DI 2 -Progress Report

A Report of the Esprit Project 8579 MIAMI
– WP 2 –

April, 1995

Written by
L. Schomaker, J. Nijtmans (NICI)
A. Camurri, F. Lavagetto, P. Morasso (DIST)
C. Benoît, T. Guiard-Marigny, B. Le Goff,
J. Robert-Ribes, A. Adjoudani (ICP)
I. Defée (RIIT)
S. Münch (UKA)
K. Hartung, J. Blauert (RUB)

Contents

Chapter 1

WP 2 – Integrating Modalities: Experiments testing human multimodal processing

In this Workpart, a set of experiments will be carried out on human subjects. The aims of the experiments are:

(a) checking and obtaining insight about the multimodal performance;

(b) development of detailed models of processing.

While in the fields of perception research and experimental psychology, many similar experiments are studied, our approach is oriented towards practical technical solutions and is integral with application scenarios described later. To deal with the complexity bottlenecks, minimized information transfer will be used. The experiments will be realized for several different aspects of processing:

a) Spatiotemporal representation and integration in HIP input subsystems

An important topic is how presentations of different modalities are combined with regard to spatiotemporal representation and integration. This can for example give information on how to attract and manipulate attention in multimedia representation. The starting point of our investigations will be the combination of visual and acoustic modalities. A lot of psychophysical experiments on this topic have been performed yet and are available from the literature. However, those investigations generally did not aim at finding a model for the common representation which is the ultimate goal of our experiments. Thus we have to start with quite simple experiments, approaching towards more and more complex and realistic scenarios. Topics that will be addressed within the experiments are:

1. how visual and acoustical information is combined to form unique objects;
2. how acoustic and visual presentations have to be designed and combined to attract and control attention (acoustic and visual icons).

To deal with the first task, visual and acoustic representations will be presented that may show congruent or divergent temporal or spatial characteristics. The characteristics that will be assessed are spatial positions and overlap, synchronous or non-synchronous motion, etc. We will start with a quite simple test condition in which a moving light dot will be displayed, either in two or three dimensions on a screen or using stereoscopic goggles. An acoustical point source will be presented to the subjects via headphones. The performance of the HIP to form single objects will be tested if the locations are overlapping or disparate and movements are synchronous or asynchronous. The aim is to derive a numerical description about to what extend the spatial and temporal attributes of visual and acoustic representations may deviate in order to form single objects.

The second task is based on the general need of multimedia representations to deal with the attention aspect. Concrete questions that will be investigated on are:

- how to achieve improvement in reaction times: using visual icons or acoustical icons separately or using combined icons. The reaction can be measured by means of tracking the head movement of a subject in front of the screen when specific icons are presented;
- how to achieve improvement of the perception of the meaning of icons: which icon can more easily be translated into a corresponding action that has to be performed.

As an advanced example that will form one of the addressed scenarios, we can regard an application where a multimodal representation is used to monitor and control a complex technical process on a big screen. If any of the parameters changes in such a manner that the controlling person has to be informed in order to react to that change, the first task is to focus his attention on the parameter. If the parameter is presented only by a visual icon, problems to attract attention arise if the icon is outside of the visual field or even at the periphery of the visual field. An additional acoustical icon that is assigned with a spatial position pointing to the visual icon is supposed to focus attention should yield a better performance. The second task is to inform the controlling person which action he should perform due to the change of the parameter. This can be achieved if the icons carry obvious meanings.

b) Visual and acoustical integration in HIP input at a symbolic level

Symbolic integration is much more complex since it depends on complexity of symbols, highly sophisticated processing, and memory. Nevertheless, experiments can be designed to reveal information about integrative processing. In these experiments, visual and acoustical inputs will be given in the form of simple sounds, corresponding to letters and words. ICP (Grenoble) has gained a lot of experience in similar experiments in the past. The experiments to be conducted in **MIAMI** will be based on those results using extended test scenarios. A testbed will be arranged by combining controlled screen displays with loudspeakers/headphones. Visual inputs will be given in the form of face/lip movement or/and written text. Corresponding acoustical input will be supplied. The human recognition rate will be tested and compared with both systems in operation as opposed to the single system. The tests will concentrate on operation in noisy and disturbing environments. Noise of various types and fading will be used to establish perceptual thresholds and the enhancement of recognition due to multimodal stimulation. Different input conditions will be used (speed, size of visual input, reduced sound quality, spatial distribution of sound sources, etc.) to check the cooperation and integration of results. Basically brief stimulations, both synchronized and desynchronized, will be investigated.

c) Visual and haptic integration

In this set of experiments, haptic and visual integration will be tested for two practically important cases of manipulation and handwriting. A manipulator with tactile feedback will be used to perform remotely simple manipulations with controlled visual participation via camera and monitor. The images will be varied in detail, quality and amount of noise. Impact of temporal and spatial synchronization will be tested for a manipulator with typically nonideal operation. Enhancement due to visual input will be assessed and fusion data from both systems will be evaluated. In a second set of experiments, the very important case of handwriting integration will be studied. Human subjects will be drawing and writing on a covered tablet, without visual input. Visual input will be activated in a controlled way by displaying the material to be written and the results of writing on a monitor. Performance will be tested for the role of visual input in speed and precision under varying amounts of features presented on the screen, under disturbances, noise, and temporal and spatial desynchronization. With respect to the models we are going to develop in WP 3, two different aspects are covered by this integration task. First, integration of gestural output with visual input from the operator's point of view. Second, the use of gestural output for control purposes, and the visual perception of the gestures by the controlled system.

Overview of all DELIVERABLES

D1: Software Tools for Multimodal Experiments *(after WP 1)*

D2: **Progress Report** *(after WP 2)* ⟸

D3: Basic Software Architecture *(after WT 3.3)*

D4: Completed Software Architecture *(after WP 3)*

D5: Symbolical Demonstrator *(after WP 4)*

D6: Analogical Demonstrator *(after WP 4)*

D7: Evaluation Report *(after WP 4)*

The current report pertains to the results of Work Package 2. The next page shows the Deliverable Description as planned. In subsequent sections, the reports for the different Work Tasks can be found. For clarity, also here the original Work Task description (synopsis) is presented, on a separate page before each section.

DELIVERABLE DESCRIPTION SHEET

ESPRIT BASIC RESEARCH Project 8579 MIAMI	Deliverable No. 2
Name of deliverable: **Progress Report** Partner responsible: NICI Date of delivery: 31/12/94 Status of deliverable: Public	Sheet 1 of 1 Issue date: 24/04/95

Technical description:

Experimental results after Workpart 2

Future use:

Information dissemination on conferences, workshops, etc.
The results are used as a basis for the work that follows in Workpart 3 and Workpart 4 within this project

Form of presentation:

Report

Chapter 2

Work Task Reports

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.1**

Task title: **Experiments:**
Visual-acoustical perception

Partner responsible: RUB

Start date: 01/07/94

End date: 30/11/94

Task manager: K. Hartung

Planned resources: RUB: 6 - RIIT: 6
(in man-months)

Sheet 1 of 1

Issue date: 24/04/95

Objective:

Evaluation of intermodal effects in the perception of visual/acoustical objects: Localization of objects (separation of objects, fusion of objects); Movement of objects; Directing attention to objects; Divided attention to objects; Intermodal enhancement in task performance.

Input:

WT 1.1, WT 1.2

Output:

Report with results

Approach:

- Definition of test parameters
- Development of test procedures
- Psychophysical tests with human subjects
- Evaluation of results

Contributions:

RUB experiments, additional software; RIIT experiments

Chapter 3

Visual-acoustical perception (WT-2.1)

Report 2.1 here

(Klaus Hartung, RUB + RIIT)

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.2**

Task title: **Experiments:**
Visual-speech perception

Partner responsible: ICP

Start date: 01/07/94

End date: 30/11/94

Task manager: C. Benoit

Planned resources: ICP: 6 - RUB: 1
(in man-months) RIIT: 3 - DIST: 2.5

Sheet 1 of 1

Issue date: 24/04/95

Objective:

To evaluate and understand how auditory and visible speech are integrated by humans: Evaluation of the natural anticipation of vision on audition in speech perception; Influence of vision on auditory speech intelligibility in adverse conditions; Visual disambiguation in the cocktail party effect; Effect of channel desynchronization on audio-visual speech perception; Effect of a spatial delocalization of the sources on speech intelligibility; Evaluation of the McGurk effect (contradictory stimuli); Intelligibility of distorted videophone face images.

Input:

WT 1.1, WT 1.2, WT 2.1

Output:

WT 2.8, WT 3.2, WT 3.4, WT 3.6

Approach:

- Definition of test conditions
- Development of test procedures for natural and synthetic stimuli
- Psychophysical tests with human subjects
- Analysis and interpretation of results

Contributions:

ICP experiments; RUB spatialization of speech recordings; RIIT videophone experiments; DIST experiments

Chapter 4

Visual-speech perception (WT-2.2)

4.1 Introduction

The problem of audio-visual speech perception is a primary example of audio-visual information integration. This has been largely investigated but there are still unknown factors which are important from the point of multimedia applications. These applications, like e.g. multimedia electronic mail, refer to the standard working conditions with computer display. On the display, there is a video image of a speaking person. To be perceptually pleasing, quality of video and speech must be sufficient and at the same time both of them have to be properly synchronized. Lack of synchronization is very annoying and stressful. The precision of synchronizatio depends on the size of the images, size of the person head, viewing distance and pciture quality. These effects are not easy to evaluate qualitatively, and this is important research topic. Data loss on the video channel is not instantly noticeable. At 25 frames/second it is quite possible to lose a frame without the viewer noticing. However, since there is no explicit time synchronization between the audio and video channels, data loss gradually degrades the lip synchronization. Lip synchronization effects become noticeable in most critical conditions when the timing of the sound relative to the video exceeds approximately -40 ms to +20 ms [13], but its importance is highly dependent on the material being viewed and the susceptibility of the viewer. Before all these factors can be estimated one needs however a testbed working in standard computer environment, enabling very precise control of of synchronization between the video and audio channels. This is a nontrivial task, and it required a lot of effort to solve it. Subsequent experiments which have been preformed by us resulted in a quite surprising conclusion that synchronization is not a very critical issue for the working conditions with computer displays. We have no clear clues as to why this happens, and we feel that further evaluations are needed before the formulation of definitive conclusions.

This is additionally supported by diffculties we met in building the testbed. The following report describes the testbed and initial experiemnts which have been performed.

4.2 The Testbed

After much investigation, we found that ther is only one available system on the market which allows recording of high quality video with audio in workstation environment, keeping the frame rate and enabling to change the audio time shift. We have assembled thus the following system:

- Sun Workstation with SunOs 4.1.3 operating system

- Parallax Video board with JPEG hardware

- Parallax Real-time Video Toolkit (RTV)

- Video camera and Microphone

The synchronization scheme can be seen in the Fig.1. The Parallax Real-time Video Toolkit (RTV Toolkit) initializes the Parallax video hardware and sets up a video window and audio channel with video camera and microphone to record video clips to hard disk and associated audio slices to a UNIX file. The audio/video (A/V) clips back in real time and convert image to pixrect files. We can get PAL sized (768x576) image sequence with full 50 fields/second (25 frames/second) and 8 kHz audio signal. Unfortunately, even this setup had many problems and only after few months it has been debugged by the manufacturer. It turned out, however that the synchronization is still not kept precisely. Also, since fast lip movements are very critical to the synchronization and evaluation of results, in fact a complete automated system for measurement of lip movements and visualizing the synchronization is necessary. This system has been designed and realized using Sun workstation.

4.3 Video Signal Processing

The video signal processing module consists of four steps, edge detection, motion estimation, lip tracking and lip motion estimation for the whole sequence,(see Fig.2)

4.3.1 Edge detection

Thresholding is used for image segmentation. The application of the thresholding technique is based on the assumption that object and background pixels in the digital image can be distinguished by their gray-level values [29]. The histogram of an image may be considered that it represents the distribution of the image brightness. Using the histogram form, it is possible to determine an optimal threshold value for segmenting the image into the two brightness regions. This approach is referred to as global thresholding. Over the past years, several techniques have been proposed for automatic global threshold selection. For a survey of thresholding techniques, see [30]. Because we only want to get estimate displacement vector of lips, and usually the contrast between lip and face is quite low, we select much lower threshold than normal, the image with contour can be seen in Fig.5.

4.3.2 Lip motion estimation

There are a couple of methods for motion estimation, e.g. reviewed in [35]. One of the widest used methods is blockmatching, as it can be implemented relatively easily. Using block-matching, a displacement vector is obtained by matching a rectangular measurement window, including a certain number of neighboring picture elements, with a corresponding measurement window within a search area, placed in the preceding or in the successive image. The match is achieved by searching the spatial position of the extremum of a matching criterion, e.g. of the mean absolute displaced frame differences (MAD). The reliability of a displacement estimate depends on the chosen size of the measurement windows, in conjunction with the present amount of motion. Thus, known blockmatching techniques fail frequently as a result of using a fixed measurement window size. The match obtained by simple block-matching is an optimum only in the sense of a minimum MAD, but frequently it does not correspond to the true motion. We used a hierarchical blockmatching to provide reliable estimates of the true displacement vectors [7]. The displacement estimate is obtained recursively at different levels of a hierarchy, using distinct sizes of measurement windows. Due to adaptive parameters, the hierarchical blockmatcher is able to cope with large displacement vector fields with high accuracy. We only estimate the motion vectors of edges.

4.3.3 Lip tracking

There are several reports on automatic lip-tracking research based on image processing techniques. Most of them use lip shapes or lip contours as the visual information for automatic recognition. Various feature extraction and pattern recognition techniques have been used in automatic lip-tracking, for example vector quantized codebooks of images [38], distance measurements [27] and Fourier descriptors to code the lip contours [23]. In this paper, we only consider the situation when user is quite close to the desk-top video camera. that is image sequence captured with big face. Then the change of distance of edge motion vectors are measured. If the distance change is large enough to exceed the fixed threshold, the center points of measurement window are pointed as possible position of lips, see Fig.6.

4.3.4 Sequence motion estimation

The motion vectors of the possible lips in the whole image sequence are estimated, the result can be seen in Fig.7. This method is very efficient in the situation like only one people face to camera with stable background. Otherwise, there may be some problems [30, 35]. Sequence Motion Estimations which can be solved by using some other face detection methods [26, 45].

4.4 Speech Signal Processing

A sequence of samples (8kHz) representing a typical speech signal is shown in Fig.8. It is evident from this figure that the properties of the speech signal change with time. For example, the excitation changes between voiced and unvoiced speech, there is significant variation in the peak amplitude of the signal, and there is considerable variation of fundamental frequency within voiced regions. The scheme of speech signal processing in this paper can be seen in Fig.3. There are three stages in the scheme. The first stage is a bank of linear filters, equally spaced on a critical-band scale [44]. This is followed by envelope demodulation (noncoherent). Noncoherent demodulation relies on detection of envelope information and is not dependent on signal phase coherence [39]. After the output of second stage third step is thresholding, The starting point of strong voiced region can be seen in Fig.9.

4.5 Experiments

Everyday experience suggests that we are aware of the correspondence between speech sounds and the movements of the speaker's lips. This is why we feel discomfort when a film soundtrack slips out of synchronism with the film. A great deal of psychological research in the area of audio-visual speech recognition has been carried out. Most researchers [10, 17] demonstrate that lip movements provide vital information for the understanding of language. The problem is what is the sensitivity of the human audiovisual integration system to the loss of synchronization. We have found in our experiments that in the computer dispaly viewing conditions, even quite substantial loss of synchronization of up to 200 ms was not significant. Further tests are needed to check if this effect is not made by other factors like imperfect lighting and picture quality.

4.6 Introduction

Several perceptual experiments have been run at the ICP-Grenoble and at the DIST-Genoa in order to better understand how auditory and visible speech are integrated by humans. This first year, efforts were focused on the influence of vision on auditory speech intelligibility in adverse conditions, on the visual disambiguation in the cocktail party effect, and on the evaluation of the natural anticipation of vision on audition. The audio-visual intelligibility of the most relevant phonetic units in French has been compared to their intelligibility in an auditory alone condition, under various conditions of background noise, depending on the kind of visual display used: the real face of a reference speaker, its lips alone, and several 3D models of facial components (lips, jaw, whole face). All the synthetic displays were animated, at the ICP-Grenoble, from facial measurements made automatically on a real speaker's face. A 3D model of the face has been evaluated in terms of correctly identified mouth gestures in a two-choice test where two synthetic mouths uttered different words, only one of them being synchronized with the audio signal after it had been degraded. Correct responses have been compare, at the DIST-Genoa, across various conditions of display rate and of number of parameters controlling the facial gestures. All those experimental results are detailed below. This chapter is divided into three paragraphs. Paragraph 4.2 presents results from two perceptual experiments run at the ICP-Grenoble; the second paragraph 4.3 presents the experimental platform worked out at the DIST-Genoa and perceptual results on discrimination of visible speech; paragraph 4.4 presents a perceptual experiment run at the ICP-Grenoble on the natural anticipation of vision on audition in speech identification, compared

4.7 Audio-visual intelligibility of talking faces

4.7.1 Displays

The experiments have been done with several kinds of display :

- natural human face with make-up lips
- model of the face (Parke's model)
- model of the lips (Guiard and Adjoudani's model)
- model of the skull
- binarized human lips

The lip model has been developped at ICP by Guiard and Adjoudani(1992, 1993). It is controlled through five parameters :

- mouth width
- mouth aperture
- mouth corner protrusion
- upper lip protrusion
- lower lip protrusion

This high resolution model is made of 200 Gouraud's shaded polygons.

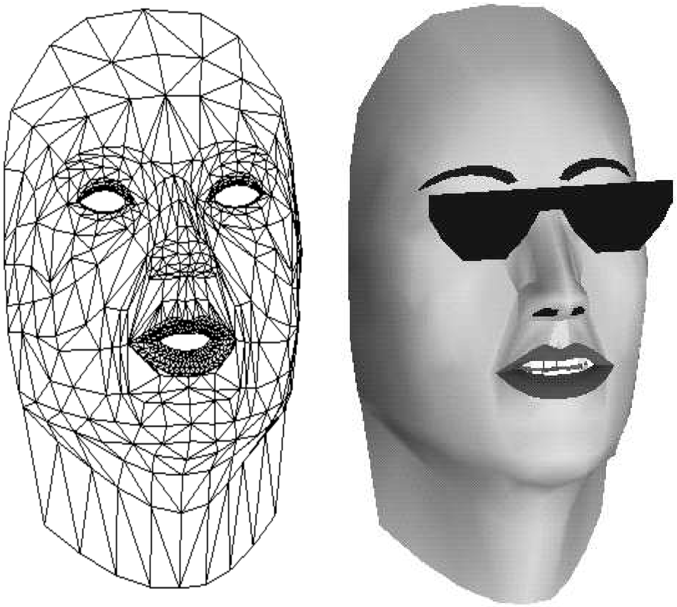

Figure 4.1: Modified version of Parke's model. Left: wireframe structure; right: Gouraud-shaded rendering

The model of the face that we used was first designed by Parke (1974)[24]. It has been implemented on a SGI graphics computer and improved for speech production by Cohen and Massaro (1993, 1994)[14][15]. It is animated through controls related to physiological gestures, e.g., "raise chin", "raise lower lip", "jaw thrust", etc. Our objective was to animate as best as possible this model from the above mentioned parameters of the lip model. A control interface has thus been developed to predict the original commands of the face model from only six parameters that are easy to measure on a speaker's face. Five parameters were used in the control of the lip model. An extra one was necessary, namely the chin vertical displacement (M). The interface mostly used linear combinations of parameters. It allows different face and animation styles to be generated, e.g., large vs. narrow face, hypo- vs. hyper-speech, and the like. The original lips were replaced with the high resolution lip model in the face model. This greatly improves the control of the face model with our six parameters.

These two parametric models are animated with several parameters files obtained from analysis of a speaker filmed form front and side.

4.7.2 First experiment

Extending the experiment by Benoît et al. (1994)[6], the audio-visual intelligibility of the face model and of the lip model have been quantified under five conditions of acoustic degradation.

Preparation of the stimuli

The speech material consisted of the natural acoustic utterances of a French speaker and of four kinds of visual display: no video, synthetic lips, synthetic face, and natural face. The two synthetic models were animated from parameter files so that no delay affected the original synchrony between audio and video.

The corpus was made of VCVCV nonsense words. V was one of the three French vowels /a/, /i/ or /y/. C was one of the six French consonants /b/, /v/, /z/, //, /R/ or /l/. The test words were embedded in a carrier sentence of the form ”C’est pas VCVCVz ?”.

1. **no video** : The eighteen different sentences were first digitized. They were then acoustically degraded by addition of white noise, at five S/N levels, by 6 dB steps. There were overall 90 audio stimuli. A pseudo-random order was used for presentation. Ten extra stimuli were appended before the actual test so that subjects could adapt to the test conditions.

   These acoustic stimuli served as a reference to the next three experimental conditions where the natural face or the synthetic models were simply synchronized to the audio part.

2. **natural face** : The original video recording of the speaker was digitized and compressed on a PC through a VIDEIS board. The front view of the lower part of the face, from the neck to the middle of the bridge of the nose, was displayed on a 15 inch monitor. The video rate was 25 ips (PAL format) with a VHS-like quality. Audio stimuli were post-synchronized with the image display. A visual alone condition was added to the five audio-visual conditions. This sub-test had 108 stimuli.

3. **synthetic lips** : the lip model was Gouraud shaded and animated at 50 ips on the 19 inch monitor of an SGI Elan. It was displayed at a 20 degrees angle view from the sagittal plane.

The actual width of the lips on the screen was roughly 10 cm. The audio file controlled the display of the model, one image being calculated every 320 audio samples.

4. **synthetic face** : the face model was Gouraud shaded and animated at 25 ips on the 19 inch monitor of an SGI Elan. It was displayed at a 20 degrees angle view from the sagittal plane. The actual width of the whole face on the screen was roughly 10 cm. The digital audio files controlled the display of the model, one image being calculated every 640 audio samples.

Procedure : 14 normal-hearing French subjects took part in the experiment. The order of presentation of the four sub-tests was balanced accross the subjects. Each sub-test lasted 20 minutes. Each subject ran no more than two sub-tests per half-day. Subjects answered through a keyboard in the "natural face" test. They answered with the mouse on the screen in the other tests. Subjects were recommended to respond to both the vowel and the consonant, as much as they could guess it. A "?" response was tolerated, however.

Global intelligibility

A test word was first considered correct only if both the vowel and the consonant were correctly identified. As for auditory and visual intelligibility of natural stimuli, the results obtained in this experiment are in agreement with those by Benoît et al. (1994)[6]. Adding the video image of the natural face shows a dramatic gain in intelligibility over presenting the audio alone, as seen in Figure 4.2. The lip model and the face model also contribute strongly to improve the intelligibility of auditory speech. Scores in lipreading conditions are not presented in Figure 4.2 simply because they are strictly identical to those obtained under the most degraded acoustic condition (S/N = -18 dB). In fact, no acoustic cues could even be detected at this noise level. Figure 4.2 shows that the synthetic lips account for one-third of the intelligiblity carried by the whole natural face, whatever the acoustic degradation. The synthetic face accounts for the two thirds of it.

The contribution of the synthetic lips and of the synthetic face to visual speech intelligibility is certainly impressive when considering the large contribution of the whole natural face. Five parameters are sufficient to animate the lip model alone. Even without the teeth, the tongue, the chin and the skin, the intelligibility carried on by the lip model is striking, and this is obtained with a very small quantity of information. As for the face model, a sixth parameter by itself almost doubles the visual intelligibility provided by the synthetic lips to the perceiver. Here again, there is no control of the tongue, but the teeth and the chin are animated, and the structure of the face is coherently displayed. The visual information provided by the face around the lips allows subjects to disambiguate confusions among spread vowels (/i/ vs. /a/) which are largely mixed up through the lips alone.

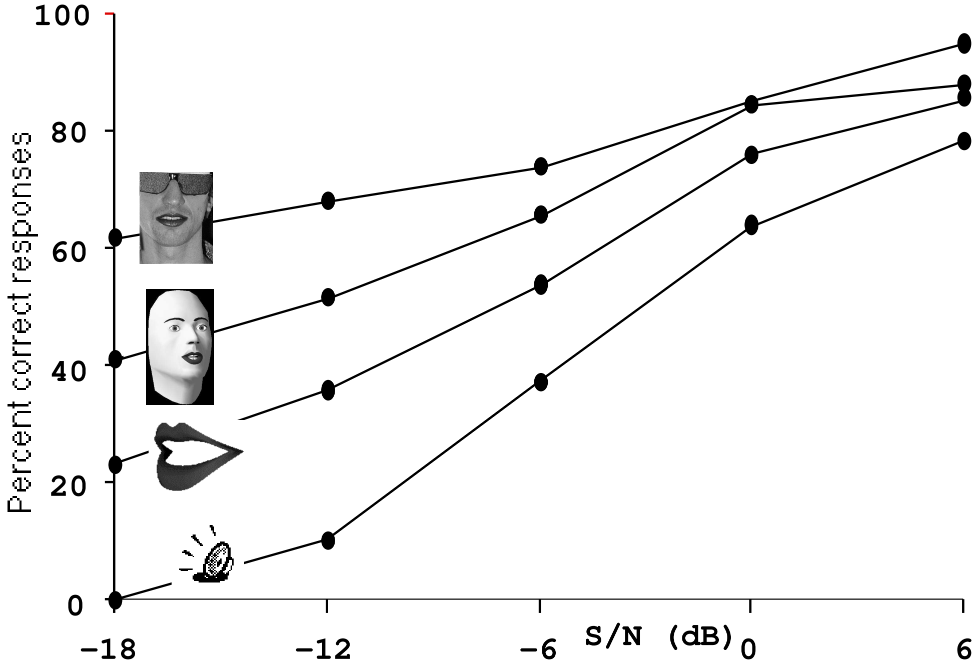

Figure 4.2: Intelligibility scores obtained by 18 subjects in the identification of 18 stimuli, as a function of acoustic degradation, depending on the mode of presentation: Audio alone, audio plus the lip model, audio plus the face model, audio plus the whole natural face (from bottom to top).

Sumby and Pollack (1954) proposed an index of the visual contribution to the missing auditory information: (I[AV]-I[A])/(1-I[A]) where I[AV] and I[A] are the Audio-Visual and Audio intelligibility scores in a given S/N condition. Figure 4.3 shows the evolution of this index along the acoustic degradation at the three S/N conditions where all differences in intelligibility are signifiant, i.e., between -18 dB and -6 dB, for the three Audio-Visual conditions. The index is remarkably constant over the acoustic conditions of degradation.

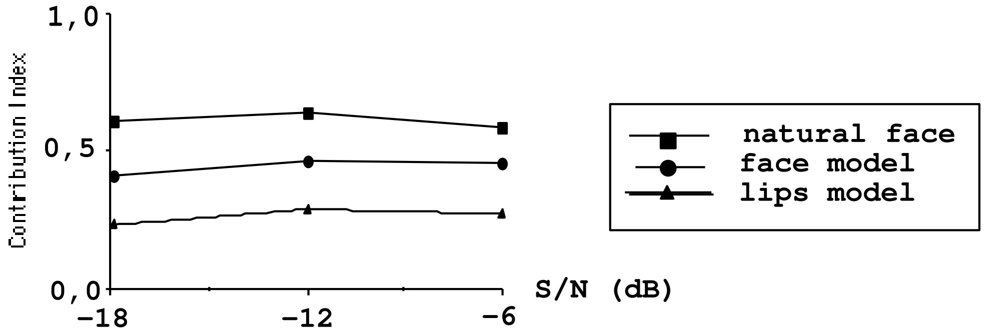

Figure 4.3: Contribution index of the visual information to missing acoustic information.

Consonant confusions

Overall, the whole natural face restores two thirds of the missing information when acoustics is degraded or missing; the facial model (tongue movements excluded) restores half of it; and the lip model restores a third of it. This is strong evidence that a very low bit rate of information (five or six parameters 25 times per second) is sufficient to transmit a great deal of the visual information carried on by the speaker's natural face, even though tongue gestures are not yet controlled.

Consonant confusions are presented in Table 1 at S/N = -12 dB where differences are at their maximum.

Whatever the consonant, there is a very strong disambiguation due to visual information. The disambiguation power follows the same hierarchy as that of global intelligibility, from the lips alone to the natural face, through the synthetic face.

- /b/ is the consonant best identified audio-visually, although it is given as a response to many /v/ stimuli, especially with the lip model (43%). The absence of teeth is obviously the reason for these confusions.
- // identification is not improved when vision of the lip model is added to audio. However, // is rather well identified with the synthetic face or with the natural face. With the lip model, there are many confusions between // and /R/ in spread-vocalic context. Moreover, not only // is never identified in a /i/ context, but it leads subjects to identify the vowel /i/ as an /a/ (/iii/ is perceived as /aRaRa/). Adding the chin and the teeth disambiguate both the carrier vowel and the carrier consonant.
- The two liquids /l/ and /R/ are mixed up in all conditions. A main reason is obviously that there is no tongue associated with the lip model, and that the tongue is not controlled in the face model. Even with the human face, confusions occur in /i/ and

Table 1. Confusion matrices of consonants, irrespective of the response on the vowel (S/N = -12 dB). Stimuli are presented in rows. Percepts are presented in columns. Scores are out of 42.

Audio only

	b		l	R	v	z	?
b	**11**	5	1	3	5	1	16
	3	**10**		7	1	5	16
l	6	3	**1**	7	4		21
R	4	1	2	**13**	3	3	16
v	6	1	2	7	**5**	1	20
z	4	7	2	1	3	**2**	23

Natural face

	b		l	R	v	z	?
b	**38**				4		
		35	1	1	3	2	
l		3	**25**	10	1	2	1
R		6	3	**25**	1	7	
v	3	3		1	**34**		1
z		9	1	1	2	**28**	1

Lip model

	b		l	R	v	z	?
b	**39**	1			1		1
		10	6	17	4		5
l			**16**	21	1		4
R	1	2	10	**21**	4		4
v	18		4	1	**13**		6
z	1	4	3	7	5	**18**	4

Face model

	b		l	R	v	z	?
b	**36**		1		3		2
		30	3	3		3	3
l		2	**14**	21	1	1	3
R	2		14	**19**	1	3	3
v	7	2	4	4	**21**	1	3
z		6	6	5	7	**15**	3

/y/ contexts, that is when the lip opening is too small for subjects to see the vertical tongue movement characteristic of /l/.

- /z/ is auditorily identified below chance, and // is then the most frequent response. This is obviously due to the background noise used. Surprisingly enough, our lip model helps subjects to disambiguate /z/ and // better than the face model. Nevertheless, a significant amount of // responses to /z/ stimuli remains when a natural face is presented. In fact, these confusions only occur in the /y/ context. /y/ has such an important coarticulatory effect that all consonants (but /b/ and /v/) look very similar when surrounded by two /y/'s. Therefore, audio-visual confusion of consonants presented in a /y/ context is mostly based on auditory similarities.

Vowel confusions

Table 2. Confusion matrices of vowels, irrespective of the response on the consonant (S/N = -12 dB). Stimuli are presented in rows. Percepts are presented in columns. Scores are out of 84.

Audio only

	a	i	y	?
a	**52**	2	4	26
i	4	**25**	34	21
y	2	12	**48**	22

Natural face

	a	i	y	?
a	**82**	1		1
i	10	**72**		2
y			**84**	0

Lip Model

	a	i	y	?
a	**76**	3	2	3
i	35	**35**	8	6
y		1	**80**	3

Face model

	a	i	y	?
a	**75**	9		
i	2	**82**		
y	1	1	**82**	

The vowel confusions are presented in Table 2 at S/N = - 12 dB where differences are at their maximum. Complementarity between audition and vision is clearly seen in Table 2. In the auditory mode, /a/ is seldom confused with /i/ or /y/, whereas /i/ and /y/ are largely mixed up. On the opposite, /y/ is seldom confused with /a/ or /i/ in the (audio-)visual mode, whereas there are many confusions between /a/ and /i/ in the audio-visual mode.

As stated above, there is almost no auditory confusion between /a/ and /i/. However, an /a/ response is given to an /i/ stimulus in 50% of the cases, whereas /a/ is almost always identified when the lip model is simultaneously displayed. This effect remains to a smaller extent with the natural face, where an /i/ stimulus still leads to /a/ responses in 12% of all cases. 70% of these latter confusions are observed with /izizi/ (perceived as /azaza/ by half of the subjects.) In fact, this is mostly due to the individual utterances /izizi/ and /azaza/ selected as stimuli for our experiment. As shown by Benoît et al. (1992)[5], /a/ shows smaller lip and jaw opening, and takes the shape of an /i/ when surrounded by /z/. They also noticed that /z/ has the same shape when surrounded by /i/ or /a/. Those statistical observations were obtained from a multi-dimensional analysis of ten utterances of /izizi/ and /azaza/. We looked back at the original data. It turns out that the /azaza/ used in our intelligibility test is amongst those with the largest lip and jaw opening. The /izizi/ here selected is also the one with the largest lip opening (and with an average jaw opening.) Differences in lip (resp. jaw) opening between our two stimuli /izizi/ and /azaza/ are in the range of 1 mm (resp. 0.5 mm.) It is thus not surprising that this hyper-articulated /azaza/ has been correctly identified, whereas half of the subjects perceived the hyper-articulated /izizi/ as another /azaza/, when they had the oportunity to see the natural face. This effect is emphasized with the lip model, where subjects cannot see the (even small) jaw movements. More surprisingly, the face model allows subjects to correctly identify /i/ on the one hand, and to respond with an /i/ percept to an /a/ stimulus. Those errors occur when /a/ is coarticulated with labial consonants in the two thirds of the cases. The fact that /izizi/ is not here perceived as /azaza/ is probably due to an insufficient control of the chin displacements by the face model.

4.7.3 Second experiment

Extending the above experiment, this new test was led to quantify jaw intelligibility.

Preparation of the stimuli

Synthetic lips and audio only tests were used as reference to the other test.

The two other kinds of display were a synthetic skull and binarized human lips.

The synthetic jaw we used for our model was first elaborated at McGill University (Guiard-Marigny and Ostry, 1995) to visualize jaw motion kinematics, during speech or mastication, recorded with an optoelectronic measurement system. The visualization uses 3D digitized upper skull and jaw with their corresponding teeth. Overall the whole facial structure is made of a mesh of 6000 polygons. Guiard-Marigny and Ostry (1995) animated this jaw model from three rotations and three translations automatically derived from the motion of a rigid structure attached to the lower teeth of a speaker. The syn-

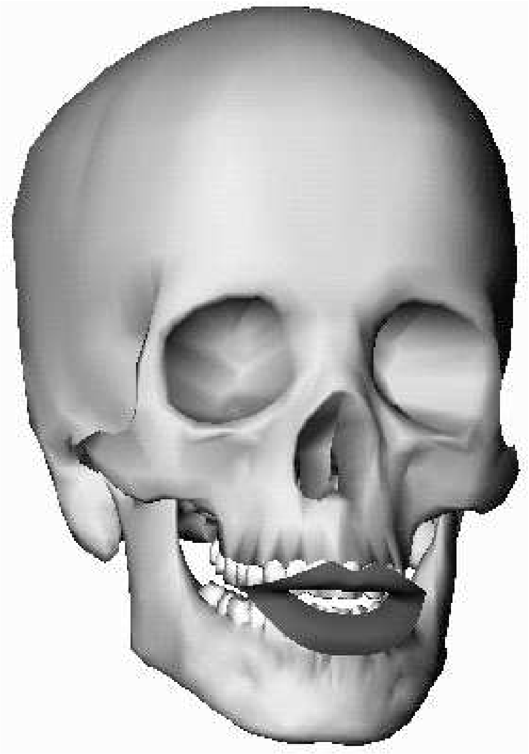

Figure 4.4: The 3D digitized skull with our lip model superimposed on.

thetic upper skull and jaw can then be animated in synchrony with the audio part of the natural speech . The lip model has been superimposed to the 3D skull with its jaw model, as shown on Figure 4.4.

Another human display was added : binarized lips. Video of blue make-up speaker passed through a chromakeyer to obtain white lips only. These ”2D” lips were displayed on a dark screen form front view.

Procedure : 20 normal-hearing French subjects took part in the experiment. The procedure was the same as above.

Global intelligibility

A test word was first considered correct only if both the vowel and the consonant were correctly identified.

Reference tests (audio only and lip model) obtained the same results as above experiment. They won’t be discussed anymore.

The measurement method we used to derive the jaw motion kinematics is not optimal because the chin and the jaw motions may somewhat differ. For instance, the jaw lowers to produce /a/ in the word /ababa/ while the lower lip raises to close the mouth for

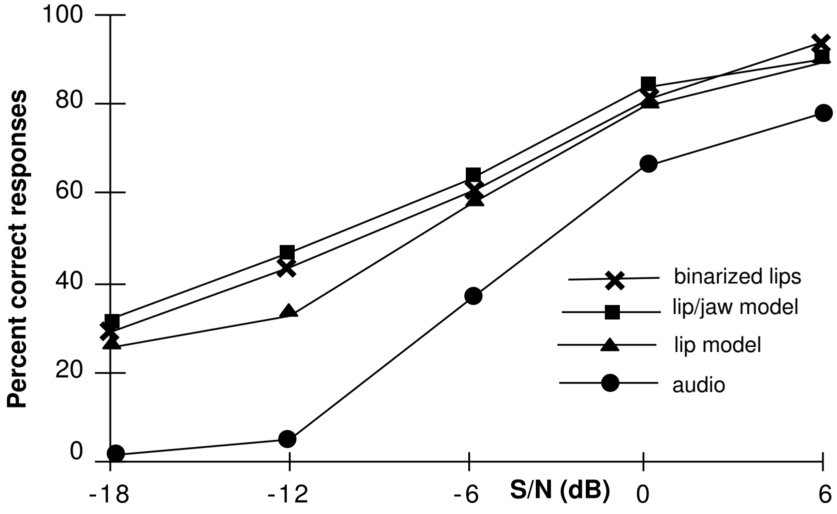

Figure 4.5: Audio-visual intelligibility of the lip model and of the lip/jaw model compared to the auditory alone intelligibility of speech, across various levels of degradation by additive noise.

the /b/. This makes the chin skin roll up over the jaw bone. Despite this, we obtained a noticeable gain in speech intelligibility when the synthetic jaw was added to the synthetic lips, as shown on Figure 4.5.

Table 3. Confusion matrices of consonants, irrespective of the response on the vowel (S/N = -12 dB). Stimuli are presented in rows. Percepts are presented in columns. Scores are out of 120.

Jaw and lips

	b		l	R	v	z	?
b	**52**		1	1	5		1
		29	6	11	3	5	6
l	1	10	14	**28**	1	2	4
R	2	3	11	**35**	1	4	4
v	14	8	6	1	**23**	4	4
z	5	7	10	2	2	**30**	4

binarized

	b		l	R	v	z	?
b	**52**				2	1	5
	1	**16**	12	18	2	3	8
l	1		18	**31**	1	2	7
R		3	10	**32**	2	5	8
v	12	2	4	1	**38**		3
z	1	1	10	4	9	**30**	5

When synchronized with the lip model, the jaw model enhances lip model visual intelligibility at several levels. The number of no-responses from subjects is divided by two. Vowel /i/ is much less confused with vowel /a/, mostly in closing consonantal context (/b/ or /v/). There are many less confusions between // and /R/, whatever the vocalic context. Finally, /b/ is no longer confused with /v/, especially in a /i/ vocalic context.

Table 4. Confusion matrices of vowels, irrespective of the response on the consonant (S/N = -12 dB). Stimuli are presented in rows. Percepts are presented in columns. Scores are out of 60.

Jaw and lips

	a	i	y	?
a	**110**	4	2	4
i	18	**88**	8	6
y		1	**117**	2

binarized lips

	a	i	y	?
a	**114**			6
i	44	**54**	12	10
y	1	1	**111**	7

On the opposite, /l/ and /v/ are more often mixed up with //, but this only occurs in rounded vocalic contexts. Vision of the jaw also leads to a larger amount of confusions between /i/ and /a/ in a /z/ context.

Binarized lips had the same score as jaw model, as seen in Figure 4.5. Transmitting black and white human lips required 180x110 pixels, i.e 19800 bits (19 ko). Transmitting jaw and lips parameters required 6x1 bytes. The intelligibility is almost the same in each case... Lack of chin position introduced the same kind of confusion as lip model, i.e confusion of /i/ and /a/. For consonnants, binarized lips confusions are the same as lip model confusions.

4.8 Influence of display rate and of parametrization on visual speech identification

The basic idea was that of implementing in software a flexible system for the subjective evaluation of bimodal comprehension of speech and, in particular, for understanding the relevance of the various mouth articulatory components in visual comprehension of speech. These components are in fact those which must be at maximum preserved is the visual syntehis of speech. These experimentations has driven the implementation of suitable algorithms for synthesizing mouth images starting from a set of 6 articulatory parameters (mouth width and height, upper lip offset, lip thickness, jaw aperture and tongue position). The system has been implemented on a Silicon Graphics Indigo 4000 XZ workstation.

4.8.1 Description of the experimental testbed

The system can be parameterized by means of buttons and control windows encharged of fixing:

- time resolution (frame/second);
- spatial resolution (pixel/frame);
- aspect ratio (frame width/height ratio);
- face/mouth region of interest (only the image of the mouth or the whole speaker face);
- the partial occlusion of some articulators (a suitable mask can be overwritten on the image to optionally occlude some parts of the mouth);
- the audio quality (by adding white noise and tuning the S/R);
- the audio-video alignement (some delay/anticipation can be optionally selected between the sound and the video);
- audio-video association (the mouth sequence can be associated to the corresponding speech or, conversely, to different speech to test audio/visual confusion);
- visual synthesis (no interpolation, linear interpolation, speech assisted synthesis from 6 articulatory parameters);
- presentation (1/2/3/4 images in the screen).

4.8.2 Visual synthesis from articulatory parameters

The above described system works on prerecorded audio-video sequences representing a phonetically and articulatory balanced corpus in italian. The corpus contains round 400 isolated italian words together with the corresponding 25 Hz video (speaker's face) and articulatory description. This last information consists of a vector of 6 articulatory parameters being (see 4.6):

LM jaw aperture;

W mouth width;

H mouth height;

dw lips closure;

Lup upper lip offset;

tongue tongue position.

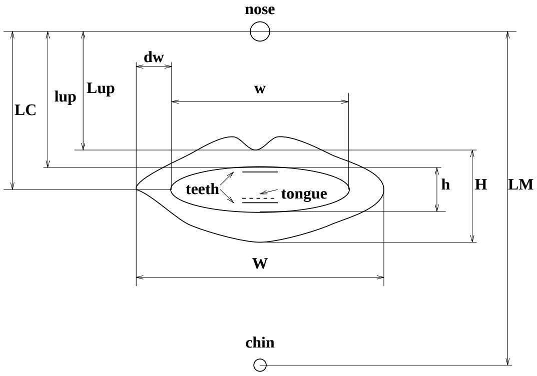

Figure 4.6:

Two different mechanisms have been employed for synthesizing the visual cues estimated from speech analysis: i) key-picture vector quantization; ii) parametric model animation. With both techniques, a variable number of articulatory parameters, ranging from a minimum of 1 to a maximum of 6, can be used to drive the synthesis of each 25 Hz picture whose quality increases in proportion to the number of parameters which are used.

Key-picture vector quantization

The thousands of 5-dimensional articulatory vectors extracted from the audio-video corpus have been clustered around a predefined number of key-configurations (in our experiments 256 configurations) which are assumed to carry the mouth articulatory information which is more relevant for the comprehension of speech. Parameter quantization works out as an addressing mechanism for retrieving the corresponding mouth image, extracted from a database of key-pictures (256). In order to save memory, key-pictures are not extracted at full resolution (128x128 pixel for the mouth region of interest) but are constructed by means of squared blocks (4x4 pixel) extracted from a codebook. This codebook contains a given number of 16-dimensional vectors (in our experiments 256) corresponding to the 4x4 pixel blocks and has been constructed through conventional algorithms of vector quantization applied over a huge training set consisting of all the video data collected in the corpus.

Video synthesis is therefore executed according to the following steps:

1. The articulatory vector associated to 20 ms of speech (25 Hz frame frequency) is quantized and coded by a 8 bit key-picture index.

2. The key-picture index is used to address a table and to select a corresponding list of 1024 8 bit block indexes.

3. Each block index is used to address a table and extract a vector of 16 pixel values which represent a 4x4 block.

4. The 1024 blocks are arranged together to form a 128x128 pixel image.

The software and hardware complexity is reasonably low and the synthesis system can be easily ported on any PC.

Parametric model animation

In this second approach, instead of being quantized, articulatory vectors are normalized to a continuous values in the interval [0, 1]. After suitable adaptation, each parameter is made compatible to the facial actions codebook (FACS) of Ekman and Friesen and is applied on a Parke facial parametric model for reproducing the mouth movements. The Parke model we have used is the one implemented at The Curtin University of Western Australia by Andrew Marriott and Valerie Hall. The graphic complexity is high requiring the use of the z-buffer, double buffering and ray tracing. The quality of the visual synthesis is less than what obtained with the previous approach (see Table 4.8) because of the following reasons:

1. the position of the tongue is most of times wrongly reproduced since it is uniquely based on the correponding articulatory parameter. With the key-picture approach, on the contrary, a whole mouth image is selected and approximated: the correct position of the tongue is already "embedded" in this picture without requiring further processing;

2. the mouth looks "very synthetic" and "impersonal". On the contrary, the use of key-picture where a "true" mouth is reproduced;

3. secondary articulatory elements like the nose contraction and cheeck inflation are definitely lost in the model while are preserved in key-pictures. The correct reproduction of these elements improves significantly the visual comprehension of speech.

4.8.3 Bimodal experiments

- One speaking frontal face with his synchronous speech. Other independent speech signals corrupting the acoustic channel. Disturbing signals have an initial S/N greater than the message signal. The S/N of the message is progressively increased with respect to diturbs whose S/N is conversely reduced. Different speakers, different messages, different disturbs in number and typology (content, male/female, ...). What are the relationships in S/N, guaranteeing comprehension?

- Similar to the previous experiment but with vocal disturbs.

- Three speaking frontal faces, far from the camera so that lips movements can be hardly perceived. One single speech signal synchronous with one of speakers. The other two speech signals are suppressed. Camera zooms progressively until an unbiased observer manages to associate the heard speech to its corresponding speaker.

- Similar to the previous experiment but with vocal and non vocal disturbs.

- Three speaking faces close to the camera but with a 90 degree rotation with respect to the focal axis. One single speech signal synchronous with one of the speakers. The other two speech signals are suppressed. Speakers rotate progressively toward a frontal position until an unbiased observer manages to associate the heard speech to its corresponding speaker.

- Similar to the previous experiment but with vocal and non vocal disturbs.

- Three frontal speaking faces close to the camera. Only the speech signal corresponding to one speaker is reproduced while the other two are suppressed. Speech is however reproduced with significant delay, so that no synchronization can be recognized with lips movements. Delay is progressively reduced until an unbiased observer manages to associate the heard speech to its corresponding speaker.

- Similar to the previous experiment but with vocal and non vocal disturbs.

- One frontal speaking face with synchronous speech. Speech signal undergoes periodic fading, suppression, amplitude/phase distortion. The visual channel is corrupted by noise, video interruptions, scene changes, frame freezing, zoom and camera panning, head motion ..... Estimation of the perception thresholds.

- The speaking mouth is partially occluded by means of a mask which can be defined from the user interface. This mask can occlude lips to a variable extent, the tongue or the corners of the mouth, thus hiding important articulatory elements. The influence on comprehension is evaluated.

4.8.4 Unimodal experiments

- One speaking frontal face. Camera is progressively zoomed in and out to evaluate sensitivity to distance. Does the observation of the whole face improve lipreading with respect to the observation only of the mouth?

- A 30 frames/second ”facial” sequence is displayed many times, each of them with decreased frequency, in order to estimate the minimum time resolution necessary for comprehension.

- The same sequence used in the previous experiment is displayed with lower and lower frame frequency interleaving interpolated frames. Effects on comprehension are evaluated.

- Through spectrogram analysis, frames corresponding to stable acoustic units (phonemes) are separated from those corresponding to unstable acoustic units (coarticulation). A new sequence is constructed by concatenating stable frames and by duplicating the last stable frame in place of those unstable. Effects on comprehension are evaluated.

- Similar to the previous experiment except for the fact that frame linear interpolation is used instead of duplication.

4.8.5 Experimental results

The above described experiments have provided useful indications on the articulatory relevance in speech comprehension as far as the mouth parameters are concerned. A basis of 6 mouth articulatory parameters has been defined thanks to extensive cross-correlation analysis (see Figure 4.7). General considerations have been found out for the definition of th optimal configuration for speech visualization:

- a time video frequency in the range [15, 25] Hz;

- a spatial resolution of at least 128x128 pixel in the mouth region;

- the superiority of frontal to side articulatory cues;

- the effective integration of frontal and side articulatory cues in presence of significant acoustic noise;

- the high sensitivity to audio-video misalignments which requires very precise synchronization;

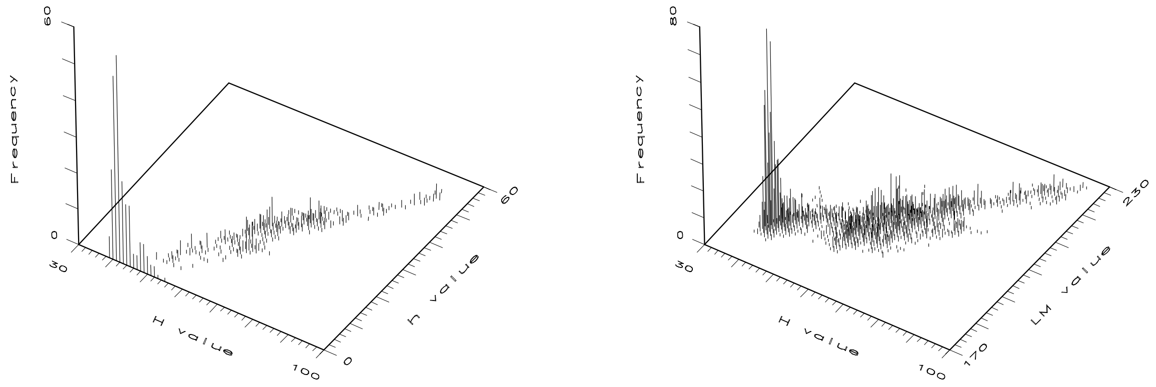

Figure 4.7:

- the subjective preference to a full-face presentation (even with still background) instead of a only-mouth presentation;

- the subjective preference to a "true" face presentation (key-pictures) instead of a synthetic face presenatation (Parke model);

- the relevance of the information associated to the tongue, more precise in key-pictures than in the Parke model;

- the relevance of secondary articulation (nose and cheecks), present in key-pictures but not in the Parke model.

4.9 Natural anticipation of vision on audition in speech

The experiment here reported aimed at quantifying the natural anticipation of vision on audition in the identification of speech segments, and to compare this effect across a natural face and a synthetic face animated from measurements made on the latter. Because of coarticulation, lip rounding occurs before audio utterance, especially for /y/. In the sentence "t'as dit /y/", the vowel /y/ produced protrusion between end of "dit" and beginning of "/y/". Thus, it gave subjects a clue for anticipation. The experiment was based on a "floating" window of 0.5 seconds showing a part of "t'as dit /y/" movie

FREQUENCY	FRAME DUPLICATION	LINEAR INTERPOLATION	SPEECH ASSISTED INTERPOLATION
12.5 Hz	92 %	94 %	94 %
8.3 Hz	80 %	84 %	94 %
6.25 Hz	52 %	58 %	94 %
5 Hz	46 %	46 %	88 %
4.16 Hz	30 %	24 %	80 %
3.5 Hz	24 %	16 %	75 %
3.125 Hz	18 %	10 %	75 %

TABLE 1

Description: 2 mouth images, one synchronous with speech and one pronouncing a slightly different word of the same duration

Characteristics of the test: high S/N, 400 italian isolated words, 10 adult and normal hearing persons.

Test: "Which of the two is the right mouth associated to the word you are hearing?"

Evaluation: percentage of correct decision

PARAMETERS	FRONTAL VIEW	SIDE VIEW	STEREO VIEWS
H, W	24 %	10 %	24 %
H,W,dw	30 %	10 %	30 %
H,W,dw,LM	46 %	14 %	46 %
H,W,dw,LM,Lup	54 %	14 %	56 %
H,W,dw,LM,Lup tongue	56 %	24 %	66 %

TABLE 2

Images synthesized through the animation of the Parke facial model at 25 Hz.

Description: 2 mouth images, one synchronous with speech and one pronouncing a slightly different word of the same duration

Characteristics of the test: high S/N, 400 italian isolated words, 10 adult and normal hearing persons.

Test: "Which of the two is the right mouth associated to the word you are hearing?"

Evaluation: percentage of correct decision

Figure 4.8:

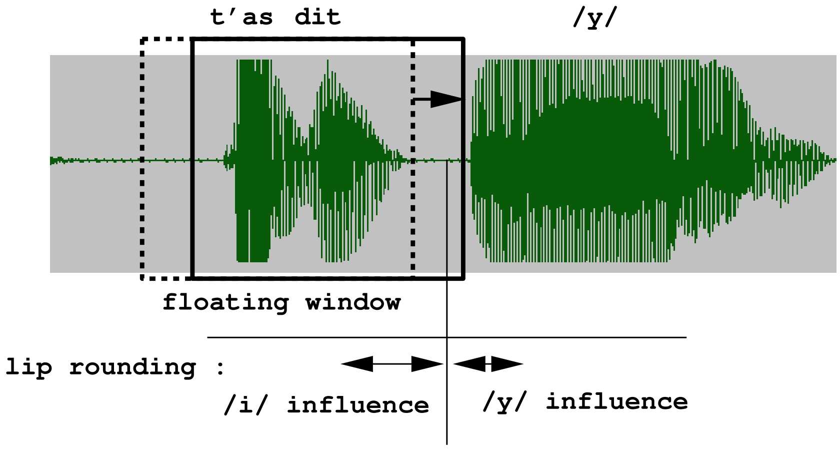

Figure 4.9: Floating window on sentence carrier.

(display of human face and synthetic face). The sentence was cut more or less time before /y/ audio utterance.

Ten subjects took part in this experiment.

4.9.1 Results

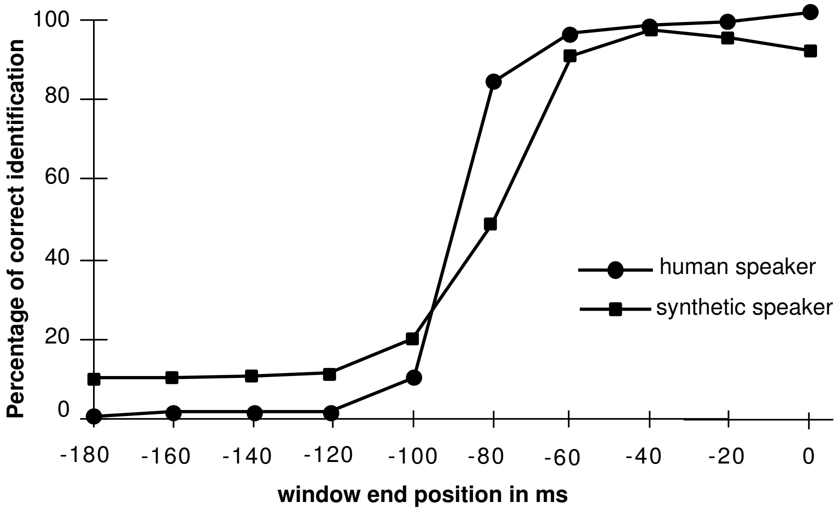

Figure 4.10: Visual identification of /y/ on two kind of display.

Figure 4.10 shows the correct identification (in %) following the position of window right bound. Audio utterance occured at 0. Recognition on human speaker occured 80 to 100 ms (50% of correct identification) before audio utterance. Recognition on synthetic face occured 80 ms before audio utterance.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.3**

Task title: **Experiments:** **Visual-gestural control** Partner responsible: DIST Start date: 01/07/94 End date: 30/11/94 Task manager: A. Camurri Planned resources: DIST: 3 - NICI: 1 - UKA: 2 (in man-months)	Sheet 1 of 1 Issue date: 24/04/95

Objective:

To study the analog control of virtual objects on the experimental platform with a pen interface: manipulation (dragging, moving, deforming); Targeting; Exploring the number of controlling degrees of freedom (pen-tip position, axial force, pen orientation). To evaluate and understand how music and human movements are integrated, by means of common metaphorical approaches.

Input:

WT 1.1, WT 1.2

Output:

Software to be used in WP 3 and WP 4, Results to be reported in WT 2.8

Approach:

- Definition of test parameters and evaluation criteria
- Development of test procedures
- Experiments with human subjects
- Evaluation of results

Contributions:

DIST experiments; NICI experiments; UKA experiments

Chapter 5

Visual-gestural control (WT-2.3)

5.1 Introduction

This section describes a preliminary model and system architecture for experimenting the human movement tracking, gesture acquisition, and its integration with animated human models and sound and music. The experiment described in this document are demonstrated in the DIST MIAMI demo videotape, presented at the first year MIAMI review meeting.

The goals of the research can be summarized as follows:

1. To experiment different categories of sensor systems, for the real-time acquisition of different human movement information. In this work, we have experimented different input systems:

    (a) handwriting:

    i. pen-based systems;

    (b) full-body movement tracking:

    i. V-scope, CosTel, MacReflex: special devices for the tracking of on-body markers;

    ii. an original fully-configurable exoskeleton device, e.g., for arms/legs movements detection;

    iii. the SoundCage dance/music system from SoundCage S.r.l.;

2. To start to develop a movement/gesture ”language” for human interaction, and explore its i ntegration with sound and music;

3. To move toward a general, integrated architecture for the high-level control of the complex tasks involved in human-machine interaction processes, including representation and planning problems. For example, there is the need for an integrated representation of both symbols and signals; also, the need for a system able to select on-line the most effective action to solve a certain interaction problem. In this phase, we developed a preliminary software architecture sufficiently flexible to allow the integration of a set of basic, different experiments in a distributed architecture, described in this document. As a further step, it is expected in the next future the development of a complete, effective implementation of a prototype of integrated control architecture supporting reasoning and a deeper integration of representations. This research demonstrated useful application both in the field of man-machine interaction and in entertainment, cultural, and artistic applications. Multimedia concerts based on the experimental devices developed in this project are in course of preparation.

5.2 The Overall System Architecture

Our distributed architecture is based on Unix (SGI Indigo and Sun Sparc) and Win32 (486 and Pentium) platforms, with particular regard to Windows 95. A beta version of Windows 95 has been adopted for experiments on real-time processing (currently beta 3, build 347). We developed a library based on sockets - both under Unix and under Win32 - allows the communication between processes possibly running on different machines. A software module for the integration with the PVM distributed environment is in course of development. The experiments described in the following sections have been implemented in such a distributed software platform, which demonstrated to be flexible enough to support and integrate all the different systems and software developed. The HARP/V-scope experiments have been implemented in both a distributed environment and a single workstation. In the first case the experiment has been implemented in two workstations running the beta version of the Microsoft Windows 95 operating system. The first is physically linked to the V-scope hardware via a high-speed serial interface (RS232C with 16550AF UART). A Windows sockets library has been developed to link subsystems running on different workstations. In the second case we use a single Pentium 90 machine (32MB RAM, an MGA Impression+ video board, and a Sound Blaster AWE32 sound board), under Windows 95, in which all the software runs locally. In both cases we use the same external audio hardware. The exoskeleton experiments run on a SGI Indigo, and is connected via sockets to the same Windows 95 workstation running the sound output subsystem.

5.2.1 The HARP software model

HARP is a system for the integrated control of agents based on a hybrid AI model, developed at DIST University of Genova. In this first year, we developed a prototype new AI model of the HARP system, and based several experiments on this platform. Every subsystem (or part of it) is modeled as an agent in the HARP preliminary model of integrated agent architecture. At the lowest level, HARP supports Microsoft OLE Automation and sockets for inter-process communication. Agents communicate with each other and are integrated with the HARP symbolic knowledge base and reasoning modules. The HARP architecture will be extended to allow a flexible approach to the integration of sound, movement, and computer animation: it will be possible to dynamically add or remove agents to tailor the system to the current context and needs. For example, let us consider a context in which a recognizer agent is active for detecting a certain class of gestures (say, a rhythmic, cadenced movement up-down of both arms). This can be achieved by a suitable processing of the markers data acquired by special sensor systems. If at a certain point that agent is not able any more to recognize that movement pattern (e.g., one or more markers become invisible for a period of time), it can communicate its goal failure to the system that will try to activate other agent(s) able to complete (e.g., by interpolation or prediction) such missing data. Another common situation regards possible change of contexts, which typically correspond to the need for recognizing different or more gestures, and possibly use them in a different way. For example, a typical change of context is due to the move of the user from an area to another in the sensorized environment. The system must load (and unload, if necessary) agents, as well as adapt itself, as needed at run-time during execution. The designer of the application can directly connect any event (simple sounds, music objects, computer animation) to any gesture the system recognizes. Moreover, he can write down possibly complex relations between sound and movement by means of the hybrid representation language. This supports integrated symbolic and subsymbolic reasoning capabilities: for example, symbolic rules and dynamic systems metaphors like force fields. In the following section we present an example of an application of the current HARP prototype.

5.3 The HARP experimental setup for movement/sound integration

The HARP/V-scope experimental application developed for experimenting sound/movement interaction is composed by four main subsystems: (a) input: the V-scope interface; (b) pre-processing and force field metaphor interaction; (c) movement recognition; (d) output:

sound and music, computer animation. Let us analyse in more detail the HARP network of agents for this experimental setup.

5.3.1 Subsystem (a) - V-scope interface

The VScope agent is designed to acquire the information on the position of a number of V-scope markers, typically placed on the body of a user. It manages both the low-level serial communication and the link with client modules. V-scope is a IR/ultrasound sensoring device developed by Lipman Ltd. for the real-time acquisition of the position of up to eight markers placed on the human body (e.g., on the articulatory joints) or in general on moving objects (e.g., a video camera). The hardware is composed by the markers, three tx/rx towers for real-time detection of markers position, and a main processing unit connected via a serial link to a computer. The sampling rate can vary from 5 to several hundreds of milliseconds per marker (20ms per marker is currently used). As for the limits due to the V-scope acquisition hardware, we are able to manage a stage whose dimension can vary from 2 to 5 meters in depth: faster sampling corresponds to a smaller area, due to the limitations of the ultrasound sensoring devices. Our experimental results show that a 12-15ms per marker is the best tradeoff between speed (a good value for human movement acquisition without loosing too much information) and stage size. The precision of the V-scope hardware is in the range of 1cm, acceptable for our application. The Vscope interface consists of an executable and two DLLs (Dynamic Link Libraries): the executable is the user interface manager and provides means for configuring V-scope settings. Low-level methods for the V-scope hardware management are encapsulated in a Microsoft Windows DLL; high-level intermodule communication methods are encapsulated in a Shared DLL (a shared-memory object). Thus we have the running EXE file which communicates with the V-scope hardware via the low-level DLL and stores the data acquired in real-time in the shared DLL, making them available to one or more clients. Besides the VScope agent itself, further agents are available: for example, the VScope Monitor is used to monitor the status of the markers placed on the dancer's body.

5.3.2 Subsystem (b) - Preprocessing

The movement data pre-processing agent can be linked either to local agents (via Microsoft OLE - Object Linking and Embedding) or to remote agents (via our Ethernet/WinSock library). This depends on the global requirements of the application: the HARP development environment is in charge of allocating agents and creating their links in the network. In a distributed environment, the motion data preprocessing agent actually executes three main tasks: it has to manage two local connections (with subsystem (a) and the Force Field Navigator agent) and one remote connection. The link with V-scope, as mentioned earlier, is achieved via a shared DLL, the link with the Force Field Navigator is based on Microsoft OLE Automation and the network link is built on standard Winsock libraries. The raw data stream from V-scope is immediately filtered, to make sure no spurious information are present and values are within a meaningful range.

5.3.3 Subsystem (c) - Movement and Feature Extraction. The force field metaphor

The communication agent reads the sensors data stream and passes it to the agents for gesture recognition and movement analysis, whose output is available to trigger or influence the activities of sound/music and animation agents. As a simple example, a feature agent might recognize that the dancer has raised his/her left arm over a certain threshold, and its output can be used to activate a certain sound processing agent. The gesture extraction task is further subdivided into several concurrent agents, each dedicated to a different kind of movement recognition task, according to the current scenario. The Gesture/Movement agents implemented in this experiments are able to recognize several different features and gestures: raising and lowering one or both hands, raising and lowering the body, opening and closing the palm of both hands, distance between hands and gesture speed. An interesting category of agents for the interpretation of movement data is based on the force field metaphor: for example, in the demo videotape we mapped the (x,y) coordinates of a marker into a force field whose three areas around peaks corresponds to similar areas of the sensorized stage characterized by different behavior (different mappings of movement/sound). The agent continuously reads the field data corresponding to the current (x,y) position and makes it available for further processing. Other agents in course of development will be able to extract higher-level features and gestures from the movement, to model complex music/movement correlations. Examples of high-level features are "how fast the movement is", "how a tempo the dancer moves". This is a kind of information which is the result of the integration over a time window. Following the results of the research in auditory perception (Leman 1990), two different ranges of time window, approximately 0,5-1s and 3-5s. are used. Experiments are in progress based on self-organizing neural networks for the classification of incoming data from sensors, including acoustic signals.

5.3.4 Subsystem (d) - Output Generation

In the simplest case, the recognized features can be linked to events: the system is presently designed to control sound and music in real time. A next step will regard the control of computer animation. The mapping of the performer's movement to sound and animation agents can be either pre-defined or dynamically updated according to the information acquiired by particular feature extraction agents. This last case currently under development, based on the new HARP model. The MIDI standard is used for sound event control: sound output agents receive MIDI commands through OLE links from movement recognition agents, and enqueue them on the MidiKer agent, which manages the low level scheduling and synchronization and the output to synthesizers.

5.3.5 The HARP/V-scope experiment

Several experiments with the system have been performed, included in the DIST MIAMI demo videotape presented at the first year MIAMI review meeting. We decided to build three different hyper-instruments (each corresponding to an agent) each placed in a different area of the test room. The three areas/hyper-instruments placed in the test room correspond to the force field shown in the figure. The three pictures of the HARP/V-scope system at work show the user in the three hyper-instrument areas (it is possible to see in the picture the computer screen with the force field window indicating the position of the user). In this experiment we used three markers - one for each hand and the third for a generic body location. This last one is useful to capture (i) the body position in the force field map (x and y coords), and (ii) the body height position (z coord), e.g., to know if the dancer is standing or crouched. Different hand gesture recognition are defined for each different hyper-instrument. The markers on the hands can be tracked only if the user keeps the hands opened; this is used to control the sound output: the start and stop of sound outputs are obtained by opening and closing a hand, respectively. When the performer is in the center of an area, we obtain the maximum presence of that particular instrument, while the other two are absent. As the dancer moves from one peak to another, a cross-fading effect from one instrument to the other is attained: in general, the output of the three instruments is mixed according to the shape of the force field. The three hyper-instruments (sound synthesis techniques) controlled in real-time in the demo are the following: - in the upper left area, a bell toll synthesis is used (EMU Proteus); - in the upper right area, a formants vowel synthesis is used (running on the IRIS/Bontempi SM1000 hardware); - in the central area, a string orchestral sound synthesis is performed (EMU Proteus).

5.4 The Exoskeleton system

The experimental hardware and software system developed deals with the real-time animation of a human model based on the movements of a real human. In the first phase of the research we spent many efforts in the reconstruction and animation phase, and less in the tracking phase. We worked for a period with the CosTel (Space Coordinates by means of electrical transducers) and MacReflex, two acquisition systems of three dimensional kinematic data similar to V-scope, but designed for use in biomechanics, neurology, robotics, and sport medicine. The main characteristics of these systems were high accuracy, high sampling rate, and high cost. The animation of a human model was carried out by tracking the movements from some relevant points of the human figure, i.e., the positions of the joints of the human skeleton, and calculating the kinematic structure to move the model. The active markers acquisition system have problems to operate on real stage with movements of an actor, who, during a performance, can rotate or change his posture in such a way that became not completely visible from the cameras. This can be a problem in certain circumstances, so we started developing a completely different acquisition device, in a certain sense complementary of the previous systems. The main characteristic such a device, a sort of the exoskeleton, is "modularity":

1. Every joint is a simple rotational joint without joint limits.

2. Composite rotations can be realized assembling different base units (see figure), as in a sort of "Lego"-like system;

3. Each joint is based on a high precision potentiometer, connected to a 16 bit AD input;

4. The complete structure can be weared by the actor, who can freely move on the stage without the limitations described for the CosTel system.

The acquisition rate is about 1 Khz, so the joint measures can be used to reconstruct real movements. Data acquired on the PC are send via sockets to the graphical workstation (SGI Indigo), where are used to animate in real time the model. "Alice", this is the name of the graphical animation, has been developed using the SGI Inventor Toolkit, a graphical tool extremely efficent to realize and real-time control complex kinematic models. The refresh time we obtain is about 20 hz, that is not the optimum, but enough to have a real tracking of the actor movements. The joint angles are also sent to a second workstation for music generation, that in real-time modifies the main parameters (pitch, modulation, timbre and amplitude) of the sound synthesized. One of the first experiments made with this architecture has been the "virtual cello player", where the demonstator used the motions of his hands to ”emulate” a cello player. This is shown in the DIST Miami demo videotape. The future activities involved in this project are toward the realization of an infra-red link between the skelethon and the personal computer to obtain a wireless system, the optimization of the mechanical structure, and on more sophisticated sound/music integration.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.4**
Task title: **Experiments:** **Handwriting-visual control** Partner responsible: NICI Start date: 01/07/94 End date: 30/11/94 Task manager: L. Schomaker Planned resources: NICI: 2 - ICP: 2 - DIST: 2.5 (in man-months)		Sheet 1 of 1 Issue date: 24/04/95

Objective:

To study the pen-driven, symbolical control of virtual object parameters on the experimental platform using Pen Gestures and Handwriting. This concerns experiments addressing timing aspects, as well as representational aspects on the input side (commands, gestures) and the output side (graphical rendering of virtual polyhedrons and the virtual face). Experiments on: Handwriting control of motion and shape; Gestural control of motion and shape; Facial feedback on recognizer status.

Input:

WT 1.1, WT 1.2

Output:

WT 2.8, WT 3.1, WT3.4 – 3.6

Approach:

- Software integration of pen library with experimental platform
- Experiments in handwriting/gestural control of motion and shape
- Experiments in facial expression feedback
- Evaluation of results

Contributions:

NICI software, experiments; ICP virtual face rendering software; DIST human movement expertise

Chapter 6

Handwriting-visual control (WT-2.4)

The effort which was originally planned for this task has been focused mainly in the development of the exoskeleton device, i.e. WT 2.3. Therefore no specific experiment has been performed and the effort has been devoted to the software integration aspect. The existing handwriting recognition system SCRIPTOR, running on a pen-based portable in the PEN-WINDOWS environment has been interfaced with the socket library also used for the WT 2.3 task. Therefore, as the user writes on the electronic paper of the PC, the recognized word and other writing parameters can be made available in real-time to the sound-system and the speaking system. Combined experiments are planned for the following research period.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.5**
Task title: **Experiments:** **Handwriting-speech control** Partner responsible: NICI Start date: 01/07/94 End date: 30/11/94 Task manager: L. Schomaker Planned resources: NICI: 2 - ICP: 2 (in man-months)		Sheet 1 of 1 Issue date: 24/04/95

Objective:

Current recognizer performance is still not optimal, both in the case of speech and in handwriting recognition. However, providing for multimodal user interfacing and easy correction protocols to the user will potentially solve this problem to a large extent. In this work task the new area of combined handwriting and speech recognition is addressed.

Input:

WT 1.1, WT 1.2

Output:

Software for bimodal interaction and for combining recognizer output, experimental results.

Approach:

- Developing a combined speech recognizer and handwriting recognizer setup on the basis of existing technology
- Experiments in Handwriting (cursive script) and Speech recognition
- Experiments in Pen Gestures and Speech recognition
- Evaluation of results

Contributions:

NICI software, experiments; ICP software

Chapter 7

Handwriting-speech control (WT-2.5)

7.1 Introduction

The integration of handwriting and speech recognition offers new possibilities in applications where the users controls a system. More specifically, a user may able to control (parameters) of physical objects - as in teleoperation - or virtual objects - as in graphics or text.

A number of application areas are possible [16]:

1. ink and speech annotation of existing documents
2. pointing by pen, data or modifier input by voice
3. the combined pen/voice typewriter: text input

The interesting functionality is derived from the complementarity of the two human output channels. Pointing to objects by speech is dull, slow and error-prone. Thus for pointing, the pen may be used. Similarly, for symbolic data entry, speech is a fast and natural channel. In text input, the combined recognition of speech and handwriting may considerably improve recognition rates. Originally, the goal was to use the existing handwriting recognition algorithms and methods within MIAMI together with an externally provided and proven speech recognition approach such as the HTK toolkit. However, it turned out that such a method does not run in real time. Because in MIAMI, the goal is to develop highly integrated, fast responding interfaces, the use of "closed" commercial speech recognition boxes is not suitable in most envisaged experiments. For this reason, we will first introduce an existing variant of speech recognition algorithms, the Recursive Markov Model (i.e., not the standard Hidden Markov Model), and show its excellent potential in the integration of modalities. Its basic virtue lies in the fact that 'evidence' for input symbols is built up gradually in (delayed) real time, as opposed to models which require fully completed utterances before the recognition process can start. In many applications, a fast response is essential. In the case of speech-controlled robots, it is undesirable to have a system reacting to the spoken command "STOP" with a delay of more than about one second. If within a given lexicon the sounds `/s/ /t/ /O/` are unique, correct recognition can take place already at that stage in the utterance. The Recursive Markov Model allows for such a response.

7.1.1 Levels of integration

In MIAMI, several levels of handwriting & speech integration will be studied. There are a number of correspondences and differences with the visual & speech integration (lip reading in speech recognition, as studied in WT. 2.2). What both approaches have in common is the fact that symbols are inferred from a combination of (1) speech data and (2) another modality, the latter containing information pertaining to the same speech utterance.

However, the fundamental difference is, that in handwriting & speech integration, there is no natural time synchronisation, as in normal speech & lip reading.

The following levels of integration can be defined:

- Separate handwriting and speech recognizers, combining the lists of most likely output words in a post processing stage
- Separate recognizers are used up to the syllable level. At higher levels the search space is combined.
- Separation between recognizers exists only at phoneme/stroke level.
- A fully integrated single speech & handwriting recognizer

In the next sections we will describe the recognition using the RMM model first, followed by a section on handwriting text and gesture recognition.

7.1.2 The Recursive Markov Model

The Recursive Markov Model (RMM) is an extension of the Hidden Markov Model(HMM) which is specially suited to integrate multiple levels and multiple modalities into a single model. This model can form the basis of a speech- or handwriting recognition system. One of the powerful techniques that can be used within the RMM is State Sharing, which makes use of the shared elements from everyday language. Examples of such shared elements are phonemes, strokes, characters, syllables and words.

Very powerful algorithms exist that can perform training and recognition. These algorithms are extensions to the Forward Backward and Viterbi algorithms that are well-known in the Hidden Markov Model.

Definition of the RMM

Many extension to HMM have been proposed, such as for example duration and language modeling. These extensions deviate more and more from the original model. For all extensions additional parameters have been introduced and more training data was needed. Computation time for recognition and training increases as well.

For a practical speech or handwriting recognition system, the number of parameters will have to be reduced without simplifying the model. This can be done by sharing common parameters. for example, if a library contains the words ’four’, ’fourteen’ and ’fourty’, then these words have the first syllable in common. However, HMM does not contain a technique to combine the parameters of this syllable. In contrast with that, RMM makes such a sharing approach possible. It turns out that the well-known Forward-Backward and Viterbi algorithms, as used in HMM training and recognition can be adapted to the new model.

A HMM consists of a collection of states and a set of transition probabilities. At every frame time the model will change state in accordance with these transition probabilities. Also at every frame time an output symbol is produced, according to another probability law defined on the states.

In RMM states are now allowed to produce more than only a single symbol. This means that a transition does not have to occur every frame time. Elementary states are introduced to be compatible with HMM-states.

Definition of RMM:

- A RMM consists of a collection of states and transitions. Each transition is associated with a parameter, describing its probability of occurrence
- These states can be elementary or non-elementary. elementary states are similar to the states in HMM.
- Elementary states are active during one frame time period, and produce a single symbol
- Non-elementary states are active during one or more frame time periods, and produce one symbol for each active frame time.

Both elementary an non-elementary states need to be further specified

Elementary states:

- A function is specified to describe the probability (or probability density) that some elementary state S produces some symbol. This symbol may be discrete or continuous. Any multivariate distribution function (e.g. Gaussian) is allowed.

Non-elementary states:

- A non-elementary state S consists of a set of child states and a set of transition probabilities stored in matrix A.
- The elements of A only depend on the assigned state S, and not on time t.
- If state S is initiated, a transition is made to one of its child states, according to the associated transition probability from matrix A.

If a transition is made from state S_i to S_j at time t, we say that state S_i has finished and S_j is initiated at time t. If S_i is initiated at time $t = t_0$ and has finished at time $t = t_1$, than we can say that S_i is active in the interval $[t_0, t_1 >$ (t_0 is included but not t_1). In the RMM, transitions do not take time. The production of an output symbol by an elementary state however takes one time unit.

For all states that produce more than one symbol, a further specification is needed. Such a non-elementary state is supposed to be a Markov chain of sub-states. In the same way these sub-states must be further specified if they also produce more than a single symbol. This recursive process ends in elementary states that produce only a single symbol. The name Recursive Markov Model (RMM) refers to this recursive way of splitting states into sub-states. The whole RMM can be considered as a single Root state, containing all other states.

An important feature of RMM is the possibility of State Sharing. This allows for more complex model descriptions without increasing the number of parameters to be trained and it is equivalent to tied transition probabilities in HMM. With RMM the recursive model enables the description of much more complex relations between states, which recursion can easily be used in the training and recognition algorithms. This will be illustrated in the following example.

Suppose an RMM has been constructed for modelling the pronunciation of all numbers between 1 and 999999. The Root state (1-999999), representing all possible numbers, can be divided in 4 sub-states. The first and the last of these, representing the numbers 1-999, are identical. The structures of the Root state (1-999999) and state (1-99) are shown in figure 7.1: Each of the states in this model has its own matrix A. However, because the first and the last child state (1-999999) are identical, it makes sense to form one matrix A which is shared between these two states. In a high-level language like C or Pascal this sharing is easily implemented using pointers. If state (1-999) is decomposed further into smaller states, new identical states will arise. For instance, there may eventually exist states named 'four' in eight different places. These represent the eight different functions of 'four', which are all used in the numbers "fourty-four thousand and fourty-four" and "four-hundred-fourteen thousand and four-hundred fourteen". Because these states not only share their matrices A but also the full lower level state structure, this principle is called State Sharing. All algorithms derived for RMM fully support State Sharing.

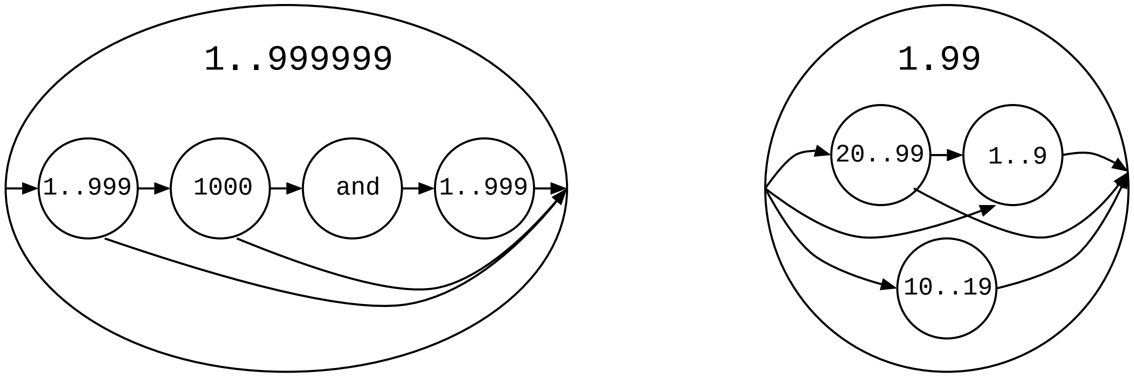

Figure 7.1: An example RMM: modeling the pronunciation of numbers 1 through 999999

7.1.3 Conclusion

We have described an algorithm which is intrinsically well-suited to the task of interactive, bimodal control by speech and handwriting. The software is written in C, and can thus be used directly in creating new interfaces for MIAMI. However, the disadvantage is that some new software development has been done, which was not anticipated at the outset of the project. We have also shown that the algorithm in principle can be used for processing handwriting data. In the next section, we will take the perspective of handwriting recognition.

7.2 Basic Problems of Segmentation in Handwriting and Speech

As soon as one starts to think about the combination of handwriting and speech recognition, the problems of (1) **mutual reference**, and (2) **segmentation** emerge. The problem of mutual reference concerns the fact that the timing of events in the two signal streams is vastly disparate. Even if the user would start writing and speaking a specific word at the same moment in time, there is no simple temporal relationship between the phonemes and the graphemes. Writing (1.5-2s/word, Dutch) is much slower than speaking (300-400ms/word, Dutch). The mapping from phonemes to graphemes (i.e., the process of spelling), is not one to one, and neither is the mapping from graphemes to phonemes (the process of pronunciation). If a mapping from phonemes to graphemes and vice versa is available, pattern matching between speech and handwriting signals can be done, using the abovementioned RMM model or a form of a dynamic time warp matching. In the matching process, the **mutual reference** problem is solved within a single utterance. However, the problem is more complicated than this, because we have to be sure first that two fragments of speech and handwriting have anything to do with eachother at all. This brings us to the next problem: segmentation. Segmentation in this context refers to the isolation of a short-lasting unit of interest from a longer-lasting signal stream of a specific modality. A stream of continuous speech can be segmented in individual words. Similarly, a stream of pen-tip movements can be segmented into words. However, for both modalities, this segmentation is difficult, especially if it has to be based on 'bottom-up' information, without linguistic or other context knowledge. The experiment described below concerns this basic segmentation process for the case of handwriting. The knowledge generated can then be applied for the case of integrated recognition later. For the time being, we will address the case of word segmentation in the production of text only. Figure 7.2 illustrates the problem. It displays the histogram of the widths of white vertical columns in handwritten texts. From handwritten texts containing sentences (six writers, 1100 words), the width of white space columns (dX) between chunks of ink was determined, and a histogram was produced. There is a small dip at about a dX of 1 mm, but it clearly is uncertain whether this is a reliable criterion for deciding if the white space is within or between words, solely from this bottom-up information.

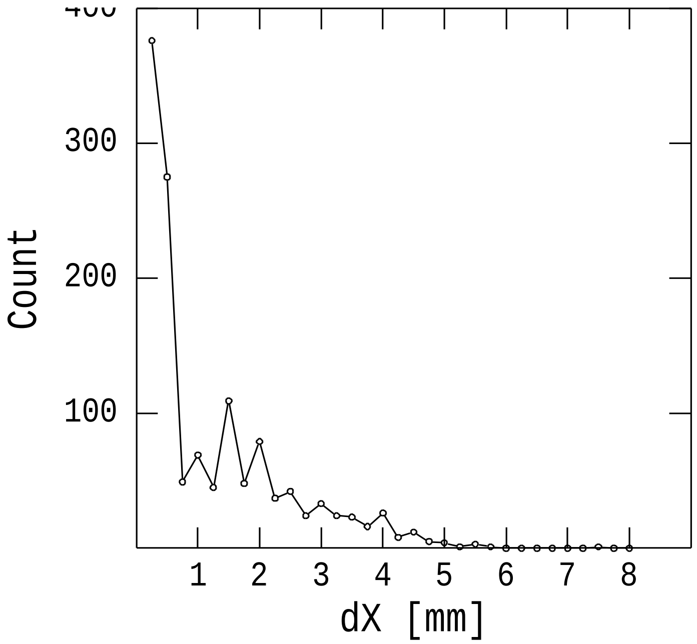

Figure 7.2: Histogram of white space width dX in handwritten text. No clear criterion for within- vs between-word white space can be observed. The histogram is based on 1310 ”pen-down chunks”, in handwritten texts by six writers, 1100 words.

From these observations we can conclude that other sources of information are needed: either from linguistic top-down ”expectation”, or from actions by the user (writer) himself.

7.2.1 User-driven word segmentation

Why segment into words? Current cursive word recognition technology requires a segmentation into words due to the dependence on a lexicon. Since, as we have seen, automatic word segmentation is almost as brittle as the word recognition itself, new pen-driven word segmentation techniques have to be defined. Again, in the case of the keyboard, it is a generally accepted practice to use the ”Enter” or ”Return” key for input validation. A number of pen-driven word segmentation methods are under study. Three methods are considered here:

1. Gesture-driven word segmentation where the user enters a tap on the digitizer on the right of the end of the current word ($> 2cm$);
2. "OK-button" word segmentation;
3. Time-out driven word segmentation.

An experiment was conducted in the form of a "Wizard of Oz" technique, to ensure high recognition rates.

- recognizer: "Wizard of Oz"
- task: copying a text from paper
- three texts of 71, 48 and 61 words
- 18 subjects
- write on 100mm guide line,
  in a 150x18 mm box on SPARCstation-2
- Wacom SD opaque tablet

The word segmentation was done by the user in one of three ways, depending on the condition.

The end of a word is given by:

- pen gesture (a tap with the pen on the paper $> 20mm$ to the right of the last-written word)
- a press on an OK-button (11x7 mm) on the screen
- a time out period in which the pen did not touch the paper (0.6, 1.0, 1.4 s)

Subjects (writers) performed a lowercase text-copying task. There were three texts, randomly distributed over the three conditions, and all subjects copied a different text in all three conditions. Although it was expected that a human reader (i.e., the "Wizard"), who looked at the isolated handwritten words would perform very well, with a close to 100% recognition rate, this was not the case at all. In a pilot experiment, the Wizard response times were well over 2s in most cases, and the human recognition rate was below

85% recognition. This was solved by creating an easy user interface for the Wizard, with a clickable menu of words to be expected from the writer in the other room. Two ”Wizards” were looking at the handwriting to ensure a more reliable response. This setup proved useful. We have same data on the human reading of (cursiv e) handwriting, which will be reported elsewhere in the MIAMI project.

Results are given in Table 1. Results for three time-out values (0.8,1.0, and 1.4s) were combined since differences where statistically negligible. The entered words were off-line truthed and processed by a stroke-based cursive recognition program reported earlier [42]. It was trained on the handwriting of 32 multinational writers and used without extra training for the 18 subjects. The word list for lexical post processing contained the words present in the text to-be-copied. Subjects were allowed to write in their own style, which was mainly mixed cursive and handprint, and often contained small-written capital characters as ”lower case”. The recognizer was originally designed for connected cursive.

	Gesture	OK-button	Time-out	
Text reading time/word	2.0	1.9	1.7 [s]	N.S.
Writing time/word	2.9	2.5	2.5 [s]	p ¡ 0.001
Top word correct	56% (20-81)	62% (21-80)	60% (8-90)	N.S.
Correct word in top five	64% (24-88)	70% (21-88)	67% (13-94)	N.S.
Avg. words/subject	55	53	53	N.S.

Figure 7.3: Table 1. Human reading and writing times, and machine recognition rates as a function of word segmentation method (N=18 subjects)

Statistically, the only significant difference is in word writing time, due to slower writing in the gesture method. This may be due to cognitive overhead in anticipation of the newly learnt gesture movement. From the user-interfacing point of view, the ”OK-button” method has the advantage that it is compatible and consistent with a ”Cancel-button” option, for input which the user considers to be sloppy. In the other two methods, this can only be robustly solved by allowing for post-hoc editing.

Table 2. shows some results on errors and questionnaire answer for the different methods. An error is defined as a segmentation which fails. Examples are: there is a time-out when the writer is not ready; the writer produced more than one or less than one word when clicking on the ”OK” button; or, in case of the gestures, the user forgets to produce the end-of-word gesture, leading to very long waiting times. The time-out condition produces most word segmentation errors.

7.2.2 Conclusion

Both handwriting and speech recognition are most reliable in the case of isolated words. The results of this study show, that for handwriting, the use of an active decision (”OK” or ”end-of-word gesture”) to enter a word into the recognition process, is acceptable. The advantage of this procedure is that it is fully compatible with an integrated recognizer for handwritten and spoken words. In the latter case, the input validation (”OK”) must hold for both modalities, and the same word. Future research will be directed at the integrated recognition, again using a Wizard of Oz setup, until the fast RMM speech recognizer is available.

	Gesture	OK-button	Time out	
Which method was fastest?	9	4	5	[#subj]
Where did you make most errors	4	6	7	[#subj]
Actual #errors	60	60	91	[#words] ($p < 0.05$)
Nwords	1093	1044	1094	[#words]
% error	5%	6%	8%	

Figure 7.4: Table 2. Questionnaire answers and actual errors performed.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.6**

Task title: **Experiments:** **Visual-motorial control** Partner responsible: UKA Start date: 01/07/94 End date: 30/11/94 Task manager: S. Münch Planned resources: UKA: 3 (in man-months)	Sheet 1 of 1 Issue date: 24/04/95

Objective:

Evaluation of performance in object manipulation with direct object manipulation or manipulation through a supervised robot; real and virtual object manipulation; expert vs. laymen as system operator

Input:

WT 1.1, WT 1.2, Experiences from the ESA project IRAS

Output:

Software to be used in WP 3 and WP 4, Results to be reported in WT 2.8

Approach:

- Definition of test parameters and evaluation criteria
- Development of test procedures
- Psychophysical experiments with human subjects (expert vs. laymen)
- Evaluation of results

Contributions:

UKA experiments

Chapter 8

Results of WT 2.6 — Visual-Motoric Control Experiments

8.1 Abstract

Devices with tactile and force feedback (also called haptic feedback) have been developed for more than 30 years, and due to new applications like virtual reality, they become more popular nowadays. However, their effect and influence in standard user interfaces and everyday tasks has not been studied and they are mainly used with special applications. Therefore, it was our intention to investigate the potential benefit of haptic feedback (if there is any) for the common computer user.

For the analysis and evaluation, we have sampled more than 10,000 records of data in more than 100 test sessions with more than 60 volunteers. The results show that haptic feedback can support simple interaction tasks, although there is only minor improvement when it is added to visual feedback. However, tactile and force feedback can replace visual feedback without losing performance, so there is a chance to balance the load on the operator's perception system between visual and haptic feedback with only minor effort. (To avoid misunderstanding: When using the term 'replace' within this context, we are not talking about abolishing the monitor, i. e. providing *no* visual feedback at all, a technique which is investigated for auditive feedback (see, e. g., [37, 36]). Instead, we want to replace visual feedback modes like color change, highlighting, etc.)

8.2 Introduction

Although we talk about the "look and feel" of graphical user interfaces (GUIs), the communication which is directed from the computer to the user is usually dominated by the *look*, whereas the *feel* is not used at all. In other words, in today's GUIs the user still uses a keyboard and a mouse as input devices (man → machine) and perceives nearly all information via the visual channel (machine → man), i. e. by looking at a monitor [12].

This type of communication has recently started to become more "balanced", mainly because of new interaction techniques which emerge from multimedia as well as virtual reality applications. In both domains, new communication methods which are not only based on vision but also include sound and tactile information are investigated, and new devices which allow to address the corresponding senses are developed. When these are included in GUIs, we talk about *multimodal interfaces*.

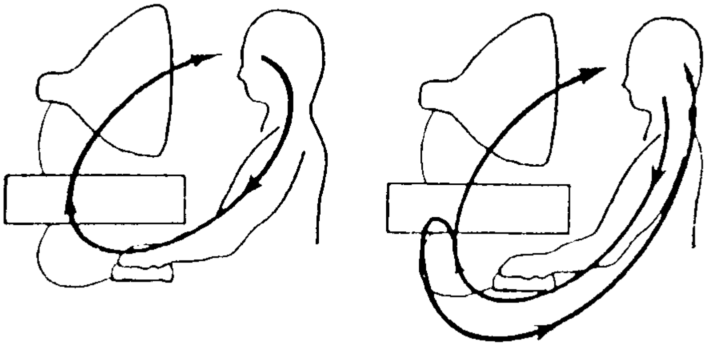

Figure 8.1: Traditional MMI scheme (left) and bimodal communication with visual and haptic feedback (right). (Taken from [20])

One of the most interesting questions that arise from these developments is how the typical computer user can benefit from them in everyday applications, i. e. how standard GUIs can be extended to multimodal UIs. Although tactile and force feedback alone have been addressed by a number of researchers (see, e. g., [11, 20, 1, 2, 28]), their effect in standard GUIs has not been investigated in depth, and most evaluations are based on very small data sets. Therefore, we have designed, implemented, realized, and evaluated several experiments directed towards the

- analysis and evaluation of the effect of different input devices as well as the
- analysis and evaluation of input devices with tactile and force feedback.

Simple, but typical interaction tasks (like *position* a cursor, *drag-and-drop* of an object, or *resize* operations) have been chosen for the experiments in order to simplify the measurements of relevant data and to perceive transferable results. In addition, two prototypes of input devices with feedback have been built and used for the experiments. The principle goal of our work is to extend the human-computer interaction in such a way that the several unidirectional communication channels which are generally used (in parallel) today will be integrated in a multimodal, bidirectional interaction scheme. The combination of input and output capabilities in one device seems to be a promising approach (see fig. 8.1).

The remainder of this chapter is organized as follows: first, we will describe the background of our research, introduce the terms tactile and force feedback, and present two devices which have been built. Next, our hypotheses, the test conditions, and the experiments that have been carried out will be described in detail. The main part of this chapter includes the evaluation and interpretation of the results obtained from the experiments. Finally, we will draw some conclusions how standard GUIs may benefit from tactile and force feedback, respectively, thus leading to more powerful multimodal interfaces.

8.3 Human-Computer Interaction (HCI)

The two basic processes which are involved on the side of the human user in any HCI task are *perception* and *control*. In addition, on both sides of the communication cognitive aspects play a fundamental role which has not been considered sufficiently in most applications. Our basic model of HCI which comprises an extrinsic interaction loop as well as an intrinsic perception/action loop is depicted in fig. 8.2.

Tactile and force feedback are important for both, the extrinsic as well as the intrinsic loop. The latter one comes into play when considering the socalled “breakaway force” that can be experienced when pressing a key or a button, or in a vision process where a coherent image is reconstructed from a series of saccades (rapid, but controlled eye movements) and fixations, e. g. when reading this text.

In this chapter, we will concentrate mainly on the first, the extrinsic loop. Although most devices provide *some* kind of tactile/force feedback (e. g. the breakaway force, see above), we will consider only those devices which have been *especially* designed for that purpose, like the Exoskeleton [25] or the PHANToM [33], to name just two of them[1].

[1] For a more detailed discussion of HCI and devices with haptic feedback, please have a look at MIAMI's 'Taxonomy Report' [41].

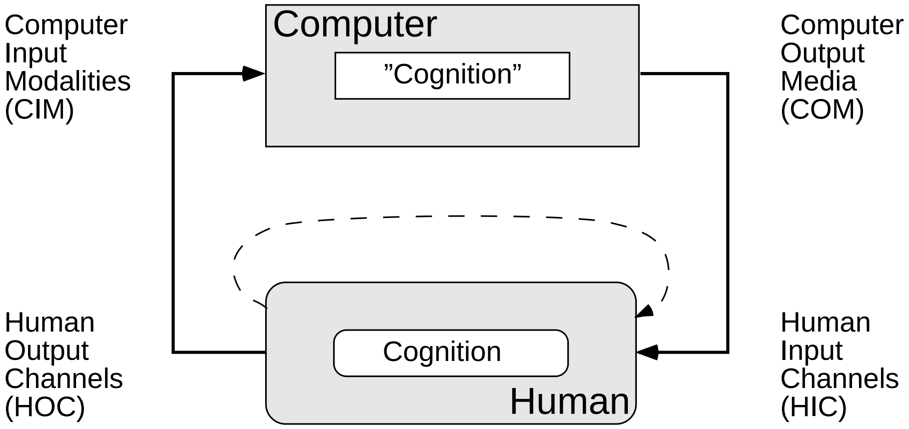
Computer Input Modalities (CIM)
Computer
"Cognition"
Computer Output Media (COM)
Human Output Channels (HOC)
Cognition
Human
Human Input Channels (HIC)

Figure 8.2: MIAMI's model for the identification of basic processes in HCI, including the extrinsic interaction information flow (solid line) and the intrinsic perception/action loop (dashed line).

8.4 Tactile and Force Feedback

Before we are going to present our experiments and the results achieved, we will motivate the topic of this chapter a little bit further.

8.4.1 Somatic senses

The distinction between *tactile* and *force* feedback is somewhat arbitrary (sometimes it is impossible to tell whether a feedback is tactile or force, therefore we will use the term 'haptic' in the following if we speak about both, tactile and force feedback), and both are related to the *somatic* senses. Some authors use 'somatic' as a synonym for the sense of touch, whereas others distinguish between a number of somatic senses, e. g. the senses of pressure, touch, vibration, cold, warmth, position, and force [22]. Most of them are not relevant for our concerns, so we will take into account only the sense of touch and the sense of force in the following sections.

There exists a number of ways to address these senses, e. g. pneumatic, vibrotactile, electrotactile, or functional neuromuscular stimulation [43]. Because we want to learn about the effects of tactile and force feedback in *standard* GUIs, not in special applications, a simple and cheap solution had to be found. We decided to use a kind of vibrotactile stimulation plus electromagnetic brakes with our mouse and to apply forces by servo motors with our joystick[2].

8.4.2 Devices

Because most input devices with haptic feedback are either too simple, not available on the market, or very expensive (from US $10,000 up to more than US $1,000,000), two input devices with haptic feedback have been designed and built for MIAMI:

Mouse with haptic feedback: Following the idea of Akamatsu and Sato [1], a standard 2-button mouse for an IBM PS/2 personal computer has been equipped with two electromagnets in its base and a pin in the left button (fig. 8.3). For input, the standard mouse driver is used; for output, the magnets and the pin can be controlled by a bit combination over the parallel printer port by our own software, so that the magnets will attract the iron mouse pad and the pin will move up and down. Both magnets and the pin can be controlled independently. In order to make the mouse usable with our SGI workstation, a communication between the PC and the workstation is established over the serial communication line. In principle, any standard mouse can be easily equipped with this kind of haptic feedback.

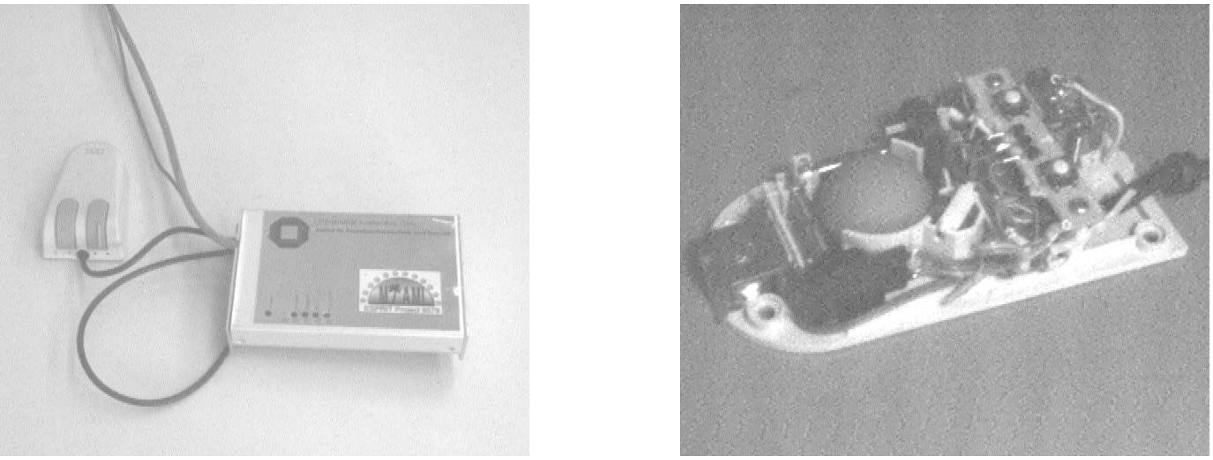

Figure 8.3: The mouse with haptic feedback which has been built for the experiments in WT 2.6. The left picture shows the mouse as it has been used, the right one shows its 'inner life'.

Joystick with force feedback: A standard analog joystick has been equipped with two servo motors and a micro controller board, see fig. 8.4. Communication between the

[2] Again, for a more detailed discussion of the somatic senses and how to address them, refer to [41].

joystick controller and a computer has been realized over a serial communication line. The joystick's motors can be controlled in order to impose a force on the stick itself, thus making force reflection possible.

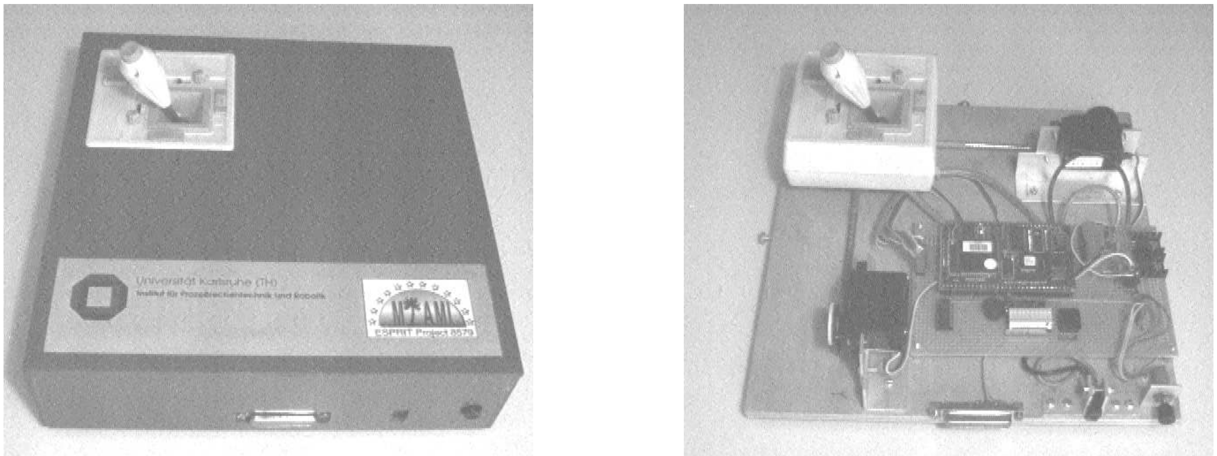
Universität Karlsruhe (TH)
Institut für Prozeßrechentechnik und Robotik
MIAMI
ESPRIT Project 8579

Figure 8.4: The joystick with force feedback which has been built for the experiments in WT 2.6. The left picture shows the joystick as it has been used, the right one shows its 'inner life'.

Another device, the , has been bought. It is the cheapest device on the market (< US $150) with tactile feedback, although in this case there is only a vibration of the device itself. For our experiments, the following five devices have been used: the mouse with haptic feedback, the joystick with force feedback, the , and two 6D input devices[3], the and the (see fig. 8.5). An interesting question is *how* the feedback provided by these devices can be used — considering the hardware as well as the software — in different applications.

Naturally, each device has its own characteristic 'behavior'. For instance, the and the are both 6D input devices, but when using the , the user's hand can rest on the table. In addition, the is much more sensible and reacts faster to the user's inputs.

A much more important aspect is the movement of the cursor relative to the user's action with the device. One way to compensate for these differences is to use a normalization function with the different device drivers, so that the cursor moves with the same speed independent of the device. In our opinion, such a normalization is not 'fair', because the common user does not have the chance to change the characteristics of a device.

[3] The 6D devices have been used to compare the user's performance in 3D tasks with the 2D devices, but they don't provide any haptic feedback. For that reason, they are not mentioned in all the evaluations of data.

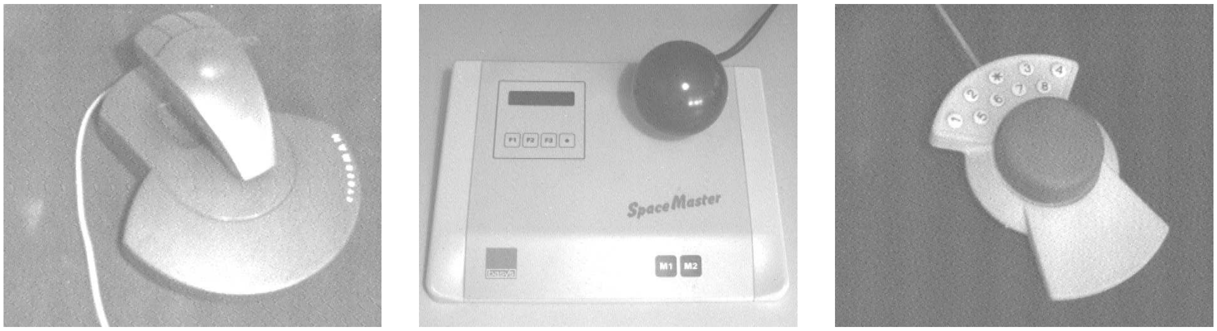

Figure 8.5: The three other devices which have been used for the experiments of WT 2.6: the (left), the (center) and the (right).

Therefore, we have tried to optimize the characteristics of each device and have integrated the parameters from this optimization phase in the intermediate layer of our Meta Device Driver (see fig. 8.8). This is one reason for the differences in execution times for the different devices[4], and we will not compare one device to another but only evaluate the differences between the various feedback modes (see section 8.6). For a comparison of input devices itself, see, e. g., [3, 4, 18, 32, 31, 34].

8.4.3 Feedback modes

Obviously, the devices which are equipped with haptic feedback capabilities realize this feedback in completely different ways. The *mouse with haptic feedback* uses two electromagnets as a kind of "brake", i. e. if a current is applied to them, the movement of the mouse will be rendered more difficult for the user, depending on the current. In addition, a pin in the left mouse button can be raised and lowered frequently, causing a kind of *vibration*. This will motivate the user to press the button.

Although in principle the current of the magnets and the frequency of the pin vibration can be controlled continuously, this will usually not be used, therefore we call this kind of feedback *binary*. Logitech's can also generate binary feedback only: If a special command is sent to the device, it starts to vibrate. Again, the frequency and duration of the vibration can be controlled with parameters, but a continuous feedback is not possible.

The situation changes completely when the *joystick* with force feedback is considered. Here, two servo motors control the position of the joystick, thus allowing a continuous control in the x/y-plane. When the user pushes the stick, but the servo motor tries to

[4] Another reason is that most user's are familiar with the mouse, some have used a joystick before, but nearly no one of our test persons had any experience with the other three devices.

move it in the opposite direction, the user gets the impression of force feedback, because the movement becomes more difficult or even impossible.

With respect to the software, several different possibilities exist to give the user a visual and/or haptic feedback. Visual feedback is used by every window manager, e. g. the border of a window is highlighted when it is entered by the mouse cursor. In order to study the effect of haptic feedback, various *feedback schemes* have been developed. Two of them will be described in more detail below:

1. The first scheme is used for simple objects in 2D. Fig. 8.6 shows a typical scene of the first task (see section 8.5.3), where the haptic feedback is launched whenever the cursor enters the target region. The same scheme can be used for obstacles, too.

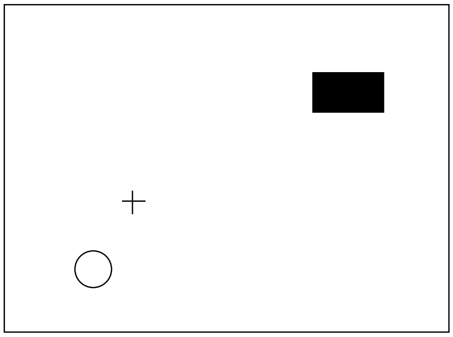

Figure 8.6: A typical scene which is used for simple 2D positioning tasks with visual and haptic feedback. The circle marks the *start position*, the black object is the *target*, and the cross indicates the *cursor*.

   For the mouse, the magnets and the pin (or a combination of both) may be used. For the , the vibration is switched on. For the joystick, things get more complicated. A force function, like the one shown in the left part of fig. 8.7, needs to be implemented. In this case, the user "feels" some resistance when entering the object, but if the center is approached, the cursor will be dragged into it.

2. The second scheme is applied to objects in 3D space which are treated as obstacles, e. g. walls in a mobile robot collision avoidance task. The magnets of the mouse can be used to stop further movement against an obstacle, and the 's vibration can be switched on for the same purpose. Again, the joystick has explicitly to be programmed with a predefined, parametrized function in order to prevent the mobile robot from being damaged. The right picture in fig. 8.7 shows the principle implementation of this function.

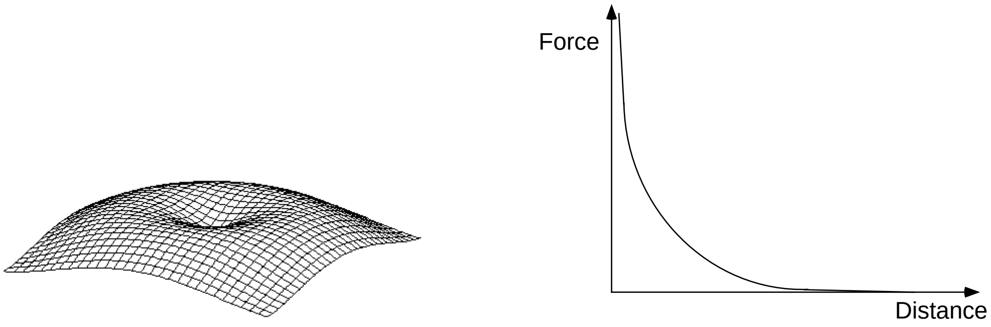

Figure 8.7: Two force function which may be applied to objects in order to control the joystick. The first one (left) is useful for objects in 2D space, whereas the second one (right) may be applied to objects in 3D space. Here, the x-axis denotes the distance between the cursor and the object, and the y-axis the force applied by the servo motors.

8.4.4 The Meta Device Driver (MDD)

In order to make the usage of the different devices as easy as possible, a general *Meta Device Driver* (MDD) has been developed for all tools (see fig. 8.8). The parameters which are sent to the devices follow the same structure as well as the values received from them. This concept has been developed in order to hide the specific characteristics of a device behind a common interface.

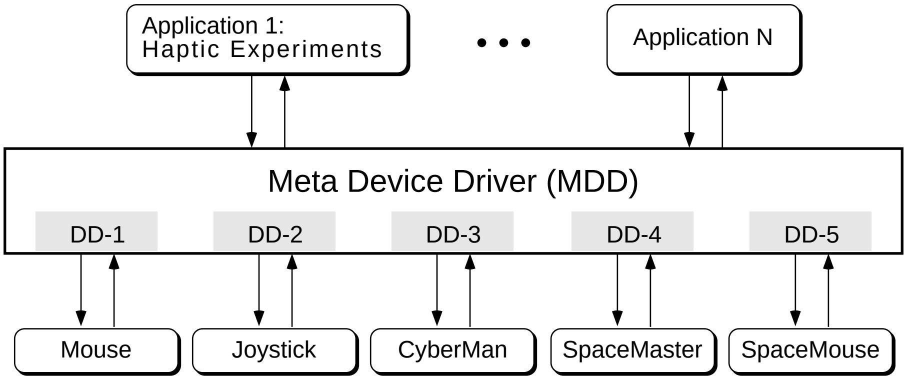

Figure 8.8: Schematic description of a general multimodal device driver, the Meta Device Driver

The MDD has been realized as a C++–library and can be linked to any application. If more devices will be available, it can easily be extended. Thus, the MDD becomes an important part of a multimodal system, because the *blending of modes* and the selection of different input devices is possible for the user at any time of the interaction process.

8.5 Experiments

Before experiments can be carried out, one has to define a number of hypotheses (*What* do we want to find out?) and a number of test conditions (*How* do we want to find it out?), like dependent and independent variables, the number of combinations, the number of subjects etc. The next step is the design and implementation of the experiments itself.

8.5.1 Hypotheses

The hypotheses which we wanted to prove are, among others, that

- haptic feedback will reduce the execution time and the accuracy in simple 2D positioning tasks, if the same device is used with and without haptic feedback;
- haptic feedback will reduce the execution time and increase the accuracy if the target region is very small;
- the (relative) changes in execution time and accuracy will be independent of the angle and distance to the target region;
- the changes described above are more significant if the objects are not highlighted when they are reached by the cursor, i. e. if no visual feedback is present;
- Fitts' law (see below) will hold for input devices with haptic feedback as well.

Fitts' law [19] states that the *movement time* (MT) of a target-oriented movement to an object with width W and distance D depends linearly on the *index of difficulty* (I_D):

$$\begin{aligned} MT &= a + b * I_D, \\ \text{with} \quad a, b &= const., I_D = \log_2(2D/W) \end{aligned}$$

This original version of Fitts' law has been extended for two-dimensional tasks (e. g. [31]), and some efforts have been done to apply it to 3D interactions as well.

8.5.2 Test conditions

In order to get sufficient sample data, comprehensive tests with a large number of subjects have to be carried out. Otherwise, statistical errors might be introduced and the results obtained might not be transferable. Because we did not expect our data to be normally distributed (which is usually an important supposition for the evaluation), our intention was to collect sufficient data. Following Bortz [8], if each combination of variables is tested at least 15 times, the evaluation is more or less admissible even if the distribution is far away from normal.

Unfortunately, the number of experiments grows which each additional independent variable. A simple example taken from one of our tasks will illustrate the situation:

> In the first task (see below), we wanted to modify the size, the distance, and the angle of an object which should be hit with the cursor. Of course, we also had to test all the different feedback modes which were supported by the devices. The following calculation shows the number of possible combinations and the number of tests which became necessary.
> For the task described above, we took three different angles (#Angles $\theta_D = 3$), three distances (#Distances D = 3), and used objects with three different sizes (#Sizes S = 3), therefore we needed $\theta_D * D * S = 3 * 3 * 3 = 27$ different scenes (graphical setups). We wanted to test each of our five devices with all combinations of feedback modes, which lead us to a total of 20 different device/feedback mode combinations (#Modes = 20). The number of test combinations (#Combinations) follows from these two values: $\#Scenes * \#Modes = 27 * 20 = 540$. Because every test had to be repeated at least 15 times (#Repetitions = 15) (see above), the total number of tests became $\#Combinations * \#Repetitions = 540 * 15 = 8,100$, which is a rather large number which corresponds to only one task!

In the end, we have performed more than 100 sessions with more than 60 volunteers and have collected over 10,000 data samples, including execution time, accuracy, and trajectory of cursor movements. The tests have been carried out in an isolated room to reduce disturbances to a minimum. Although the workstation has been connected to our local network, during the experiments no other processes have been run.

The subjects have either performed task #1 and task #3 or task #2 (see section 8.5.3). First, they were given an introduction where they got accustomed to the devices (between 15 and 25 minutes). In the second phase, the first group had to do 126 positionings for task #1 with a short break after 63 had been finished and repeat task #3 eight times in the end. The second group had to do 72 drag-and-drop operations in 3D space (task #2) with a short break after 36 operations had been finished. Finally, the volunteers have been asked for their subjective impression of the devices and feedback modes. The whole test for one subject took about 75 minutes.

The device/feedback mode combinations have been randomized as well as the different scenes in order to avoid training effects. Each combination has been used seven times in a row in task #1 and four times in a row in task #2. Each subject has exactly used each combination once.

It is important to mention that we have not measured reaction but only execution times, i. e. the timer has been started when the subjects moved the cursor out of the starting position, so that they had enough time to study the scene.

8.5.3 Tasks

In order to prove our hypotheses, we have designed and implemented the following tasks. The number after each independent variable indicates the cardinality of this parameter, the number of combinations has to be multiplied with '15' (the number of repetitions) to get the total number of tests.

1. **Positioning in 2D:** A cursor had to be placed in a 2D plane as fast and accurate as possible at a rectangular region, the *target*. This typical task (selecting something by clicking on it) can be found in every standard GUI.
   Visual feedback (VF): color change of target upon entering.
   Haptic feedback (HF): Mouse: rising of the pin, setting the magnets under current; : vibration; Joystick: application of a force function.
   Independent variables (IV): size of target region (3), angle from start position to target region (3), distance from start position to target region (3), devices and feedback modes (20).
   Number of combinations (NoC): $3 * 3 * 3 * 20 = 540$
   Dependent variables (DV): execution time, accuracy.

2. **Drag-and-drop in 3D:** This task consisted of two subtasks: first, the cursor had to be moved inside a cube (the *object*) where a button had to be pressed in order to 'grasp' it. Then, the cube had to be dragged inside a larger sphere (the *target*) where the button had to be released.
   VF, HF: see “Positioning in 2D”.
   IV: object size (2), relative position between start position and target region (4), devices and feedback modes (18).
   NoC: $2 * 4 * 18 = 144$
   DV: time until button press, time until button release, accuracy.

3. **Resizing of windows in 2D:** Four partly overlapping windows have been presented to the subjects. The task was to resize all of them with the following two constraints:

(a) The background should be covered by the windows as much as possible.

(b) The region of overlapping windows should be reduced to a minimum.

In other words, the idea was to fill the space as precisely as possible with the four windows. For resizing, the users could drag the windows' borders (unidirectional) or corners (bidirectional). This task has been performed with the mouse alone in order to reduce the number of tests. The mouse has been selected because it is the most important device with respect to standard interfaces and it supports all three feedback modes, visual, force, and tactile.
VF: color change of border or corner upon entering.
HF: Mouse: rising of the pin, setting the magnets under current.
IV: feedback modes (8).
NoC: 8
DV: total execution time, size of uncovered region, size of multicovered region (overlapping windows).

A typical setup for a scene in 2D is shown in fig. 8.6. Here, we have used 'plain' graphics. In the 3D task, the cursor is realized as a cross hair and the objects are placed in a kind of 'virtual box'. By adding a grid to the walls and rendering the scene with several light sources which produce shadows on the wall, it is much easier to find out the exact position of an object and to control the cursor[5].

8.5.4 Hard- and software

All experiments have been carried out using a Silicon Graphics Indigo workstation with R4000 CPU (100MHz), 64 MByte main memory, and GR2-XZ graphics system (24 bit graphics with Z-buffer). The operating system has been IRIX 5.2, and the experiments have been implemented using OpenInventor 2.0 for the graphics, PVM 3.3 for the communication, and Tcl/Tk (7.3/3.6) for the GUI. The communication with the devices has been realized by using standard functions for the serial line.

The general software structure which has been used for all experiments carried out is shown in fig. 8.9. The user controls the application via the *user interface* and by using one of several devices' *motor control*. He/she receives different kinds of feedback via both, the *visual* and the *somatic* perception channel. The MDD introduced earlier (see fig.8.8) which is realized as a library has been integrated in the application. It communicates with

[5] Unfortunately, a picture of the 3D scene (screendump) can not be provided because the contrast is too bad in blackandwhite or greyscale.

the specific drivers for each device by using a socket communication realized with PVM (Parallel Virtual Machine).

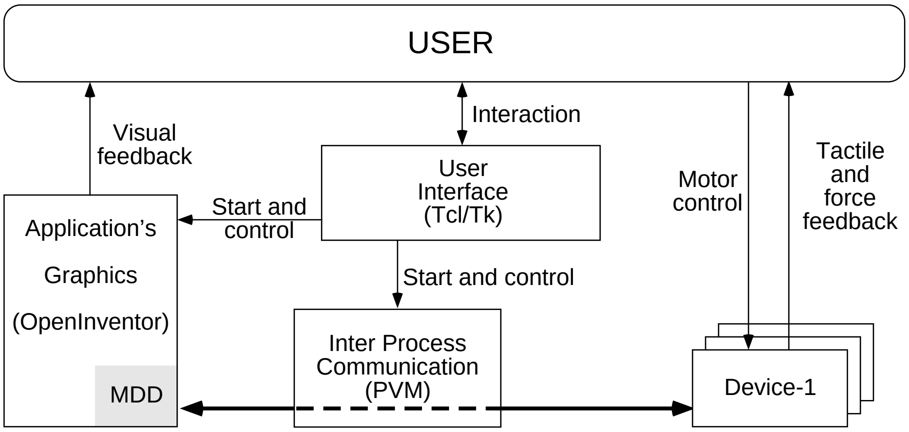

Figure 8.9: Software structure of visual-motoric interaction tasks. The tools which have been used to realize the experiments are denoted in braces "()".

8.6 Results

The evaluation of the collected data revealed a number of interesting results which will be presented in the following subsections. In short, not all hypotheses could be proved. The most important result is that the extension of a graphical system with haptic feedback does not improve the performance significantly, but that in some cases the visual feedback can be completely replaced by haptic feedback without any drawback.

8.6.1 Evaluation method

For the evaluation of our data, we have used the *Statistical Analysis System (SAS)*. The main aspects which have been investigated are the influence of the feedback mode with respect to mean execution times and accuracy and the relative differences between the different modes. In order to evaluate these data, we have used the *General Linear Model (GLM)* and ran the Scheffe test on a 5% level (i. e., the test revealed whether a hypothesis could be proved with a probability of at least 95%).

8.6.2 Results of task #1

The following diagrams will help to answer the questions

- if haptic feedback can *replace* or *support* visual feedback and
- if there is an advantage at all if visual feedback or haptic feedback is provided.

Fig. 8.10 shows that although visual feedback does improve the execution time of simple positioning tasks in 2D, the speedup is not significant.

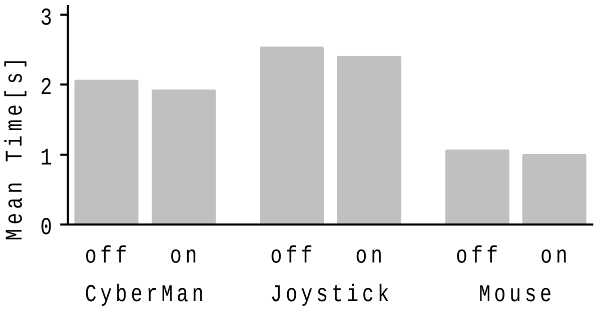

Figure 8.10: Mean execution time for task #1 with and without visual feedback. When the variables θ_D, D, and S are not considered, there is no significant improvement through visual feedback alone.

A more general question is whether the feedback mode has any significant influence to the execution time of task #1 at all. Table 8.1 shows that this is the case for three of five devices, namely the mouse, the , and the .

Device	F-value	p
Mouse	$F(7, 3244) = 2.23$	$p < 0.0296$
Joystick	—	—
	$F(3, 1629) = 9.63$	$p < 0.0001$
	$F(1, 818) = 3.20$	$p < 0.0742$
	—	—

Table 8.1: Significance of the feedback mode for task #1

In fig. 8.11, it can be seen that tactile feedback will improve the positioning time in 2D significantly by about 10% for the Mouse ($F(1, 816) = 6.33, p < 0.0121$) and 29% for the ($F(1, 816) = 16.07, p < 0.0001$), respectively.

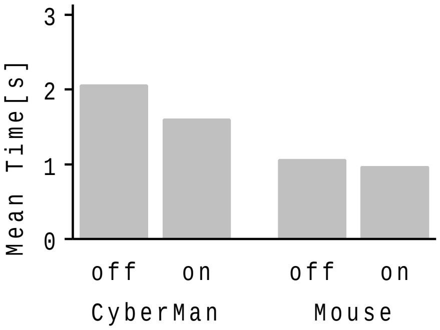

Figure 8.11: Mean execution time for task #1 with and without tactile feedback. When the variables θ_D, D, and S are not considered, there is a significant improvement through tactile feedback alone.

In combination with the results shown in fig. 8.10, it seems to be reasonable to replace visual feedback with tactile feedback. The differences between both modes are depicted in fig. 8.12. Surprisingly, for the the tactile feedback alone is much better than the visual feedback alone (about 20%, $F(1, 816) = 10.74, p < 0.0011$).

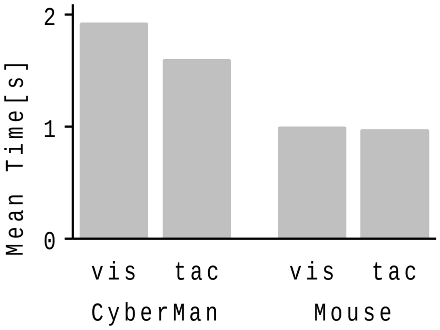

Figure 8.12: Tactile feedback is superior to visual feedback in simple 2D positioning tasks.

Will the results be even better when both feedback modes are used in combination? Unfortunately, our data does not indicate any further improvement. However, two results are worth mentioning here because they have been achieved with the mouse which is the most important device for standard tasks:

1. from all 'single feedback modes', the tactile feedback led to the shortest execution times; and
2. the best results have been achieved with a combination of all feedback modes (see table 8.2).

Mode	N	Mean [s]	StdDev [s]
no	409	1.05906	0.58657
v	405	0.99020	0.54806
f	407	0.98720	0.47270
v, t	405	0.97665	0.47921
f, t	405	0.96704	0.48565
t	409	0.96547	0.47100
v, f	407	0.96039	0.46182
v, f, t	405	0.93223	0.46181

Table 8.2: Execution times for the different feedback mode combinations for task #1, achieved with the mouse. N is the number of data records, Mean is the mean execution time, and StdDev is the standard deviation. The different modes are **v**isual, **f**orce, and **t**actile feedback.

One of our hypotheses ("that the (relative) changes in execution time and accuracy will be independent of the angle and distance to the target region") was completely contradicted by our data, see the following figures 8.13 and 8.14.

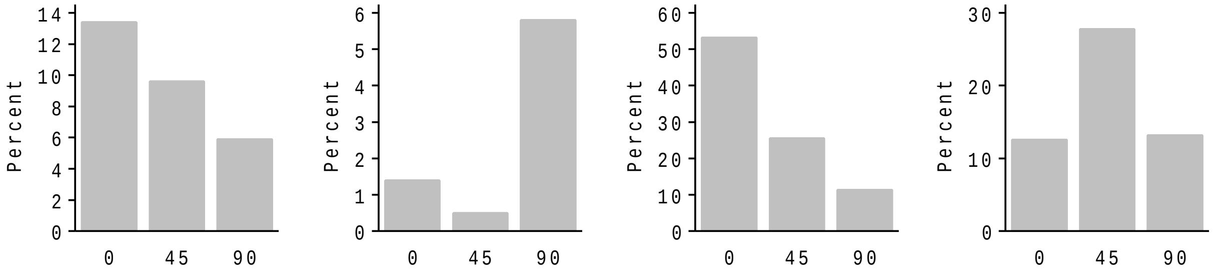

Figure 8.13: From left to right: Relative speedup of execution time from no to tactile (1) and from visual to tactile (2) feedback for the mouse and the (3 and 4), not considering the variables D and S. The x-axis shows the different angles θ_D (0 = 'east', 90 = 'north').

At the moment, we are not able to explain the cause for the differences in different angles and distances, but with one exception the tactile feedback was the fastest mode compared to 'no feedback' and 'visual feedback'.

In the paragraphs above, we have only evaluated our data with respect to execution time, but we also wanted to find out something about the accuracy. Which feedback mode does support the positioning of the cursor best? Surprisingly, the best results for the mouse have been achieved when there was no feedback at all (see fig. 8.15). One possible explanation is that the subjects used more concentration in this case (the execution time increased), whereas otherwise they relied too much on the feedback. Similar results can be found in the left part of fig. 8.16 (), and only the accuracy of the joystick could be

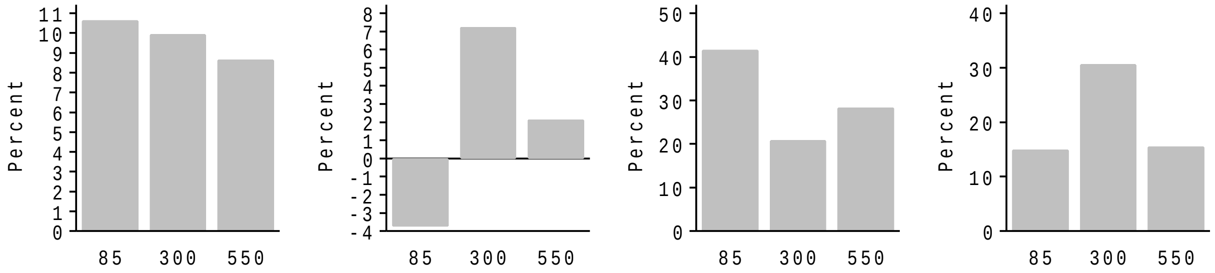

Figure 8.14: From left to right: Relative speedup of execution time from no to tactile (1) and from visual to tactile (2) feedback for the mouse and the (3 and 4), not considering the variables θ_D and S. The x-axis shows the different distances D in pixel.

improved (right part of fig. 8.16). However, it is also important to notice the *range* of the error rate, which is very high for the (from 7.9% to 21.8%, with a mean error rate of 14.5%) but very low for the mouse (1.9% – 5.5%, mean = 3.7%).

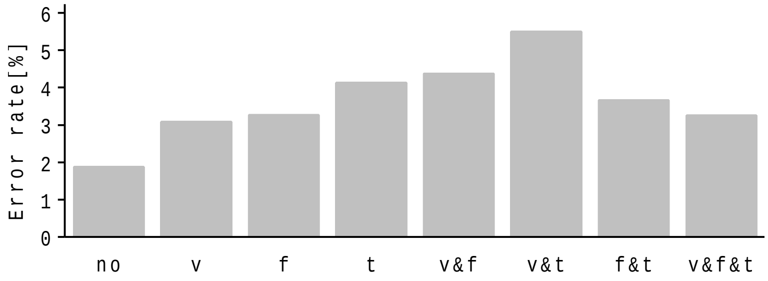

Figure 8.15: Error rates (relative misshits) for the mouse in 2D, separated by feedback mode combinations (v=visual, f=force, t=tactile).

An interesting question is whether the results hold when only small target regions are considered, i. e. when the task gets more difficult. The results show that the numbers are very similar to those where all target region sizes have been taken into account. However, the improvement introduced by the different feedback modes is even larger.

Until now, we have only considered *tactile* feedback, but we have also used devices which provide *force* feedback. For the mouse, we have found minor improvements if force feedback is used, but for the joystick the execution time has increased. Unfortunately, the results are not significant here. Similar results have been found when force feedback is compared to visual feedback.

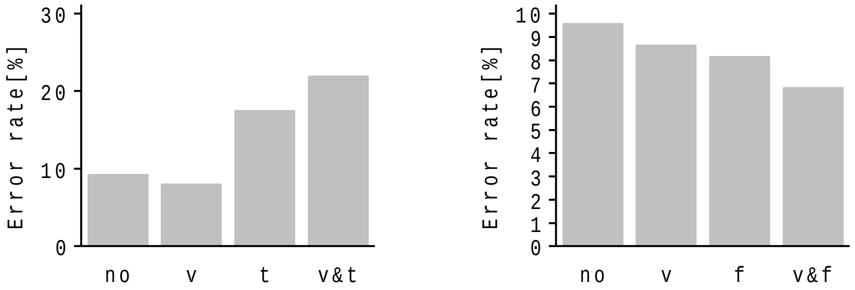

Figure 8.16: Error rates (relative misshits) for the (left) and the joystick (right) in 2D, separated by feedback mode combinations (v=visual, f=force, t=tactile).

8.6.3 Results of task #2

Task #2 (drag-and-drop in 3D) has been separated in two subtasks: first, the cursor had to be positioned in the object in order to grasp it; then the object could be dragged in the final (or target) position. The execution times for the *positioning* and the *collecting* phase will be evaluated separately.

The most interesting questions which our data will answer are:

- Is the feedback mode significant for the positioning and/or the collecting time?
- Is there a significant difference between the four modes a) no feedback, b) visual feedback, c) haptic feedback, and d) visual and haptic feedback, and if so, which mode is best?

Device	F-value	p
Mouse	$F(7, 1151) = 4.73$	$p < 0.0001$
Joystick	$F(3, 603) = 4.19$	$p < 0.0060$
	—	—
	$F(1, 289) = 4.11$	$p < 0.0434$

Table 8.3: Significance of the feedback mode in the positioning phase of task #2

Table 8.3 shows that the feedback mode is indeed significant for three of four devices, with the exception of the where the sum of the squared errors has been too large to reveal significant results. In addition, it is important to notice that the standard deviation for the is extraordinarily large (e. g., 9.14sec with a mean execution time of 13.84sec for tactile feedback alone).

As expected, even the positioning of the cursor took much more time than in 2D. Most of the test persons found it difficult to determine the exact path of the cursor in advance, and the movements were usually oriented along one axis at a time. Therefore, we expected the different feedback modes a much more valuable help than in 2D, which could be proved partially only by our data (see fig. 8.17).

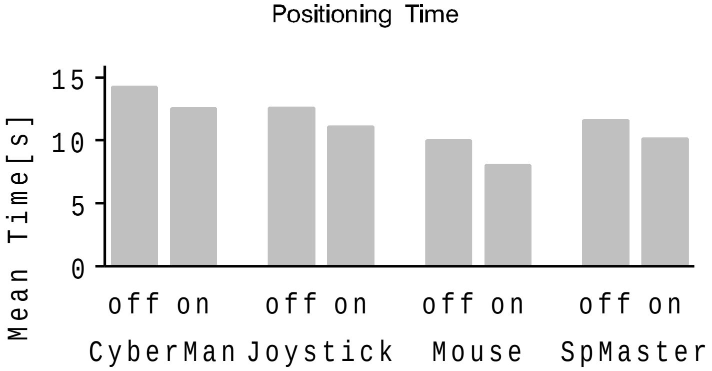

Figure 8.17: Mean execution time for task #2 with and without visual feedback. When the object size and its relative position is not considered, the differences between the two modes are significant for the mouse, the joystick, and the .

In contrast to the task performed in 2D, the execution time was significantly reduced for the mouse (about 24%, $F(1, 294) = 11.26, p < 0.0009$), the joystick (about 14%, $F(1, 300) = 4.39, p < 0.0370$), and the (about 14%, $F(1,\ 289) = 4.11, p < 0.0434$) if visual feedback has been provided.

Again, we have to ask whether haptic feedback is superior to visual feedback and which combination of feedback modes is the best. Fig. 8.18 shows that the situation is in principle the same as in 2D (see fig. 8.11 and fig. 8.12).

Tactile feedback is significantly faster than no feedback for the mouse (by about 24%, $F(1, 286) = 10.97, p < 0.0010$), but not for the (i. e., it is a little bit faster, but the speedup is not significant). Here, the visual feedback yields the best results, whereas with the mouse the positioning times are nearly identical (tactile feedback: 8.01044sec; visual feedback: 8.01812sec). The combination of both modes is best for the mouse (7.74093sec), but not for the where visual feedback alone was the fastest mode.

Interestingly, the joystick's performance has been decreased when the force feedback mode has been used (13.4252sec), whereas visual feedback alone led to the fastest positioning

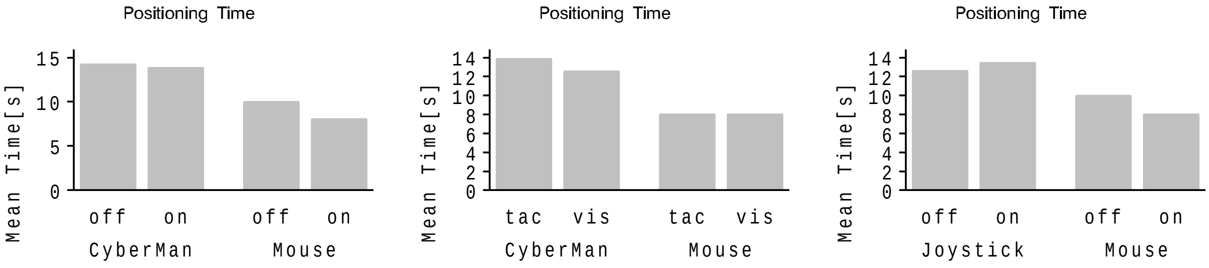

Figure 8.18: Comparison of mean execution time for the positioning phase of task #2. Left: tactile vs. no feedback. Center: tactile vs. visual feedback. Right: force vs. no feedback.

times (11.0700*sec*). At this point, it is not clear whether visual feedback is really superior to force feedback or if the delay is due to the additional calculations for the force feedback performed by the joystick's micro controller.

In the second phase of this task, the subjects had to drag the objects inside a sphere. An error occurred when the object was released while its center has been in the sphere but the object was not fully covered by the sphere. If the object's center was outside the sphere, the subjects could grasp the object and try to drag it inside the sphere again.

In this phase, the performance was best supported by visual feedback (see fig. 8.19), but surprisingly the improvement with respect to the execution time has not been significant. Even worse, the feedback mode for any of our devices could not be proved to be significant during the collecting phase.

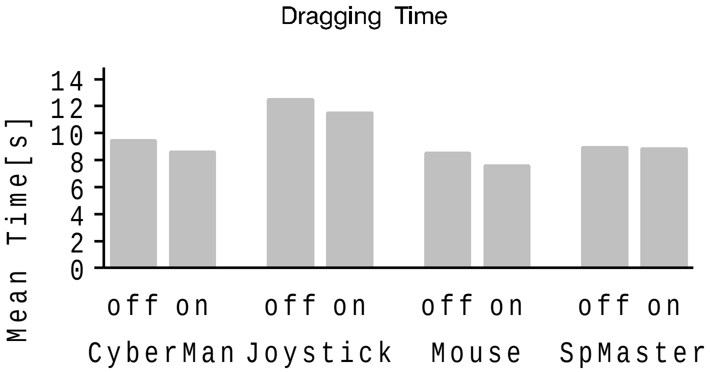

Figure 8.19: Mean execution time for the collecting phase of task #2 with and without visual feedback.

If there is no improvement with respect to the execution time, could at least the error rate be reduced with visual or haptic feedback or a combination of both? Fig. 8.20 shows that this is indeed true for the most interesting device, the mouse. The error rate without any feedback is 9.5%, and it drops down to 5.8% for visual, 6.8% for tactile, and only 3.6% for a combination of visual and tactile feedback. Force feedback alone has not improved the accuracy (error rate: 10.1%), which is consistent with the subjective impressions of our test persons.

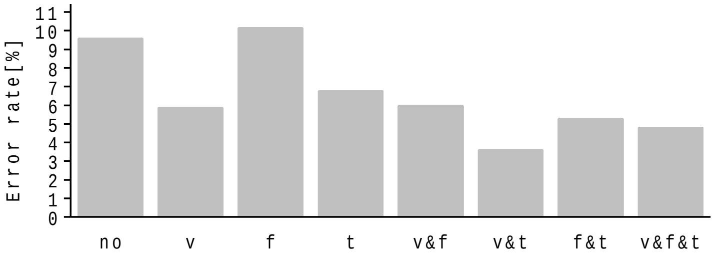

Figure 8.20: Error rates for the mouse in 3D, separated by feedback mode combinations (v=visual, f=force, t=tactile).

For the , the change in the error rate has been as small as the change in execution time. For the (fig. 8.21, left), the error rates are again worse than for the mouse, and the tactile feedback (14.3%) as well as a combination of tactile and visual feedback (17%) have not yeld the best results, which was achieved by visual feedback alone (8.3%).

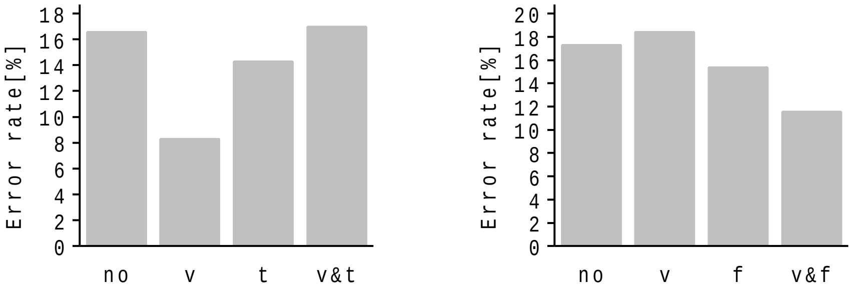

Figure 8.21: Error rates for the (left) and the joystick (right) in 3D, separated by feedback mode combinations (v=visual, f=force, t=tactile).

Very surprisingly, the visual feedback was no aid when the joystick was used, but the error rate has increased (18.4%) compared to no feedback at all (17.3%). Here, the force feedback was better (15.4%), and the combination of both, visual and force feedback, reduced the error rate to a minimum of 11.5% (see right picture of fig. 8.21).

8.6.4 Open questions and future work

Unfortunately, we have underestimated the difficulties in the evaluation process of our recorded data. The large amounts of data (> 10,000 records) and the influence of the various variables which have been investigated has increased the complexity of our task. In addition, the evaluation showed some effects which shoud be investigated in the future but which could not be included in WT 2.6 due to limited resources and time. The following list gives a brief summary of interesting aspects, and some of them will be explored within the next phases of MIAMI.

- The evaluation of task #3 is not finished yet.
- At the moment, the data has not been evaluated towards the question whether Fitts' law holds for devices with haptic feedback as well. This step will be done in the near future.
- The question of training effects has not been addressed yet. In our experiments, we have randomized the order of the tests (see section 8.5.2) in order to avoid these effects. However, longterm studies may be useful for inter individual comparisons of the input devices.
- A comparison between the subjective impression of the volunteers (we have used a questionnaire) and the objectively recorded data has not been done yet.
- More devices could be included in the experiments. Unfortunately, devices with haptic feedback are usually very expensive and therefore in most cases not affordable.
- Especially in 3D, a number of interesting experiments could be carried out, including metaphors for a mapping of 3D to 2D, the use of a head mounted display (HMD) or shutter glasses which provide a stereo view, or real interactions in 3D with a robot.
- The effect of haptic feedback under stress conditions, i. e. when the load on the operators visual channel is very high, has not been taken into account yet.
- The integration of haptic feedback in standard interfaces as well as the adaptation of feedback functions by observing and interpreting the user's actions is currently under investigation.
- A very interesting aspect, the integration of more than two modalities, will be investigated in the next phases of MIAMI. Especially the combination of the visual, the acoustical, and the haptic channel (machine → man) seems to be a very promising approach.

8.7 Conclusion

In the last years, there are more and more devices available on the market which provide their operator not only with input capabilities but with some kind of tactile or force feedback as well. Most of them are only used in special applications like VR, CAD, or robotics, and they are usually rather expensive. However, the question whether there is some potential benefit for the common user in everyday tasks when standard GUIs are extended with haptic feedback has not been investigated in depth yet.

Our experiments, which have been limited to five devices only (three with haptic feedback), have revealed that haptic feedback can replace and support visual feedback like color change or highlighting. The effects are more impressive in the 2D positioning task than in the 3D drag-and-drop task, but at least positioning is supported by haptic feedback in 3D, too. This is noteworthy because the role of 3D GUIs will certainly become more important in the future [40].

Another result is that most users liked the tactile feedback which is provided by the mouse best. It does not influence the handling of the mouse but supports the 'look and feel' of what the user sees. Interestingly, this is also the kind of feedback which is very easy and cheap to realize with most types of mice. The other devices which are usually used for special applications only could not affect the mouse's outstanding position.

Currently, we are investigating *how* haptic feedback can be used best to support the user. Therefore, we are implementing an intelligent agent which observes and analyzes the user's (inter-) actions and generates more specific feedback. The idea is to model the user's behavior and to anticipate the next actions in order to provide exactly the kind of feedback that helps best to fulfill a specific task.

Another question which will be addressed in the near future is how the haptic feedback will be integrated in standard UIs. Several ideas, including the mapping of the interface's relief to feedback functions, installing 'wavy' menus, or adding 'haptic semantics' to standard widgets like frames and buttons, will be realized and evaluated in our future work.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.7**

Task title: **Experiments:** **Cognitive-acoustical aspects** Partner responsible: RUB Start date: 01/07/94 End date: 30/11/94 Task manager: K. Hartung Planned resources: RUB: 3 - NICI: 1 (in man-months)	Sheet 1 of 1 Issue date: 24/04/95

Objective:

Evaluation of the influence of auditory stimulus parameters on recognition, recall, discrimination and decoding of information (number of features to be decoded simultaneously, use of ”real life” stimuli vs. designed stimuli (”ear”-cons))

Input:

WT 1.1, WT 1.2

Output:

Report with results

Approach:

- Definition of test parameters
- Development of test procedures
- Psychophysical tests with human subjects
- Evaluation of results

Contributions:

RUB experiment, software; NICI experiment

Chapter 9

Cognitive-acoustical aspects (WT-2.7)

9.1 Introduction

Common human-machine interfaces make extensive use of visual feedback. A few application make use of sound. For giving auditory feedback speech and non-speech sound output can be used. The use of non-speech sound can be divided in two main categories. One category are ”everyday sounds that convey information about events in the computer or in remote environment by analogy with everyday sound-producing events” [21]. They are called auditory icons. As an example for an auditory icon, think of the sound of a trashcan as an icon for the successful deletion of a computer file. The other group are ”abstract, synthetic tones which can be used in structured combinations to create sound messages to represent parts of an interface” [9]. These are called earcons. An example is a simple melody indicating an error status. Earcons are more easily parametrised than icons, yet, earcons have to be learned.

For the application-oriented construction of auditory icons it follows that an auditory icon is optimal in its form when it is optimal in its effect, i.e., when the information for which the icon is meant to stand is actually communicated. But not only the aspect of a correct understanding of the icon’s referent is a quality aspect; it is of equal importance that the process of its identification and understanding is as short as possible. Information would best be transferred when the icon is dealt with in the same way as the stimulus in a stimulus-response-chain. The listener should not be given the task to interpret the icon but simply to translate it.

Another important aspect of the construction of auditory icons is the one of the listener’s familiarity with the scenario. An auditory icon is understood on the basis of the listener's background knowledge of the whole domain. In order to reach an optimal effect, the icon must be an integral part of the domain, it looses in function when it is perceived as an alien element. There are some approaches to contruct and parametrize auditory icons. Most of them work in the domain of sound generation (e.g. physical parameters of the model) and do not consider the perceptual characteristics of the icon. From a psychoacoustically motivated point of view it is necessary first analyse main characteristic features of the referents that auditory icons should be signs for, and secondly to investigate how these features and their overall structures can be mirrored by acoustic images so that listeners have a common understanding without being trained or without a long term learning phase.

In contrast to auditory icons, earcons do not have their acoustic/auditory form for being similar to their referent, but because their form is based on explicit conventions. These conventions have to be known to the listeners - otherwise they are not capable of understanding the message. After all, earcons are real symbols. They are acoustical carriers of signs which can easily be memorised and/or applied. Their advantage lies in their simplicity with regard to their form. However, training is necessary to achieve that they are quickly and safely understood.

9.1.1 Goal of this investigation

It is beyond the scope of this study to investigate auditory icons and earcons together. Psychoacoustical studies about the parametrization of auditory icons need long-term psychoacoustical research. Also auditory earcons are specific to a certain domain, which restricts a transfer of the results to other applications. In this study we concentrated on earcons, because parametrization can be achieved with simples means and they are not specific to a task or domain.

We want to examine if earcons can be used as an aid in navigating in complex environments and to deliver information. Applications might be multimodal interfaces for complex databases or telepresence environments. In both cases the user has to be informed about his current position, the status of the machine and possible actions (e.g.: directions to move.) These information can be deivered by earcons and might replace or enhance visual and tactile information.

Brewster et. al [9]tested the effectiveness of earcons and developed guidelines for the design of earcons. Timbre, Rhythm and complex intra-earcon structures are very good to differenciate earcons. Intensity should not be used.

In order to test the effectiveness of earcons we designed an abstract application. This programm consists of a menu with three levels. The first level has four branches. Each branch is leading to the next level which has a node with four other branches. The third has sixty-four different items. If the user wants to reach one of these items he has to choose on each level, in which directions he wants to go. One method for coding the actual level might be colour or textual items. In this experiment we used earcons instead. Each level is identified by a certain attribute. The first level offers four different rhythm as symbols for each item. In the second level each type of item is distinguished by a different intervall or pitch contour. The last level uses different timbres to symbolize the different types of item. Each auditory attribute stands for a different level and a change in each attribute symbolizes a different position within a level.

9.1.2 Methods used for investigating

At the moment there are no detailed guidelines for the design of earcons. There is an indefinitly large amount of possible earcons, but only a few of them will serve well. The earcons should be easy to tell apart and good to memorize. For the application mentioned above, parametrization should be possible (modifying rhythm, changing intervall size).

The number of possible rhythms is reduced by several constrictions. The presention of a earcon should be very short (less than a second). If the duration is longer visual reaction will always be faster than auditory. Following the results of the study of Brewster et al. [9] the shortest tone should last at least 125 ms. Even with that restriction 16 rhythms remain. In order to select four rhythms wich are most easy to tell apart a similarity test for these 16 rhythms was designed.

The intervalls should not be greater than an octave. Otherwise grouping effects might occur, if fast rhythms are used.

9.2 Experiment 1: Dissimilarity test for different rhythms

A pair of two rhythms out of sixteen rhythms was presented. The subjects had to judge the perceived similarity or dissimilarity on a scale ranging from 0 (very differrent) to 9 (very similar). Two subjects participated on this experiment. This test should lead to a measure of perceptual distance of the different rhythms. It turned out that this method cannot be used for testing the rhythms. Some rhythms were perceived as very different and the judgements were very reliable. But for a lot of other rhythms, which have been judged a less different, the judgement was influenced by the order of presentation. Also the subjects found the task extremely difficult for these rhythms. Because of this effects we decided not to continue these experiments and choose four rhythms which have been undoubtly perceived as different rhythms.

9.3 Experiment 2: Learning and Recognition of Auditory Icons

9.3.1 Subjects

Sixteen persons served as subjects. Seven of the subjects were musically trained. They played an instrument and were able to read music. The age of the subjects was betweem 24 and 34 years.

9.3.2 Method

The experiment was conducted in 3 phases. A learning-phase or the prototype earcons, a recall phase and phase were recognition of the combination of the prototype earcons was tested.

In the first phase the subjects learned the prototype earcons and the associated meaning for each level. The four different prototypes of each level were played 12 times in a random order. After the presentation of each earcon, four alternatives for the answer were displayed. The subjects had to type the first letter of the textual representation of the item on a keyboard. In case of a correct answer they received a positive feedback. Beside the recording of the stimulus-answer pair the time for the reaction was recorded. This experiment should give some imformation about learning rate and the time necessary to learn the different prototype earcons and their meaning.

In the second phase, testing the recall of the earcons, was performed one day after the initial training. The test procedure was exactly the same as for the first phase. The results of this experiment can show how good the subjects are in memorizing the different prototype earcons.

In the third phase, following immediatly the second phase, eight complex earcons, combinations of the prototype earcons, were presented four times in a random order. After presentation of the stimulus four alternatives for each level werde displayed on the screen and the subjects had to select one of these for each level, by typing the number of the item on a keyboard. The received positive feedback for correct answers. With this test we wanted to find out, if the subjects are able to decode the information, which complex earcons contain.

After all experiments subjects were informally asked about their experience with the task and their strategies.

The meaning of the earcons

The earcons in this example were used to describe computer programs. Each computer program had three attributes. The first attibute describes the operating system or computer the software runs (SUN, SGI, UNIX, DOS). This is represented by different intervalls. The second attribute is the type of software (spreadsheet, wordprocessing, database, compiler) which is coded by different rhythms. The third attribute is the price (1000 DM, 500 DM, 200 DM, 100 DM) which is mapped to different timbres.

9.3.3 Sounds used

All sounds were played by a MIDI-sound modul (Roland Sound-Canvas SC55), which was controlled by a SGI-Workstation (Indy).

Rhythm

The four rhythms were chossen from the set described above. The length of a complete pattern was five eighth units. Each pattern included a different number of notes, ranging from two to five notes. In the following table 1 denotes an eighth-note, 0 denotes an eighth pause.

Number	Rhythm
1	10100
2	10110
3	11011
4	11111

During the first and second phase these rhythms were presented with a cowbell sound (MIDI-channel: 10, Prog-No: 01, Note-No: 56)

Pitch contour

Four different intervalls centered around C5 (MIDI-Code 75) have been selected. The intervall range was allways less than one octave. There were two contours making a down movement and two in the opposite direction. The absolute intervall for all contours was different .

Number	Start-Note	End-Note
1	72 (C 5)	78 (F#5)
2	72 (C 5)	76 (E 5)
3	72 (C 5)	70 (Bb4)
4	72 (C 5)	65 (F 4)

For the presentation in phase one and two the intervalls were presented with a flute-sound (MIDI-channel : 5, Progr-No : 74) and rhythm 1.

Timbre

The next tabel shows the timbres which were choosen. In some informal sessions these four instruments were very easy to tell apart.

Number	Timbre	MIDI-Channel	Prog-No.
1	Piano	6	01
2	Trumpet	7	57
3	Violin	8	41
4	Organ 1	9	17

In phase one and two each timbre was presented with rhyhtm 1.

All earcons are played with a tempo of 240 beats/minute, which leads to a maximal duration of 625 ms for each earcon.

Complex earcons

In the third phase 12 complex earcons which were composed by using the rhythms, intervalls and timbres of the primitive earcons were tested. The first line in the table shows one of theses earcons. Using rhythm 4, pitch 3 and timbre 1 leads to a group of 5 eighth notes (C5, Bb4, C5, Bb4, C5, Bb4) played with a piano sound.

Number	Rhythm	Pitch	Timbre
1	4	3	1
2	3	2	2
3	1	1	4
4	4	1	3
5	2	3	4
6	1	4	2
7	4	3	1
8	3	2	3
9	2	4	1
10	3	1	4
11	1	2	3
12	2	4	2

9.3.4 Results

Learning phase

In this phase the subjects learned the meaning of the different primitive earcons.

The score of correct recognitions could be defined as the ratio of correct answers and the number of presented stimuli. This definition does not consider that a subject can reach a high score just by giving the same answer, without knowing anything about the meaning of earcons. In our case the subject had four alternatives which would lead to score of 25% in this case. If you want to correct the score for this guessing effect the following formula has to be used.

$$s = \frac{100}{n}(r - \frac{n-r}{a-1}) \tag{9.1}$$

with s representing the corrected score, n the number of stimuli, r the number of correct answers and a the number of alternatives.

Just for giving a qualitative impression fig. 9.1 shows the score for the three different type of earcons as a function of time. In the beginning, the subjects have to find out which meaning is mapped to each earcon. Although the number of subjects is too small for reaching a statistical significant level, it seems that the timbre earcons are learned faster than rhythm and pitch.

The difficulties in learning and recalling the different earcons are not evenly distributed within a class of earcons. Table 9.1 shows a confusion matrix for the rhythms earcons.

Rhythm Earcon	Answer 1	 2	 3	 4
1	**71.88**	12.46	10.15	5.51
2	9.57	**63.19**	20.87	6.38
3	9.57	21.16	**60.29**	8.99
4	4.64	7.25	6.09	**82.03**

Table 9.1: Confusion matrix for rhythm earcons, phase 1

Pitch Earcon	Answer 1	 2	 3	 4
1	**88.41**	4.64	4.06	2.90
2	9.86	**73.33**	8.12	8.70
3	6.96	7.83	**71.01**	14.20
4	7.54	5.22	11.88	**75.36**

Table 9.2: Confusion matrix for pitch earcons, phase 1

In each line the relative distribution of answers is listed. When earcon 1 was presented the subjects recognized this earcon in 71.88% of the presentation. In 12.46% cases they confused it with earcon 2. When earcon 2 was presented it was confused in 9.57% of the presentations with earcon 1. It seems that rhythm 3 and 2 are very difficult to tell apart (appr. 20% confusions.)

For the pitch earcons a lot of confusions can be observed for the earcons 3 and 4 (table 9.2). These two earcons have have in common that the contour performs a down movement.

Among the timbre icons the piano sound (earcon 1) has the highest recognition rate (table 9.3).

Recall phase

Figure 9.2 shows the score of correct recognitions for the different type of earcons in the recall phase. (rhythm: 90.82%, pitch: 87.18%, timbre: 97,10%, entire poulation 91.72%). The differences in the mean scores for the different groups do not reach the 5%-significance level with any test.

The confusion matrix for the rhythm earcons in the recall phase (table 9.4) shows a

Timbre	Answer			
Earcon	1	2	3	4
1	**92.46**	2.90	2.61	2.03
2	3.77	**82.61**	11.01	2.61
3	4.06	6.96	**83.48**	5.51
4	3.48	4.06	6.67	**85.80**

Table 9.3: Confusion matrix for timbre earcons, phase 1

tendency,that the rhythm 2 and rhythm 3 are more difficult to distinguish than the rest. This confirms the observation of the learning phase.

For the pitch earcons (table 9.5) it is found that the intervall 1 is confused with intervall 2 and that the intervall 3 with 4. Intervall 1 and 2 perform an up movement while 3 and 4 move in the opposite direction.

In the recall phase no significant differences for the recognition of the different timbres can be observed (table 9.5).

Rhythm	Answer			
Earcon	1	2	3	4
1	**97.99**	0.33	1.00	0.67
2	2.01	**85.28**	11.71	1.00
3	2.01	9.03	**88.63**	0.33
4	1.00	0.33	0.67	**97.99**

Table 9.4: Confusion matrix for rhythm earcons, phase 2

Pitch	Answer			
Earcon	1	2	3	4
1	**93.98**	5.02	1.00	0.00
2	5.35	**90.97**	1.67	2.00
3	3.34	1.34	**88.29**	7.02
4	4.35	0.00	7.36	**88.30**

Table 9.5: Confusion matrix for pitch earcons, phase 2

Timbre	Answer			
Earcon	1	2	3	4
1	**98.66**	0.33	0.33	0.67
2	1.00	**95.65**	3.01	0.33
3	0.33	1.00	**98.33**	0.33
4	0.67	0.67	0.00	**98.66**

Table 9.6: Confusion matrix for timbre earcons, phase 2

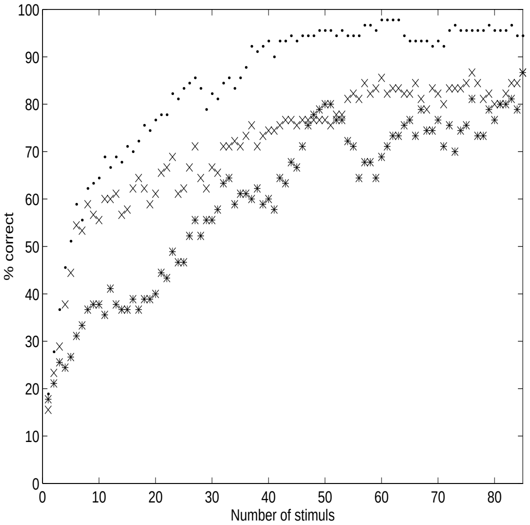

Figure 9.1: Percentage of correct recognitions for different earcons as a function of stimulus number. $*$: rhythm, $\times$: pitch, $\cdot$: timbre.

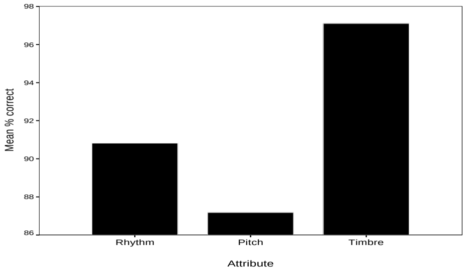

Figure 9.2: Breakdown of scores per attribute for phase 2

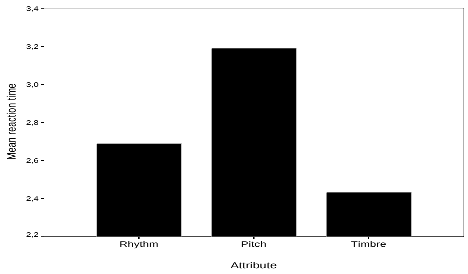

Figure 9.3: Mean reaction time per attribute

Figure 9.3 shows the mean reaction time for three different earcons. (rhythm: 2.69 s, pitch 3.19 s, timbre: 2.43 s). The difference between the reaction time for pitch and timbre is significant at a 5% level (Scheffé-Test). The other differences are statistically not significant.

Discussion of Musicians vs. Nonmusicians

As half of the tested subjects were musicians the results might be biased by the different experience and training of musicians and non-musicians. Musicians might have less difficutly in distinguishing and recognizing rhythms, intervalls or timbres than non-musician. To test this factor the scores and reactions times have been analyzed seperately for musicians and nonmusicians. Fig. 9.4 shows a breakdown of scores for the different attributes for non-musicians and musicians.

attribute	non-musician	musician	difference	
rhythm	88.89	92.76	3.87	N.S.
pitch	87.44	86.96	-0.48	N.S
timbre	95.16	98.75	3.59	N.S

A t-test shows that the differences between musicians and non-musicians are not significant.

The differences in reaction time between musicians and non-musicians shows fig. 9.5. The reaction time for the pitch-earcon is for both groups much longer than for rhythm and

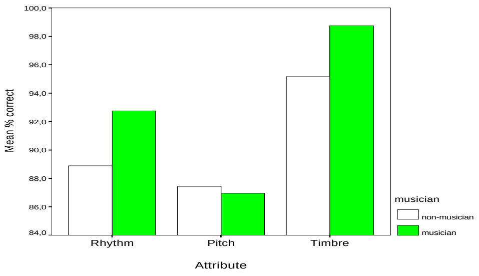

Figure 9.4: Scores of musicians and non-musicians

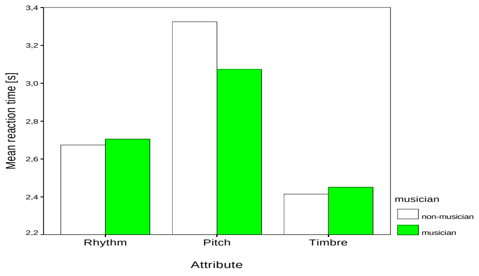

Figure 9.5: Reaction time of musicians and non-musicians

timbre.

The same observation can be made for the reaction time. Again the differences between the groups do not reach a significant level. The differences between different attributes are much bigger than the differences between the musicians and non-musicians.

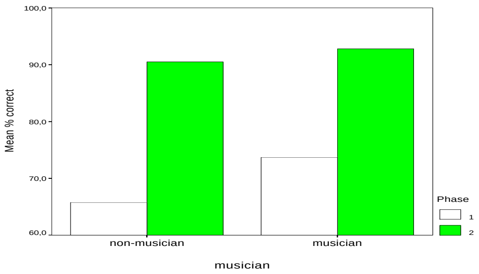

Figure 9.6: Score for musicians and non-musicians for phase 1 and 2

attribute	non-musician	musician	difference	
rhythm	2.67s	2.70s	0.03s	N.S.
pitch	3.32s	3.07s	-0.25s	N.S.
timbre	2.41s	2.45s	0.036s	N.S.

These results suggest, that after the training period no significant differences between musicians and non-musicians can be observed. But this does not mean that there has been no difference between the groups when they have perform the task the first time.

Fig. 9.6 and fig. 9.7 show the scores and reaction time during the training phase and in the recognition phase. The graphs suggest that musicians learn faster than the non-musicians. For a test of significance much more subjects are necessary.

Recognition of complex earcons

The table 9.7 shows the results for the recognition of complex earcons. Again the recognition of the pitch contour atrributes is worse than for rhythm or timbre.

A comparison of the score for the complex earcons and simple earcons (table 9.8) shows a decrease in recognition rate for pitch coded attributes embedded in complex earcons. Further test of complex earcons have to proof if this is significant.

The mean of the reaction time for complex earcons is 30.77 s per earcon (STDV = 15).

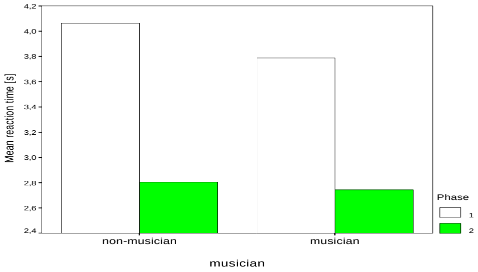

Figure 9.7: Reaction time of musicians and non-musicians

9.4 Discussion

The performance of the subjects in our experiments for the primitive earcons are much better than the report of Brewster et al [9]. It is not possible to compare the influence of all auditory attributes but it seems that in our experiments the decoding of the timbre information in the complex earcons is better (this study: 94%, Brewster < 70%). In that study the subjects had to learn complex earcons while they were played in a random order. The whole set was played three times. The authors do not give any details how the subject responded, but it can be assumed that the subjects responded orally. In our experiment which tested the primitive earcons the subjects had to type the first letter of the item the earcon stood for. One subject explicitly reported, that he did not learn the meaning of the icon, but to press the right button, when he heard the sound. This means that a motoric action and not the meaning was learned. This subject performed very bad in the experiment with the complex icons. It is possible that other subjects used the same strategy. This might explain the changes in the performance between the tests of primitive earcons and the complex earcons. The reaction time for complex earcons seems rather large and the subjects found this task difficult. But it has to be considered, that all subjects listened the first time to complex earcons and did not have any chance for training during the experiment. Long-term experiments are required to find out if training can reduce the time necessary to translate the coded information.

9.5 Conclusions

The results of this experiments show that after a relativly short training period simple earcons can easily be recognized and used as a means of communication. Scores for primitive earcons are between 87% and 97%. Reaction times are between 2.43 s and 3.19 s. The earcons used in this experiment were easily memorized after a day after learning. There are not enough data for a final conclusion but the first results of the complex earcon experiments show, that the subjects were able to decode the information which was conveyed by the complex earcons. The subjects never had listened to a complex earcon before and only new the primitive elements the earcons was built from. This data support the idea that a multidimensional mapping of information on a single earcon can be understood if the subjects knows the rules which were used.

The results show that after training there are no significant differences in recognition performance between musicians and nonmusicians.

9.6 Future work

For the future it is planned to continue the tests with primitive earcons and complex earcons. Questions which have to be adressed are how many different items which can be coded by rhythm, pitch contour and timbre. It has to be tested if different features of the earcons influence the performance. Better methods for designing sets of earcons have to be developed and psyhcoacustically verified. Other experiments will adress application of earcons in a navigation task. With respect to the scenarios in workpart 3 the influence of visual-feedback, auditory-feedback and the combination of auditory and visual feedback will be investigated.

TA Work Task Synopsis

ESPRIT BASIC RESEARCH Project 8579 MIAMI	WP No. 2	WT No. **2.8**

Task title: **Dissemination of results** Partner responsible: NICI Start date: 01/12/94 End date: 31/12/94 Task manager: L. Schomaker Planned resources: NICI: 1 - DIST: 1 - ICP: 2 (in man-months) RIIT: 1 - UKA: 1 - RUB: 1	Sheet 1 of 1 Issue date: 24/04/95

Objective:

Results from the experiments on bimodal interaction must be collected and distributed among all partners. The results are evaluated in order to be able to define the system architecture in WP3.

Input:

WT 2.1 – WT 2.7

Output:

Interim Report 1

Approach:

- Collection of reports from WT 2.1 – WT 2.7.
- Distribution among partners
- Evaluation of results pertaining to WP3.

Contributions:

NICI coordination; ALL reports

Number	level 1	level 2	level 3
	rhythm	pitch	timbre
1	80	80	90
2	70	80	90
3	90	70	90
4	80	50	100
5	90	70	90
6	90	100	90
7	90	80	100
8	80	60	100
9	100	60	100
10	90	50	90
11	90	70	100
12	80	70	90
total	85.83	70.00	94.17

Table 9.7: Scores for complex earcons

	rhythm	pitch	timbre
simple earcons	90.82	87.18	97,10
complex earcons	85.83	70.00	94.17
difference	-5	-17	-3

Table 9.8: Comparison of scores of simple and complex earcons

Chapter 10

Dissemination of results (WT-2.8)

Report 2.8 here

(Lambert Schomaker, NICI + DIST + ICP + RIIT + UKA + RUB)

Bibliography

[1] M. Akamatsu and S. Sato. A multi-modal mouse with tactile and force feedback. *Int. Journ. of Human-Computer Studies*, 40:443–453, 1994.

[2] R. Balakrishnan, C. Ware, and T. Smith. Virtual Hand Tool with Force Feedback. In C. Plaison, editor, *CHI'94 Conference Companion*, pages 83–84, Boston, MA, April 1994. ACM/SIGCHI, ACM Press.

[3] R. J. Beaton et al. An Evaluation of Input Devices for 3-D Computer Display Workstations. *SPIE*, 761:94–101, 1987.

[4] R. J. Beaton and N. Weiman. User Evaluation of Cursor-Positioning Devices for 3-D Display Workstations. *SPIE*, 902:53–58, 1988.

[5] C. Benoît, M. Lallouache, T. Mohamadi, and C. Abry. *A set of French visemes for visual speech synthesis.* Elsevier Science Publishers B.V., North-Holland, Amsterdam, 1992.

[6] T. Benoît, C. Mohamadi and S. Kandel. Audio-visual intelligibility of french speech in noise. In *Journal of Speech & Hearing Research*, volume 37, pages 1195–1203, 1994.

[7] M. Bierling. Displacement estimation by hierarchical blockmatching. In *3rd SPIE Symp. Visual Commun. Image Process.* Cambridge, USA, Nov. 1988.

[8] J. Bortz. *Lehrbuch der Statistik.* Springer-Verlag, 1977.

[9] S. A. Brewster, P. C. Wright, and A. D. N. Edwards. A detailed investigation into the effectiveness of earcons. In G. Kramer, editor, *Auditory Display*, pages 471–498, Reading, Massachusets, 1994. Santa Fe Institute, Addison Wesley.

[10] N. M. Brooke and Q. Summerfield. Analysis, synthesis, and perception of visible articulatory movements. In *J. Phonetics*, volume 11, pages 63–76, 1983.

[11] F. P. Brooks, Jr. et al. Project GROPE - Haptic Displays for Scientific Visualization. In F. Baskett, editor, *SIGGRAPH'90 Conf. Proc.*, volume 24, pages 177–185, Dallas, TX, Aug. 1990. ACM.

[12] W. Buxton. Human Skills in Interface Design. In L. MacDonald and J. Vince, editors, *Interacting with Virtual Environments*, chapter 1, pages 1–12. John Wiley & Sons Ltd., 1994.

[13] CCIR. Tolerances for transmission time differences between the vision and sounnd components of a television signal. *Recommendation 717*, 1990.

[14] M. M. Cohen and D. W. Massaro. Modelling coarticulation in synthetic visual speech. In *Proceedings of Computer Animation93*, Geneve, Suisse, 1993.

[15] M. M. Cohen and D. W. Massaro. Development and experimentation with synthetic visible speech. In *Behavioral Research Methods, Instrumentation, & Computers*, pages 260–265, 1994.

[16] H. D. Crane and D. Rtischev. Pen and voice unite: adding pen and voice input to today's user interfaces opens the door for more natural communication with your computer. *Byte*, 18:98–102, Oct. 1993.

[17] B. Dodd and R. Campbell. *The psychology of lip-reading.* Lawrence Erlbaum Assoc. Ltd., London, 1987.

[18] W. Felger. How interactive visualization can benefit from multidimensional input devices. *SPIE*, 1668:15–24, 1992.

[19] P. M. Fitts. The Information Capacity Of the Human Motor System in Controlling the Amplitude of Movement. *Journal of Experimental Psychology*, 47(6):381–391, June 1954.

[20] Y. Fukui and M. Shimojo. Edge Tracing of Virtual Shape Using Input Device with Force Feedback. *Systems and Computers in Japan*, 23(5):94–104, 1992.

[21] W. W. Gaver. Using an creating auditory icons. In G. Kramer, editor, *Auditory Display*, pages 417–446, Reading, Massachusets, 1994. Santa Fe Institute, Addison Wesley.

[22] G. Geiser. Mensch-Maschine Kommunikation. Oldenbourg, 1990.

[23] N. L. Hesselmann. Structural analysis of lip-contours for isolated spoken vowels using fourier descriptors. In *Speech Communication*, volume 2, pages 327–340, 1983.

[24] P. F. I. *A parametric model for human faces.* PhD thesis, University of Utah, Department of Computer Sciences, 1974.

[25] B. M. Jau. Anthropomorphic Exoskeleton Dual Arm-Hand Telerobot Controller. In *IROS'88 Conf. Proc.*, pages 715–718, Tokyo, October/November 1988. IEEE.

[26] X. Jia and M. S. Nixon. Analyzing front view face profiles for face recognition via the walsh transform. In *Pattern Recognition Lett.*, volume 15, pages 551–558, 1994.

[27] F. E. K. and A. A. Montgomery. Automatic optically based recognition of speech. In *Pattern Recognition Lett.*, volume 8, pages 159–164, 1988.

[28] W. Kerstner, G. Pigel, and M. Tscheligi. The FeelMouse: Making Computer Screens Feelable. In W. Zagler et al., editors, *Computers for Handicapped Persons. ICCHP'94 Conf. Proc.* Springer-Verlag, 1994.

[29] J. Kittler and J. Illingworth. Minimum error thresholding. In *Pattern recognition*, volume 19, pages 41–47, 1986.

[30] R. Kohler. A segmentation system based on thresholding. In *Computer Graphics Image Processing*, volume 15, pages 319–338, 1981.

[31] I. S. MacKenzie and W. Buxton. Extending Fitts' Law to Two-Dimensional Tasks. In P. Bauersfeld, J. Bennett, and G. Lynch, editors, *CHI'92 Conf. Proc.*, pages 219–226. ACM/SIGCHI, ACM Press, May 1992.

[32] I. S. MacKenzie, A. Sellen, and W. Buxton. A comparison of input devices in elemental pointing and dragging tasks. In S. P. Robertson, G. M. Olson, and J. S. Olson, editors, *CHI '91 Conf. Proc.*, pages 161–166. ACM/SIGCHI, ACM Press, 1991.

[33] T. H. Massie and J. K. Salisbury. The PHANToM Haptic Interface: a Device for Probing Virtual Objects. In *Proc. of the ASME Winter Annual Meeting, Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Chicago, 1994.

[34] A. Murata. An Experimental Evaluation of Mouse, Joystick, Joycard, Lightpen, Trackball and Touchscreen for Pointing – Basic Study on Human Interface Design. In H.-J. Bullinger, editor, *Human Aspects in Computing: Design and Use of Interactive Systems and Work with Terminals*, pages 123–127. Elsevier Publishers, B. V., 1991.

[35] H. G. Musmann, P. Pirsch, and H. J. Grallert. Advances in picture coding. In *Proc.IEEE*, 73, pages 631–670, Apr. 1985.

[36] E. D. Mynatt. Auditory Representations of Graphical User Interfaces. In G. Kramer, editor, *Auditory Display*, pages 533–553, Reading, MA, 1994. Santa Fe Institute, Addison Wesley.

[37] E. D. Mynatt and W. K. Edwards. Mapping GUIs to Auditory Interfaces. In *UIST'92 Conf. Proc.*, pages 61–70, Monterey, CA, Nov. 1992. ACM.

[38] E. D. Petajan, B. Bischoff, and D. Bodoff. An improved automatic lipreading system to enhance speech recognition. In *ACM SIGCHI*, 88, pages 19–25, 1988.

[39] K. C. Pohlmann. Advanced digital audio. In *SAMS*, pages 273–286, 1991.

[40] G. G. Robertson, S. K. Card, and J. D. Mackinlay. Information Visualization using 3D Interactive Animation. *Communications of the ACM*, 36(4):57–71, Apr. 1993.

[41] L. Schomaker et al. A Taxonomy of Multimodal Interaction in the Human Information Processing System. Internal report, 8579 MIAMI, Feb. 1995.

[42] L. R. B. Schomaker. Using Stroke- or Character-based Self-organizing Maps in the Recognition of On-line, Connected Cursive Script. *Pattern Recognition*, 26(3):443–450, 1993.

[43] K. B. Shimoga. A Survey of Perceptual Feedback Issues in Dexterous Telemanipulation: Part II. Finger Touch Feedback. In *VRAIS'93 Conf. Proc.*, Seattle, Sept. 1993. IEEE, IEEE Service Center.

[44] S. Sneff. A joint synchrony/mean-rate model of auditory speech processing. In *Proc. J.of Phonetics*, volume 16, pages 55–76, Jan. 1988.

[45] G. Yang and T. S. Huang. Human face detection in a complex background. In *Pattern Recognition*, volume 27, pages 53–63, 1994.