digit '3'.

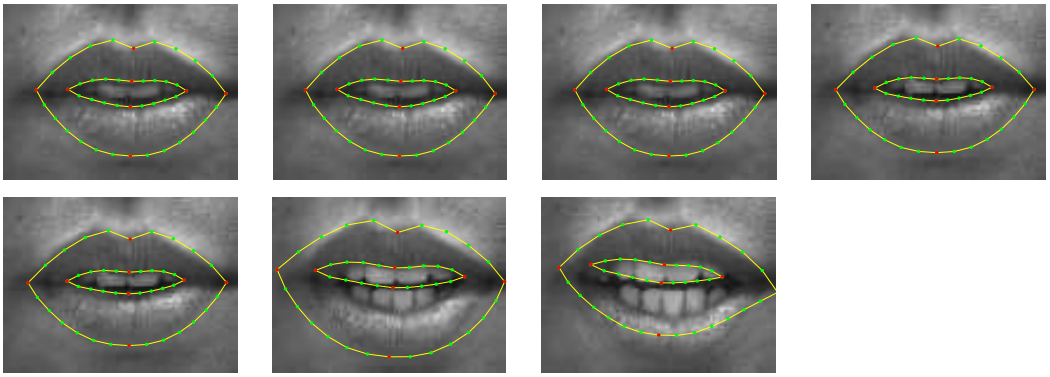


Figure 5.16: Poorly tracked example from the Tulips database for the digit '3'. The inner contour becomes attached to the strong edge define by the upper teeth.

5.2 Active Appearance Model Lip Tracking

This section is the result of collaborative work with Dr Tim Cootes at the Wolfson Image Analysis Unit, University of Manchester. Figures 5.19 and 5.20 are reproduced from [131].

An active appearance model (AAM) models both shape and greylevel appearance in a single statistical model and is due to Cootes [49], who also describes a fast iterative fitting technique. In the lipreading context it combines the greylevel analysis approaches of [23, 32, 34,63,84,104,142,178,195] with the shape analysis of [9,48,58,91,93,161,168,176,190,191,197].

There are some examples of using both greylevels and shape. Luettin [107, 109, 114] used the GLDM fit parameters as well as the PDM shape parameters from an ASM fit, and Bregler [23–28] used nonlinearly shape-constrained snakes to find the lips for an eigen analysis. However, neither combine greylevel and shape in a *single* statistically learned model. An active appearance model is an extension of both of these techniques, it unifies eigen analysis of the greylevels and ASM lip tracking.

The active appearance model is trained from the same set of landmark point labelled images of the AVletters databases that were used for the PDM in Section 5.1.1. The shape part of an AAM is exactly that PDM,

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s \tag{5.22}$$

where $\overline{\mathbf{x}}$ is the mean shape, $\mathbf{P}_s$ is the matrix of t_s orthogonal modes of variation and $\mathbf{b}_s$ the

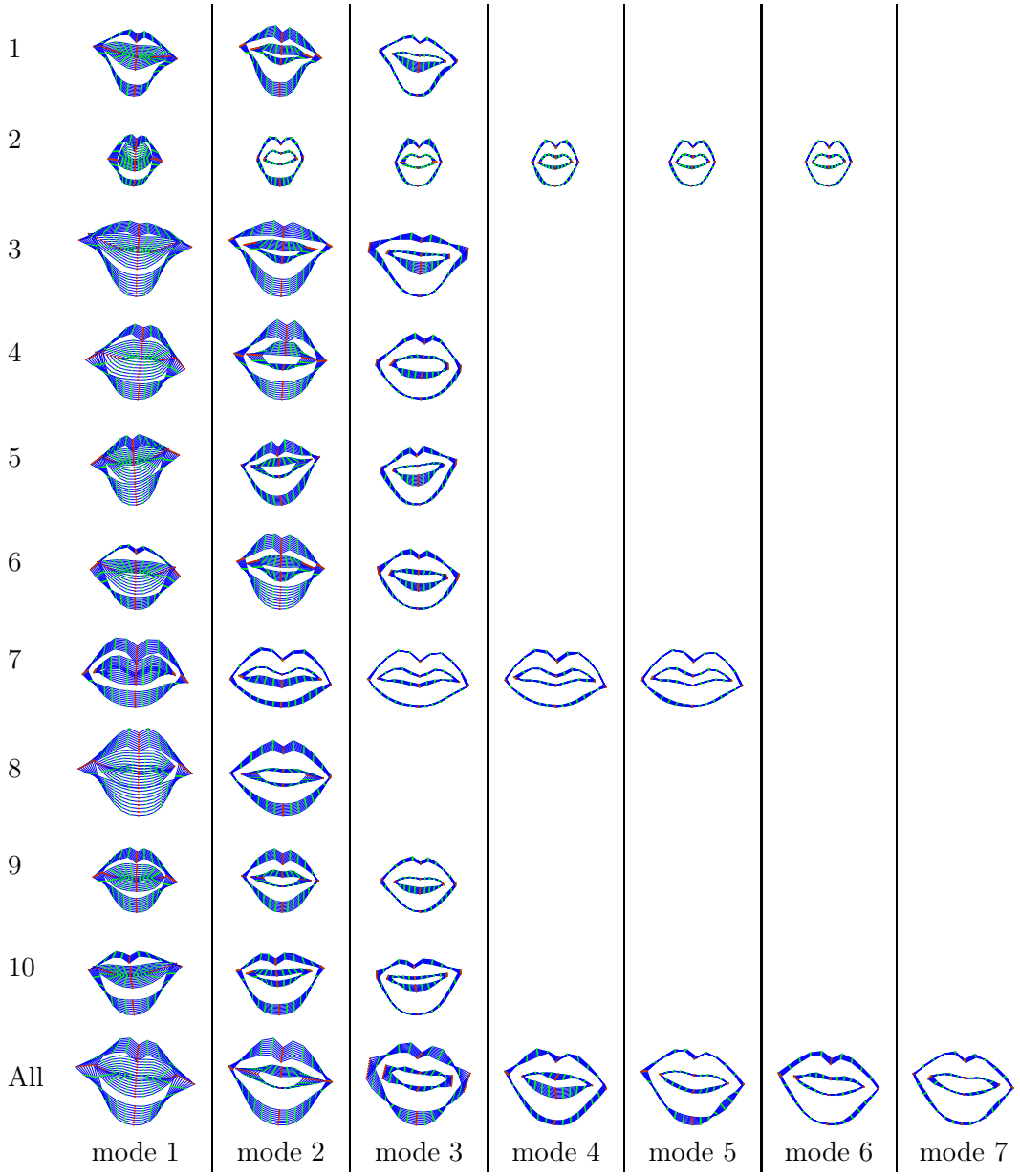


Figure 5.17: Per talker PDM's for the AVletters database. Talkers two and seven have moustaches. The extra modes for them may be due to mislabelling the landmark points which are much harder to place when the lip contours are obscured.

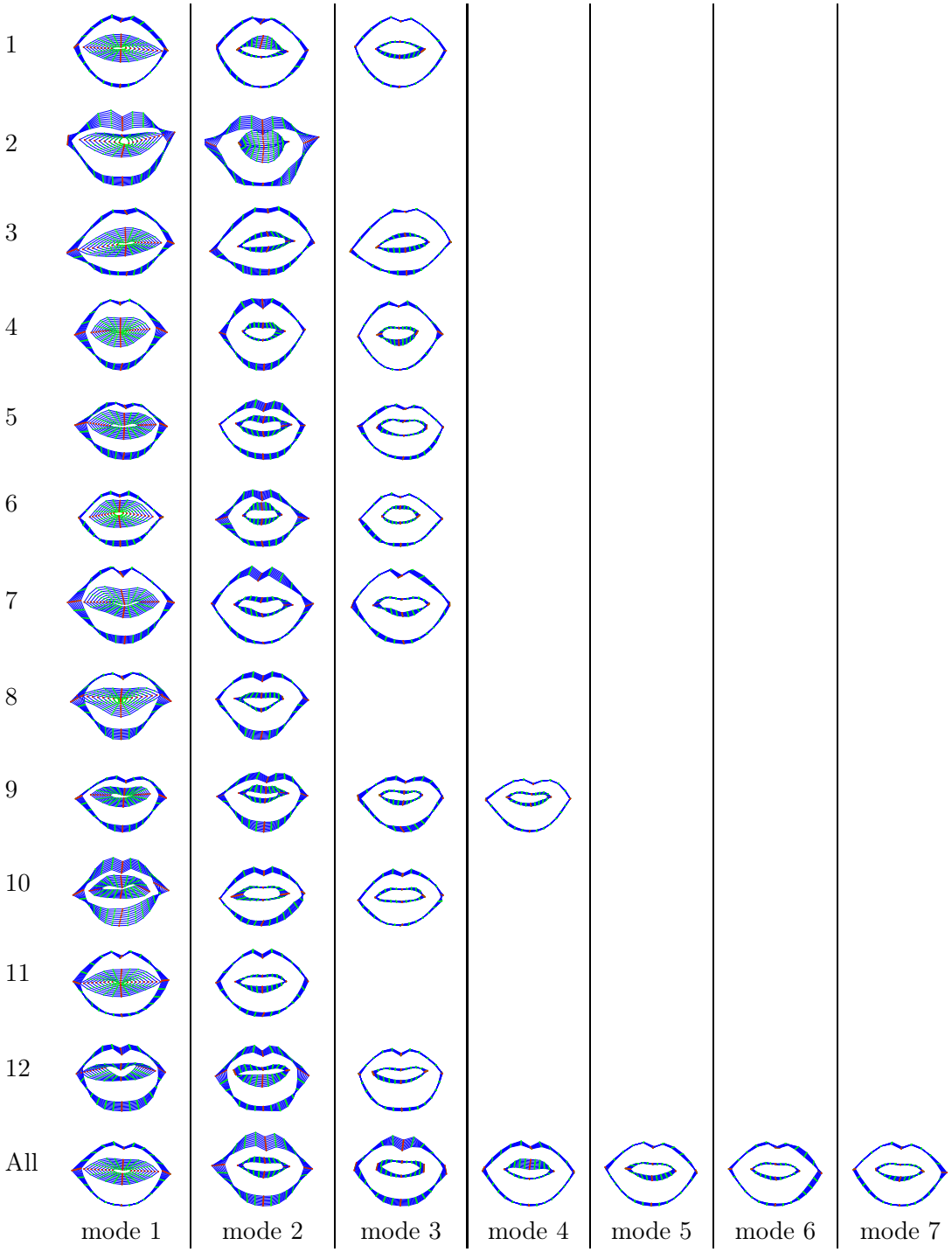


Figure 5.18: Per talker PDM's for the Tulips database. There are much less scale and mean shape differences than are seen in the AVletters database.

vector of t_s shape parameters.

A greylevel appearance model is built by warping each training image so the landmark points lie on the mean shape, $\overline{\mathbf{x}}$, normalising each image for shape. The greylevel values, $\mathbf{g}_{raw}$ are sampled within the landmark points of this shape normalised image. These are normalised over the training set for lighting variation using an iterative approach to find the best scaling, α and offset, β,

$$\mathbf{g} = (\mathbf{g}_{raw} - \beta\mathbf{1})/\alpha \tag{5.23}$$

where α and β are chosen to best match the normalised mean greylevel appearance, $\overline{\mathbf{g}}$. The mean appearance is scaled and offset for zero mean and unity variance so the values of α and β are calculated using,

$$\alpha = \mathbf{g}_{raw}\overline{\mathbf{g}} \tag{5.24}$$

$$\beta = (\mathbf{g}_{raw}\mathbf{1})/n \tag{5.25}$$

where n is the number of elements in the greylevel appearance vector $\mathbf{g}$. As when aligning the shape models of the PDM a stable normalised appearance model is obtained by aligning to the first model, re-estimating the mean, transforming and re-iterating.

The greylevel appearance model is calculated using PCA on the normalised greylevel data to identify the major modes of variation about the mean,

$$\mathbf{g} = \overline{\mathbf{g}} + \mathbf{P}_g\mathbf{b}_g \tag{5.26}$$

where $\mathbf{P}_g$ is the set of t_g orthogonal modes of variation of the greylevel appearance and $\mathbf{b}_g$ a vector of t_g weights.

This extends the greylevel profile modelling described in Section 5.1.2 to model the entire greylevel appearance within the landmark points rather than just profiles taken at the normal of each point. It is a principal component analysis of the shape and greylevel normalised pixel intensities within the shape defined by the landmark points of the hand labelled training images.

The AAM is built by applying a further PCA to identify the correlation between the shape parameters $\mathbf{b}_s$ and greylevel appearance parameters $\mathbf{b}_g$. A concatenated shape and greylevel appearance vector is formed for each example,

$$\mathbf{b} = \begin{pmatrix} \mathbf{W}_s\mathbf{b}_s \\ \mathbf{b}_g \end{pmatrix} = \begin{pmatrix} \mathbf{W}_s\mathbf{P}_s^T(\mathbf{x} - \overline{\mathbf{x}}) \\ \mathbf{P}_g^T(\mathbf{g} - \overline{\mathbf{g}}) \end{pmatrix} \tag{5.27}$$

where $\mathbf{W}$ is a diagonal weight matrix for each shape parameter chosen to normalise the difference in units between the shape and greylevel appearance parameters and remove PCA scaling problems [49].

This gives a combined shape and greylevel appearance model,

$$\mathbf{b} = \mathbf{Q}\mathbf{c} \tag{5.28}$$

where $\mathbf{Q}$ is the matrix of t_a eigenvectors and $\mathbf{c}$ the vector of t_a *appearance* parameters. Since the shape and greylevel appearance parameters have zero mean weights, $\mathbf{c}$ is also zero mean.

Since the model is linear, shape and appearance can be expressed independently in terms of $\mathbf{c}$,

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{P}_s\mathbf{W}_s\mathbf{Q}_s\mathbf{c} \tag{5.29}$$

$$\mathbf{g} = \overline{\mathbf{g}} + \mathbf{P}_g\mathbf{Q}_g\mathbf{c} \tag{5.30}$$

where,

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_s \\ \mathbf{Q}_g \end{pmatrix} \tag{5.31}$$

Figure 5.19 shows the first three modes at ± 2 standard deviations about the mean of the combined appearance model trained on the AVletters database. The full model has 37 modes of variation.

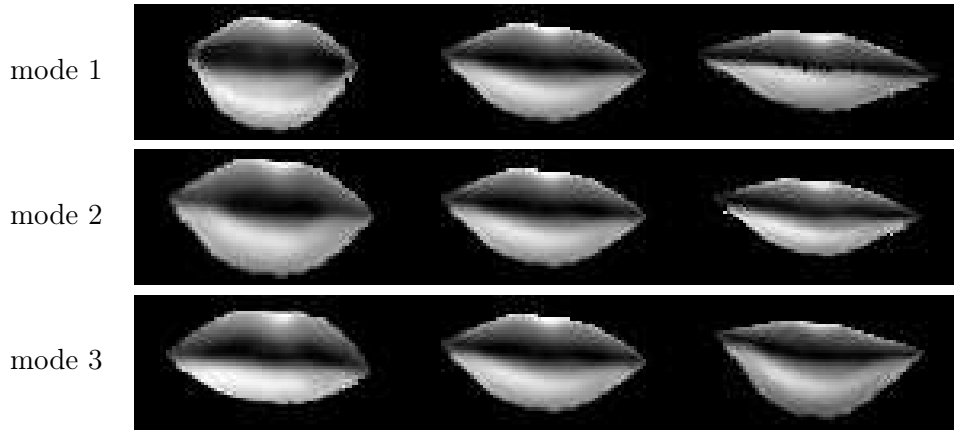


Figure 5.19: Combined shape and greylevel appearance model. First three modes of variation of at ± 2 standard deviations about the mean.

To fit an appearance model to an image, the Active Appearance Model algorithm [49] is used to find the best pose and appearance parameters. The fitting process minimises the difference between the example image and that synthesised by the current model parameters. If the normalised greylevel appearance parameters of the image are $\mathbf{g}_i$ and the model synthesised values, from Equation (5.30), $\mathbf{g}_m$, the difference is,

$$\delta\mathbf{g} = \mathbf{g}_i - \mathbf{g}_m \tag{5.32}$$

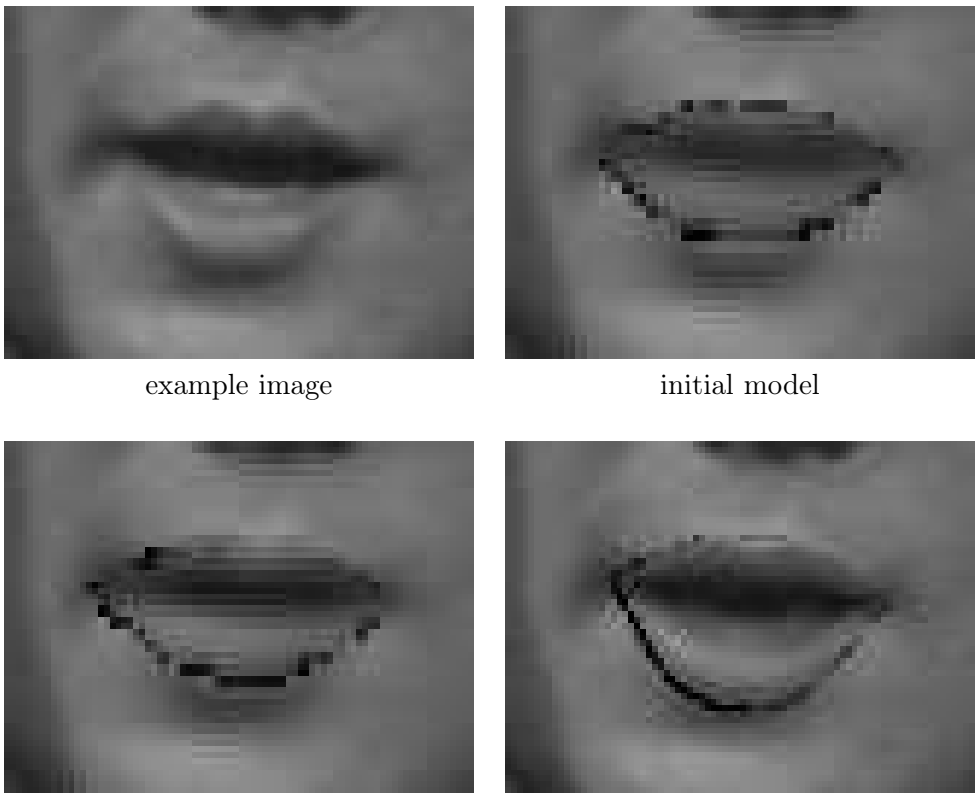
The AAM algorithm simplifies this high dimensional optimisation problem by learning in advance how to update the model parameters given the current difference image. Over a limited range of displacements, a linear model can accurately predict the correct model update from the difference image. The update model, $\mathbf{R}$, is calculated from the statistics obtained by systematically displacing the the model pose and appearance parameters in the training images. To iteratively fit an AAM to an image the model parameters are updated at each iteration using the update model,

$$\mathbf{c} \mapsto \mathbf{c} - \mathbf{R}\delta\mathbf{g} \tag{5.33}$$

until no significant change occurs. A similar procedure is used for the pose parameters. The accurate prediction range of the linear update model can be increased by using a multi-resolution fitting approach, similar to that discussed in Section 5.1.5

Example iterations from a fit are shown in Figure 5.20. The model is initialised in the centre of images with the mean appearance parameters at the coarse resolution. After 15 iterations the model has converged at the fine scale; this took less than one second on a 166MHz Pentium. The converged appearance parameters form 37 dimensional lipreading feature vectors.

Results using AAM's to track and identify faces in image sequences using temporal



example image | initial model

after 4 iterations | after 15 iterations

Figure 5.20: Example of AAM search. The model is initialised in the centre of the image at coarse scale. Convergence using the AAM algorithm took 15 iterations.

Kalman filtering are given in Edwards [65]. The lipreading parameters obtained by Cootes [131] on the AVletters database used no Kalman filtering. The model was initialised at the centre of each image frame in turn with the mean appearance.

Chapter 6

Hidden Markov Model Recognition

A significant point in speech recognition research was the introduction of stochastic modelling of the speech signal. Early speech recognition systems used a pattern matching approach where unknown parameterised speech utterances were classified by comparison with a set of templates trained for each word in the vocabulary. Differences in time-scale between templates were accommodated by a linear stretching or compressing of the time axis. A major problem with this approach is that variation in speaking rate, even by the same talker, can cause classification error even if the speech parameters themselves are robust. This temporal variation between a given utterance and the template can be effectively reduced using dynamic programming, or dynamic time warping (DTW), to find the 'best fit' alignment. Another problem is in the choice of the reference templates which must respect the natural variation in speech within and between talkers yet also be descriptive enough to discriminate effectively between the different words of the vocabulary. For any realistic task, the training data is unlikely to cluster into non-overlapping areas from which clearly distinct templates can be drawn. One solution to this is to use multiple templates for each speech unit.

A more successful and unified approach is the modelling of both the speech parameters and the temporal sequence in a statistical framework such as a hidden Markov model (HMM). A HMM models a speech event as a stochastic finite state machine that may only be observed through another stochastic process. This allows both temporal and speech variability in a single parametric model.

Alternative approaches for speech recognition such as linguistic knowledge based or artificial intelligence methods are not considered here as they have not shown performance to match the statistical methods and are not easily applied to other tasks, such as visual or audio-visual speech recognition.

This chapter is a brief overview of the theory of HMM's which is covered more fully elsewhere [57, 103, 140, 159, 160, 194]. There are two steps in building a hidden Markov model classifier; first the model parameters are estimated from training utterances (training) and then these parameters are used to find the most likely model to have produced an example utterance (recognition). The second, recognition, stage is presented first to introduce the concepts and terminology.

All of the recognition experiments in this thesis were performed using hidden Markov models to model the feature vector sequences of both acoustic and visual speech.

6.1 Hidden Markov Models

The first assumption when using hidden Markov models is that when windowed over a short time period the speech signal is statistically stationary, so that speech can be modelled as a

sequence of discrete feature vectors, or observations, derived from each (equally spaced) window. This assumes the window width can both be short enough that the speech articulators do not significantly change in that time yet long enough that the features extract relevant speech information. As speech is a continuous process this is only an approximation. Speech is modelled as a sequence of discrete observations, O_t, made at each windowing time point, or frame, t.

$$\mathbf{O} = O_1, O_2, \ldots, O_T \tag{6.1}$$

The second assumption is that the observation sequence can be modelled by a doubly stochastic finite state machine. At any time step the model is in a state, s, and will output an observation, O. At the next time step the model may change state and will output an new observation. This process is governed by the model topology chosen and the probabilities separately associated with changing state and outputting a given observation. The resulting observation sequence cannot be unambiguously mapped to a corresponding state sequence, hence the term 'hidden'. The state at time t depends only on the state at time $t-1$ and the associated probability of changing state, the Markov property. An example HMM is shown in Figure 6.1.

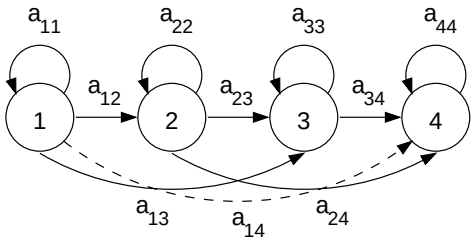


Figure 6.1: Example left to right hidden Markov model with skip transitions.

This example is a left to right model, i.e. all transitions are either to the same state or to the right. The transition probabilities, a_{ij}, are usually written for a model with N states as an $N \times N$ matrix $\mathbf{A}$.

$$\mathbf{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1N} \\ a_{21} & a_{22} & \ldots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \ldots & a_{NN} \end{bmatrix} \tag{6.2}$$

The value of a_{ij} is the probability of making the transition to state j from state i at the next time step. In general HMM state transitions can occur from any state to any other but left to right topologies are always used in speech recognition to model the temporal order of speech sounds (which cannot go back in time). For a left to right model $\mathbf{A}$ is upper triangular. There must always be a transition from each state so each row must sum to a probability of one,

$$\sum_j a_{ij} = 1, \qquad \forall i \tag{6.3}$$

For a discrete HMM, also associated with each state j is a probability, b_{jk}, of outputting one of the k symbols of a discrete alphabet of M possible symbols, $v_1, v_2, \ldots, v_M$. This is

represented by the $N \times M$ matrix $\mathbf{B}$.

$$\mathbf{B} = [b_{jk}] = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1M} \\ b_{21} & b_{22} & \dots & b_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ b_{N1} & b_{N2} & \dots & b_{NM} \end{bmatrix} \tag{6.4}$$

At each time step there must be an output symbol so each row of $\mathbf{B}$ must also sum to a probability of one,

$$\sum_k b_{jk} = 1, \qquad \forall j \tag{6.5}$$

For a continuous HMM the discrete probabilities b_{jk} are replaced with a continuous probability density function: this will be covered in Section 6.4.

An initial $1 \times N$ vector, $\pi = [\pi_1, \pi_2, \dots, \pi_N]$, defines the initial state probabilities for time $t = 0$; after $t = 0$, these are defined by the state transition matrix, $\mathbf{A}$. A common value for π is $\pi_1 = 1$ and all other values are zero. An N-state discrete HMM with an alphabet of M-symbols is then fully defined by the parameter set, $\mathbf{M} = [\pi, \mathbf{A}, \mathbf{B}]$.

6.2 Recognition

For isolated word recognition the task is to compute the most likely class, k, given an unknown observation sequence, $\mathbf{O}$, and a vocabulary of words, $\mathbf{W} = w_1, w_2, \dots, w_W$.

$$k = \underset{i=1,2,\dots,W}{\arg\max} \{P(w_i \mid \mathbf{O})\} \tag{6.6}$$

Using Bayes rule this can be rewritten as:

$$P(w_i \mid \mathbf{O}) = \frac{P(\mathbf{O} \mid w_i)P(w_i)}{P(\mathbf{O})} \tag{6.7}$$

As $P(\mathbf{O}) = \sum_{i=1}^{W} P(\mathbf{O} \mid w_i)P(w_i)$ this term is a scaling factor that is constant over all words and may be ignored. The prior probabilities $P(w_i)$ are defined by a language model for a given application. If all words are considered to be equally likely this term will also be constant. Hence Equation (6.6) becomes,

$$k = \underset{i=1,2,\dots,W}{\arg\max} \{P(\mathbf{O} \mid w_i)\} \tag{6.8}$$

Assuming that a hidden Markov model can accurately model the observation sequences for each word in the vocabulary and there are W models, one for each word, $\mathbf{M}_1, \mathbf{M}_2, \dots, \mathbf{M}_W$, then it is only necessary to calculate,

$$k = \underset{i=1,2,\dots,W}{\arg\max} \{P(\mathbf{O} \mid \mathbf{M}_i)\} \tag{6.9}$$

Equation (6.9) cannot be computed by simply considering all possible state sequences that could have produced the observation sequence. In general there could be N^T sequences for an N-state model given T-observation frames which quickly becomes unrealistically large. To reduce the computation a recursive algorithm such as the *Baum-Welch* algorithm or *Viterbi* algorithm can be used.

6.2.1 Baum-Welch Recognition

The Baum-Welch algorithm calculates the *forward probabilities*, $\alpha_t(j)$, which are the joint probabilities of emitting the partial observation sequence, $O_1, O_2, \ldots, O_t$, and being in state, s_j at time, t.

$$\alpha_t(j) = P(O_1, O_2, \ldots, O_t,\ s_j@t \mid \mathbf{M}) \tag{6.10}$$

The forward probability at the next time step depends only on the previous forward probability and the associated probabilities of changing state and emitting the next seen observation. This allows the forward probabilities to be calculated recursively,

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(O_{t+1}) \qquad t = 1, 2, \ldots, T-1 \tag{6.11}$$

given the initial values,

$$\alpha_1(j) = \pi(j) b_j(O_1) \tag{6.12}$$

The probability of emitting the sequence of T frames, $\mathbf{O}$, and ending in state j is $\alpha_T(j)$ so the total Baum-Welch probability for a given model is found by summing over all states,

$$P(\mathbf{O} \mid \mathbf{M}) = P^{BW} = \sum_{j=1}^{N} \alpha_T(j) \tag{6.13}$$

6.2.2 Viterbi Recognition

If the most likely state sequence is required, as well as the probability that the model produced the observation sequence, then the summation of Equation (6.11) can be replaced with the maximum operator. The result is a dynamic programming, or best path search, otherwise known as the *Viterbi* algorithm[1].

The highest probability at each time step can be calculated recursively similarly to the forward probabilities in Equation (6.11),

$$\phi_{t+1}(j) = \max_{i=1,2,\ldots,N} [\phi_t(i) a_{ij}] b_j(O_{t+1}) \qquad t = 1, 2, \ldots, T-1 \tag{6.14}$$

with initial values,

$$\phi_1(j) = \pi(j) b_j(O_1) \tag{6.15}$$

Similarly with Equation (6.13) the probability of emitting the T-frames sequence $\mathbf{O}$ and ending in state j is $\phi_T(j)$ and the total Viterbi probability for a given model is found by maximising over all states,

$$P(\mathbf{O} \mid \mathbf{M}) = P^{V} = \max_{j=1,2,\ldots,N} [\phi_T(j)] \tag{6.16}$$

To recover the most likely state sequence, the state at time $t-1$ that maximised Equation (6.14) is recorded at each time t and state j,

$$\psi_{t+1}(j) = \arg\max_{i=1,2,\ldots,N} [\phi_t(i) a_{ij}] \qquad t = 1, 2, \ldots, T-1 \tag{6.17}$$

The most likely state sequence is recovered by back tracking from the most likely final state

[1]The 'optimal' state sequence can be obtained for the Baum-Welch algorithm [159] but does not consider the probability of occurrence of the state sequence and may not be valid in terms of the model topology.

at time T that maximised Equation (6.16),

$$\psi_T(j) = \underset{i=1,2,\ldots,N}{\arg\max}[\phi_T(i)] \tag{6.18}$$

which gives the most likely state, k, for time $T-1$. From $\psi_{T-1}(k)$ the most likely state at $T-2$ is obtained and so on until the most likely state at $t=1$ is recovered and the most likely path is obtained.

6.3 Training

The training problem is: given a model, $\mathbf{M} = [\pi, \mathbf{A}, \mathbf{B}]$, what are the optimal parameters to represent the training data? This cannot be solved directly [159] but $P(\mathbf{O} \mid \mathbf{M})$ can be maximised using iterative methods. The method most often used is the Baum-Welch (forward-backward) algorithm.

Similar to the forward probabilities, $\alpha_t(j)$, defined in Equation (6.10) the *backward* probabilities, $\beta_t(i)$, are the joint probabilities of emitting the partial observation sequence, $O_{t+1}, O_{t+2}, \ldots, O_T$, starting from state, s_i at time, t,

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots, O_T \mid s_i@t,\ \mathbf{M}) \tag{6.19}$$

Like the forward probabilities the backward probabilities can also be calculated recursively, but backwards from time T, dependent only the previous β's, the state transition matrix and associated output probabilities,

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \qquad t = T-1, T-2, \ldots, 1 \tag{6.20}$$

with the initialisation $\beta_T(i) = 1\ \forall i$.

The forward probabilities are the joint probabilities of emitting the partial observation up to time t and being in state j. The backward probabilities are the conditional probabilities of being in state j at t and continuing to emit the rest of the observation sequence, Figure 6.2. The forward and backward probabilities are defined this way that so the probability of being

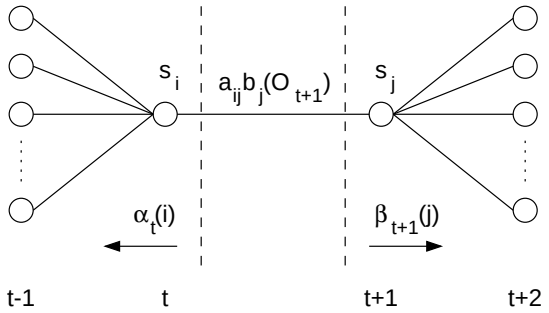


Figure 6.2: Illustration of the forward and backward probabilities. The α's are the probability of the partial sequence to t, the β's the partial sequence from $t+1$. Due to [159]

in state j at time t can be calculated for a given observation sequence,

$$\alpha_t(j)\beta_t(j) = P(\mathbf{O},\ s_j@t \mid \mathbf{M}) \tag{6.21}$$

These probabilities are needed for the iterative Baum-Welch re-estimation formula. Given an estimate of the model parameters, $\mathbf{M}$ and a training observation sequence, $\mathbf{O}$, these formulae calculate new model parameters, $\mathbf{M}'$, that have been shown to converge to the local maxima of $P(\mathbf{O} \mid \mathbf{M})$ [12].

$$
\begin{aligned}
a'_{ij} &= \frac{P(\text{transition from } s_i \text{ to } s_j \mid \mathbf{O}, \mathbf{M})}{P(\text{transition from } s_i \text{ to any state} \mid \mathbf{O}, \mathbf{M})} && (6.22)\\
b'_{jk} &= \frac{P(\text{emitting symbol } k \text{ from } s_j \mid \mathbf{O}, \mathbf{M})}{P(\text{emitting any symbol from } s_j \mid \mathbf{O}, \mathbf{M})} && (6.23)\\
\pi'_i &= P(\text{observation sequence begins in } s_i \mid \mathbf{O}, \mathbf{M}) && (6.24)\\
\mathbf{M}' &= [\pi', \mathbf{A}', \mathbf{B}'] && (6.25)
\end{aligned}
$$

To calculate these Rabiner [159] introduces two new variables. The probability of being in state i at time t given the observation sequence $\mathbf{O}$ and model $\mathbf{M}$,

$$
\begin{aligned}
\gamma_t(i) &= P(s_i@t \mid \mathbf{O},\ \mathbf{M})\\
&= \frac{P(\mathbf{O},\ s_i@t,\ \mathbf{M})}{P(\mathbf{O},\ \mathbf{M})}\\
&= \frac{P(\mathbf{O},\ s_i@t \mid \mathbf{M})P(\mathbf{M})}{P(\mathbf{O} \mid \mathbf{M})P(\mathbf{M})}\\
&= \frac{P(\mathbf{O},\ s_i@t \mid \mathbf{M})}{P(\mathbf{O} \mid \mathbf{M})} && (6.26)
\end{aligned}
$$

using Equation (6.21) this becomes,

$$
\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(\mathbf{O} \mid \mathbf{M})} \tag{6.27}
$$

To calculate the numerator of Equation (6.22) the probability of being in state i at time t and state j at $t+1$ must also be determined,

$$
\begin{aligned}
\xi_t(i,j) &= P(s_i@t,\ s_j@t+1 \mid \mathbf{O}, \mathbf{M})\\
&= \frac{P(s_i@t,\ s_j@t+1,\ \mathbf{O}, \mathbf{M})}{P(\mathbf{O}, \mathbf{M})}\\
&= \frac{P(s_i@t,\ s_j@t+1,\ \mathbf{O} \mid \mathbf{M})P(\mathbf{M})}{P(\mathbf{O} \mid \mathbf{M})P(\mathbf{M})}\\
&= \frac{P(s_i@t,\ s_j@t+1,\ \mathbf{O} \mid \mathbf{M})}{P(\mathbf{O} \mid \mathbf{M})} && (6.28)
\end{aligned}
$$

using the forward probabilities to state i at time t, Equation (6.10), and the backward probabilities from state j at $t+1$, Equation (6.19), and accounting for the transition from i to j (see Figure 6.2) this becomes,

$$
\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(\mathbf{O} \mid \mathbf{M})} \tag{6.29}
$$

The probability of transition from state i to j is then the sum of $\xi_t(i,j)$ over all time frames and the probability of being in state i the sum of $\gamma_t(i)$ over all time so Equation (6.22)

becomes,

$$\begin{aligned} a'_{ij} &= \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\ &= \frac{\sum_{t=1}^{T-1} \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(\mathbf{O} \mid \mathbf{M})}}{\sum_{t=1}^{T-1} \frac{\alpha_t(i)\beta_t(i)}{P(\mathbf{O} \mid \mathbf{M})}} \end{aligned} \tag{6.30}$$

The probability of emitting symbol k from state j is the sum of the probability of being in state j over all times that symbol k was seen. The probability of emitting any symbol from j is the probability of being in state j so Equation (6.23) can be written,

$$\begin{aligned} b'_{jk} &= \frac{\sum_{t \ni O_t = v_k} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \\ &= \frac{\sum_{t \ni O_t = v_k} \frac{\alpha_t(j)\beta_t(j)}{P(\mathbf{O} \mid \mathbf{M})}}{\sum_{t=1}^{T} \frac{\alpha_t(j)\beta_t(j)}{P(\mathbf{O} \mid \mathbf{M})}} \end{aligned} \tag{6.31}$$

Finally, the probability of starting in state i is the probability of being in state i at time $t = 1$, giving the initial state probability vector update,

$$\begin{aligned} \pi'_i &= \gamma_1(i) \\ &= \frac{\alpha_1(i)\beta_1(i)}{P(\mathbf{O} \mid \mathbf{M})} \end{aligned} \tag{6.32}$$

The Baum-Welch re-estimation formulae iteratively estimate a new model $\mathbf{M}'$ where $P(\mathbf{O} \mid \mathbf{M}') \geq P(\mathbf{O} \mid \mathbf{M})$. This process is continued until it converges and there is no change in the model likelihood or a maximum number of iterations is reached.

6.4 Continuous Density Functions

A discrete HMM can only emit one of the finite alphabet of M symbols. Observation vectors must therefore be quantised to one of these symbols so introducing extra noise and error. If the alphabet is made very large then there are a large number of probabilities to estimate during training. An alternative is to represent the continuous distribution of observation vectors using a parametric model such as a multivariate Gaussian. The $\mathbf{B}$ matrix is then replaced by these parameters.

It has been shown that an arbitrary probability density function (pdf) can be approximated using a weighted sum (a mixture) of multivariate Gaussian pdf's. The emission prob-

ability can be modelled approximately, depending on the number of mixture components, using,

$$b_j(O_t) = \sum_{m=1}^{X} c_{jm}\mathcal{N}(O_t, \mu_{jm}, \Sigma_{jm}) \qquad j = 1, 2, \ldots, N \tag{6.33}$$

where X is the number of mixture components, c_{jm} is the weight of component m in state j and $\mathcal{N}(O_t, \mu_{jm}, \Sigma_{jm})$ is the probability of the vector O_t from the multivariate Gaussian distribution with the dimensionality of O_t, mean vector μ_{jm} and covariance matrix Σ_{jm},

$$\mathcal{N}(O, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}}\, e^{-\frac{1}{2}(O-\mu)'\,\Sigma^{-1}(O-\mu)} \tag{6.34}$$

6.4.1 Continuous Baum-Welch Training

The recognition process is unaffected by the change to continuous distributions but training must now estimate the parameters of the pdf rather than discrete probabilities.

For a single state, single component HMM, the maximum likelihood estimates for μ_j and Σ_j would be the averages,

$$\mu'_j = \frac{1}{T}\sum_{t=1}^{T} O_t \tag{6.35}$$

$$\Sigma'_j = \frac{1}{T}\sum_{t=1}^{T} (O_t - \mu_j)(O_t - \mu_j)' \tag{6.36}$$

To extend this to multiple states where observations cannot be assigned to any given state, the average is replaced by the probability of being in state j at time t, $\gamma_t(j)$,

$$\mu'_j = \frac{\sum_{t=1}^{T} \gamma_t(j) O_t}{\sum_{t=1}^{T} \gamma_t(j)} \tag{6.37}$$

$$\Sigma'_j = \frac{\sum_{t=1}^{T} \gamma_t(j)(O_t - \mu_j)(O_t - \mu_j)'}{\sum_{t=1}^{T} \gamma_t(j)} \tag{6.38}$$

For multiple components $\gamma_t(j)$, Equation (6.27), is extended to $\gamma_t(j,k)$ the probability of being in state j at t with the k'th mixture component accounting for $\mathbf{O}$,

$$\gamma_t(j,k) = \frac{\alpha_t(j)\beta_t(j)}{P(\mathbf{O} \mid \mathbf{M})} \left[\frac{c_{jk}\mathcal{N}(O_t, \mu_{jk}, \Sigma_{jk})}{\sum_{m=1}^{X} c_{jm}\mathcal{N}(O_t, \mu_{jm}, \Sigma_{jm})} \right] \tag{6.39}$$

and the final parameter re-estimation formulae can be written,

$$\mu'_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) O_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{6.40}$$

$$\Sigma'_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)(O_t - \mu_{jk})(O_t - \mu_{jk})'}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{6.41}$$

The component weights are re-estimated by calculating the ratio of the probability of being in state j for the kth mixture component and the probability of being in state j,

$$c'_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{X} \gamma_t(j,k)} \tag{6.42}$$

6.5 Application

Unless otherwise stated, all of the HMM recognition experiments presented in this thesis were implemented using the Hidden Markov Model Toolkit, HTK version 2.1 from Entropic [194].

Only continuous density HMM's were used, and because of the small size of both of the audio-visual databases used, Chapter 3, it was difficult to re-estimate model parameters for too complex HMM's. For this reason all HMM's were left to right with no skip transitions and used only diagonal covariance Gaussian mixture components (modes).

Model prototypes were initialised using a maximum of 20 iterations of the Viterbi algorithm to segment the training utterances and estimate model parameters. These initial model parameters were then refined using up to 40 iterations of Baum-Welch re-estimation.

All modelling was at the word-level and all data was hand labelled. For the visual speech features the labels were the name of the utterances, e.g. 'A' or '3'. The audio of the AVletters database was further segmented into periods of silence, utterance, silence to allow for the quiet periods before and after lip movement. For audio recognition an additional silence model was trained on these parts of the utterances and the recognition grammar was extended to allow an optional silence before and after the speech. HTK allows a further embedded re-estimation phase where composite models are re-estimated over utterance sequences. As only word-level models were used, and both databases were segmented into words, this was not required.

For most experiments the number of states and number of Gaussian modes per state were varied to find the best recognition performance.

Chapter 7

Visual Speech Recognition

This chapter describes visual speech recognition experiments using the low- and high-level analysis methods described in Chapters 4 and 5. The results are obtained using only the visual parts of either the AVletters or Tulips databases. Integrated audio-visual recognition will be discussed in the next chapter.

The first section covers the experiments and results obtained using the low-level one and two dimensional sieve-based multiscale spatial analysis (MSA). Initial experiments found that the area based features do not to perform as well as vertical length scale-histograms. A more detailed experiment to try and determine the best analysis parameters for scale-histograms across both databases is then described.

These results are compared to those obtained using the high-level active shape model (ASM) lip tracker for both databases. Initial experiments used only the single resolution tracker. These were extended to include results from the multi-resolution tracker, and per-talker shape modelling multi-resolution tracker, only for the larger AVletters database.

Finally the results obtained using the active appearance model (AAM) tracker are described for the AVletters database only.

7.1 Methods

As described in Section 6.5, all of the recognition results were obtained using left to right, continuous density HMM's with one or more diagonal covariance matrix Gaussian modes associated with each state.

In all cases the recognition task is word-level and multi-talker, i.e. trained and tested using examples from all talkers. Models are trained for all words (letters or digits) in the database using examples from all of the talkers. For the AVletters database the training set is the first two utterances of each of the letters from all talkers (520 utterances) and the test set is the third utterance from all talkers (260 utterances). For the Tulips database the training set is the first utterance of each digit (48 utterances), and the test set the second (48 utterances).

In many cases the HMM parameters were systematically varied to find the model topology that gives best recognition accuracy. Only the number of states and number of Gaussian modes per state were changed.

The effect of interpolating the data in time by a factor two was also investigated. This was first used to resample the visual features to the same rate as the audio (from 40ms to 20ms) for an early integration experiment, see next Chapter. However, this was found to have beneficial effect even for the visual-only recognition task. This is probably due to the small size of the database. Interpolation creates more data, on which the models can be better trained.

Delta features (differences between features) have been found by many researchers to be beneficial and are commonly used in acoustic speech recognition. These were also investigated, but found to give poor performance on all visual speech features. A key to the different feature vector types used in given in Table 7.1.

Vector Type	Meaning
N	normal
ND	normal + deltas
I	interpolated
ID	interpolated + deltas
D	deltas
DI	interpolated deltas

Table 7.1: Results tables key.

All tabulated recognition results are given in percent correct for the word-level task. The recognition grammar forced the classification of only a single letter for the AVletters database or single digit for the Tulips database. Unless stated otherwise all results were obtained using HTK version 2.1.

7.2 Multiscale Spatial Analysis Results

This section covers all of the low-level multiscale spatial analysis methods described in Chapter 4. The 2D area-sieve based analysis is described first, then the more successful 1D length-sieve based. All results were obtained for the AVletters database. Only the final scale-histogram results were also obtained for the Tulips database.

7.2.1 Area Tracking

Table 7.2 shows the recognition accuracies obtained using the simple area tracking described in Section 4.4.1 on the AVletters database. These results were obtained using five state HMM's with a single Gaussian mode associated with each state. Unlike later results these were obtained using HTK version 1.4. Because it was impossible to re-estimate variances using the Baum-Welch algorithm on this data, fixed variance HMM's were used. The variance was calculated directly from the training data. During training only the means and transition probabilities were updated.

As the recognition performance was so low these results are also shown using models trained and tested on individual talkers. These results are mostly not much above chance ($1/26 = 3.9\%$) and indicate that this method does not extract useful features.

7.2.2 Height and Width Tracking

Table 7.3 shows the recognition accuracies obtained using the height and width tracker described in Section 4.4.2. These results were also obtained using five state HMM's with a single Gaussian mode associated with each state, using HTK version 1.4. It was also impossible to re-estimate variances using the Baum-Welch algorithm on this data, so fixed variance HMM's were used.

These results are also generally poor, but the per-talker modelled results show that it was much more successful for some talkers. No vector type can be clearly identified as better than any other.

Talker	N	ND	I	ID	D	DI
1	7.7	11.5	3.9	7.7	7.7	7.7
2	3.9	3.9	11.5	7.7	3.9	3.9
3	26.9	11.5	7.7	7.7	3.9	11.5
4	7.7	15.4	11.5	15.4	19.2	7.7
5	19.2	23.1	15.4	23.1	3.9	0.0
6	11.5	11.5	11.5	7.7	15.4	7.7
7	7.7	11.5	7.7	7.7	3.9	3.9
8	26.9	23.1	15.4	23.1	23.1	19.2
9	11.5	3.9	7.7	7.7	0.0	3.9
10	11.5	7.7	7.7	3.9	7.7	7.7
All	6.9	8.5	8.9	7.7	7.7	8.5

Table 7.2: Per-talker results for area-tracking. Five state, single mode, fixed variance HMM. Recognition accuracy in %.

Talker	N	ND	I	ID	D	DI
1	15.4	15.4	7.7	3.9	7.7	7.7
2	11.5	7.7	7.7	3.9	3.9	3.9
3	19.2	11.5	11.5	11.5	7.7	11.5
4	42.3	42.3	42.3	42.3	30.8	19.2
5	15.4	23.1	15.4	23.1	3.9	3.9
6	19.2	15.4	15.4	15.4	15.4	7.7
7	15.4	15.4	7.7	11.5	11.5	15.4
8	23.1	34.6	26.9	30.8	15.4	11.5
9	7.7	3.9	7.7	11.5	0.0	0.0
10	3.9	7.7	7.7	7.7	7.7	3.9
All	8.9	7.3	5.8	7.3	7.3	9.6

Table 7.3: Per-talker results for height and width tracking. Five state, single mode, fixed variance HMM. Recognition accuracy in %.

7.2.3 Area Histogram

The results obtained using the area-histograms described Section 4.4.3 are shown in Table 7.4 and 7.5 for linear and squared spaced histograms respectively.

These results, and all others following, were not obtained using fixed variance HMM's. The variances were re-estimated, where possible, using the Baum-Welch algorithm. When it was not possible to re-estimate any model the experiment failed and this is indicated by a dash in the results tables. Fixed variance models were not used because it was found that whenever it was possible to re-estimate the variances, the accuracy was higher.

Results were obtained using area-histograms generated with positive extrema processing opening, o-sieves, negative extrema processing closing, c-sieves and bipolar extrema processing recursive median, m-sieves. For all of these PCA was used to reduce the 60 scale channel features to either the top 10 or 20 PCA components. Both covariance matrix and correlation matrix PCA calculation was used, denoted at Cov and Cor in the results tables.

		Cov		Cor	
Sieve Type	PCA	N	I	N	I
opening	10	18.9	15.0	8.9	9.2
	20	17.3	16.2	5.8	8.5
closing	10	-	-	8.9	8.9
	20	-	-	6.5	5.4
recursive median	10	18.5	21.9	8.1	9.6
	20	15.4	18.1	10.0	6.9

Table 7.4: Linear spaced area histogram results. Ten state, single mode HMM. Recognition accuracy in %. Dashes indicate models could not be trained.

		Cov		Cor	
Sieve Type	PCA	N	I	N	I
opening	10	13.1	11.2	10.8	6.5
	20	18.9	20.0	7.7	9.6
closing	10	16.9	16.5	17.3	13.9
	20	-	-	14.2	14.2
recursive median	10	17.3	18.1	15.0	18.5
	20	21.9	18.5	9.6	9.6

Table 7.5: Squared spaced area histogram results. Ten state, single mode HMM. Recognition accuracy in %.

These results are significantly higher than those obtained using either area tracking or height and width tracking so only the full multi-talker recognition task results are shown. The results show that calculating PCA with the covariance matrix was usually better. The best results were obtained using an m-sieve. Temporal interpolation was not always beneficial. There are no results using the exponentially spaced area-histogram channels because no models could be trained.

7.2.4 Scale Histogram

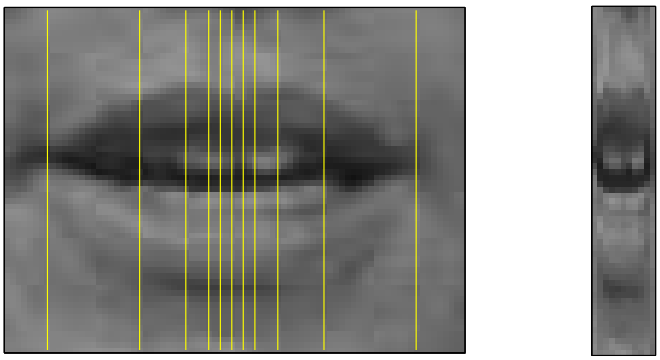
The results presented in this section were all obtained using visual features derived from the vertical scale-histograms discussed in Section 4.4.4. Some early results are presented first as

they set the direction for the more extensively tested later experiments carried out on both AVletters and Tulips databases.

Initial Experiments

The first published results obtained using 1D median, m-sieve derived vertical scale-histograms [127] considered only four of the ten talkers of the AVletters database. The first publication of the full database results [128] compared the recognition accuracies obtained after using PCA to reduce the dimensionality of the vertical scale histograms from 60, the height of the images. These experiments investigated the effect of using either 10 or 20 principal components to form visual features, calculated using both covariance and correlation matrices for comparison. See Section 4.5 and Appendix A for discussion.

Additionally, results were obtained using a non-sequential raster scan that concentrates on the centre of the mouth image. The motivation for this was to test the hypothesis that most of the information available in a vertical scale-histogram is mouth opening. Concentrating the analysis toward the centre columns of the image reduces the number of granules counted from the edges of the mouth, and biases in favour of those from the lip mid-line. Figure 7.1 illustrates this with an example image. In practice the exponential, centred, raster scan was implemented using the normal linear raster scan on an image constructed from the spaced scan lines.



(a) original image (b) rastered image

Figure 7.1: Exponential raster scanning prior to scale-histogram analysis. Image features away from the centre of the image do not contribute to the analysis. The original image (a) is column-wise sampled to form the horizontally subsampled image (b).

The initial experiments also compared the scale-histograms obtained using an m-sieve that preserves the baseline DC[1] component of the image with one that ignores this value during the analysis. As the example in Figure 4.12 showed it is possible for this to change the entire decomposition. The effect of temporal interpolation by a factor of two (from 40ms frame rate to 20ms) was also investigated. Table 7.6 summarises these results. These results were obtained using 10 state HMM's with each state associated with a single mode Gaussian.

The results show that for scale-histograms temporal interpolation of the data generally gives better accuracy. Principal component analysis using the covariance matrix was better if a DC ignoring sieve was used. If the DC level was preserved accuracy was improved by assuming all variables to have equal weight and using the correlation matrix.

[1] As discussed in Section 4.3.3.

			Cov		Cor	
DC	Raster	PCA	N	I	N	I
Preserve	linear	10	24.2	24.6	26.5	28.5
		20	25.4	23.5	22.3	27.3
	exponential	10	21.5	23.9	21.5	18.9
		20	24.6	26.5	21.9	23.9
Ignore	linear	10	30.4	30.0	21.9	23.9
		20	28.5	34.2	23.1	22.7
	exponential	10	23.5	23.9	16.9	19.2
		20	23.9	25.0	19.6	26.2

Table 7.6: Initial scale-histogram results. The scale-histograms of an m-sieve were tested using linear and exponential centred rastering for DC preserving and ignoring sieves. Number of PCA coefficients used and type of PCA were also tested. A ten state, single Gaussian mode HMM was used. Recognition accuracies in %.

The centre biased exponential raster scanned scale-histograms always gave lower results than simply scanning across the entire image. This suggests that useful information is obtained from the outer edges of the mouth images.

Full Experiments

The initial scale-histogram results were extended to cover three of the different types of sieve listed in Tables 4.1 and 4.2. As well as the median, m-sieve, which process maxima and minima, c-sieves (minima) and o-sieves (maxima) were used.

For these experiments the effects of altering the HMM parameters were also investigated. The number of states was either five, seven or nine and associated with each state were either one or three, diagonal covariance matrix, Gaussian modes. The full experimental parameters are listed in Table 7.7. A full investigation requires 288 experiments.

Attribute	Settings		
Sieve type	median, m	opening, o	closing, c
DC Baseline	preserve	ignore	
Interpolate	N, 25Hz	I, 50Hz	
PCA components	10	20	
PCA type	covariance	correlation	
HMM states	5	7	9
Gaussian modes	1	3	

Table 7.7: Scale-histogram experiment. All 288 possibilities were tried for all scale-histogram types and both databases.

Section 4.4.4 showed example scale-histograms formed by summing a function of the amplitude of each granule rather than simply counting the number at each scale. The experiment described by the parameters in Table 7.7 was repeated for each of these scale-histogram types. The different types of scale-histogram and their designation in the results tables is listed in Table 7.8.

This 288×4 experiment was extended by considering what happens if the scale-histograms are linearly scaled. For all of the scale-histogram Figures in Chapter 4 a linear scale factor was used to allow the visualisation of the high scales. Each scale histogram component was

Key	Type	Details
sh	scale count	count of granules at each scale
a	amplitude sum	sum of amplitudes at each scale
$\|a\|$	magnitude sum	sum of absolute amplitude at each scale
a^2	power sum	sum of squared amplitude at each scale

Table 7.8: Key to scale histogram types.

multiplied by its scale value. Generally the results obtained using correlation matrix principal components are poorer than those obtained with the covariance matrix. This suggest that all scales should not be treated equally for the PCA.

Finally, the 288×8 event experiment was doubled again to test everything on the Tulips database as well as AVletters. The final scale-histogram experiment involved testing $288 \times 16 = 4,608$ different configurations. This took approximately 500 hours of computation on an SGI O2 workstation. The full results of this are given in Appendix B and only the trends identified are presented in this section.

An example set of results is given in Table 7.10. These 288 results were obtained using an unscaled magnitude sum $|a|$ scale-histogram. The three types of sieve split the table into three columns. Each of these is split into two columns for DC ignoring (NoDc) and DC preserving (PresDc) sieves. For each of these the results are plotted column-wise for PCA using the covariance matrix (Cov) or correlation matrix (Cor). The rows cover each of the 24 different HMM topologies used. The key for HMM type is given in Table 7.9. For example, 20Is5m3 means 20 PCA coefficients from interpolated data modelled using a 5 state HMM with 3 Gaussian modes per state.

Attribute	Option		
No. PCA coeffs	10	20	
Vector type	N	I	
No. states	s5	s7	s9
No. modes	m1	m3	

Table 7.9: Key to HMM parameters.

The trends in these experiments can be most easily seen by plotting the results in Table 7.10 as intensity images for each of the different types of scale-histogram and for both databases. These are plotted in Figure 7.3 for the AVletters database and Figure 7.4 for the Tulips database. The results are plotted in the same way as in Table 7.10 and Figure 7.2 shows a key for interpreting them.

Clear trends can be seen in the results on the AVletters databases. The fine horizontal striping indicates that using three Gaussian modes is nearly always better than one. The central o-sieve column gives generally lower performance than either the left m-sieve or right c-sieve. The DC component ignoring columns are usually better than those sieved to preserve this value: note that DC makes no difference to the negative extrema processing c-sieve which often gives the best performance. The columns indicating PCA with the covariance matrix are generally higher than those for the correlation matrix. For linear scaled scale-histograms this is less pronounced, and overall the results show that linear scaling was not beneficial. Linear scaling of the vectors should have no effect on the results obtained using PCA of the correlation matrix because of the variance normalisation. The differences seen are due to errors from the numerical precision of the calculation of the correlation matrix. That these

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	21.54	15.38	16.54	14.62	20.00	11.92	16.54	15.00	20.77	14.62	20.77	14.62
10Ns5m3	35.77	29.23	29.62	NaN	29.23	28.46	25.38	25.38	33.46	32.69	33.46	32.69
10Ns7m1	22.69	18.46	22.69	18.46	23.85	13.85	19.23	18.46	23.85	20.00	23.85	20.00
10Ns7m3	35.77	31.15	29.23	29.23	35.00	30.38	28.46	27.69	36.15	30.00	36.15	30.00
10Ns9m1	24.23	21.15	21.54	17.31	29.62	11.92	22.69	21.54	28.08	21.54	28.08	21.54
10Ns9m3	35.77	30.38	29.62	33.85	36.54	28.85	29.62	29.23	37.31	34.62	37.31	34.62
10Is5m1	18.46	15.38	16.92	13.46	19.62	13.46	19.23	16.15	16.54	16.54	16.54	16.54
10Is5m3	32.69	30.38	28.46	26.92	33.46	28.08	25.00	28.08	30.77	27.69	30.77	27.69
10Is7m1	21.15	18.85	23.08	18.08	26.15	16.54	19.23	21.92	25.38	20.77	25.38	20.77
10Is7m3	33.85	30.38	31.15	29.23	34.62	28.46	27.31	30.38	37.69	33.08	37.69	NaN
10Is9m1	25.00	20.38	23.46	22.69	29.23	14.62	24.62	20.77	30.00	23.08	30.00	23.08
10Is9m3	37.69	33.08	34.23	32.31	38.85	26.54	30.77	29.23	37.31	35.38	37.31	35.38
20Ns5m1	20.77	16.15	19.23	15.77	20.38	15.38	28.85	16.15	22.31	14.62	22.31	14.62
20Ns5m3	35.77	28.08	25.38	26.15	31.15	31.15	31.92	24.23	40.00	28.46	40.00	28.46
20Ns7m1	25.77	18.46	20.00	20.38	20.77	18.08	29.62	21.54	26.92	24.23	26.92	24.23
20Ns7m3	38.85	30.00	26.15	30.38	33.85	31.54	33.46	29.62	36.15	34.62	36.15	34.62
20Ns9m1	28.08	17.69	23.85	18.46	24.62	22.31	31.15	26.15	30.00	25.00	30.00	25.00
20Ns9m3	39.62	31.15	31.54	31.92	33.85	30.00	36.54	35.00	41.54	35.77	41.54	35.77
20Is5m1	19.62	15.00	15.77	13.85	18.85	16.54	23.46	17.69	24.62	15.38	24.62	15.38
20Is5m3	36.15	28.08	25.00	23.46	29.62	27.69	31.92	28.46	36.15	31.54	36.15	31.54
20Is7m1	28.08	20.00	20.00	19.62	25.77	20.00	30.00	22.31	27.31	22.31	27.31	22.31
20Is7m3	36.92	32.69	27.31	30.38	33.85	32.69	31.54	28.85	36.54	35.77	36.54	35.77
20Is9m1	30.00	20.00	23.85	20.38	27.69	21.92	30.00	21.92	32.69	25.38	32.69	25.38
20Is9m3	40.77	35.38	29.23	28.85	32.31	31.92	30.00	33.85	44.62	38.08	44.62	38.08

Table 7.10: Example set of results using unscaled $|a|$ scale-histograms on the AVletters database.

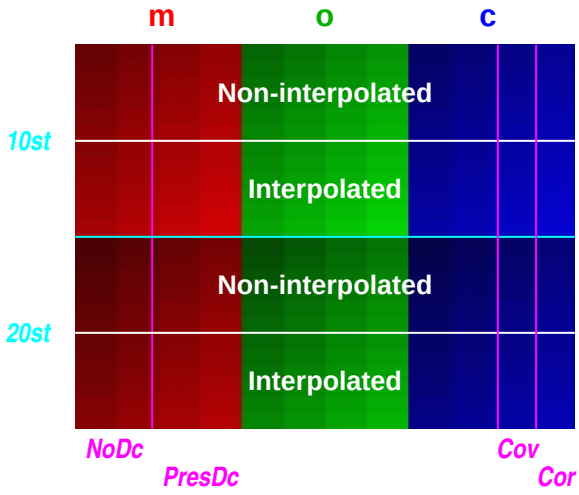


Figure 7.2: Key to scale-histogram result Figures.

small differences have an effect on the PCA and resulting features and recognition accuracy demonstrates how statistically fragile these results are. The highest results are in the lower halves of the tables, showing the results using twenty PCA coefficients are better than those using ten.

Some of these trends are also seen on the results from the Tulips database. The covariance matrix derived PCA features are distinctly higher than the correlation results. Also, the o-sieve column is the poorest, and the linear scaled histogram results are lower than the unscaled. Unlike the AVletters results the best results on Tulips are obtained using single a Gaussian mode per state. In some cases three modes per state models could not be re-estimated: these are shown by the horizontal black lines through Figure 7.4 indicating no result. The best results are in the upper halves of the tables, for ten PCA component features. For the m- and o-sieves features from DC preserving sieves are usually the better, there is no difference for c-sieves.

These results suggest dispensing with all linear scaled, o-sieve and correlation matrix results. This leaves just m- or c- sieves with features formed from the top ten or twenty coefficients of a PCA analysis of the covariance matrix of the unscaled scale-histograms. Taken across both databases it is better to sieve ignoring the DC value.

These best-subset results are plotted for the AVletters database in Figure 7.5 for the top twenty PCA features. For the Tulips database ten PCA coefficients is better and these are likewise plotted in Figure 7.6.

Summary of Scale-Histogram Results

Figures 7.5 and 7.6 show that it is difficult to clearly state either the best type of sieve or the best type of scale-histogram to use. Comparing the results across both databases the best results are obtained using closing c-sieves on interpolated data from amplitude sum a scale-histograms. Note that there is no difference between amplitude sum, a, and magnitude sum $|a|$ for either o- or c-sieves because they extract, respectively, only positive or negative extrema. The magnitude sum is identical to an amplitude sum, apart from a sign change in the case of a c-sieve.

The best features were formed from either the top ten (Tulips) or twenty (AVletters) PCA components from the covariance matrix of temporally interpolated data. These are summarised in Table 7.11. The best result on the Tulips database is 77.1% correct. For the

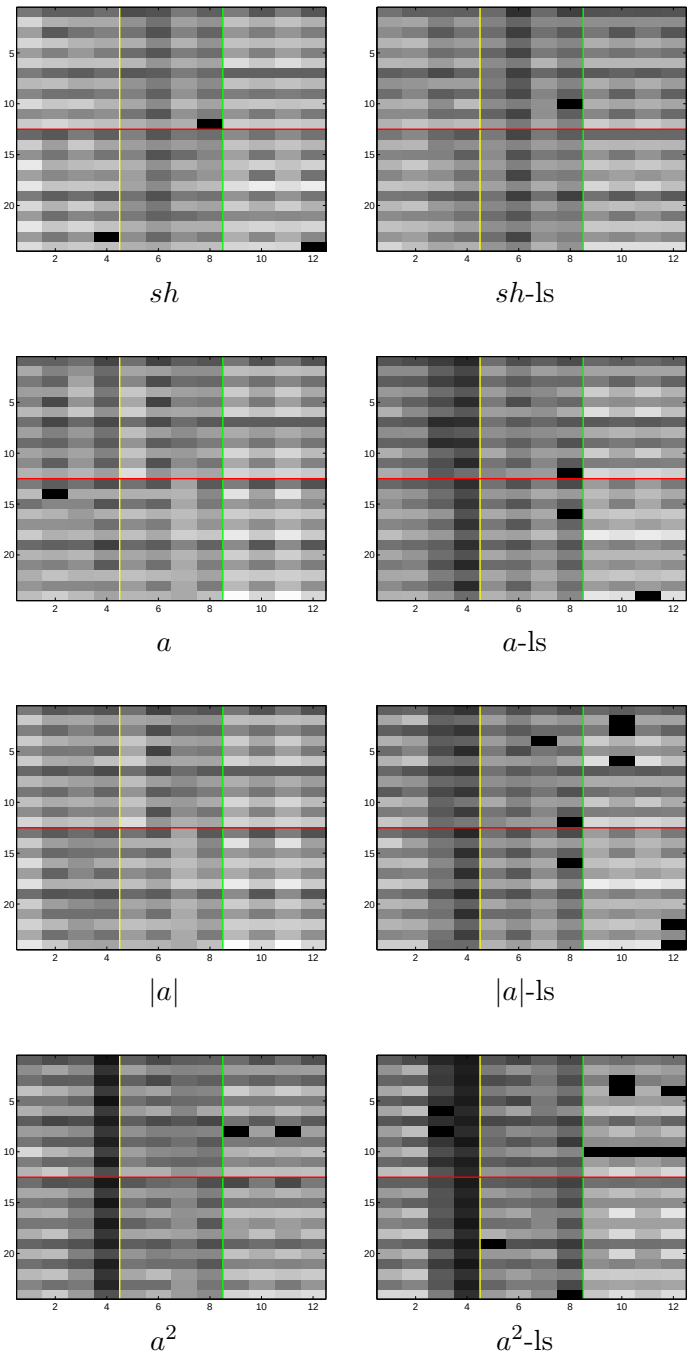


Figure 7.3: Scale-histogram results for the AVletters database. Left column plot recognition accuracy as greylevel intensity for the four types of scale histogram. Right column is the same for the linearly scaled histograms. All to the same intensity scale, white is high.

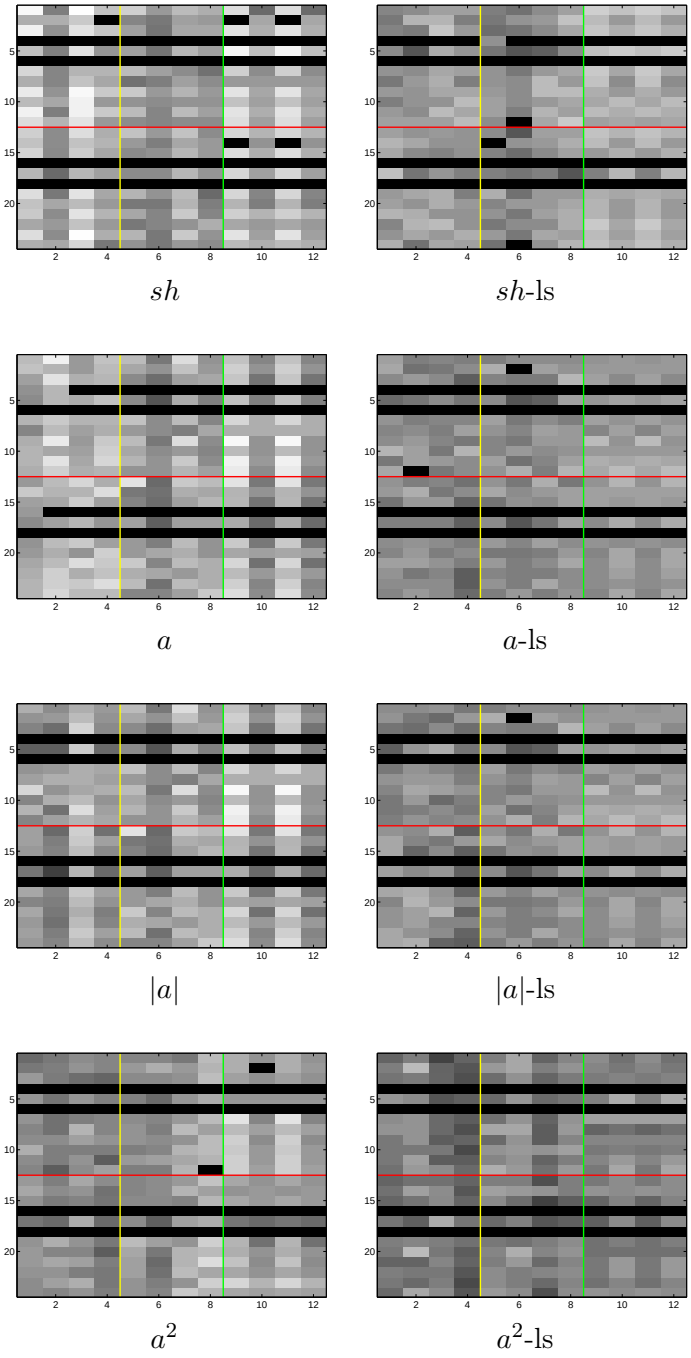


Figure 7.4: Scale-histogram results for the Tulips database. Left column plot recognition accuracy as greylevel intensity for the four types of scale histogram. Right column is the same for the linearly scaled histograms. All to the same intensity scale, white is high.

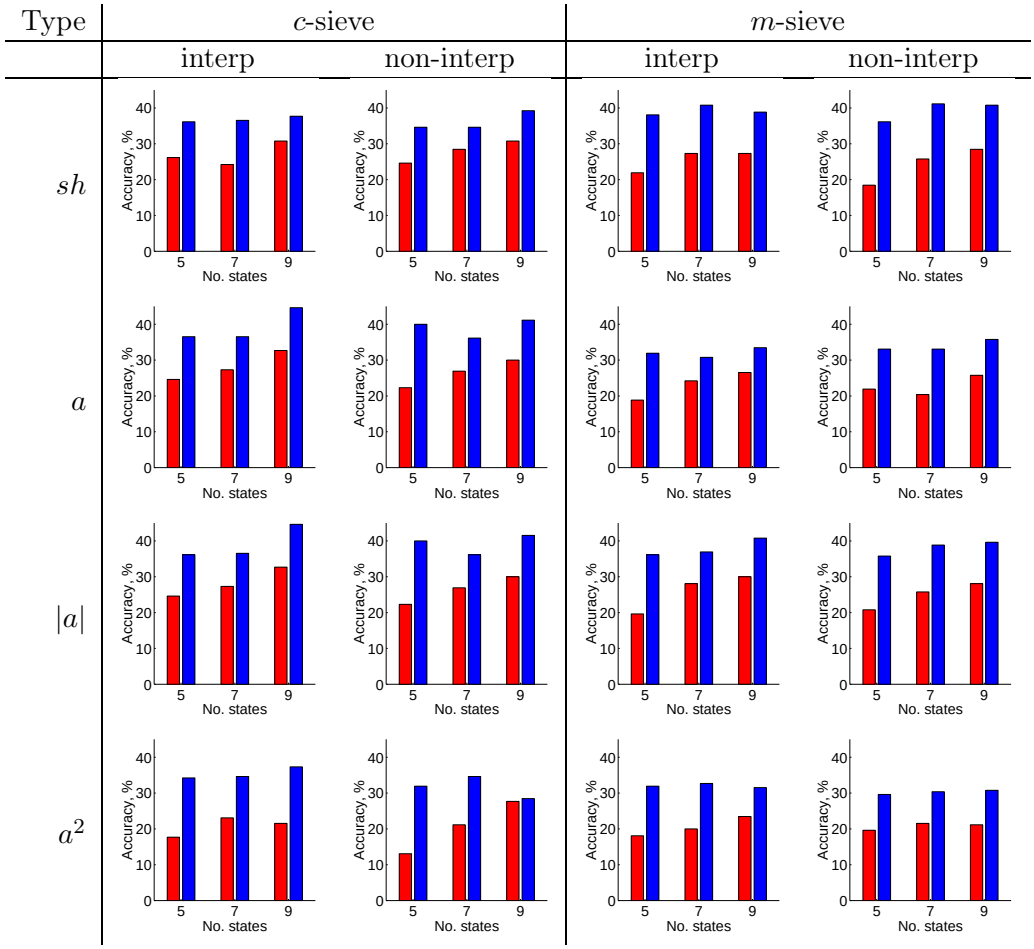


Figure 7.5: Best scale-histogram results for the AVletters database. Left c-sieve. Right m-sieve. Shows how varying the HMM parameters: number of states (abscissa) and Gaussian modes (blue columns, 3, red, 1) affects recognition accuracy (ordinate) for interpolated and non-interpolated AVletters data. Features are the top twenty components from PCA using the covariance matrix. The scale-histograms are formed using DC ignoring sieves.

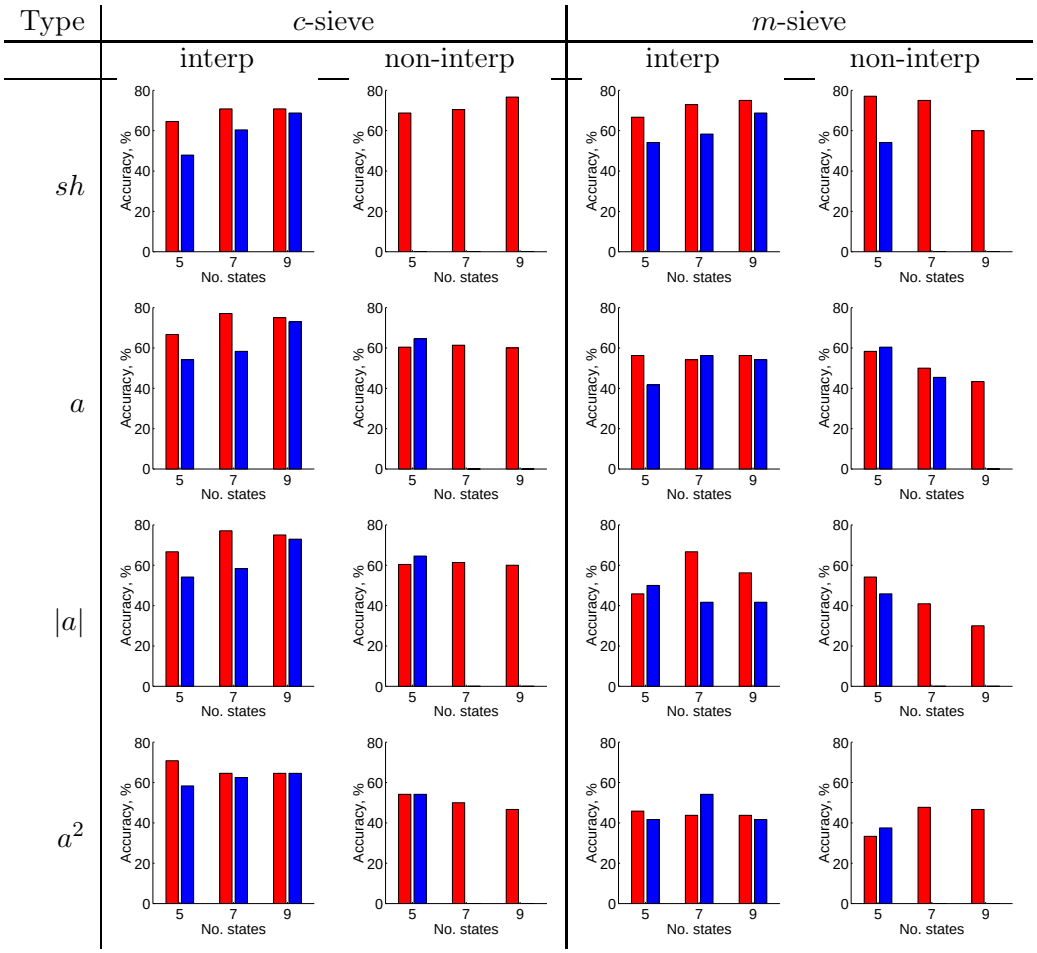


Figure 7.6: Best scale-histogram results for the Tulips database. Left c-sieve. Right m-sieve. Shows how varying the HMM parameters: number of states (abscissa) and Gaussian modes (blue columns, 3, red, 1) affects recognition accuracy (ordinate) for interpolated and non-interpolated Tulips data. Features are the top ten components from PCA using the covariance matrix. The scale-histograms are formed using DC ignoring sieves.

larger and more complex AVletters database the best result is 44.6% correct.

States		5		7		9	
Modes		1	3	1	3	1	3
AVletters	10	16.5	30.8	25.4	37.7	30.0	37.3
	20	24.6	36.1	27.3	36.5	32.7	**44.6**
Tulips	10	66.7	54.2	**77.1**	58.3	75.0	72.9
	20	62.5	52.1	66.7	58.3	64.6	68.7

Table 7.11: Scale-histogram best recognition accuracies, %, for Tulips and AVletters with variations in the HMM parameters: no. states and no. Gaussian modes per state. Top half shows results for 10 PCA coefficients, bottom half for 20 PCA coefficients.

This suggests that when using sieves to extract visual speech features on either of the databases considered it is the dark image regions that capture most information.

In practice, if lighting conditions were such that the mouth cavity was brighter than the face this would no longer capture the same information. However, this would be a rather strange situation almost certainly involving direct light into the mouth. Features derived using bipolar recursive median sieves have similar performance on the AVletters database and might be expected to be more robust to different skin tone and lighting conditions. However, the more brightly illuminated Tulips database the results show recursive median sieves perform significantly poorer than closing sieves. A solution to this problem may be to only measure the negative extrema obtained using an m-sieve.

These results do not clearly indicate that using amplitude information as well as scale is useful. Although the best results were obtained using magnitude sum scale histograms these were only marginally better than summing only the number of granules at each scale. Also, they introduce a dependency on the lighting conditions. The magnitude squared scale-histograms were always the poorest which suggests that it is incorrect to assume that high amplitude granules should be considered more significant than low amplitude granules.

7.3 Active Shape Model Results

The results in this section were all obtained using the active shape model (ASM) lip trackers described in Section 5.1. In all cases the visual speech features were the tracked sequence of seven dimensional vectors defining points in the space of valid lip shapes represented by the point distribution model (PDM). All of the results were obtained using a simplex minimisation fit on each image. Results were obtained using various fitting algorithms under different modelling conditions. Initial results were obtained for both AVletters and Tulips databases. Further development of the tracking algorithm concentrated on the more challenging AVletters task.

7.3.1 Basic Tracker

The first ASM lip tracker results on the AVletters database were published in [129]. These were obtained using the single resolution tracker described in Section 5.1.3. To improve the accuracy separate greylevel profile models (GLDM's) were built for each talker. The shape parameters, $b_1, b_2, \ldots, b_{t_s}$, formed the features for recognition.

The results in this section were obtained using the standard MATLAB implementation of simplex minimisation. This differs slightly from the Numerical Recipes algorithm used for

later results as it constructs the initial simplex using fractional perturbations from the initial conditions. This implementation takes several seconds to fit to each image frame.

Results were obtained using this method for both AVletters and Tulips databases by initialising the PDM at the mean shape in the centre of each image in turn. These results are given in Table 7.12 for a range of HMM parameters. The dashes indicate models topologies that could not be trained.

States		3			5			7			9		
Modes		1	3	5	1	3	5	1	3	5	1	3	5
AVletters	N	10.8	15.0	13.4	15.8	17.7	14.2	17.3	18.9	17.3	17.7	-	-
	I	16.2	15.4	16.9	18.1	21.5	18.1	21.2	20.8	-	22.3	-	-
Tulips	N	56.3	56.3	47.9	58.3	56.3	-	75.0	-	-	76.7	-	-
	I	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.12: Initial ASM tracker results for AVletters and Tulips databases. Recognition accuracy in % for different HMM states and Gaussian modes per state. N indicates raw data, I temporally interpolated by two. Dashes mean models could not be trained.

Unsurprisingly the best accuracy on the simpler Tulips database is much higher at 76.7% than obtained on the harder AVletters database, 22.3%. The Tulips result compares very closely to the best 77.1% obtained using MSA in the previous section. The AVletters result is significantly lower than the 44.6% MSA result. This may be due to two effects; firstly that the ASM shape parameters do not have as much discrimination over the larger letters task, and secondly that the lip contours in AVletters are harder to track.

7.3.2 Multi-resolution Tracker

The initial ASM results were extended using the two-stage coarse to fine multi-resolution tracker described in Section 5.1.5. This implementation used the Numerical Recipes simplex code which differs from the MATLAB code by initialising the simplex with user specified perturbations. This compiled code also uses only single floating point precision rather than double. Simplex iterations are much faster than the interpreted MATLAB code, for the lip contour model each takes approximately 0.5ms on an R10000 175MHz SGI O2 workstation. Results were only obtained for the more challenging AVletters database.

Four different tracking algorithms were tried. The first initialised each PDM at the mean shape in the centre of all images. The second starts at the mean shape and centre of the first image and subsequent images are initialised at the previous fit shape and pose. These two methods were repeated using either nine or eleven pixel length greylevel profile models. The initial simplex was constructed by perturbing the initial position by a translation of two pixels in both x and y, an increase in scale of 10%, a rotation of 0.2 radians and each shape parameter was increased by 0.5 standard deviations. These experiments are summarised in Table 7.13.

The effect on recognition accuracy was also compared for greylevel modelling for each talker (local GLDM's) and using a single model for all talkers (global GLDM). The results using per-talker, local GLDM's are given in Table 7.14 for experiment ASM1 and Tables 7.15, 7.16 and 7.17 for experiments ASM2, ASM3 and ASM4 respectively.

These experiments show that the longer (eleven pixel length) greylevel profile models track more reliably than the nine pixel models. It is also better to initialise the tracker fresh on each image rather than from the last fit position. This works better because it is hard for the simplex search to recover from positions away from the local minima. Once the tracker

Experiment	Attribute	Option
ASM1	Greylevel profile Initialisation	9 pixels last fit
ASM2	Greylevel profile Initialisation	11 pixels last fit
ASM3	Greylevel profile Initialisation	9 pixels centred mean shape
ASM4	Greylevel profile Initialisation	11 pixels centred mean shape

Table 7.13: ASM experiment summary.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	10.0	-	18.5	-	-	25.0	-	-	-	-	-	-
I	-	-	-	-	-	-	-	-	-	-	-	-
ND	10.0	-	-	-	-	-	-	-	-	-	-	-
ID	9.2	14.6	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.14: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database. Experiment ASM1 – GLDM length 9 pixels, initialised at last frames fit position.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	8.9	16.2	-	10.8	16.5	23.1	13.9	20.8	-	-	-	-
I	8.9	11.9	-	9.6	21.5	-	-	23.1	24.2	18.1	-	23.1
ND	-	-	-	-	-	-	-	-	-	-	-	-
ID	9.6	14.6	18.5	11.9	-	18.5	9.2	-	-	12.3	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.15: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database. Experiment ASM2 – GLDM length 11 pixels, initialise at last frames fit position.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	9.6	19.2	19.2	14.2	22.7	25.0	17.7	24.6	25.0	17.7	26.9	25.4
I	9.6	20.0	21.2	11.2	25.0	25.4	17.7	26.9	24.2	20.0	25.4	25.4
ND	10.8	-	-	-	-	-	-	-	-	-	-	-
ID	11.5	-	18.5	12.7	20.0	-	17.7	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.16: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database. Experiment ASM3 – GLDM length 9 pixels, initialise with the mean shape at the centre of each image.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	10.0	18.5	19.2	13.5	23.1	25.4	14.6	21.5	26.2	20.8	23.5	25.4
I	6.6	16.5	19.2	10.4	19.2	21.2	15.8	25.8	24.6	18.5	22.7	26.9
ND	-	-	-	-	-	-	-	-	-	-	-	-
ID	8.9	15.0	18.1	10.0	-	26.2	13.1	-	-	14.2	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.17: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database. Experiment ASM4 – GLDM length 11 pixels, initialise with the mean shape at the centre of each image.

has lost the lip contour the tracking fails for the rest of the sequence. Initialising at the mean shape in the centre of each frame ignores the knowledge of the previous lip shape but is able to recover from tracking errors in individual images. Better results can be expected by using for example a statistical temporal tracking framework [17, 93] to model the dynamic nature of the lips.

The corresponding results obtained using a the global greylevel profile model trained across all talkers are given in Tables 7.18, 7.19, 7.20 and 7.21. These results are always lower than those obtained using local GLDM's trained for each talker. The differences between the talkers are large and in capturing all this variation the database wide model becomes less specific to lip contours.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	6.9	6.5	-	8.9	-	-	-	-	-	-	-	-
I	6.2	-	5.8	6.5	8.9	9.2	-	8.5	6.9	8.1	7.7	6.2
ND	8.9	-	-	6.5	-	-	10.0	-	-	-	-	-
ID	7.3	7.3	-	-	-	-	8.5	-	-	10.0	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.18: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using a database wide GLDM. Experiment ASM1 – GLDM length 9 pixels, initialise at last frames fit position.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	7.3	8.5	6.9	9.2	7.3	10.0	10.8	9.6	5.8	9.6	8.1	7.3
I	6.2	9.2	8.5	7.7	8.9	8.9	8.1	10.0	10.0	7.7	10.8	10.0
ND	10.8	-	-	-	-	-	-	-	-	-	-	-
ID	9.2	-	6.5	6.9	-	-	8.1	-	-	10.8	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.19: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using a database wide GLDM. Experiment ASM2 – GLDM length 11 pixels, initialise at last frames fit position.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	8.1	11.5	10.4	10.0	15.0	-	9.23	11.2	7.3	10.8	16.9	10.8
I	8.5	9.2	9.6	11.5	12.7	11.2	11.5	11.5	9.6	13.9	13.1	8.5
ND	5.4	-	-	-	-	-	-	-	-	-	-	-
ID	10.0	-	-	10.0	-	-	10.0	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.20: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using a database wide GLDM. Experiment ASM3 – GLDM length 9 pixels, initialise with the mean shape at the centre of each image.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	10.0	10.8	9.6	10.0	11.5	-	14.2	10.4	9.2	11.9	12.3	-
I	11.2	11.2	9.6	7.7	13.9	8.9	12.3	13.1	10.0	12.3	11.2	-
ND	10.8	-	-	-	-	-	-	-	-	-	-	-
ID	8.9	10.0	-	9.2	-	3.9	10.0	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.21: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using a database wide GLDM. Experiment ASM4 – GLDM length 11 pixels, initialise with the mean shape at the centre of each image.

7.3.3 Per-talker Multi-resolution Tracker

The results in the previous section clearly show that using a separate GLDM for each talker gives better performance. This is because the cost function does not consider the inter-talker variation and is more selective. The shape space of the simplex search can also be constrained for each individual talker by using the PDM trained for each talker.

To be able to train multi-talker HMM's the individuals fit parameters were mapped through the landmark point coordinates to the best fit available using the talker independent PDM. This is described in Section 5.1.6. The results for the four conditions in Table 7.13 are given in Tables 7.22, 7.23, 7.24 and 7.25.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	5.4	10.4	13.5	10.0	11.5	20.8	11.5	17.3	17.7	14.6	19.2	20.8
I	4.6	10.0	13.5	9.6	15.8	17.7	8.1	15.8	20.8	13.1	16.9	20.8
ND	7.3	12.3	-	6.9	-	21.9	11.5	-	-	-	-	-
ID	5.8	11.2	16.2	9.2	14.6	19.2	10.4	-	-	11.2	-	-
D	-	-	-	-	-	-	-	-	-	-	-	-

Table 7.22: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using per-talker GLDM's and per-talker PDM's. Experiment ASM1 – GLDM length 9 pixels, initialise at last frames fit position.

Despite better tracking performance the per-talker PDM results do not improve on those obtained using the database wide PDM. This is due to the large variation between talkers, see Figure 5.17. Mapping low dimensional talker dependent axes is only sensible if the rotations

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	3.9	9.2	15.0	10.8	14.2	17.7	12.7	14.6	21.5	12.3	18.9	21.2
I	5.4	10.8	-	10.8	12.3	-	8.5	13.9	26.5	11.9	16.5	-
ND	5.38	-	14.2	9.6	-	-	11.5	-	-	15.8	-	-
ID	7.3	11.2	15.8	6.9	16.2	17.7	7.3	-	19.6	14.2	-	-
D	6.9	-	-	-	-	-	-	-	-	-	-	-

Table 7.23: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using per-talker GLDM's and per-talker PDM's. Experiment ASM2 – GLDM length 11 pixels, initialise at last frames fit position.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	10.4	11.5	15.4	10.4	13.5	20.0	11.9	18.1	17.3	15.0	17.3	21.9
I	5.8	11.9	11.2	10.4	13.9	20.0	10.7	16.2	-	11.5	18.5	-
ND	8.9	-	13.5	11.5	-	21.5	15.4	-	-	-	-	-
ID	9.2	8.5	-	9.6	17.3	-	10.8	-	-	11.2	-	-
D	6.9	-	-	7.7	-	-	-	-	-	-	-	-

Table 7.24: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using per-talker GLDM's and per-talker PDM's. Experiment ASM3 – GLDM length 9 pixels, initialise with the mean shape at the centre of each image.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	6.2	10.8	13.5	10.4	15.8	19.6	14.2	16.2	20.0	17.7	18.5	21.5
I	6.2	8.9	12.7	10.8	15.0	20.4	12.7	15.8	21.2	12.3	16.9	23.5
ND	8.5	-	-	9.2	-	-	11.2	-	-	-	-	-
ID	7.7	10.8	11.2	9.2	-	19.2	10.0	-	-	14.6	-	-
D	8.9	-	-	3.9	-	-	-	-	-	-	-	-

Table 7.25: Recognition accuracy in % for multi-resolution ASM lip tracking on the AVletters database using per-talker GLDM's and per-talker PDM's. Experiment ASM4 – GLDM length 11 pixels, initialise with the mean shape at the centre of each image.

map the same sort of modes onto the same global axes. Figure 5.17 shows that in some cases the per-talker modes are similar to the global PDM, but generally this mapping cannot be guaranteed because of the wide range in talker lip shapes and talking style.

7.4 Active Appearance Model Results

This section describes the recognition results obtained using the active appearance model (AAM) tracker described in Section 5.2. The features were formed from the 37 dimensional vectors that define a point in the combined shape and greylevel appearance space. Initially the full 37 dimensional features were used. Additional experiments investigated using only the top 20, 10 or 5. No temporal modelling was used, each frame was initialised with the mean shape and appearance at the centre of the low resolution image.

The results are shown in Table 7.26 for the full 37 dimensional vectors, and Tables 7.27, 7.28 and 7.29 for the 20, 10 and 5 dimensional vectors.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	21.2	26.5	31.5	23.1	34.2	39.2	29.6	37.3	40.8	29.2	37.3	36.9
I	20.4	27.7	34.6	23.1	32.3	41.9	30.0	38.5	39.2	31.9	36.9	38.9
ND	18.5	-	-	24.2	-	-	24.2	-	-	-	-	-
ID	18.9	26.2	-	23.5	-	41.2	28.9	37.7	-	25.4	-	-
D	6.9	-	-	8.9	-	-	-	-	-	-	-	-

Table 7.26: Recognition accuracy in % for multi-resolution AAM lip tracking on the AVletters database. Full 37 dimensional vectors.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	18.5	27.7	30.0	22.7	34.2	36.2	26.5	36.9	38.5	27.3	40.8	37.3
I	17.7	26.5	33.1	23.9	33.9	41.5	27.3	35.0	40.8	30.0	36.9	39.6
ND	17.3	18.9	-	23.1	-	-	26.9	-	-	-	-	-
ID	18.5	24.6	36.9	21.9	-	37.7	21.9	35.0	-	29.3	-	-
D	7.3	-	-	10.0	-	-	-	-	-	-	-	-

Table 7.27: Recognition accuracy in % for multi-resolution AAM lip tracking on the AVletters database. 20 dimensional vectors.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	14.2	26.9	32.3	19.2	33.1	33.5	23.5	30.0	-	24.2	32.7	-
I	10.8	25.4	29.2	16.5	28.1	35.4	23.1	33.1	37.3	23.1	36.2	38.1
ND	14.6	-	-	19.2	-	-	-	-	-	-	-	-
ID	11.9	20.8	-	22.7	26.2	37.7	19.2	-	-	20.4	-	-
D	7.7	-	-	-	-	-	-	-	-	-	-	-

Table 7.28: Recognition accuracy in % for multi-resolution AAM lip tracking on the AVletters database. 10 dimensional vectors.

States	3			5			7			9		
Modes	1	3	5	1	3	5	1	3	5	1	3	5
N	12.7	19.2	21.9	15.8	26.9	28.1	20.0	24.6	-	20.4	27.3	-
I	11.2	19.6	-	16.2	25.4	-	18.9	32.7	31.2	19.2	28.9	-
ND	13.9	-	-	-	-	-	-	-	-	-	-	-
ID	12.3	18.9	-	18.5	31.9	30.4	20.4	-	-	20.0	-	-
D	6.2	-	-	-	-	-	-	-	-	-	-	-

Table 7.29: Recognition accuracy in % for multi-resolution AAM lip tracking on the AVletters database. 5 dimensional vectors.

7.5 Summary of Results

The best results, on both databases and for all methods were obtained using MSA. In nearly all cases, for all methods, temporally interpolating the data give improved recognition accuracy. All of the best results were obtained using interpolated data.

The best MSA results were obtained using a magnitude sum scale-histogram calculated using a closing sieve and PCA transformed to the top 20 directions using the covariance matrix. The best ASM results were obtained using the dual-resolution tracker with separate GLDM's for each talker. The best AAM results used all 37 components of the appearance vectors. These results are summarised in Table 7.30. (For this summary additional five Gaussian mode experiments were run for MSA under the best conditions—the best result remained unchanged.)

States	5			7			9		
Modes	1	3	5	1	3	5	1	3	5
ASM	10.4	19.2	21.2	15.8	25.8	24.6	18.5	22.7	**26.9**
AAM	23.1	32.3	**41.9**	30.0	38.5	39.2	31.9	36.9	38.9
MSA	24.6	36.1	41.5	27.3	36.5	40.4	32.7	**44.6**	41.2

Table 7.30: Summary of best visual recognition results. Recognition accuracy in % for ASM, AAM and MSA on the AVletters database.

These results show that using only shape information, the ASM lip tracker, is not as effective as combining with greylevel appearance, the AAM lip tracker, or analysing the image by scale, MSA. This finding is in agreement with Bregler [26] who used his lip tracker only to locate the area for eigen analysis and Luettin [107] who combined the GLDM and PDM parameters of his ASM lip tracker for improved performance.

Chapter 8

Audio-Visual Speech Recognition

This chapter describes combined audio-visual speech recognition experiments for all of the different visual speech features described in the previous chapters. This is the ultimate test of any visual speech feature; how much contribution does it add when combined with acoustic speech features?

The benefits humans gain by combining auditory and visual speech have long been known [14, 66, 145, 148, 182], and it has been clearly demonstrated that speech perception is bimodal [117, 135]. However, precisely how and when humans integrate the auditory and visual stimuli is still unknown [22,75,122,123,163,169,183], but appears likely to occur before phonetic categorisation. Broadly, the various integration strategies can be classed as either *early* or *late*. With early integration the auditory and visual information are combined before classification. For late integration separate classifiers operate on each modality and an integrated result is obtained by somehow fusing the two results.

Many combined audio-visual speech recognition systems have been reported. The first audio-visual speech recognition system by Petajan [151, 152] used late integration and combined the ranked results from each recogniser with a set of heuristics. Other examples of late integration are [23,56,63,119,131,157,175,178]. Examples of systems using early integration are [24, 27, 28, 34, 93, 155, 177, 186, 191, 195, 196]. Also, several comparisons of early and late integration have been made [1, 84, 98, 136, 142, 164]. Reviews of several different audio-visual speech recognition systems can be found in Goldschen [72] and Henneke [85].

This chapter discusses and compares both early and late integration strategies for an audio-visual speech recognition task on the AVletters database. Results are obtained over a range of noise conditions using the lipreading results of all three methods described in chapters 4 and 5.

8.1 Audio-visual Integration

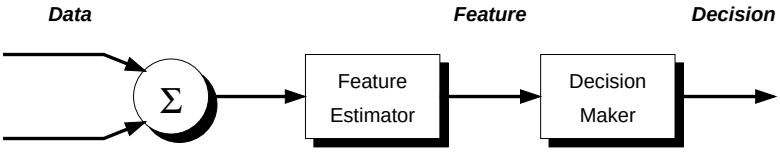
Audio-visual speech integration is a subset of the general problem of multimodal (and multisensor) data fusion. The problem of integrating multiple sources of information occurs in many other fields, for example, medical imaging, military target recognition [77] and location [167], scene analysis [138] and human computer interaction (HCI) [76,174]. The primary question is; at what level should data fusion occur? Generally, three distinct levels are considered,

1. data fusion;

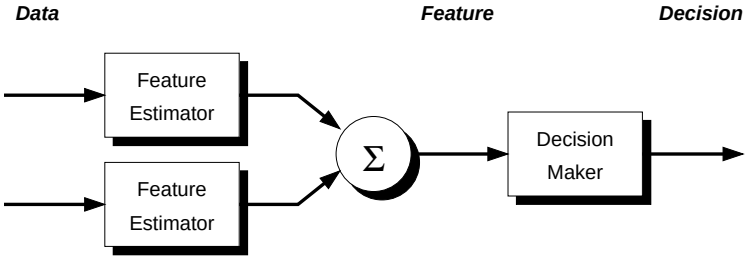
2. feature fusion;

3. decision fusion.

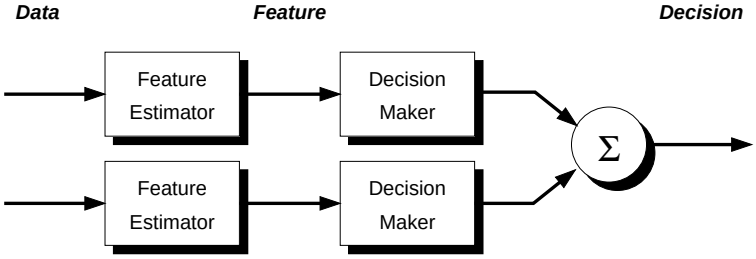
These are illustrated in Figure 8.1. A further refinement of this classification introduces additional levels between 1 and 2, and 2 and 3 [59].



(a) Data fusion.



(b) Feature fusion (*early integration*).



(c) Decision fusion (*late integration*).

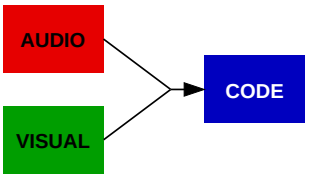
Figure 8.1: Levels of data fusion.

According to this classification data fusion is the lowest level and involves integrating the raw data. This can only occur if the data is of the same type (for example averaging the result of two or more thermometers) and so is not immediately applicable to audio-visual speech integration. Feature fusion integrates the different modalities after features have been extracted from the raw data. For example, audio and visual parameters can be concatenated to form a single, composite audio-visual speech vector. This is otherwise known as early integration.

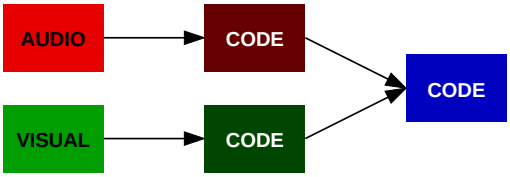
The term 'decision level fusion' is used when decisions are made on the features of each modality separately. For example, two recognisers are used, one for visual features and one for audio. The final classification is made by combining the output of both classifiers to find the most likely utterance. This is usually called late integration.

Robert-Ribes *et. al* [163, 169] have proposed four models for audio-visual integration. The first is *direct identification* (DI) which is identical to feature fusion (early integration). The second, *separate identification*, is the same as decision fusion (late integration). The remaining models attempt data fusion by remapping the auditory and visual data into a common space. The *dominant recoding* (DR) model considers the auditory modality to be the dominant and maps the visual input to an auditory representation. The *motor recoding* (MR) model maps both auditory and visual data into an amodal, neither auditory or visual,

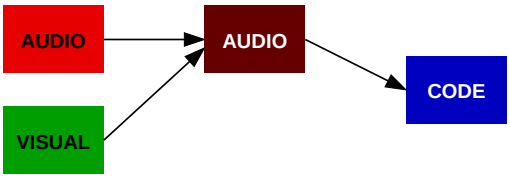
space. These are illustrated in Figure 8.2.



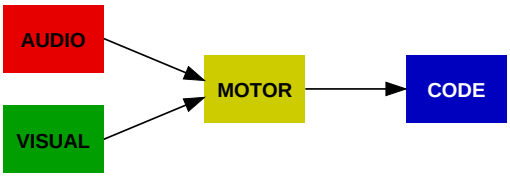
(a) DI, direct identification (*early integration*)



(b) SI, separate identification (*late integration*)



(c) DR, dominant recoding



(d) MR, motor recoding

Figure 8.2: Audio-visual integration models proposed by [163, 169].

So far only one audio-visual speech recognition system has been described using the MR model [163], although no results were given. An example of DR is the mapping of static vowel images to amplitude spectra by Yuhas [195,196]. Most other implementations use either early (DI) or late (SI) integration. For current practical systems there is good reason for this. The remapping function required for both DR and MR models will have to be statistically learnt from the training data. Given the limited amount of audio-visual training data currently available, Chapter 3, it is already difficult to build statistical models for recognition. Further modelling error due to a poor mapping function at the data level is undesirable. For these reasons DR and MR models are not discussed further.

There are several issues when choosing between early and late integration. Visual features are often generated at a slower rate than audio features. The primary reason for this is the widespread use of TV specification cameras and frame grabbers operating at 50/60 fields per second. The use of early integration usually involves an interpolation step to synchronise visual features to the typically more frequent audio features. The main problem with early integration is the increase in dimensionality due to the concatenation of modalities, with the effect that more training data is required. The second problem is higher sensitivity to failure of a modality, although this is related to the training problem. For example, if the visual feature is unavailable because the talker looked away from the camera, then early integration

will fail unless it has been trained to cover this. In general, it hard to incorporate a measure of confidence for either modality when using early integration.

By contrast, late integration ignores all temporal synchrony between modalities and models them completely separately. Because each model is trained in a separate feature space, training is less of an issue than for early integration on the same task. Late integration is more robust to the failure of a modality—the failed modality can be simply ignored or its contribution weighted down. Independent modelling also allows different topologies to be used for audio and visual features, and they are free to operate at different rates. An additional question arises for late integration; how should the results from each classifier be combined? Kittler [95] found empirically that summing the probabilities from each classifier performed better than using either product, min, max, median or majority vote and was least sensitive to estimation error. The clear disadvantage of late integration is that correlation between the modalities cannot be directly exploited.

Some attempts have been made to combine the benefits of both early and late integration. An example using separate, but interconnected, audio and visual models is Boltzmann zippers [85]. For early integration, a cross-product HMM architecture has been successfully used to improve accuracy by allowing some temporal 'slip' between audio and visual components [30,31,33,186]. This method has the additional benefit of separating the initial training of the audio and visual features and combining them only for final training and recognition. A similar multi-stream HMM architecture has also proved successful for continuous audio-visual recognition [108].

A requirement of a successful audio-visual integration scheme is that the combined accuracy should be greater than or equal to either modality alone. If this is not the case it would be better to simply use only the better modality. Ideally the integrated performance should be greater than either audio- or visual-only. The psychological studies show that humans always report higher intelligibility when presented with bimodal speech [14, 66, 145, 148, 182].

8.2 Audio Recognition

For all integration experiments the acoustic features used were Mel frequency cepstral coefficients (MFCC's) calculated using the HTK hidden Markov model toolkit [194]. These features have been shown to be more effective than for example linear prediction coefficients or linear filterbank analysis [60]. The Mel scale models the nonlinear frequency response of the human ear by spacing filterbank channels on a nonlinear axis that is approximately linear up to 1000Hz and logarithmic above 1000Hz. This is defined in HTK using,

$$\mathrm{Mel}(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{8.1}$$

which is implemented using a Fourier transform by summing the magnitude from overlapping triangular filters spaced along the Mel scale.

Cepstral coefficients are calculated by taking the discrete cosine transform (DCT) of the log of the Mel scale filterbank magnitudes, m_i, using,

$$c_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} m_j \cos\left(\frac{\pi i}{N}(j - 0.5)\right) \tag{8.2}$$

where N is the number of filterbank channels. The raw filterbank magnitudes are often highly correlated and the DCT transformation is used to decorrelate them.

Little experimentation was done for the acoustic speech features. All acoustic features were the top 12 MFCC coefficients from a 16 filterbank analysis. In addition to these, log energy, delta MFCC and delta log energy coefficients were added, a total of 26 parameters.

Audio-only recognition results using a nine state HMM with a single diagonal covariance Gaussian mode for signal-to-noise ratios from clean to -10dB are shown in Figure 8.3. Due to variations during recording and differences in talker loudness the 'clean' audio SNR varies from 25 to 30dB depending on the talker. All other SNR's were obtained by adding the required amount of additive Gaussian white noise to the clean waveform.

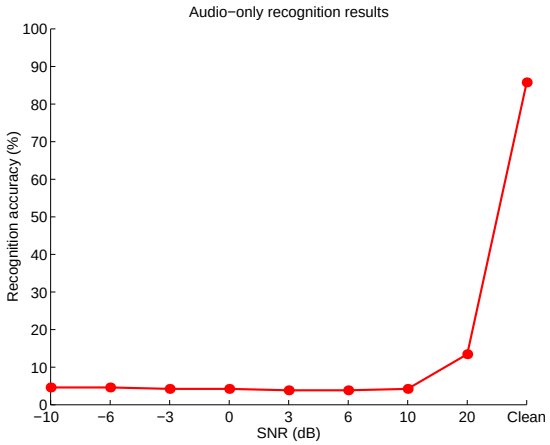


Figure 8.3: Audio-only recognition results for increasing noise power.

The models were trained on the clean speech and tested with speech parameterised from speech at all SNR's. The clean speech accuracy of 85.8% falls dramatically when noise is added.

8.3 Early Integration Results

For the early integration task composite features were formed by concatenating the temporally interpolated visual features obtained using either the best ASM tracking, AAM tracking or MSA features. As well as giving better accuracy for the visual-only recognition task, interpolation to 20ms means the visual features match the rate of the audio features.

Two conditions were tested. First, results were obtained by building separate models for each SNR—matched model conditions. These results indicate the best possible performance that can be obtained because they directly model the audio noise. For the second experiment, models were trained on the clean audio-visual data and tested at all SNR's. For all experiments, the HMM topology was varied using either 7 or 9 states with either 1 or 3 modes per state. The early integrated, matched model, audio-visual accuracies obtained using the best ASM tracker features are shown in Table 8.1.

Likewise, results using the full 37 dimensional AAM tracker features are shown in Table 8.2. To reduce the dimensionality of the combined AAM audio-visual space, a further experiment used only the top twenty AAM features. For the visual only task this was shown to hardly affect recognition accuracy, Table 7.27. When using early integrated audio-visual models the extra parameters may be unfairly biasing against this method because of the small amount of training data and much larger feature space. These results are shown in Figure 8.3.

Results using the best MSA features, (Table 7.11), are shown in Table 8.4. Finally

matched model results for audio only are given in Table 8.5 for comparison. The best results from this experiment all used nine HMM states with three modes per state for AAM and MSA, and one mode per state for ASM and audio. They are plotted for comparison in Figure 8.4.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	83.5	79.2	72.3	71.9	69.2	63.1	57.7	60.4	37.3
	3	81.2	-	-	-	-	-	63.9	61.2	50.0
9	1	85.0	81.9	75.0	71.9	65.4	62.7	62.7	59.6	45.0
	3	85.0	-	-	-	-	-	-	63.5	48.5

Table 8.1: Matched model early integration results using ASM visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	73.9	75.0	70.8	66.9	61.9	59.6	53.1	51.9	37.3
	3	67.3	71.5	68.5	67.3	64.2	61.2	58.1	56.9	50.8
9	1	73.9	71.9	70.4	67.7	66.9	63.5	60.8	60.4	43.1
	3	73.5	74.23	69.6	69.2	69.2	66.2	61.9	57.3	47.7

Table 8.2: Matched model early integration results using AAM visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	79.6	78.1	70.8	71.5	66.9	66.2	56.9	55.0	43.5
	3	75.8	77.7	71.9	71.5	66.9	66.5	63.1	63.5	53.5
9	1	80.0	78.1	73.1	73.5	68.5	62.7	65.8	60.8	44.2
	3	77.7	76.5	75.0	73.1	73.9	67.3	68.1	62.7	52.3

Table 8.3: Matched model early integration results using top twenty AAM visual features. Recognition accuracy in %.

Apart from the exception at high SNR for AAM features, all integrated audio-visual results are higher than the audio only results. By using matched models the audio-only results stay high even at very low SNR and only fall below the visual-only results at -10dB. At this SNR all of the audio-visual results are greater than either the audio- or visual-alone results, the requirement of a successful integration scheme. Unfortunately, this is not fully demonstrated in this case due to the high audio-only performance.

The results for the clean trained models are given for ASM visual features in Table 8.6 and for AAM, AAM top twenty, MSA and audio-only in Tables 8.7, 8.8, 8.9 and 8.10 respectively. The best results are plotted for comparison in Figure 8.5.

The trained-clean results are all above the audio-only accuracy, but only just. Only in the clean testing conditions do they satisfy the audio-visual greater than audio or visual requirement (apart from the AAM clean results). These results highlight the problem that combined audio-visual models are unable to cope when the audio part of the feature vectors is degraded by noise. They were not trained for these conditions, and the visual contribution cannot be separated, so is of little or no use. Similar results are reported in [29], and were improved by compensating for the audio noise prior to recognition. This is an attempt to get back to the clean conditions in which the models were trained by pre-processing the data.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	84.2	79.6	76.5	75.0	71.5	69.6	62.7	58.1	46.2
	3	84.2	83.9	78.9	76.2	71.2	66.5	70.0	64.2	50.4
9	1	86.2	85.4	80.8	76.9	70.8	68.5	68.1	64.2	54.2
	3	85.8	86.5	80.0	78.5	71.9	68.9	69.6	65.4	58.1

Table 8.4: Matched model early integration results using MSA visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	80.8	75.4	67.7	61.54	60.4	55.0	45.8	44.6	33.9
	3	-	-	-	-	-	-	-	-	35.0
9	1	85.8	78.5	70.8	61.5	59.6	55.8	51.2	49.2	35.0
	3	-	-	-	-	-	-	-	-	32.3

Table 8.5: Matched model audio results. Recognition accuracy in %.

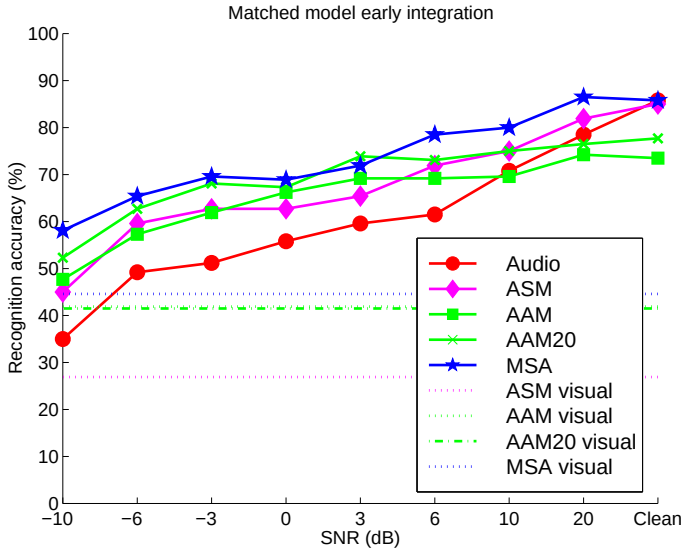


Figure 8.4: Best matched model early integration results for all methods.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	83.5	12.7	3.9	3.5	3.5	3.9	3.9	3.9	3.9
	3	81.2	28.5	10.0	7.3	6.2	5.0	5.4	4.2	4.6
9	1	85.0	19.6	6.2	5.4	3.5	3.9	3.9	3.9	3.9
	3	85.0	32.7	15.4	12.7	9.6	7.7	6.2	5.0	4.2

Table 8.6: Trained-clean early integration results using ASM visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	73.9	29.2	16.5	13.1	11.5	9.2	10.0	8.1	8.1
	3	67.3	36.2	20.4	20.0	16.9	15.4	12.7	12.7	11.9
9	1	73.9	26.2	17.3	13.5	11.9	11.5	10.0	7.7	7.3
	3	73.5	33.9	23.9	19.6	15.4	13.9	12.7	11.2	11.5

Table 8.7: Trained-clean early integration results using AAM visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	79.6	23.5	9.2	6.2	6.2	6.5	5.4	5.0	4.2
	3	75.8	26.2	19.6	16.2	13.1	12.7	10.8	8.1	6.9
9	1	80.0	26.9	11.9	8.5	6.9	6.9	5.8	4.6	5.0
	3	77.7	29.6	19.2	14.2	13.1	13.1	11.5	10.8	8.9

Table 8.8: Trained-clean early integration results using top twenty AAM visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	84.2	15.8	4.2	3.9	3.5	3.9	3.9	3.9	3.9
	3	84.2	30.4	13.9	12.7	9.6	8.1	6.2	5.8	6.5
9	1	86.2	18.5	5.4	5.0	3.9	3.9	3.9	3.9	3.9
	3	85.8	25.8	10.4	7.7	6.5	5.0	6.2	5.4	4.6

Table 8.9: Trained-clean early integration results using MSA visual features. Recognition accuracy in %.

States	Modes	Clean	20dB	10dB	6dB	3dB	0dB	-3dB	-6dB	-10dB
7	1	80.8	14.2	4.6	4.6	4.6	4.6	3.9	3.9	33.9
	3	-	-	-	-	-	-	-	-	-
9	1	85.8	13.5	4.2	3.9	3.9	4.2	4.2	4.6	4.6
	3	-	-	-	-	-	-	-	-	-

Table 8.10: Trained-clean audio results. Recognition accuracy in %.

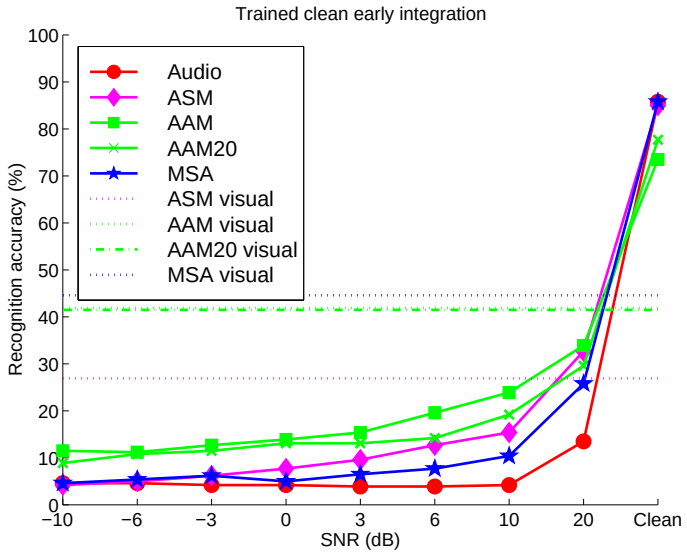


Figure 8.5: Best trained-clean early integration results for all methods.

This is a valid solution to the problem, but will ultimately fail when the noise compensation can no longer cope. Under these conditions a late integration scheme is able to focus on the modality providing the most reliable features.

8.4 Late Integration Results

This section presents results obtained by Dr Stephen Cox, first published as [56] and updated in [131, 132]

The late integration experiments presented in this section were obtained for the same task using the same visual features as the previous section. However, in order to use spectral subtraction noise compensation, the audio parameters are not MFCC's but the unlogged outputs of a 24 channel filterbank analysis. Although these could be converted to MFCC's, and better accuracy obtained, this was not done in these experiments. The audio-only results are therefore lower than those obtained using MFCC's previously.

From each, independent, audio and visual recogniser a set of likelihoods are obtained for each of the V words of the vocabulary. These are fused to give the integrated audio-visual result, w^*, using a weighted sum of log-likelihoods (equivalent to a product of the probabilities),

$$w^* = \underset{i=1,2,\ldots,V}{\arg\max} \{\alpha L(w_i|A) + (1-\alpha)L(w_i|V)\} \tag{8.3}$$

where $L(w_i|A)$ is the log-likelihood of the ith word from the audio recogniser and $L(w_i|V)$ likewise for the visual recogniser. The weight, α, allows the relative contribution from each recogniser to be varied.

The original paper [56] investigated audio-visual integration using an exhaustive search to find the best possible α for each utterance and for each SNR. These results were also compared with those using audio noise compensation with either spectral subtraction [19] or matched models. It was found that noise compensation could give better results than weighted audio-visual late integration using uncompensated audio. However, further experiments demonstrated that visual information could be used to further improve noise compensated audio features and that the integration weight α could be accurately estimated given

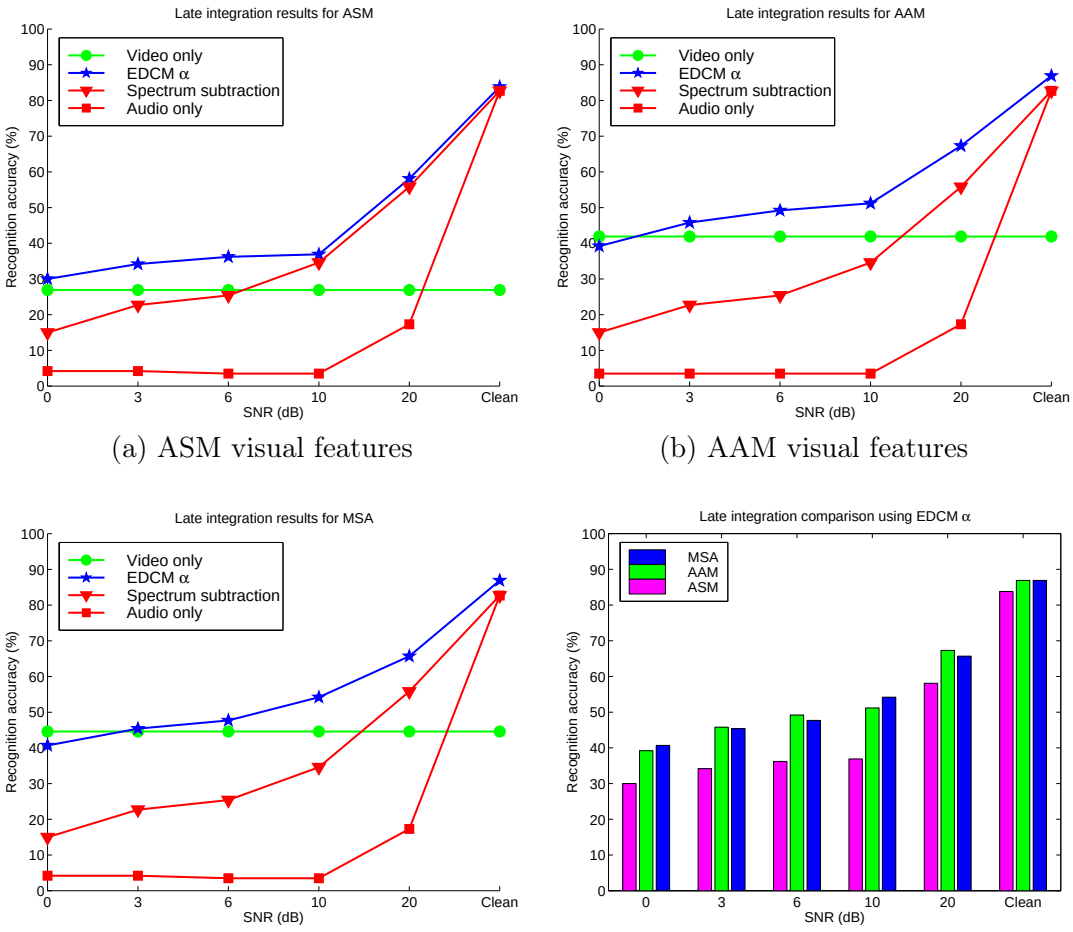
an estimate of the SNR. The estimate of the SNR could also be reliably predicted from the test utterances.

To choose α a confidence measure was formed based on the uncertainty of the audio recogniser about a word at each SNR. If the set of legal input words is denoted X and the recognised words Y the entropy derived confidence measure (EDCM) estimate for α is,

$$\alpha = 1 - \frac{H(X|Y)}{H(X|Y)_{max}} \tag{8.4}$$

The late integration results using EDCM to estimate α are given in Figure 8.6(a) using the best ASM result. The uncompensated audio rapidly falls to chance. This was improved using spectral subtraction noise compensation, but was further improved by adding the visual information. Similar results are obtained using the best AAM features, Figure 8.6(b), and MSA features, Figure 8.6(c). Only at the lowest SNR is the audio-visual greater than audio or visual requirement lost. A comparison of all three methods is shown in Figure 8.6(d). There are no results using the top twenty AAM features because there is no penalty in using more visual features for late integration.

As expected, given the similar visual-only scores of AAM and MSA, the integration results are also very similar. The ASM results are correspondingly lower.



(a) ASM visual features

(b) AAM visual features

(c) MSA visual features

(d) EDCM integration comparison

Figure 8.6: Late integration results for all methods. Uncompensated audio (red squares) rapidly approaches chance performance as noise levels increase. Spectral subtraction give improved accuracy (red triangles) but adding any of the visual feature types provides further improvement (blue stars). Visual-only accuracy is shown for comparison (green circles).

Chapter 9

Summary and Future Work

9.1 Summary

This thesis has described and evaluated six potential methods for visual speech analysis. Three (area tracking, height and width tracking and area-histograms) are dismissed on the basis of visual-only recognition experiments that show they are not effective in extracting useful lipreading information. The remaining three are: a low-level 1D multiscale spatial analysis (MSA) that measures the distribution of vertical length image features; and two high-level lip shape based analyses. The first measures lip shape using an inner and outer lip contour Active Shape Model (ASM). The second is a recent extension, implemented by Cootes [49], that statistically combines models of lip shape and greylevel appearance in an Active Appearance Model (AAM). The best visual-only results for all three methods are summarised in Table 9.1.

Method	ASM	AAM	MSA
AVletters	26.9	41.9	44.6
Tulips	76.7	N/A	77.1

Table 9.1: Visual-only results summary. Best accuracies, in %, obtained for the AVletters and Tulips database. The AAM tracker was not used on the Tulips database.

The visual-only recognition accuracies for the AVletters database show that using an ASM tracker to obtain only lip shape information is the poorest. The addition of greylevel appearance as well as shape modelling by the AAM greatly improves the results. Note that the ASM and AAM are trained on an identical set of landmark point labelled training images. However, while the models themselves are therefore directly comparable, the fitting algorithms are not. The errors in both of these model-based approached may be due to two effects. First, the tracking algorithms may have failed to find the global minimum of the cost function and so failed to find the best model fit to the image. Both the ASM simplex tracker and the AAM update model are only able to find local minima. If the real lip position, shape or appearance is too far from the starting position then the converged solution may not be the best. The second source of error is due to the modelling. If the training set did not cover all the variation in lip shape and appearance, then the statistical shape and appearance models do not accurately represent the data.

Improved results for both model-based trackers should be expected through the use of a temporal Kalman filter framework. This has proved successful in constraining and controlling

dynamic contours for many applications [16,17], including lipreading [93]. Active appearance models have recently been implemented in a similar framework for improved face identification from a series of images [65].

The best visual-only results were obtained using MSA. After an extensive search to find the best parameters it was found that magnitude sum scale histograms from a closing, c-sieve were the best. For c-sieves DC preservation is not an issue as only negative extrema are processed and order is not important. The best results used the top 20 PCA components calculated using the covariance matrix.

An important difference between MSA and both ASM and AAM is that MSA requires no training. The hand labelling of landmark points is prone to error and requires significant manual input to label a representative training set. MSA was able to achieve higher performance without any manual intervention on only crudely hand aligned mouth images. For the Tulips database Gray [73] was able to improve his low-level analysis results by re-aligning the database using the pose values from Luettin's ASM tracker [107]. While in practice MSA has been shown to be very robust to minor translations, any pixel-based analysis is ultimately dependent on the accurate location of the region of interest.

The clear observation is that better results should be obtainable by combining both high-level tracking and low-level analysis. The Active Appearance Model does this by combining shape and greylevel appearance. That the AAM recognition results are only slightly lower than those obtained with MSA must be treated with caution for such small training and test sets. On the basis of these results it cannot be conclusively claimed that MSA is better or worse than using AAM's. Neither are they mutually exclusive. If, on a more comprehensive task, it was found that MSA was better than the greylevel eigen-analysis used for appearance modelling then the AAM's can still be used as superior trackers, to more accurately locate the mouth than an ASM, for MSA analysis.

The results on the Tulips subset of digits database show that, for the simpler task, there is no real difference between MSA and ASM. When moving to the more complex AVletters database, ASM has worse performance than MSA. This is due to the problem of modelling error (a proportionally smaller amount of the database can be hand labelled) and different imaging conditions. The Tulips database has far clearer lip contours thanks to the additional desk level lamp during recording. The untrained, direct image analysis, used by MSA is better able to extend to the larger more difficult task.

In finding that shape is a poor lipreading feature compared to direct image analysis these results support the comments by Bregler and Omohundro [26] and oppose those of Kaucic, Dalton and Blake [93].

For all three of the best methods, early and late integration were compared. In matched model conditions early integration performed well, aided largely by the robust audio performance. Testing trained-clean early integration models highlighted the problems of this method. When one modality is degraded it degrades the recognition rate because the model has no way of dealing with this unknown situation. Better results were obtained using late integration with an entropy derived confidence measure (EDCM) to automatically change the relative weight between the audio and visual recognisers. For those experiments it was found that noise compensation could be used to obtain better results than using visual information with uncompensated audio. But, the best performance is obtained using all methods. In other words, one may always improve performance through appropriate signal processing, but this can be improved further by using additional visual information.

Finally, a significant part of this work was concerned with the acquisition of an aligned audio-visual database. The AVletters database was the result of this effort and has already been distributed to several non-UK researchers who wish to benchmark their systems (and

would rather avoid the trouble of recording their own). However, no claim can be made that it is a good database on which researchers can compare results. To make reliable comparisons of statistical methods a much larger database would be required. The major problem that has so far prevented this is distribution. Digitised acoustic speech databases typically require several CDROM's and large amounts of disk space during parameterisation. This problem is several orders of magnitude worse when visual data is added, and is currently impractical. The only viable current solution is real-time parameterisation of video data. This completely avoids the problem of having to store massive amounts of high bandwidth digital video.

There are further advantages of a real-time system that have been well proven in the vision and speech communities:

1. real-time implementations discourage slow algorithms;

2. real-time systems are less likely to train on unwanted features in the database since the data is less controlled;

3. real-time implementations encourage researchers to eliminate database dependent "tweaky parameters".

9.2 Real-time Implementations

To address the practicality of claiming that real-time visual analysis is required, fast implementations of the ASM tracker (only single resolution) and MSA analysis were written.

The real-time MSA system is capable of processing images continuously at 30fps (NTSC frame rate) on an R10000 175MHz SGI O2 workstation[1]. Scale histograms are constructed for each image frame, plotted on screen, Figure 9.2, and PCA transformed to find only the top components. An example demonstration of this system implemented a real-time, simple two-word 'yes' or 'no' vocabulary, recogniser for a single talker. This was demonstrated for five days continuously at the 'Tomorrow's World Live' event in March 1997 and performed well.

A major limitation of the real-time MSA analysis is that the talker had to position himself so an alignment box contains the mouth. To remove this limitation, a near real-time implementation of the ASM lip tracker was written. An example frame is shown in Figure 9.2. This is computationally limited by the large number of simplex evaluations required for each image frame and only runs at between 10-20fps depending on convergence tolerance and lighting conditions. The example screen shot shows the area around the model has been cutout and displayed in its own window. This is the proposed analysis window in which MSA can be applied. However, at the moment this system is unable to automatically initially locate the lips or recover if they are lost.

Preliminary experiments show that both systems will run at the same time, although the ASM tracker frame rate drops much more significantly as the processor time-slices between the processes. The computationally much simpler MSA is able to run at almost full rate.

This is a completely general implementation and has also been used for near real-time face tracking that could more reliably locate the lips and mouth area.

[1]This is certainly no longer state of the art or an excessive requirement.

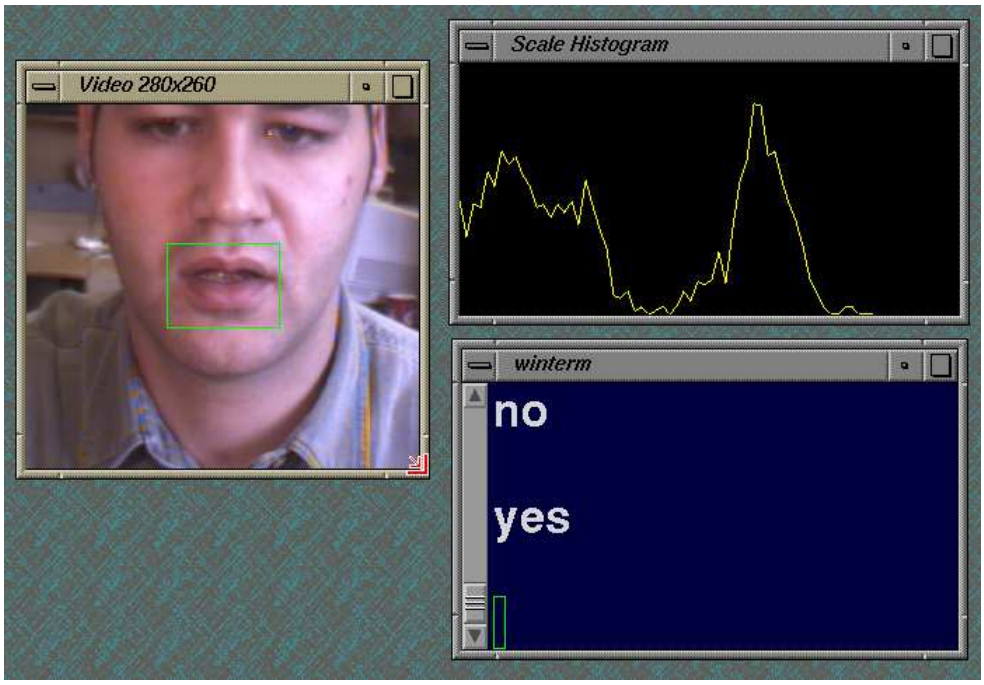


Figure 9.1: Real-time MSA analysis.

9.3 Future Work

A major extension to this work is required to move from isolated to continuous speech. Some progress in this direction has been made in the recording of a database of digit triples for two talkers. This database is also recorded at two different zoom positions to allow quantitative testing of a system for normalising image scale. This would be required to extend MSA to work on images other than at a single fixed size.

An important source of information so far ignored is colour. Several researchers have improved their lip trackers by highlighting the lips with colour processing. Significant improvements to the ASM tracker should be seen by modelling the colour of the lip contour boundary rather than just intensity.

Figure 9.2: Real-time ASM tracker.

Returning to the comments on the benefits of vision by Summerfield [183] (Section 2.1.4) it is clear that the identity of the talker, his location, whether or not he is talking and to whom, has so far been ignored. After the commitment is made to implement an audio-visual speech recognition system, this information is available for little extra effort. As well as providing more robust recognition, vision can disambiguate situations where there are multiple talkers or where the talker turns to address someone else, for example. It is perhaps in these real-world situations, that cannot be resolved by audition alone, where visual speech analysis will have the most to offer.

Appendix A

Principal Component Analysis

Principal component analysis (PCA) [43, 89, 99] is a technique for analysing multivariate data that identifies a set of new orthogonal variables known as principal components. These principal components are linear combinations of the original variables and represent a rotation of the n dimensional axes of the data. The directions are ordered by decreasing importance so the first principal component is the axis that describes most variance, the second component is the orthogonal axis that describes as much as possible of the remaining variance and so on. This transforms correlated data into a decorrlelated space that can be approximated using r ($\leq n$) dimensions. If the data is highly correlated r can be much less than n and yet still describe most of the variance. Principal component analysis is also called the Hotelling transform [87] or Karhunen-Loève expansion [2].

If $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ is an n-dimensional random variable with mean $\overline{\mathbf{x}}$ and covariance $\mathbf{S}$ from N samples,

$$\overline{x}_k = \frac{1}{N}\sum_{i=1}^{N} x_{ki} \qquad k = 1, \ldots, n \tag{A.1}$$

$$\overline{\mathbf{x}} = [\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n]^T \tag{A.2}$$

$$\mathbf{S} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T \tag{A.3}$$

the task is to find the new set of variables, $b_1, b_2, \ldots, b_n$, which are uncorrelated and have variances that decrease from first to last. Each is a linear combination of the original variables so,

$$b_j = p_{1j}x_1 + p_{2j}x_2 + \ldots + p_{nj}x_n \tag{A.4}$$

$$= \mathbf{p}_j^T\mathbf{x} \tag{A.5}$$

where $\mathbf{p}_j = [p_{1j}, p_{2j}, \ldots, p_{nj}]^T$ is a vector of constants which is constrained to be a unit length vector so the transform is orthogonal, i.e. each $\mathbf{p}_j$ represents a new axis and distances are preserved,

$$\mathbf{p}_j^T\mathbf{p}_j = \sum_{k=1}^{n} p_{kj}^2 = 1 \tag{A.6}$$

The first principal component is defined as the linear combination $b_1 = \mathbf{p}_1^T\mathbf{x}$ of the original variables that has largest possible variance given the constraint of Equation (A.6).

The sample variance of b_1 is,

$$S_{b_1}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(b_{1_i} - \overline{b}_1)^2 \tag{A.7}$$

But, $b_{1_i} - \overline{b}_1 = \mathbf{p}_1^T\mathbf{x}_i - \mathbf{p}_1^T\overline{\mathbf{x}} = \mathbf{p}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})$ from Equation (A.5) so,

$$(b_{1_i} - \overline{b}_1)^2 = \mathbf{p}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T\mathbf{p}_1 \tag{A.8}$$

and,

$$\sum_{i=1}^{N}(b_{1_i} - \overline{b}_1)^2 = \sum_{i=1}^{N}\mathbf{p}_1^T(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T\mathbf{p}_1 \tag{A.9}$$

$$= \mathbf{p}_1^T\left[\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T\right]\mathbf{p}_1 \tag{A.10}$$

since $\mathbf{p}_1$ is constant over i. Substituting from Equation (A.3), Equation (A.7) becomes,

$$S_{b_1}^2 = \mathbf{p}_1^T\left[\frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^T\right]\mathbf{p}_1 = \mathbf{p}_1^T\mathbf{S}\mathbf{p}_1 \tag{A.11}$$

The standard method for maximising a function of several variables with one or more constraints is to use *Lagrange* (undetermined) multipliers. With a single constraint the stationary points of a differentiable function of n variables $f(x_1, x_2, \ldots, x_n)$ subject to the constraint $g(x_1, x_2, \ldots, x_n) = c$ are such that there exists a Lagrange multiplier, λ, so that,

$$\frac{\partial f}{\partial x_i} - \lambda\frac{\partial g}{\partial x_i} = 0 \qquad i = 1, \ldots, n \tag{A.12}$$

at the stationary points. Given the Lagrangian function,

$$L(\mathbf{x}) = f(\mathbf{x}) - \lambda[g(\mathbf{x}) - c] \tag{A.13}$$

the set of equations in Equation (A.12) can be written,

$$\frac{\partial L}{\partial \mathbf{x}} = \mathbf{0} \tag{A.14}$$

Applying this to the maximisation of Equation (A.11),

$$L(\mathbf{p}_1) = \mathbf{p}_1^T\mathbf{S}\mathbf{p}_1 - \lambda(\mathbf{p}_1^T\mathbf{p}_1 - 1) \tag{A.15}$$

and differentiating,

$$\frac{\partial L}{\partial \mathbf{p}_1} = 2\mathbf{S}\mathbf{p}_1 - 2\lambda\mathbf{p}_1 \tag{A.16}$$

Setting to zeros gives,

$$(\mathbf{S} - \lambda\mathbf{I})\mathbf{p}_1 = \mathbf{0} \tag{A.17}$$

$$\text{or} \quad \mathbf{S}\mathbf{p}_1 = \lambda\mathbf{p}_1 \tag{A.18}$$

This is a homogeneous set of n equations with n unknowns. For a solution to $\mathbf{p}_1$ other than the null vector $(\mathbf{S} - \lambda\mathbf{I})$ must be singular so λ is chosen such that,

$$|\mathbf{S} - \lambda\mathbf{I}| = 0 \tag{A.19}$$

A non-zero solution only exists if λ is an eigenvalue of $\mathbf{S}$, but in general $\mathbf{S}$ will have n eigenvalues, $\lambda_1, \lambda_2, \ldots, \lambda_n$. To choose the required eigenvalue note from Equation (A.18),

$$\mathbf{p}_1^T\mathbf{S}\mathbf{p}_1 = \lambda\mathbf{p}_1^T\mathbf{p}_1 \tag{A.20}$$

and since $\mathbf{p}_j^T\mathbf{p}_j = 1$ then $\mathbf{p}_1^T\mathbf{S}\mathbf{p}_1 = \lambda$. As this is what needs to be maximised λ is chosen as the *largest* eigenvalue, λ_1. From Equation (A.18) the principal component $\mathbf{p}_1$ is the eigenvector corresponding to the largest eigenvalue.

The second principal component, $b_2 = \mathbf{p}_2^T\mathbf{x}$, is obtained in a similar way. However, there is the additional constraint that b_2 must be uncorrelated with b_1, that is,

$$\mathbf{p}_2^T\mathbf{p}_1 = \mathbf{p}_1^T\mathbf{p}_2 = \sum_{i=1}^{n} p_{1i}p_{2i} = 0 \tag{A.21}$$

i.e. $\mathbf{p}_2$ and $\mathbf{p}_1$ must be orthogonal.

To maximise the variance of b_2, which from Equation (A.11) is,

$$S_{b_2}^2 = \mathbf{p}_2^T\mathbf{S}\mathbf{p}_2 \tag{A.22}$$

for both constraints requires two Lagrange multipliers,

$$L(\mathbf{p}_2) = \mathbf{p}_2^T\mathbf{S}\mathbf{p}_2 - \lambda(\mathbf{p}_2^T\mathbf{p}_2 - 1) - \delta\mathbf{p}_2^T\mathbf{p}_1 \tag{A.23}$$

where δ is the second Lagrange multiplier.

At the stationary points,

$$\frac{\partial L}{\partial \mathbf{p}_2} = 2(\mathbf{S} - \lambda\mathbf{I})\mathbf{p}_2 - \delta\mathbf{p}_1 = \mathbf{0} \tag{A.24}$$

by pre-multiplying with $\mathbf{p}_1^T$,

$$2\mathbf{p}_1^T\mathbf{S}\mathbf{p}_2 - \delta = 0 \tag{A.25}$$

since $\mathbf{p}_2^T\mathbf{p}_1 = 0$. This also means $\mathbf{p}_2^T\mathbf{S}\mathbf{p}_1 = \mathbf{p}_1^T\mathbf{S}\mathbf{p}_2 = 0$ since $\mathbf{S}$ is symmetric and $\mathbf{p}_2^T\mathbf{S}\mathbf{p}_1$ is a scalar so δ is zero at the stationary points and Equation (A.24) becomes,

$$(\mathbf{S} - \lambda\mathbf{I})\mathbf{p}_2 = \mathbf{0} \tag{A.26}$$

This is analogous to the the first principal component but with the added constraint that the second principal component must have the most variance *after* the the variance of the first component has been accounted for. The second principal component is thus the eigenvalue associated with the second largest eigenvalue, λ_2.

This can be continued for all the remaining principal components. The jth component is the linear combination $b_j = \mathbf{p}_j^T\mathbf{x}$ which is orthogonal to all previous combinations,

$$\mathbf{p}_j^T\mathbf{p}_i = 0 \qquad \forall\ i < j \tag{A.27}$$

with maximum variance. Maximising this variance involves an expression with j Lagrange

multipliers. Generalising from the description above the eigenvector associated with the jth largest eigenvalue is the jth principal component.

The $n \times n$ eigenvectors can be represented using the matrix,

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n] \tag{A.28}$$

and the $n \times 1$ vector of principal components by $\mathbf{b} = [b_1, b_2, \ldots, b_n]^T$, so,

$$\mathbf{b} = \mathbf{P}^T \mathbf{x} \tag{A.29}$$

The geometric (rotation of the axes) interpretation of PCA considers the lines n dimensional sample space which define the hyper-plane onto which sample members are projected. In this case the coefficients, $\mathbf{p}_i$ of the ith principal component b_i may also be referred to as the *ith* principal component. In this sense $\mathbf{P}$, the matrix of eigenvectors, defines the directions or *modes of variation* found by the principal component analysis. This interpretation is how PCA is generally used in this thesis.

A.1 Variance Contribution

The $n \times n$ covariance matrix of the orthogonal, ranked by variance, principal components $\mathbf{b}$ is, by the definition of principal components,

$$\mathbf{S_b} = \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & & \ddots & \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix} \tag{A.30}$$

The proportion of the variance described by the ith component is,

$$\frac{\lambda_i}{\sum_{j=1}^{n} \lambda_j} \tag{A.31}$$

and the proportion of the variance described by the first t components,

$$\frac{\sum_{j=1}^{t} \lambda_j}{\sum_{j=1}^{n} \lambda_j} \tag{A.32}$$

This can be used for dimensionality reduction by the allowing the percentage of variance to be retained in the smaller t dimensional space to be chosen, say 95%.

A.2 Mean Correction

Equation (A.29) relates the sample $\mathbf{x}$ to the principal components $\mathbf{b}$ such that $\mathbf{b}$ has zero mean (it is removed when calculating the covariance). The means can be corrected with the

appropriate vector of means, Equation (A.3),

$$\mathbf{b} = \mathbf{P}^T(\mathbf{x} - \overline{\mathbf{x}}) \tag{A.33}$$

which geometrically is a translation by the mean to the origin followed by an orthogonal rotation of the axes.

A.3 Inverse Transform

Principal component analysis is a linear transform and $\mathbf{P}$ is, by definition, orthogonal so $\mathbf{P}^T\mathbf{P} = \mathbf{I}$ and Equation (A.33) becomes,

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{P}\mathbf{b} \tag{A.34}$$

A.4 Scaling Problem

From the definition of principal components the first component is the linear combination that describes the most variance. This can be a problem if the sample variables have different scaling or units. For example one variable might be measured in millimetres and another in kilograms. The variables that exhibit greatest variance will dominate the principal component analysis, and nothing is gained as any correlation between variables is lost in the large scale difference. A solution is to standardise the data prior to analysis, by subtracting the mean and dividing by the standard deviation of each variable in turn. This normalises the covariance matrix and is the correlation matrix, whose correlation coefficients are,

$$r_{ij} = \frac{1}{N-1}\sum_{r=1}^{N}\frac{(x_{ri} - \overline{x}_i)(x_{ri} - \overline{x}_i)}{s_i s_j} \tag{A.35}$$

where s_i is the variance of variable i,

$$s_i^2 = \frac{1}{N-1}\sum_{r=1}^{N}(x_{rj} - \overline{x}_j)^2 \tag{A.36}$$

The correlation matrix, $\mathbf{R}$ is then,

$$\mathbf{R} = \begin{bmatrix} 1 & r_{12} & \dots & r_{1n} \\ r_{21} & 1 & \dots & r_{2n} \\ \vdots & & \ddots & \\ r_{n1} & r_{n2} & \dots & 1 \end{bmatrix} \tag{A.37}$$

This differs from the covariance matrix by the variance normalisation. The elements of $\mathbf{S}$ are calculated using,

$$s_{ij} = \frac{1}{N-1}\sum_{r=1}^{N}(x_{ri} - \overline{x}_i)(x_{ri} - \overline{x}_i) \tag{A.38}$$

where,

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1n} \\ s_{21} & s_{22} & \dots & s_{2n} \\ \vdots & & \ddots & \\ s_{n1} & s_{n2} & \dots & s_{nn} \end{bmatrix} \tag{A.39}$$

By using the correlation matrix $\mathbf{R}$ instead of the covariance matrix $\mathbf{S}$ for principal component analysis the sample space is normalised, so that each variable is considered equally important. If the variables are not thought to be equally significant the correlation matrix should not be used.

A.5 MATLAB Implementation

The following MATLAB was used to calculate the point distribution models (PDM's) in Section 5.1.1 and uses the same notation as above. Only standard MATLAB (version 5.2 is current) commands are required. The principal components are calculated using the eigenanalysis of the covariance matrix. The remaining commands ensure the eigenvalues and corresponding eigenvectors are in variance order and identify how many are required to account for `varSum` of the variance. When calculating PDM's `varSum` was set 0.95.

```
% Do stats
Xm = mean(models);
S = cov(models);

% Principal component analysis
[V D] = eig(S);
[evals I] = sort(diag(D));
evecs = V(:,I);

% Max first
evals = flipud(evals);
evecs = fliplr(evecs);

% Keep how many?
vsum = cumsum(evals) / sum(evals);
t = min(find(vsum >= varSum));

% Return only first t modes
P = evecs(:,1:t);
b = evals(1:t);
Xm = Xm';
```

Appendix B

MSA Scale-Histogram Results

B.1 AVletters Database

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	21.15	16.92	16.92	19.62	13.85	14.62	21.54	16.92	21.92	15.38	21.92	15.38
10Ns5m3	37.69	31.15	30.00	30.00	25.77	20.38	28.46	26.15	34.23	31.15	34.23	31.15
10Ns7m1	28.08	16.15	20.38	22.69	20.38	14.62	23.08	18.08	26.54	22.31	26.54	22.31
10Ns7m3	37.69	35.00	31.92	30.38	30.77	24.23	31.54	26.15	34.23	33.08	34.23	33.08
10Ns9m1	27.69	22.69	26.92	22.31	19.23	15.38	22.31	19.62	28.08	22.69	28.08	22.69
10Ns9m3	36.54	35.00	33.85	34.23	32.31	26.54	30.77	28.85	38.85	35.38	38.85	35.38
10NIs5m	21.92	18.46	16.54	19.23	15.77	16.54	21.92	18.85	17.31	17.31	17.31	17.31
10NIs5m3	32.69	33.46	30.38	31.15	26.15	22.31	28.08	24.62	34.23	32.69	34.23	32.69
10NIs7m1	23.85	20.38	20.00	21.92	22.69	17.31	25.77	20.00	22.31	20.77	22.31	20.77
10NIs7m3	38.46	35.00	36.15	31.92	29.62	24.62	33.08	28.85	34.23	35.00	34.23	35.00
10NIs9m1	24.62	25.77	23.85	25.38	20.77	15.38	25.77	22.31	26.15	21.54	26.15	21.54
10NIs9m3	35.77	37.69	35.38	33.85	31.15	28.46	30.77	NaN	36.15	35.00	36.15	35.00
20Ns5m1	18.46	15.77	21.92	16.92	22.69	15.38	21.15	19.62	24.62	20.38	24.62	20.38
20Ns5m3	36.15	28.08	35.38	29.23	28.08	25.38	27.69	25.77	34.62	32.31	34.62	32.31
20Ns7m1	25.77	20.77	26.15	20.77	23.08	15.00	19.23	23.46	28.46	20.77	28.46	20.77
20Ns7m3	41.15	31.15	33.08	34.23	31.15	25.77	31.15	28.08	34.62	36.92	34.62	36.92
20Ns9m1	28.46	24.23	26.15	24.62	28.08	19.62	23.46	26.54	30.77	20.38	30.77	20.38
20Ns9m3	40.77	34.62	35.00	34.62	28.85	24.62	31.54	33.08	39.23	41.92	39.23	41.92
20NIs5m1	21.92	16.92	19.62	16.15	20.38	14.62	21.54	20.38	26.15	20.77	26.15	20.77
20NIs5m3	38.08	31.54	36.15	31.54	25.77	21.54	30.38	28.85	36.15	31.54	36.15	31.54
20NIs7m1	27.31	19.62	23.85	21.92	21.54	17.69	19.23	21.15	24.23	24.62	24.23	24.62
20NIs7m3	40.77	35.00	37.69	34.23	31.15	27.69	30.77	30.77	36.54	37.69	36.54	37.69
20NIs9m1	27.31	21.15	26.15	NaN	25.00	18.46	25.00	21.92	30.77	24.23	30.77	24.23
20NIs9m3	38.85	34.62	35.77	33.85	31.15	30.00	33.46	33.85	37.69	38.46	37.69	NaN

Table B.1: AVletters, *sh*

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	17.69	17.31	19.62	12.31	20.00	11.92	16.54	15.00	20.77	14.62	20.77	14.62
10Ns5m3	30.00	23.08	25.77	18.85	29.23	28.46	25.38	25.38	33.46	32.69	33.46	32.69
10Ns7m1	21.54	17.69	28.85	16.15	23.85	13.85	19.23	18.46	23.85	20.00	23.85	20.00
10Ns7m3	28.08	27.69	33.46	21.54	35.00	30.38	28.46	27.69	35.77	30.00	35.77	30.00
10Ns9m1	22.69	13.08	26.15	16.54	29.62	11.92	22.69	21.54	28.08	21.54	28.08	21.54
10Ns9m3	32.69	29.23	35.00	25.77	36.54	28.85	29.62	29.23	37.31	34.62	37.31	34.62
10Is5m1	20.00	12.69	18.46	13.08	19.62	13.46	19.23	16.15	16.54	16.54	16.54	16.54
10Is5m3	28.46	22.69	26.92	20.38	33.46	28.08	25.00	28.08	30.38	27.69	30.38	27.69
10Is7m1	19.23	17.69	22.69	16.15	26.15	16.54	19.23	21.92	25.38	20.77	25.38	20.77
10Is7m3	28.85	26.54	28.85	23.08	34.62	28.46	27.31	30.38	37.69	33.08	37.69	33.08
10Is9m1	24.62	16.92	26.54	16.92	29.23	14.62	24.62	20.77	30.00	23.08	30.00	23.08
10Is9m3	31.54	29.23	31.92	28.08	38.85	26.54	30.77	29.23	37.31	35.38	37.31	35.38
20Ns5m1	21.92	16.54	20.38	14.23	20.38	15.38	28.85	16.15	22.31	14.62	22.31	14.62
20Ns5m3	33.08	NaN	30.00	30.00	31.15	31.15	31.92	24.23	40.00	28.85	40.00	28.85
20Ns7m1	20.38	23.85	23.85	15.00	20.77	18.08	29.62	21.54	26.92	24.23	26.92	24.23
20Ns7m3	33.08	31.54	27.69	30.77	33.85	31.54	33.46	29.62	36.15	34.62	36.15	34.62
20Ns9m1	25.77	25.77	25.38	20.00	24.62	22.31	31.15	26.15	30.00	25.00	30.00	25.00
20Ns9m3	35.77	31.15	32.31	29.23	33.85	30.00	36.54	35.00	41.15	35.77	41.15	35.77
20Is5m1	18.85	17.31	18.08	11.54	18.85	16.54	23.46	17.69	24.62	15.38	24.62	15.38
20Is5m3	31.92	30.77	32.31	26.92	29.62	27.69	31.92	28.46	36.54	31.54	36.54	31.54
20Is7m1	24.23	22.69	25.38	16.15	25.77	20.00	30.00	22.31	27.31	22.31	27.31	22.31
20Is7m3	30.77	34.23	31.92	32.31	33.85	32.69	31.54	28.85	36.54	35.38	36.54	35.38
20Is9m1	26.54	24.23	24.23	23.46	27.69	21.92	30.00	21.92	32.69	25.38	32.69	25.38
20Is9m3	33.46	36.54	33.08	28.85	32.31	31.92	30.00	33.85	44.62	38.08	44.62	38.08

Table B.2: AVletters, *a*

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	21.54	15.38	16.54	14.62	20.00	11.92	16.54	15.00	20.77	14.62	20.77	14.62
10Ns5m3	35.77	29.23	29.62	27.31	29.23	28.46	25.38	25.38	33.46	32.69	33.46	32.69
10Ns7m1	22.69	18.46	22.69	18.46	23.85	13.85	19.23	18.46	23.85	20.00	23.85	20.00
10Ns7m3	35.77	31.15	29.23	29.23	35.00	30.38	28.46	27.69	36.15	30.00	36.15	30.00
10Ns9m1	24.23	21.15	21.54	17.31	29.62	11.92	22.69	21.54	28.08	21.54	28.08	21.54
10Ns9m3	35.77	30.38	29.62	33.85	36.54	28.85	29.62	29.23	37.31	34.62	37.31	34.62
10NIs5m	18.46	15.38	16.92	13.46	19.62	13.46	19.23	16.15	16.54	16.54	16.54	16.54
10NIs5m3	32.69	30.38	28.46	26.92	33.46	28.08	25.00	28.08	30.77	27.69	30.77	27.69
10NIs7m1	21.15	18.85	23.08	18.08	26.15	16.54	19.23	21.92	25.38	20.77	25.38	20.77
10NIs7m3	33.85	30.38	31.15	29.23	34.62	28.46	27.31	30.38	37.69	33.08	37.69	NaN
10NIs9m1	25.00	20.38	23.46	22.69	29.23	14.62	24.62	20.77	30.00	23.08	30.00	23.08
10NIs9m3	37.69	33.08	34.23	32.31	38.85	26.54	30.77	29.23	37.31	35.38	37.31	35.38
20Ns5m1	20.77	16.15	19.23	15.77	20.38	15.38	28.85	16.15	22.31	14.62	22.31	14.62
20Ns5m3	35.77	28.08	25.38	26.15	31.15	31.15	31.92	24.23	40.00	28.46	40.00	28.46
20Ns7m1	25.77	18.46	20.00	20.38	20.77	18.08	29.62	21.54	26.92	24.23	26.92	24.23
20Ns7m3	38.85	30.00	26.15	30.38	33.85	31.54	33.46	29.62	36.15	34.62	36.15	34.62
20Ns9m1	28.08	17.69	23.85	18.46	24.62	22.31	31.15	26.15	30.00	25.00	30.00	25.00
20Ns9m3	39.62	31.15	31.54	31.92	33.85	30.00	36.54	35.00	41.54	35.77	41.54	35.77
20NIs5m1	19.62	15.00	15.77	13.85	18.85	16.54	23.46	17.69	24.62	15.38	24.62	15.38
20NIs5m3	36.15	28.08	25.00	23.46	29.62	27.69	31.92	28.46	36.15	31.54	36.15	31.54
20NIs7m1	28.08	20.00	20.00	19.62	25.77	20.00	30.00	22.31	27.31	22.31	27.31	22.31
20NIs7m3	36.92	32.69	27.31	30.38	33.85	32.69	31.54	28.85	36.54	35.77	36.54	35.77
20NIs9m1	30.00	20.00	23.85	20.38	27.69	21.92	30.00	21.92	32.69	25.38	32.69	25.38
20NIs9m3	40.77	35.38	29.23	28.85	32.31	31.92	30.00	33.85	44.62	38.08	44.62	38.08

Table B.3: AVletters, $|a|$

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	16.92	13.85	16.15	3.85	16.92	13.08	16.54	16.92	19.23	14.23	19.23	14.23
10Ns5m3	25.00	28.85	24.62	9.23	28.46	24.62	23.46	23.85	27.69	28.08	27.69	28.08
10Ns7m1	20.00	16.54	16.92	5.00	17.31	13.08	18.46	16.92	19.62	17.31	19.62	17.31
10Ns7m3	34.23	31.15	28.08	10.38	27.31	28.85	23.08	23.85	36.15	32.69	36.15	32.69
10Ns9m1	20.77	20.77	18.08	3.46	20.00	16.92	18.85	19.23	22.31	22.31	22.31	22.31
10Ns9m3	31.15	28.08	28.08	10.38	27.69	30.00	23.85	29.62	33.46	30.38	33.46	30.38
10NIs5m	16.54	12.31	13.85	5.38	16.92	13.46	15.38	12.69	20.38	16.15	20.38	16.15
10NIs5m3	27.69	26.92	24.62	10.00	25.77	22.69	22.69	23.46	NaN	28.46	NaN	28.46
10NIs7m1	20.77	16.54	16.54	5.77	18.46	17.31	20.00	16.15	21.54	20.77	21.54	20.77
10NIs7m3	38.08	32.31	29.62	11.54	28.08	24.62	23.85	21.54	31.54	33.08	31.54	33.08
10NIs9m1	22.31	18.46	18.85	5.00	22.69	18.85	18.08	18.46	24.23	22.69	24.23	22.69
10NIs9m3	31.92	35.38	27.69	13.85	31.15	31.15	26.15	27.31	33.46	33.85	33.46	33.85
20Ns5m1	19.62	14.62	15.38	4.62	15.77	18.46	19.62	14.23	13.08	21.54	13.08	21.54
20Ns5m3	29.62	29.62	26.15	10.77	28.46	26.54	26.15	22.69	31.92	29.23	31.92	29.23
20Ns7m1	21.54	20.77	17.69	4.23	20.77	17.69	23.46	16.15	21.15	21.54	21.15	21.54
20Ns7m3	30.38	31.15	23.46	11.54	27.31	29.23	25.77	26.54	34.62	33.85	34.62	33.85
20Ns9m1	21.15	21.15	20.00	5.00	17.69	20.38	25.00	15.77	27.69	25.38	27.69	25.38
20Ns9m3	30.77	34.23	24.62	10.00	29.23	30.00	28.46	26.15	28.46	33.85	28.46	33.85
20NIs5m1	18.08	16.54	14.62	6.15	16.92	15.77	19.62	14.23	17.69	19.62	17.69	19.62
20NIs5m3	31.92	33.08	22.69	13.85	26.54	23.08	24.62	24.23	34.23	31.15	34.23	31.15
20NIs7m1	20.00	18.08	15.38	5.77	20.77	19.62	19.62	15.77	23.08	23.85	23.08	23.85
20NIs7m3	32.69	34.23	26.15	9.62	29.62	33.08	27.69	24.23	34.62	35.77	34.62	35.77
20NIs9m1	23.46	20.00	21.54	6.92	19.62	19.62	26.92	20.77	21.54	26.54	21.54	26.54
20NIs9m3	31.54	35.77	27.69	14.62	28.46	31.92	27.31	24.62	37.31	38.85	37.31	38.85

Table B.4: AVletters, a^2

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	20.38	16.92	17.31	16.92	18.85	8.46	18.85	11.54	14.62	15.00	14.62	15.00
10Ns5m3	28.46	29.23	23.08	27.31	23.08	18.08	24.23	23.85	25.38	25.00	25.38	25.00
10Ns7m1	24.62	22.31	16.15	20.38	19.23	11.15	20.00	16.15	21.15	15.38	21.15	15.38
10Ns7m3	31.92	33.08	26.15	27.69	25.77	18.85	26.15	23.85	30.77	28.46	30.77	28.46
10Ns9m1	24.23	25.00	19.62	21.15	18.08	12.31	21.54	18.46	22.31	20.38	22.31	20.38
10Ns9m3	32.31	31.92	27.69	31.15	24.62	21.92	31.92	25.38	32.31	32.31	32.31	32.31
10NIs5m	16.92	18.08	13.46	17.69	20.77	11.54	18.46	14.62	16.15	17.69	16.15	17.69
10NIs5m3	25.77	26.54	28.08	28.08	22.31	16.15	27.69	23.46	30.77	27.69	30.77	27.69
10NIs7m1	19.62	23.85	16.54	18.46	18.46	13.08	23.46	17.31	23.85	17.31	23.85	17.31
10NIs7m3	30.77	31.54	25.00	33.08	24.62	22.31	25.77	NaN	31.15	29.23	31.15	29.23
10NIs9m1	24.62	24.62	19.23	19.62	17.69	12.31	25.38	16.92	23.85	18.85	23.85	18.85
10NIs9m3	31.54	34.62	31.92	33.08	27.31	21.54	29.62	23.08	36.15	33.08	36.15	33.08
20Ns5m1	21.15	24.23	18.46	15.77	21.54	12.31	21.54	16.54	18.46	18.46	18.46	18.46
20Ns5m3	31.54	32.31	23.46	25.38	25.77	23.46	27.69	25.38	32.69	32.31	32.69	32.31
20Ns7m1	23.46	23.08	22.69	16.92	24.62	14.62	21.54	19.62	26.15	21.15	26.15	21.15
20Ns7m3	31.92	35.38	30.00	26.54	27.69	22.69	28.85	27.69	35.38	33.46	35.38	33.46
20Ns9m1	23.85	26.92	25.00	20.00	23.46	12.31	22.69	21.92	26.54	28.46	26.54	28.46
20Ns9m3	34.62	33.85	32.69	27.69	28.46	24.23	30.38	26.15	37.69	33.85	37.69	33.85
20NIs5m1	21.15	20.77	16.92	15.77	20.38	11.15	20.00	11.54	20.38	16.15	20.38	16.15
20NIs5m3	32.69	34.23	27.31	24.62	24.23	21.54	28.46	23.85	32.31	33.85	32.31	33.85
20NIs7m1	24.62	23.85	23.46	20.38	19.62	15.38	21.92	20.77	25.77	21.15	25.77	21.15
20NIs7m3	34.23	33.08	31.54	28.46	25.38	23.46	27.31	26.54	35.38	35.00	35.38	35.00
20NIs9m1	26.54	28.08	21.92	19.62	19.62	12.31	25.00	22.69	26.54	24.23	26.54	24.23
20NIs9m3	37.69	33.85	29.62	31.15	26.54	23.46	31.54	26.15	39.23	39.23	39.23	39.23

Table B.5: AVletters, *sh*, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	15.00	13.46	10.77	7.31	19.62	14.23	16.92	13.85	18.46	16.54	18.46	16.54
10Ns5m3	24.62	24.62	16.92	12.69	28.08	23.46	28.08	26.15	28.85	28.85	28.85	28.85
10Ns7m1	16.54	16.15	11.54	8.85	19.23	16.15	18.85	18.46	21.92	17.69	21.92	17.69
10Ns7m3	26.92	26.15	19.23	15.77	27.69	23.46	24.62	28.46	36.15	31.54	36.15	31.54
10Ns9m1	21.15	17.31	13.46	8.08	21.54	13.85	20.77	18.46	26.15	25.00	26.15	25.00
10Ns9m3	31.15	24.62	19.62	15.77	28.08	27.31	25.77	30.38	39.23	33.85	39.23	33.85
10NIs5m	16.54	15.77	8.08	8.46	20.38	16.15	18.46	15.38	16.15	17.31	16.15	17.31
10NIs5m3	25.00	25.38	15.00	16.15	25.38	23.85	25.00	26.15	30.38	31.92	30.38	31.92
10NIs7m1	17.69	17.31	8.08	8.46	20.38	16.92	20.77	14.62	19.62	21.54	19.62	21.54
10NIs7m3	30.77	26.92	18.46	15.00	30.00	23.85	28.08	27.31	36.15	30.38	36.15	30.38
10NIs9m1	19.62	18.46	14.23	7.69	20.77	16.92	23.85	16.15	22.31	22.69	22.31	22.69
10NIs9m3	30.38	27.69	16.15	19.62	24.23	26.92	27.31	NaN	38.08	36.54	38.08	36.54
20Ns5m1	18.46	16.54	15.00	8.46	20.38	14.62	22.31	13.46	27.69	21.54	27.69	21.54
20Ns5m3	27.69	26.15	19.62	13.46	26.92	25.77	30.00	27.31	33.08	30.00	33.08	30.00
20Ns7m1	25.00	20.77	16.92	9.62	20.00	15.38	22.31	16.92	29.62	22.69	29.62	22.69
20Ns7m3	30.38	29.62	26.54	14.62	30.00	26.92	30.00	NaN	34.23	35.38	34.23	35.38
20Ns9m1	23.85	22.69	18.46	9.23	21.15	15.77	24.62	17.69	28.08	30.38	28.08	30.38
20Ns9m3	33.46	30.00	26.92	17.31	29.23	28.46	33.08	25.00	41.54	40.00	41.54	40.00
20NIs5m1	20.38	18.85	16.92	8.46	17.69	14.62	21.15	15.00	25.00	22.69	25.00	22.69
20NIs5m3	29.62	26.54	19.62	14.62	29.62	26.15	28.08	22.69	32.69	32.69	32.69	32.69
20NIs7m1	21.54	22.69	14.62	7.31	19.23	18.85	24.62	16.54	26.92	25.00	26.92	25.00
20NIs7m3	29.62	30.00	22.31	18.08	29.62	25.77	30.38	25.38	33.85	35.00	33.85	35.00
20NIs9m1	24.62	21.54	18.08	7.69	25.00	17.31	26.92	19.62	27.69	25.00	27.69	25.00
20NIs9m3	31.54	31.92	26.54	19.23	31.54	28.08	31.92	24.23	40.77	39.62	NaN	39.62

Table B.6: AVletters, a, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	16.92	18.08	11.92	9.23	19.62	14.23	16.92	13.85	18.46	16.54	18.46	16.54
10Ns5m3	29.62	33.46	17.69	18.85	28.08	23.46	28.08	26.15	29.23	NaN	29.23	28.85
10Ns7m1	16.92	15.00	11.92	9.62	19.23	16.15	18.85	18.46	21.92	NaN	21.92	17.69
10Ns7m3	31.15	28.46	16.54	17.31	27.69	23.46	NaN	28.46	36.15	31.54	36.15	31.54
10Ns9m1	19.23	23.46	13.46	8.08	21.54	13.85	20.77	18.46	26.15	25.00	26.15	25.00
10Ns9m3	31.54	34.62	21.15	20.00	28.08	27.31	25.77	30.38	39.23	NaN	39.23	33.85
10NIs5m	18.46	15.00	11.15	9.62	20.38	16.15	18.46	15.38	16.15	17.31	16.15	17.31
10NIs5m3	27.69	26.15	18.46	17.31	25.38	23.85	25.00	26.15	30.38	30.00	30.38	30.00
10NIs7m1	16.15	20.00	13.08	10.00	20.38	16.92	20.77	14.62	19.62	21.54	19.62	21.54
10NIs7m3	32.69	30.77	18.08	20.77	30.00	23.85	28.08	27.31	35.38	30.38	35.38	30.38
10NIs9m1	21.92	22.69	15.77	10.77	20.77	16.92	23.85	16.15	22.31	22.69	22.31	22.69
10NIs9m3	35.00	32.69	19.23	21.15	24.23	26.92	27.31	NaN	38.08	36.54	38.08	36.54
20Ns5m1	19.62	17.31	15.00	7.31	20.38	14.62	22.31	13.46	27.69	21.54	27.69	21.54
20Ns5m3	36.54	28.46	21.15	18.08	26.92	25.77	30.00	27.31	32.69	30.00	32.69	30.00
20Ns7m1	22.69	21.92	18.85	8.46	20.00	15.38	22.31	16.92	29.62	22.69	29.62	22.69
20Ns7m3	32.69	33.46	24.23	16.92	30.00	26.92	30.00	NaN	33.85	35.38	33.85	35.38
20Ns9m1	22.31	23.08	20.00	8.46	21.15	15.77	24.62	17.69	28.08	30.38	28.08	30.38
20Ns9m3	35.38	34.23	22.31	18.85	29.23	28.46	33.08	25.00	41.92	40.00	41.92	40.00
20NIs5m1	20.77	18.85	13.85	7.69	17.69	14.62	21.15	15.00	25.00	22.69	25.00	22.69
20NIs5m3	32.31	31.54	21.54	17.31	29.62	26.15	28.08	22.69	32.69	33.08	32.69	33.08
20NIs7m1	24.23	21.15	20.00	8.46	19.23	18.85	24.62	16.54	26.92	25.00	26.92	25.00
20NIs7m3	31.92	30.77	25.77	18.08	29.62	25.77	30.38	25.38	33.85	35.00	33.85	NaN
20NIs9m1	23.85	24.62	20.38	11.15	25.00	17.31	26.92	19.62	27.69	25.00	27.69	25.00
20NIs9m3	36.15	36.15	19.23	18.85	31.54	28.08	31.92	24.23	40.77	39.62	40.77	NaN

Table B.7: AVletters, $|a|$, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	16.54	13.08	6.92	4.62	15.00	12.31	16.92	13.08	20.77	19.62	20.77	19.62
10Ns5m3	26.15	31.15	11.15	5.77	21.92	26.15	22.69	18.08	29.62	27.69	29.62	27.69
10Ns7m1	18.08	16.92	10.38	5.00	13.85	15.00	20.77	13.85	22.31	NaN	22.31	24.23
10Ns7m3	29.62	33.08	13.46	6.54	21.92	24.23	23.85	20.38	31.15	NaN	31.15	NaN
10Ns9m1	16.54	15.00	8.85	5.00	15.38	12.31	21.15	14.23	27.69	26.54	27.69	26.54
10Ns9m3	29.23	33.46	NaN	7.69	24.62	28.85	21.92	25.00	35.00	36.15	35.00	36.15
10NIs5m	15.38	13.46	8.46	5.38	12.69	13.46	15.38	11.54	21.92	17.69	21.92	17.69
10NIs5m3	28.08	33.46	NaN	8.08	20.38	24.62	23.46	18.85	29.23	33.08	29.23	33.08
10NIs7m1	20.00	15.77	9.62	4.23	18.46	15.38	18.08	10.38	24.62	25.00	24.62	25.00
10NIs7m3	29.23	32.69	13.85	9.23	23.46	25.77	27.31	23.85	NaN	NaN	NaN	NaN
10NIs9m1	18.46	17.31	8.46	5.38	16.54	15.00	16.15	15.00	28.08	24.62	28.08	24.62
10NIs9m3	33.46	34.62	15.00	7.69	26.15	25.00	25.00	20.77	34.23	38.08	34.23	38.08
20Ns5m1	17.31	17.31	12.31	5.00	16.54	15.77	14.62	11.92	20.00	19.23	20.00	19.23
20Ns5m3	27.31	28.85	15.00	6.92	27.31	25.38	23.08	16.54	31.15	33.46	31.15	33.46
20Ns7m1	20.00	17.31	14.62	5.38	18.85	19.62	18.46	12.69	25.00	25.00	25.00	25.00
20Ns7m3	29.23	33.85	16.15	6.92	27.31	29.23	23.85	23.08	31.54	40.38	31.54	40.38
20Ns9m1	23.08	21.92	13.85	4.23	19.62	20.00	23.85	14.62	28.46	28.46	28.46	28.46
20Ns9m3	33.85	35.77	19.23	8.08	31.15	28.08	26.54	21.92	32.31	38.46	32.31	38.46
20NIs5m1	18.85	17.31	10.00	4.62	NaN	14.62	13.85	10.38	20.00	20.38	20.00	20.38
20NIs5m3	29.62	32.69	18.08	7.69	26.92	26.92	29.62	17.69	28.08	38.08	28.08	38.08
20NIs7m1	17.69	20.77	10.77	4.23	18.46	17.31	19.62	11.15	25.00	23.08	25.00	23.08
20NIs7m3	31.92	35.38	16.15	8.85	25.00	28.08	24.62	23.08	35.77	36.15	35.77	36.15
20NIs9m1	20.77	23.46	15.38	5.38	18.08	21.92	21.92	13.08	28.46	29.23	28.46	29.23
20NIs9m3	38.08	37.69	22.31	8.46	27.31	31.15	25.00	NaN	35.00	38.85	35.00	38.85

Table B.8: AVletters, a^2, linear scaled

B.2 Tulips Database

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	77.08	39.58	79.17	60.42	64.58	47.92	58.33	54.17	68.75	33.33	68.75	33.33
10Ns5m3	54.17	54.17	62.50	NaN	41.67	37.50	50.00	47.92	NaN	52.08	NaN	52.08
10Ns7m1	75.00	45.45	70.45	61.36	50.00	36.36	52.27	52.27	70.45	54.55	70.45	54.55
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	60.00	40.00	70.00	46.67	50.00	36.67	56.67	50.00	76.67	60.00	76.67	60.00
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	66.67	41.67	56.25	56.25	58.33	50.00	43.75	35.42	64.58	43.75	64.58	43.75
10NIs5m3	54.17	41.67	60.42	52.08	35.42	39.58	50.00	43.75	47.92	47.92	47.92	47.92
10NIs7m1	72.92	45.83	77.08	62.50	62.50	47.92	56.25	47.92	70.83	52.08	70.83	52.08
10NIs7m3	58.33	52.08	75.00	47.92	45.83	43.75	45.83	45.83	60.42	50.00	60.42	50.00
10NIs9m1	75.00	39.58	75.00	62.50	58.33	41.67	58.33	50.00	70.83	54.17	70.83	54.17
10NIs9m3	68.75	56.25	70.83	47.92	43.75	41.67	43.75	43.75	68.75	50.00	68.75	50.00
20Ns5m1	64.58	43.75	68.75	54.17	60.42	47.92	60.42	43.75	66.67	56.25	66.67	56.25
20Ns5m3	60.42	43.75	60.42	56.25	47.92	39.58	52.08	47.92	NaN	45.83	NaN	45.83
20Ns7m1	65.91	43.18	63.64	52.27	43.18	45.45	56.82	47.73	68.18	54.55	68.18	54.55
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	56.67	33.33	56.67	50.00	43.33	43.33	56.67	46.67	66.67	50.00	66.67	50.00
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	68.75	37.50	70.83	52.08	56.25	45.83	58.33	33.33	66.67	41.67	66.67	41.67
20NIs5m3	60.42	52.08	58.33	47.92	37.50	37.50	56.25	39.58	56.25	39.58	56.25	39.58
20NIs7m1	62.50	43.75	70.83	56.25	52.08	41.67	62.50	41.67	64.58	56.25	64.58	56.25
20NIs7m3	52.08	47.92	60.42	47.92	37.50	41.67	45.83	54.17	52.08	47.92	52.08	47.92
20NIs9m1	68.75	43.75	79.17	54.17	56.25	37.50	62.50	45.83	70.83	50.00	70.83	50.00
20NIs9m3	54.17	52.08	66.67	37.50	45.83	37.50	47.92	43.75	58.33	54.17	58.33	54.17

Table B.9: Tulips, *sh*

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	58.33	75.00	47.92	60.42	58.33	37.50	68.75	45.83	60.42	41.67	60.42	41.67
10Ns5m3	60.42	56.25	47.92	58.33	52.08	45.83	60.42	47.92	64.58	45.83	64.58	45.83
10Ns7m1	50.00	70.45	54.55	56.82	43.18	34.09	50.00	52.27	61.36	36.36	61.36	36.36
10Ns7m3	45.45	56.82	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	43.33	56.67	53.33	60.00	43.33	26.67	53.33	40.00	60.00	40.00	60.00	40.00
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	56.25	62.50	56.25	56.25	60.42	43.75	58.33	41.67	66.67	54.17	66.67	54.17
10NIs5m3	41.67	58.33	50.00	54.17	47.92	43.75	39.58	52.08	54.17	54.17	54.17	54.17
10NIs7m1	54.17	72.92	47.92	62.50	58.33	37.50	68.75	43.75	77.08	45.83	77.08	45.83
10NIs7m3	56.25	52.08	50.00	64.58	52.08	43.75	47.92	41.67	58.33	45.83	58.33	45.83
10NIs9m1	56.25	70.83	52.08	66.67	56.25	45.83	62.50	43.75	75.00	47.92	75.00	47.92
10NIs9m3	54.17	64.58	54.17	56.25	45.83	45.83	56.25	50.00	72.92	43.75	72.92	43.75
20Ns5m1	47.92	58.33	66.67	54.17	70.83	33.33	62.50	58.33	70.83	37.50	70.83	37.50
20Ns5m3	62.50	56.25	56.25	68.75	37.50	35.42	54.17	52.08	56.25	45.83	56.25	45.83
20Ns7m1	50.00	52.27	63.64	50.00	40.91	34.09	50.00	56.82	56.82	36.36	56.82	36.36
20Ns7m3	47.73	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	43.33	50.00	56.67	46.67	50.00	30.00	50.00	50.00	53.33	33.33	53.33	33.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	45.83	58.33	62.50	50.00	47.92	39.58	58.33	45.83	62.50	37.50	62.50	37.50
20NIs5m3	47.92	54.17	45.83	64.58	47.92	52.08	54.17	45.83	52.08	50.00	52.08	50.00
20NIs7m1	52.08	58.33	68.75	62.50	62.50	41.67	58.33	54.17	66.67	35.42	66.67	35.42
20NIs7m3	54.17	64.58	54.17	60.42	54.17	52.08	47.92	54.17	58.33	47.92	58.33	47.92
20NIs9m1	56.25	64.58	62.50	68.75	58.33	35.42	58.33	54.17	64.58	41.67	64.58	41.67
20NIs9m3	56.25	66.67	58.33	58.33	41.67	41.67	50.00	62.50	68.75	43.75	68.75	43.75

Table B.10: Tulips, *a*

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	54.17	43.75	54.17	47.92	58.33	37.50	68.75	45.83	60.42	41.67	60.42	41.67
10Ns5m3	45.83	47.92	58.33	45.83	52.08	45.83	60.42	47.92	64.58	45.83	64.58	45.83
10Ns7m1	40.91	34.09	65.91	43.18	43.18	34.09	50.00	52.27	61.36	36.36	61.36	36.36
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	30.00	30.00	63.33	33.33	43.33	26.67	53.33	40.00	60.00	40.00	60.00	40.00
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	45.83	47.92	54.17	52.08	60.42	43.75	58.33	41.67	66.67	54.17	66.67	54.17
10NIs5m3	50.00	54.17	54.17	50.00	47.92	43.75	39.58	52.08	54.17	54.17	54.17	54.17
10NIs7m1	66.67	45.83	52.08	54.17	58.33	37.50	68.75	43.75	77.08	45.83	77.08	45.83
10NIs7m3	41.67	54.17	58.33	45.83	52.08	43.75	47.92	41.67	58.33	45.83	58.33	45.83
10NIs9m1	56.25	35.42	68.75	50.00	56.25	45.83	62.50	43.75	75.00	47.92	75.00	47.92
10NIs9m3	41.67	50.00	56.25	52.08	45.83	45.83	56.25	50.00	72.92	43.75	72.92	43.75
20Ns5m1	50.00	33.33	64.58	35.42	70.83	33.33	62.50	58.33	70.83	37.50	70.83	37.50
20Ns5m3	50.00	39.58	62.50	50.00	37.50	35.42	54.17	52.08	56.25	45.83	56.25	45.83
20Ns7m1	47.73	40.91	59.09	43.18	40.91	34.09	50.00	56.82	56.82	36.36	56.82	36.36
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	36.67	20.00	56.67	33.33	50.00	30.00	50.00	50.00	53.33	33.33	53.33	33.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	52.08	41.67	62.50	50.00	47.92	39.58	58.33	45.83	62.50	37.50	62.50	37.50
20NIs5m3	41.67	39.58	58.33	45.83	47.92	52.08	54.17	45.83	52.08	50.00	52.08	50.00
20NIs7m1	52.08	37.50	54.17	39.58	62.50	41.67	58.33	54.17	66.67	35.42	66.67	35.42
20NIs7m3	47.92	35.42	58.33	41.67	54.17	52.08	47.92	54.17	58.33	47.92	58.33	47.92
20NIs9m1	50.00	39.58	62.50	50.00	58.33	35.42	58.33	54.17	64.58	41.67	64.58	41.67
20NIs9m3	47.92	41.67	58.33	43.75	41.67	41.67	50.00	62.50	68.75	NaN	68.75	43.75

Table B.11: Tulips, $|a|$

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	33.33	37.50	43.75	41.67	41.67	41.67	37.50	58.33	54.17	45.83	54.17	45.83
10Ns5m3	37.50	39.58	45.83	41.67	47.92	54.17	47.92	56.25	54.17	NaN	54.17	47.92
10Ns7m1	47.73	38.64	34.09	31.82	43.18	43.18	45.45	59.09	50.00	45.45	50.00	45.45
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	46.67	36.67	46.67	40.00	40.00	33.33	46.67	53.33	46.67	46.67	46.67	46.67
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	45.83	35.42	39.58	41.67	43.75	41.67	47.92	58.33	70.83	41.67	70.83	41.67
10NIs5m3	41.67	35.42	56.25	41.67	45.83	39.58	58.33	50.00	58.33	50.00	58.33	50.00
10NIs7m1	43.75	43.75	50.00	45.83	45.83	43.75	45.83	58.33	64.58	43.75	64.58	43.75
10NIs7m3	54.17	39.58	39.58	39.58	39.58	41.67	56.25	58.33	62.50	47.92	62.50	47.92
10NIs9m1	43.75	39.58	45.83	29.17	47.92	50.00	52.08	56.25	64.58	47.92	64.58	47.92
10NIs9m3	41.67	29.17	43.75	50.00	39.58	39.58	56.25	NaN	64.58	50.00	64.58	50.00
20Ns5m1	47.92	39.58	52.08	43.75	41.67	43.75	47.92	60.42	52.08	45.83	52.08	45.83
20Ns5m3	52.08	43.75	45.83	35.42	41.67	43.75	56.25	50.00	45.83	47.92	45.83	47.92
20Ns7m1	45.45	38.64	40.91	36.36	43.18	43.18	40.91	56.82	52.27	47.73	52.27	47.73
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	36.67	33.33	53.33	33.33	43.33	30.00	50.00	53.33	36.67	33.33	36.67	33.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	45.83	54.17	45.83	37.50	58.33	50.00	45.83	64.58	56.25	35.42	56.25	35.42
20NIs5m3	47.92	41.67	41.67	29.17	52.08	37.50	56.25	58.33	54.17	45.83	54.17	45.83
20NIs7m1	47.92	52.08	50.00	39.58	50.00	33.33	56.25	68.75	60.42	43.75	60.42	43.75
20NIs7m3	45.83	45.83	39.58	37.50	50.00	35.42	58.33	52.08	52.08	45.83	52.08	45.83
20NIs9m1	43.75	50.00	54.17	37.50	45.83	39.58	45.83	64.58	66.67	56.25	66.67	56.25
20NIs9m3	43.75	41.67	45.83	39.58	45.83	41.67	62.50	56.25	52.08	50.00	52.08	50.00

Table B.12: Tulips, a^2

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	41.67	43.75	50.00	50.00	39.58	35.42	43.75	50.00	64.58	60.42	64.58	60.42
10Ns5m3	37.50	56.25	56.25	39.58	37.50	45.83	47.92	60.42	58.33	47.92	58.33	47.92
10Ns7m1	40.91	38.64	61.36	47.73	34.09	36.36	43.18	52.27	61.36	54.55	61.36	54.55
10Ns7m3	NaN	NaN	NaN	NaN	45.45	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	43.33	26.67	53.33	46.67	36.67	26.67	30.00	43.33	60.00	63.33	60.00	63.33
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	45.83	43.75	52.08	50.00	45.83	37.50	47.92	52.08	62.50	52.08	62.50	52.08
10NIs5m3	37.50	47.92	54.17	39.58	45.83	43.75	58.33	47.92	60.42	43.75	60.42	43.75
10NIs7m1	47.92	45.83	52.08	52.08	47.92	43.75	58.33	58.33	54.17	47.92	54.17	47.92
10NIs7m3	39.58	45.83	45.83	58.33	50.00	43.75	47.92	52.08	52.08	58.33	52.08	58.33
10NIs9m1	47.92	43.75	58.33	52.08	50.00	39.58	56.25	50.00	58.33	54.17	58.33	54.17
10NIs9m3	50.00	50.00	50.00	58.33	43.75	NaN	52.08	62.50	50.00	52.08	50.00	52.08
20Ns5m1	50.00	45.83	47.92	45.83	43.75	27.08	50.00	50.00	60.42	56.25	60.42	56.25
20Ns5m3	56.25	52.08	45.83	45.83	NaN	45.83	47.92	54.17	58.33	47.92	58.33	47.92
20Ns7m1	52.27	43.18	40.91	45.45	40.91	34.09	43.18	50.00	47.73	54.55	47.73	54.55
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	60.00	36.67	46.67	40.00	40.00	36.67	40.00	26.67	40.00	56.67	40.00	56.67
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	47.92	52.08	54.17	39.58	54.17	35.42	47.92	47.92	54.17	52.08	54.17	52.08
20NIs5m3	50.00	45.83	54.17	47.92	41.67	47.92	47.92	52.08	56.25	47.92	56.25	47.92
20NIs7m1	56.25	54.17	45.83	45.83	45.83	37.50	50.00	54.17	56.25	47.92	56.25	47.92
20NIs7m3	41.67	56.25	58.33	45.83	37.50	43.75	43.75	54.17	62.50	52.08	62.50	52.08
20NIs9m1	45.83	52.08	47.92	54.17	47.92	37.50	52.08	50.00	58.33	50.00	58.33	50.00
20NIs9m3	54.17	35.42	52.08	50.00	35.42	NaN	45.83	56.25	60.42	54.17	60.42	54.17

Table B.13: Tulips, *sh*, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	50.00	39.58	41.67	39.58	39.58	43.75	43.75	43.75	45.83	47.92	45.83	47.92
10Ns5m3	52.08	35.42	37.50	43.75	54.17	NaN	50.00	45.83	47.92	47.92	47.92	47.92
10Ns7m1	43.18	47.73	43.18	29.55	40.91	38.64	38.64	56.82	50.00	43.18	50.00	43.18
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	33.33	36.67	43.33	36.67	43.33	26.67	30.00	50.00	46.67	50.00	46.67	50.00
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	45.83	39.58	41.67	41.67	50.00	37.50	43.75	39.58	50.00	45.83	50.00	45.83
10NIs5m3	41.67	43.75	39.58	50.00	45.83	41.67	50.00	47.92	41.67	50.00	41.67	50.00
10NIs7m1	43.75	47.92	41.67	37.50	41.67	37.50	47.92	47.92	58.33	45.83	58.33	45.83
10NIs7m3	56.25	50.00	37.50	56.25	41.67	47.92	43.75	45.83	47.92	47.92	47.92	47.92
10NIs9m1	37.50	50.00	47.92	43.75	50.00	35.42	41.67	50.00	54.17	52.08	54.17	52.08
10NIs9m3	52.08	NaN	37.50	43.75	45.83	54.17	39.58	56.25	52.08	58.33	52.08	58.33
20Ns5m1	45.83	39.58	41.67	35.42	45.83	37.50	54.17	37.50	45.83	56.25	45.83	56.25
20Ns5m3	47.92	54.17	37.50	43.75	52.08	43.75	47.92	50.00	50.00	50.00	50.00	50.00
20Ns7m1	36.36	47.73	47.73	31.82	36.36	45.45	43.18	50.00	50.00	47.73	50.00	47.73
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	40.00	40.00	40.00	30.00	43.33	26.67	36.67	50.00	43.33	53.33	43.33	53.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	43.75	45.83	43.75	39.58	47.92	41.67	50.00	43.75	45.83	43.75	45.83	43.75
20NIs5m3	41.67	47.92	39.58	43.75	37.50	37.50	39.58	47.92	41.67	52.08	41.67	52.08
20NIs7m1	50.00	50.00	45.83	39.58	47.92	39.58	43.75	52.08	43.75	52.08	43.75	52.08
20NIs7m3	45.83	37.50	41.67	29.17	41.67	39.58	39.58	43.75	43.75	50.00	43.75	50.00
20NIs9m1	43.75	43.75	43.75	29.17	45.83	39.58	52.08	43.75	43.75	45.83	43.75	45.83
20NIs9m3	41.67	47.92	43.75	33.33	39.58	47.92	39.58	45.83	41.67	52.08	41.67	52.08

Table B.14: Tulips, a, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	43.75	43.75	45.83	37.50	39.58	43.75	43.75	43.75	45.83	47.92	45.83	47.92
10Ns5m3	45.83	37.50	33.33	47.92	54.17	NaN	50.00	45.83	47.92	47.92	47.92	47.92
10Ns7m1	36.36	34.09	43.18	34.09	40.91	38.64	38.64	56.82	50.00	43.18	50.00	43.18
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	30.00	46.67	53.33	36.67	43.33	26.67	30.00	50.00	46.67	50.00	46.67	50.00
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	37.50	45.83	41.67	37.50	50.00	37.50	43.75	39.58	50.00	45.83	50.00	45.83
10NIs5m3	45.83	47.92	47.92	39.58	45.83	41.67	50.00	47.92	41.67	50.00	41.67	50.00
10NIs7m1	41.67	52.08	50.00	45.83	41.67	37.50	47.92	47.92	58.33	45.83	58.33	45.83
10NIs7m3	37.50	47.92	33.33	39.58	41.67	47.92	43.75	45.83	47.92	47.92	47.92	47.92
10NIs9m1	37.50	39.58	43.75	37.50	50.00	35.42	41.67	50.00	54.17	52.08	54.17	52.08
10NIs9m3	39.58	45.83	41.67	43.75	45.83	54.17	39.58	56.25	52.08	58.33	52.08	58.33
20Ns5m1	45.83	43.75	54.17	29.17	45.83	37.50	54.17	37.50	45.83	56.25	45.83	56.25
20Ns5m3	52.08	41.67	37.50	47.92	52.08	43.75	47.92	50.00	50.00	47.92	50.00	47.92
20Ns7m1	52.27	31.82	43.18	29.55	36.36	45.45	43.18	50.00	50.00	47.73	50.00	47.73
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	50.00	43.33	46.67	30.00	43.33	26.67	36.67	50.00	43.33	53.33	43.33	53.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	52.08	54.17	45.83	31.25	47.92	41.67	50.00	43.75	45.83	43.75	45.83	43.75
20NIs5m3	41.67	56.25	47.92	43.75	37.50	37.50	39.58	47.92	41.67	52.08	41.67	52.08
20NIs7m1	45.83	50.00	56.25	31.25	47.92	39.58	43.75	52.08	43.75	52.08	43.75	52.08
20NIs7m3	45.83	50.00	41.67	41.67	41.67	39.58	39.58	43.75	43.75	50.00	43.75	50.00
20NIs9m1	52.08	45.83	43.75	31.25	45.83	39.58	52.08	43.75	43.75	45.83	43.75	45.83
20NIs9m3	52.08	54.17	31.25	29.17	39.58	47.92	39.58	45.83	41.67	52.08	41.67	52.08

Table B.15: Tulips, $|a|$, linear scaled

	m				o				c			
	NoDc		PresDc		NoDc		PresDc		NoDc		PresDc	
Type	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor	Cov	Cor
10Ns5m1	31.25	39.58	20.83	31.25	39.58	52.08	31.25	41.67	43.75	35.42	43.75	35.42
10Ns5m3	39.58	58.33	31.25	27.08	47.92	52.08	39.58	45.83	41.67	37.50	41.67	37.50
10Ns7m1	36.36	38.64	31.82	25.00	34.09	45.45	31.82	38.64	38.64	40.91	38.64	40.91
10Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10Ns9m1	46.67	43.33	26.67	30.00	33.33	46.67	33.33	36.67	40.00	53.33	40.00	53.33
10Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
10NIs5m	43.75	45.83	33.33	29.17	35.42	45.83	31.25	45.83	50.00	50.00	50.00	50.00
10NIs5m3	45.83	45.83	33.33	25.00	52.08	52.08	29.17	47.92	31.25	35.42	31.25	35.42
10NIs7m1	37.50	45.83	29.17	31.25	33.33	45.83	29.17	41.67	41.67	41.67	41.67	41.67
10NIs7m3	37.50	37.50	37.50	22.92	45.83	47.92	45.83	50.00	37.50	39.58	37.50	39.58
10NIs9m1	37.50	50.00	29.17	29.17	41.67	45.83	31.25	39.58	41.67	47.92	41.67	47.92
10NIs9m3	35.42	47.92	43.75	33.33	54.17	47.92	35.42	47.92	37.50	52.08	37.50	52.08
20Ns5m1	31.25	35.42	35.42	31.25	39.58	39.58	25.00	39.58	37.50	43.75	37.50	43.75
20Ns5m3	35.42	50.00	43.75	27.08	50.00	47.92	45.83	39.58	45.83	43.75	45.83	43.75
20Ns7m1	31.82	36.36	43.18	31.82	43.18	34.09	20.45	29.55	31.82	40.91	31.82	40.91
20Ns7m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20Ns9m1	36.67	30.00	53.33	36.67	33.33	40.00	26.67	30.00	30.00	43.33	30.00	43.33
20Ns9m3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
20NIs5m1	39.58	41.67	39.58	35.42	37.50	35.42	39.58	43.75	35.42	39.58	35.42	39.58
20NIs5m3	37.50	58.33	37.50	29.17	41.67	54.17	41.67	45.83	35.42	35.42	35.42	35.42
20NIs7m1	31.25	31.25	41.67	31.25	39.58	47.92	33.33	41.67	37.50	47.92	37.50	47.92
20NIs7m3	39.58	37.50	37.50	27.08	43.75	39.58	41.67	43.75	37.50	47.92	37.50	47.92
20NIs9m1	37.50	33.33	39.58	22.92	43.75	39.58	33.33	41.67	35.42	41.67	35.42	41.67
20NIs9m3	33.33	52.08	43.75	27.08	47.92	43.75	25.00	43.75	45.83	43.75	45.83	43.75

Table B.16: Tulips, a^2, linear scaled

Bibliography

[1] A. Adjoudani and C. Benoît. On the integration of auditory and visual parameters in an HMM-based ASR. In Stork and Hennecke [181], pages 461–471.

[2] H. C. Andrews. *Introduction to Mathematical Techniques in Pattern Recognition.* John Wiley, 1983.

[3] J. A. Bangham, P. Chardaire, C. J. Pye, and P. D. Ling. Mulitscale nonlinear decomposition: The sieve decomposition theorem. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(5):529–539, 1996.

[4] J. A. Bangham, R. Harvey, P. Ling, and R. V. Aldridge. Morphological scale-space preserving transforms in many dimensions. *Journal of Electronic Imaging*, 5(3):283–299, July 1996.

[5] J. A. Bangham, R. Harvey, P. Ling, and R. V. Aldridge. Nonlinear scale-space from n-dimensional sieves. *Proc. European Conference on Computer Vision*, 1:189–198, 1996.

[6] J. A. Bangham, S. J. Impey, and F. W. D. Woodhams. A fast 1D sieve transform for multiscale signal decomposition. In *Proc. European Conference on Signal and Image Processing*, pages 1621–1624, 1984.

[7] J. A. Bangham, P. Ling, and R. Harvey. Scale-space from nonlinear filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(5):520–528, 1996.

[8] J. A. Bangham, P. Ling, and R. Young. Mulitscale recursive medians, scale-space and transforms with applications to image processing. *IEEE Trans. Image Processing*, 5(6):1043–1048, 1996.

[9] S. Basu, N. Oliver, and A. Pentland. 3D modeling and tracking of human lip motions. In *Proc. International Conference on Computer Vision*, 1998.

[10] S. Basu and A. Pentland. Recovering 3D lip structure from 2D observations using a model trained from video. In Benoît and Campbell [13], pages 121–124.

[11] S. Basu and A. Pentland. A three-dimensional model of human lip motions trained from video. Technical Report 441, MIT, Media Laboratory, 1997.

[12] L. E. Baum and J. A. Egon. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*, 3:1–8, 1972.

[13] C. Benoît and R. Campbell, editors. *Proceedings of the ESCA Workshop on Audio-Visual Speech Processing*, Rhodes, Sept. 1997.

[14] C. Benoît, T. Guiard-Marigny, B. Le Goff, and A. Adjoudani. Which components of the face do humans and machines best speechread? In Stork and Hennecke [181], pages 315–328.

[15] M. Black and Y. Yacoob. Tracking and recognising rigid and non-rigid facial motions using local parametric models of image motion. In *Proc. 5th Int. Conf. on Computer Vision*, pages 374–381, 1995.

[16] A. Blake, B. Bascle, M. Isard, and J. MacCormick. Statistical models of visual shape and motion. *Proc. Roy. Soc. Lond. A*, 356:1283–1302, 1998.

[17] A. Blake and M. Isard. *Active Contours: The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion.* Springer-Verlag, 1998.

[18] A. Blake and A. Yuille, editors. *Active Vision.* MIT Press, 1992.

[19] S. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 27:113–120, 1979.

[20] A. Bosson. *Robustness of Scale Spaces.* PhD thesis, School of Information Systems, University of East Anglia, 1998.

[21] A. Bosson and R. Harvey. Using occlusion models to evaluate scale space processors. In *Proc. IEEE International Conference on Image Processing*, 1998.

[22] L. D. Braida. Crossmodal integration in the identification of consonant segments. *The Quarterly Journal of Experimental Psychology*, 43A(3):647–677, 1991.

[23] C. Bregler, H. Hild, S. Manke, and A. Waibel. Improving connected letter recognition by lipreading. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 557–560, Minneapolis, 1993. IEEE.

[24] C. Bregler and Y. Konig. 'Eigenlips' for robust speech recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, pages 669–672, Adelaide, 1994. IEEE.

[25] C. Bregler and S. M. Omohundro. Surface learning with applications to lipreading. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 43–50, San Francisco, CA, 1994. Morgan Kaufmann.

[26] C. Bregler and S. M. Omohundro. *Learning Visual Models for Lipreading*, chapter 13, pages 301–320. Volume 9 of Shah and Jain [173], 1997.

[27] C. Bregler, S. M. Omohundro, and Y. Konig. A hybrid approach to bimodal speech recognition. In *28th Annual Asilomar Conference on Signals, Systems, and Computers*, volume 1, pages 556–560, Pacific Grove, CA, Nov. 1994.

[28] C. Bregler, S. M. Omohundro, and J. Shi. Towards a robust speechreading dialog system. In Stork and Hennecke [181], pages 409–423.

[29] N. M. Brooke. Talking heads and speech recognisers that can see: The computer processing of visual speech signals. In Stork and Hennecke [181], pages 351–371.

[30] N. M. Brooke. Using the visual component in automatic speech recognition. In *Proc. International Conference on Spoken Language Processing*, volume 1, pages 1656–1659, Philadelphia, PA, Oct. 1996.

[31] N. M. Brooke. Computational aspects of visual speech: machines that can speechread and simulate talking faces. In Campbell et al. [41], pages 109–122.

[32] N. M. Brooke and S. D. Scott. PCA image coding schemes and visual speech intelligibility. *Proc. Institute of Acoustics*, 16(5):123–129, 1994.

[33] N. M. Brooke, S. D. Scott, and M. J. Tomlinson. Making talking heads and speechreading with computers. In *IEE Colloquium on Integrated Audio-Visual Processing*, 1996/213, pages 2/1–2/6, Savoy Place, London, Nov. 1996.

[34] N. M. Brooke, M. J. Tomlinson, and R. K. Moore. Automatic speech recognition that includes visual speech cues. *Proc. Institute of Acoustics*, 16(5):15–22, 1994.

[35] J. Bulwer. *Philocopus, or the Deaf and Dumbe Mans Friend*. Humphrey and Moseley, 1648.

[36] D. Burnham and B. Dodd. Auditory-visual speech perception as a direct process: The McGurk effect in infants and across languages. In Stork and Hennecke [181], pages 103–114.

[37] R. Campbell. Lipreading. In A. W. Young and H. D. Ellis, editors, *Handbook of Research on Face Processing*, chapter 4, pages 187–205. North-Holland, 1989.

[38] R. Campbell. The neuropsychology of lipreading. *Phil. Trans. R. Soc. Lond. B*, 335:39–45, 1992.

[39] R. Campbell. Seeing brains reading speech: A review and speculations. In Stork and Hennecke [181], pages 115–133.

[40] R. Campbell. Seeing speech in space and time: Psychological and neurological findings. In *Proc. International Conference on Spoken Language Processing*, volume 3, pages 1493–1496, Philadelphia, PA, Oct. 1996.

[41] R. Campbell, B. Dodd, and D. Burnham, editors. *Hearing by Eye II: Advances in the Psychology of Speechreading and Auditory-visual Speech*. Psychology Press, 1998.

[42] D. Chandramohan and P. L. Silsbee. A multiple deformable template approach for visual speech recognition. In *Proc. International Conference on Spoken Language Processing*, volume 1, pages 50–53, Philadelphia, PA, Oct. 1996.

[43] C. Chatfield and A. J. Collins. *Introduction to Multivariate Analysis*. Chapman and Hall, 1991.

[44] R. Chellappa, C. L. Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. *Proceedings of the IEE*, 83(5):705–740, May 1995.

[45] T. Chen and R. R. Rao. Audio-visual integration in multimodal communication. *Proceedings of the IEEE*, 86(5):837–852, May 1998.

[46] C. C. Chibelushi, S. Gandon, J. S. D. Mason, F. Deravi, and R. D. Johnston. Desing issues for a digital audio-visual integrated database. In *IEE Colloquium on Integrated Audio-Visual Processing*, number 1996/213, pages 7/1–7/7, Savoy Place, London, Nov. 1996.

[47] J. Clark and C. Yallop. *An Introduction to Phonetics and Phonology*. Blackwell, second edition, 1995.

[48] T. Coianiz, L. Torresani, and B. Caprile. 2D deformable models for visual speech analysis. In Stork and Hennecke [181], pages 391–398.

[49] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proc. European Conference on Computer Vision*, pages 484–498, June 1998.

[50] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. *Image and Vision Computing*, 12(6):355–366, 1994.

[51] T. F. Cootes and C. J. Taylor. Active shape models – 'smart snakes'. In D. Hogg and R. Boyle, editors, *Proc. British Machine Vision Conference*, pages 266–275. BMVA Press, 1992.

[52] T. F. Cootes and C. J. Taylor. Active shape model search using local grey-level models: a quantitative evaluation. In J. Illingworth, editor, *Proc. British Machine Vision Conference*, pages 639–648. BMVA Press, 1993.

[53] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Training models of shape from sets of examples. In D. Hogg and R. Boyle, editors, *Proc. British Machine Vision Conference*, pages 9–18. BMVA Press, 1992.

[54] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, Jan. 1995.

[55] T. F. Cootes, C. J. Taylor, and A. Lanitis. Active shape models: Evaluation of a multiresolution method for improving image search. In E. Hancock, editor, *Proc. British Machine Vision Conference*, pages 327–336, 1994.

[56] S. Cox, I. Matthews, and A. Bangham. Combining noise compensation with visual information in speech recognition. In Benoît and Campbell [13], pages 53–56.

[57] S. J. Cox. Hidden Markov models for automatic speech recognition: theory and application. *British Telecom Technical Journal*, 6(2):105–115, Apr. 1988.

[58] B. Dalton, R. Kaucic, and A. Blake. Automatic speechreading using dynamic contours. In Stork and Hennecke [181], pages 373–382.

[59] B. V. Dasarathy. Sensor fusion and potential exploitation—inovative architectures and illustrative approaches. *Proc. IEEE*, 85(1):24–38, Jan. 1997.

[60] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 28(4):357–366, Aug. 1980.

[61] B. Dodd and R. Campbell, editors. *Hearing by Eye: The Psychology of Lip-reading*. Lawrence Erlbaum Associates, London, 1987.

[62] E. R. Dougherty and J. Astola. *An Introduction to Nonlinear Image Processing*, volume 16 of *TT*. SPIE, 1994.

[63] P. Duchnowski, M. Hunke, D. Büsching, U. Meier, and A. Waibel. Toward movement-invariant automatic lip-reading and speech recognition. In *Proc. International Conference on Spoken Language Processing*, pages 109–112, 1995.

[64] P. Duchnowski, U. Meier, and A. Waibel. See me, hear me: Integrating automatic speech recognition and lip-reading. In *Proc. International Conference on Spoken Language Processing*, 1994.

[65] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Face recognition using active appearance models. In *Proc. European Conference on Computer Vision*, pages 582–595, June 1998.

[66] N. P. Erber. Auditory-visual perception of speech. *Journal of Speech and Hearing Disorders*, 40:481–492, 1975.

[67] K. E. Finn and A. A. Montgomery. Automatic optically-based recognition of speech. *Pattern Recognition Letters*, 8(3):159–164, 1988.

[68] J. Fransen, D. Pye, T. Robinson, P. Woodland, and S. Young. WSJCAM0 corpus and recording description. Technical Report TR.192, Cambridge University, Engineering Department, Speech Group, Sept. 1994.

[69] M. French-St. George and R. G. Stoker. Speechreading: An historical perspective. *The Volta Review*, 90(5):17–31, Sept. 1988.

[70] A. J. Goldschen. *Continuous Automatic Speech Recognition by Lipreading*. PhD thesis, George Washington University, 1993.

[71] A. J. Goldschen, O. N. Garcia, and E. D. Petajan. Rationale for phoneme-viseme mapping and feature selection in visual speech recognition. In Stork and Hennecke [181], pages 505–515.

[72] A. J. Goldschen, O. S. Garcia, and E. D. Petajan. *Continuous Automatic Speech Recognition by Lipreading*, chapter 14, pages 321–343. Volume 9 of Shah and Jain [173], 1997.

[73] M. S. Gray, J. R. Movellan, and T. J. Sejnowski. Dynamic features for visual speechreading: A systematic comparison. In M. C. Mozer, M. I. Jordon, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, Cambridge, MA, 1997.

[74] K. P. Green. The use of auditory and visual information in phonetic perception. In Stork and Hennecke [181], pages 55–77.

[75] K. P. Green. The use of auditory and visual information during phonetic processing: implications for theories of speech perception. In Campbell et al. [41], pages 3–25.

[76] J. K. Hackett and M. Shah. Multi-sensor fusion: A perspective. In *Proc. IEEE Conf. on Robotics and Automation*, 1324–1330, 1990.

[77] D. L. Hall and J. Llinas. An introduction to multisensor data fusion. *Proc. IEEE*, 85(1):6–23, Jan. 1997.

[78] R. M. Haralick, S. R. Sternberg, and X. Zhuang. Image analysis using mathematical morphology. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(4):532–550, July 1987.

[79] R. Harvey, A. Bosson, and J. A. Bangham. Robustness of some scale-spaces. In *Proc. British Machine Vision Conference*, volume 1, pages 11–20, 1997.

[80] R. Harvey, I. Matthews, J. A. Bangham, and S. Cox. Lip reading from scale-space measurements. In *Proc. Computer Vision and Pattern Recognition*, pages 582–587, Puerto Rico, June 1997. IEEE.

[81] J. Haslam, C. J. Taylor, and T. F. Cootes. A probabilistic fitness measure for deformable template models. In *Proc. British Machine Vision Conference*, pages 33–42. BMVA Press, 1994.

[82] M. E. Hennecke. *Audio-Visual Speech Recognition: Preprocessing, Learning and Sensory Integration.* PhD thesis, Stanford University, Sept. 1997.

[83] M. E. Hennecke, K. V. Prasad, and D. G. Stork. Using deformable templates to infer visual speech dynamics. In *28th Annual Asilomar Conference on Signals, Systems, and Computers*, volume 1, pages 578–582, 1994.

[84] M. E. Hennecke, K. V. Prasad, and D. G. Stork. Automatic speech recognition system using acoustic and visual signals. In *29th Annual Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1995.

[85] M. E. Hennecke, D. G. Stork, and K. V. Prasad. Visionary speech: Looking ahead to practical speechreading systems. In Stork and Hennecke [181], pages 331–349.

[86] A. Hill and C. J. Taylor. Automatic landmark generation for point distribution models. In *Proc. British Machine Vision Conference*, pages 429–438, 1994.

[87] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 23:417–441, 498–520, 1933.

[88] P. L. Jackson, A. A. Montgomery, and C. A. Binnie. Perceptual dimensions underlying vowel lipreading performance. *Journal of Speech and Hearing Research*, 19:796–812, 1976.

[89] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice-Hall, third edition, 1992.

[90] T. R. Jordan and P. C. Sergeant. Effects of facial image size on visual and audio-visual speech recogniton. In Campbell et al. [41], pages 155–176.

[91] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[92] R. Kaucic and A. Blake. Accurate, real-time, unadorned lip tracking. In *Proc 6th Int. Conf. Computer Vision*, 1998.

[93] R. Kaucic, B. Dalton, and A. Blake. Real-time lip tracking for audio-visual speech recognition applications. In B. Buxton and R. Cipolla, editors, *Proc. European Conference on Computer Vision*, volume II of *Lecture Notes in Computer Science*, pages 376–387, Cambridge, Apr. 1996. Springer-Verlag.

[94] R. Kaucic, D. Reynard, and A. Blake. Real-time lip trackers for use in audio-visual speech recognition. In *IEE Colloquium on Integrated Audio-Visual Processing*, 1996/213, pages 3/1–3/6, Savoy Place, London, Nov. 1996.

[95] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, Mar. 1998.

[96] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[97] P. B. Kricos. Differences in visual intelligibility across talkers. In Stork and Hennecke [181], pages 43–53.

[98] G. Krone, B. Talle, A. Wichert, and G. Palm. Neural architectures for sensorfusion in speechrecognition. In Benoît and Campbell [13], pages 57–60.

[99] W. J. Krzanowski. *Principles of Multivariate Analysis: A User's Perspective*. Oxford Statistical Science. Oxford University Press, 1996.

[100] P. K. Kuhl and A. N. Meltzoff. The bimodal perception of speech in infancy. *Science*, 218:1138–1141, Dec. 1982.

[101] P. Ladefoged. *A Course In Phonetics*. Harcourt Brace, third edition, 1993.

[102] C. H. Lee, J. S. Kim, and K. H. Park. Automatic human face location in a complex background using motion and color information. *Pattern Recognition*, 29(11):1877–1889, 1996.

[103] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An introduction to the application of the theory of probabilistic functions of a markov process to automatic speech recognition. *The Bell System Technical Journal*, 62(4):1035–1074, Apr. 1983.

[104] N. Li, S. Dettmer, and M. Shah. *Visually Recognizing Speech Using Eigensequences*, chapter 15, pages 345–371. Volume 9 of Shah and Jain [173], 1997.

[105] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic, 1994.

[106] J. Luettin. Towards speaker independent continuous speechreading. In *Proc. of the European Conference on Speech Communication and Technology*, 1997.

[107] J. Luettin. *Visual Speech and Speaker Recognition*. PhD thesis, University of Sheffield, May 1997.

[108] J. Luettin and S. Dupont. Continuous audio-visual speech recognition. In *Proc. European Conference on Computer Vision*, pages 657–673, June 1998.

[109] J. Luettin and N. A. Thacker. Speechreading using probabilistic models. *Computer Vision and Image Understanding*, 65(2):163–178, Feb. 1997.

[110] J. Luettin, N. A. Thacker, and S. W. Beet. Active shape models for visual feature extraction. In Stork and Hennecke [181], pages 383–390.

[111] J. Luettin, N. A. Thacker, and S. W. Beet. Learning to recognise talking faces. In *Proceedings of the International Conference on Pattern Recognition (ICPR'96)*, volume 4, pages 55–59. IAPR, 1996.

[112] J. Luettin, N. A. Thacker, and S. W. Beet. Locating and tracking facial speech features. In *Proceedings of the International Conference on Pattern Recognition (ICPR'96)*, volume 1, pages 652–656. IAPR, 1996.

[113] J. Luettin, N. A. Thacker, and S. W. Beet. Speaker identification by lipreading. In *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP'96)*, volume 1, pages 62–65, 1996.

[114] J. Luettin, N. A. Thacker, and S. W. Beet. Speechreading using shape and intensity information. In *Proceedings of the 4th International Conference on Spoken Language Processing (ICSLP'96)*, volume 1, pages 58–61, 1996.

[115] J. Luettin, N. A. Thacker, and S. W. Beet. Statistical lip modelling for visual speech recognition. In G. Ramponi, G. L. Sicuranza, S. Carrato, and S. Marsi, editors, *Signal Processing VIII Theories and Applications*, volume I, pages 137–140, Trieste, Sept. 1996.

[116] J. Luettin, N. A. Thacker, and S. W. Beet. Visual speech recognition using active shape models and hidden markov models. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 817–820, Atlanta, GA, May 1996. IEEE.

[117] J. MacDonald and H. McGurk. Visual influences on speech perception processes. *Perception and Psychophysics*, 24:253–257, 1978.

[118] P. C. Mahalanobis. On the generalised distance in statistics. *Proc. National Institute Science India*, 2:49–55, 1936.

[119] M. W. Mak and W. G. Allen. Lip-motion analysis for speech segmentation in noise. *Speech Communication*, 14:279–296, 1994.

[120] M. W. Mak and W. G. Allen. A lip-tracking system based on morphological processing and block matching techniques. *Signal Processing: Image Communication*, 6:335–348, 1994.

[121] K. Mase and A. Pentland. Automatic lipreading by optical-flow analysis. *Systems and Computers in Japan*, 22(6):67–75, 1991.

[122] D. W. Massaro. Speech perception by ear and eye. In Dodd and Campbell [61], pages 53–85.

[123] D. W. Massaro. Bimodal speech perception: A progress report. In Stork and Hennecke [181], pages 79–101.

[124] D. W. Massaro and M. M. Cohen. Evaluation and integration of visual and auditory information in speech perception. *Journal of Experimental Psychology: Human Perception and Performance*, 9(5):753–771, 1983.

[125] D. W. Massaro and D. G. Stork. Speech recognition and sensory integration. *American Scientist*, 86, May 1998.

[126] G. Matheron. *Random Sets and Integral Geometry*. Wiley, 1975.

[127] I. Matthews, J. A. Bangham, and S. Cox. Audiovisual speech recognition using multiscale nonlinear image decomposition. In *Proc. International Conference on Spoken Language Processing*, volume 1, pages 38–41, Philadelphia, PA, Oct. 1996.

[128] I. Matthews, J. A. Bangham, and S. Cox. Scale based features for audiovisual speech recognition. In *IEE Colloquium on Integrated Audio-Visual Processing*, 1996/213, pages 8/1–8/7, Savoy Place, London, Nov. 1996.

[129] I. Matthews, J. A. Bangham, R. Harvey, and S. Cox. A comparison of active shape model and scale decomposition based features for visual speech recognition. In *Proc. European Conference on Computer Vision*, pages 514–528, June 1998.

[130] I. Matthews, J. A. Bangham, R. Harvey, and S. Cox. Nonlinear scale decomposition based features for visual speech recognition. In *EUSIPCO'98*, page Accepted, 1998.

[131] I. Matthews, T. Cootes, S. Cox, R. Harvey, and J. A. Bangham. Lipreading from shape, shading and scale. In *Proc. Auditory-Visual Speech Processing*, page Accepted, Dec. 1998.

[132] I. Matthews, T. Cootes, S. Cox, R. Harvey, and J. A. Bangham. Lipreading from shape, shading and scale. In *Proc. Institute of Acoustics*, page Accepted, Nov. 1998.

[133] M. McGrath and Q. Summerfield. Intermodal timing relations and audio-visual speech recognition. *Journal of the Acoustical Society of America*, 77(2):678–685, Feb. 1985.

[134] M. McGrath, Q. Summerfield, and N. M. Brooke. Roles of lips and teeth in lipreading vowels. *Proc. Institute of Acoustics*, 6(4):401–408, 1984.

[135] H. McGurk and J. McDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, Dec. 1976.

[136] U. Meier, W. Hürst, and P. Duchnowski. Adaptive bimodal sensor fusion for automatic speechreading. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 833–836, Atlanta, GA, May 1996. IEEE.

[137] U. Meier, R. Stiefelhagen, and J. Yang. Preprocessing of visual speech under real world conditions. In Benoît and Campbell [13], pages 113–116.

[138] A. Mitiche and J. K. Aggarwal. Multiple sensor integration/fusion through image processing a review. *Optical Engineering*, 25(3):380–386, Mar. 1986.

[139] A. A. Montgomery and P. L. Jackson. Physical characteristics of the lips underlying vowel lipreading performance. *Journal of the Acoustical Society of America*, 73(6):2134–2144, June 1983.

[140] R. Moore. Recognition—the stochastic modelling approach. In C. Rowden, editor, *Speech Processing*, chapter 7, pages 223–255. McGraw-Hill, 1992.

[141] J. R. Movellan. Visual speech recognition with stochastic networks. In G. Tesauro, D. Toruetzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7. MIT Press, Cambridge, MA, 1995.

[142] J. R. Movellan and G. Chadderdon. Channel seperability in the audio visual integration of speech: A bayesian approach. In Stork and Hennecke [181], pages 473–487.

[143] J. R. Movellan and P. Mineiro. Modularity and catastrophic fusion: A bayesian approach with applications to audiovisual speech recognition. Technical Report 97.01, University of California, San Diego, Department of Cognitive Science, Jan. 1996.

[144] K. G. Munhall and E. Vatikiots-Bateson. The moving face during speech communication. In Campbell et al. [41], pages 123–139.

[145] K. K. Neely. Effect of visual factors on the intelligibility of speech. *Journal of the Acoustical Society of America*, 28(6):1275–1277, Nov. 1956.

[146] J. A. Nelder and R. Mead. A simplex method for function minimisation. *Computer Journal*, 7(4):308–313, 1965.

[147] E. H. Nitchie. *New Lessons in Lipreading.* J.B. Lippincutt, 1950.

[148] J. J. O'Neill. Contributions of the visual components of oral symbols to speech comprehension. *Journal of Speech and Hearing Disorders*, 19:429–439, 1954.

[149] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(7):629–639, July 1990.

[150] E. Petajan and H. P. Graf. Robust face feature analysis for automatic speechreading and character animation. In Stork and Hennecke [181], pages 425–436.

[151] E. D. Petajan. *Automatic Lipreading to Enhance Speech Recognition.* PhD thesis, University of Illinois, Urbana-Champaign, 1984.

[152] E. D. Petajan, B. J. Bischoff, D. A. Bodoff, and N. M. Brooke. An improved automatic lipreading system to enhance speech recognition. Technical Report TM 11251-871012-11, AT&T Bell Labs, Oct. 1987.

[153] C. Pirazzi et al. The pixel rosetta stone: Packings and colorspaces. `http://reality.sgi.com/~pirazzi`, 1998.

[154] I. Pitas and A. N. Venetsanopoulos. *Nonlinear Digital Filters: Principles and Applications.* Kluwer Academic, 1991.

[155] G. Potamianos, Cosatto, H. P. Graf, and D. B. Roe. Speaker independent audio-visual database for bimodal ASR. In Benoît and Campbell [13], pages 65–68.

[156] C. A. Poynton. *A Technical Inroduction to Digital Video.* John Wiley & Sons, Inc., 1996.

[157] K. V. Prasad, D. G. Stork, and G. J. Wolff. Preprocesing video images for neural learning of lipreading. Technical Report CRC-TR-9326, Ricoh California Research Center, 1993.

[158] W. H. Press, S. A. Teukosky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipies in C: The Art of Scientific Computing.* Cambridge University Press, second edition, 1995.

[159] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Signal Processing. Prentice Hall, 1993.

[160] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, Feb. 1989.

[161] R. R. Rao and R. M. Mesereau. Lip modelling for visual speech recognition. In *28th Annual Asilomar Conference on Signals, Systems, and Computers*, volume 1, pages 587–590, 1994.

[162] D. Reisberg, J. McLean, and A. Goldfield. Easy to hear but hard to understand: A lip-reading advantage with intact auditory stimuli. In Dodd and Campbell [61], pages 97–113.

[163] J. Robert-Ribes, M. Piquemal, J.-L. Schwartz, and P. Escudier. Exploiting sensor fusion architectures and stimuli complementarity in AV speech recognition. In Stork and Hennecke [181], pages 193–210.

[164] A. Rogozan, P. Deléglise, and M. Alissali. Adaptive determination of audio and visual weights for automatic speech recognition. In Benoît and Campbell [13], pages 61–64.

[165] L. D. Rosenblum and H. M. Saldaña. An audiovisual test of kinematic primitives for visual speech perception. *Journal of Experimental Psychology*, 22(2):318–331, 1996.

[166] L. D. Rosenblum and H. M. Saldaña. Time-varying information for visual speech perception. In Campbell et al. [41], pages 61–81.

[167] T. Rowntree. The intelligent aircraft. *IEE Review*, pages 23–27, Jan. 1993.

[168] M. U. R. Sánchez, J. Matas, and J. Kittler. Statistical chromaticity-based lip tracking with B-splines. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, Munich, Apr. 1997.

[169] J.-L. Schwartz, J. Robert-Ribes, and P. Escudier. Ten years after Summerfield: a taxonomy of models for audio-visual fusion in speech perception. In Campbell et al. [41], pages 85–108.

[170] J. Serra. *Image Analysis and Mathematical Morphology*, volume 1. Academic Press, 1982.

[171] J. Serra. Introduction to mathematical morphology. *Computer Vision, Graphics and Image Processing*, 35:283–305, 1986.

[172] J. Serra. *Image Analysis and Mathematical Morphology*, volume 2. Academic Press, 1988.

[173] M. Shah and R. Jain, editors. *Motion-Based Recognition*, volume 9 of *Computational Imaging and Vision*. Kluwer Academic, 1997.

[174] R. Sharma, V. I. Pavlović, and T. S. Huang. Toward multimodal human-computer interface. *Proc. IEEE*, 86(5):853–869, May 1998.

[175] P. L. Silsbee. *Computer Lipreading for Improved Accuracy in Automatic Speech Recognition*. PhD thesis, The University of Texas, Austin, Dec. 1993.

[176] P. L. Silsbee. Motion in deformable templates. In *Proc. IEEE International Conference on Image Processing*, volume 1, pages 323–327, 1994.

[177] P. L. Silsbee. Audiovisual sensory integration using hidden markov models. In Stork and Hennecke [181], pages 489–496.

[178] P. L. Silsbee. Computer lipreading for improved accuracy in automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4(5):337–351, Sept. 1996.

[179] P. L. Silsbee and A. C. Bovik. Medium vocabulary audiovisual speech recognition. In *New Advances and Trends in Speech Recogntion and Coding*, pages 13–16. NATO ASI, 1993.

[180] D. G. Stork, editor. *HAL's Legacy: 2001's Computer as Dream and Reality*. MIT Press, 1997.

[181] D. G. Stork and M. E. Hennecke, editors. *Speechreading by Humans and Machines: Models, Systems and Applications*, volume 150 of *NATO ASI Series F: Computer and Systems Sciences*. Springer-Verlag, Berlin, 1996.

[182] W. H. Sumby and I. Pollack. Visual contribution to speech intelligibility in noise. *Journal of the Acoustical Society of America*, 26(2):212–215, Mar. 1954.

[183] Q. Summerfield. Some preliminaries to a comprehensive account of audio-visual speech perception. In Dodd and Campbell [61], pages 3–51.

[184] Q. Summerfield. Lipreading and audio-visual speech perception. *Phil. Trans. R. Soc. Lond. B*, 335:71–78, 1992.

[185] Q. Summerfield and M. McGrath. Detection and resolution of audio-visual incompatibility in the perception of vowels. *Quarterly Journal of Experimental Psychology*, 36(A):51–74, 1984.

[186] M. J. Tomlinson, M. J. Russell, and N. M. Brooke. Integrating audio and visual information to provide highly robust speech recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 821–824, Atlanta, GA, May 1996. IEEE.

[187] J. Utley. A test of lip reading ability. *Journal of Speech Disorders*, 11:109–116, 1946.

[188] E. Vatikiotis-Bateson, K. G. Munhall, and M. Hirayama. The dynamics of audiovisual behavior in speech. In Stork and Hennecke [181], pages 221–232.

[189] E. Vatikiotis-Bateson, K. G. Munhall, Y. Kasahara, F. Garcia, and H. Yehia. Characterizing audiovisual information during speech. In *Proc. International Conference on Spoken Language Processing*, volume 3, pages 1485–1488, Philadelphia, PA, Oct. 1996.

[190] M. Vogt. Fast matching of a dynamic lip model to color video sequences under regular illumination conditions. In Stork and Hennecke [181], pages 399–407.

[191] M. Vogt. Interpreted multi-state lip models for audio-visual speech recognition. In Benoît and Campbell [13], pages 125–128.

[192] B. E. Walden, R. A. Prosek, A. A. Montgomery, C. K. Scherr, and C. J. Jones. Effects of training on the visual recognition of consonants. *Journal of Speech and Hearing Research*, 20:130–145, 1977.

[193] A. P. Witkin. Scale-space filtering. *Proc. 8th International Joint Conference on Artificial Intelligence*, 2:1019–1022, 1983.

[194] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland. *The HTK Book*. Cambridge University, 1996.

[195] B. P. Yuhas, M. H. Goldstein, Jr., and T. J. Sejnowski. Integration of acoustic and visual speech signals using neural networks. *IEEE Communications Magazine*, 27:65–71, 1989.

[196] B. P. Yuhas, M. H. Goldstein, Jr., T. J. Sejnowski, and R. E. Jenkins. Neural network models of sensory integration for improved vowel recognition. *Proceedings of the IEEE*, 78(10):1658–1667, Oct. 1990.

[197] A. L. Yuille, P. W. Hallinan, and D. S. Cohen. Feature extraction from faces using deformable templates. *International Journal of Computer Vision*, 8(2):99–111, 1992.