Audio-Video Automatic Speech Recognition: An Example of Improved Performance through Multimodal Sensor Input

Roland Goecke[1,2]

[1]Autonomous Systems and Sensing Technologies, National ICT Australia, Canberra, Australia,
[2]Australian National University, RSISE, Canberra, Australia
Email: roland.goecke@nicta.com.au

Abstract

One of the advantages of multimodal HCI technology is the performance improvement that can be gained over conventional single-modality technology by employing complementary sensors in different modalities. Such information is particular useful in practical, real-world applications where the application's performance must be robust against all kinds of noise. An example is the domain of automatic speech recognition (ASR). Traditionally, ASR systems only use acoustic information from the audio modality. In the presence of acoustic noise, the performance drops quickly. However, it can and has been shown that the incorporation of additional visual speech information from the video modality improves the performance significantly, so that AV ASR systems can be employed in applications areas where audio-only ASR systems would fail, thus opening new application areas for ASR technology. In this paper, a non-intrusive (no artificial markers), real-time 3D lip tracking system is presented as well as its application to AV ASR. The multivariate statistical analysis 'co-inertia' analysis is also shown, which offers improved numerical stability over other multivariate analyses even for small sample sizes.

Keywords: Audio-Video Speech Processing, 3D Stereo Lip Tracking

1 Introduction

It is widely accepted these days, that a key to robustness in HCI in many real-world environments is the use of multimodal input from various sensors, where the different modalities contain some redundancy in the information that is encoded in the signals. An example is the improved performance which can be achieved in automatic speech recognition (ASR) systems that use a combination of audio and video input, rather than only audio input. Audio-only ASR systems may work well in good acoustic conditions but can fail unpredictably in noisy conditions. Audio-Video (AV) ASR systems have been shown to improve the recognition performance considerably in such conditions (e.g. (Stork & Hennecke 1996, Potamianos, Neti, Gravier, Garg & Senior 2003)).

Although significant advances have been made in each of the two modalities in recent years, e.g. real-time audio-only speech processing and real-time face tracking, the robustness of the systems still needs to be improved to be employable in real-world applications. Much research has been done on audio-only speech processing, while the area of visual speech processing still lacks robust methods. For example, many research systems have used lip make-up (e.g. blue lipstick) or coloured dots on the lips and the face to simplify the lip tracking problem but it is obvious that such solutions are not practical in real-world applications. The first part of this paper presents a non-intrusive real-time 3D lip tracking algorithm based on a stereo camera system. Given the decreasing costs of cameras, stereo camera systems are feasible in many application areas these days.

A second issue addressed in this paper is the issue of relationships between variables across different modalities. Questions such as which variables or combination of variables in each modality are related to variables or combination of variables in the other modality arise. The machine learning approach assumes that, given enough training samples, the relationships can be learned from the data. Another approach is the statistical analysis of such relationships. Some multivariate methods, such as canonical correlation analysis, can suffer from collinearity in the sample data, leading to instabilities in the results. To overcome these problems, coinertia analysis (COIA) was developed by Dolédec and Chessel in 1994, in which the number of parameters relative to the sample size does not affect the accuracy and stability of the results (Dolédec & Chessel 1994). COIA is described in the second part of this paper together with some experimental results of applying both 3D lip tracking and COIA to AV ASR.

2 Related Work

2.1 Feature Extraction

Facial features must be extracted on which video speech parameters can be based. Two main streams of feature extraction can be identified: *implicit feature extraction* and *explicit feature extraction*.

2.1.1 Implicit Feature Extraction

A part of the image data which contains the mouth area is taken as is, and the pixel values are used as input of the recognition engine (e.g. HMM, artificial neural network). Thus, the recogniser learns the typical pixel patterns associated with certain lip movements. A principal component analysis (PCA) or linear discriminant analysis (LDA) can be employed to reduce the dimensionality of the input vector and

National ICT Australia (NICTA) is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council.

to define the main directions of variation. Only a few principal components are typically required to account for almost all variation. Examples of such implicit feature extraction systems are the ones by Meier *et al.* (Meier, Huerst & Duchnowski 1996, Meier, Stiefelhagen, Yang & Waibel 2000), Movellan and Chadderdon (Movellan & Chadderdon 1996), and Potamianos *et al.* (Potamianos, Verma, Neti, Iyengar & Basu 2000, Potamianos, Luettin & Neti 2001).

Implicit feature extraction avoids explicitly finding facial feature points and preserves both shape and appearance information (Scanlon & Reilly 2001). The disadvantages are that without a PCA or similar technique, the dimensionality of the input vector becomes very large (e.g. a 20×15 pixels window results in a vector with 300 elements!) and some effort must be made to compensate illumination changes (either by having a well-illuminated face or by using a normalised colour space at least). Most important of all, however, is the fact that the systems can only be trained for one specific angle of the face towards the camera and can thus not cope with a freely moving head, unless several recognisers were trained for different head poses and some sort of interpolation between these were used. Alternatively, some head pose compensation method based on image warping could be employed.

2.1.2 Explicit Feature Extraction

Here, image processing techniques are used to extract the position of mouth features, which are certain points on the lips (e.g. lip corners) as well as the internal and external lip contour line or the position of the teeth. Parameters describing the shape and the movements in the mouth region are then derived from the positions of these features in the image data. The effect of the overall head movement must be eliminated in the set of parameters to be extracted. Only components comprising mouth region movements are wanted. Parameter sets are often based on studies on what human perceivers appear to use for AV speech perception (see (Stork & Hennecke 1996)).

Image-based methods employ image processing techniques such as integral projection, thresholding, and edge detection to find relevant feature points (Yang, Stiefelhagen, Meier & Waibel 1998, Petajan & Graf 1996). Many of the problems with image-based methods arise from the lack of sufficient contrast in the mouth region. Moreover, the contrast of the lips to the surrounding skin is illumination-dependent, which led to the use of specially made-up lips and other artificial tracking aids, which are impractical to use in real-world applications.

Template matching algorithms are based on the cross-correlation of images which are taken as 2D functions (Russ 1995). Some part of an image — the template — is moved across the target image and the correlation values are calculated for each position. The position with the highest correlation value is the one with the highest degree of similarity. Noise as well as shape or pose differences in video sequences affect the matching process. Since the shape of the lips changes quickly and quite significantly while speaking, static image templates do not work well for the mouth region.

Deformable 2D models overcome these problems. Two prominent methods are active shape models (ASM) and active appearance models (AAM) (Cootes, Taylor, Cooper & Graham 1995, Cootes, Edwards & Taylor 1998). An ASM can only deform in ways characteristic to the class of objects it represents. These characteristics are learned from a set of training images and stored in a point distribution model. Whereas deformable templates and active contour models align to strong gradients for locating the object, ASMs learn the typical shape deformation and use it during feature search. Any normalised lip shape can be approximated using the learned mean shape and the first few principal modes of variation. AAMs are an extension of ASMs and combine the shape model with a statistical model of the grey-values in the region around each point of the ASM. By iteratively minimising the difference of the grey-values of the model and the image, the parameters of the shape model can be updated to fit the model better to the lip shape.

Finally, fully model-based approaches fit a 3D lip model to the image (Basu, Oliver & Pentland 1998). The 3D lip model consists of a 3D surface or volume model which is backprojected into the 2D image space. The best fit of the model mouth shape to the mouth shape in the image is found by adjusting the model parameters in an optimisation process. It is possible to combine this approach with a training phase in which possible mouth shapes are learned from labelled training data, so as to constrain the optimisation process to permissable lip shapes.

3 Face Tracking

The 3D lip tracking algorithm builds on a non-intrusive real-time stereo vision face tracking system (Newman, Matsumoto, Rougeaux & Zelinsky 2000). No facial markers or special make-up are required, but the system still achieves a high degree of accuracy. Such properties are desirable in an AV ASR system because artificial tracking aids pose the risk of inhibiting the speaker from speaking naturally. A calibrated stereo vision system has the advantage that depth information (distance from cameras to object) can be recovered from the stereo disparity. A calibrated monocular camera system can only estimate depth — or the object's 3D position in general — if the object dimensions and its orientation (pose) are known, which is obviously not the case in unrestricted face tracking. Applied to face and lip tracking, stereo vision has the advantage that 3D coordinates of (e.g. lip) feature points can be measured accurately irrespective of the head pose. For correct lip tracking results, it is important to separate general head movements from lip feature point movements.

The head tracking system is based on template matching using normalised cross-correlation and is able to track a person's movements at video frame rate. The system consists of two calibrated standard, colour analog NTSC video cameras. The camera outputs are multiplexed at half the vertical resolution into a single 512x480 image (Figures 1 and 2). Details can be found in (Newman et al. 2000). The multiplexing leads to a loss of information on the vertical axis but the advantage is that stereo image processing can be performed on any standard PC without the need for special hardware.

The 3D lip tracking algorithm is applied to the mouth regions in each camera's image which are automatically determined during the head tracking based on the head pose (Figure 2). The algorithm combines colour information from the images with knowledge about the structure of the mouth region for different degrees of mouth openness. Many other lip tracking algorithms do not take advantage of such *a priori* knowledge, even though it can improve the performance. For example, in an open mouth, one often expects to see teeth, so why not specifically look for them?

Measuring the 3D coordinates of certain feature points on the inner lip contour leads to a variety of parameters describing the shape of the lips in 3D.

From just 4 feature points - the lip corners as well as the midpoints of upper and lower lip - 3D measures such as mouth width, mouth height, and lip protrusion can easily be determined. The inner lip contour was preferred over the outer lip contour for a number of reasons. Firstly, people differ in the generic shape of their lips. Some people have thicker lips than others, some have stronger pro-trusion (in the rest state) than others. Extracting the outer lip contour would mean that such personal characteristics influence the measurements, while the inner lip contour can truly be considered as the final boundary of the vocal tract. Hence, inner lip contour measurements are better suited for the investigation of relationships between audio and video speech parameters. Secondly, the difference between lip colour and the surrounding facial skin can be quite small. Many lip tracking methods have difficulty in coping with this lack of contrast, if employed on tracking the outer lip contour. Furthermore, facial hair affects the visibility of the outer lip contour. Given the different appearance of the oral cavity, the inner lip contour does not suffer from these problems.

4 3D Lip Tracking

4.1 Overview

The requirements of a lip tracking algorithm generally depend on the application. In the case of AV speech processing, an algorithm that is both fast and accurate is needed. Lip movements during speech production can be very quick and changes in mouth shape (mouth closed, mouth partially open, mouth wide open, lips rounded, lips spread etc.) can take place in a time span as short as 10ms. This highlights the need for a real-time algorithm which tracks the lip movements continuously. At the same time, accuracy is of great importance for the results of the statistical analysis described in 5 to have any meaningful value. It is particularly important to distinguish apparent distortions in mouth shape due to head pose (rotation) from speech production-related mouth deformations. Furthermore, the algorithm must be able to cope with different personally characteristic lip shapes as well as various mouth shapes ranging from a completely closed mouth to a fully open mouth in which upper and lower teeth as well as the tongue may or may not be visible (Figure 1 bottom). Finally, a lip tracking algorithm must take the level of illumination in the lower face half into account.

As discussed in Section 2, lip tracking can be either implicit or explicit. Implicit lip tracking analyses the statistical behaviour of feature vectors representing the pixels of the mouth area. Explicit lip tracking, on the other hand, attempts to fit a 2D or 3D lip model to the observations by locating facial feature points that define the model. Such an explicit approach was followed in this work because it was expected to facilitate the interpretation of the analysis of the AV relationships.

4.1.1 Extracting the Mouth Region

The lip tracking algorithm assumes that the face has been located in the video stream from the two cameras and that this face is being tracked. The face tracking system estimates the head pose and from it, the locations of eye and mouth corners according to the 3D face model.

The mouth window is a rectangular part of the image containing the lips, oral cavity, and some of the surrounding facial skin as shown in Figure 2. To account for differences in the cameras, the image data

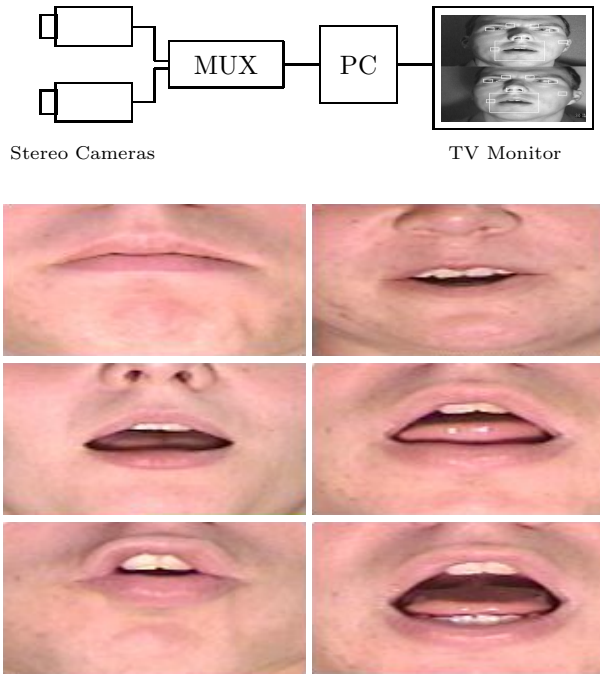

Figure 1: Top: Outline of the combined stereo vision face and lip tracking system. Bottom: Different degrees of mouth openness as well as teeth and tongue visibility.

in the mouth windows are mean-normalised. The orientation of the mouth follows the general head pose. In a general lip tracking algorithm, the angle of rotation around the z axis (deviation from the horizontal xy plane) ϕ should be considered when the mouth window is extracted. An image processing technique such as warping can then be applied to the mouth window to realign the mouth horizontally. If the speaker's head is approximately in upright position ($\pm 20°$), then the influence of ϕ can be neglected and the axes of the mouth window rectangle can be aligned with the image coordinate axes. Lip tracking is performed on the mouth windows rather than the entire image, which reduces the amount of processing that is required and limits the error due to a lip tracking failure.

4.1.2 Combining Colour and Structure

The lip tracking algorithm has been designed to find the coordinates of the four lip feature points necessary to define the parameter set of our lip model, i.e. the two lip corners plus the midpoints of upper and lower lip. The algorithm combines colour information from the images with knowledge about the structure of the mouth area (Goecke 2005). Colour information is a powerful cue in facial feature detection. However, the YUV (also known as YIQ) signal from the NTSC cameras alone is of little use, because the image signal is encoded into an intensity (Y) signal and two colour difference signals (U, V). The YUV signal can be transformed into a standard computer RGB signal. However, images in the RGB colour space are affected by changes in illumination. A better choice is the HSV colour space which separates hue (H) and saturation (S) from intensity (V).

There is a clear difference in the saturation values between skin/lips, teeth, and oral cavity. The dark oral cavity exhibits the largest saturation values and the teeth the smallest, while the skin values lie between these two extremes. A combination of intensity (Y) and saturation (S) values is, therefore, used

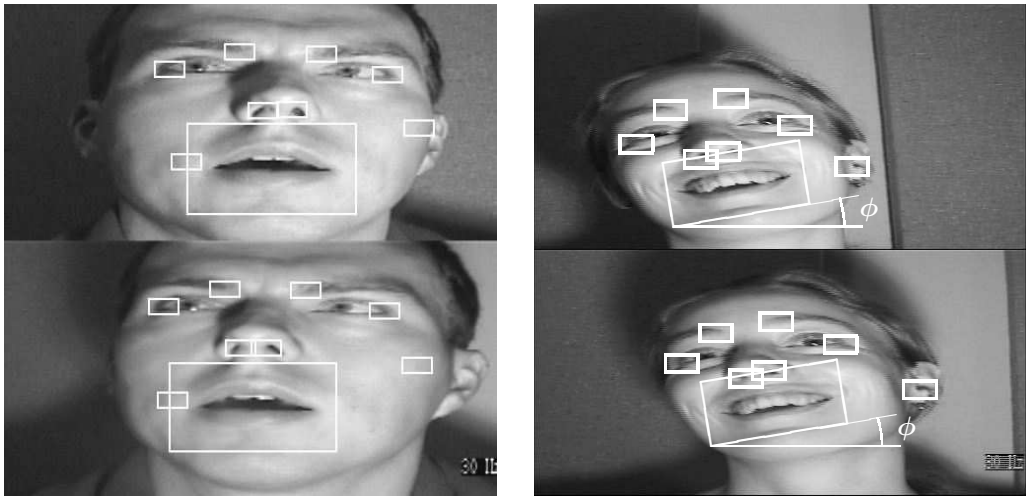

Figure 2: Extracting the mouth region: Large rectangles enclose automatically selected mouth windows.

throughout the algorithm.

The lip tracking algorithm is a three-stage process outlined in the following paragraphs and described in detail in Sections 4.3 – 4.5. The first step determines the general degree of mouth openness. The lip tracking algorithm must be able to handle mouth shapes ranging from a completely closed mouth to a wide open mouth. No single image processing technique is likely to give good results for all possible mouth shapes. However, by pre-classifying mouth shapes into one of three categories based on mouth openness, specific techniques individually targeted at each category can then be applied to give better results. These categories are

- closed mouth,
- partially open mouth, and
- wide open mouth.

In the second step, the lip corners are found. Here, the *a priori* knowledge about the structure of the mouth area becomes useful. For example, if the mouth is closed, teeth will not be visible, so the shadow line between upper and lower lip is the outstanding feature. Various definitions of what constitutes the inner lip contour of a closed mouth are possible. In this study, the shadow line between the lips was considered to be part of the inner lip contour. Therefore, the algorithm looks for this line. When the mouth is open, it is very likely that either or both the upper and lower teeth are visible, so the algorithm looks for them as well as for the oral cavity. By tailoring the algorithm in this way to fit a particular situation, more accurate results can be obtained than from a general-purpose, 'one-size-fits-all' algorithm. The first and second steps are applied separately to both the left and right camera images. Once the 2D image positions of the lip corners in both views are known, their 3D positions can be calculated. This result is then used in the third and final step, in which the positions of the lip midpoints are determined.

The face tracking and lip tracking systems together run at video frame rate on an average PC (Pentium IV, 3.0GHz, 1GB RAM). Alternatively, offline processing of recorded sequences, as was done in the experiments described in Section 6, avoids any limitations set by the hardware and software, and is then only limited by the properties of the recording equipment. In this study, the limiting factor was the NTSC frame rate of 30Hz determined by the analogue cameras used. That means that one video frame was taken every 33ms, which captured a lot of detail of lip movements, but information about faster lip motion was lost.

4.2 Algorithm Techniques

Various techniques are used several times in the lip tracking algorithm and are, therefore, discussed here. Firstly, there is *dynamic thresholding*. Whenever a threshold operation takes place, the threshold is determined dynamically at that time, instead of using hard-coded threshold values, to improve robustness. That is, the starting value of the threshold is chosen to be overly conservative, so that no pixel value in the area of interest will pass it. The threshold is then iteratively changed until the value has been found, at which pixel values start passing the threshold. The algorithm then continues to use this threshold value for the rest of the processing of that frame. In this way, the algorithm adapts itself to changes in illumination and different skin tones.

Secondly, whenever a particular pixel position is tested, not only the pixel value of that position is checked, but also of up to n (empirically set to 9) other pixels in the neighbourhood (but some pixel positions away). Selection is done by pixel masks like this one

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & X & 0 & 1 & 0 & 1 \end{bmatrix} \qquad (1)$$

where X indicates the current pixel position, a 1 relates to other checked pixel positions and a 0 to pixel positions ignored in the test. *Voting* takes place for each such pixel X, and only when a certain number of positive votes is reached, does it indicate that a threshold has been passed and that position is accepted as being correct. Voting turns weak cues into strong ones. It can be either 'hard' or 'soft'. The former requires a majority of votes, for example two thirds, and is used for finding lip contours. The latter only requires a few positive votes. It indicates the presence of a particular feature. Soft voting is only used for detecting visible teeth.

Thirdly, two modules which are used numerous times in the algorithm, test for the *shadow line* between the lips and for the *visibility of teeth* in the image data. The shadow line is detected by the typical high saturation and low intensity values. Teeth can be distinguished from other parts of the face by their characteristic low saturation and high intensity values. However, since some skin parts can show similar values, the teeth check must also pass an edge detection test, which looks for the horizontal edge between

the lip and the teeth. A 3×3 vertical gradient filter is applied

$$K = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad . \tag{2}$$

Finally, *integral projection* is a technique, in which all the pixel values of one row or column of pixels are summed. This technique can be extremely effective in locating facial features, if the rectangular image part, on which it is performed, is chosen suitably, as was shown in Kanade's pioneering work (Kanade 1973). Let $I(x, y)$ denote the pixel value, e.g. intensity or saturation, at the coordinates (x, y). Horizontal and vertical integral projection are then respectively defined as

$$H(y) = \sum_{x=x_1}^{x_2} I(x, y) \tag{3}$$

$$V(x) = \sum_{y=y_1}^{y_2} I(x, y) \tag{4}$$

where x_1, x_2, y_1, y_2 denote the boundary coordinates of the image part under investigation. Horizontal integral projection is useful for detecting vertical gradients and vertical integral projection similarly for horizontal gradients.

4.3 Step 1: Determine Mouth Openness

To determine the degree of mouth openness, the vertical positions of the lip midpoints must be determined. For the following calculations, the horizontal position of the lip midpoints is (temporarily) considered to be at the middle between the left and right boundaries of the mouth window. This estimate is close enough to the true position to start the algorithm for finding the lip corners (Step 2), but the position is recalculated (Step 3) after the lip corner positions are found.

Horizontal integral projection on the intensity values of the mouth window pixels is used to find a starting estimate of the vertical positions of the lip midpoints (Figure 3 top). These sometimes rough estimates need to be refined. If the mouth is closed, either the shadow line between the lips (correct) or the external lip contour (incorrect) is found. If the mouth is open, either or both the upper and lower teeth are visible and the horizontal integral projection detects either the edge between lip flesh and teeth (correct), or between teeth and oral cavity (incorrect), or in rare cases the outer lip contour (incorrect).

Let us first look at correcting the midpoint of the lower lip. To test if a correction is necessary, an imaginary vertical line from the lower boundary of the mouth window to the midpoint of the upper lip at the estimated horizontal position is followed upwards (Figure 3 bottom left). Necessarily, the lower lip midpoint cannot lie above the upper lip midpoint. While walking along the line, the algorithm checks for either the shadow line between the lips or for the appearance of teeth. If either is found and the position is different from the one obtained from the previous horizontal integral projection, the position is updated. This vertical position might be just off the lip contour, so in a final step, the algorithm adjusts the vertical position of the lower lip midpoint to the lip pixel bordering the oral cavity.

Secondly, the vertical position of the midpoint of the upper lip is corrected, if necessary. The algorithm first tests for the appearance of teeth above the position of the upper lip midpoint estimated from the horizontal integral projection. If found, it means that the edge between oral cavity and teeth was detected and, hence, the position is moved upwards until no further teeth pixels are detected above the current position. Subsequently, a second test for teeth, this time below the current position, is performed. If the edge between teeth and upper lip was found correctly by either of the steps before, there are teeth pixels below the current position. However, if the outer lip contour was detected, no teeth pixels are found below. In that case, the algorithm starts just above the lower lip midpoint and moves upwards until the edge between teeth and upper lip is found (Figure 3 bottom right). Again, the vertical position of the upper lip midpoint is finally accurately placed on the lip pixel forming the edge to the oral cavity. If neither of these tests indicates any necessary changes in the lip midpoint positions, then the coordinates found in the horizontal integral projection step are retained.

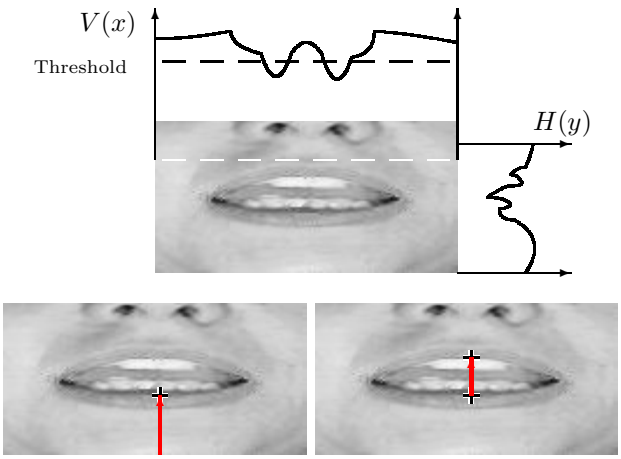

Figure 3: Step 1 - Top: Nostril detection by vertical integral projection in the top quarter of the mouth window. Horizontal integral projection to find vertical position of lip midpoints. Bottom: Possible correction of lower and upper lip midpoints.

4.3.1 Nostril Detection

The size of the mouth window is chosen to be sufficiently large to contain the mouth area under all circumstances. As a result, the nostrils are sometimes included in the mouth window (Figure 1). Since intensity values are used in the horizontal integral projection step, the nostrils' low intensity values potentially lead to incorrect results for the position of the lip midpoints. Therefore, a *nostril detection* algorithm was also developed. The top quarter of the mouth window is scanned for the minimum and maximum pixel column using vertical integral projection on the intensity values. A threshold to determine nostril candidates is set by

$$T = \text{Min} + \frac{\text{Max} - \text{Min}}{3} \quad . \tag{5}$$

The horizontal position of nostril candidates is determined by testing the pixel column sums with this threshold. Values below the threshold are candidates, but they are only confirmed as horizontal nostril positions, if they extend at least 3 pixels wide horizontally. The lower edge of the nostril is then found using edge detection and the vertical position closest to the upper lip ('lowest' nostril position in the image) is taken as the new upper boundary of the mouth window used for determining the mouth openness described before.

4.4 Step 2: Find Lip Corners

So far, the vertical positions of the lip midpoints have been established at their temporary horizontal positions. The 2D distance between the lip midpoints defines the following steps in the algorithm. If the distance is less than 15 pixels, the mouth is either fully closed or only partially open. Otherwise, the mouth is considered to be wide open. The threshold of 15 pixels is an experimentally determined heuristic. It is equivalent to about 15mm in 3D space for an object at a distance of about 600mm.

4.4.1 Mouth Closed or Partially Open

If the mouth is fully closed or only partially open, a vertical integral projection would not yield enough information to find the lip corners reliably. Thus, starting from the current position of the lower lip midpoint, a search along the shadow line to either side is performed through a cycle of tests (Figure 4 top left).

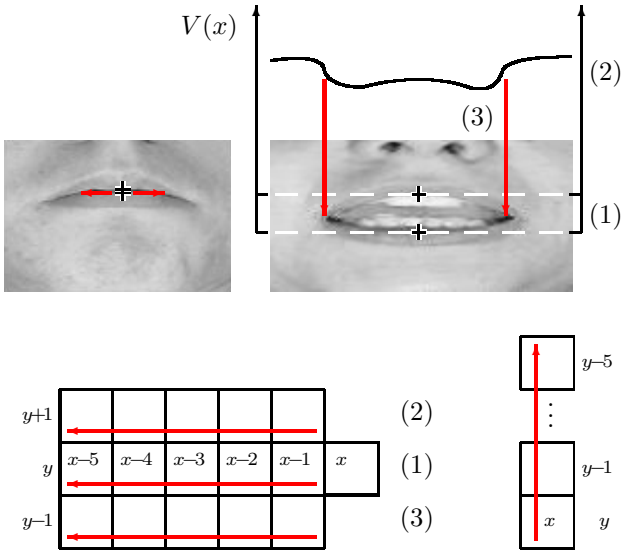

Figure 4: Step 2 - Top left: Moving along the shadow line for closed or partially open mouth. Top right: Vertical integral projection for wide open mouth. Bottom left: Checking for discontinuities in the shadow line. Bottom right: Testing for shadow line pixels above current position.

Let us consider the speaker's right lip corner, noting that the speaker's left lip corner is found similarly, except for moving in the opposite direction. Starting from the midpoint (x, y) of the lower lip, the algorithm moves left in image space. This line of pixels is marked (1) in Figure 4, bottom left. Testing the five pixel positions $(x-1, y), \ldots, (x-5, y)$ enables the algorithm to jump over pixels, where the shadow line is discontinuous due to image noise. If one of the pixels indicates the continuation of the shadow line, the current position is moved to $(x-1, y)$. If not, the test is repeated first for $(x-1, y+1), \ldots, (x-5, y+1)$ (pixel line (2) in Figure 4, bottom left) and then for $(x-1, y-1), \ldots, (x-5, y-1)$ (pixel line (3) in Figure 4, bottom left). For positive tests, the current position is moved to $(x-1, y+1)$ and $(x-1, y-1)$, respectively. The reason for testing different vertical positions y is that the inner lip contour, which corresponds to the shadow line, of the lower lip is not necessarily a straight line but can be curved up or down, depending on the generic lip shape of the speaker and the mouth shape during speech production. If there are no more shadow line pixels ahead, the algorithm tests the five pixels $(x, y-1), \ldots, (x, y-5)$ above the current position (x, y) (Figure 4, bottom right). If a shadow line pixel is found, the current position is moved to $(x, y-1)$ and the test cycle is repeated. Otherwise, the lip corner has been found.

In rare cases, the shadow line is discontinuous for more than five pixels. Therefore, the found lip corner positions are checked to be at least 25 pixels away from the midpoint of the lower lip. Otherwise, the search along the shadow line is restarted from a point 25 pixels away from the lip midpoint. For an object at a distance of 600mm from the camera, 25 pixels are equivalent to 10–12mm in Euclidean space. A distance of at least 10mm to either side of the lip midpoint, or at least 20mm mouth width in total, has been found to be a reasonable lower bound for any mouth shape experienced during speech production.

4.4.2 Mouth Wide Open

If the mouth is wide open, vertical integral projection on the intensity values of the image gives reliable estimates of the horizontal positions of the lip corners (Figure 4 top right). The vertical positions of the midpoints of upper and lower lip (1), determined in Step 1, define the vertical range for the integral projection (2). The largest changes in the resulting values determine the horizontal positions of the lip corners (3). Once these have been found approximately, a search along the (vertical) pixel columns through these horizontal positions looks for the pixels with the lowest intensity value and the highest saturation value which corresponds to the internal lip contour in the lip corner. The resulting pixel positions from the intensity and saturation searches are averaged to yield an estimate of the vertical position, which makes the algorithm more robust against misleading pixel values. Given that the accuracy of the results from both searches is unknown, averaging offers a way of most likely reducing any error. Finally, the found positions are refined by using the search technique along the shadow line described above for the closed or partially open mouth, but with the current estimated positions as starting points.

4.5 Step 3: Find Lip Midpoints

Now that the lip corner coordinates are established, the horizontal position of the lip midpoints, which has so far simply been the middle between left and right boundaries of the mouth window, needs to be recalculated. From the 2D image coordinates of the lip corners in the left and right stereo images, their 3D coordinates $\vec{l}$ and $\vec{r}$ are calculated using the known camera parameters. Based on these 3D coordinates, the centre point $\vec{c}$ between these two points is computed as

$$\vec{c} = \frac{\vec{l} + \vec{r}}{2} \quad . \tag{6}$$

Since the lip corner coordinates could be wrong, a linear combination of the previous midpoint estimates and the newly computed centre point $\vec{c}$ is used to determine the likely centre point $\vec{c}\,'$

$$\vec{c}\,'[k] \;=\; \alpha\,\vec{c}[k] \;+\; (1-\alpha)\,\vec{c}\,'[k-1] \tag{7}$$

where k is the frame number. The linear factor α is a confidence measure for the current frame. Information about the head pose from the general face tracker is used to define a normal vector perpendicular to an imaginary face plane and pointing away from the face. Then, the likely centre point $\vec{c}\,'$ is moved 5mm along this vector (Figure 5). An analysis of test video data had shown that the lip midpoints protrude

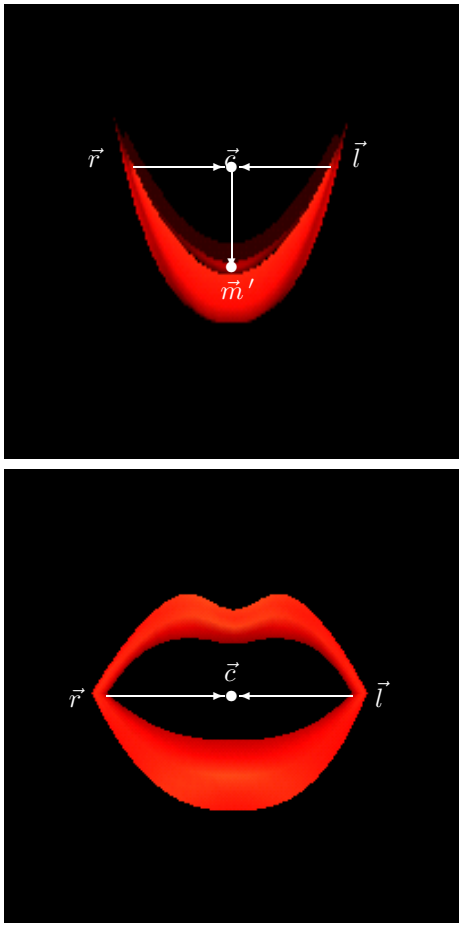

Figure 5: Step 3 - Finding the horizontal position of the lip midpoints (viewed from front and above).

about 5mm more than the lip corners. The likely lip midpoint $\vec{m}\,'$ is then back projected into image space and the x coordinate of that point taken as the horizontal position of the lip midpoints in each of the two images. After this, small adjustments to the vertical positions determined in Step 1 are likely and can be made in the same way, as when finding the exact lip contour at the end of Step 1. Finally, the 2D coordinates of the lip midpoints in the stereo image pair are combined to give their respective 3D coordinates.

5 Co-Inertia Analysis

Coinertia analysis (COIA) is a relatively new multivariate statistical analysis for coupling two (or more) parameter sets by investigating linear combinations of these. The term 'inertia' is used as a synonym for variability. The method is related to other multivariate analyses such as canonical correspondence analysis (CCA), redundancy analysis, and canonical correlation analysis (CANCOR) (Gittins 1985). COIA can also be coupled easily with other statistical methods, e.g. principal component analysis. First, these methods are performed on the data of the two domains separately, and then a COIA follows. In fact, it can be shown that COIA is a generalisation of many multivariate methods (Dray, Chessel & Thioulouse 2003).

COIA is very similar to CCA and CANCOR. The term 'inertia' is used as a synonym for variance. COIA also rotates the data into a new coordinate system and the new variables are linear combinations of the variables in each set. Where CANCOR maximises the correlation between the two sets, the square covariance is maximised in COIA

$$cov(A,V) \;=\; corr(A,V) * \sqrt{var(A)} * \sqrt{var(V)} \quad . \tag{8}$$

COIA finds a mathematical compromise between the correlation $corr(A,V)$, the variance in the audio set $var(A)$, and the variance in the video set $var(V)$. COIA can also be seen as aiming to find orthogonal vectors — the coinertia axes — in the two sets which maximise the coinertia value. The number of axes is equivalent to the rank of the covariance matrix.

The advantage of COIA is its numerical stability. The number of parameters relative to the sample size does not affect the accuracy and stability of the results (Dolédec & Chessel 1994). The results of the method do not suffer in the presence of collinearity and the consistency between the correlation and the coefficients is very good (Dray et al. 2003), which makes it a well-suited method for studying relationships in multimodal data even for small sample sizes.

COIA provides a number of measures for the analysis of the relationships between two sets of variables, such as the co-inertia value, the ratio of variance projected onto the new axes to the overall variance, the weights of the linear combinations and the RV coefficient. For details, see (Dolédec & Chessel 1994, Dray et al. 2003). In the experiments in the next section, the co-inertia value is used, which is a global measure of the co-structure in the two sets.

6 Experiments

In this section, the 3D lip tracking algorithm and the COIA shall be employed in an AV ASR experiment. Data from the AVOZES data corpus is used (Goecke & Millar 2004). A subset of AVOZES is used which consists of the 10 female speakers and the CVC-word utterances, which cover the 18 vocalic phonemes of Australian English.

As a baseline, an audio-only ASR system is used. Using the HTK toolkit, a 3-state left-right HMM with no skips is built for each monophone. 13 MFCC parameters and their delta and delta-delta parameters were used as audio speech variables. From the monophone HMMs, context-dependent triphone HMMs are built by simply cloning the monophone HMMs and re-estimating them using triphone transcriptions. In the experiments, the leave-one-out method is used. For each of the speakers, the recogniser is trained with data from the other nine speakers and then tested on the left-out speaker's data. The word error rate (WER) results are shown in Table 1. The relatively high WER for some speakers can be explained by the limited training data. Confusions typically occurred between short and long realisations of the same sound (e.g. /æ/ vs. /ɜ:/), and among low to mid-low front to central vocalic phonemes (e.g. /ɛ/ vs. /æ/). In the joint AV recognizer, the mouth width, mouth height, protrusion of upper lip, protrusion of lower lip, and relative teeth count [9] are added as visual speech variables, including their delta and delta-delta variables. A feature fusion approach is taken here, i.e. the values of the video speech variables are added to the audio feature vectors. Then, the HMMs are retrained and again the recognizer is tested using the approach as for the audio-only case. The results are shown in Table 1. The inclusion of visual speech information improves the WER by about 20-30% relative.

Adding the co-inertia value from COIA to the recogniser leads to a further improvement by about 1.5-2% relative on average. These preliminary results need to be further tested to investigate, if the addition of more values from COIA further improves the recognition rate. In particular, adding the coefficients

Speaker	Audio-Only	AV	AV + COIA
f1	27.8	22.0	21.5
f2	27.8	20.3	19.8
f3	27.8	21.3	20.6
f4	33.3	23.4	22.8
f5	11.1	7.8	7.5
f6	16.7	13.6	13.2
f7	11.1	8.1	7.9
f8	33.3	23.9	23.2
f9	38.9	28.3	27.9
f10	33.3	25.0	24.6

Table 1: Experimental results: WER in % for the audio-only case, the audio+video (AV) case, and for the AV plus co-inertia results case

of the linear combinations of variables is expected to have a positive effect, as these coefficients appear to be phoneme-specific and hence would be very useful to distinguish them.

7 Conclusions

This paper has focussed on two important subareas of AV ASR as an example of how multimodal sensor input can improve the performance. A non-intrusive real-time 3D lip tracking algorithm has been presented which brings us a step closer to using such technology in real-world applications, where it is impractical to rely on artificial tracking aids. Future work will test the algorithm further for robustness to visual noise as it is present in real-world applications. Secondly, this paper presented the multivariate statistical analysis COIA as a useful tool for the analysis of linear relationships between sets of variables. Its main advantage is the numerical stability even for small sample sizes. Future work will investigate further measures from COIA for AV ASR systems.

References

Basu, S., Oliver, N. & Pentland, A. (1998), '3d lip shapes from video: A combined physical-statistical model', *Speech Communication* **26**(1–2), 131–148.

Cootes, T., Edwards, G. & Taylor, C. (1998), Active Appearance Models, *in* 'Proc. ECCV'98', Vol. 2, Freiburg, Germany, pp. 484–498.

Cootes, T., Taylor, C., Cooper, D. & Graham, J. (1995), 'Active shape models - their training and applications', *Computer Vision and Image Understanding* **61**(1), 38–59.

Dolédec, S. & Chessel, D. (1994), 'Co-inertia analysis: an alternative method for studying species-environment relationships', *Freshwater Biology* **31**, 277–294.

Dray, S., Chessel, D. & Thioulouse, J. (2003), 'Co-inertia analysis amd the linking of ecological tables', *Ecology* . (in press).

Gittins, R. (1985), *Canonical Analysis*, Springer-Verlag, Berlin, Germany.

Goecke, R. (2005), 3D Lip Tracking and Co-inertia Analysis for Improved Robustness of Audio-Video Automatic Speech Recognition, *in* 'Proceedings of the Auditory-Visual Speech Processing Workshop AVSP 2005', Vancouver Island, Canada, pp. 109–114.

Goecke, R. & Millar, J. (2004), A Detailed Description of the AVOZES Data Corpus, *in* 'Proc. 10th Austral. Int. Conf. Speech Science & Technology 2004', Sydney, Australia, pp. 486–491.

Kanade, T. (1973), Picture Processing System by Computer Complex and Recognition of Human Faces, PhD thesis, Kyoto University, Japan.

Meier, U., Huerst, W. & Duchnowski, P. (1996), Adaptive Bimodal Sensor Fusion For Automatic Speechreading, *in* 'Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP'96', Atlanta (GA), USA, pp. 833–836.

Meier, U., Stiefelhagen, R., Yang, J. & Waibel, A. (2000), 'Towards Unrestricted Lipreading', *International Journal of Pattern Recognition and Artificial Intelligence* **14**(5), 571–585.

Movellan, J. & Chadderdon, G. (1996), Channel Separability in the Audio-Visual Integration of Speech: A Bayesian Approach, *in* D. Stork & M. Hennecke, eds, 'Speechreading by Humans and Machines', Vol. 150 of *NATO ASI Series*, Springer-Verlag, Berlin, Germany, pp. 473–487.

Newman, R., Matsumoto, Y., Rougeaux, S. & Zelinsky, A. (2000), Real-Time Stereo Tracking for Head Pose and Gaze Estimation, *in* 'Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition FG'2000', Grenoble, France, pp. 122–128.

Petajan, E. & Graf, H. (1996), Robust Face Feature Analysis for Automatic Speachreading (sic) and Character Animation, *in* D. Stork & M. Hennecke, eds, 'Speechreading by Humans and Machines', Vol. 150 of *NATO ASI Series*, Springer-Verlag, Berlin, Germany, pp. 425–436.

Potamianos, G., Luettin, J. & Neti, C. (2001), Hierarchical Discriminant Features for Audio-Visual LVCSR, *in* 'Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP'01', Vol. 1, IEEE, Salt Lake City (UT), USA, pp. 165–168.

Potamianos, G., Neti, C., Gravier, G., Garg, A. & Senior, A. (2003), 'Recent Advances in the Automatic Recognition of Audiovisual Speech', *Proceedings of the IEEE* **91**(9), 1306–1326.

Potamianos, G., Verma, A., Neti, C., Iyengar, G. & Basu, S. (2000), A Cascade Image Transform for Speaker Independent Automatic Speechreading, *in* 'Proceedings of the 2000 IEEE International Conference on Multimedia and Expo', Vol. 2, IEEE, New York (NY), USA, pp. 1097–1100.

Russ, J. (1995), *The Image Processing Handbook*, 2nd edn, CRC Press, Boca Raton (FL), USA.

Scanlon, P. & Reilly, R. (2001), Feature Analysis for Automatic Speechreading, *in* J.-L. Dugelay & K. Rose, eds, 'Proceedings of the 2001 IEEE Fourth Workshop on Multimedia Signal Processing', IEEE, Cannes, France, pp. 625–630.

Stork, D. & Hennecke, M., eds (1996), *Speechreading by Humans and Machines*, Vol. 150 of *NATO ASI Series*, Springer-Verlag, Berlin, Germany.

Yang, J., Stiefelhagen, R., Meier, U. & Waibel, A. (1998), Real-time Face and Facial Feature Tracking and Applications, *in* 'Proc. Int. Conf. Auditory-Visual Speech Processing AVSP'98', Terrigal, Australia, pp. 79–84.