Dialog-driven Video-realistic Image-based Eye Animation

Von der Fakultät für Elektrotechnik und Informatik

der Gottfried Wilhelm Leibniz Universität Hannover

zur Erlangung des akademischen Grades

Doktor-Ingenieur

genehmigte

Dissertation

von

Dipl.-Ing. Axel Weißenfeld

geb. am 3. August 1976 in Langenhagen

2010

1.	Referent:	Prof. Dr.-Ing. J. Ostermann
2.	Referent:	Prof. Dr. rer. nat. V. Blanz
	Tag der Promotion:	29.10.2010

Acknowledgements

Many people have contributed to the making of this book. First of all I would like to thank my wife for her incredible patience, encouragement and great suggestions. Thanks to her I managed to finalize this book. Lots of thanks also to my family and friends for their constant support.

I would like to express my gratitude to Professor Dr.-Ing. J. Ostermann for being my supervisor, for his invaluable support and the given opportunity to be a part of his research team. I would like to thank Professor Dr. rer. nat. V. Blanz for willingly agreeing to serve on my committee. A great motivation for my research was to work with Kang Liu. Not only did I have some fruitful discussions about facial animations, he also introduced me to the Chinese culture.

As a scholarship holder of the "Stiftung der Deutschen Wirtschaft", I would like to thank the endowment for the financial support of my scientific work. The opportunity of being part of the endowment was a great experience.

Last but not least I feel very obliged to all my colleagues and students at the Institut für Informationsverarbeitung of the Leibniz Universität Hannover for making such a complex project possible by spending so much time and effort. Thanks a lot to you all.

Kurzfassung

Die heutige Mensch-Maschine-Kommunikation besteht überwiegend aus Texteingabe und Mausclicks einerseits und Text-, Bild- und Grafikausgabe andererseits. In Zukunft kann die Maschine die Interaktion auch mit einer synthetischen Sprachausgabe in Verbindung mit einer fotorealistischen Gesichtsausgabe bereichern. Unter einer fotorealistischen Animation wird in dieser Arbeit folgendes verstanden: Die Animation darf nicht von einer Videoaufnahme zu unterscheiden sein und sie muss ein als natürlich empfundenes menschliches Verhalten aufzeigen. Dafür müssen glatte Mundbewegungen sowie passende nicht-verbale Artikulationen, die u.a. aus Mimik, Kopf- und Augenbewegungen bestehen, erzeugt werden. Diese Arbeit konzentriert sich auf den letzten Punkt, und es wird ein neues image-based Animationssystem vorgestellt, welches zu beliebigen Sprachausgaben die passenden Augenbewegungen erzeugt. Das Augenanimationssystem setzt sich wiederum aus zwei Teilen zusammen: einem Augensteuerungsmodell und einem Rendering-Engine, welches Animationen durch die Kombination eines 3D Augapfelmodells mit einem passenden image-based Augenmodell erzeugt.

Die Steuerung der Augen basiert auf der Physiologie des menschlichen Auges sowie einer statistischen Analyse der Augenbewegungen von menschlichen Probanden. Zu diesem Zweck werden zwei Experimente definiert, mit denen die Blickbewegungen ebenso wie das Augenblinzeln der beiden Probanden während eines Gespräches analysiert werden können. Wie bereits in früheren Publikationen erwähnt, unterscheiden sich die Augenbewegungen beim Menschen während des Zuhörens und des Sprechens. Laut unserer Analyse können zwei unabhängige endliche Automaten mit je zwei Zuständen die Augenbewegungen und das Augenblinzeln beim Zuhören erzeugen. Im Gegensatz dazu muss während des Sprechens ein integriertes Modell die Blickrichtung und das Augenblinzeln steuern, da beide Ereignisse gekoppelt sind. Schwerpunkt dieser Arbeit ist der Entwurf dieses integrierten Augensteuerungsmodells, das automatisch passende Augenbewegungen und Augenblinzeln durch phonetische sowie prosodische Informationen zu beliebigen Sprachausgaben generiert. Eine Analyse der Blickrichtung der Probanden zeigt, dass der Blick, falls er auf den Gesprächspartner gerichtet ist, nicht starr, sondern zu verschiedenen Positionen im Gesicht wechselt. Da diese Blickrichtungsänderungen andere Eigenschaften aufweisen, ist das Modell zur Steuerung der Blickrichtungen durch einen zusätzlichen endlichen Automaten verfeinert, der diese Eigenschaften modelliert. Außerdem ist das Augenbewegungsmodell, welches den Vestibulo-Okular-Reflex und Sakkaden erzeugen kann, verbessert, indem das Listing'sche Gesetz die Neigung des Kopfes sowie die Kopplung zwischen vertikalen Sakkaden und Augenblinzeln berücksichtigt. Darüber hinaus wurde ein neuartiger endlicher Automat eingeführt, der die gesprochene Sprache und gleichzeitig den zeitlichen Verlauf zur Steuerung des Augenblinzelns berücksichtigt.

Die Bewertung des neuen Augensteuerungsmodells erfolgt durch einen subjektiven Test, bei dem die Teilnehmer zwischen realen und animierten Videos, die entweder durch das Referenzverfahren oder mit Hilfe des neuen Augensteuerungsmodells erzeugt wer-

den, unterscheiden müssen. Die Auswertung des Tests ergab, dass die Teilnehmer 78% des Referenzverfahrens, aber lediglich 54% des neuen Augensteuerungsmodells richtig identifizieren konnten. Die Nullhypothese, dass die Teilnehmer nicht in der Lage sind, zwischen einer Videoaufnahme und einer Animation zu unterscheiden, wird mittels einer Binomialverteilung getestet. Dabei wird die Nullhypothese für das Referenzverfahren abgelehnt, aber für das neue Modell bestätigt. Da sowohl die Anzahl der richtig erkannten Videos bei ca. 50% liegt als auch die Nullhypothese bestätigt wird und die Teilnehmer des subjektiven Tests im anschließenden Interview keine Mängel oder Fehler bei Videos, die mit dem neuen Augensteuerungsmodell erzeugt wurden, angaben, ist das Resümee dieser Arbeit, dass das neue Animationssystem fotorealistische Augenanimationen erzeugt, was bisher noch nicht erreicht wurde.

Des Weiteren wird in dieser Arbeit ein modell- und gradientenbasierter Algorithmus zur Schätzung der Kopfposition, der die Anforderungen eines image-based Animationssystems erfüllt, vorgestellt. Bei diesem Algorithmus wurde die Genauigkeit durch die Einführung einer neuen Gewichtung der Merkmalspunkte sowie eines neuen Ansatzes für die Aktualisierung der Texturinformationen verbessert.

Schlagwörter: image-based, Gesichtsanimation, Augenanimation, Bewegungsschätzung des Kopfes, Bildverarbeitung, Computergraphik

Abstract

Talking-heads are useful to give a face to multimedia applications such as virtual operators or news readers in dialog systems. However, their great commercial potentials can only become true, if talking-heads are indistinguishable from real recorded videos and at the same time correctly model the human-like behavior. For this, mouth as well as non-verbal behaviors such as head movements, facial expressions and eye movements need to be generated. In this project, we focus on the latter and a novel image-based system for creating video-realistic eye animations for talking-heads to arbitrary spoken output is elaborated. Our eye animation system consists of two parts: eye control unit and rendering engine, which synthesizes eye animations by combining 3D and image-based models.

The designed eye control unit is based on eye movement physiology as well as the statistical analysis of recorded human beings. For this, we designed two experiments, in which we analyzed gaze as well as blink patterns of two human beings. As shown in previous publications, eye movements vary while listening and talking. In listening mode, two finite state machines, each with two states, generate the gaze and blink patterns, since these patterns are not coupled as determined our analysis. We focus on talking mode and are the first researchers to design a new model, which fully automatically couples eye blinks and movements with phonetic as well as prosodic information extracted from spoken language. At the same time, we design one integrated model, which considers the coupling between gaze shifts and eye blinks as determined by our experiment. Our analysis reveals that the eye gaze moves across the face while looking at the interlocutor. These gaze shifts have other characteristics than the shifts performed to switch from mutual gaze to gaze away. Therefore, we extend the presently known simple gaze model by refining mutual gaze. Furthermore, we improve the eye movement models, which generate the vestibulo-ocular reflex and saccades, by considering head tilts, torsion and eyelid movements. In addition, a novel finite state machine is introduced, which considers the spoken output and the temporal course to generate eye blinks.

The eye animation system is evaluated by a subjective test in which participants discriminate between real and animated videos, which are either created by a reference method or our designed eye control unit. The analysis of the test reveals that participants correctly identified the real video with 78% and 54% of the reference and our proposed method, respectively. Testing the null hypothesis with the binomial distribution indicates that the hypothesis that participants are not able to distinguish between real and animated sequences is rejected with respect to the reference method, but retained with our proposed method. We conclude the new eye animation system creates video-realistic eye animations for a talking-head, which has not been achieved before, since the correctly identified video samples are close to chance level, the null hypothesis is retained and participants did not criticize our video samples in an interview following the subjective test.

As a minor issue an appropriate model-based and gradient-based head pose estimation algorithm as required by an image-based animation system is presented. Here the accuracy of the algorithm is improved by two approaches: Firstly, feature points, which are

tracked in the image sequence, obtain new weights in order to better compensate non-rigid motion. Secondly, a novel approach of updating texture information is introduced, which allows to estimate larger out-of-plane rotations of the head.

Keywords: image-based, facial animation, eye animation, head pose estimation, computer vision, computer graphics

Contents

Symbols

Notation:

$(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$	world coordinate system
$(\mathbf{X}^c, \mathbf{Y}^c, \mathbf{Z}^c)$	camera coordinate system
$(\mathbf{X}^o, \mathbf{Y}^o, \mathbf{Z}^o)$	object coordinate system
$\bar{C}_H$	normalized cost function of C_H
$\bar{\mathbf{I}}(\mathbf{p}, t)$	mean free intensity value of luminance signal of image t at the 2D position $\mathbf{p}$
$\bar{\mathbf{J}}_i(\mathbf{p})$	mean free intensity value of luminance signal of template i at the 2D position $\mathbf{p}$
$\bar{I}_t$	average intensity value of luminance signal of image t
$\bar{J}_i$	average intensity value of luminance signal of template i
$\bar{X}$	sample mean
$\beta_x, \beta_y, \beta_z$	euler angels of eye globe
$\check{f}_e(x)$	normalized and shifted exponential distribution
$\check{f}_{ln}$	shifted and normalized lognormal distribution
$\check{\mathbf{I}}(\mathbf{p}, t)$	true intensity value of luminance signal of image t at the 2D position $\mathbf{p}$
$\Delta\mathbf{R}$	linearized rotation matrix
$\Delta I^{\mathbf{p}}$	neighboring luminance difference of$\mathbf{p}$
δ_e	systemetic error due to offset between projected circle center and barycenter of the projected cirlce
$\hat{f}_e(x)$	shifted exponential distribution
$\hat{f}_{ln}$	shifted lognormal distribution
$\hat{t}_g^{y_1}$	time remaining in state y_1
$\hat{\mathbf{H}}_j$	head pose of reference frame j projected on the unit sphere
$\hat{\mathbf{I}}(\mathbf{p}, j)$	intensity value of luminance signal of reference frame j at the 2D position $\mathbf{p}$
$\hat{\mathbf{r}}$	average angular velocity of $\hat{s}$
$\hat{s}$	sub-trajectory of head motion
ι	word
κ_1, κ_2	radial lens distortion parameters
λ	motion parameters
λ_e	parameter of exponential distribution
λ_r	regularization parameter
MV^s	mean velocity of a saccade

μ_{ln}, σ_{ln}	parameters of lognormal distribution
Ω	set of 2D feature points
Ω^*	$\Omega^* \subset \Omega$
$\omega_x, \omega_y, \omega_z$	Euler angles
ϕ	angle between the line of sight to the corresponding 3D feature point
ϕ_w	positive function
$\psi_M(u)$	weight function
ρ^J	correlation coefficient between image $\mathbf{I}(t)$ and template $\mathbf{J}_i$
$\rho_M(u)$	robust cost function
σ_I	standard deviation of the luminance difference of the current frame t and reference frame j
σ_I	standard deviation of the luminance difference of the current frame t and reference frame j
σ_M	scale factor of $\rho_M(u)$
σ_{F0}	standard deviation of F0
τ_v	duration of a frame
θ_h	rotation angle of q_h
θ_q	rotation angle of q
υ	word duration
ς	classification of word as slow, medium or fast
$\mathbf{c}$	principal point
$\mathbf{E}$	3D point on computer screen (POR)
$\mathbf{e}$	2D point on computer screen (POR)
$\mathbf{F}$	parametric motion model
$\mathbf{G}$	3D barycenter of object
$\mathbf{I}(\mathbf{p},t)$	intensity value of luminance signal of image t at the 2D position $\mathbf{p}$
$\mathbf{J}_i(\mathbf{p})$	intensity value of luminance signal of template i at the 2D position $\mathbf{p}$
$\mathbf{M}$	rotation matrix (quaternion)
$\mathbf{n}$	pixel image coordinate
$\mathbf{P}$	3D feature point in world coordinates
$\mathbf{p}$	2D image point
$\mathbf{p}_d$	distorted 2D image point
$\mathbf{q}$	vector part of quaternion
$\mathbf{R}$	rotation matrix (Euler angels)
$\mathbf{r}$	angular velocity
$\mathbf{T}$	translation vector
$\mathbf{v}_q$	rotation axis of q
$\mathbf{F}_\lambda$	derivative of parametric motion model
$\mathbf{j} = [j_x, j_y]^T$	2D shift aligning frame $\mathbf{I}$ and template $\mathbf{J}$
$\mathbf{P}^c$	3D feature point in camera coordinates
$\mathbf{P}^o$	3D feature point in object coordinates

$\mathbf{H}_t$	head pose of image t projected on the unit sphere
$\mathbf{I}_\mathbf{p} = [\mathbf{I}_x, \mathbf{I}_y]^T$	spatial image gradient of luminance signal
ξ_c	threshold of neighboring luminance difference
ξ_d	threshold of the maximum displacement between two consecutive frames
ξ_R	threshold to classify words
ξ_s	threshold to segment head motion
ξ_{F0}	threshold containing top 10% of F0
ξ_v	threshold to classify syllable as emphasized
$\zeta^\mathbf{p}$	weight of feature point $\mathbf{p}$
$\zeta^\mathbf{p}_C$	sub-weight of $\zeta^\mathbf{p}$
$\zeta^\mathbf{p}_D$	sub-weight of $\zeta^\mathbf{p}$
$\zeta^\mathbf{p}_E$	sub-weight of $\zeta^\mathbf{p}$
$\zeta^\mathbf{p}_G$	sub-weight of $\zeta^\mathbf{p}$
$\zeta^\mathbf{p}_I$	sub-weight of $\zeta^\mathbf{p}$
$\zeta^\mathbf{p}_V$	sub-weight of $\zeta^\mathbf{p}$
A^s	magnitude of a saccade
$b(N_b, p_b, j)$	binomial distribution function with the binomial coefficient $\binom{N_b}{j}$ and probability p_b of success on each experiment
C	cost function of the residual error r
C'	cost function of the residual error r'
$c_1, c_2, ...$	scalars
c_a	cepstrum
C_H	cost function of current frame t and reference image $\hat{\mathbf{I}}(j)$
C_s	cost function of $\hat{s}$
D^s	duration of a saccade
d^s	slope relating D^s with A^s
D^s_0	constant relating D^s with A^s
D_1, D_2	interval endpoints
f	focal length
f_a	audio frequency
f_e	exponential distribution
f_{ln}	lognormal distribution
h_H	Hann window
i_x, i_y	image sampling values
l_i, l_f	initial and final frame of $\hat{s}$
L_q	quaternion rotation operator
n_a	audio sample
n_G	Gaussian noise of intensity value of CCD sensor
N_x, N_y	image width and height in pixels
n_{F0}	number of frames above ξ_{F0}
o	observation

$p(y_1)$	experimental probability
$p(y_1, y_2)$	joint probability
$p_{ROI}^{1j}, p_{ROI}^{2j}, p_{ROI}^{3j}$	interval endpoints of state j modeling the transition probability from one ROI to the next
p_ι	probability distribution of the word duration
p_{y_1,y_2}	transition probability
q	quaternion
q_0, q_1, q_2, q_3	scalars of quaternion
q_h	unit quaternion describing head rotation
r	residual error of luminance signal of $\mathbf{p} \in \Omega$ between two frames using the approximation of Equation (3.23)
$r'(\mathbf{p}; \lambda)$	residual error of luminance signal of $\mathbf{p} \in \Omega$ between two frames
r_d	distance between distorted image point and principal point
r_e	marker radius
r_l	distance between two consecutive reference frames on the unit sphere
$R_\iota(i)$	probability that the duration υ of the word ι is located within a time interval
S	sample standard deviation
s_e	scaling factor relating computer screen pixel with the metric mm
s_x, s_y	scaling factor relating image points with pixel image coordinates
sl	syllable
t	index which labels frame sampled at the point of time t
t_{ROI}^j	number of frames remaining in ROI in state j
$t_b^{y_1}$	number of frames remaining in state y_1
$t_g^{y_1}$	number of frames remaining in state y_1
t_r	discrete points of time
t_{ga}	number of frames remaining in GA in talking mode
t_{sl}	duration of sl
v_d	spatial displacement between two consecutive frames
v_{F0}	variation and activity of F0
w_B	Blackman window
w_J, h_J	width and height of template $\mathbf{J}$
$w_{\mathbf{p},\mathbf{p}^*}$	weight of luminance difference of $\mathbf{p}$ and $\mathbf{p}^*$
x	continuous value
X_a	audio signal in frequency domain
x_a	windowed audio signal
x_a'	original audio signal
x_k	discrete time value
x_e	shift of exponential distribution
x_{ln}	shift of lognormal distribution

z_e	distance between the principal point and center of the marker

Abbreviations and Acronyms:

2D	two dimensional
3D	three dimensional
B	executing an eye blink
C	observation: consonant
CCD	charged coupled device
E	observation: end of sentence
F0	fundamental frequency
F0'	course estimation of fundamental frequency
FSM	finite state machine
FW	observation: filling word
GA	gaze away
GS	observation: gaze shift
IRLS	iteratively re-weighted least squares
LED	light emitting diode
MG	mutual gaze
MLE	maximum likelihood estimation
NB	not executing an eye blink
OT	observation: other
POR	point of regard
ROI	region of interest
SSR	observation: slow speech rate
V	observation: vowel
VOR	vestibulo-ocular reflex
WB	observation: word boundary
WP	observation: word prominence
ECU	eye control unit
TTS	text-to-speech

1 Introduction

Computer aided modeling of human faces usually requires a lot of manual control to achieve realistic animations and to prevent unrealistic or non-human like results. Humans are very sensitive to any abnormal lineaments, so that facial animation remains a challenging task till today. Facial animation combined with spoken output, also known as talking-head, can be used as a modern human-machine interface [103].

Talking-heads give a face to spoken output, which is either generated by a text-to-speech synthesizer (TTS) or recorded from a human subject. Dialog systems, as used in e-commerce and e-care, can integrate facial animations with synthesized speech generated by a TTS synthesizer in web sites to improve human-machine communication (Figure 1.1). For instance, a virtual, personal adviser can navigate and assist a customer through web sites. Subjective tests indicate that e-commerce web sites with an integrated facial animation and spoken output achieve higher customer satisfaction [105] [102]. Furthermore, talking-heads can be given as add-ons to audio productions. The talking-head can read audio books, so customers have the choice between audio only and a talking-head reading the story. The recording of human subjects in a professional studio is time consuming, e.g. the news speaker has to be painted their faces, and tedious, e.g. studio set-up has to be appropriately configurated. Instead of producing expensive TV and video productions, a talking-head can be animated by the spoken output of the human subject. In comparison to video a voice recording is very easy to accomplish and inexpensive.

Face animation research started in the early 70's by Parke [106]. In those days facial animation was limited by the available hardware, so that only primitive models describing the rough shape of a 3D face were animated. The first models were created by painting a set of polygons on a human face and taking 2D images from different views. A 3D-polygonal face is designed by reconstructing 3D points from corresponding 2D feature points. Hence, this algorithm requires manual assistance to generate 3D models and a patient human subject. Animation takes place by interpolating between two existing static expressions each defined by a simple 3D polygonal face. The animation described is very limited, since only intermediate expressions between known static expressions can be generated. In the last decade the quality of facial animation has significantly improved due to better computer systems, which have more power and increasing software capabilities. The animation techniques range from animating 3D models to image-based rendering of models [103]. Image-based animation processes only 2D images, so that animations are synthesized by combining different recorded 2D images. The techniques are so diverse because of potentially very different commercial applications. We use an image-based system since we believe that in the near future this type of system will allow to animate

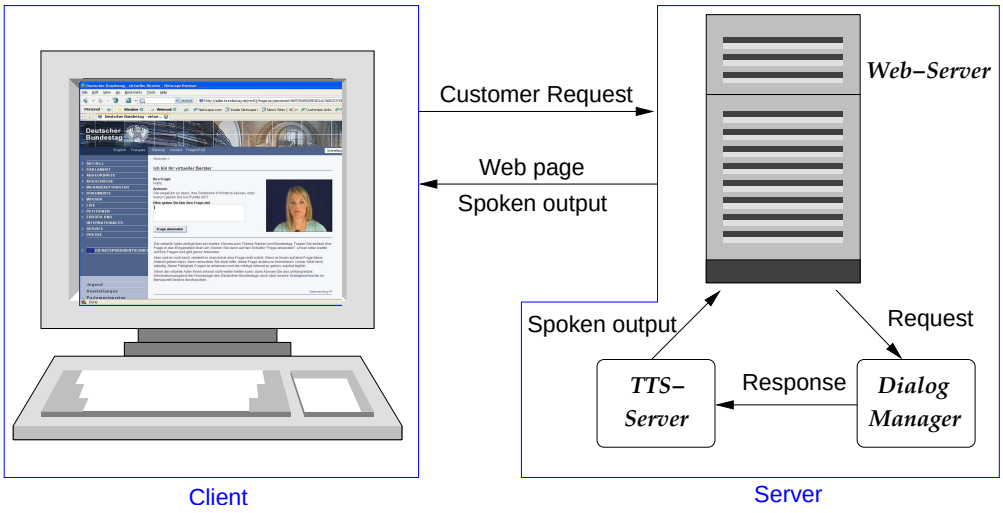

Figure 1.1: An e-commerce store integrates a personal adviser into its web site.

avatars, which cannot be distinguished from a real person.

Image-based facial animation, which was introduced by [18, 47, 31], concentrates on synthesizing smooth mouth animations by replacing the mouth area in a background sequence with previously stored samples in a database (Figure 1.2). The samples in the database, which may contain over ten thousand images, are normalized. Normalizing means to compensate for head pose variations. Background sequences are recorded video sequences of a human subject with typical short head movements. However, the proposed systems have several shortcomings, which are important for achieving video-realistic facial animations. We define video-realism as the synthesis of facial animations, which are indistinguishable from real recorded videos and at the same time correctly model the human-like behavior. According to Mehrabian [97] the impact of a message is about 7% verbal (words only), 38% vocal (including tone of voice, inflection, and other sounds) and 55% nonverbal. Hence, facial animations need to appropriately model non-verbal communication to spoken output to appear human-like. However, facial expressions, head and eye movements are mainly neglected in image-based systems. This work focuses on replacing the eye area to generate video-realistic eye animations to spoken output, since eyes play an essential role as a major channel of non-verbal communication. Note, that our proposed eye control unit (ECU) may also be used in 3D animation systems.

In order to overlay facial parts in a background sequence, the accurate head pose needs to be estimated in order to appropriately warp the normalized mouth sample to the estimated pose in the background sequence. Otherwise if the head pose is either not accurately measured in the background sequence or the mouth samples are not correctly normalized, the synthesized facial animations do look jerky [140]. The methods used in [18, 47, 31] do not satisfactorily solve this problem. While Bregler et al. [18] deter-

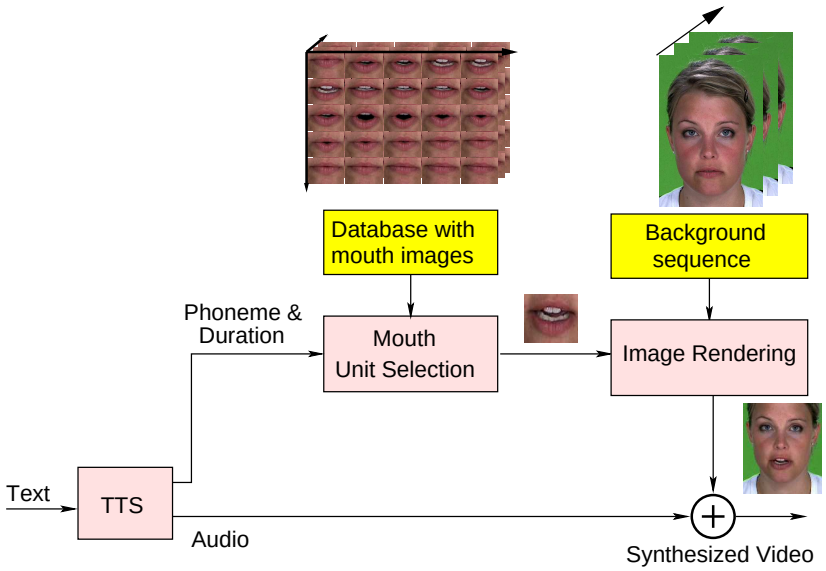

Figure 1.2: Image-based facial animation system: The mouth of the background sequence is replaced by a sample from the database.

mine 54 fiduciary points, Ezzat et al. [47] track feature points located in a plane head mask. Afterwards both perform an affine transformation to compensate head pose variations. However, this method is not able to correctly compensate out-of-plane rotations. Cosatto et al. [31] determine the four eye corners and two nostrils to estimate the head pose. Therefore, the nostrils must be always visible during the recordings and strong head pitches may not be executed. In order to overcome these problems a second focus of this work is to design a motion estimation algorithm, which overcomes the previously described restrictions while estimating the 3D head pose with great accuracy.

Head Pose Estimation

A head tracking system estimates the rigid motion of the human face throughout an image sequence (Figure 1.3). Head tracking systems are important for many applications in computer vision like expression analysis, face identification, model-based coding and 3D facial animation systems. Head motion can be used to recognize simple gestures, like head shaking or nodding, or for capturing a person's focus of attention, providing a natural clue for human machine interfaces.

Many variations of motion estimation algorithms for different applications have been proposed in literature. Differences can be noticed in the boundary conditions, like the use of calibrated or uncalibrated image sequences or monocular and stereo vision. In this work, we only analyze techniques, which process monocular image sequences. In

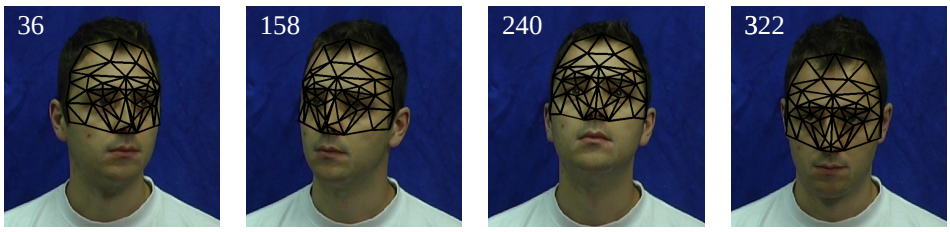

Figure 1.3: Head pose estimation in an image sequence. The head shape is modeled by a simple face model.

order to evaluate the existing methods in literature, we define the specific requirements of our application. In our set-up, a single news speaker is recorded and we can fully control the illumination of the scene. The head is always visible without any occlusions. Furthermore, our application requires to estimate the head pose with high accuracy.

In general, existing approaches can be divided into motion-based and model-based techniques.

In the first approach, the techniques track 2D feature points and use the face model only to transform 2D motion vectors into face model motion vectors. Various techniques [92, 118, 31] track a small number of distinct facial features, such as eye corners or nostrils throughout the image sequence. The displacements between corresponding feature points can be estimated using optical flow or block-based motion estimation algorithms or more sophisticated algorithms taking the specific attributes of the human face into account to automatically detect these features. All these methods require that these features are always visible and only work well when these features can be reliably and with great accuracy tracked over the image sequence. In [11] a large number of feature points with a large gradient are tracked and the head pose is calculated by minimizing the difference between the model and estimated 2D flow field. However, it is difficult to reliably track 2D facial features, especially if the head performs large out-of-plane rotations and local movements due to varying facial expressions. Malciu and Preteux [94] extended [11] by additionally calculating the difference between the image texture and back projected face model texture. Both errors are weighted and minimized by the downhill simplex method. A disadvantage of this approach is that algorithms are sensitive to the selected weights and therefore may not reach the global minimum. The work of Vacchetti et al. [135] focuses on estimating large camera displacements, extreme aspect changes and partial occlusions. Initially, a small set of reference frames consisting of the face texture and corresponding head pose are manually provided. The 3D head pose is estimated by tracking 2D feature points and taking 2D and 3D correspondences into account. The initially estimated pose is improved by bundle adjustment, which takes neighboring reference frames into account, too. This approach has the above mentioned disadvantage that it is difficult to

reliably track facial feature points in 2D.

A model-based tracker stores texture information of the object and tries to adapt the position of the face model to fit to the new frame without the use of 2D motion vectors. Model-based motion estimation can be accomplished by image registration in texture space, statistical trackers or optical flow. Note, that also combinations of these techniques exist.

In literature, various planar model-based methods have been proposed [58], which model the face as a plane and use a single face texture to estimate the pose. These approaches have the advantage of a simple initialization, but obviously lack in accuracy. The early algorithms were designed to estimate 2D motions in images and not the 3D motion of a head.

Cascia et al. [83] developed a method to estimate the head pose by image registration in texture space. A cylinder with a single texture-map of the face models the head. A new head is estimated by computing coefficients of linear combinations of a set of bases templates, which are generated by changing the pose of the texture-map. This approach cannot deal with varying facial expressions.

Ahlberg's algorithm for face tracking [75] is based on active appearance models [42]. The appearance model is a joint statistical model of the shape and the face texture, which is generated by calculating eigenfaces. The active appearance model uses a directed search algorithm for adapting the face model to the image. For better results, a simple block motion estimator determines the global motion of the face before the proposed algorithm is applied. The accuracy of his algorithm is evaluated by estimating the head pose in two synthetic sequences. The average x- and y-translation error is between 1.6 pel and 2.9 pel and too large for our application [75].

In general, statistical trackers try to determine the global minimum of some kind of cost function. While in [142] an evolution strategy optimizer is used to minimize the luminance difference between corresponding 2D image and 3D object features, Davoine et al. [40] use a partical filter to find the 3D head pose using the active appearance model as proposed by Ahlberg [75]. Both approaches have the disadvantage that the degrees of freedom is very high and therefore a large number of combinations exist, so that the global minimum may not be reached. Marks et al. [95] present a generative model and its associated filtering algorithm, which belongs to the class of conditionally Gaussian processes [24]. Their generative model consists of the head pose, face and background texture. In our application, however, we are recording the human subjects in front of a single color curtain without any structure so that their extension cannot improve the head pose estimation in our application.

Bergen et al. [14] proposed a hierarchical optical flow estimation and presented different 3D motion models such as a rigid head model. Many model-based coding algorithms are based on their work in order to predict the motion of the face [104, 90, 43, 126]. These methods among others slightly vary in the techniques to avoid error accumulation in the long-term parameter estimation. Liu et al. [90] as well as Eisert et al. [43] use a feedback loop to avoid error accumulation. After the head pose is estimated, the face model is

rendered in the new pose. The synthetically rendered image is used to estimate the head pose in the consecutive frame. Some model-based coding algorithms strive to rather maximize the peak signal-to-noise ratio than estimating the true 3D head pose. For instance, Eisert et al. [43] extend their basic algorithm by automatically calculating illumination parameters of the scene. Their best results are achieved by using a reflectance map [68], which increases the peak signal-to-noise ratio. In order to adapt their reflectance map to a particular scene a large number of parameters need to be determined. Thus, maximizing the peak signal-to-noise ratio may be achieved due to the large degree of freedom of the system, but the motion pararameters may not be more precisely estimated. In [36] a Kalman filter combines the estimated 2D flow field with edge force in order to avoid error accumulation. A difficulty is to appropriately combine the flow with the edge force. This difficulty can be avoided if an approach e.g. from [43] is taken.

Some algorithms estimate the movements of the human head and the non-rigid motion, which derives from human's facial expression [43, 36, 77, 75, 86, 143]. It has not been shown that the rigid motion estimation accuracy is increased. Firstly, some methods [86, 77] decouple the parameters and first estimate the rigid and afterwards the non-rigid motion. Secondly, the success of estimating the local motion strongly depends on the face model. Only local motions, which are defined by the face model, can be estimated. Hence, algorithms [75] using simple face models can only estimate a few local motions, e.g. a jaw drop. Other algorithms use sophisticated anatomically face models in order to allow the estimation of a wide range of local motions [43, 36, 77]. A difficulty using a complex face model is the adaption to an initial frame, especially to determine the initial facial expression. An inaccurate adaptation, however, may result in a poor parameter estimation [143]. Thirdly, since humans are able to perform a large varity of changing facial expressions, it is very difficult to describe them by one model. Because of the privously described obstacles and since we do not need the information of the current facial expression in our application, we are not estimating the local motions.

The work of Xiao et al. [144] uses optical flow to directly estimate the 3D head pose. Their main work deals with improving the robustness of optical flow by a combination of two techniques. Firstly, feature points are weighted in order to reduce the impact of outliers on the parameter estimation [124]. Secondly, the face texture is automatically updated during the tracking while contingently re-registering the current frame to a reference frame in order to prevent a drift. More details of their algorithm are given in Section 3.7. We regard the work of Xiao et al. [144] as a reference method, since they use a model-based approach, which directly estimates the 3D head pose, with a large number of promising extensions to improve the robustness and accuracy. Hence, their algorithm is able to estimate the 3D head pose without the limitations analyzed in other image-based animation systems [18, 47, 31]. Their approach, however, has still a number of issues, which are either not or only insufficiently addressed. While their work focuses to reliably track arbitrary image sequences, our main goal is to estimate the head pose as accurate as possible in image sequences, which fullfill the previously described requirements. Consequently, additional knowledge and constraints can be used to an overall improvement

of the accuracy of the parameter estimation. We concentrate on the main two issues regarding our application: the detection of outliers and the update of texture information.

A gradient-based algorithm estimates the motion of the face model by calculating the optical flow using the luminance values of feature points. A luminance value of a feature point can be considered as texture information. Most head pose estimation algorithms either use robust cost functions [70, 142] or weighted feature points [144, 104, 43, 2, 89, 17] in order to improve the robustness. The latter approach is usually used by gradient-based algorithms. We improve the approach of weighting feature points as presented in the reference method [144] by scrutinizing their proposed weights to our application and applying additional weights. In literature, a large number of weights are proposed, which we will examine and integrate in our algorithm, if they improve the accuracy of our algorithm.

Large out-of-plane rotations cannot be accurately estimated, because the texture information is largely changing. Therefore, a careful selection and update of texture information of feature points is important for an accurate motion estimation. In literature, the update of texture points has been widely neglected. Mostly either feature points are updated after each frame or feature points are only once initially selected. Both approaches have disadvantages, a drift may occur in the first while the second approach is unable to estimate large out-of-plane rotations. A few works do consider using dynamic templates. Vacchetti et al. [135] use additional reference frames, which are manually created during a training, in order to be able to estimate large head pose variations. A disadvantage of their apporach is that they require error-prone manual interference. Xiao et al. [144] as well as Anisetti et al. [2] propose to add additional reference frames while tracking the human. Their algorithms update the texture information after estimating the head pose resulting in an error accumulation. If the current estimated head pose is close to one of the previously stored reference frames, then this reference frame is used for re-alignment. Their approach relies on exchanging the texture of the entire face model, which disregards the valuable information provided by the current reference frame. Our proposed algorithm has the advantages that we consider our particular recording set-up to select appropriate reference frames and only update the texture of certain feature points. Furthermore, our algorithm automatically determines appropriate reference frames and does not require error-prone manual interference.

Eye Animation Systems

Eye animation systems (Figure 1.4) consist of a ECU and a rendering engine to synthesize animations [141]. The ECU consists of models controlling gaze patterns, blinks and the dynamics of human eye movements and sends generated eye control parameters to the rendering engine. Optionally eye animation systems have a unit extracting audio features from the spoken output, which are sent to the ECU. Recently, eye animations have received more attentions in the facial animation community [30, 87]. Eye animations are either controlled by conversational rules, statistical models or a combination of both.

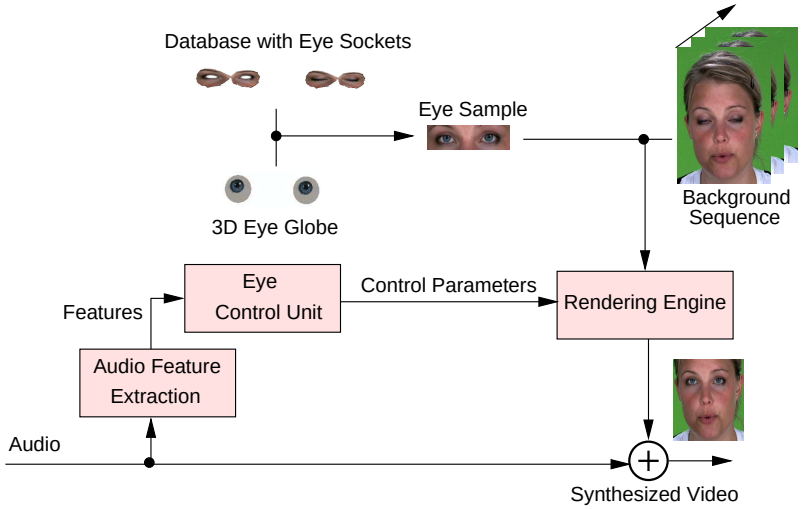

Figure 1.4: Image-based eye animation system: Initially phonetic and prosodic information are extracted from the audio. The eye control unit generates eye blinks and movements and sends control parameters to the rendering engine.

Conversational rules take the spoken language into account in order to determine gaze. For instance, when the speaker wants to give her turn of speaking to the listener, she usually gazes at the listener at the end of the utterance. Statistical models are based on measurements of the human eyes. For example, a statistical model may be used to determine the magnitude of a saccade. Saccades are rapid eye movements repositioning the eye gaze to new locations in the visual environment.

Cassell et al. [23] explore the problem of designing conversational agents with appropriate gaze behavior during dialogues with human users. They determine that gaze models should not only include turn-taking phenomena, but also the information structure of the propositional content of an utterance. Cassell et al. [22] describe a system which automatically generates and animates conversations between multiple human-like agents with facial expressions, intonation, eye gaze, head motion, and arm gestures. While a dialogue manager produces the conversation, a parallel transition network follows simple conversational rules. This network can execute an action in the simulation and simultaneously make state transitions either rule-based or probabilistically.

Colburn et al. [28] present behavior models of eye gaze patterns in the context of real-time verbal communication. The gaze model, which consists of a hierarchical state machine, is based on psychological studies and measurements. The state machine distinguishes between speaking and listening of the avatar. Each state has two modes, looking away and looking at the interlocutor and switches between these modes with a certain probability. Furthermore, the user's gaze, which is tracked by a camera, controls both modes. In experiments they investigate that even a simple gaze model induces changes in

human eye gaze behavior.

In the work of Poggi et al. [107] they generate facial expressions, such as gaze shifts and blinks, for speech. They concentrate on the visual display of intentions through facial animation based on semantic data. Performatives, e.g. a request or information, induce a defined facial expression. They also discuss how the degree of certainty, the power relationship, the type of social encounter, and the affective state effect the facial animation.

Heylen et al. [67] report an experiment that investigated the effects of different eye gaze behaviors of cartoon-like talking heads on the quality of human-machine dialogues. Instead of designing a precise gaze pattern model, they are interested in the effects of gaze on the dialogue quality. For this, they compare gaze patterns, which include some human-like gaze patterns with two other versions, in which the gaze shifts are kept minimal and randomly. Human-like gaze patterns are generated by considering simple rules taking the spoken output into account. Subjective tests are performed, in which participants are booking with the assistance of a cartoon-like talking-head two concerts. Results show that even a crude implementation of human gaze patterns has significant positive effects on the conversation.

Fakuyama et al. [54] implemented a two state Markov model to control gaze movements based on results of psychological experiments. They varied three parameters of their model: amount of gaze, duration of gaze and the gaze points while averted. The transition probabilities of the two-state Markov model are adapted to these variations. In experiments an eyes-only agent indicates that gaze parameters can reliably induce impressions.

Garau et al. [55] investigated the impact of eye gaze of humanoid avatars on participant's perception during a conversation. Responses to dyadic conversations of avatars with audio-only, random head and eye movements, and "inferred-gaze" avatar, which combines a talking and listening mode for the eye gaze, are analyzed. The eye gaze model is based on social psychology research on the differences in gaze patterns while speaking and listening. In order to evaluate the impact of gaze, a role-playing negotiation task was developed. Results indicate, that the random-gaze did not provide a significant improvement over pure audio. However, the inferred-gaze avatar significantly outperforms the other two types.

The work so far described mainly focused on eye engagements by using simple statistical models, by distinguishing between listening and talking mode as well as simple conversational rules between humans and virtual characters. Conversational rules, however, are only capable to well model gaze patterns for exactely pre-defined conversational settings with a limited set of predefined sentences. Hence, if eye animations need to be automatically generated to arbitrary spoken output, e.g. as provided by an audio book, these models are not appropriate. The described works included neither the dynamics of eye movements nor eye blinks.

Cosatto [30] implements eye blinks and eye globe motions with probabilistic state machines. Blink patterns are modeled with a Markov model. Eye movements are generated by moving the gaze point between the interlocutor and an imaginary desk. Additional

random saccadic eye motions are performed. The animation is rendered by combining a 3D eye globe model with image-based rendering of the eye area. However, the proposed method is not further evaluated, so that we cannot grade the quality of the modeling. In general, the proposed control models are very simple and do not take audio features into account. Hence, his models do not distinguish between listening and talking, although human gaze patterns strongly vary [4]. Thus, the animations will not appear video-realistic.

Lee et al. [85, 87] propose a comprehensive statistical model to control eye motion developed from their own gaze tracking analysis of real people. The avatar can be in one of the three cognitive states: listening, talking or thinking. For this, a human operator manually segments the original eye-tracking video. Each state has its own model and probabilities to perform saccades. The gaze pattern consists of two states, looking at and away from the interlocutor. These states as well as the execution of saccades are modeled by measured probability distributions. In addition, their model considers head rotations. Thus, if the direction of the head rotation has changed and its amplitude is larger than a threshold, then a gaze shift accompanying the head motion is created. Subjective tests with a cartoon-like avatar show the proposed statistical model achieves higher scores than a stationary and random eye movements model. We regard their work as a reference method, since on the one hand their model takes many details into account such as modeling the dynamics of saccades and considering a thinking mode. On the other hand their approach of designing a model based on measured statistics guarantees a great flexibility. However, their method has still several shortcomings. Manually distinguishing between talking and thinking mode is redundant, since the spoken language already contains this information. Their designed model does not distinguish between small gaze shifts with short fixations within the facial area of the interlocutor and looking away. Hence, a large saccade may be executed with a short duration in the look away state resulting in flickering eye motions, which look unnatural. They do not model eye blinks.

In the work of Deng et al. [38], an automated eye animation is presented in which new eye motions and blinks are synthesized by texture synthesis. A database with information of the eye blink signal and eye gaze position is generated by analyzing recorded sequences. The initial eye gaze and eye blink position are randomly selected. Afterwards a number of similar samples of the database are determined and one is randomly selected, which determines the new gaze and blink position. In this way an animation path for eye gaze and eye blinks is generated. Statistical dependencies between gaze and blinks are not explicitly considered. The animation is evaluated by subjective tests. The animations are synthesized by using the model derived in [85] for gaze and eye blinks generated by a Poisson distribution or the proposed model, which was favored. However, their designed eye animation is only generated for listening mode, while our main focus is on synthesizing realistic eye movements while talking.

Masuko and Hoshino [96] present a method to generate eye and head movements synchronized with the conversation of virtual actors. Their model is mainly based on works of [54] and [85]. Moreover, their model induces head rotations due to the distance between the current and subsequent gaze point. They subjectively evaluate their results by

comparing video with only blinks, only eye movement without head movement and the proposed method, which achieved the best scores. A comparison with [85] is missing. Since the control of head rotation is challenging using image-based rendering, we do not take their extension into account.

The work of Ma and Deng [93] is based on training a Dynamic Coupled Component Analysis (DDCA) model by recording the gaze and head movements of six humans for 150 s each. Eye animations are generated by the DDCA model using a head motion sequence as input. They evaluated their approach by subjective tests. For this, they created animations of the original recorded sequence, the model derived in [85] as well as in [38] and their proposed model. The original was favored followed by their new approach, [85] and [38]. However, their work raises questions. Firstly, their approach of predicting eye movements from head motion is founded by the research carried out in [53]. In [53] the saccade kinematics consisting of the velocity and duration of the saccade movement are investigated in different experiments in which the saccade kinematics of three monkeys during gaze shifts of e.g. 35-40° accompanied by large head movements are measured. The saccade velocity and duration is slightly varying, but most likely not visible in an animation sequence, which is rendered with 25 frames per second. The behavior of eye movements of humans in a conversation, however, is not considered in their work. In addition, only a dependency between saccade kinematics and head movements are presented in [53] but not a model describing the coupling. Secondly, the DDCA-based model only learns linear dependencies between eye and head movements. Therefore, it is questionable that this simple model is able to control gaze shifts by only using head movements. How can gaze shifts be predicted? Thirdly, their model has the drawback that the speech content is not integrated, although a large amount of social psychology research on gaze patterns has been carried out. Fourthly, in their evaluation they generated the video samples of the reference method [85] only by taking the simple head-gaze model into account. However, one part of the model of Lee et al. [85] generates gaze shifts to speech content, which is not considered in their results. Since a meaningful model of predicting the saccade kinematics from head movements is not available yet, we do not consider this relationship in our work. Furthermore, we do not compare our method with [93], since we are interested in realistic eye movements during face-to-face interactions.

In the previous paragraphs it has been shown that a number of issues are either not or only insufficiently addressed. Instead of manually adding the mode "thinking" as in [85], we want to automatically control gaze movements with spoken language. This approach has the advantage of generating eye movements to arbitrary spoken output without manual interference. Although, Cranach [32] already showed a correlation between eye movements and blinks in 1969, eye control models proposed in literature neglected this aspect. Our designed model will integrate possible statistical dependencies. In order to model eye blinks, we will explore if eye blinks can be controlled by spoken output as investigated in a psychological study [29]. Since flickering eye motions result in unnatural-looking animations, we will refine the model of generating saccades proposed in [85] based on a careful analysis of mutual gaze. In order to overall improve the quality of the

eye animation, we will integrate research results from studies about eye movement physiology carried out in the field of neurophysiology and ophthalmology [21]. Furthermore, an appropriate image-based rendering engine needs to be developed.

Thesis Overview

This thesis is organized as follows. In Section 2, the fundamentals of human eyes like anatomy are briefly explained. Moreover, the movement physiology and the different types of eye movements as well as social psychological studies on gaze and eye blinks are explored. Section 3, is the central chapter of computer vision in this thesis. Here, the fundamentals of computer vision, including the scene model, 3D motion of objects and model camera are briefly summarized. The algorithm to estimate the head pose and its extensions to improve the robustness and accuracy are described. In Section 4, the experimental set-up of recording human subjects in a two-way conversation as well as different types of algorithms to analyze the recorded data are explained. The determination of statistical properties of eye blinks and eye movements, which are later needed by the ECU, are investigated in Section 5. Here new dependencies, e.g. the relation between spoken output and eye blinks, are explored, which have not undergone a quantitative analysis before. Section 6, "Eye Control Unit", is the central part of this thesis. Here, the designed ECU for the eye animation is described in detail. The section starts with a new model to generate eye movements, while taking eye movement physiology into account. In the following sub-sections, new models to create eye movements and blinks for listening as well as talking mode are derived. While in listening mode two independent models to steer eye movements and blinks are designed, in talking mode we derive a new model, which considers the dependencies between eye movements and blinks. Furthermore, we refine the typical eye gaze model consisting of mutual gaze and gaze away by considering small gaze shifts performed across the face of the interlocutor. Section 7, explores the designed rendering engine, which synthesizes novel eye animations with the control parameters provided from the ECU. The results of the improved head pose estimation algorithm and a subjective test to evaluate our proposed eye animations are provided in Section 8. This thesis ends with a conclusion in Section 9.

2 Fundamentals of Human Eyes

This chapter explains the fundamentals of human eyes. Firstly, the anatomy of the eye globe and muscles are explicated. Secondly, the movement physiology and the different types of eye movements are briefly explained. Thirdly, a short introduction to eye blinks is given. Finally, social psychological studies on gaze and eye blinks relevant for designing the control unit are explored.

2.1 Anatomy of the Human Eye

2.1.1 Anatomy of the Globe

For the human eye to produce a clear vision it is comprised of several discrete parts, which must operate together like a sophisticated camera. For a better understanding, the various ocular structures are contemplated for the scenario of the path of light traveling through the eye (Figure 2.1).

1. *Sclera* — Opaque, fibrous, protective layer of the eye containing collagen and elastic fibers.

2. *Choroid* — Middle eye skin, the middle layer of the globe, between retina and sclera.

3. *Cornea* — The cornea is the transparent front part of the eye, which covers the pupil, iris and anterior chamber. The ray of light first encounters the tear film. Because the surface of the eye must maintain a moist state at all times, there are glands located in and near the eyelids, which produce tears and an oil that are used together for moistening the eye.

4. *Iris* — The iris is the colored ring of tissue suspended behind the cornea and immediately in front of the lens. The opening in the center of the iris is called pupil. The iris regulates the amount of light entering the eye by adjusting the size of the pupil and can be compared to the diaphragm of a camera.

5. *Lens* — When objects are near or far from us, the task of the lens is to adapt the focus so that the objects are viewed clear and sharp. The crystalline lens, which is the next structure the ray of light encounters, changes it's shape to accomplish the focus shift. Over the years the lens becomes less flexible and changing focus is becoming more difficult.

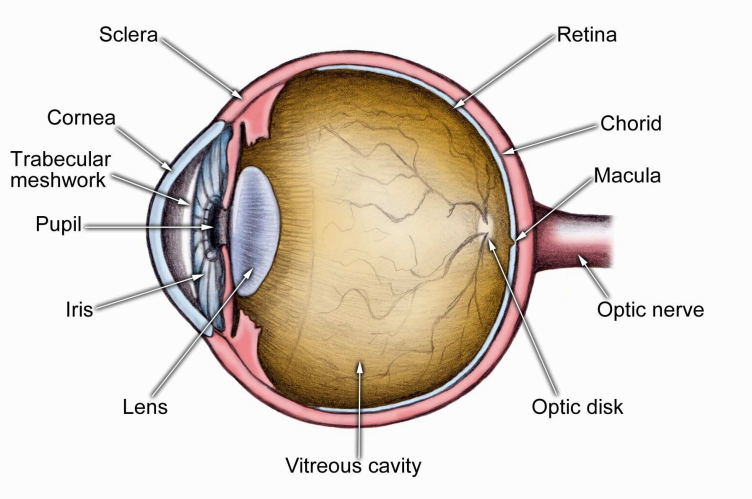

Figure 2.1: Anatomy of the Globe [45].

6. *Vitreous* — This substance fills the body of the eye and is firmly attached to the retina. In the early stages of life it is podgy and clear but can become more fluid and like water, which may result in detaching from the retina.

7. *Retina* — The innermost wall of the eye is the retina, which acts like a target of a camera to the ray of light. The retina, which contains photoreceptor cells, converts the rays of light to electrical signals, which are then carried to the brain. The photoreceptor cells can be divided into rods, which are more light-sensitive, and cones, that allow the perception of color and fine details. The retina can be divided into two parts. While the outer parts, that mainly contain rods, are for the peripheral vision, the inner or center area, also called macula, is responsible for the color and high resolution vision. Fovea is the name for the very center of the macula and has a high concentration of cones, which make it the only part of the retina capable of 20/20 vision1.

8. *Optic Nerve* — The optic nerve gathers the information from the retina as electrical signals and delivers these signals to the brain. There the information is formed and interpreted into a visual image. The position where the nerve enters the globe is called 'blind spot', since there are no rods or cones. Normally, a person does not notice this blind spot since rapid movements of the eye and processing in the brain compensate for this absent information.

2.1.2 Eye Muscles

In general, eyes move within six degrees of freedom three translations within the socket and three rotations. There mainly are seven muscles responsible for eye movements [35]:

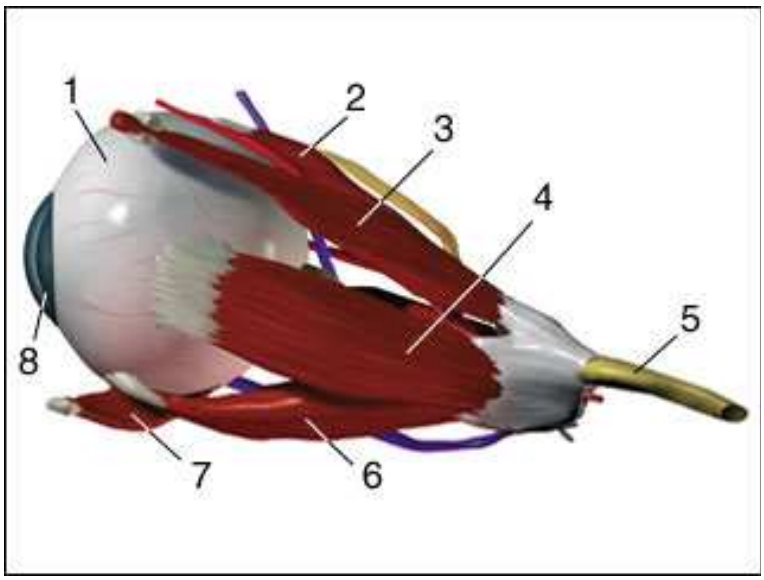

Figure 2.2: Eye Muscles [65]: 1. Eye globe, 2. Superior rectus muscle, 3. Oblique superior muscle, 4. Medial rectus muscle, 5. Optic nerve, 6. Inferior rectus muscle, 7. Oblique inferior muscle, 8. Cornea.

- *Superior rectus muscle* — Upward
- *Lateral rectus muscle* — Sideways outward
- *Inferior rectus muscle* — Downward
- *Medial rectus muscle* — Sideways inward
- *Levator palpebrae muscle* — Raises the upper eyelid
- *Oblique superior muscle* — Downward and inside
- *Oblique inferior muscle* — Upward and outside,

which are displayed in Figure 2.2. Within the orbit (eyehole), the eye is surrounded by flexible fat pads, which besides rotation, permit globe translation up to a certain limit. Usually, this translation is negligibly small.

Vertical saccadic eye movements cause the levator palpebrae muscle to move the eyelid. For upward saccadic eye movements, the levator palpebrae muscle activity raises the eyelid, while downward saccades transiently turns off the levator palpebrae allowing the eyelid to fall [46].

2.2 Eye Movement Physiology

Human eyes detect objects over a visual angle of 200°, but the high resolution part of the retina the fovea covers only 1° of the visual field [41]. The neuronal system involved in generating eye movements is known as oculomotor system. The oculomotor system has two major functions, to bring targets onto the fovea and to keep them there. There are five basic types of eye movements used to reposition the fovea [21], which can be classified by their function. The following two movements stabilize or fixate the retina over a stationary object of interest when the head moves:

1. *Vestibulo-ocular reflex (VOR)* — During head movements in any direction the vestibular sense organs signal how fast the head is rotating, so that the oculomotor system responds by rotating the eyes in the opposite direction. This stabilizes the eyes relative to the external world and keeps visual images fixed on the fovea. This reflex is almost always active in order to allow humans to see clearly while moving. The latency is only approximately 14 ms. [21]

2. *Optokinetic nystagmus* — Optokinetic nystagmus uses visual input to hold images stable on the fovea during sustained or slow head rotation. The time course of this movement is of a saw tooth pattern. The latency is between 60 and 100 ms. [21]

Three movements keep the fovea on a visual target:

1. *Saccadic movements* — Saccadic eye movements, which are both voluntary and reflex movements, take their name from the French 'saccade', meaning 'jerk', and connoting a discontinuous, stepwise manner of movement as opposed to a fluent, continuous one [12]. Saccades are rapid eye movements used in repositioning the fovea from one gaze position to another. These are rapid changes in the point of fixation, due to fixation reflexes, or during scanning of objects, reading, or under voluntary control. They must balance the conflicting demands of speed and accuracy, in order to minimize both time spent in transit and making corrective movements. Saccades range in duration between 10 and 100 ms [41] and can be as fast as $600\,\frac{\text{degrees}}{\text{s}}$ with an extremely high initial acceleration up to $3 \cdot 10^5\,\frac{\text{degrees}}{\text{s}^2}$. There are few conventions used in the eye movement literature to describe saccades such as direction and magnitude. Under direction of a saccade we understand the direction in which the eye rotates. The amplitude or magnitude of a saccade describes the angle the eye globes rotate from starting position to end position. Naturally occurring saccades rarely have a magnitude greater than 15° [6]. Furthermore, saccades are characterized by its duration and velocity. Saccadic eye movements essentially depend on the magnitude of the movement but there are also a number of secondary, physiological and psychological factors.

The duration D^s is the amount of time necessary to execute the saccade and proportional to its magnitude A^s within a restricted range of 5° to 50° [12]

$$D^s = D_0^s + d^s \cdot A^s, \tag{2.1}$$

with the slope d^s between 1.5 to 2.4 $\frac{\text{ms}}{\text{degrees}}$ and D_0^s ranges from 20 to 30 ms.

By definition, the mean velocity (MV^s) of saccadic eye movements is calculated directly from their duration and amplitude [12]

$$\text{MV}^s = \frac{A^s}{D^s}. \tag{2.2}$$

However, the peak velocity of saccades is independent of the duration D^s. The peak velocity initially rises in proportion to the saccadic amplitude and then saturates as the amplitude becomes larger [12].

Saccades can be divided into a number of subclasses [12]. Here only the most important types of saccades are briefly explained:

a) *Refixation saccades* — Guide the eyes to targets in the visual environment that have been previously selected.

b) *Scanning saccades* — Move from feature to feature and can be thought of as sequences of refixation saccades.

c) *Gaze saccades* — Are accompanied by a head rotation in the same direction [16, 137]. Large gaze shifts always include a head rotation under natural conditions.

d) *Microsaccades* — Are tiny movements (as low as 0.05°), which occur continuously and have random direction with respect to fixation target. The role of microsaccades in visual perception has been a highly debated topic which is still largely unresolved [21].

2. *Smooth pursuits* — Smooth pursuits move the eyes to keep a single moving target on the fovea. The oculomotor system computes how fast a target is moving and moves the eyes accordingly. Humans can usually not execute a smooth pursuit without a real target. This movement is quite different from saccades, because it is smooth and slower (up to 150 $\frac{\text{degrees}}{\text{s}}$). In reality both movements, saccades and smooth pursuit, are often used to track moving targets.

3. *Vergence system* — Vergence aligns the eyes to look at targets with different depths. During vergence movements the eyes move in opposite directions to keep the image of the target aligned on each fovea.

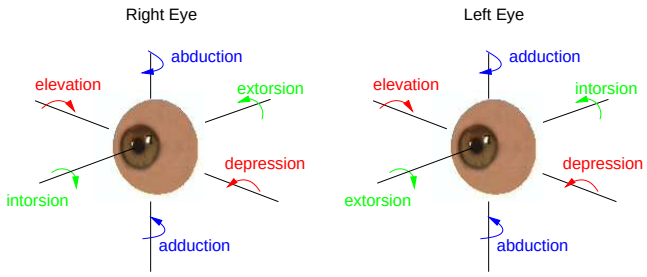

Figure 2.3: Rotational directions of both eyes.

The movement of the eye globe approximately follows a rotation of its center (Figure 2.3). A few terms are briefly explained as used in medical literature for describing eye globe rotations [20]. Each extraocular muscle rotates the globe in a specific direction. Elevation and depression of the eye are termed sursumduction and desursumduction and have the same direction for both eyes. Adduction and abduction are respectively denoted as nasal and temporal eye movements. Incycloduction or incyclotorsion is denoted as intorsion, while excycloduction or excyclotorsion is denoted as extorsion. Both describe a torsional eye movement.

The line of sight is defined as a vector from the globe center to the pupil. If the eye looks straight forward, then a plane perpendicular to the line of sight and intersecting with the globe center can be placed. In this plane the horizontal and vertical axis of rotations are located. Eye positions can be classified by their rotational properties [78]:

- *Primary position* — The eye looks straight forward with the head straight and fixed.
- *Secondary position* — From the primary position, a rotation either around the horizontal or vertical axis is performed.
- *Tertiary position* — From the primary position, a rotation around the horizontal and vertical axis is performed.

2.3 Eye Blinks

Blinking is a rapid closing and opening of the eyelid. It is an essential function of the eyes that fullfills the biological purpose to wet the cornea and remove irritants from the surface of the cornea. The blink rate reflects psychological arousal. The normal, resting blink rate of a human being is 12.5 closures per minute, with the average blink lasting one quarter of a second. Significantly faster rates may reflect emotional stress [61], as arousal, e.g. in a fight. The rate decreases during concentrated thinking or attention to visual objects [108].

In general, there are three types of eye blinks [4]:

- *Periodic blinking* — This occurs on average 12.5 times per minute in adults, though much less for infants. The main purpose is to lubricate and protect the eyes. The blinks last 0.2 to 0.3 s, though the period of darkness is only 0.1 s or less. People are aware of the flicker but not of the blackness, probably because of the persistence of images. Blinks occur particularly when attention is relaxed. The brain centres responsible for periodic blinking are not known.
- *Reflex blinking* — This occurs in response to irritation of the cornea, eyelashes or other parts of the eye or face, by the approach of objects, bright lights or sudden noises.
- *Voluntary blinks* — At different speed these blinks, of one or both eyes, can be produced voluntarily.

2.4 Social Psychological Studies on Gaze Patterns and Eye Blinks

A large amount of research has been carried out to study gaze behavior. Direct gaze versus gaze away is fairly easy to measure with respect to facial expressions and therefore this part has been extensively studied in the framework of nonverbal behavior. A comprehensive overview of social psychological studies on gaze is given in [50].

In general, eyes play an important role in the non-verbal communication of humans. Gazes are important information in face-to-face interaction. It serves at least four distinct communicative functions [79, 4]: providing feedback, regulating conversation flow, communicating emotional information and avoiding distraction by restricting visual input. The most important conclusions from these studies can be summarized as [4]:

1. *Sending social signals and regulating the flow of conversation*
   a) Speakers use glances to full-stop signals and other grammatical breaks.
   b) Glances are used by speakers to emphasize particular words or phrases.
   c) The speaker stops and looks at the listener to indicate a turn.
   d) Listeners use glances to indicate continued attention and willingness to listen.
2. *Channel to receive information*
   a) During breaks speakers look up to obtain feedback from the interlocutor.
   b) Listeners look at speakers in order to study their facial expressions.
3. *Expressing emotions*
   a) Humans tend to look down in case of sadness or shame.

4. *Aversion of gaze*

    a) Speakers do not look all the time at the interlocutor to avoid an overflow of information.

    b) Aversion of gaze can act as a deliberate signal that a person is thinking.

In general, the amount of time a person looks at the interlocutor is much higher if he is listening instead of talking [4]. Kendon [79] found additional changes in gaze direction, such as the speaker looking away from the interlocutor at the beginning of an utterance and towards the interlocutor at the end.

Ellyson et al. [44] analyzed the conversation of females with different levels of expertise on a series of topics. Speakers gazed a significantly greater proportion of the time while speaking about familiar topics, compared to their gaze behavior talking about less familiar topics. Expert knowledge may reduce the labor involved in discussing certain topics, enabling a speaker to spend more time monitoring his or her audience without disrupting the flow of discourse [50]. A person trying to be persuasive looks more at the interlocutor, and is also perceived as persuasive [97].

There are obviously dependencies between spoken language and gaze patterns, because speakers control their conversational flow by the spoken language as well as by gaze. For instance, at the end of an utterance, the speaker stops talking while simultaneously looking at the interlocutor.

Not only gaze patterns but also eye blinks do have a dependency with spoken language as explored by Condon and Ogston [29]. They investigated that eye blinks mainly occur during vocalization at the beginning of words or utterances, the initial vowel of a word and following the termination of a word.

Hence, it is reasonable to design models, which control eye movements and blinks by spoken language.

3 Head Pose Estimation

In this section first the fundamentals of computer vision, including the scene model, 3D motion of objects and model camera are briefly summarized (Section 3.1 to 3.4). Afterwards the designed algorithm to estimate the head pose is explained (Section 3.5) and extensions to improve the robustness (Section 3.6).

3.1 Scene Model

The scene model (Figure 3.1) defines the real world with a parametric model. It consists of a model camera, a model illumination of the scene, and model objects. An accurate description of the real world by the scene model is aspired. However, there are inconsistencies between the real world and assumed models, because the properties and characteristics of the real world are very complicated and not known. Hence, the scene model simplifies the real world.

In the scene the model objects and the model camera are related to each other through a fixed, right-handed, orthonormal world coordinate system $(\mathbf{X},\mathbf{Y},\mathbf{Z})$. A point $\mathbf{P}$ in the scene can be defined uniquely with its 3D world coordinates $\mathbf{P} = [X,Y,Z]^T$. Relative to this world coordinate system, various local coordinate systems can be defined, e.g. a camera coordinate system $(\mathbf{X}^c,\mathbf{Y}^c,\mathbf{Z}^c)$.

We assume that the scene is illuminated by an ambient lighting, which is a simplification of the real world. The ambient light source is not located at any particular place, and it spreads in all directions uniformly. It lights all objects in the same way, because it is not dependent on any other lighting factors like direction or position. Furthermore, we assume that all model objects have diffuse reflecting surfaces. Only under these assumptions, brightness changes in the image sequence can be associated with movements of objects in the scene.

The model camera describes the projection of 3D points $\mathbf{P}^c$ given in camera coordinates onto the camera target. In this thesis, the camera coordinate system is aligned with the world coordinate system. Hence, $\mathbf{P} = \mathbf{P}^c$ and 3D points given in world coordinates can be directly mapped onto the camera target.

The model object approximates the 3D shape of a real object by a triangular mesh, which is defined by its 3D vertices and connectivity. We assume that the real objects in the scene are rigid and therefore the model object, too. Each model object has its own local object coordinate system $(\mathbf{X}^o,\mathbf{Y}^o,\mathbf{Z}^o)$.

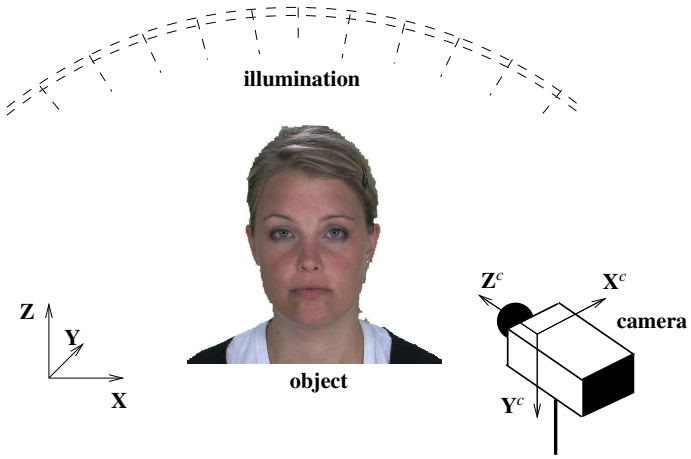

Figure 3.1: 3D scene model

3.2 Rotation in 3D Space

Rotation in 3D can be defined in many ways. In most of the literature, either Euler angles or Euler's theorem are used to define rotation [34]. The angle of rotation about a coordinate axis is called Euler angle. A sequence of such rotations is represented by rotation matrices. Euler's theorem [19] states that a rotation is defined by an axis and an angle rotation and can be represented by a quaternion [59].

In this work we use rotation matrices as well as quaterions to describe 3D rotations. The algorithm to estimate the head pose uses rotation matrices (Section 3.2.1), while the eye control model uses quaternions to express the rotation of the eye globe (Section 3.2.2).

3.2.1 Rotation Matrices

When Euler angles are used, a general orientation is written as a series of rotations about three mutually orthogonal axes in space. In a Cartesian coordinate system usually the **X**, **Y** and **Z** axes are used (Figure 3.2). Hence, the rotation matrix **R** is defined by three consecutive rotations around **X**-, **Y**- and **Z**-axis with the rotation angles ω_x, ω_y and ω_z

$$
\begin{aligned}
\mathbf{R} &= \mathbf{R}_z\mathbf{R}_y\mathbf{R}_x \\
&= \begin{bmatrix} \cos(\omega_z) & -\sin(\omega_z) & 0 \\ \sin(\omega_z) & \cos(\omega_z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\omega_y) & 0 & \sin(\omega_y) \\ 0 & 1 & 0 \\ -\sin(\omega_y) & 0 & \cos(\omega_y) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\omega_x) & -\sin(\omega_x) \\ 0 & \sin(\omega_x) & \cos(\omega_x) \end{bmatrix} \\
&= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}
\end{aligned} \tag{3.1}
$$

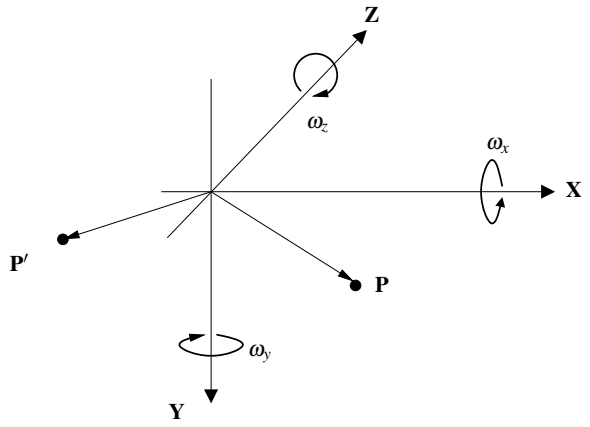

Figure 3.2: Rotation of point **P** around the **X**-, **Y**- and **Z**-axis to **P′**.

with the components

$$
\begin{aligned}
r_{11} &= \cos(\omega_y)\cos(\omega_z) \\
r_{12} &= \sin(\omega_x)\sin(\omega_y)\cos(\omega_z) - \cos(\omega_x)\sin(\omega_z) \\
r_{13} &= \cos(\omega_x)\sin(\omega_y)\cos(\omega_z) + \sin(\omega_x)\sin(\omega_z) \\
r_{21} &= \cos(\omega_y)\sin(\omega_z) \\
r_{22} &= \sin(\omega_x)\sin(\omega_y)\sin(\omega_z) + \cos(\omega_x)\cos(\omega_z) \\
r_{23} &= \cos(\omega_x)\sin(\omega_y)\sin(\omega_z) - \sin(\omega_x)\cos(\omega_z) \\
r_{31} &= -\sin(\omega_y) \\
r_{32} &= \sin(\omega_x)\cos(\omega_y) \\
r_{33} &= \cos(\omega_x)\cos(\omega_y).
\end{aligned}
$$

To execute rotations with rotation matrices has several shortcomings [34]. For instance, the user must express rotations in respect to a certain convention that defines in which order the three basis rotations are applied. Different conventions yield different results. Furthermore, it is possible to create a series of rotations, where one degree of freedom in the rotation is lost, which is called gimbal lock.

The point **P** is rotated to **P′** by applying the rotation matrix **R** in the following way

$$\mathbf{P'} = \mathbf{RP}. \tag{3.2}$$

3.2.2 Quaternions

To execute rotations with quaternions is more intuitive and offers several advantages over rotation matrices [34]. A quaternion $q = (q_0, q_1, q_2, q_3)$ is a hyper-complex number of

rank 4, where q_0, q_1, q_2 and q_3 are scalars [81]. An alternative way of representing a quaternion is to define a scalar part q_0 and vector part $\mathbf{q} = \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$ in $\mathbb{R}^3$ leading to

$$\begin{aligned} q &= q_0 + \mathbf{q} \\ &= q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3. \end{aligned} \tag{3.3}$$

The sum of two quaternions $p = p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3$ and $q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$ is defined by adding the corresponding components

$$p + q = (p_0 + q_0) + \mathbf{i}(p_1 + q_1) + \mathbf{j}(p_2 + q_2) + \mathbf{k}(p_3 + q_3). \tag{3.4}$$

The multiplication of these two quaternions is defined as

$$\begin{aligned} pq &= (p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3)(q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3) \\ &= p_0q_0 - \mathbf{pq} + p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p} \times \mathbf{q} \end{aligned} \tag{3.5}$$

with the following fundamental products

$$\begin{aligned} \mathbf{i}^2 &= \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \\ \mathbf{ij} &= \mathbf{k} = -\mathbf{ji} \\ \mathbf{jk} &= \mathbf{i} = -\mathbf{kj} \\ \mathbf{ki} &= \mathbf{j} = -\mathbf{ik}. \end{aligned} \tag{3.6}$$

The complex conjugate of a quaternion q^*, which is also equal to the inverse q^{-1}, is defined as

$$q^* = q_0 - \mathbf{i}q_1 - \mathbf{j}q_2 - \mathbf{k}q_3. \tag{3.7}$$

The norm $N(q)$ of a quaternion q, denoted as $|q|$ is defined as

$$\begin{aligned} N(q) &= \sqrt{(q^*q)} \\ &= \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}. \end{aligned} \tag{3.8}$$

Each unit quaternion q can be associated with a rotation by an angle $\theta_q \in [0, \pi]$ about an axis $\mathbf{v}$ in the following form

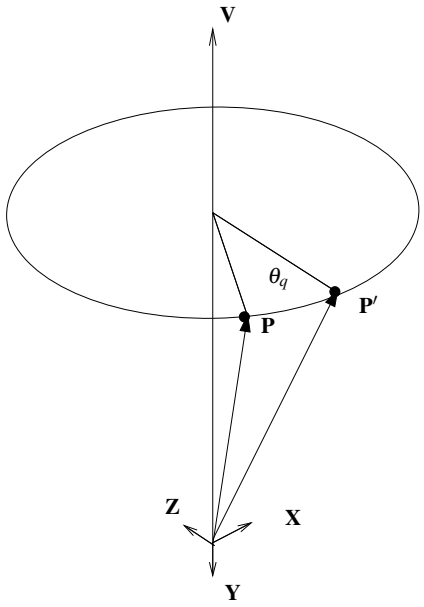

Figure 3.3: Rotation of the 3D point **P** around **v** with the angle θ_q to **P**′.

$$\begin{aligned} q_0 &= cos\left(\frac{\theta_q}{2}\right) \\ |\mathbf{q}| &= \sqrt{q_1^2+q_2^2+q_3^2} = sin\left(\frac{\theta_q}{2}\right) \\ \mathbf{v}_q &= \frac{\mathbf{q}}{|\mathbf{q}|} \end{aligned} \tag{3.9}$$

with

$$q = cos\left(\frac{\theta_q}{2}\right) + sin\left(\frac{\theta_q}{2}\right)\mathbf{v}_q. \tag{3.10}$$

The rotation of a 3D point $\mathbf{P} = [X,Y,Z]^T$ is accomplished with the quaternion rotation operator $L_q(P_q)$ as illustrated in Figure 3.3. For this **P** is converted to its corresponding quaternion $P_q = 0 + \mathbf{i}X + \mathbf{j}Y + \mathbf{k}Z$. $L_q(P_q)$ rotates the quaternion P_q about an axis $\mathbf{v}_q$ by an angle θ_q resulting in

$$P'_q = L_q(P_q) = qP_qq^* \tag{3.11}$$

with the quaternion $P'_q = 0 + \mathbf{i}X' + \mathbf{j}Y' + \mathbf{k}Z'$. The new 3D position $\mathbf{P} = [X',Y',Z']^T$ is extracted from P'_q.

It is often more convenient in applications to rewrite Equation (3.11) in matrix form. Then the quaternion P'_q is equal to

$$\begin{aligned}
\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} &= \mathbf{M} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \\
&= \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \\
&= \begin{bmatrix} 2(q_0^2+q_1^2)-1 & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & 2(q_0^2+q_2^2)-1 & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & 2(q_0^2+q_3^2)-1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \qquad (3.12)
\end{aligned}$$

Note that matrix $\mathbf{M}$ is equivalent to the rotation matrix $\mathbf{R}$. Hence, the conversion of a rotation matrix $\mathbf{R}$ to a quaternion q and vice versa is easily to achieve. Given the rotation matrix $\mathbf{R}$ we can determine the corresponding unit quaternion q as

$$\begin{aligned}
q_0 &= \frac{\sqrt{r_{11}+r_{22}+r_{33}+1}}{2} \\
q_1 &= \frac{r_{23}-r_{32}}{4q_0} \\
q_2 &= \frac{r_{31}-r_{13}}{4q_0} \\
q_3 &= \frac{r_{12}-r_{21}}{4q_0}. \qquad (3.13)
\end{aligned}$$

In order to convert a unit quaternion q into its corresponding rotation angles, we first convert the unit quaternion q into its corresponding matrix $\mathbf{M}$ (Equation (3.12)). Afterwards the rotation angels ω_x, ω_y and ω_z can be extracted [120].

3.3 Rigid Motion Model

A 3D face model describes the geometric shape of the head by approximating the surface by triangles. Each triangle consists of three 3D points and each of these points $\mathbf{P}^o = [X^o, Y^o, Z^o]^T$ is given in object coordinates. The center of the object coordinate system $(\mathbf{X}^o, \mathbf{Y}^o, \mathbf{Z}^o)$ is located in the barycenter of the object $\mathbf{G} = [G_x, G_y, G_z]^T$. The position and orientation of the object coordinate system to the world coordinate system is described by the barycenter of the object $\mathbf{G}$ and an initial rotation $\mathbf{R}_0$. The transformation of an object point $\mathbf{P}^o$ into world coordinates is defined as:

$$\mathbf{P} = \mathbf{R}_0\mathbf{P}^o + \mathbf{G}. \qquad (3.14)$$

The rigid movement of the object can be described as a rotation $\mathbf{R}$ in the barycenter of the object and translation $\mathbf{T} = [t_x, t_y, t_z]^T$ of the object (Figure 3.4). Hence, the new position of a point $\mathbf{P}$ is

$$\mathbf{P}' = \mathbf{R}(\mathbf{P} - \mathbf{G}) + \mathbf{G} + \mathbf{T}, \tag{3.15}$$

with the new object center $\mathbf{G}' = \mathbf{G} + \mathbf{T}$.

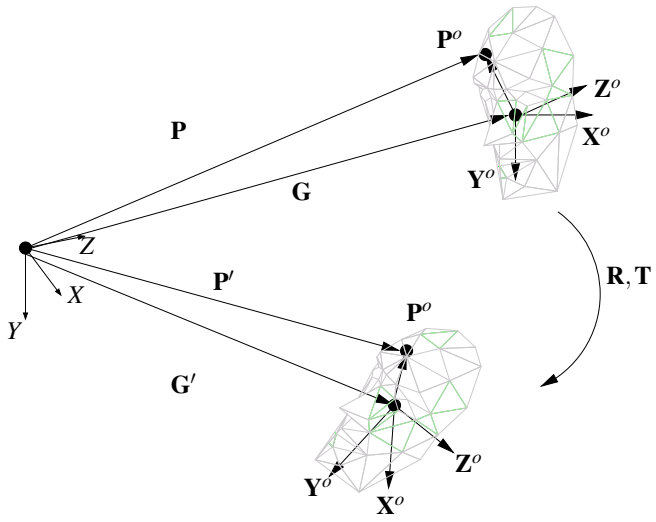

Figure 3.4: The object point $\mathbf{P}^o$ can be transformed in world coordinates $\mathbf{P}$ by an initial rotation $\mathbf{R}_0$ and translation of the object's barycenter $\mathbf{G}$. Afterwards the rigid object is rotated $\mathbf{R}$ and translated $\mathbf{T}$ to the new position $\mathbf{P}'$, whereas the object point $\mathbf{P}^o$ remains fixed.

3.4 Camera Model

The camera model describes the projection of a 3D object onto a sampled and quantized 2D image. The model camera consists of a series of three consecutive parts: the perspective projection, the lense model and the CCD sensor model (Figure 3.5). The parameters of these models can be divided into intrinsic and extrinsic camera parameters. The latter describe the relation between object and camera coordinate systems. Since the camera and world coordinate are already aligned, the estimation of the extrinsic camera parameters is redundant. The intrinsic camera parameters are initially estimated with a camera calibration technique [131, 49].

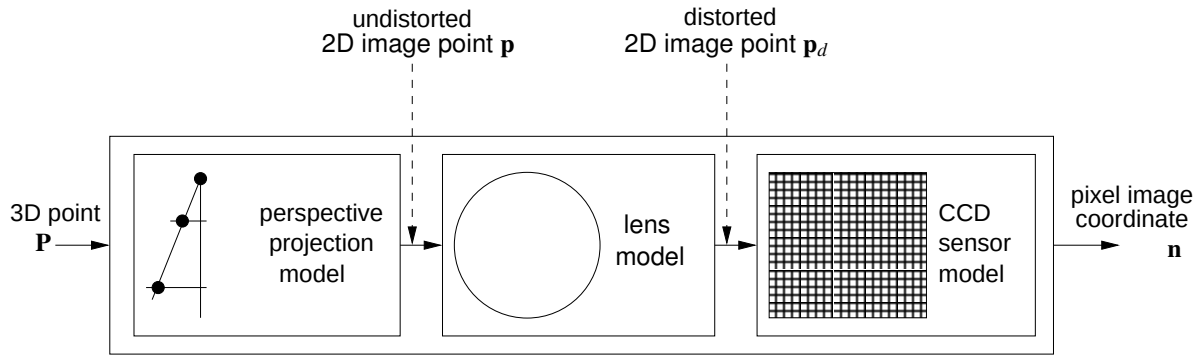

Figure 3.5: The camera model consists of a perspective projection model, a lens model and a CCD camera sensor model.

3.4.1 Perspective Projection Model

The perspective projection describes the relation between a 3D point $\mathbf{P}$ and its corresponding 2D feature point $\mathbf{p}$ in the image plane as shown in Figure 3.6. The distance from the camera center to the principal point $\mathbf{c}$ is the focal length f. Then the following ratio relates $\mathbf{P} = [X,Y,Z]^T$ and $\mathbf{p} = [x,y]^T$:

$$\begin{aligned} \frac{x}{f} &= \frac{X}{Z} \\ \frac{y}{f} &= \frac{Y}{Z} \end{aligned} \tag{3.16}$$

3.4.2 Lens Model

The perspective projection models the characteristics of the ideal pin-hole camera. However, due to the system of lenses in a real camera the rays are non-linearly distorted. The dominant geometrical distortion is the radial lens distortion [131], which accounts for barrel or pincushion distortions. The radial lens distortion is approximated by its first two coefficients κ_1 and κ_2. With this model the relation between the undistorted image point $\mathbf{p} = [x,y]^T$ and the corresponding distorted image point $\mathbf{p}_d = [x_d,y_d]^T$ is described according to [122]

$$\begin{aligned} x &= x_d + x_d(\kappa_1 r_d^2 + \kappa_2 r_d^4) \\ y &= y_d + y_d(\kappa_1 r_d^2 + \kappa_2 r_d^4), \end{aligned} \tag{3.17}$$

with $r_d = \sqrt{x_d^2 + y_d^2}$.

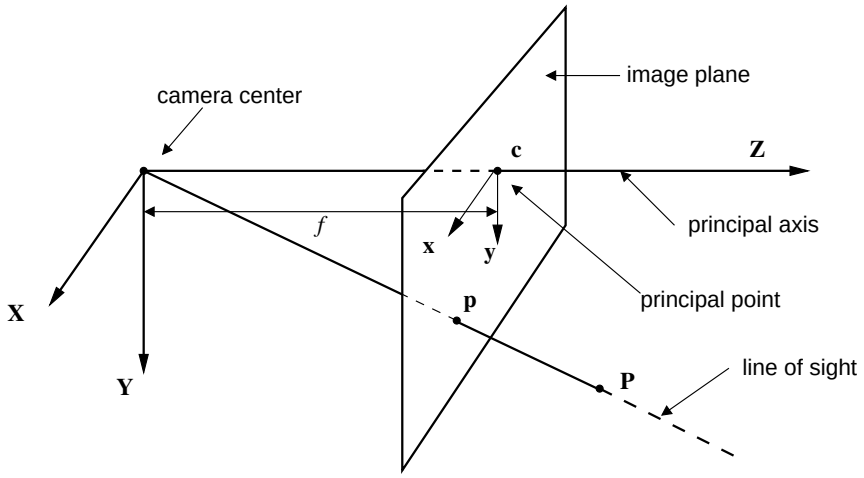

Figure 3.6: Perspective projection: A 3D point **P** is mapped along the line of sight to the 2D point **p**, which is located in the image plane. The distance between principal point **c** and camera center is the focal length f.

3.4.3 CCD Camera Sensor Model

A CCD camera sensor consists of $N_x \times N_y$ CCD elements, which are denoted as pixels, as shown in Figure 3.7. In general, a single CCD element or pixel has a rectangular shape. Its size is defined by the scaling factors s_x and s_y, which are used to relate pixels to the metric mm. The origin of the camera sensor is the upper left part, whereas the principal point or image center is located at $\mathbf{c} = [c_x, c_y]^T$. The pixel image coordinate $\mathbf{n} = [n_x, n_y]^T$ and its corresponding 2D feature point $\mathbf{p}_d = [x_d, y_d]^T$ are related as

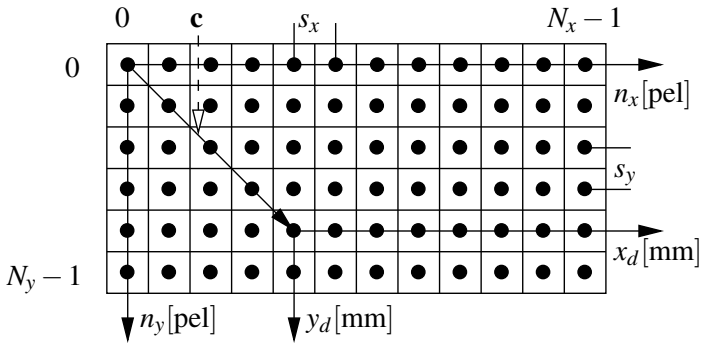

Figure 3.7: CCD camera sensor

$$\begin{aligned} x_d &= s_x(n_x - c_x) \\ y_d &= s_y(n_y - c_y). \end{aligned} \tag{3.18}$$

A property of CCD cameras is to add noise n_G to the sampled luminance value. n_G consists of different components such as thermal noise. A comprehensive overview of camera noise is given in [74]. For the sake of simplicity, we assume that n_G has a Gaussian distribution with mean zero and variance σ_{n_G}. The sampled luminance value $\mathbf{I}(\mathbf{n},t)$ is the sum of the true luminance value $\breve{\mathbf{I}}(\mathbf{n},t)$ and noise

$$\mathbf{I}(\mathbf{n},t) = \breve{\mathbf{I}}(\mathbf{n},t) + n_G(\mathbf{n}). \tag{3.19}$$

3.5 Motion Estimation Algorithm

In this section the algorithm to determine head pose parameters is explored (Figure 3.8). Initially, the intrinsic camera parameters are estimated in order to compensate lens distortion and to estimate the focal length. Afterwards the 3D face model and image sequence are registered in a common coordinate frame. For this, facial features such as eye corners and nostrils are manually selected. Afterwards, the initial head pose, which is defined by $\mathbf{R}_0$ and $\mathbf{T}_0$, is calculated by a pose estimation technique [37], which uses corresponding 2D image coordinates and 3D feature points of the face model. After this initialization the head pose is estimated for the video sequence, which works as follows.

Let $\mathbf{I}(\mathbf{p},t)$ be the brightness at the 2D image point $\mathbf{p} = [x,y]^T$ in the image $\mathbf{I}$ recorded at the point of time t. The initial frame ($t = 0$) to which the rigid face model is adapted is denoted as $\hat{\mathbf{I}}(0)$ and referred to as reference frame. The reference frame contains an area of texture information marked by the face model and the corresponding head pose $(\mathbf{R}_0, \mathbf{T}_0)$. In this area, a number of feature points containing the texture information are defined by $\mathbf{p} \in \Omega$. These points must have distinct visual characteristics such as a high gradient. We use the Harris detector for feature point detection [63], which are tracked throughout the image sequence. The number of feature points is a trade-off between accuracy and computational effort. Since our focus is to estimate the accurate head pose, we use a large number between 1500 and 2500 feature points. It is important, that an adequate number of feature points are located at the edge of the face.

The 3D point corresponding to $\mathbf{p}$ is denoted as $\mathbf{P} = [X,Y,Z]^T$. The projection of a moving 3D point onto the image plane is described by a parametric motion model defined as $\mathbf{F}(\mathbf{p},\lambda)$, parameterized by $\lambda = [\omega_x, \omega_y, \omega_z, t_x, t_y, t_z]^T$ with $\mathbf{F}(\mathbf{p},\mathbf{0}) = \mathbf{p}$.

The problem of motion estimation of a rigid face model can be stated as (Figure 3.9): At time t, a 3D point $\mathbf{P}$ is moved from its original position as seen in the reference frame to a new position $\mathbf{P}'$. Similarly, the 2D image point $\mathbf{p}$ on the camera target with the luminance

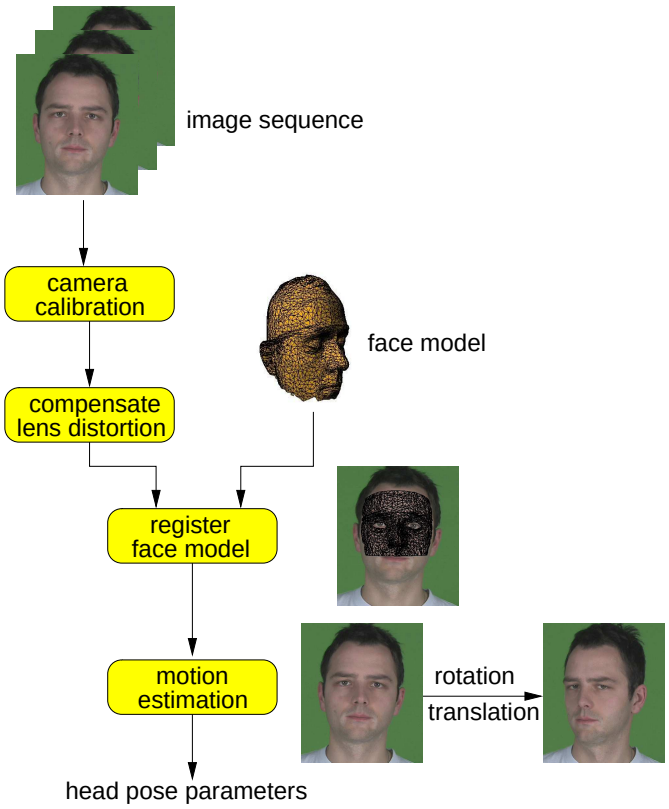

Figure 3.8: Block diagram of estimating head pose parameters. A head-shoulder image sequence and a face model need to be provided.

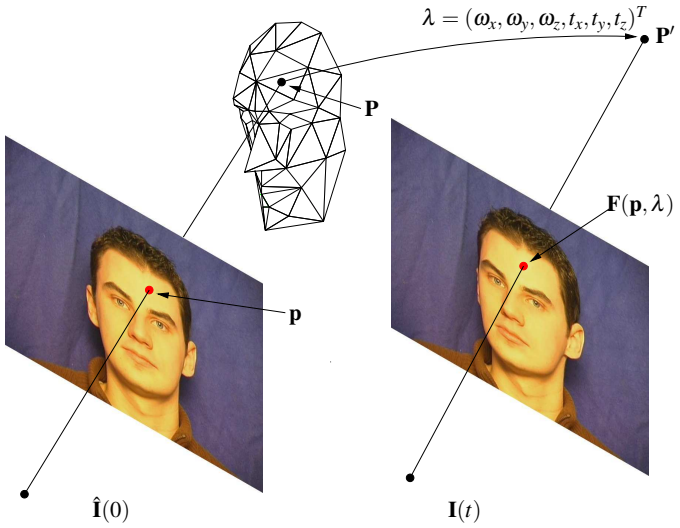

Figure 3.9: Motion of a 2D feature point from $\mathbf{F}(\mathbf{p},\mathbf{0})$ in the reference frame $\hat{\mathbf{I}}(0)$ to $\mathbf{F}(\mathbf{p},\lambda)$ in image $\mathbf{I}(t)$, while the corresponding 3D feature point moves from $\mathbf{P}$ to $\mathbf{P}'$.

value $\hat{\mathbf{I}}(\mathbf{p},0)$ in the reference frame, is moved from $\mathbf{F}(\mathbf{p},\mathbf{0})$ to the position $\mathbf{F}(\mathbf{p},\lambda)$ in frame $\mathbf{I}(t)$. Assuming ambient illumination and diffuse reflecting surfaces,

$$\hat{\mathbf{I}}(\mathbf{p},0) = \mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) \qquad \forall \mathbf{p} \in \Omega \tag{3.20}$$

holds. In practise, the luminance values in Equation (3.20) are not equal but sligtly different leading to the residual error $r'(\mathbf{p};\lambda)$, which is difined as

$$r'(\mathbf{p};\lambda) = \mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},0) \qquad \forall \mathbf{p} \in \Omega. \tag{3.21}$$

For motion estimation we minimize the cost function

$$\begin{aligned} C'(\lambda) &= \sum_{\mathbf{p}\in\Omega} r'^2(\mathbf{p};\lambda) \\ &= \sum_{\mathbf{p}\in\Omega} \left[\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},0)\right]^2. \end{aligned} \tag{3.22}$$

We can solve Equation (3.22) with optical flow, which assumes a linear signal model.

For this $\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t)$ is approximated by a Taylor polynomial of first order

$$\begin{aligned}\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) \approx & \mathbf{I}(\mathbf{F}(\mathbf{p},0),t) + \left.\frac{\partial \mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t)}{\partial \mathbf{F}(\mathbf{p},\lambda)}\right|_{\lambda=0} \cdot \\ & \left.\frac{\partial \mathbf{F}(\mathbf{p},\lambda)}{\partial \lambda}\right|_{\lambda=0} \lambda,\end{aligned} \tag{3.23}$$

where $\mathbf{I}_\mathbf{p}(\mathbf{p},t) = \left.\frac{\partial \mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t)}{\partial \mathbf{F}(\mathbf{p},\lambda)}\right|_{\lambda=0}$ is the spatial gradient and $\mathbf{F}_\lambda(\mathbf{p},\mathbf{0}) = \left.\frac{\partial \mathbf{F}(\mathbf{p},\lambda)}{\partial \lambda}\right|_{\lambda=0}$ the derivative of the parametric motion model.

$\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t)$ in Equation (3.21) can be replaced by Equation (3.23) and the following residual error $r(\mathbf{p};\lambda)$ is obtained

$$r(\mathbf{p};\lambda) = \mathbf{I}_\mathbf{p}(\mathbf{p},t)\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})\lambda + \mathbf{I}(\mathbf{p},t) - \hat{\mathbf{I}}(\mathbf{p},0) \qquad \forall \mathbf{p} \in \Omega. \tag{3.24}$$

Using the new residual error in Equation (3.24) leads to the following cost function

$$\begin{aligned}C'(\lambda) \approx C(\lambda) &= \sum_{\mathbf{p}\in\Omega} r^2(\mathbf{p};\lambda) \\ &= \sum_{\mathbf{p}\in\Omega} \left[\mathbf{I}_\mathbf{p}(\mathbf{p},t)\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})\lambda + \mathbf{I}(\mathbf{p},t) - \hat{\mathbf{I}}(\mathbf{p},0)\right]^2.\end{aligned} \tag{3.25}$$

In order to find the minimum of Equation (3.25), $C(\lambda)$ is differentiated with respect to λ and set equal to zero. Then λ is iteratively calculated by means of incremental motion parameters λ_i

$$\begin{aligned}\lambda_i = - & \left[\sum_{\mathbf{p}\in\Omega} [\mathbf{I}_\mathbf{p}(\mathbf{p},t)\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})]^T [\mathbf{I}_\mathbf{p}(\mathbf{p},t)\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})]\right]^{-1} \\ & \sum_{\mathbf{p}\in\Omega} [\mathbf{I}(\mathbf{p},t) - \hat{\mathbf{I}}(\mathbf{p},0)][\mathbf{I}_\mathbf{p}(\mathbf{p},t)\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})]^T\end{aligned} \tag{3.26}$$

After every estimation of λ_i, the face model is moved by λ_i. The updated face model with the new feature positions, e.g. $\mathbf{p} \rightarrow \mathbf{F}(\mathbf{p},\lambda_1)$ defines the starting point of the new estimation, which is continued until the motion parameters converge, i.e. $\lambda_i \rightarrow \mathbf{0}$.

Subpel Interpolation

Solving Equation (3.26) iteratively requires the current luminance values of the image feature points $\mathbf{p} \in \Omega$. For this, the 2D image point $\mathbf{p}$ is converted in pixel image coordinates $\mathbf{n}$. The location of $\mathbf{n}$ is not necessarily directly located at an pixel image coordinate,

but rather at a subpel position. Hence, the corresponding luminance value of $\mathbf{n}$ has to be interpolated from the surrounding pixel image coordinates.

The Shannon sampling theorem states, that a perfect reconstruction of sampled values is possible, if the sampling frequency is at least two times the signal frequency. If we assume that this condition is satisfied, then the continuous signal can be reconstruced with a 2D sinc function. Hence, the luminance value of an image point given in pixel image coordinates $\mathbf{n} = [n_x, n_y]^T$ is equal to

$$\mathbf{I}(\mathbf{n},t) = \sum_{i_x=-\infty}^{+\infty} \sum_{i_y=-\infty}^{+\infty} \mathbf{I}([i_x,i_y]^T,t) \cdot \mathrm{si}\left[\frac{\pi}{s_x}(n_x s_x - i_x s_x)\right] \cdot \mathrm{si}\left[\frac{\pi}{s_y}(n_y s_y - i_y s_y)\right] \tag{3.27}$$

with i_x and i_y denoting the sampling values.

Since a CCD array can only store a finite number of luminance values a windowed sinc function is used to achieve satisfying interpolation results with few samples. For this a Blackman window [62] with length $2M+1$ is used. Thus, Equation (3.27) is rewritten as

$$\mathbf{I}(\mathbf{n},t) \approx \sum_{i_x=-M}^{M} \sum_{i_y=-M}^{M} w_B(i_x) \cdot w_B(i_y) \cdot \mathbf{I}([i_x,i_y]^T,t) \cdot \mathrm{si}\left[\frac{\pi}{s_x}(n_x s_x - i_x s_x)\right] \cdot \mathrm{si}\left[\frac{\pi}{s_y}(n_y s_y - i_y s_y)\right] \tag{3.28}$$

with

$$w_B(m) = 0.42 + 0.5\cos\left(\frac{2\pi m}{2M+1}\right) + 0.08\cos\left(\frac{4\pi m}{2M+1}\right) \qquad -M \leq m \leq M. \tag{3.29}$$

Spatial Gradient

The method of numerical differentiation is very important since differences between first order pixel differences and higher order central differences are very noticeable [10]. It is commonly recommended to apply spatio-temporal pre-smoothing to the image sequence before differentiation. In this way the amplification of additive noise is prevented when combined with gradient operations and the data locally behaves like tilt planes.

The spatial gradient $\mathbf{I}_\mathbf{p} = [\mathbf{I}_x, \mathbf{I}_y]^T$ can be well approximated by not only considering the Taylor polynomial of first order but of second order as presented in [15]. In this way also higher spatial frequencies are well approximated.

In order to determine the spatial gradient of a 2D image point $\mathbf{p}$, it is converted to pixel image coordinates $\mathbf{n}$. Then the spatial gradient in x-direction $\mathbf{I}_x$ of $\mathbf{n} = [n_x, n_y]^T$ is approximated as the average gradient in the current image $\mathbf{I}(t)$ and reference image $\hat{\mathbf{I}}(0)$ resulting in

$$\mathbf{I}_x(\mathbf{n},t) = \frac{1}{2}\left[\bar{\mathbf{I}}_x(\mathbf{n},t) + \bar{\hat{\mathbf{I}}}_x(\mathbf{n},0)\right] \tag{3.30}$$

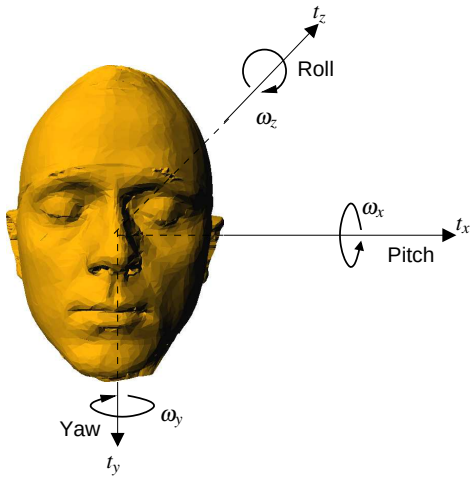

Figure 3.10: The pose of the face model is defined by its three rotation angles $(\omega_x, \omega_y, \omega_z)$ and translation $\mathbf{T} = [t_x, t_y, t_z]^T$.

with

$$\bar{\mathbf{I}}_x(\mathbf{n},t) = \sum_{i=1}^{N+1} h_H(i) \frac{(-1)^{i-1}}{i} \left[\mathbf{I}(n_x+i, n_y, t) - \mathbf{I}(n_x-i, n_y, t)\right]. \tag{3.31}$$

In each row 2N sample points are considered to determine the gradient. The convergence characteristic of the truncated series is improved by a Hann window [62]

$$h_H(i) = \frac{1}{2}\left[1 - cos\left(\frac{\pi(N+1+i)}{N+1}\right)\right]. \tag{3.32}$$

The spatial gradient in y-direction $\mathbf{I}_y$ is computed accordingly.

Object Model

The geometric shape of the subject's head is approximated by a 3D face model, in order to be able to estimate spatial movements. In order to obtain a precise face model, we use a laser scanner, which automatically captures the precise 3D shape of objects by using a 3D shape acquisition system. 3D scans have a high resolution with over 250000 vertices (Figure 3.10). The mesh vertices of the 3D scan are semantically unrelated and facial animation parameters as defined in MPEG-4 cannot be directly extracted from the scan, if needed. The out-of-plane rotations are denoted as yaw and pitch and the in-plane rotation is denoted as roll.

Parametric Motion Model

The parametric motion model $\mathbf{F}$ describes the motion of a 2D feature point $\mathbf{p}$ from $F(\mathbf{p},\mathbf{0})$ to $\mathbf{F}(\mathbf{p},\lambda)$ by first moving $\mathbf{P}$ to $\mathbf{P}'$ and then projecting the 3D point onto the camera target (Figure 3.9). The motion of $\mathbf{P}$ to $\mathbf{P}'$ is described by a rotation and translation (Equation 3.15). Since the relative rotation described by λ_i (Equation 3.26) can be expected to be very small, a linearized version of the rotation matrix of Equation (3.1), which is denoted as $\Delta\mathbf{R}$, is used. Under the assumption $\cos(x) \approx 1$ and $\sin(x) \approx x$ the rotation matrix in Equation (3.1) is simplified to

$$\Delta\mathbf{R} \approx \begin{bmatrix} 1 & -\omega_z & \omega_y \\ \omega_z & 1 & -\omega_x \\ -\omega_y & \omega_x & 1 \end{bmatrix}. \tag{3.33}$$

A small motion of a feature point $\mathbf{P} = [X,Y,Z]^T$ according to Equation (3.15) and using the linearized rotation matrix of Equation (3.33) results in

$$\mathbf{P}' = \begin{bmatrix} (X-G_x)-(Y-G_y)\omega_z+(Z-G_z)\omega_y+G_x+t_x \\ (X-G_x)\omega_z+(Y-G_y)-(Z-G_z)\omega_x+G_y+t_y \\ -(X-G_x)\omega_y+(Y-G_y)\omega_x+(Z-G_z)+G_z+t_z \end{bmatrix} \tag{3.34}$$

with the new position $\mathbf{P}'$. Hence, the parametric motion model $\mathbf{F}$ using Equation (3.16) and (3.34) describes the motion of a 2D image point $\mathbf{p}$ as

$$\mathbf{F}(\mathbf{p},\lambda) = f \begin{bmatrix} \frac{(X-G_x)-(Y-G_y)\omega_z+(Z-G_z)\omega_y+G_x+t_x}{-(X-G_x)\omega_y+(Y-G_y)\omega_x+(Z-G_z)+G_z+t_z} \\ \frac{(X-G_x)\omega_z+(Y-G_y)-(Z-G_z)\omega_x+G_y+t_y}{-(X-G_x)\omega_y+(Y-G_y)\omega_x+(Z-G_z)+G_z+t_z} \end{bmatrix} \tag{3.35}$$

in which f is the focal length of the camera.

In order to determine $\mathbf{F}_\lambda(\mathbf{p},\mathbf{0})$ the partial derivative with respect to λ is calculated:

$$\begin{aligned} \mathbf{F}_\lambda(\mathbf{p},\mathbf{0}) &= \left.\frac{\partial\mathbf{F}(\mathbf{p},\lambda)}{\partial\lambda}\right|_{\lambda=0} \\ &= \frac{f}{Z^2}\begin{bmatrix} \mathbf{F}_{\omega_x} & \mathbf{F}_{\omega_y} & \mathbf{F}_{\omega_z} & \mathbf{F}_{t_x} & \mathbf{F}_{t_y} & \mathbf{F}_{t_z} \end{bmatrix} \end{aligned} \tag{3.36}$$

with

$$
\begin{aligned}
\mathbf{F}_{\omega_x} &= \begin{bmatrix} -X(Y-G_y) \\ -Y(Y-G_y)+Z(Z-G_z) \end{bmatrix} \\
\mathbf{F}_{\omega_y} &= \begin{bmatrix} X(X-G_x)+Z(Z-G_z) \\ (X-G_x)Y \end{bmatrix} \\
\mathbf{F}_{\omega_z} &= \begin{bmatrix} -(Y-P_y)Z \\ (X-G_x)Z \end{bmatrix} \\
\mathbf{F}_{t_x} &= \begin{bmatrix} Z \\ 0 \end{bmatrix} \\
\mathbf{F}_{t_y} &= \begin{bmatrix} 0 \\ Z \end{bmatrix} \\
\mathbf{F}_{t_z} &= \begin{bmatrix} -X \\ -Y \end{bmatrix}.
\end{aligned}
$$

Hierarchical Motion Estimation

Since gradient-based motion estimation algorithms assume a linear signal model, only small motions between two consecutive frames are accurately estimated. A hierarchical implementation of the gradient-based motion estimation algorithm [14] enables to determine larger motions between consecutive frames. For this two resolution pyramids one of the reference frame $\hat{\mathbf{I}}(0)$ and one of frame $\mathbf{I}(t)$ are determined (Figure 3.11). An image with a lower resolution is obtained by low pass filtering the image with a higher resolution and subsampling by a factor of two. The motion parameters are iteratively estimated by first using the image with the lowest resolution and finally the original image (top to bottom of the resolution pyramid). As a side effect the computational effort is reduced by a hierarchical implementation.

Summary of the Motion Estimation Algorithm

In summary, the previously described motion estimation algorithm consists of different components:

1. *subpel interpolation* We suggest to use the si interpolation, which allows a very accurate interpolation.

2. *spatial gradient* The spatial gradient is calculated as proposed in [15]. This approach allows to approximate the gradient by a second order Taylor polynomial.

3. *object model* A precise 3D face model is used to model the human head. Many algorithms in literature use very simple models e.g. [83, 75, 144].

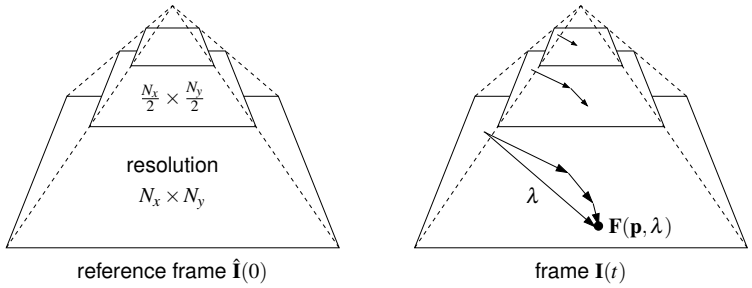

Figure 3.11: Hierarchical implementation of the gradient-based motion estimation algorithm enables to determine larger motions between consecutive frames. Here the original frame has a resolution of $N_x \times N_y$ and the frames with lower resolution are reduced by a factor of two. The motion parameters between the reference frame $\hat{\mathbf{I}}(0)$ and frame $\mathbf{I}(t)$ are determined from top to bottom.

4. *parametric motion model* The motion model as used in e.g. [144, 43, 104] gives the opportunity to directly estimate the 3D head pose.

5. *hierarchical motion estimation* The hierarchical implementation as suggested in [14] of the motion estimation allows to estimate larger motions between two frames.

Each component is selected in the best way to satisfy our requirements to accurately measure the head pose.

3.6 Improving the Robustnes of Motion Estimation

The gradient-based algorithm tries to track feature points $\mathbf{p} \in \Omega$ with distinctive visual characteristics throughout the image sequence $\mathbf{I}$. Under perfect conditions Equation (3.20) holds. In real image sequences, however, one or more of the following effects may occur:

1. Ambient illumination as well as diffuse reflection as assumed in Equation (3.20) are usually not present.

2. Occlusions of the face.

3. Camera noise.

4. Violations of the assumed linear signal model in Equation (3.23).

5. Local deformations.

6. Large head pose variations.

(a) (b)

Figure 3.12: (a) Light box. (b) Ambient lighting and diffuse reflections ensure constant illumination in different head poses.

Therefore, the luminance value of a feature point $\mathbf{p}$ in the reference $\hat{\mathbf{I}}(\mathbf{p},0)$ and current frame $\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t)$ are not equal, but may be at most similar resulting in

$$\hat{\mathbf{I}}(\mathbf{p},0) \approx I(\mathbf{F}(\mathbf{p},\lambda),t) \qquad \forall\, \mathbf{p} \in \Omega. \tag{3.37}$$

In the following, we briefly analyze the different types of violations and their impact on the motion estimation in our experimental set-up.

1. During the recording the human subject is placed in a light box [57], which is wraped with diffusor foil (Figure 3.12a). This foil diffuses the incident light resulting in an ambient illumination. Furthermore, the human subject wears a special type of make-up, which prevents specular reflections on the face. Therefore, we are able to provide almost perfect ambient illumination, while preventing specular reflections on the face. The results of a recoding of a human subject in a light box are depicted in Figure 3.12b.

2. In our recordings the face remains always visible and consequently occlusions do not occur.

3. Since professional studio cameras are used, induced camera noise is relatively low. The camera producer guarantees a signal to noise ration of 54 dB in our used mode.

4. Violations of the assumed linear signal model are reduced due to a high spatial and temporal sampling frequency as well as a hierarchical motion estimation (Section 3.5). The image sequences are recorded with 60 Hz progressive and an image resolution of 880×720 pel.

5. Local deformations regularly occur since the recorded subject is talking and varying his facial expressions while being recorded.

6. Large head pose variations may also occur during a conversation.

While the first four effects are neglectable, local deformations as well as large head pose variations need to be taken into account to achieve a prescise head pose estimation. In the following two Sections, solutions to these problems are presented. In order to prevent the influence of local deformations on the rigid motion estimation, we need to eliminate those feature points, which are contaminated by local motions (Section 3.6.1). Those feature points are denoted as outliers. Large head pose variations are accurately estimated with an automatic update of texture information (Section 3.6.2).

3.6.1 Weighting of Feature Points

The field of robust statistics addresses the problem that parametric models are usually only approximations of the phenomenece being modeled. In particular, outliers violate the assumptions of parametric models. Hence, one main goal of robust statistics is to identify outliers and to increase the robustness of the estimation. The breakdown point, which describes the minimum fraction of outlying data that can cause an estimate to diverge arbitrarly far from the true estimate, is most commonly used to characterize robustness [115]. A comprehensive overview of robust parameter estimation in computer vision can be found in [124].

If the errors of the luminance values of the feature points are independent and have a Gaussian distribution, then the optimal estimator is a least square minimization (Equation (3.25)) [8]. The breakdown point of a least square minimization, however, is zero, since a single outlier can lead the least square fit arbitrarly far away from the true fit. Hence, more robust estimators need to be used, since outliers may be present in our application.

In order to improve the least square minimization MLE-estimators or short M-estimators can be used. Note, that for the sake of simplicity we use the same notation to describe the MLE-estimator as used to describe the head pose estimation, e.g. Equation (3.25). In general, the MLE-estimator minimizes the following term

$$\underset{\lambda}{\operatorname{argmin}} \sum_{\mathbf{p}\in\Omega} \rho_M \left(\frac{r(\mathbf{p};\lambda)}{\sigma_M} \right), \tag{3.38}$$

where $\rho_M(u)$ is a robust cost function and σ_M a scale factor. The minimization of Equation (3.38) is solved by finding λ such that

$$0 = \sum_{\mathbf{p}\in\Omega} \psi_M \left(\frac{r(\mathbf{p};\lambda)}{\sigma_M} \right) \frac{\partial r(\mathbf{p};\lambda)}{\partial \lambda} \frac{1}{\sigma_M}, \tag{3.39}$$

where $\psi_M(u) = \frac{\partial \rho_M(u)}{\partial u}$. In literature a large number of estimators have been proposed, e.g. Huber [70]. A common next step [70] is to introduce a weight function $\psi_M(u) =$

$\zeta(u) \cdot u$ and to solve

$$0 = \sum_{\mathbf{p} \in \Omega} \zeta^{\mathbf{p}} \left(\frac{r(\mathbf{p};\lambda)}{\sigma_M} \right) \frac{\partial r(\mathbf{p};\lambda)}{\partial \lambda} \frac{1}{\sigma_M^2} r(\mathbf{p};\lambda), \tag{3.40}$$

kown as "iteratively reweighted least squares" (IRLS). The idea of IRLS is to assign weights to the residuals in order to control them. Heigh weights are assigned to 'good' data and low weights to outliers. Using the residual error in Equation (3.24) and solving Equation (3.40) leads to the incremental motion parameters λ_i

$$\begin{aligned} \lambda_i = - & \left[\sum_{\mathbf{p} \in \Omega} \zeta^{\mathbf{p}} [\mathbf{I}_{\mathbf{p}}(\mathbf{p},t) \mathbf{F}_\lambda(\mathbf{p},\mathbf{0})]^T [\mathbf{I}_{\mathbf{p}}(\mathbf{p},t) \mathbf{F}_\lambda(\mathbf{p},\mathbf{0})] \right]^{-1} \\ & \sum_{\mathbf{p} \in \Omega} \zeta^{\mathbf{p}} [\mathbf{I}(\mathbf{p},t) - \hat{\mathbf{I}}(\mathbf{p},0)] [\mathbf{I}_{\mathbf{p}}(\mathbf{p},t) \mathbf{F}_\lambda(\mathbf{p},\mathbf{0})]^T. \end{aligned} \tag{3.41}$$

The weight $\zeta^{\mathbf{p}}$, which is determined for each feature point $\mathbf{p} \in \Omega$ after each iteration, consists of different sub-weights. In literature, a large number of sub-weights have been proposed. In the following, the sub-weights used in our motion estimation system are analyzed.

1. The first sub-weight ζ_I relates to the residual error $r'(\mathbf{p};\lambda)$ of Equation (3.21) and is defined as [17]

$$\zeta_I^{\mathbf{p}} = \exp\left(-\frac{[r'(\mathbf{p};\lambda)]^2}{2\sigma_I^2} \right) \qquad \forall \mathbf{p} \in \Omega, \tag{3.42}$$

   with the standard deviation σ_I. The standard deviation σ_I is calculated by selecting the median from all residuals resulting in

$$\sigma_I = 1.4826 \cdot \underset{\mathbf{p} \in \Omega}{\text{median}} |\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},0)|. \tag{3.43}$$

   The median is multiplied by 1.4286 in order to convert the median into the standard deviation of a normal distribution [115]. This approach has the advantage that outliers do not influence the calculation of the variance.

2. The sub-weight ζ_I is compensated by the sub-weight ζ_G as proposed by [144], which is useful in gradient-based techniques. The basic idea is to give feature points with a large spatial gradient $\mathbf{I}_{\mathbf{p}}$ also a larger weight, since these feature points usually have a large residual error and therefore a small weight ζ_I. As a result the

sub-weight $\zeta_G^{\mathbf{p}}$ is introduced, which considers the spatial gradient and is calculated as

$$\zeta_G^{\mathbf{p}} = c_6 \left(1 - \exp\left(-\frac{\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)}{2c_7^2}\right)\right) \qquad \forall \mathbf{p} \in \Omega, \tag{3.44}$$

with c_6 and c_7 as scalars, in which c_6 is reduced each iteration. The sub-weights $\zeta_I^{\mathbf{p}}$ and $\zeta_G^{\mathbf{p}}$ are added, so that they may compensate each other. For instance, a corner in an image will have a large gradient (large $\zeta_G^{\mathbf{p}}$), whereas little motion results in a large luminance difference and therefore a small weight $\zeta_I^{\mathbf{p}}$. Hence, by combining both weights a corner, which may have valuable information, is considered by the algorithm.

3. If a feature point $\mathbf{p}$ is not visible, then this feature point obtains the sub-weight $\zeta_V^{\mathbf{p}} = 0$. Checking for visibility is necessary, since we are using a precise face model of the human subject. Hence, some feature points may be occluded in certain head poses. This sub-weight is determined as

$$\zeta_V^{\mathbf{p}} = \begin{cases} 1 & ;\text{visible} \\ 0 & ;\text{else} \end{cases} \qquad \forall \mathbf{p} \in \Omega \tag{3.45}$$

Note, this check can be easily and without significant computational effort performed by using the z-buffer of openGL.

4. Our analysis revealed that the additional sub-weight $\zeta_E^{\mathbf{p}}$ to determine outliers has been very useful [104, 43]. For this sub-weight, we determine the displacement v_d between two consecutive frames as

$$v_d = \frac{|r'(\mathbf{p};\boldsymbol{\lambda})|}{|\mathbf{I}_{\mathbf{p}}|}. \tag{3.46}$$

If the displacement exceeds a pre-set threshold ξ_d, then the current feature point is labeled as an outlier

$$\zeta_E^{\mathbf{p}} = \begin{cases} 1 & ;v_d \leq \xi_d \\ 0 & ;v_d > \xi_d \end{cases} \qquad \forall \mathbf{p} \in \Omega \tag{3.47}$$

This has been very useful, if large local deformations occur, like raising the eye brows. Without this sub-weight the estimated motion may slightly follow local motions. A disadvantage of the sub-weight $\zeta_E^{\mathbf{p}}$ is to manually adjust the threshold ξ_d to each image sequence. Hence, if the estimation algorithm needs to be very flexible and accuracy is not the major focus, this sub-weight should not be used. However, if accuracy is important then this sub-weight is very useful, since the user can set a constraint of the maximum motion between two consecutive frames.

5. The previously proposed sub-weights are determined for each feature point without taking its neighborhood into account. The last sub-weight ζ_C considers the spatial relation between feature points and identifies outliers among them. The weight ζ_C is determined as follows: Each feature point $\mathbf{p}$ has a number of neighboring vertices $\mathbf{p}^* \in \Omega^* \subset \Omega$. Thus, we can define the neighboring luminance difference $\Delta I^{\mathbf{p}}$ by the weighted averages over its neighborhood

$$\Delta I^{\mathbf{p}} = \sum_{\mathbf{p}^* \in \Omega^*} w_{\mathbf{p},\mathbf{p}^*} \left[r'(\mathbf{p};\lambda) - r'(\mathbf{p}^*;\lambda) \right], \tag{3.48}$$

where the weights $w_{\mathbf{p},\mathbf{p}^*}$ are positive numbers that add up to one ($\sum_{\mathbf{p}^* \in \Omega^*} w_{\mathbf{p},\mathbf{p}^*} = 1$) for each feature point $\mathbf{p}$. The weight is defined as

$$w_{\mathbf{p},\mathbf{p}^*} = \frac{\phi_w(\mathbf{p},\mathbf{p}^*)}{\sum_{\mathbf{p}^{**} \in \Omega^*} \phi_w(\mathbf{p},\mathbf{p}^{**})} \tag{3.49}$$

with the positive function ϕ_w. We define this function as the inverse of the Euclidean distance of the neighboring vertices resulting in

$$\phi_w(\mathbf{p},\mathbf{p}^*) = \frac{1}{||\mathbf{p}-\mathbf{p}^*||_{L2}}. \tag{3.50}$$

Finally, we define the sub-weight as

$$\zeta_C^{\mathbf{p}} = \begin{cases} 1 & ;|\Delta I^{\mathbf{p}}| \leq \xi_c \\ 0 & ;|\Delta I^{\mathbf{p}}| > \xi_c \end{cases} \qquad \forall \mathbf{p} \in \Omega \tag{3.51}$$

with the threshold ξ_c. Thus, if the difference $|\Delta I^{\mathbf{p}}|$ is larger than the threshold ξ_c, then this feature point does not fit in its neighborhood. Therefore, this feature point is declared as an outlier. Note, that this weight is first applied after a few iterations when the sub-weight ζ_G is close to zero.

6. The sub-weight ζ_D as proposed by [144] takes the angle ϕ between the line of sight to the corresponding 3D feature point $\mathbf{P}$ and the surface normal of the face model's triangle on which $\mathbf{P}$ is located into account. The idea is that if the triangle is parallel to the camera target, then the texture information provided by the image has the highest resolution. Hence, these feature points obtain a large weight. On the other hand if the surface normal and camera target are almost perpendicular, then only little texture information is provided and these feature points receive a small weight. $\zeta_D^{\mathbf{p}}$ is defined as

$$\zeta_D^{\mathbf{p}} = \begin{cases} c_1 \left[1 - |\phi| \frac{2}{\pi}\right]^2 & ;|\phi| < \frac{\pi}{2} \\ 0 & ;|\phi| \geq \frac{\pi}{2} \end{cases} \qquad \forall \mathbf{p} \in \Omega, \tag{3.52}$$

where c_1 is a scalar. We do not take this weights into account, because the number of feature points located on triangles with a small angle ϕ (center of the mask) is much higher than with a large angle ϕ (edge of the mask). Hence, the sub-weight ζ_D is already implicitly applied by our system by selecting more feature points at triangles with a small angle ϕ.

Finally, for each feature point $\mathbf{p}$ its corresponding weight $\zeta^{\mathbf{p}}$ is calculated as

$$\zeta^{\mathbf{p}} = (\zeta_I^{\mathbf{p}} + \zeta_G^{\mathbf{p}}) \cdot \zeta_V^{\mathbf{p}} \cdot \zeta_E^{\mathbf{p}} \cdot \zeta_C^{\mathbf{p}} \qquad \forall \mathbf{p} \in \Omega. \tag{3.53}$$

3.6.2 Automatic Update of Texture Information of Feature Points

The motion estimation algorithm described so far minimizes the luminance difference of the initially selected feature points between the current $\mathbf{I}(t)$ and the reference frame $\hat{\mathbf{I}}(t=0)$. Therefore, feature points are initially selected in the reference frame and projected onto the face model. Furthermore, their luminance values are set. The described procedure can be also understood as texturing the face model.

In order to estimate out-of-plane motion parameters, a perfect face texture is desirable. Consequently, an infinite number of images taken with different head poses should be provided. Since we try to illuminate the recorded person with ambient light (Section 3.6), translation and in-plane rotation do almost not change the texture of an image. Thus, the pose of the head, which is important for texturing, is only characterized by the two out-of-plane rotations ω_x and ω_y (Figure 3.10). These angles describe the pose of the head versus the camera target, which does not move in our set-up and can be understood as spherical coordinates marking a position on the unit sphere

$$\begin{aligned} H_x &= sin(\omega_x)cos(\omega_y) \\ H_y &= sin(\omega_x)sin(\omega_y) \\ H_z &= cos(\omega_x). \end{aligned} \tag{3.54}$$

The position $\mathbf{H}_t = [H_x, H_y, H_z]^T$ on the unit sphere characterizes the head pose of a particular image $\mathbf{I}(t)$. All images, which provide texture information for the motion estimation, are denoted as reference frame $\hat{\mathbf{H}}_j$ with index j. Its corresponding texture information is denoted as $\hat{\mathbf{I}}(j)$. Consequently, the new cost function $C_H(\lambda)$ can be determined by taking Equation (3.22) and the weight ζ of Equation (3.53) into account resulting in

$$C_H(\lambda) = \sum_{\mathbf{p} \in \Omega} \zeta^{\mathbf{p}} \left[\mathbf{I}(\mathbf{F}(\mathbf{p}, \lambda), t) - \hat{\mathbf{I}}(\mathbf{p}, j) \right]^2 \tag{3.55}$$

with the j reference frame. By default the initial image is the first reference frame $\hat{\mathbf{I}}(0)$. In Figure 3.13 the position of five reference frames are exemplarily shown.

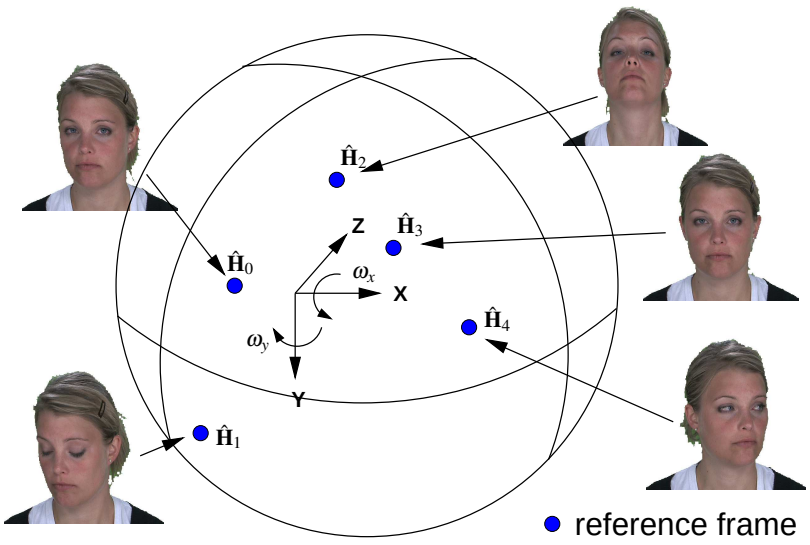

Figure 3.13: The positions and corresponding texture information of five reference frames ($\hat{\mathbf{H}}_0$ to $\hat{\mathbf{H}}_4$) on the unit sphere are presented.

Extending the Motion Estimation Algorithm with a Texture Update

In Figure 3.14 a chart diagram of the extended motion estimation algorithm is shown. For the sake of simplicity, the motion estimation algorithm is displayed as a single block. During the initialization the first frame $\mathbf{I}(t_0)$ is stored as initial reference frame denoted as $\hat{\mathbf{I}}(0)$ in the database, so that the texture information of the selected feature points is available. The image sequence and the face model are the input to the main algorithm. Frame t is loaded and the new head pose estimated by using the texture information of the feature points by the current reference frame $\hat{\mathbf{I}}(j)$. After the face model is moved to the new location the condition for updating the current reference frame is evaluated. If the condition is satisfied, then either a former reference frame $\hat{\mathbf{I}}(l)$ is selected or a new reference frame $\hat{\mathbf{I}}(j+1)$ is generated and stored in the database. Afterwards the texture information of the feature points is updated. If the condition of updating is not satisfied, the head pose in the next frame is calculated by using the same reference frame $\hat{\mathbf{I}}(j)$.

Conditions for Updating the Reference Frame

After the motion parameters are estimated for frame t, its position $\mathbf{H}_t$ on the unit sphere is calculated by using Equation (3.54). Two conditions for creating a new reference frame need to be satisfied.

The first condition is formulated as follows: If the Euclidian distance between the position of the current frame $\mathbf{H}_t$ and current reference frame $\hat{\mathbf{H}}_j$ on the unit sphere is

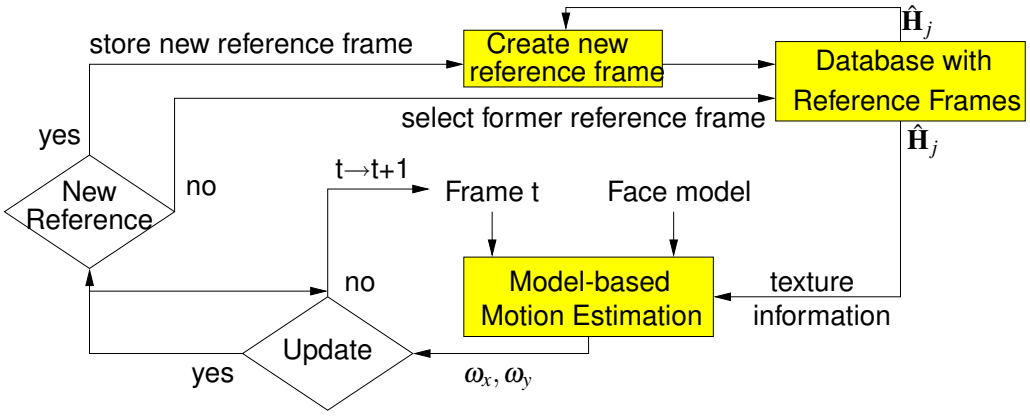

Figure 3.14: Chart diagram of the extended model-based motion estimation algorithm with an automatic update of reference frames providing new texture information.

larger than a threshold c_3, then the first condition is satisfied. This can be formulated as

$$\| \mathbf{H}_t - \hat{\mathbf{H}}_j \|_{L2} > c_3. \tag{3.56}$$

This condition verifies, that a new reference frame is only generated due to out-of-plane rotations.

The second condition verifies that only frames with a small cost function $\bar{C}$ can be declared as reference frames. For this the following condition needs to be satisfied

$$\bar{C}_H = \frac{\sum_{\mathbf{p}\in\Omega} \zeta^{\mathbf{p}} \left[\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},j)\right]^2}{\sum_{\mathbf{p}\in\Omega} \zeta^{\mathbf{p}}} < c_4 \tag{3.57}$$

with the threshold c_4. Hence, only if the normalized cost function $\bar{C}_H$ is less than a threshold c_4, the current frame can become the new reference frame $\hat{\mathbf{I}}(j+1)$. Additionally, the total sum of the weights $\sum_{\mathbf{p}\in\Omega} \zeta^{\mathbf{p}}$ must be larger than a threshold. This verifies that the reference frame provides a large number of feature points, which can be reliably tracked.

After the estimation of the head pose the system proves whether a former reference frame is selected for the further calculations. The distance r_l between two consecutive reference frames on the unit sphere is determined as

$$r_l = \| \hat{\mathbf{H}}_{l+1} - \hat{\mathbf{H}}_l \|_{L2} \qquad \forall l \in [0, j-1] \tag{3.58}$$

If a former reference frame is selected is determined by calculating the distance of the current position of a frame on the unit sphere $\mathbf{H}_t$ and all former reference frames

$$\| \mathbf{H}_t - \hat{\mathbf{H}}_l \|_{L2} \leq r_l \qquad \forall l \in [0, j-1]. \tag{3.59}$$

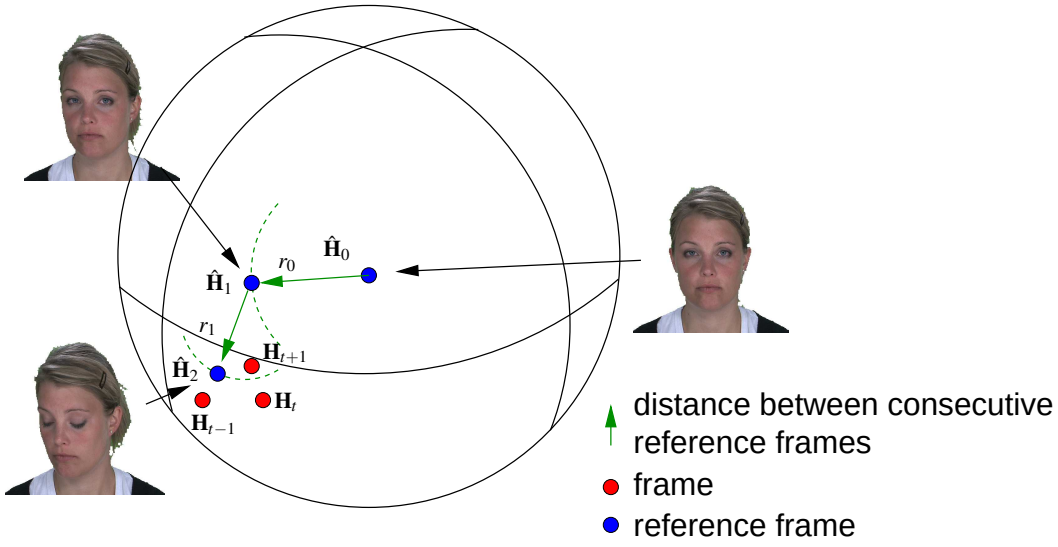

Figure 3.15: The positions of three reference frames ($\hat{\mathbf{H}}_0$, $\hat{\mathbf{H}}_1$, $\hat{\mathbf{H}}_2$) with texture information on the unit sphere are stored in the database. The distance r_l between two consecutive reference frames is also displayed. In order to estimate the motion parameters from frame $\mathbf{H}_{t-1}$ to $\mathbf{H}_t$, the reference frame $\hat{\mathbf{I}}(2)$ is used. Afterwards the condition for updating texture information is checked but not satisfied. Hence, the motion parameters of the next frame $\mathbf{H}_{t+1}$ are determined with the same reference frame $\hat{\mathbf{I}}(2)$. The new head position of frame $\mathbf{H}_{t+1}$ on the unit sphere is within the radius r_1. Thus, the condition for selecting a previous reference frame is satisfied and the texture information of the feature points are updated with reference frame $\hat{\mathbf{I}}(1)$.

If this condition is satisfied, then the reference frame with the lowest index l is selected as the new reference frame $\hat{\mathbf{H}}_l$ for estimating the hereafter following motion. In this way former reference frames are used again to provide texture information for motion estimation, and thus preventing a drift. In Figure 3.15 an example is given.

Updating Texture Information

If the condition of Equation (3.59) is satisfied, the head rotates back into a former position. Then the former reference frame with the lowest index satisfying Equation (3.59) is selected and the luminance value of all feature points are updated by this reference frame.

Otherwise if the first two conditions are satisfied, then a new reference frame is created, which will be used for the consecutive motion estimation. The most simple approach is to update all feature points. The disadvantage of this approach is obvious, since the valuable information of the current reference frame are simply deleted. Hence, we only update

certain feature points and try to integrate the information of the current reference frame in the new one. If the following two conditions are satisfied, then a feature point is updated. Firstly, if the angle $|\phi|$ as defined in Equation (3.52) decreases in the current with respect to the previous reference frame . Secondly, the residual error needs to be less than

$$|\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},j)| \leq c_5\sigma_I \tag{3.60}$$

with the scalar $c_5 \in [2.0, 4.5]$ and the standard deviation σ_I of Equation (3.43). This condition prevents that outliers are used for the further calculations. Furthermore, additional features are added to triangles, if the number of visible feature points in a certain face area is too low.

3.7 Reference Method

In the work of Xiao et al. [144] all components of the motion estimation algorithm of Section 3.5 are not described. The interpolation of the luminance of feature points as well as the spatial gradient estimation are left out in their work. Whereas they only use a simple cylindrical model as object model, they use the same parametric motion model and a hierarchical implementation of the motion estimation. For a fair comparison, both algorithms use the same components as introduced in Section 3.5. Their work mainly deals with improving the robustness of a gradient-based motion estimation. They multiply the sub-weight ζ_D with the sum of the sub-weights ζ_I and ζ_G as described in Section 3.6.1. Since they use a simple cylindrical head model, visibility is already checked by the sub-weight ζ_D. However, since we use a detailed face model, feature points may be occluded. Therefore, the sub-weight ζ_V is added. Hence, for each feature point $\mathbf{p}$ its corresponding weight $\zeta^{\mathbf{p}}$ is calculated as

$$\zeta^{\mathbf{p}} = (\zeta_I^{\mathbf{p}} + \zeta_G^{\mathbf{p}}) \cdot \zeta_D^{\mathbf{p}} \cdot \zeta_V^{\mathbf{p}} \qquad \forall \mathbf{p} \in \Omega. \tag{3.61}$$

In order to achieve long-term robustness, they propose a method to dynamicly update templates and re-registration. After the head pose is estimated in the current frame $\mathbf{I}(t)$, it automatically becomes the next reference frame $\hat{\mathbf{I}}(j+1)$, which is used to calculate the next head pose. A feature point $\mathbf{p}$ of $\hat{\mathbf{I}}(j+1)$ will be weighted as

$$\zeta^{\mathbf{p}} = \begin{cases} (\zeta_I^{\mathbf{p}} + \zeta_G^{\mathbf{p}}) \cdot \zeta_D^{\mathbf{p}} \cdot \zeta_V^{\mathbf{p}} & ; |\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},j)| \leq c_2\sigma_I \\ 0 & ; |\mathbf{I}(\mathbf{F}(\mathbf{p},\lambda),t) - \hat{\mathbf{I}}(\mathbf{p},j)| > c_2\sigma_I \end{cases} \tag{3.62}$$

where $c_2 \in [2.5, 3.5]$ is a scalar and σ_I calculated with Equation (3.43). In order to prevent error accumulation certain frames are stored as reference frames. Under two conditions the currently created reference frame is discarded. Whenever the current head pose is close to a stored reference frame, this reference frame is used for the further estimation. Furthermore, if the the cost function $\bar{C}_H$ of Equation (3.57) exceeds a pre-set threshold, the closest available reference frame is selected for future estimation.

Moreover, they suggest to a use a regularization technique before inverting the ill-conditioned matrix $\left[\sum_{\mathbf{p}\in\Omega}\zeta^{\mathbf{p}}[\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})]^T[\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})]\right]$ of Equation (3.41). The regularization parameter λ_r is incorporated in Equation (3.25) resulting in

$$\begin{aligned}\lambda_i = -&\left[\sum_{\mathbf{p}\in\Omega}\zeta^{\mathbf{p}}\left([\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})]^T[\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})]+\lambda_r\mathbf{F}_{\lambda}^T(\mathbf{p},\mathbf{0})\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})\right)\right]^{-1}\\ &\sum_{\mathbf{p}\in\Omega}\zeta^{\mathbf{p}}[\mathbf{I}(\mathbf{p},t)-\mathbf{I}(\mathbf{p},t_0)][\mathbf{I}_{\mathbf{p}}(\mathbf{p},t)\mathbf{F}_{\lambda}(\mathbf{p},\mathbf{0})]^T.\end{aligned} \tag{3.63}$$

Our analysis [100] revealed, however, that regularization strongly depends on the parameter λ_r. An appropriate value for λ_r may slightly improve the head pose accuracy in one sequence, but decrease the accuracy in another sequence. Hence, we do not apply regularization.

4 Analysis of Recorded Speech and Video

In this section, we conduct two experiments in which we record human subjects in a two-way conversation (Section 4.1) and describe the algorithms to analyze the recorded data. The goal is to prepare the recorded data, so that in Section 5 statistical dependencies between eye blinks, gaze shifts and spoken language can be determined.

The block diagram in Figure 4.1 depicts the analysis steps of the recorded data. Note that only the data recorded in the first experiment needs to be processed (Figure 4.1). Since a large amount of data has to be analyzed, the designed algorithms must be highly automatic. First the recorded video is manually divided into segments in which the recorded human subject is listening and talking, since the gaze behavior varies [4]. All frames of the 'listening segments' are only labeled with their gaze and blink patterns. The recorded video is manually labeled with its corresponding gaze state. In Section 4.2 the algorithm to automatically label each frame with the information whether an eye blink is executed or not is explained. All frames of the 'talking segments' are additionally labeled with audio information (Section 4.3).

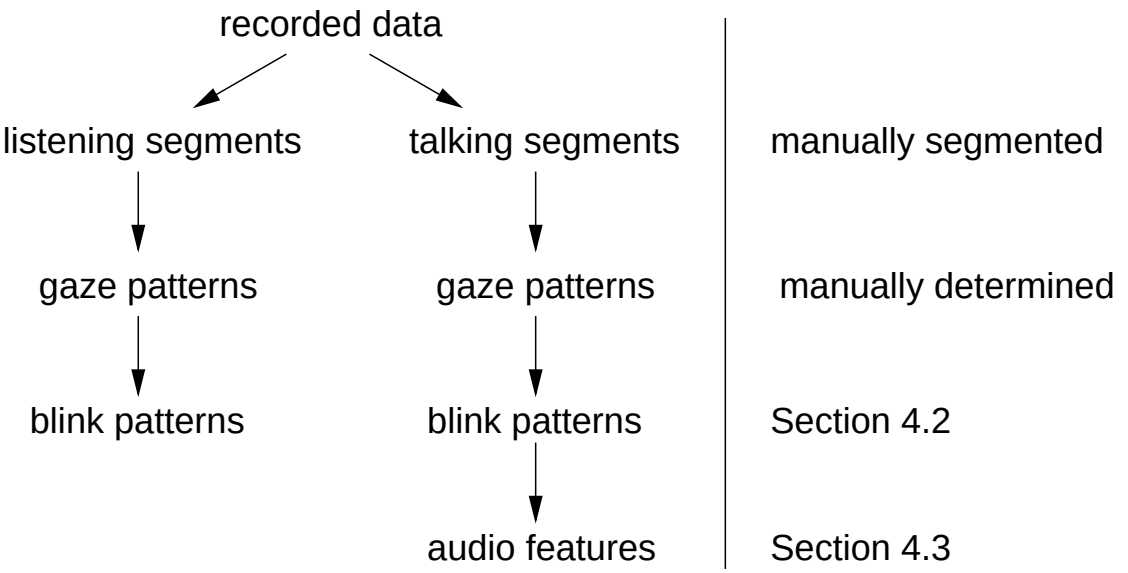

Figure 4.1: Block diagram of analyzing the recorded video sequences of the first experiment.

4.1 Recording Human Subjects

We distinguish between two gaze states mutual gaze (MG) and gaze away (GA). We define MG as the state, in which the direction of gaze is located within the facial area of the interlocutor consisting of mouth and eye area. If the gaze is not in this defined area, then the system is in GA. If the speaker switches from MG to GA, a gaze shift (GS) is performed. In order to analyze the statistical dependencies between gaze and blink patterns as well as spoken language we conduct the experimental set-up in Section 4.1.1.

The dynamics of saccadic eye movements cannot be analyzed in the first set-up, since ordinary cameras are used to record human subjects. However, the spatial sampling frequency of these cameras is too low, if recording head and shoulder. Therefore, in the second experimental set-up an eye tracker is used (Section 4.1.2), which measures the point of regard (POR). Under POR we understand the direction of gaze, where a person is looking. A significant drawback of eye trackers is either the necessity of wearing an eye tracking device or the restriction of a stationary head pose. Hence, we only use eye trackers to determine characteristics, which cannot be analyzed by the first experiment, e.g. the characteristics of saccades.

4.1.1 Set-up 1: Camera Recording

In the first set-up, we record in two sessions a conversation of two persons who are interviewing each other and discussing current-affairs. In each session, which lasted for 30 minutes, the same moderator and a different human subject participated. They are sitting in front of a table and facing each other (Figure 4.2). The camera, which is recording in PAL progressive mode with 25 frames per second, is located next to the head of the moderator. A microphone is positioned on the table, so that audio and video can be synchronously recorded.

Both human subjects are informed, that not eye but mouth movements and facial expressions are investigated in this study in order to avoid potential change of eye movement behavior. In each session the beginning of the conversation is not recorded in order to let the human subject get used to the set-up. For instance, the camera located next to the head may distract. After a while (5-7 minutes), we believe that the subjects acclimate and their subconscious controls the eye movements. While recording took place in a lab environment, subjects seemed to behave naturally.

4.1.2 Set-up 2: Eye Tracker

First, the different types of eye trackers are briefly explained and afterwards the experimental set-up.

Figure 4.2: During a conversation human subjects are recorded in order to analyze the characteristical eye movement behavior.

Eye Tracking Techniques

There are four broad groups of eye movement measurement techniques: electro-oculography, scleral contact lens/search coil, photo- or video oculography and video-based combined pupil and corneal reflection [41].

Electro-oculography measures the skin's electric potential differences of electrodes placed around the eye (Figure 4.3) and has the advantage of a large measurement range of $\pm 70°$ [147]. The recorded potentials are in the range of 15 to 200 μV. This method measures the eye globe rotation relative to head position.

Scleral contact lens or search coil involves a mechanical or optical reference object

Figure 4.3: Example of electro-oculography eye movement measurement [48].

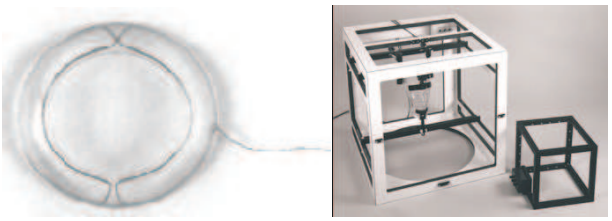

Figure 4.4: Example of search coil embedded in contact lens and electromagnetic field frames for search coil eye movement measurement [121].

Figure 4.5: Example of infra-red limbus apparatus [121].

mounted on a contact lens, which is then worn directly on the eye (Figure 4.4). The principal method employs a wire coil, which is then measured moving through an electromagnetic field. Although this method achieves an accuracy of 5 to 10 arcseconds over a limited range of about 5° [147], it is also the most intrusive method. For instance, the insertion of the lens requires care and experience.

Photo- and video oculography group a wide variety of recording eye movements involving the measurement of distinguishable eye features in order to measure the rotation of the eye globe. In this group measurement techniques may or may not be made automatically and may involve time-consuming visual inspection of a human operator. A possible hardware realization is illustrated in Figure 4.5.

While the methods above are in general suitable for eye globe rotation measurements, the last category consisting of video-based methods measure the POR, which involves the measurement of distinguishable features of the eyes (Figure 4.6). As features usually the pupil center and the corneal reflection are measured [33]. In order to detect the corneal reflection the eye is illuminated by a low power, infrared light emitting diode (LED).

Set-up

In our set-up we use a table-mounted eye tracker of the last category [84, 27, 25], which enables us to measure the POR. In this way we are able to analyze the characteristics of eye movements in more detail. The eye tracker is used in a video conference system, in

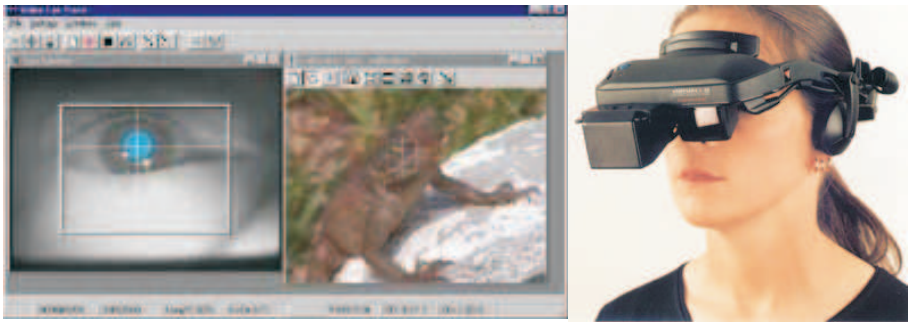

Figure 4.6: Example of head-mounted video-based head tracker [71].

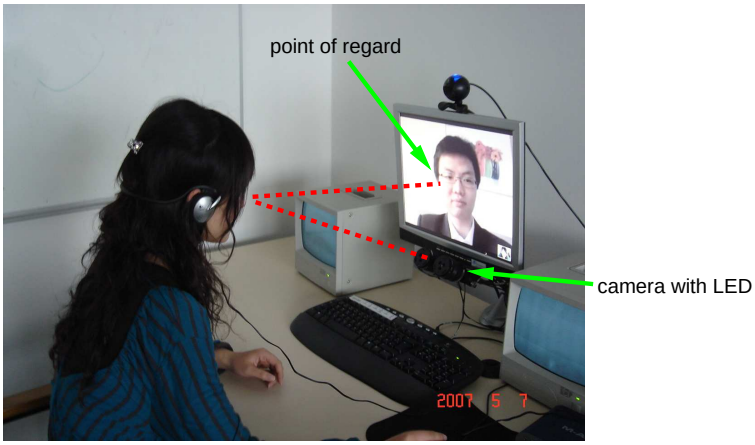

Figure 4.7: The participant's eye movements are measured during a video conference with an eye tracker.

which the communication takes place over a camera and microphone (Figure 4.7). This set-up is most similar with respect to possible applications of talking-heads, e.g. in a dialog system. The conversational settings are the same as in in the first set-up. Before the experiment the computer screen is adjusted to the participant's height by aligning the middle of the computer screen with the primary eye position. The distance between computer screen and eyes is manually measured and approximately 450 mm. After an initial calibration the table-mounted eye tracker with a sample frequency of 60 Hz is able to measure the POR on a computer screen. The head pose, however, may only vary a few centimeters, because only in this distance the focus range compensation [26] prevents the accumulation of measurement errors. If large head rotations are performed, then the tracker is not able to measure the POR anymore. The eye tracker as specified in [84] achieves an average bias error of 38 mm of the spatial gaze point over the monitor screen. Bias errors result from inaccuracies in the measurement of head range, asymmetries of the pupil opening about the eye's optic axis, and astigmatism, where vision is blurred by an irregular shaped cornea. Furthermore, image noise induces an error of 15 mm, if the head has a distance of 510 mm to the monitor screen [84].

The eye tracker processes the measured POR and automatically determines fixations and saccades [3, 117]. Therefore, this type of eye tracker gives the opportunity to determine the direction and magnitude of saccades. The PORs before $\mathbf{e}_1 = [e_{x_1}, e_{y_1}]^T$ and after $\mathbf{e}_2 = [e_{x_2}, e_{y_2}]^T$ a saccade is performed are measured in computer screen coordinates by the eye tracker (Figure 4.8). The size of a computer screen pixel is defined by the scaling factor s_e, which converts the computer screen coordinates to mm. The direction of the saccade can be directly extracted from $\mathbf{e}_1$ and $\mathbf{e}_2$. In order to determine the magnitude of the saccade A^s, the 2D point $\mathbf{e}$ needs to be extended to 3D by taking the distance between eye and computer screen into account. Since the distance between eyes and computer screen is approximately 450 mm, the corresponding 3D points on the computer screen are $\mathbf{E}_1 = [e_{x_1}, e_{y_1}, \frac{450}{s_e}]^T$ and $\mathbf{E}_2 = [e_{x_2}, e_{y_2}, \frac{450}{s_e}]^T$. Then the magnitude of the saccade A^s is equal to

$$A^s = acos\left[\frac{\mathbf{E}_1 \cdot \mathbf{E}_2}{|\mathbf{E}_1||\mathbf{E}_2|}\right]. \tag{4.1}$$

The largest rotation angle between primary eye position and the corner of the computer screen is approximately 25.6°. Note, that the magnitude A^s in Equation (4.1) is not precisely calculated, since the distance between eye and computer screen is only estimated and varies because of small head pose variations of the participant. For instance, if the distance is only 400 mm instead of 450 mm as in the example above, then the magnitude increases to 28.3°. However, the small variations do not severely contaminate the measured magnitude of saccades. First of all, measurements are approximated by a well-known probability distribution, so that small variations of the head pose are equalized. Furthermore, our analysis reveals that 86% of the magnitude A^s of the saccades are less than 15° as stated in [6].

Summing up, this type of eye tracker gives the opportunity to analyze gaze shifts of

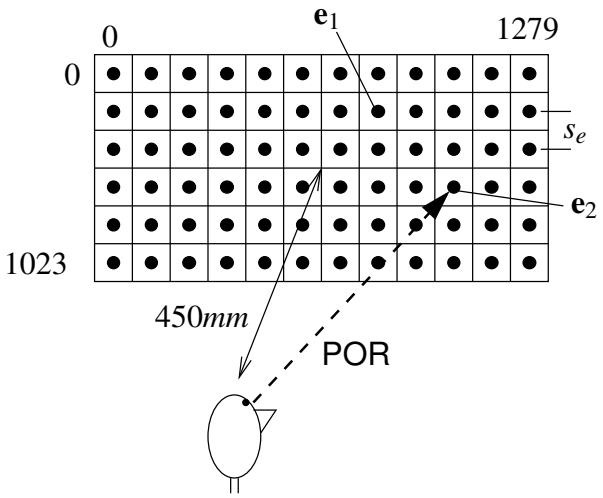

Figure 4.8: The eye tracker automatically determines the POR on the computer screen (dashed arrow). The POR before $\mathbf{e}_1$ and after $\mathbf{e}_2$ a saccade is performed are exemplarily presented. The screen resolution is 1280×1024 and the distance between head and screen is approximately 450 mm (solid arrow).

human subjects in a conversation in more detail. For this, we can define regions of interest (ROIs), e.g. mouth area, and measure the probability that the POR is located within this ROI.

4.2 Eye Blink Detection

Since a large amount of data needs to be processed, an algorithm to automatically detect eye blinks in an image $\mathbf{I}(t)$ is designed. For this, an algorithm is developed, which determines whether the eyes are closed or opened (Figure 4.9).

Initially the head pose is estimated (Section 3.5) and the image sequence normalized. Normalizing means to compensate any head pose variations. Hence, changes of the eye area are only due to local iris and eyelid movements. In order to decide if the eyes are opened or closed, a template matching is executed. For this, a number of frames of the recorded video sequence are normalized. Afterwards two templates $\mathbf{J}_1$ and $\mathbf{J}_2$ of size (w_J, h_J), one with open and one with closed eyes, are extracted from the normalized frames. Each template $\mathbf{J}_i$ consists of the left and right eye corner of one eye. The middle of the eye is excluded, since not only the eyelid but also the iris moves. From both templates $\mathbf{J}_i$ the average pixel value of each template $\bar{J}_i$ $\left[\bar{J}_i = \frac{1}{w_J \cdot h_J} \sum_{i_x=0}^{w_J} \sum_{i_y=0}^{h_J} \mathbf{J}_i(i_x, i_y)\right]$ is subtracted resulting in $\bar{\mathbf{J}}_i$. In order to calculate the correlation coefficient $\rho_i(t)$ the coordinate systems of the image $\mathbf{I}(t)$ and template $\bar{\mathbf{J}}_i$ are aligned by a shift $\mathbf{j} = [j_x, j_y]^T$.

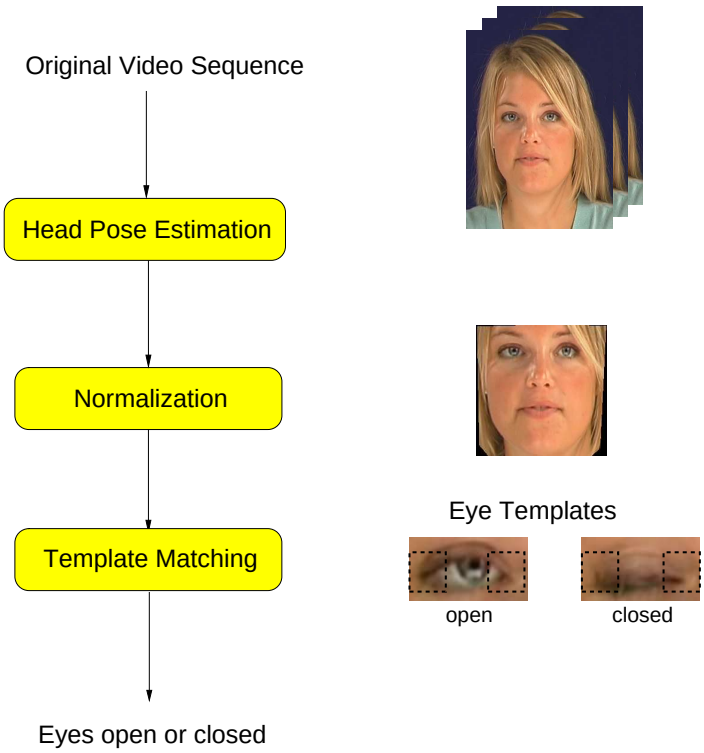

Figure 4.9: Block diagram of automatically detecting eye blinks. Before normalizing the head pose is estimated. Afterwards a template matching based on two templates indicates whether the eyes are opened or closed. As templates both eye corners are used (indicated by the black dashed rectangle).

The correlation coefficient $\rho_i^J(t)$ is calculated for each image $\mathbf{I}(t)$ and template $\bar{\mathbf{J}}_i$

$$\rho_i^J(t) = \left| \frac{\sum_{i_x=0}^{w_J} \sum_{i_y=0}^{h_J} \bar{\mathbf{J}}_i(i_x, i_y) \bar{\mathbf{I}}(i_x + j_x, i_y + j_y, t)}{\sqrt{\sum_{i_x=0}^{w_J} \sum_{i_y=0}^{h_J} \bar{\mathbf{J}}_i^2(i_x, i_y) \bar{\mathbf{I}}^2(i_x + j_x, i_y + j_y, t)}} \right|, \tag{4.2}$$

where $\bar{\mathbf{I}} = \mathbf{I} - \bar{I}$ and $\bar{I} = \frac{1}{w_J \cdot h_J} \sum_{i_x=0}^{w_J} \sum_{i_y=0}^{h_J} \mathbf{I}(i_x + j_x, i_y + j_y, t)$. The template $\bar{\mathbf{J}}_i$ with the higher correlation coefficient indicates whether the eyes are opened or closed. The designed algorithm is very robust and achieves an accuracy of approximately 99% [145].

4.3 Audio Analysis

In order to determine statistical dependencies between eye movements, blinks and spoken output, the recorded video needs to be phonetically labeled (Section 4.3.1). However,

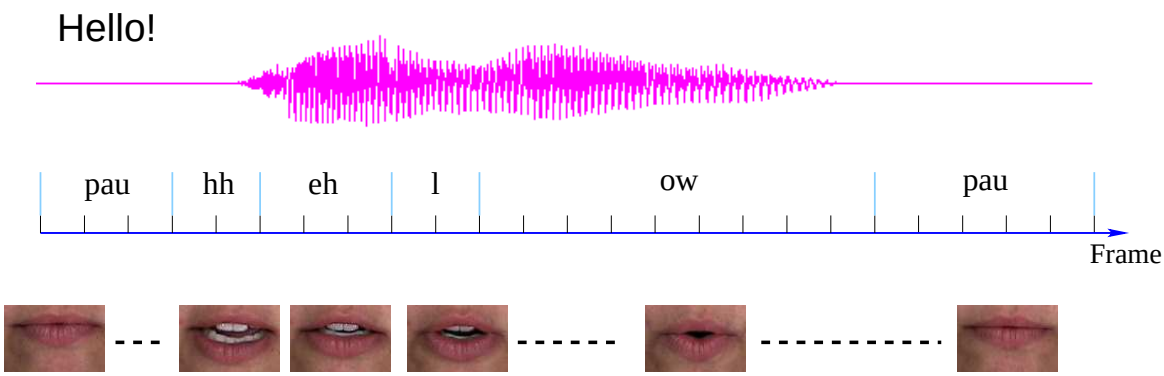

Figure 4.10: Phoneme labeling assigns a phoneme to each frame of the video.

spoken language conveys not only words, but a wide range of other information about timing, intonation etc. that is often collectively called spoken prosody. In more detail, the most important characteristics of prosody are pause, phoneme duration (speech rate), pitch variations and loudness [69]. Linguists refer to prosodic events as long-time speech features in contrast to phonemes. Prosodic features have often a strong correlation to human movements, e.g. head movements [82]. Hence, eye movements and blinks may also show statistical dependencies with prosodic events. While pauses are already detected by the phonetic labeling, the extraction of further prosodic features is explained in Section 4.3.2 and 4.3.3.

4.3.1 Phoneme Labeling

We perform audio alignment on the recorded sequences using the speech recognition software HTK [146], which is publicly available. First the speech recognition software is trained by providing a training corpus of recorded sentences of the speaker. Afterwards given an audio sequence and an associated text transcript of the speech being uttered, HTK uses forced Viterbi search to find the optimal phoneme durations for the given audio sequences. In Figure 4.10 each frame of the video is labeled with its phonetic context of the spoken output.

4.3.2 Rate of Speech

The rate of speech is used to classify the words of the spoken output as fast, slow or medium. In general, there are, alignment-based and signal-based measures to calculate the rate of speech [98]. The latter can be calculated before speech recognition and used as additional information to improve the speech recognition. However, these measures are not as reliable as alignment-based methods, which are based on considering the duration

of the aligned phonemes or words. We use an alignment-based method as proposed in [148] in order to classify the words of the spoken output.

Each word ι is assigned its measured duration υ, which is known from the phoneme labeling (Section 4.3.1), and classification ς as slow, medium or fast. In order to classify a word ι with the duration υ we define a function

$$\varsigma = \begin{cases} \text{fast} & ; \quad 0 \leq \upsilon < D_1(\iota) \\ \text{medium} & ; \quad D_1(\iota) \leq \upsilon < D_2(\iota) \\ \text{slow} & ; \quad D_2(\iota) \leq \upsilon < \infty \end{cases} \tag{4.3}$$

with the three time intervals $[0, D_1(\iota))$, $[D_1(\iota), D_2(\iota))$ and $[D_2(\iota), \infty)$. The interval endpoints $D_1(\iota)$ and $D_2(\iota)$ depend on the current word ι.

In order to determine the interval endpoints $D_1(\iota)$ and $D_2(\iota)$ for each word ι the probability distribution of the word duration $p_\iota(t_r)$ is calculated. In practice, it is difficult to directly estimate $p_\iota(t_r)$ for each word because of the spareness of the recorded data. Instead of words, we can use phonemes under the assumption, that in a word the duration distribution of its component units, such as phonemes, are independent of each other [148]. For each phoneme a relative frequency distribution of its duration is measured from the recorded audio. Since a large amount of data is available, we assume that this relative frequency distribution is equal to its probability distribution. Then the probability $p_\iota(t_r)$ of a word ι is calculated by the convolution of its phonetic probability distributions (Figure 4.11). For any subset i defining a time interval $[t_1^i, t_2^i]$, we determine its corresponding probability $R_\iota(i)$ as follows

$$R_\iota(i) = \sum_{t_r = t_1^i}^{t_2^i} p_\iota(t_r) \tag{4.4}$$

giving the probability that the duration υ of the word ι is located within the time interval $[t_1^i, t_2^i]$. For instance, the interval classified as slow ($i = 3$) ranges from $t_1^3 = D_2(\iota)$ to $t_2^3 \to \infty$. If we set each $R_\iota(i)$ to a certain threshold ξ_R, e.g. $\xi_R = 0.2$, then the interval endpoints $D_1(\iota)$ and $D_2(\iota)$ can be calculated. While the threshold ξ_R remains constant for all words, the interval endpoints vary for each word.

4.3.3 Emphasis Detection

Speech prominence detection is useful in many spoken language applications such as improving automatic speech recognition. First the terms stress, accent and prominence, which are all related to speech prominence detection, are clarified.

Stress generally refers to an idealized location in an English word that is a potential site for phonetic prominence effects, such as extruded pitch. This information comes from a standard lexicon. For instance, the second syllable of the word "em-**ploy**-er" is said to have an abstract property of lexical stress.

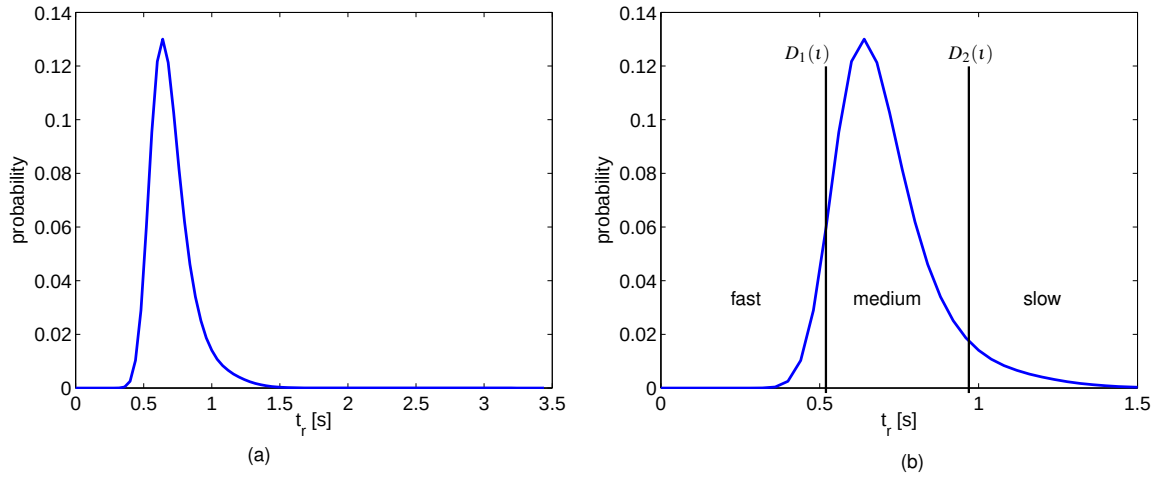

Figure 4.11: (a) The probability distribution of duration $p_\iota(t_r)$ of the word ι='especially' is presented. (b) The corresponding intervals slow, medium and fast with the interval endpoints $D_1(\iota)$ and $D_2(\iota)$ are added.

Accent, which is realized via extruded pitch, is the signaling of semantic salience by phonetic means. Lexically stressed syllables often receive a prosodic accent. However, in an actual utterance, the location of stress in a word may be overridden, e.g. 'I didn't say emlpoy**er**, but employ**ee**'. Pitch accents can be labeled according to the well-known ToBI (Tones and Break Indices) prosody classification scheme [13].

There are different definitions of the term prominence, e.g. in [125] prominence and accent are interchangeable. We use Terken's definition [130], who defines prominence as 'word or syllables that are perceived as standing out from the environment', because glances are used by speakers to accompany particularly emphasized words or phrases [4].

The algorithm to detect emphasized words in utterances is based on the fundamental frequency (F0). Hence, first the term F0, the extraction of F0 and finally the algorithm detecting prominence are explained.

Fundamental Frequency (F0)

The human voice is produced in the larynx, which is a part of the throat. There are two small pieces of tissue that stretch across the larynx with a small opening between them, which are called vocal cords. The most fundamental distinction between sound types in speech is the voice and voiceless distinction. What in the speech production mechanism creates this fundamental distinction? The phoneme is considered voiced, when the vocal cords vibrate during phoneme articulation. Throughout their durations vowels are voiced. The vocal folds vibrate at slower or faster rates, from as low as 60 Hz for a large man, to

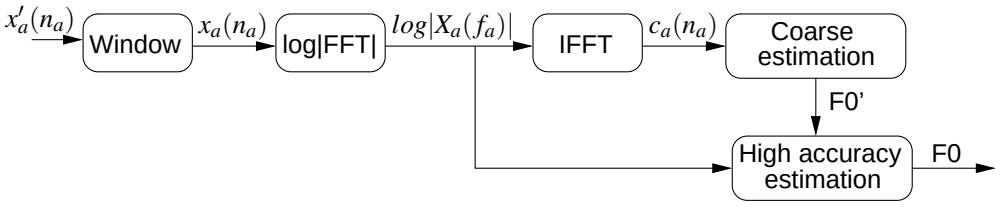

Figure 4.12: Block diagram of the fundamental frequency estimator [101].

as high as 300 Hz or higher for a small woman or child [69]. The rate of cycling (opening and closing) of the vocal cords in the larynx during phonation of voiced sounds is called F0 [69].

Our F0 estimator uses [110, 101], which estimates F0 with the cepstrum. The name cepstrum comes from reversing the first four letters in the word 'spectrum', indicating a modified spectrum. The independent variable related to the cepstrum transform has been called 'quefrency', and since this variable is very closely related to time it is acceptable to refer to this variable as time.

The basic block diagram of the estimator is presented in Figure 4.12. Before F0 can be determined, the estimator converts the audio signal $x'_a(n_a)$ into a cepstrum $c_a(n_a)$. Since speech is non-stationary, a short-term analysis is performed. For this, the audio signal $x'_a(n_a)$ is broken into a number of small segments with a duration of 20 ms by a Hann window [62]. The signal $x_a(n_a)$ is converted to the frequency domain $X_a(f_a)$. Afterwards the log magnitude of the spectrum of $X_a(f_a)$ is calculated resulting in $log(|X_a(f_a)|)$. In order to calculate the cepstrum $c_a(n_a)$ the signal $log(|X_a(f_a)|)$ is converted by the inverse FFT into the quefrequency domain (similar to time domain). F0 is estimated in two steps: First a coarse estimation of the fundamental frequency F0' is based on searching the cepstrum's peak location on a coarse grid [112]. Afterwards the resolution in frequency domain of the spectrum $log(|X_a(f_a)|)$ is in the surrounding of F0' increased. Around F0' possible candidates for F0 are selected and the energy of each candidate is calculated. For this the energy of multiples of the candidate's frequency is added. The candidate with the highest energy is F0. In Figure 4.13 an example of the speech wave and corresponding estimated fundamental frequency are presented.

Detection of Emphasized Words

A large number of proposed algorithms try to automatically detect accents (e.g. [128, 127]), whereas we try to automatically determine emphasized words. We use the algorithm proposed in [5, 80] to automatically detect areas of high pitch activity and variability that indicate an emphasis. For the sake of simplicity and because of our satisfying results, we only use two from the proposed metrics in [5, 80].

First we convert words into their syllables denoted as sl. Afterwards the amount of activity and variations of F0 $v_{F0}(sl)$ for each syllable sl is determined. $v_{F0}(sl)$ is com-

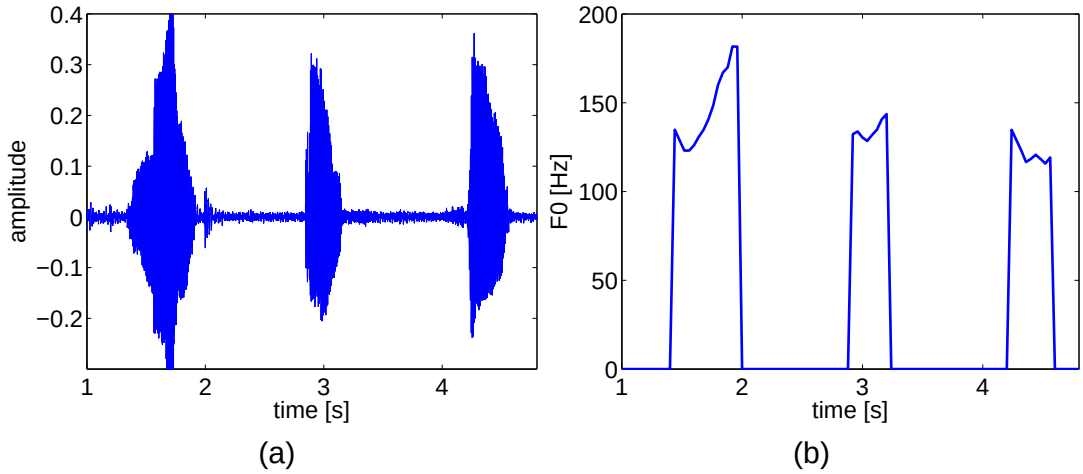

Figure 4.13: (a) Speech wave of man uttering ’1’, ’2’, ’3’. (b) Estimation of F0 of the speech wave in (a).

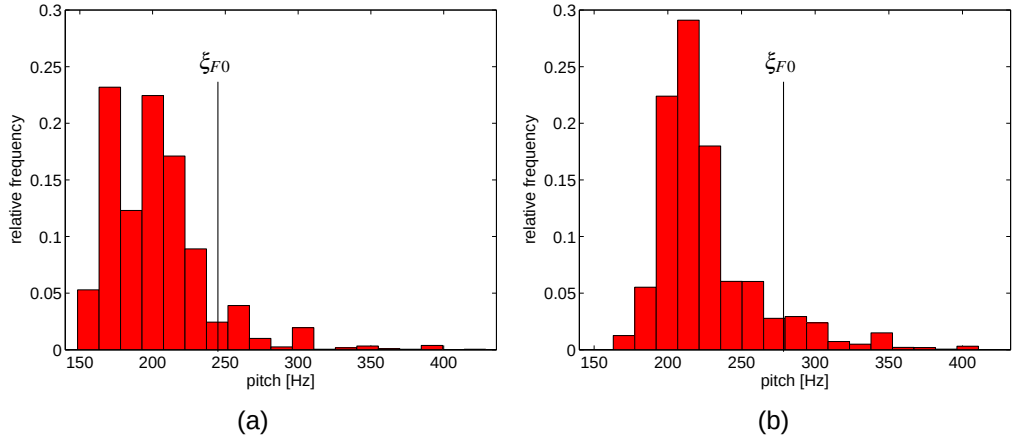

Figure 4.14: Relative frequency distributions of pitch variation of male (a) and female (b) speaker. The threshold ξ_{F0} is chosen to include the top 10% of F0, which strongly depends on the individual.

posed of two characteristics: the standard deviation σ_{F0} and number of frames n_{F0} above a threshold ξ_{F0}.

Since the fundamental frequency varies between humans, ξ_{F0} needs to be adapted to the speaker. For this, ξ_{F0} is selected to contain the top 10% of F0 values (Figure 4.14). The variation and activity $v_{F0}(sl)$ for a syllable sl is calculated as

$$v_{F0}(sl) = w_1 \cdot \sigma_{F0} + \frac{w_2 \cdot n_{F0}}{t_{sl}}, \tag{4.5}$$

with the the duration t_{sl} of sl and weights w_1 and w_2.

A large value of v_{F0} either indicates a high variation, activity or both of F0. If v_{F0} is larger than a threshold ξ_v, the entire word is declared as emphasized. The thresholds ξ_{F0} and ξ_v as well as the weights w_1 and w_2 need to be adapted to the individual, since each speaker has a different variation of his fundamental frequency. A speaker who uses a lot of pitch variation in his regular speech would need to have an essentially large variation v_{F0} in order to be considered as emphasizing his words. Likewise, a relatively monotonic speaker would not need to have such a large jump in pitch variation in order to be considered as emphasizing his words.

The weights and threshold are set to the individual by a short training. For this, a few utterances are manually labeled with prominence. Afterwards the weights and thresholds are adjusted until the labeled prominence is automatically detected in the manually labeled utterances. After the training we manually labeled additional utterances in order to evaluate the designed algorithm. The algorithm is able to correctly detect over 92% of the additionally labeled utterances [111].

5 Statistical Properties of Eye Blinks and Movements

In this section important statistical properties of eye blinks and eye movements are investigated. Note that extreme values of eye movements and blinks are eliminated in order to prevent unnatural animations.

First gaze patterns and blinks are analyzed while listening and talking (Section 5.1) and additionally with respect to spoken language (Section 5.2). The characteristics of saccades such as direction and magnitude are investigated in Section 5.3. Finally, dependencies between head movements and saccades as well as eye blinks and saccades are analyzed (Section 5.4 and 5.5). The distributions may highly vary between individuals, because each human has his own nonverbal behavior. If useful, we present the results of both recorded human subjects.

5.1 Gaze and Blink Patterns

Psychological studies show a significant difference of human gaze patterns while listening and talking (Section 2.4). Hence, for each mode the relative frequency distributions of remaining in MG and GA as well as the duration between two consecutive eye blinks, which is denoted as non-blink duration, are calculated.

The relative frequency distributions of duration in MG and GA of human subject 1 while listening and talking are illustrated in Figure 5.1 and 5.2, respectively. In order to describe these relative frequency distributions it is useful to model them by a well-known probability distribution such as Poisson or geometric distribution. Before we decided to use the lognormal distribution to model the relative frequency distributions of durations of MG and GA, we tried other well-known probability distributions. However, we were not able to appropriately fit any one of these with the measured distributions. For instance, the Poisson distribution decreases too fast, i.e. large durations in MG cannot be modeled.

The lognormal distribution is defined as

$$f_{ln}(x) = \begin{cases} \frac{1}{x\sigma_{ln}\sqrt{2\pi}} \cdot e^{-\frac{\left(\ln(x)-\mu_{ln}\right)^2}{2\sigma_{ln}^2}} & ; \quad x > 0 \\ 0.0 & ; \quad x \leq 0 \end{cases} \tag{5.1}$$

with the parameters μ_{ln} and σ_{ln} [1]. Lognormal distributions are often used if measurements show a more or less skewed distribution. Skewed distributions are particularly common when mean values are low, variances large, and all values equal or larger than

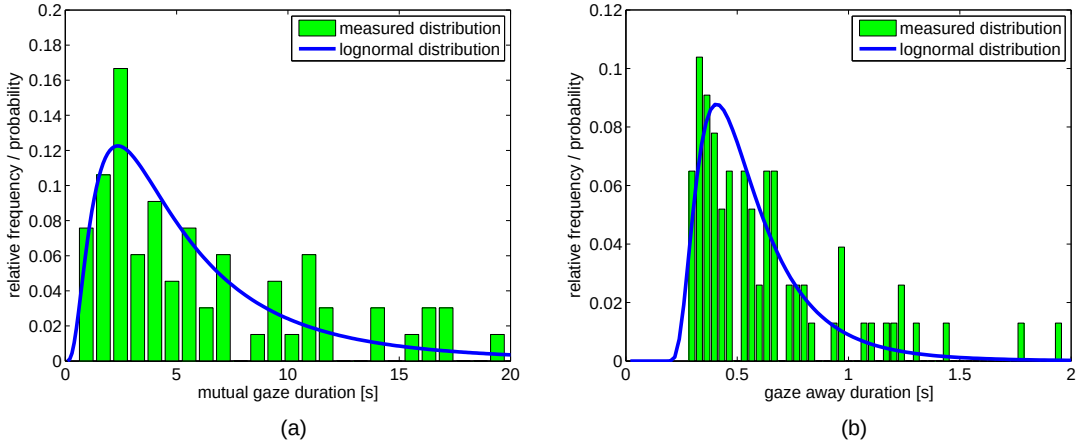

Figure 5.1: Relative frequency and fitted lognormal distribution of the duration of (a) mutual gaze ($\bar{X} = 5.35\,\text{s}$, $S = 5.63\,\text{s}$) and (b) gaze away ($\bar{X} = 0.63\,\text{s}$, $S = 0.35\,\text{s}$) while listening.

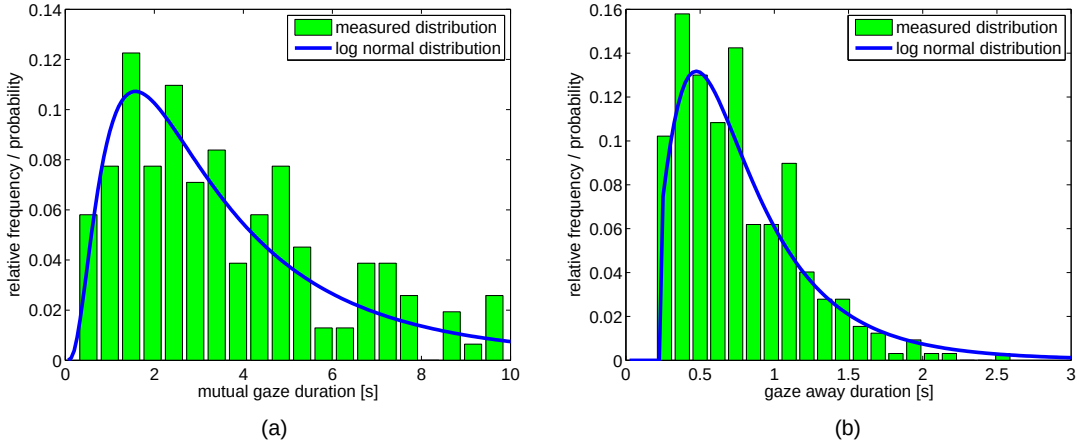

Figure 5.2: Relative frequency and fitted lognormal distribution of the duration of (a) mutual gaze ($\bar{X} = 1.6\,\text{s}$, $S = 0.83\,\text{s}$) and (b) gaze away ($\bar{X} = 0.89\,\text{s}$, $S = 0.52\,\text{s}$) while talking.

Table 5.1: The sample mean $\bar{X}$ and sample standard deviation S of mutual gaze duration, gaze away duration and non-blink duration of two recorded persons while listening and talking.

	Listening				Talking			
	Subject 1		Subject 2		Subject 1		Subject 2	
	$\bar{X}$ [s]	S [s]	$\bar{X}$ [s]	S [s]	$\bar{X}$ [s]	S [s]	$\bar{X}$ [s]	S [s]
Mutual gaze	11.93	10.59	5.35	5.63	4.17	3.16	1.60	0.83
Gaze away	0.68	0.3	0.63	0.35	0.76	0.39	0.89	0.52
Eye blinks	5.29	4.15	2.55	1.52	5.67	4.76	2.44	1.15

zero [91], e.g. the duration between two consecutive gaze shifts. These skewed distributions often closely fit the lognormal distribution [116], which are used in various fields such as geology, human medicine, ecology, linguistics, social sciences and economics.

For modeling the lognormal distribution $f_{ln}(x)$ from Equation (5.1) is fitted to the relative frequency distribution by a maximum likelihood estimation (MLE) resulting in

$$\hat{f}_{ln}(x) = \begin{cases} \frac{1}{(x-x_{ln})\sigma_{ln}\sqrt{2\pi}} \cdot e^{-\frac{\left(\ln(x-x_{ln})-\mu_{ln}\right)^2}{2\sigma_{ln}^2}} & ; \quad x > x_{ln} > 0 \\ 0.0 & ; \quad x \leq x_{ln} \end{cases} \tag{5.2}$$

with the shift x_{ln}. Since we evaluate $\hat{f}_{ln}(x)$ only at a discrete set of uniformly spaced points in time $x_k, \forall k \in [1,K]$, $\hat{f}_{ln}(x)$ is normalized, resulting in $\check{f}_{ln}(x_k)$. For $\check{f}_{ln}(x_k)$

$$\sum_{k=1}^{K} \check{f}_{ln}(x_k) = 1 \tag{5.3}$$

applies. The endpoints of the interval are the smallest x_1 and largest x_K measured durations.

We characterize the relative frequency distributions of both subjects by the sample mean $\bar{X}$ and standard deviation S (Table 5.1), which is common for data with a lognormal distribution [91]. Comparing the values while listening and talking in Table 5.1 depicts that the amount of time a person looks at the interlocutor is much higher while listening, which is consistent with [4]. We also acknowledge the great variation of the sample mean and sample standard deviation between both persons. This indicates, that each person has its own gaze pattern. Hence, in order to model an individual human character, his or her characteristic patterns need to be measured and modeled.

Furthermore, we observe, that the person returns to MG after executing a gaze shift in listening mode, whereas in talking mode the speaker may execute two consecutive gaze shifts. The second consecutive gaze shift is executed with an experimental probability of

Table 5.2: The experimental probability that the POR is located within one of the ROI, if the model is in mutual gaze.

left eye	right eye	mouth area	face
23.0%	20.8%	41.3%	14.9%

34%. Therefore, it is very reasonable to distinguish between listening and talking mode and to design appropriate models for each mode.

The analysis of eye tracking data indicates that humans do not constantly stare at the interlocutor in MG, but rather move the POR across the face, which is observed in both modes. In order to analyse the POR within MG in more detail, we define four regions of interest (ROIs): left eye, right eye, mouth area and facial area, which is not described by one of the previous ROIs. The first three ROIs are defined as an elliptic region (Figure 5.3a). In order to calculate the experimental probability that the POR is located within one ROI, the measured fixations, which range between 0.5 to 2.5 s, are assigned to the corresponding ROI. As an example the distribution of fixations of the ROI 'mouth area' is depicted in Figure 5.3b. The durations of fixations of the other ROIs are similar. The measured fixation duration is modeled by a normalized exponential distribution $\check{f}_e(x)$. An exponential distribution is defined as

$$f_e(x) = \begin{cases} \lambda_e e^{-\lambda_e x} & ; \quad x \geq 0 \\ 0.0 & ; \quad x < 0, \end{cases} \tag{5.4}$$

with the parameter λ_e. For modeling, $f_e(x)$ is fitted to the relative frequency distribution by MLE resulting in

$$\hat{f}_e(x) = \begin{cases} \lambda_e e^{-\lambda_e (x - x_e)} & ; \quad x \geq x_e \geq 0 \\ 0.0 & ; \quad x < x_e, \end{cases} \tag{5.5}$$

with the shift x_e. However, we evaluate $\hat{f}_e(x)$ only in the interval $[x_s, x_l]$, in which $x_s = x_e$ is the smallest and x_l the largest measured fixation duration. Therefore, $\hat{f}_e(x)$ is normalized in analogous manner to Equation (5.3) resulting in $\check{f}_e(x)$. Finally, the experimental probability that the POR is located within one particular ROI is calculated (Table 5.2). The measured probabilities are very similar for both modes.

In order to characterize the temporal course of blinks their non-blink duration distributions are calculated, which remain about the same while listening and talking (Table 5.1). The relative frequency distributions of two human subjects are displayed in Figure 5.4, which are also modeled by fitted lognormal distributions.

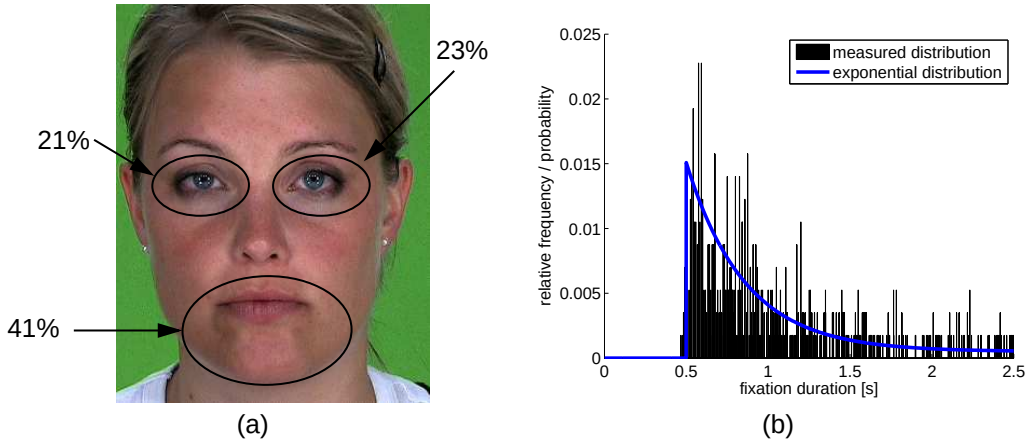

Figure 5.3: (a) The ROIs of the left and right eye as well as mouth area are marked by an ellipse. Each ROI is labeled with its corresponding experimental probability, that the POR is located within this ROI. (b) The relative frequency distribution of the duration of fixations within the ROI 'mouth area'. The distribution is modeled by a fitted and normalized exponential distribution.

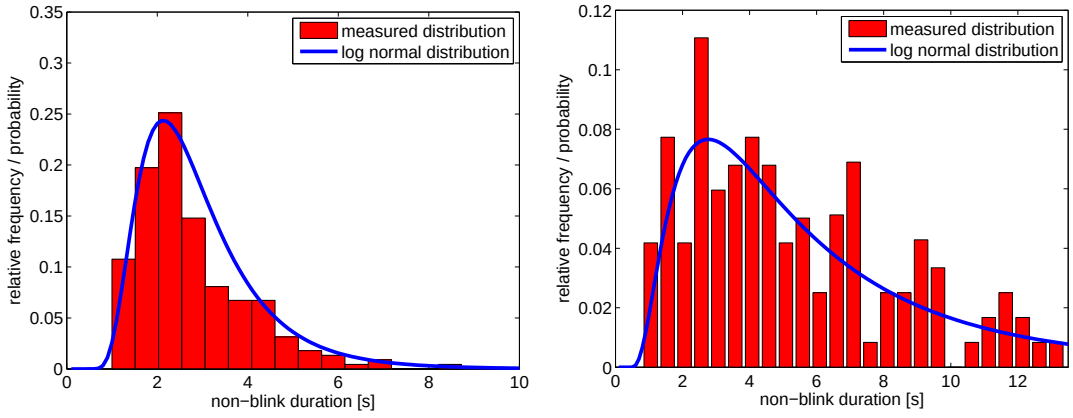

Figure 5.4: The relative frequency distributions of duration between two consecutive eye blinks is presented for two human subjects. Both frequency distributions are approximated by fitted lognormal distributions.

5.2 Gaze Patterns, Eye Blinks and Spoken Language

In this section the statistical dependencies between eye movements and spoken language as well as eye blinks and spoken language are determined.

5.2.1 Gaze Pattern and Spoken Language

Audio features, which have a dependency with gaze shifts and eye blinks, are denoted as observations o. For instance, audio features indicating a pause are labeled with the observation 'word boundary or pause' (WB). We are interested in determine the experimental conditional probability $p(\text{GS}|o)$ that the speaker performs a gaze shift (GS) given an observation o. If the audio features are extracted and converted to their corresponding observations and if we know $p(\text{GS}|o)$ as well, then we can generate new gaze patterns to arbitrary spoken output.

In the following the dependency between observations and GS is analyzed in detail. Each word of the spoken output is automatically classified as fast, slow or medium (Section 4.3.2). Since $p(\text{GS}|o = \text{normal}) \approx p(\text{GS}|o = \text{fast})$ we do not distinguish between these two types of observations. Words classified as slow are labeled with the observation 'slow speech rate' (SSR) because of $p(\text{GS}|o = \text{SSR}) \gg p(\text{GS}|o = \text{normal})$.

Since gaze patterns vary while thinking (Section 2.4), we want to automatically detect thinking mode. The three observations WB, SSR and 'filling word' (FW) may indicate that the speaker is hesitating and therefore in thinking mode. Some speakers use filling words (FW) such as 'ehm' while talking. These words are automatically detected by the phoneme labeling. All three observations do have a high probability $p(\text{GS}|o)$. Hence, if one of these observations occurs, then with a high probability a gaze shift is performed. On the other hand during 'word prominence' (WP) the speaker usually looks to the interlocutor and therefore $p(\text{GS}|o \neq \text{WP}) \gg p(\text{GS}|o = \text{WP})$. The observation WP indicates emphasized words (Section 4.3.3).

Moreover, we analyze the gaze behavior at the beginning and end of utterances. At the end of an utterance, the speaker always glances to the interlocutor as a turn-taking signal, which is consistent with [79]. Hence, we introduce the observation 'end of utterance' (E). Kendon et al. [79] observed that at the beginning of an utterance a GS is performed. We analyzed this in more detail and determined, that a GS is typically executed, if the speaker is in thinking mode, which is indicated by one of the following observations: WB, SSR or FW. All other words which are not labeled by one of the previous observations are labeled as 'other' (OT). An example of a labeled video is presented in Figure 5.5.

In Table 5.3 the experimental conditional probabilities $p(o|\text{GS})$ and $p(\text{GS}|o)$ are depicted for two human subjects. The probability $p(o|\text{GS})$ illustrates the distribution of observations while performing a GS and indicates the pertinence of a type of observation on the gaze patterns. For instance, subject 1 executes 33.5% of her GS during WB.

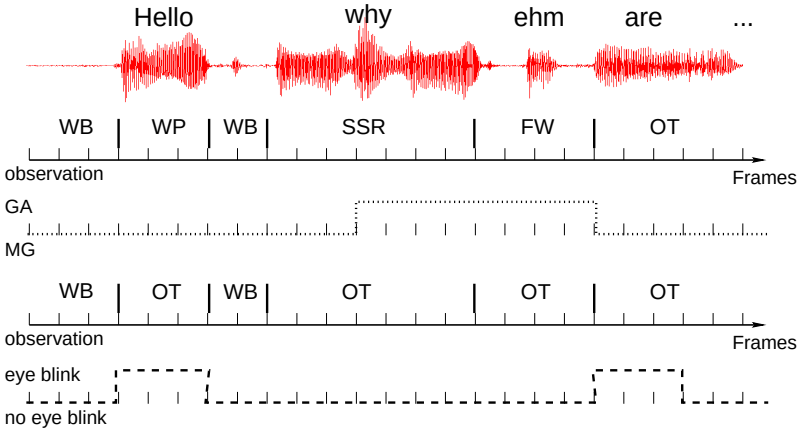

Figure 5.5: The speech waveform of a segment, which is labeled with its corresponding observations, is depicted. The speaker is either in MG or GA (illustrated by dotted line). The dashed line illustrates when the speaker performs eye blinks.

Table 5.3: For two recorded human subjects, the experimental conditional probabilities of gaze shifts and observations are presented (WB: word boundary, SSR: slow speech rate, FW: filling word, WP: word prominence, OT: other).

		WB	FW	SSR	WP	OT
Subject 1	$p(o\|\text{GS})$	33.5%	9.0%	33.0%	3.7%	20.8%
	$p(\text{GS}\|o)$	1.3%	1.5%	0.7%	0.1%	0.2%
Subject 2	$p(o\|\text{GS})$	41.2%	9.4%	14.6%	9.0%	29.6%
	$p(\text{GS}\|o)$	4.3%	3.3%	6.0%	0.4%	0.8%

Table 5.4: For two recorded human subjects, the experimental conditional probabilities of eye blinks and observations are presented (WB: word boundary, C: consonant, V: vowel).

	WB		C		V	
	$p(o\|\mathrm{B})$	$p(\mathrm{B}\|o)$	$p(o\|\mathrm{B})$	$p(\mathrm{B}\|o)$	$p(o\|\mathrm{B})$	$p(\mathrm{B}\|o)$
Subject 1	60.7%	3.4%	16.0%	1.3%	23.3%	1.3%
Subject 2	52.1%	1.8%	18.5%	0.4%	29.4%	0.4%

Another aspect is the relation between observations and the corresponding duration of remaining in GA. The probability distributions of duration of remaining in GA are very similar for each observation. Therefore, we assume, that the duration of GA is not influenced by the current observation. Note that statistical dependencies between small shifts moving the POR from one ROI to another and spoken language are not taken into account.

5.2.2 Eye Blinks and Spoken Language

Condon and Ogston [29] observed that eye blinks mainly occur during vocalization at the beginning of words or utterances, the initial vowel of a word and following the termination of a word. Hence, we label each frame of an utterance with one of the following observations: 'vowel' (V), 'consonant' (C) and 'word boundary' (WB). Since vowels already account for vocalization, we neglect vocalized consonants.

First, we calculate the experimental conditional probability $p(o|\mathrm{B})$ that the observation o occurs if a blink B is executed (Table 5.4). A large number of blinks are performed at WB, e.g. $p(o = \mathrm{WB}|\mathrm{B}) = 0.61$ of subject 1. Afterwards we determine the conditional probability $p(\mathrm{B}|o)$ that a blink is performed given o. This conditional probability indicates a high statistical dependency between WB and blinks, while the other two observations have low conditional probabilities. Since $p(\mathrm{B}|o = \mathrm{C}) \approx p(\mathrm{B}|o = \mathrm{V})$ the system does not distinguish between V and C, and labeles these simply as OT. Hence, we can use the same type of observations as for gaze patterns by converting all types of observations to OT except the observation WB (Figure 5.5). In Figure 5.6 two snapshots of the recorded video file are presented. Each file is labeled with its corresponding observations and whether the eyes are opened or closed.

5.3 Characteristics of Saccades

We analyze the characteristics of saccades, which induce a shift from MG to GA or vice versa. The characteristics of saccades are only briefly analyzed, since in [85] a comprehensive investigation is already presented. The important characteristics of saccades,

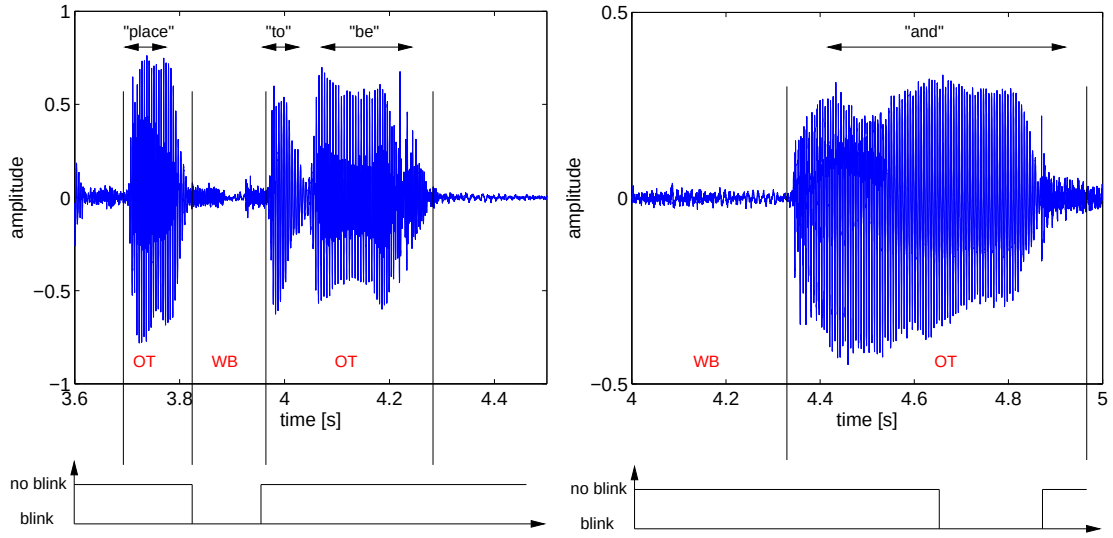

Figure 5.6: Two snapshots of the recorded video sequences. Each video file is labeled by its observations: word boundary (WB) or other (OT). Furthermore, the file is labled with eye blinks, which are automatically detected.

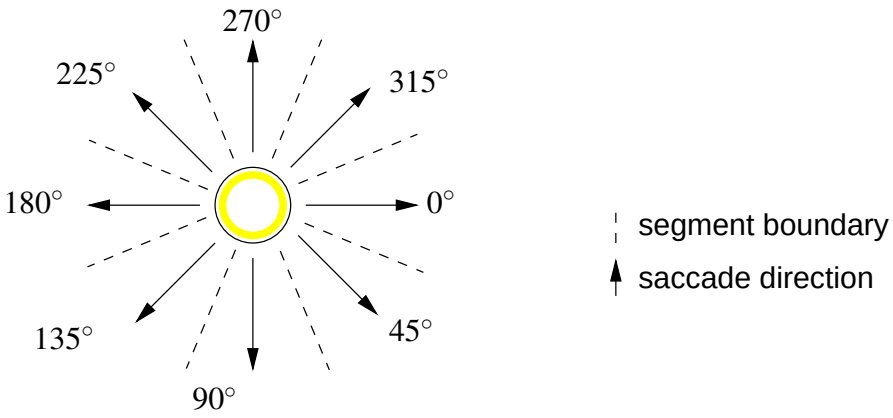

Figure 5.7: The directions of saccades are clustered into eight segments. Each segment covers 45°.

Table 5.5: Relative frequency distribution of saccade directions.

direction	relative frequency	direction	relative frequency
0°	0.16	180°	0.19
45°	0.09	225°	0.09
90°	0.15	270°	0.16
135°	0.08	315°	0.08

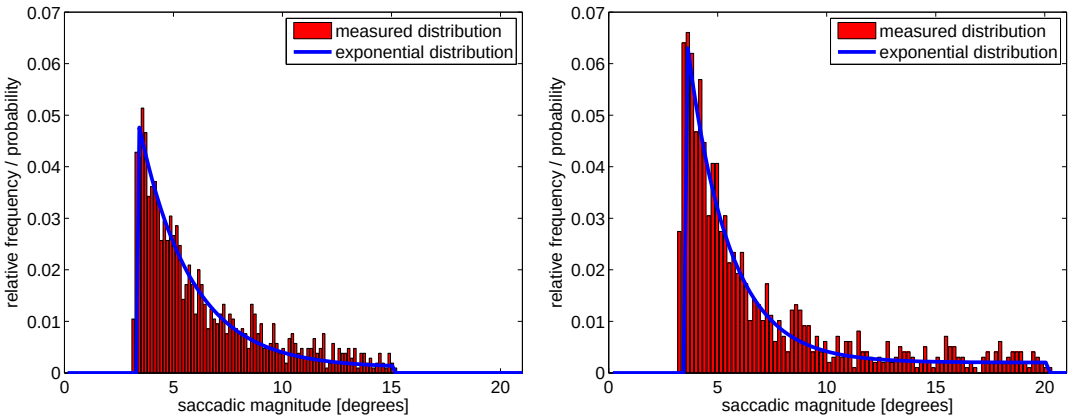

Figure 5.8: Relative frequency distributions of the magnitude of large saccades are superimposed with fitted exponential distributions. In general, the saccadic magnitude is larger while talking than listening. (a) listening mode ($\bar{X} = 6.3\,^\circ$, $S = 2.9\,^\circ$). (b) talking mode ($\bar{X} = 7.2\,^\circ$, $S = 4.5\,^\circ$).

which are analyzed in this section, are their direction and magnitude A^s. The direction of the saccadic eye movements are divided into eight evenly spaced partitions, each covering 45° (Figure 5.7). The relative frequency distribution (Table 5.5) indicates that there is a two times higher probability of performing vertical or horizontal instead of diagonal eye movements, which is consistent with [85].

The measured relative frequency distributions of saccadic magnitude, which are depicted in Figure 5.8, are modeled by exponential distributions (Equation (5.5)), which are normalized by taking the smallest and largest measured saccadic amplitude A^s into account. In general, the sample mean $\bar{X}$ as well as the standard deviation S of the saccadic magnitudes are larger in talking than listening mode.

5.4 Gaze Shifts and Head Movements

Humans employ varying amounts of saccadic gaze shifts in association with head movements. The probability of executing head movements increases as the saccadic magnitude grows. On the other hand if the amplitude of the evoked head movement grows, the probability of executing a saccade increases. These qualitative observations regarding the dependencies between head movements and saccades are generally accepted [16, 137].

Stahl [123] presents an attempt to describe the relation between head rotation and horizontal saccadic magnitudes in more detail. Within a certain magnitude range head movements and gaze shifts are independent. He shows this correlation only for horizontal saccades, whereas we assume, that head rotations also accompany saccades in other directions. Thus, if the saccadic magnitude A^s is larger than a threshold e.g. 15°, then the head movement accompanies the saccade. Instead of generating a head movement we adapt the direction of the saccade according to the head motion of the background sequence.

Speech is usually accompanied with head movements. According to [9] the head movement amplitude is related to the gaze shift amplitude. In order to determine the head movement amplitude, we need to analyze the head motion. For instance, a nod of the head often accompanies a stress on a word [56]. A nod may consist of two sub-trajectories, a downward and upward pitch rotation. In the following, an algorithm to automatically segment the head motion into sub-trajectories is designed. The algorithm is based on clustering the head motion by variations of the angular velocity $\mathbf{r}$ between consecutive frames.

First the head pose parameters are estimated as described in Section 3.5. Since noise implies pose errors, the previous estimated rotation angels ω_x, ω_y and ω_z are initially low-pass filtered before the angular velocity $\mathbf{r}$ is calculated. The direction of $\mathbf{r}$ gives the direction of the axis about which the rigid body is rotating, whereas the magnitude $|\mathbf{r}|$ tells how fast the body is turning. In order to determine the angular velocity $\mathbf{r}_t$ between two consecutive frames $\mathbf{I}(t)$ and $\mathbf{I}(t-1)$ the rotation expressed by the unit quaternion q between both frames is determined. From this unit quaternion we can determine the rotation axis $\mathbf{v}_q$ and rotation angle θ_q using Equation (3.9). Hence, the angular velocity $\mathbf{r}_t$ is equal to the product

$$\mathbf{r}_t = \mathbf{v}_q \cdot \theta_q. \tag{5.6}$$

Each sub-trajectory l, denoted as $\hat{s}_l = (\hat{\mathbf{r}}_l, l_i, l_f)$, is characterized by its average angular velocity

$$\hat{\mathbf{r}}_l = \frac{1}{l_f - l_i + 1} \sum_{j=l_i}^{l_f} \mathbf{r}_j, \tag{5.7}$$

where l_i is the initial and l_f the final frame position of the sub-trajectory l in the image sequence $\mathbf{I}(t)$. The trajectories are clustered by defining an empirically chosen error

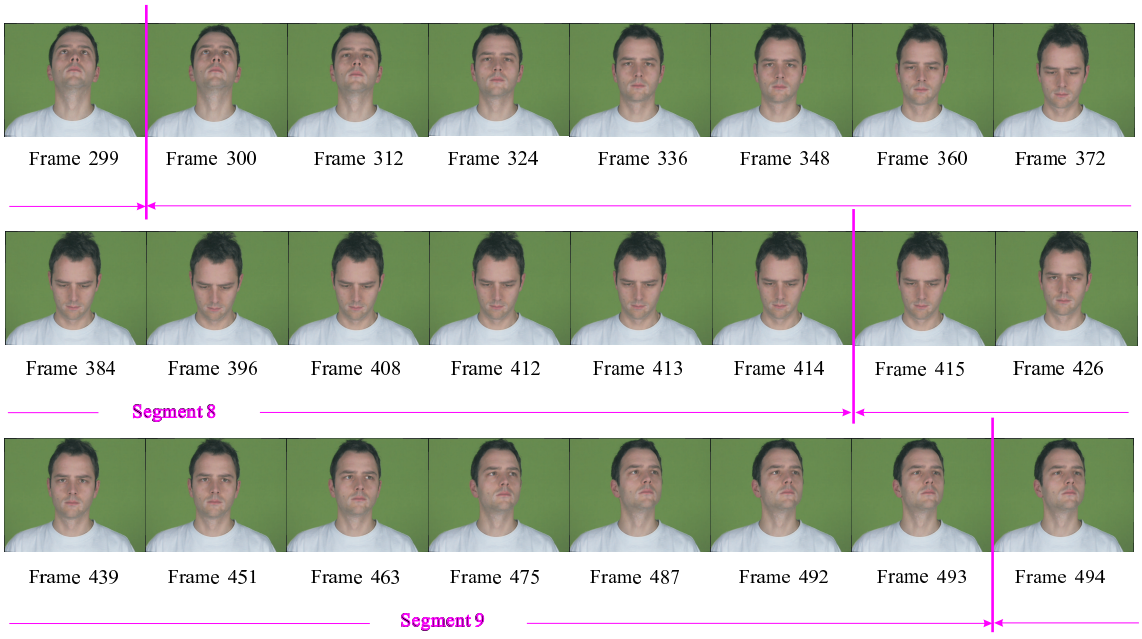

Figure 5.9: In this image sequence, segment 8 and 9 are illustrating the segmentation. In segment 8 the head pitches down, while in segment 9 the head rotates to the left and up.

threshold ξ_s and cost function $C_s(\mathbf{r}_t, \hat{s}_l)$

$$C_s(\mathbf{r}_t, \hat{s}_l) = \| \mathbf{r}_t - \hat{\mathbf{r}}_l \|_{L2}, \tag{5.8}$$

between the current frame $\mathbf{I}(t)$ and trajectory $\hat{s}_l$. If the cost $C_s(\mathbf{r}_t, \hat{s}_l)$ is larger than the threshold ξ_s a new segment is started ($l \rightarrow l+1$). Otherwise frame $\mathbf{I}(t)$ is added to the current segment $\hat{s}_l$ by setting $l_f = t$ and updating the current average angular velocity $\hat{\mathbf{r}}_l$ with Equation (5.7). In Figure 5.9 a segmented recorded sequence is presented in which the major motion trajectories are determined.

Practically speaking the head motion in the background sequence is partitioned into its sub-trajectories $\hat{s}_l$. Each sub-trajectory $\hat{s}_l = (\hat{\mathbf{r}}_l, l_i, l_f)$ can be described by a unit quaternion q_{h_l} (Equation 3.9), which describes the rotation of the head from frame l_i to frame l_f. The corresponding rotation angle θ_{h_l} from the quaternion q_{h_l} gives the magnitude of the head rotation. If θ_{h_l} is larger than a threshold, a saccade accompanying the head rotation is generated.

5.5 Gaze Shifts and Eye Blinks

Eye animations concentrate on gaze patterns and usually do not pay attention on eye blinks. For instance, Lee et al. [85] observed a dependency between both events but did

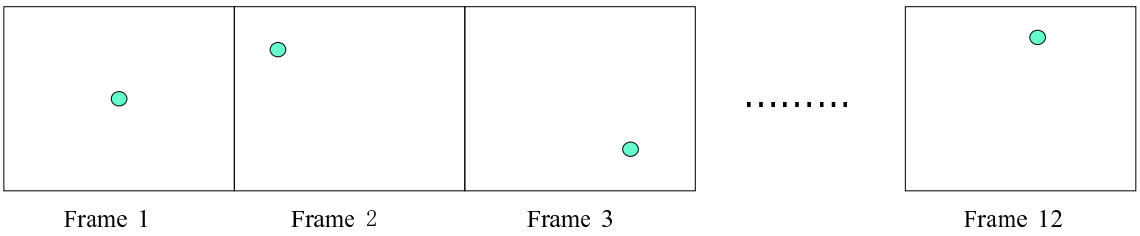

Figure 5.10: An eye tracker measures the POR, while the participant tracks the green dot throughout the image sequence. The saccadic magnitudes vary between 5° and 30°.

not analyze this aspect any further. In 1969 Cranach et al. [32] already investigated a correlation between gaze shifts and blinks. Namely the larger the gaze shift the higher the probability of simultaneously executing a blink and gaze shift. Their investigations are supported by [138, 46].

In this Section, we specify the statistical dependency between gaze shifts and eye blinks in more quantitative terms. We analyze the dependencies between B and GS and determine the experimental conditional probability $p(\mathrm{B}|\mathrm{GS})$ in the recorded videos. Note that eye blinks do not induce gaze shifts and thus, $p(\mathrm{GS}|\mathrm{B})$ is not taken into account.

While listening $p(\mathrm{B}|\mathrm{GS})$ is only 2.7% of subject 1, whereas it increases to 12.6% for subject 2. This high variation is due to the different non-verbal behavior of both subjects. While subject 2 regularly executes large head rotations, e.g. while laughing, subject 1 exhibits less emotions. Since a large head rotation induces a large gaze shift which in turn induces an eye blink, $p(\mathrm{B}|\mathrm{GS})$ of subject 2 is much higher than that of subject 1. In case of facial animation we do not get any feedback from the interlocutor in listening mode, and therefore the talking-head only performs typical short head movements in the background sequence. If we only consider eye blinks during small or normal head movements $p(\mathrm{B}|\mathrm{GS})$ decreases to 2.5%. Then we may assume that the events B and GS are statistically independent because of $p(\mathrm{B}|\mathrm{GS}) = 2.7\% \approx p(\mathrm{B}) = 2.82\%$. Hence, we are allowed to design two independent control models in listening mode, one for eye blinks and one for eye movements.

In talking mode, $p(\mathrm{B}|\mathrm{GS})$ significantly increases to 23.2% of subject 1 and even 64.5% of subject 2. This high probability is due to the higher number of large saccades performed while talking. As a result we need to couple saccades and blinks in one control model. For this, we analyze the dependency in more detail by designing an experiment in which we measure the experimental conditional probability $p(\mathrm{B}|A^s)$ of executing a blink B, given the saccadic magnitude A^s. Note that A^s already implies a gaze shift. In the experiment we present single images on which a single green dot is displayed (Figure 5.10). We ask the participant to follow the green dot in the image sequence with the eyes, while measuring the POR with an eye tracker. The spatial difference between consecutive dots is between 5° and 30°. We can determine whether a blink is executed while changing the POR. The

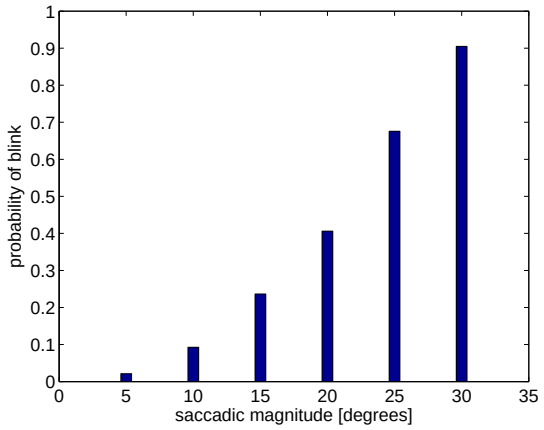

Figure 5.11: The experimental conditional probability $p(\mathrm{B}|A^s)$ of executing an eye blink, given the magnitude of a saccade A^s. In general, the probability increases with an increase of A^s.

experimental results (Figure 5.11) can be described by the following function

$$p(\mathrm{B}|A^s) = \begin{cases} 0.02 & ; \quad 5 \leq A^s < 7.5 \\ 0.09 & ; \quad 7.5 \leq A^s < 12.5 \\ 0.24 & ; \quad 12.5 \leq A^s < 17.5 \\ 0.41 & ; \quad 17.5 \leq A^s < 22.5 \\ 0.68 & ; \quad 22.5 \leq A^s < 27.5 \\ 0.90 & ; \quad \text{else.} \end{cases} \tag{5.9}$$

In general, $p(\mathrm{B}|A^s)$ increases by an increase of the saccadic magnitude A^s as proposed in [32]. Since the execution of a blink during a large saccade is controlled by the subconscious and speech does not seem to be a dominant factor, we assume, that the statistical relationship between a saccade and an eye blink in this experiment is the same as in a two-way conversation.

6 Eye Control Unit

First the characteristics of eye globe rotations (Section 6.1) and models to generate the different types of eye movements are explored (Section 6.2). The models of the ECU, which control eye blinks and movements while listening and talking, are explained in Section 6.3 and 6.4 respectively. An overview of the ECU is depicted in Figure 6.1. The ECU selects the appropriate models depending on the current mode, which in turn control eyelid movements as well as eye globe rotations. The mutual gaze model and the models of eye movements are used by both modes.

6.1 Characteristics of Eye Globe Rotation

Eye globe orientation can be either described by the three rotation angles β_x, β_y and β_z (Equation 3.1) or a quaternion q with the rotation axis $\mathbf{v}_q$ and rotation angle θ_q (Equation 3.10) as illustrated in Figure 6.2. Donders' law [39] states that each time the eye looks in a particular direction, it only assumes one 3D orientation. Only if the eyes are in primary position, torsion is not induced.

In order to better illustrate the problem, an example from literature is used [21]. In Figure 6.3 two commonly used coordinate systems in eye movement research, Fick's [51] and Helmholtz's [66] system are presented. There are two possible ways of specifying the tertiary position. Either first a rotation around the vertical or horizontal axis is performed resulting in a different orientation of the cross in both cases. Humans, however, perceive the environment in the same way, whether humans first look up and then to the right or vice versa, which agrees with Donders' law. Hence, the different orientations caused by the different orders of applied rotations are compensated by inducing torsion.

Listing's law, which is considered to be one of the most important principles in eye movement physiology, further states exactely what those torsional values are [66, 133]. It states when the head is upright, stationary and the eyes fixating a distant object, all rotation axes lie within the same plane, denoted as Listing's plane. In our defined eye coordinate system Listing's plane is orthogonal to the primary gaze position and intersects the globe center. Since the primary gaze position is equal with $\mathbf{Z}^o$, Listing's plane is spanned by the object coordinate axes $\mathbf{X}^o$ and $\mathbf{Y}^o$ of the eye globe (Figure 6.2). Thus, the rotation axis $\mathbf{v}_q$ is located within this plane, which is achieved by setting $q_3 = 0$ of the quaternion representation in Equation (3.3). In case we use the rotation matrix of Equation (3.1) to describe the rotation of the eye globe, a torsion is induced. The amount of rotation β_z can be determined with simple algebraic manipulation by setting in Equation (3.13) q_3 to zero resulting in

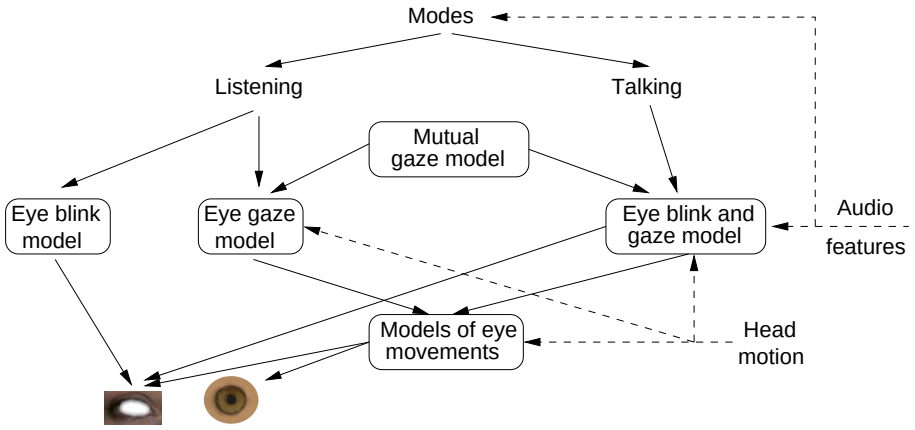

Figure 6.1: Overview of the eye control unit: While in listening mode two independent models control the eyes, in talking mode one model is designed. Both modes share the mutual gaze model and models of eye movements. Whereas the eyelid position is controlled by the eye blink models as well as the models of eye movements, the eye globe is only steered by the latter. Head motion and audio features are input parameters (dashed arrows).

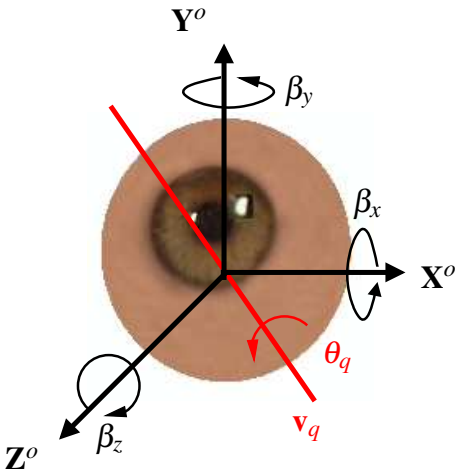

Figure 6.2: Each eye globe has its own object coordinate system $(\mathbf{X}^o, \mathbf{Y}^o, \mathbf{Z}^o)$ with the origin intersecting with the globe center. If the eye looks straight forward, then the line of sight is equal to $\mathbf{Z}^o$ (primary position). Rotation of the eye globe can be either executed by a rotation matrix with the Euler angels $(\beta_x, \beta_y, \beta_z)$ or quaternion q with the rotation angle θ_q along the rotation axis $\mathbf{v}_q$.

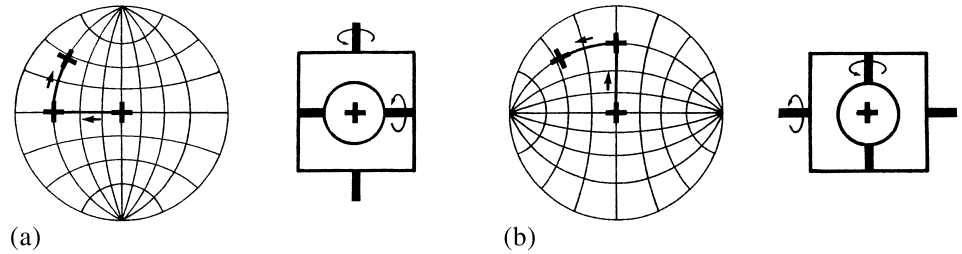

Figure 6.3: There are two possible was of reaching the tertiary position [21]: (a) Fick's system [51] executes two consecutive rotations first around the vertical and then horizontal axis. (b) Helmhotz's system [66] performs these rotations vice versa. The final orientation of the cross, however, varies.

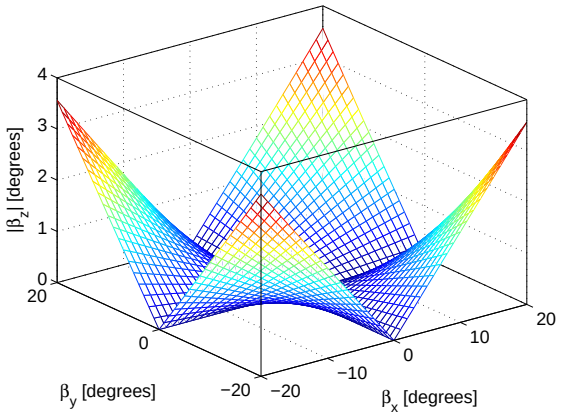

Figure 6.4: Torsional angle β_z with respect to β_x and β_y. If large saccades of 20° are executed then the torsional component may reach 3.5°.

$$
\begin{aligned}
r_{21} &= r_{12} \\
cos(\beta_y)sin(\beta_z) &= sin(\beta_x)sin(\beta_y)cos(\beta_z) - cos(\beta_x)sin(\beta_z) \\
sin(\beta_z)\,[cos(\beta_y) + cos(\beta_x)] &= sin(\beta_x)sin(\beta_y)cos(\beta_z) \\
tan(\beta_z) &= \frac{sin(\beta_x)sin(\beta_y)}{cos(\beta_y) + cos(\beta_x)} \\
\beta_z &= atan\left[\frac{sin(\beta_x)sin(\beta_y)}{cos(\beta_y) + cos(\beta_x)}\right], \forall \beta_x, \beta_y \in (-\frac{\pi}{2}, \frac{\pi}{2}). \quad (6.1)
\end{aligned}
$$

The torsional influence of Listing's law on saccades is presented in Figure 6.4.
Since Listing's law is only fulfilled under the mentioned conditions, we need to deter-

mine the impact on violations on Listing's law. In our system, the talking-head looks to the interlocutor and not at a far distant object. Saccades with different levels of binocular convergence, denoted as vergence movements, obey a variant of Listing's law called L2 [134]. Binocular convergence refers to the vergence angle, which is the angle between both eyes and the POR. In order to estimate the influence of the vergence movements on Listing's law, we assume a distance of 60 mm between both eye globes and a distance of 1200 mm between the talking-head and interlocutor resulting in a vergence angle of 2.8°. In [134] the influence of L2 on the orientation of Listing's plane is one fourth of the vergence angle in our example resulting in 0.7°. Thus, in our set-up, the influence of binocular convergence is negligible small. Furthermore, eye movements, which neither start nor end in the primary position, do only fulfill Listing's law by a rule called half-angle [64], which we take into account. Under this condition the plane of rotation is tilted by half the angle between the momentary and primary position.

While pitch and yaw head motions cause only a small change in the orientation of Listing's plane, head tilts induce ocular counterroll to keep the image upright. Schworm et al. [119] acquired data from five subjects using video oculography devices to measure ocular counterroll induced by head tilts (Figure 6.5). The ocular counterrolls induce a torsion to the eye globe and thus a rotation around $\mathbf{Z}^o$ (Figure 6.2). For the sake of simplicity, we use the average counterroll of both eyes, although in [119] a slightly different counterroll between the left and right eye was measured. For the animation the course of the measured values is approximated by a cubic interpolation as suggested in [109]

$$\beta_z = p_1 \cdot \omega_z^3 + p_2 \cdot \omega_z^2 + p_3 \cdot \omega_z + p_4 \tag{6.2}$$

with ω_z equal to the head tilt and the parameters p_1 to p_4 equal to $3.7 \cdot 10^{-5}$, $-3.1 \cdot 10^{-6}$, $-2.1 \cdot 10^{-1}$ and $-2.3 \cdot 10^{-1}$, respectively. The sign in Equation (6.2) depends on the direction of head tilt. The torsional component according to Equation (6.1) and (6.2) may add up to 15°. Head tilts do not only influence saccades but VOR as well.

6.2 Models of Eye Movements

Two types of eye movements saccades and VOR are executed in the animation system. For each movement we design a model.

6.2.1 Model of Saccadic Movements

In order to shift the gaze from mutual gaze to gaze away, a large saccade is performed. The generation of a saccade is based on the work of [85]. We improve their model by including Listing's law, head tilts and eyelid movements. Saccades are generated in six steps:

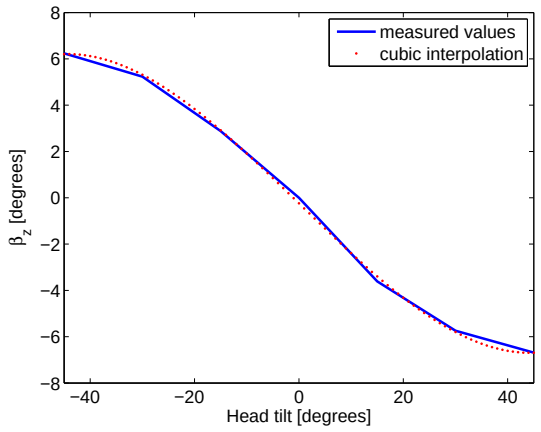

Figure 6.5: In [119] the relationship between head tilts and eye counterrolls is investigated. We model these measurements by a cubic interpolation.

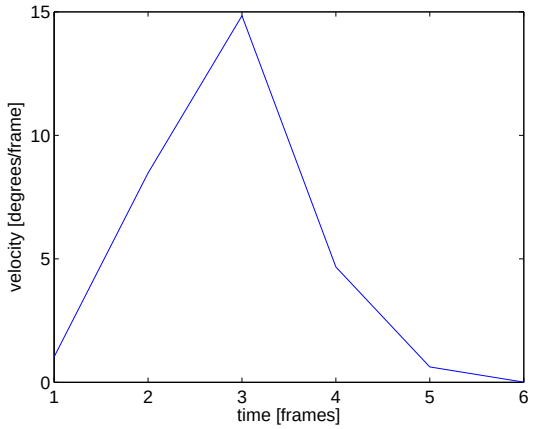

Figure 6.6: Instantaneous velocity function of saccades with a sampling frequency of 30 frames per second [85].

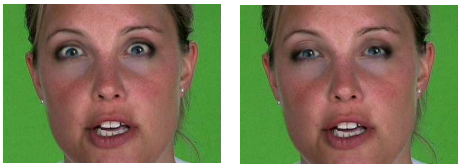

Figure 6.7: The relationship between vertical saccadic movements and eyelid position is illustrated. While on the left the eyeball and lid movements are independently controlled, the right shows our proposed method where the eyelid follows the eye globe motion.

1. First, the magnitude A^s of the saccade is determined by modeling one of the fitted exponential distribution in Figure 5.8. Depending on the current mode listening or talking either the distribution in Figure 5.8a or Figure 5.8b is selected. In order to model the exponential distribution rejection sampling is used (Appendix A). Note that rejection sampling gives the opportunity to model arbitrary 2D probability distributions.

2. The direction of the saccade is determined by modeling the relative frequency distribution in Table 5.5. If either the saccadic magnitude A^s or head rotation θ_h is larger than a threshold (Section 5.4), the saccade accompanies the head rotation.

3. The direction and magnitude A^s of the saccade is converted to the rotation angles β_x and β_y. The amount of induced torsion β_z is determined according to Equation (6.1) and (6.2).

4. The duration of the saccade D^s is the amount of time necessary to execute the saccade and is proportional to its magnitude as defined in Equation (2.1). For example, to execute a saccade with magnitude $A^s = 15^\circ$ requires less than 61 ms.

5. In general, a saccade starts with a quick acceleration followed by an equally rapid deceleration [12]. In [85], an "instantaneous velocity curve" of measured saccades is provided (Figure 6.6). This curve describes the typical course of the velocity of a saccade. First, the speed rises to a maximum and then drastically drops. In synthesis, the velocity of the saccade is then determined using the instantaneous velocity curve in Figure 6.6, which is adapted to the desired duration D^s.

6. Vertical saccades and eyelid movements are coupled as explained in Section 2.1.2. For this, we define multiple saccadic magnitude thresholds in vertical direction (up and down). If the saccadic magnitude is larger than an empirically selected threshold, then the appropriate eyelid is selected from the database. Figure 6.7 shows two frames extracted from a synthesized video with and without considering this aspect.

If the gaze is shifted from GA to MG, then the direction and magnitude of the executed saccade are already defined. Hence, only the parameters of the executed saccade need to be determined. If the system is in MG and the POR is moved from one ROI to another only very small saccades are executed. Whereas the duration D^s and velocity are negligible, torsion and eyelid movements are taken into account.

6.2.2 Model of Vestibulo-ocular Reflex (VOR)

In order to fixate the retina onto an object during head rotation, VOR executes eye movements compensating head motions. The performance of VOR is determined by the combined activation of the simicircular canals and the otolith organs as well as by the coordinated function of central integrative processes that drive the oculomotor system. The VOR can be further divided into angular and linear VOR [113]. Since linear VOR has only very little influence in our set-up, we do not take linear VOR into account.

For perfect compensation, the rotation axes of the eye and head need to be parallel. It has been shown, however, that the axis of eye rotation during VOR neither meets the needs for perfect 3D gaze stabilization nor Listing's law. It is a compromise of both constraints. Since the variations are neglectable small, we perform a perfect compensation. Furthermore, the latency of VOR is less than 14ms and therefore neglectably small in our system.

The thresholds of eye movements in vertical direction, which in turn induce an eyelid motion as described in Section 6.2.1, are also considered in this model.

6.3 Listening Mode

In listening mode the animation system is waiting for input from the user, e.g. a mouse click. Since the system does not monitor the user, we do not know the user's attitude and therefore we cannot generate the expected nonverbal behavior of the talking head. Hence, we only synthesize neutral head movements and small variations of the facial expression, which are defined by the background sequence. Under this condition we can design two independent control models, one for eye blinks and one for eye movements, because eye blinks and gaze shifts are statistically independent (Section 5.5).

6.3.1 Eye Gaze Pattern

A finite state machine (FSM) [76] with two states MG and GA, synthesizes new gaze patterns. The designed FSM is depicted as a statechart in Figure 6.8. The semantics of statecharts are explained in Appendix B. Note, that all successive FSMs are illustrated as statecharts, too. Initially the duration $\hat{t}_g^{mg}$ or $\hat{t}_g^{ga}$ of remaining in the current state, which are both given in seconds, is determined by modeling the normalized lognormal distributions $\check{f}_{ln}$ in Figure 5.1.

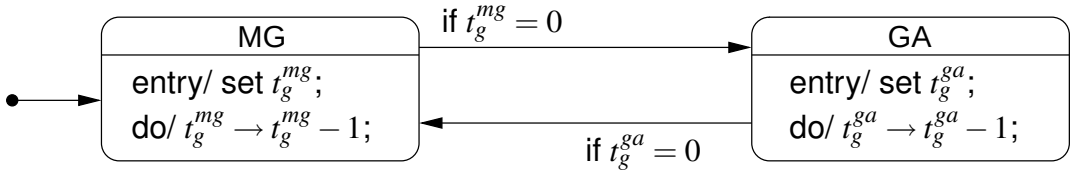

Figure 6.8: FSM to generate gaze patterns in listening mode. Two states, MG and GA, are used to generate gaze patterns. The duration t_g^{mg} and t_g^{ga} are determined by modeling the distributions in Figure 5.1.

In order to convert the durations $\hat{t}_g^{mg}$ and $\hat{t}_g^{ga}$ into number of frames, we divide $\hat{t}_g^{mg}$ and $\hat{t}_g^{ga}$ by the duration of a frame τ_v resulting in t_g^{mg} and t_g^{ga}. For instance, the duration of a frame τ_v lasts for $0.04s$ if we synthesize 25 frames per second. Note that in all following FSMs we determine the duration in frames instead of seconds, since the number of frames is obviously proportional to time in seconds.

By default, the animation starts in mutual gaze. While the model remains in the same state t_g^{mg} and t_g^{ga} decrease. If t_g^{mg} or t_g^{ga} is eventually equal to zero, a saccade is performed to switch the state (Section 6.2.1). Note that the dependency between head motion and saccades is taken into account, too (Section 5.4).

Mutual Gaze

In MG in both modes the eyes move across the face (Section 5.1). This aspect is taken into account to better model the measured distributions. In this way, we prevent jerky eye movements, which occur if a saccade with a large magnitude and a short duration in MG and GA is generated as observed in [85].

If the FSM is in MG (Figure 6.8), the eyes still perform small gaze shifts from one ROI to another (Figure 5.3a). This observation is modeled by refining the state MG with a second FSM with four states each representing one ROI. The initial state is randomly selected by modeling the experimental probabilities in Table 5.2. Initially after entering the state j, the number of frames t_{ROI}^j of remaining in the current state j is determined by modeling the normalized exponential distribution. An example of such a distribution is illustrated in Figure 5.3b. The FSM remains in the current state until t_{ROI}^j is equal to zero. After leaving the current the next state is determined by testing a second condition, which models the measured probabilities in Table 5.2. For this, for each state j the interval endpoints p_{ROI}^{1j}, p_{ROI}^{2j} and p_{ROI}^{3j} are accordingly set. Note that the current state j is left, if the duration of remaining in MG ends.

As an example the statechart of one state the ROI 'mouth area' is illustrated in Figure 6.9. The other states do have an identical structure, but with different interval endpoints, which are determined according to Table 5.2.

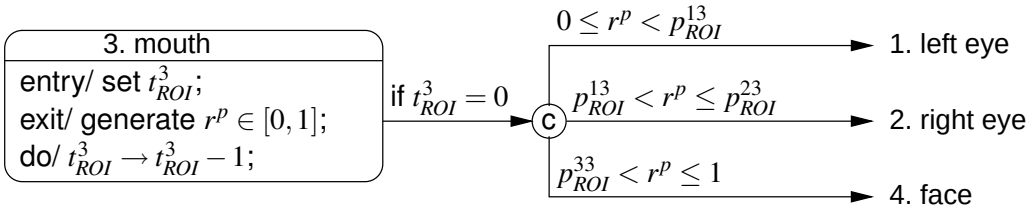

Figure 6.9: As an example only the state 'mouth area' ($j = 3$) from the FSM modeling mutual gaze is depicted. When the state is entered the duration t^3_{ROI} is set. If t^3_{ROI} is equal to zero, then the condition connector selects the next state depending on which condition is satisfied.

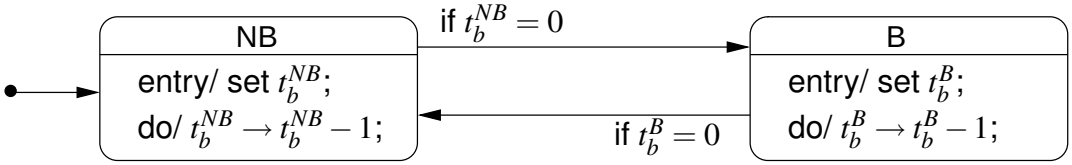

Figure 6.10: FSM of the eye blink model in listening mode with the states no eye blink (NB) and execute an eye blink (B).

6.3.2 Blink Patterns

A FSM with two states 'no blink' NB and 'blink' B as depicted in Figure 6.10, generates novel eye blink patterns. In the default state NB, the eyes are open and in B an eye blink is performed. When initially entering a state the number of frames remaining in the current state either t^{NB}_b or t^B_b is set by modeling the lognormal distribution in Figure 5.4. Afterwards the model remains in the current state until t^{NB}_b or t^B_b is equal to zero. In state B the value t^B_b is set according to the duration of the current eye blink, which may also vary.

6.4 Talking Mode

In talking mode two independent models controlling gaze shifts and eye blinks cannot be designed, since eye movements and blinks are coupled (Section 5.5). Hence, we propose an algorithm, which iteratively determines an animation path (Figure 6.11). The animation path contains for each frame of the synthesized video all information consisting of the gaze and blink patterns as well as the characteristics of the performed saccades in order to generate eye animations. In the following each step of the algorithm is explained:

1. First each frame of the animation path is labeled with its corresponding observation

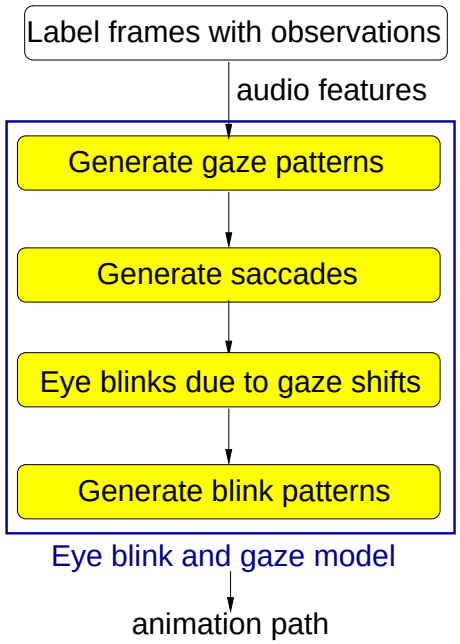

Figure 6.11: Block diagram of generating the animation path in talking mode.

o as extracted from the spoken output.

2. For the entire animation, the gaze patterns are determined by a FSM with three states MG, GA_1 and GA_2 (Figure 6.12). The FSM starts in the default state MG. In this state, the current observation o is extracted from the spoken output and a random number r^p is generated. If r^p is smaller than the experimental conditional probability $p(\mathrm{GS}|o)$, the state is changed to GA_1. In order to switch the state from MG to GA a saccade is executed. Since the duration t_g^{ga} of remaining in GA_1 as well as GA_2 is independent of o, t_g^{ga} is determined by modeling the lognormal distribution in Figure 5.2b. A second consecutive gaze shift in a different direction is executed, if the system switches to the state GA_2. The transition in this state is achieved with a probability of 34%. After these one or two GS, the model returns to MG. At the end of the utterance, the talking-head is looking to the interlocutor (MG). Note, that gaze shifts due to strong head motion are considered (Section 5.4). While the system remains is MG, the POR is varied by a FSM as explained in Listening Mode (Section 6.3.1).

3. The magnitudes A^s and directions of all executed saccades are determined. Preliminary studies indicate that there are no statistical dependencies between the saccadic magnitude of previously executed saccades, GA duration and the current saccade. Hence, each large saccade can be independently generated as described in Section 6.2.1. If necessary, the head motion in the background sequence is taken into account (Section 5.4).

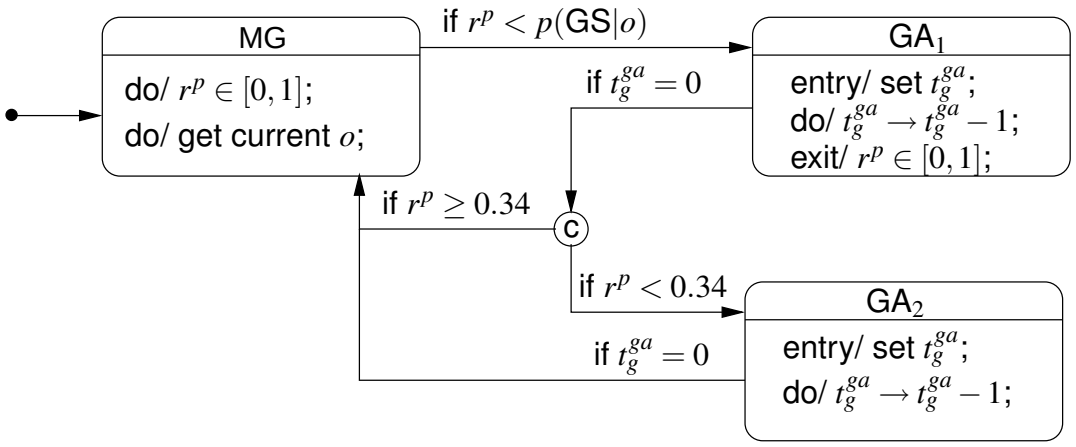

Figure 6.12: FSM: Gaze model in talking mode.

4. Eye blinks, which are simultaneously executed with a gaze shift, are added to the animation path. Since the magnitude A^s of large saccades as well as the experimental conditional probability $p(\mathrm{B}|A^s)$ are known, the probability of executing a blink can be calculated. A blink is executed, if the random number from a uniform probability distribution between zero and one is less than the conditional probability $p(\mathrm{B}|A^s)$.

5. Finally, additional eye blinks are added to the animation path by a more sophisticated FSM than designed for Listening Mode (Section 6.3.2). While the model synthesizing new gaze patterns in step 2 uses the conditional probability $p(\mathrm{GS}|o)$ (Figure 6.12), eye blinks cannot be generated by only taking the observation o into account. The temporal dependency of blinks must be considered, since eye blinks fulfill the biological purpose to regularly wet the cornea and remove irritants from the surface of the cornea and therefore humans do regularly blink.

   Hence, we determine the conditional probability $p(\mathrm{B}|t_b^{NB}, o)$ of performing a blink B given observation o and time t_b^{NB} passed since the last blink. With simple algebraic manipulation we can easily derive

$$p(\mathrm{B}|t_b^{NB}, o) = \frac{p(o|\mathrm{B}, t_b^{NB}) \cdot p(\mathrm{B}|t_b^{NB})}{p(o|t_b^{NB})}. \tag{6.3}$$

   Neglecting statistical dependencies between o and t_b^{NB} we can rewrite Equation (6.3) as

$$p(\mathrm{B}|t_b^{NB}, o) = \frac{p(o|\mathrm{B}) \cdot p(\mathrm{B}|t_b^{NB})}{p(o)}. \tag{6.4}$$

The conditional probability $p(o|\mathrm{B})$ is already measured (Table 5.4), $p(\mathrm{B}|t_b^{NB})$ is modeled by the lognormal distribution in Figure 5.4, and $p(o)$ can easily be measured from the recorded corpus (Section 4.1).

In order to generate eye blinks we design a FSM with three states NB_0, NB and B (Figure 6.13). While in NB_0 and NB the eyes are open, in B an eye blink is executed. Initially and after the execution of an eye blink the machine starts in the default state NB_0, which sets t_b^{NB} to one. After the initialization the state is changed from NB_0 to NB. Each time the current observation o is determined, a random number r^p is generated and the duration t_b^{NB} is increased. The machine switches the blink state, if r^p is smaller than the transition probability $p_{t_b^{NB},o}$. Since we do know the probability $p(\mathrm{B}_{t_b^{NB},o})$ of switching to state B given the observation o and duration t_b^{NB}, we can relate the states NB and B in Figure 6.13 with the transition probability $p_{t_b^{NB},o}$ as

$$p(\mathrm{B}_{t_b^{NB},o}) = p_{t_b^{NB},o} \cdot p(\mathrm{NB}_{t_b^{NB}-1}), \qquad \forall t_b^{NB} > 1 \tag{6.5}$$

with

$$p(\mathrm{NB}_{t_b^{NB}-1}) = 1 - \sum_{l=1}^{t_b^{NB}-1} p(\mathrm{B}_l). \tag{6.6}$$

Since the probability $p(\mathrm{B}_{t_b^{NB},o})$ is obviously equal to the conditional probability $p(\mathrm{B}|t_b^{NB},o)$, Equation (6.4) and (6.5) can be combined resulting in

$$p_{t_b^{NB},o} = \frac{p(\mathrm{o}|B) \cdot p(\mathrm{B}|t_b^{NB})}{p(o) \cdot p(\mathrm{NB}_{t_b^{NB}-1})} \tag{6.7}$$

giving the transition probability of the FSM for performing a blink given t_b^{NB} and o (Figure 6.13). If the state B is entered, initially the duration t_b^B of the eye blink is determined. After the blink, the FSM switches to the default state NB_0.

An example of generating the animation path for a given spoken output is illustrated in Figure 6.14.

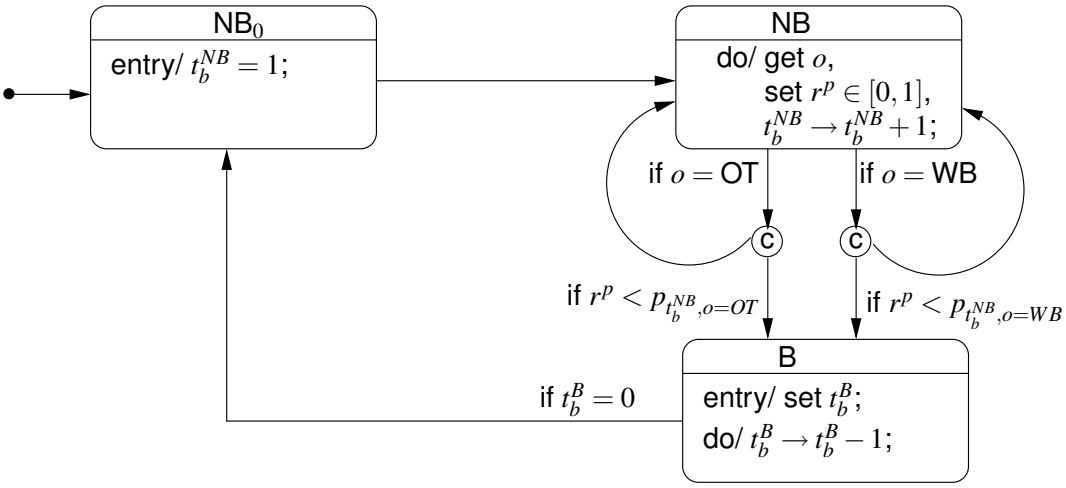

Figure 6.13: In talking mode a FSM with three states NB_0, NB, and B generates eye blinks. The transition probability $p_{t_b^{NB},o}$ from the state NB to B depends on the duration t_b^{NB} and current observation o.

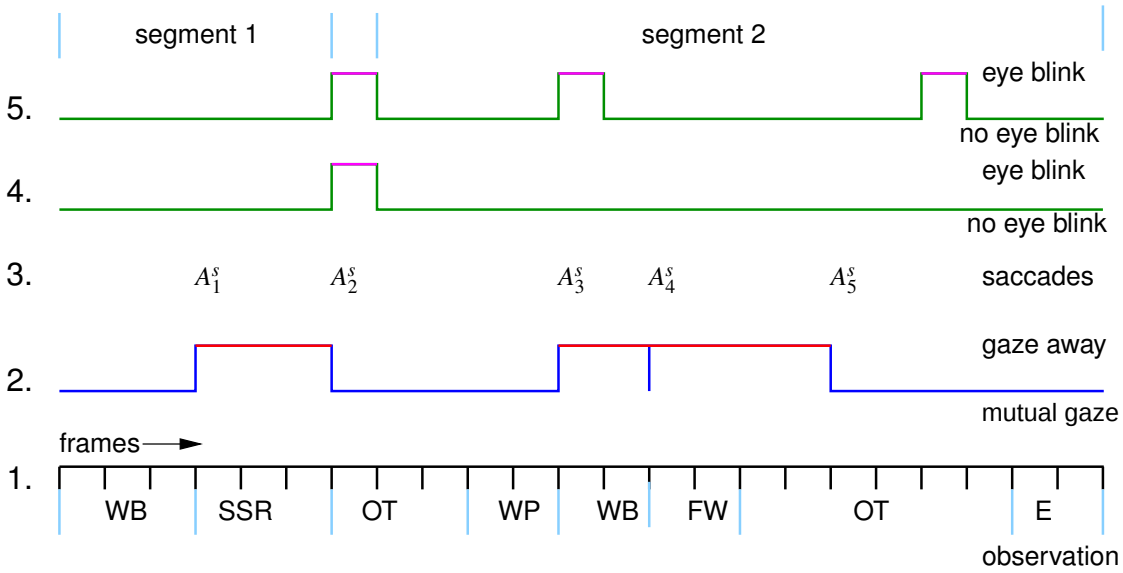

Figure 6.14: An example illustrates the generation of an animation path: 1. Each frame is labeled with its corresponding observation o extracted from the spoken output. 2. In order to generate gaze patterns, each frame is labeled with its corresponding state mutual gaze or gaze away. In this example first a single and afterwards two consecutive gaze shifts are executed. For better illustration, the gaze shifts in MG are not depicted. 3. The magnitude A^s and direction of saccades are generated. In the example five large saccades are executed from which A_4^s is a second consecutive saccade. 4. Eye blinks, which are simultaneously executed with a gaze shift, are determined in this step. In the example one eye blink is executed while performing a gaze shift. 5. Finally, additional eye blinks are added to each segment.

7 Rendering Engine

Since modeling human eyes is a difficult task, a sophisticated rendering engine needs to be designed (Figure 1.4). The iris contains specular reflections that need to be correctly modeled to achieve life-like looking eyes. In the image-based approach, however, the position of the specular reflections depends on the head's position in the recorded sequence. Hence, eye images cannot be simply normalized and rendered in a different position. Therefore, a rendering engine is developed which combines a 3D model and image-based rendering and is based on [30]. Note, that in [7] an advanced model to render eye movements is described, which we do not consider in this work, since our work focuses on the ECU and not the rendering engine itself. Moreover, we obtain satisfying results with our developed rendering engine.

In order to animate a talking-head, the following data has to be initially prepared: The eye globe is modelled by a half sphere. We use a high-resolution image of the human eye as texture for the globe. The image is post-processed by image processing software in order to delete specular reflections and complete the eye texture. A more detailed description of the precise anatomy of the eye globe and characteristics of the eye texture are given in [129, 88, 52]. Moreover, textures with specular reflections are generated. The eye socket and eyelid are modeled by a number of images stored in a database in which the person executes a blink. The surface of the eye area including the eye socket is approximated by a 3D eye model, which is acquired by a 3D laser scan (Section 3.5).

The eye animation is rendered in two steps: Firstly, the eye globes, which are synthesized by texturing half spheres, are rotated according to the eye control parameters. Afterwards specular lights are added at the appropriate positions on the eye globe by taking the eye pose and virtual spot light positions into account. The eye globes are combined with the eye socket image, which is retrieved from the database according to the eye control parameters. Different durations of eye blinks are generated by repeating or removing images from the recorded blink in the database. The rendered image is denoted as eye sample (Figure 7.1).

Secondly, image rendering overlays the eye sample over a background video sequence by warping the sample into the correct pose. In order to conceal illumination differences between an image of the background video and the eye sample, the samples are blended in the background sequence using alpha-blending (Figure 7.2).

Note, that the rendering engine processes 25 frames per second on a modern desktop workstation. The computational effort of eye control unit (Section 6) is negligible small, since the ECU consists only of simple FSMs. Therefore, real time animations can be generated.

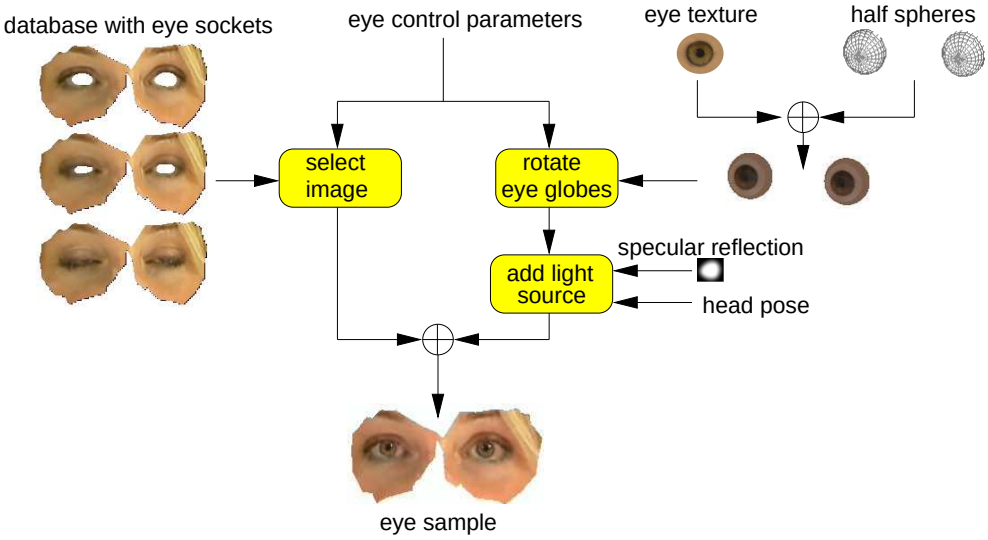

Figure 7.1: The eye control unit provides the control parameters to rotate the eye globes, which are modelled by half spheres with eye texture. Specular reflections are added to the eye globes by considering the position of the eyes, head pose and virtual spot light. According to the control parameters the eye socket image is selcted and combined with the eye globe.

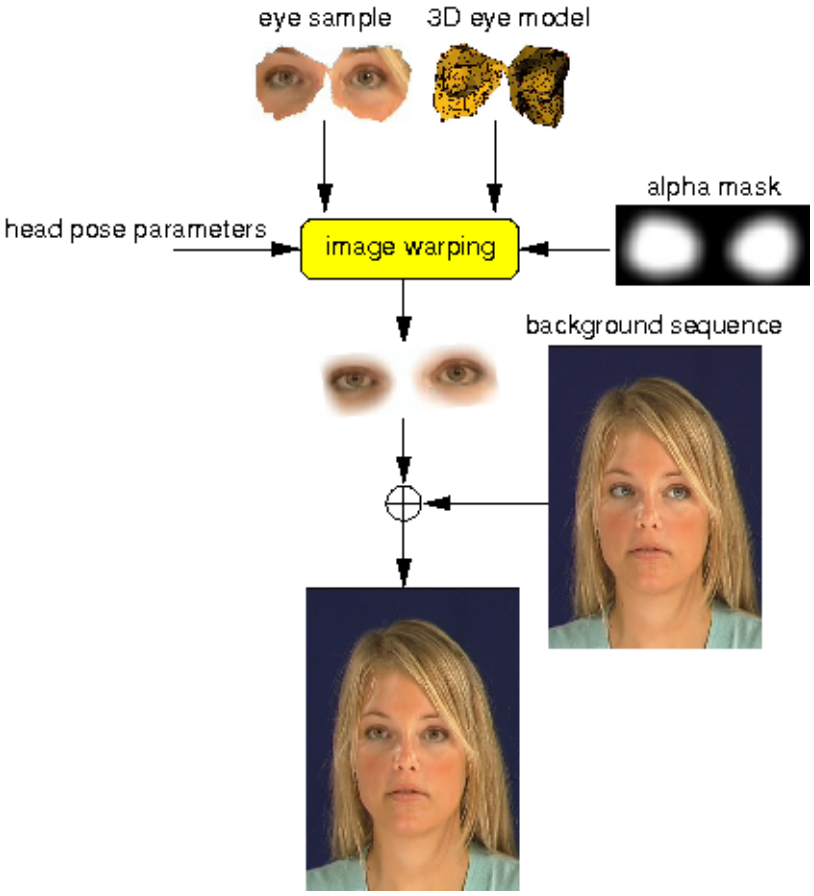

Figure 7.2: Eye sample is combined with the background sequence by warping the sample into the correct head pose and alpha blending.

8 Results

In the first part (Section 8.1) the results of the extended motion estimation algorithm versus a reference method are described. In the second part (Section 8.2) the conducted subjective test is described. This test measures the ability to distinguish between our animations, a reference method and recorded video clips.

8.1 Head Motion Estimation

The designed motion estimation algorithm of Section 3.5 is able to accurately estimate the head pose as required by image-based animation systems. Samples of smooth mouth animations in which the head pose is estimated by our designed algorithm are on our web site[1]. We evaluated the described motion estimation algorithm using real sequences, which are 8 bit gray images at PAL resolution (880×720 pel). The sequences are recorded by a Thomson Viper HD Camera in progressive mode with 60 Hz. In each of the sequences, the width of the face averages 300 pel. The face model is adapted to fit the geometric shape of the human head and is positioned in the initial frame before the motion estimation is performed.

8.1.1 Tracking Markers

In order to evaluate the motion estimation, two markers are added to the face of the human subject. The markers with a radius of 3.75 mm are glued to the forehead of the human subject (Figure 8.1), since the subject does not wrinkle her forehead while speaking or varying her facial expression. Hence, both markers are not moved due to local motions. As markers we use either green or blue dots, because these colors are nearly complements of the color of the human skin. Hence, these markers can be accurately tracked with color segmentation in RGB space.

In order to evaluate the accuracy of the motion estimation algorithm, in the initial frame of the image sequence the centers of the markers are mapped onto the face model. Henceforth, these markers are notated as 3D virtual markers. The displacement error is the Euclidean distance between the center of the marker determined by color segmentation and the corresponding projected 3D virtual marker onto the camera target [57]. The displacement error is calculated in the unit pel.

[1] http://www.tnt.uni-hannover.de/project/facialanimation/demo/

Figure 8.1: Two blue markers glued to the forehead are tracked and used as reference for the motion estimation.

For the analysis of the motion estimation algorithm the markers need to be reliably tracked with a high accuracy throughout the sequence. Hence, we evaluate the accuracy of the color segmentation by generating a synthetic image sequence. The synthetic sequence, which is generated by OpenGL, has 300 images and the error standard deviation of the estimated and true dot center is 0.15 pel. There are two reasons for this displacement. Firstly, we cannot fully control how OpenGL renders a 3D scene. Furthermore, we induce a systematic measurement error, since the projection of a circle's center does not coincide with the barycenter of the corresponding ellipse. Therefore, the barycenter of the marker is not the same as the barycenter of the ellipse. The systematic measurement error δ_e between the projected center of the marker and its measured barycenter is

$$\delta_e = \left| \frac{f r_e^2 sin(\sigma_e) cos(\sigma_e)}{(f+z_e)^2 - r_e^2 sin^2(\sigma_e)} \right|. \tag{8.1}$$

with the observation angle σ_e, radius r_e, focal length f and distance z_e between the principal point and center of the marker. The derivation of Equation (8.1) is formulated in Section C. In Figure 8.2 the relationship between the error δ_e with respect to the parameters z_e and σ_e are depicted. The largest error δ_e is given at an observation angle of σ_e equal to 45° and the shortest distance z_e. Here the maximum calculated error is less than 0.012 pel and therefore does not severely contaminate our measurements. Hence, this systemetic error is negligible.

Before we start the motion estimation, the markers and their associated illumination effects have to be removed from the image sequence. For this an algorithm is used, which replaces each pixel identified as being part of a marker with human skin. A smooth transition between the border of skin and marker is achieved by weighting different neighbors classified as skin.

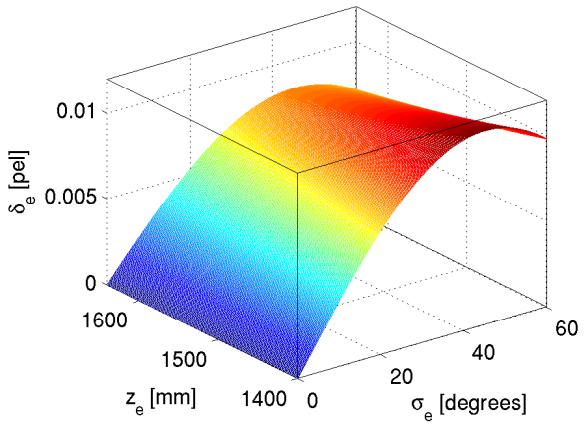

Figure 8.2: The error δ_e between the measured and true barycenter of a projected marker with respect to the observation angle σ_e and distance z_e. The focal length f is equal to 32.95 mm and the radius of the marker r_e equal to 3.75 mm. The largest error occurs with an observation angle θ_e equal to 45° and the shortest distance z_e between the principal point and barycenter of the marker.

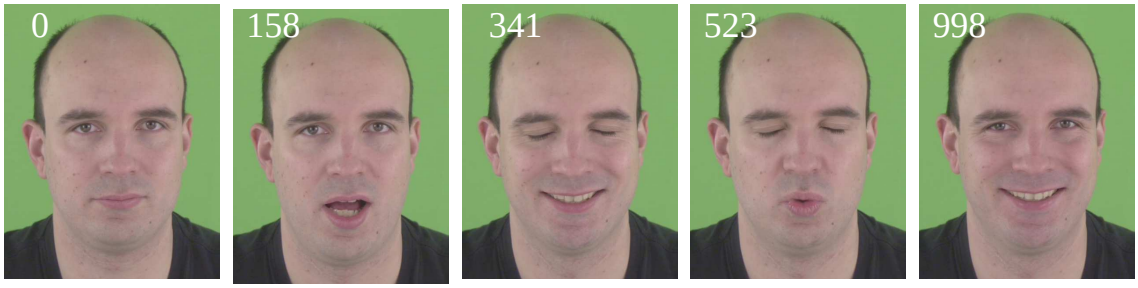

Figure 8.3: Sample images from the second sequence, in which the speaker continuously varies his facial expression.

8.1.2 Weighting Feature Points

In two similar image sequences in which the recorded person performs many different facial expressions while talking (Figure 8.3), the head pose is exemplarely estimated to present the improvements of the proposed weights over the reference method [144]. In these sequences a large number of feature points do not satisfy the rigid head motion model of Equation (3.37) and therefore disturb the estimation. Hence, these features need to be identified as outliers. Both algorithms are identical except for the applied weights. Whereas the reference method uses Equation (3.62) to determine the weight ζ, our systems uses Equation (3.53). We set the following parameters in both methods: $c_G = 1.5$ and $\sigma_G = 128$ in Equation (3.44). Additionally, we set $c_1 = 1.2$ in Equation (3.52) of the reference method. In our method, we set the threshold ξ_d to 5.5 in Equation (3.47) and ξ_c to 12 in Equation (3.51).

The displacement errors of both sequences are presented in Figure 8.4. Analyzing the displacement error shows that our proposed method achieves mostly better results than the reference method in both observed image sequences. The average displacement error is reduced from 0.53 pel to 0.46 pel (Figure 8.4a) and from 1.32 pel to 1.14 pel (Figure 8.4b), which is equal to a reduction of 13% and 14%, respectively. In each sequence, the displacement error significantly increases one time due to very strong facial expressions while simultaneously quickly nodding. Nevertheless, the maximum peaks are greatly reduced e.g. a reduction from 2.0 pel to 1.5 pel. Hence, the accuracy of the head pose even in challenging scenes is significantly improved by our proposed method. The improvements are due to the additional two sub-weights. One sub-weight gives the user the opportunity to apply a priori knowledge of the head movements by defining a maximum displacement between consecutive frames. Thus, features, which exceed this displacement, are identified as outliers. The other sub-weight considers the spatial relation of the residuals of neighboring feature points. If the residual of a feature strongly varies from its neighbors, then it is identified as an outlier. Hence, facial areas with large residuals are automatically segmented and not considered in the estimation.

8.1.3 Update Texture Information

In order to show the performance of our extensions, three image sequences of two different persons are analyzed. Our extensions are first compared with never and always performing a texture update and afterwards with the reference method [144]. In Figure 8.5 the motion estimation results, from the first sequence with over 800 images, are presented. Here the quality of the algorithms with a single reference frame versus the proposed texture update is visible with the naked eye. In the middle row the pose of the face model is not correct, whereas in the last row the head seems to be glued to the face.

Unfortunately, the error accumulation of always updating texture information is not as obviously visible by the naked eye as never updating. Hence, in Figure 8.6 the displacements errors of the three methods are shown. The displacement errors averaged over 800

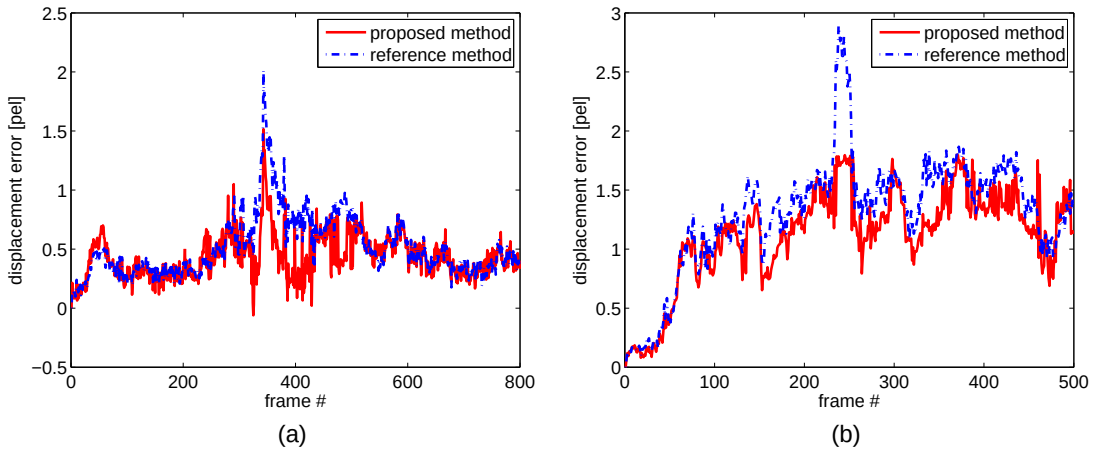

Figure 8.4: Comparison between the proposed strategy (solid red curve) versus reference method [144] (blue dashed curve) of weighting feature points in two image sequences.

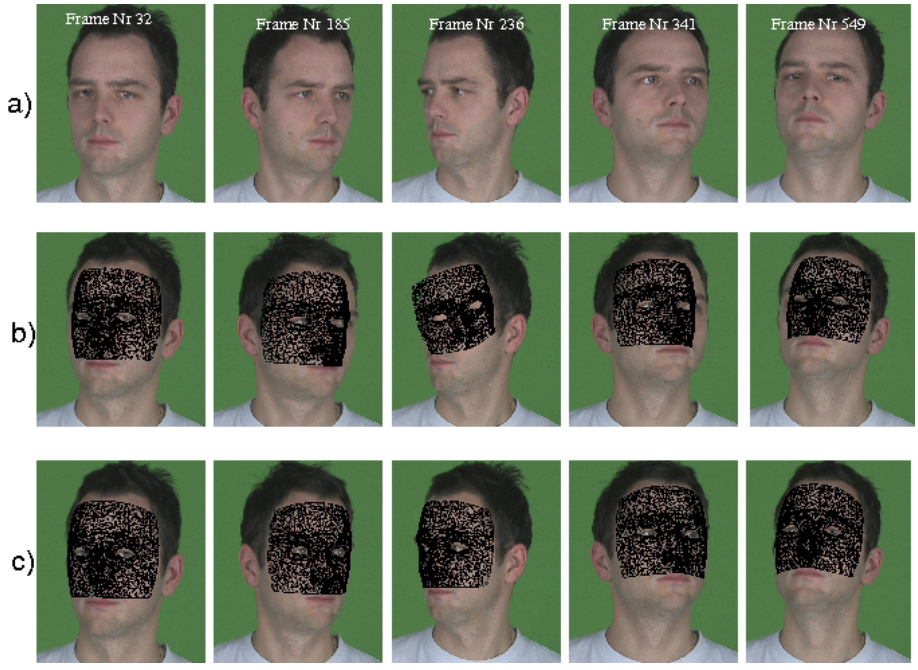

Figure 8.5: In (a) only the original frames are plotted, while in (b) and (c) the face model is plotted in the estimated pose of the head. In (b) only one reference frame is used to estimate the pose of the head. Large out of plane rotations result in large errors. In (c) the proposed algorithm is displayed and the accurate pose is estimated.

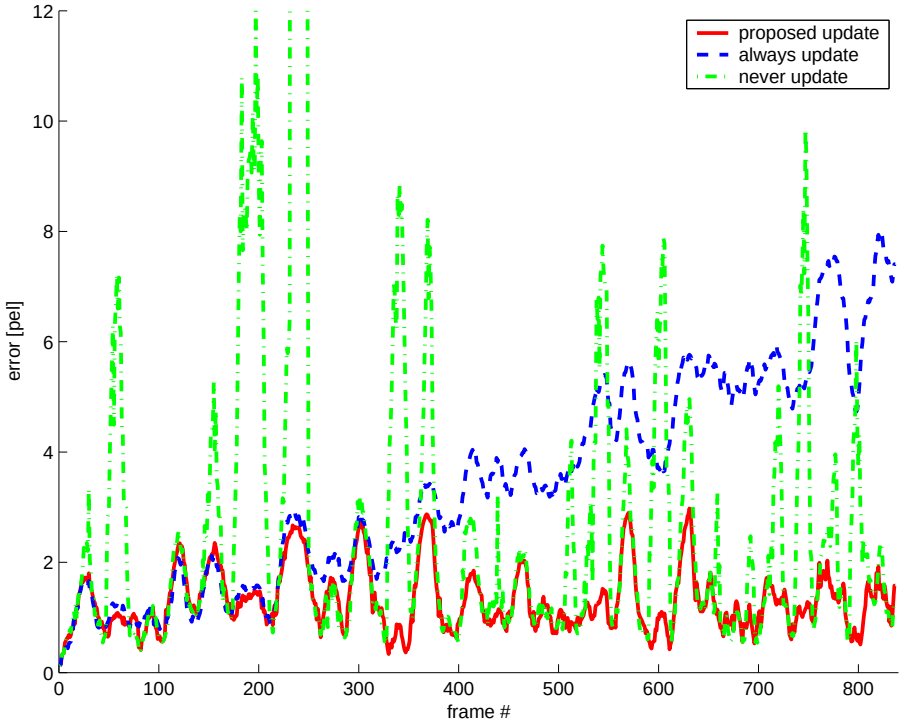

Figure 8.6: Comparison of the displacement error between the proposed strategy (solid red curve) of updating texture information versus never (green dashed-dotted curve) and always (blue dashed curve) updating.

frames of the proposed strategy, always and never update are 1.3 pel, 3.4 pel and 3.2 pel, respectively.

Obviously, the proposed update significantly reduces the error with respect to never or always updating the texture information as already assumed in the introduction. If the texture information is never updated then the error highly increases by large out-of plane rotations. If the texture is always updated, then the error accumulates over the sequence. Hence, the error is large at the end of a long sequence. Therefore, these two approaches do not lead to satisfying results and an efficient texture update is necessary.

In order to evaluate the improvements with respect to the reference method [144], we calculate the head poses in the second and third sequence, each with nearly 800 frames, with the reference as well as our method and calculate the displacement errors. In both sequences the recorded person performs various head movements and at the same time varying facial expressions. For a better comparison both methods use the same weights as described in Equation (3.62) with the following parameters: $c_G = 1.5$ and $\sigma_G = 128$ in Equation (3.44) and $c_1 = 1$ in Equation (3.52). Thus, only the approach of updating the texture information of the reference frame differs. In order to get a good set of parameters, we tested different sets with the second sequence and calculated the average displacement

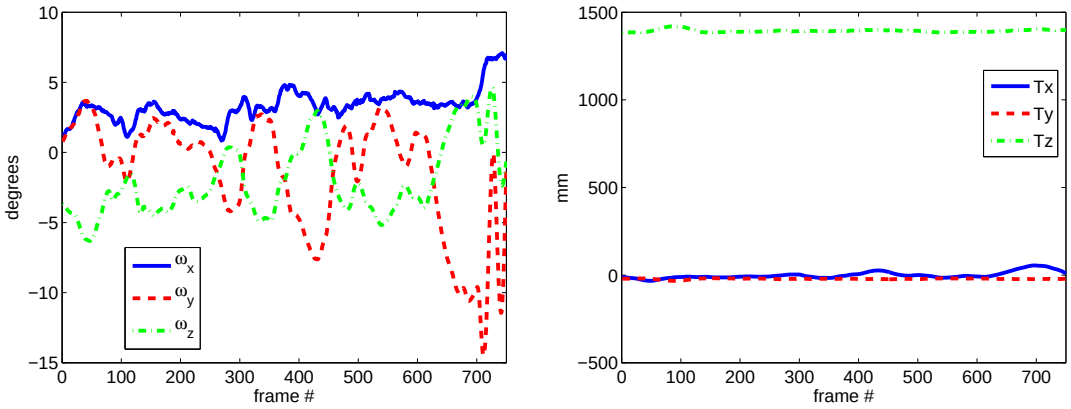

Figure 8.7: Head motion parameters of the second sequence as estimated with our proposed method.

error. The following parameters, which were also used to calculate the motion parameters in the third sequence, caused the smallest displacement error: $c_3 = 0.1$ in Equation (3.56, $c_4 = 5.8$ in Equation (3.57) and $c_2 = 3.2$ in Equation (3.62). In the reference method, the distance between stored reference frames and the current frame is a combination of the weighted difference of the rotation angles and translations. The weights are selected so that our proposed method and the reference method generate around the same number of reference frames while estimating the head motion in the second and third sequence.

The head motion parameters of the second sequence as estimated by our method are exemplarily depicted in Figure 8.7. The displacement errors of the second and third sequence of the reference versus our proposed method are presented in Figure 8.8. Analyzing the displacement error shows that our proposed method shows always better results than the reference method in both observed image sequences. The peaks of the displacement error are due to short and fast head nods while simultaneously varying facial expressions. Hence, estimating the accurate head pose is difficult due to the large number of outliers. In the second sequence, the average displacement error is reduced from 0.53 pel to 0.41 pel, which is an improvement of nearly 23% (Figure 8.8a). The maximum error of 2.4 pel is reduced to 1.7 pel, which is a reduction of nearly 30%. In the third sequence the average Euclidean distance is reduced by 28% from 0.44 pel to 0.32 pel (Figure 8.8b). The average displacement error is much smaller in the second sequence than in the first, since slower and less head movements are performed. Furthermore, the facial expressions are only slightly varying in the third sequence.

The error reduction of our proposed method is mainly due to five reasons. Firstly, with respect to the reference method, we only update a reference frame due to out-of-plane rotations. Hence, if both methods use the same amount of reference frames, then we have selected the better images as reference frames. Secondly, we do not update the reference

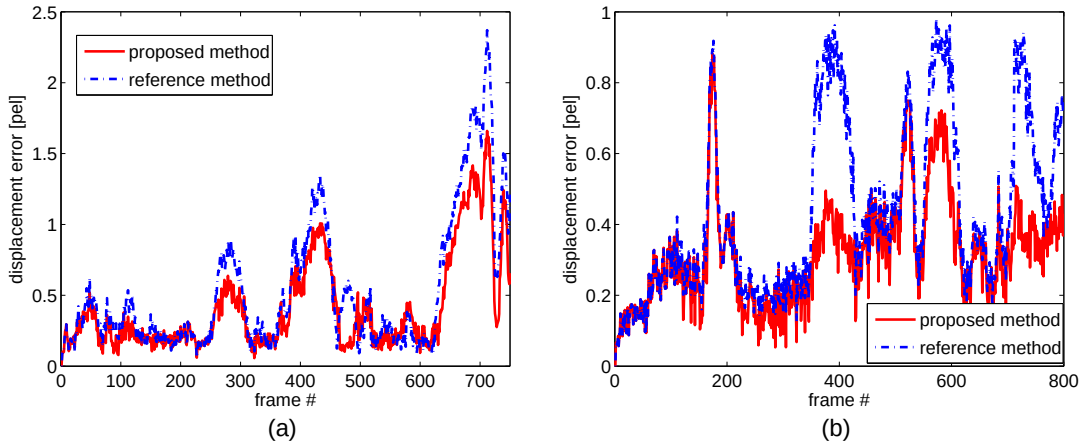

Figure 8.8: Comparison between the proposed strategy (solid red curve) versus reference method [144] (blue dashed curve) of updating texture information. Whereas in the second sequence (a) exaggeratedly strong and fast head movements are performed, in the third sequence (b) only typical head movements as performed by a news reader are executed.

frame after estimating the head pose as in the reference method. Each update induces a small drift, even if a re-alignment is performed laster on. Thirdly, we re-register to the most previous reference frame if the condition of Equation (3.59) is satisfied, while the reference method re-registers to the reference frame closest to that of the current frame. Hence, our approach allows to most suitable rectify the accumulated error. Fourthly, the new reference frame is a combination of the previous one and the current image, so that the valuable texture information of the previous reference frame is used for further calculations. Fifthly, additional feature points are added in our approach at the edge of the face model if necessary. This is important, since a sufficient amount of appropriate feature points are required to accurately determine large out-of-plane head poses.

8.2 Subjective Tests of Eye Animations

The quality of the synthesized eye animations of our models is evaluated by a subjective test, which is based on human judgments of various aspects of experienced stimulus material [72]. Under subjective evaluation Mullin et al. [99] understand "Any information that originates from users, experts or observers can be considered to be subjective data".

The quality of the synthesized animations of our eye animation system is evaluated by measuring a participant's ability to tell the animated from the real video. This type of evaluation is also denoted as "Turing Test" [132]. Next the set-up of the subjective test is briefly explained.

Figure 8.9: (a) English female speaker. (b) German male speaker.

Altogether 25 people whose age ranges between 17 to 59 years participated and who reported to have normal hearing and normal or corrected-to-normal vision. The number of professional participants is restricted to 20% in order to ensure better correlation of results with potential users. Participants are considered professionals if they have experience and knowledge in information technology or video processing. Participants with reluctant attitude towards technology were left out of the test in order to ensure to have participants interested in our type of research.

The general viewing conditions are set as recommended in ITU-R BT.500-11 and ITU-T P.910 [72, 73]. The videos with a resolution of 480×384 pel are MPEG-1 encoded with best image quality and displayed with the Windows Media Player. Videos of eye animations with our proposed system can be found on our web site[1]. Two types of test material are presented to participants: an English female and a German male speaker (Figure 8.9). Altogether 8 different utterances with duration between 2 s and 22 s are prepared. These clips are not used for previously training the models of the eye control unit. In the video clips, the speaker on the one hand utters typical sentences used by a virtual operator in a dialog system and on the other hand the speaker describes his new apartment. Eye animations are generated by using the spoken output and the speakers head movements as input parameters to the eye animation system (Figure 6.1). Only eye movements and blinks are varied, because we are only focusing on this part. We only test the talking mode, since we focused on this mode. Each test session begins with an introduction of the purpose and goals of the experiment and instructions are given to the participants.

Pairs of real and synthetic image sequences of the same utterance are presented as stimuli, one immediately after the other in randomized order. We compare the reference method [85] (Type II), which is extended by the eye blink model for Listening mode for generating blinks, as well as our proposed method (Type III) with respect to the original video (Type I). The participants' task is to tell the order of the presented real and animated

[1] http://www.tnt.uni-hannover.de/project/facialanimation/demo/index.html

Table 8.1: Responses to pair presentations (sample mean $\bar{X}$, standard deviation S and p-value). Type I: original, Type II: reference method [85], Type III: proposed method.

	Type I versus Type II			Type I versus Type III		
	$\bar{X}$	S	p	$\bar{X}$	S	p
correct answers	0.78	0.14	$< 10^{-6}$	0.54	0.19	≈ 0.32

videos. If a participant cannot tell them apart, he has to guess. Hence, if a participant is never able to distinguish between real and animated video, then the percentage of correct answers is close to chance level (50%).

The average score of correctly identifying the order of the real and animated videos is presented in Table 8.1. Participants correctly identified the order of sequences of Type I and II with 78%. Hence, most participants are able to distinguish between real and synthetic sequences. The average score of correctly identifying the order of original video and our proposed method is only 54%, which is close to chance level.

For both pairs, we propose a hypothesis about the relation between real and synthetic sequences. Our null hypothesis states that the correctly identified orders are chance level, hence $H_0 : \mu = 0.5$. The alternative hypothesis is that it is not chance level $H_1 : \mu \neq 0.5$. Since the outcome of our subjective test is dichotomous, the null hypothesis is tested with the binomial distribution. Its distribution function is given by the formula

$$b(N_b, p_b, j) = \binom{N_b}{j} p_b^j (1 - p_b)^{N_b - j} \tag{8.2}$$

where $\binom{N_b}{j}$ is a binomial coefficient and p_b is the probability of success on each experiment. Following we set the criterion of significance α to 0.05, which equals a 95% confidence interval. Calculating the binomial distribution and computing the p value ($p < 10^{-6}$) between Type I and II indicates that p is located within the area of rejection. Therefore, the null hypothesis is rejected. The computed p value ($p \approx 0.32$) of the binomial distribution of Type I and III is not located within the area of rejection (Table 8.1). Thus, the null hypothesis is retained. While participants were often able to discriminate between the original and reference method, they were usually not able to distinguish between the original and our proposed model.

In Table 8.2 the correct answers and the duration of each video clip are presented. The number of correct answers of a clip does not increase by an increase of its duration. Figure 8.10 shows the binomial distribution of the subjective tests of Type I and III of the 25 participants ($N_b = 25$). All sequences are located inside the area of retaining H_0.

The subjective test was followed by an open interview on the participants' evaluation criteria during the test. In the following the differences in quality are analyzed in more detail.

Table 8.2: Average number of correct answers and the duration of each video clip (Type I: original, Type III: proposed method).

	Type I versus Type III							
sequence	1	2	3	4	5	6	7	8
duration [s]	7	19	18	7	7	7	22	21
correct answers	0.62	0.42	0.52	0.62	0.64	0.68	0.44	0.54

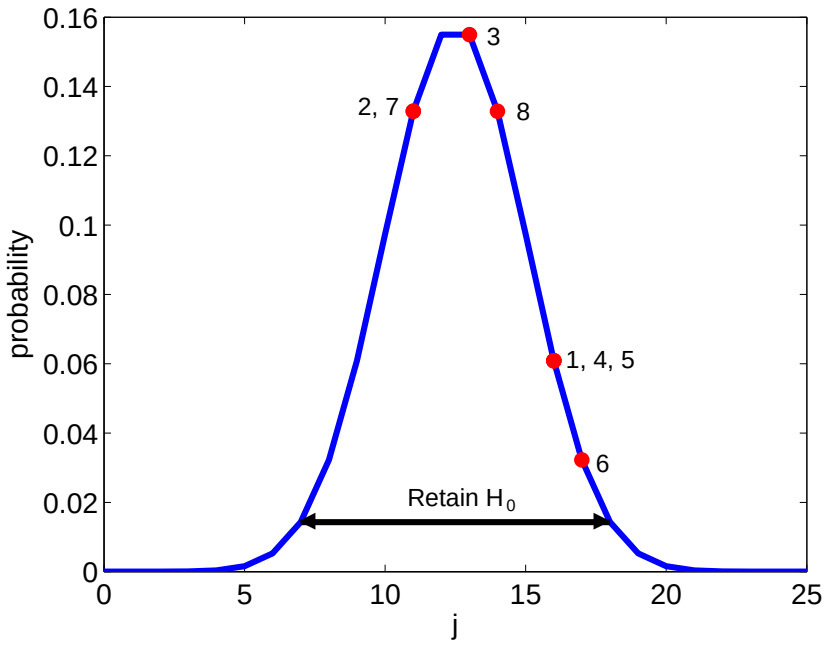

Figure 8.10: The binomial distribution of the subjective tests of Type I and III. The number of participants is 25 ($N_b = 25$), $p_b = 0.5$ and j is the number of correctly identified videos (Equation (8.2)). The 8 video samples are marked as red dots on the distribution and are located within the area of retaining H_0.

Our model of generating saccades is based on the work of the reference method, but improved by taking Listing's Law, head tilts and eyelid movements into account. In comparison to our model, participants reported that some saccadic eye movements did look artificial in the reference method. Whereas eyelid movements had most impact, head tilts had some, Listing's Law was not mentioned. Nevertheless, our suggested extensions improve the saccadic eye movements.

The coupling between large saccades and eye blinks as integrated in our model does also improve the animation. Some participants reported that it appears unnatural to perform a large saccade without simultaneously executing a blink as done in the reference method.

Some participants disliked the observed jerky eye movements in the reference method. These movements occur if a large saccade with a short duration in gaze away is performed. We prevent these movements by refining the mutual gaze state. Thereby, we distinguish between small saccades performed in mutual gaze and saccades to shift the gaze from mutual gaze to gaze away and vice versa. These saccades have very different characteristics. For instance, eye movements in mutual gaze have a shorter fixation duration than in gaze away. Participants, however, did not observe that the talking head is varying his POR within mutual gaze. Hence, this aspect does not seem to improve the quality. On the other hand if the image resolution is increased, this aspect may be realized.

In general, the timing of executing eye blinks as well as gaze shifts was preferred in our animations than in the reference method. For instance, some participants expressed that it seems inappropriate to perform a shift from mutual gaze to gaze away while simultaneously emphasizing a word.

The eye blink model was not mentioned by participants during the interview. Therefore, we showed 15 participants the same video sequences as before. We varied, however, the eye blink patterns. One time these patterns are generated by a simple model (listening mode) and one time by our designed model in talking mode. After presenting both videos participants were asked to select the preferred eye animation. 13 out of 15 participants, which equals 87%, selected the model used in talking mode. Hence, we conclude that the designed eye blink model in talking mode contributes to the overall animation quality, too.

As presented in Table 8.2 the percentage of correct answers is not 0.5 for all sequences but varies between the different clips. Since the number of participants is uneven the correct answers cannot be equal to 0.5. We analyzed video clip 6 in more detail, since this clip could be best descriminated between original and animation. For the analysis we compared clip 6 with clip 3, which was hard to discriminate. However, neither the rendering quality nor the type of eye movements caught our eye. Furthermore, participants did not criticize the video samples created by our model during the interview. Therefore, we assume that the variation is due to the small sample size.

Summing up, participants of the subjective test were not able to discriminate between original clips and eye animations created by our ECU. At the same time an increase of the duration of a clip does not result in a raise of the correct answers. The main improvements

of the designed ECU are the automatic steering of the eye gaze and blink with spoken language as well as considerung the coupling between gaze shifts and blinks. Furthermore, saccadic eye movements are improved by considering Listing's law, head tilts and eyelid movements. The traditional model of gaze shifts is refined by considering short fixations within mutual gaze. Finally, a new eye blink model is presented which takes temporal as well as observations into account to create eye blinks.

9 Conclusion

In this work, we have presented a system that is capable of synthesizing video-realistic eye animations, since non-verbal communication has been mainly neglected by the latest image-based animation systems. However, incorporating non-verbal cues in facial animation systems is essential to create video-realistic animations. Therefore, the main emphasis of this work is the designed ECU. A minor issue of this work was to design an algorithm, which allows to accurately estimate the head pose. Algorithms used by other image-based systems, however, encounter problems to accurately estimate the head pose. But without accurate head poses, smooth animations cannot be created by an image-based system.

Head Pose Estimation

We designed a 3D model-based motion estimation algorithm based on the work of Xiao et al. [144], which also serves as the reference method. This algorithm allows to accurately measure the head pose. A weak point analysis of their method indicated two main issues to improve. Firstly, the detection of outliers is very important so that local movements in the face do not influence the head pose estimation. Secondly, the update of texture information is necessary to determine out-of-plane rotations. For evaluating the new methods we estimate the head pose in real image sequences in which markers are glued to the forehead of the recorded person. As error measure serves the displacement error, which is the Euclidean distance between the image location of the center of markers as determined by a 2D color segmentation and the predicted image location of the center of the markers by the model-based algorithm.

Weighting feature points is a robust method to detect outliers and assign high weights to "good" feature points, while outliers only obtain a small weight. In our method the weight consists of several sub-weights, in which one binary sub-weight accounts for visibility of features in the current frame. One sub-weight decreases with an increasing residual error. Hence, a feature point with a large luminance difference receives a small weight. Another sub-weight, which is reduced after each iteration, is related to the spatial gradient. A large gradient obtains a large weight. These two sub-weights are summed as proposed in the reference method [144] and consequently compensate each other. Hence, features with a large gradient e.g. corners also contribute to the estimation. One sub-weight gives the user the opportunity to define a maximum displacement of features between two consecutive frames. Hence, a priori knowledge about the scene, e.g. the recorded person performs only slow head nods, can be integrated and improve the estimation. Finally, one last sub-weight takes the spatial relation between neighboring feature points into account to

identify outliers. For this, the ratio of the luminance difference of each feature point with its neighbors is calculated. If this ratio is larger than a threshold, the corresponding feature is an outlier. Through this weight facial areas, which do not satisfy the rigid motion model, are automatically identified.

In two image sequences, one with a news speaker as well as over 1000 frames and the other with over 500 frames the displacement error is reduced from 0.53 pel to 0.46 pel and 1.32 pel to 1.14 pel, which is an improvement of nearly 13% and 14%, respectively. In addition, the error peaks were reduced, too. The error reduction is due to the two sub-weights taking the maximum displacement of features between two frames and the spatial relation between neighboring feature points into account. These sub-weights identify many outliers.

Our proposed extensions of the motion estimation algorithm to automatically update the texture information of the reference frame works as follows. Since in our application we can create a nearly perfect ambient illumination and diffuse reflection of the recorded person, the texture variations are due to out-of-plane rotations. These rotations are described by yaw and pitch, which can be also understood as spherical coordinates marking a position on the unit sphere. Hence, each frame, which becomes a reference frame, is characterized by its position on the unit sphere. Two conditions need to be satisfied to create a new reference frame. Firstly, the Euclidian distance between the current frame and reference frame on the unit sphere needs to be larger than a threshold. Secondly, the luminance difference of feature points between the current and reference frame may not exceed a threshold. Only under these circumstances we can verify that the head pose is accurately measured in the current frame. A new reference frame is created by combining the texture information of the current and previous reference frames. If necessary, additional feature points are added to the face model. A re-alignment to a previous reference frame is executed, if the position on the unit sphere of the current frame is close to the position of a previously stored reference frame. Then the entire texture is updated.

In the experimental analysis, we first proved the assumption as stated in the introduction that a texture update is necessary, since never or always updating the texture information induce a large displacement error or drift, respectively. Comparing the reference method with our proposals in two image sequences shows, that the displacement errors are reduced from 0.53 pel to 0.41 pel and 0.44 pel to 0.32 pel by our method, which is an improvement of nearly 23% and 28%, respectively. In addition, the error peaks are also significantly reduced. The error reduction is mainly due to an optimized selection of the reference frames by taking the recording set-up into account and that a new reference frame takes previous as well as the current texture information into account.

Eye Animation Systems

A novel image-based eye animation system consisting of a ECU and a rendering engine was developed. The designed ECU is based on eye movement physiology and the statistical analysis of recorded human subjects in a two-way conversation. For the statistical

analysis, two experiments are conducted in order to record typical eye and blink movements during a conversation. Audio as well as video processing algorithms to automatically analyze the recorded data were presented. In general, the studied speech processing algorithms can be used to automatically extract audio features in an eye animation system, as e.g. needed to read audio books by a talking-head.

The ECU consists of different models controlling gaze, eye blinks as well as eye movements. In the presented animation system two types of eye movements, saccades and vestibulo-ocular reflex, are executed. Saccades are executed to perform a gaze shift e.g. from mutual gaze to gaze away. The model to create saccades is mainly based on the work of [85]. In their model saccades are characterized by direction, magnitude, velocity and acceleration. The vestibulo-ocular reflex is performed in order to keep the image fixed on the fovea and compensate head motion. We improve these eye movements by three extensions in our designed eye movement model. We integrated Listing's law, which describes the three-dimensional orientation of the eye and its axes of rotation. Secondly, our model includes that head tilts induce a torque. Furthermore, our model couples eyelid movements with vertical saccades as noted in eye movement physiology.

We distinguish between listening and talking mode as suggested in literature. In listening mode, two independent models control eye blinks and gaze shifts, since gaze and blink patterns are statistically independent as presented in our work. Both models are realized as simple finite state machines (FSM). The durations of mutual gaze, gaze away and the duration between two consecutive eye blinks are modeled as lognormal distributions. In this thesis we focused on talking mode and designed one integrated eye blink and gaze model, because data analysis showed that eye blinks and gaze movements are coupled in this mode. In this model eye blinks and gaze shifts are controlled by spoken language, since our analysis revealed statistical dependencies between eye blinks and gaze movements with spoken language. While eye blinks mainly occur at word boundaries, gaze shifts are usually performed in thinking mode, e.g. indicated by a filling word. On the other hand if words are emphasized, the speaker usually looks to the interlocutor. Our approach allows to automatically generate appropriate gaze shifts and eye blinks to arbitrary spoken language. In detail, the integrated eye blink and gaze model gradually generates the eye control parameters by using the observations, which are extracted from the spoken language, as input. A FSM with three states consisting of one mutual gaze and two gaze away states generates the gaze patterns. Gaze shifts are determined by taking the conditional probability of performing a gaze shift given an observation and the current observation into account. After a gaze shift the model remains in the next state for a certain duration, which is modeled by a lognormal distribution, and afterwards a second consecutive shift may be optionally performed. In order to perform a gaze shift a saccade is executed. Hence, the parameters of these saccades are determined by the previously described model of eye movements. The magnitude of a saccade and the probability of performing a blink are related as we determined in an experiment. Thus, we are able to determine eye blinks, that are simultaneously performed with a saccade. Finally, additional eye blinks are added by a FSM, which considers the temporal dependency as well

as the dependencies to the spoken language. This FSM is also realized by three states and takes the duration since the last executed eye blink as well as the current observation into account to generate additional eye blinks.

While being in mutual gaze humans vary the point of regard across the face. This observation is included in our work by extending the traditional simple model of mutual gaze and gaze away. In mutual gaze, a second model generates small gaze shifts within mutual gaze. Our observation is also important, since the fixations after the execution of a gaze shift varies. After performing a gaze shift from mutual gaze to gaze away or vice versa, the eyes remain for a longer duration in the new position. If small gaze shifts within mutual gaze are executed, the eyes only remain for a short fixation in the new position. The distinction between these two types of gaze shifts allows to better model human eye movements. That is because jerky eye movements may occur, if large saccades with a short duration in mutual gaze are executed.

Eventually, the ECU also takes head movements into account, since gaze shifts and head movements are coupled. For this, we presented an algorithm, which can automatically segment head movements. If these segmented movements are larger than a threshold, a saccade is executed.

Summing up, our designed ECU has the following innovations with the respect to the state-of-the-art:

- gaze shifts and eye blinks are fully controlled by audio features in talking mode, which allows to automatically create eye animations to arbritrary spoken output
- one integrated model steers eye blinks and gaze shifts in talking mode
- saccadic eye movements are improved by considering Listing's law, head tilts as well as the coupling between vertical saccades and eye blinks
- mutual gaze state is refined by a model taking short fixations within this state into account
- the eye blink model considers the duration until the last executed eye blink as well as the current audio features to generate eye blinks

The rendering engine synthesizes the eye animation with the provided control parameters from the ECU. The key idea of the designed rendering engine is the combination of 3D eye globes with the eye socket and eyelid, which are image-based modeled. In this way, the position of the pupil can be fully controlled and specular reflections can be added to the iris in the rendering process.

We conducted a subjective test in which the quality of the eye animation is evaluated by the participant's ability to discriminate between real and animated videos. In this test, pairs of real and animated sequences of the same utterance, which are either created by the reference method [85] or our designed ECU, are presented as stimuli. The analysis

of the test reveals that participants correctly identified the real video with 78% and 54% of the reference and our proposed method, respectively. Hence, as a null hypothesis we state participants are not able to distinguish between real and animated sequences. Testing the null hypothesis with the binomial distribution indicates that this hypothesis is rejected with respect to the reference method, but retained with our proposed method. The probabilities of all video samples of our model are located within the area of retaining the null hypothesis. In the interview following the subjective test not a single participant criticized the video samples generated by our proposed ECU. Moreover, the correctly discriminated video clips are not coupled to their duration. Since the correctly identified video samples are close to chance level, the null hypothesis is retained and participants did not complain about video samples, we conclude that the new eye animation system creates video-realistic eye animations for a talking-head, which has not been achieved before.

The ECU is not limited to image-based systems, but may be also used in facial animations based on 3D models. Furthermore, the designed ECU gives the opportunity to transfer eye parameters between persons. Since the measured eye movement distributions highly vary between individuals, a talking-head may appear more or less lively depending on the selected data. The computational effort of the ECU is negligible, since the models are implemented as FSMs. The rendering engine is capable to effortlessly render a complete image-based talking head with 25 frames per second, so that real time animations as required in dialog systems can be generated.

Applications of a talking-head as a personal adviser in an e-commerce store require video-realistic animations. Until now video-realistic mouth and eye animations can be generated by our image-based animation system. Both features are very important for a conversation. Nevertheless, further features e.g. facial expressions or head movements need to be integrated to the image-based animation system. On the other hand, the quality of TTS-synthesizers need to be improved so that humans cannot distinguish between synthesized speech and a human voice anymore.

A Rejection Sampling

In order to model arbitrary 2D distributions like the lognormal distribution rejection sampling is used [136, 114], which belongs to the Monte Carlo Methods. Rejection sampling is a method to draw independent samples from an arbitrary probability distribution. Imagine the discrete probability distribution $f_p(x_k)$ is too complicated to sample from it directly. We assume that we have a simpler discrete proposal probability distribution $Q_p(x_k)$, which holds

$$Q_p(x_k) \geq f_p(x_k), \qquad \forall x_k, \tag{A.1}$$

from which we can generate samples. For better illustration we assume that $f_p(x_k)$ is a 2D distribution and $Q_p(x_k)$ has a rectangular shape (Figure A.1). Then, a sample with two random numbers (r_1^p, r_2^p), which are statistically independent, uniformly distributed and located within $Q_p(x_k)$, is generated. This sample can be viewed as selecting a point located within the two-dimensional plane of $Q_p(x_k)$. Finally, we evaluate $f_p(x_k)$ and either accept or reject the sample (r_1^p, r_2^p) by comparing the random number r_2^p with the function value $f_p(r_1^p)$. If $r_2^p > f_p(r_1^p)$ then the sample is rejected. Otherwise it is accepted. For instance, if we model the lognormal distribution in Figure 5.4a and assume the sample (r_1^p, r_2^p) to be accepted, then r_1^p is equal to the duration until the next eye blink is performed. Hence, arbitrary distributions can be easily modeled using rejection sampling.

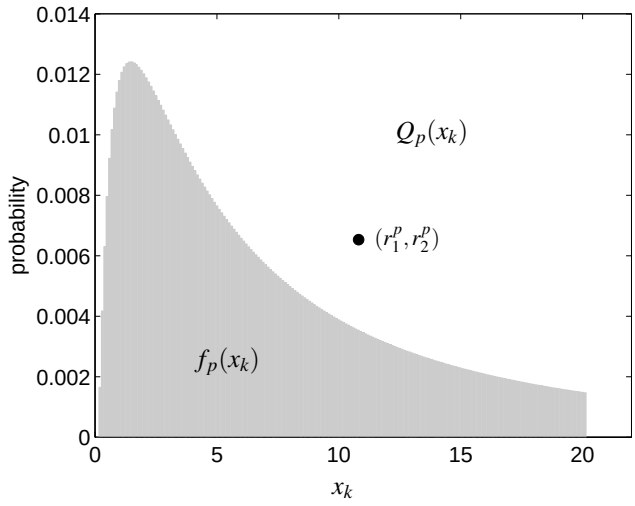

Figure A.1: Rejection sampling is based on defining a proposal probability distribution $Q_p(x_k)$, which includes the probability distribution $f_p(x_k)$ to be modeled (grey area). A randomly generated sample (r_1^p, r_2^p) is accepted, if the sample is located within the area enclosed by $f_p(x_k)$.

B Semantics of Statecharts

A natural technique for describing the dynamics of a system is to use a FSM. FSMs have an appealing visual representation in the form of state-transition diagrams. As in conventional state-transition diagrams, statecharts are constructed basically from states and transitions [60]. The states are depicted as rectilinear boxes with rounded corners. Each state is named and appears inside its box. Furthermore, for each state an entry, exit and do action can be defined. The latter action is executed as long as the system remains in the state. The transitions are drawn as arrows, with the triggers serving as labels. Events and conditions, which can come from external as well as internal sources, serve as triggers. The notation is depicted in Figure B.1.

The initial state is specified by a small arrow emanating from a small black circle (Figure B.2a). Statecharts may employ condition connectors, which are also called C-connectors, as shown in Figure B.2b. These C-connectors give the opportunity after an event to switch to different states depending on the satisfied condition. In general, the conditions along the branches emanating from the C-connector must be exclusive, but there can be more than two such branches.

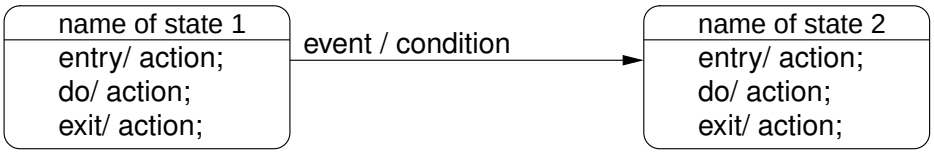

Figure B.1: States with actions, transition and an event trigger.

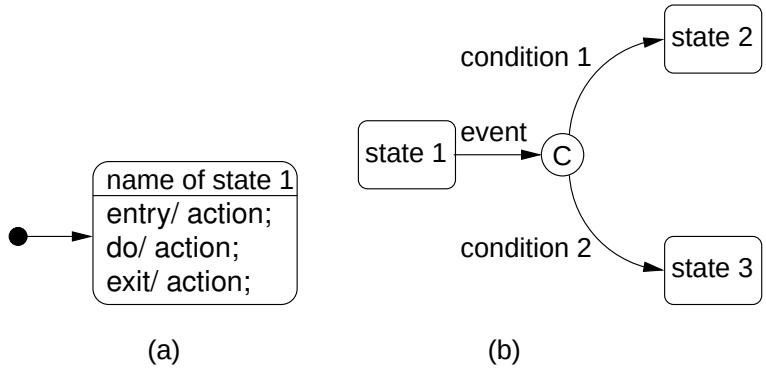

Figure B.2: (a) Entry state. (b) After an event occured, C-connectors allow to switch to different states depending on the satisfied condition.

C Derivation of the Systematic Error of Tracking Circles

In the following, we derive Equation (8.1) as briefly presented in [139]. The location of the projection of a circle center differs from the barycenter of the projected circle, since a circle is projected as an ellipse. The larger the out-of-plane rotations, the larger the eccentricity of the projected ellipse and thus the systematic error. For the sake of simplicity, we do not consider translations, which are insignificant with respect to rotation. Furthermore, if we assume a set-up as illustrated in Figure C.1, then the relationship of the camera model and marker can be reduced to 2D.

The exterior points $\mathbf{P}_a$ and $\mathbf{P}_b$ of the circle are projected to $\mathbf{p}_a$ and $\mathbf{p}_b$ on the camera target (Figure C.1). While the length a is the distance between the principal point and $\mathbf{p}_a$, the length b is the distance between the principal point and $\mathbf{p}_b$. Thus, the systematic measurement error is equal to

$$\delta_e = \left| \frac{a-b}{2} \right|. \tag{C.1}$$

Following, we are relating the lengths a and b with the lengths a' and b'. Using the theorem on intersecting lines, we can determine a as

$$a = \frac{a'f}{f+z_e} \tag{C.2}$$

and in an analogous manner b as

$$b = \frac{b'f}{f+z_e}. \tag{C.3}$$

In a next step, we try to replace a' in Equation (C.2) by taking the specific geometry in Figure C.1 into account. We can formulate the following three equations:

1.
$$a' = \frac{r_e s_1}{s_2} \tag{C.4}$$

2.
$$sin(\sigma_e) = \frac{z_e + f}{s_2 + r_e} \tag{C.5}$$

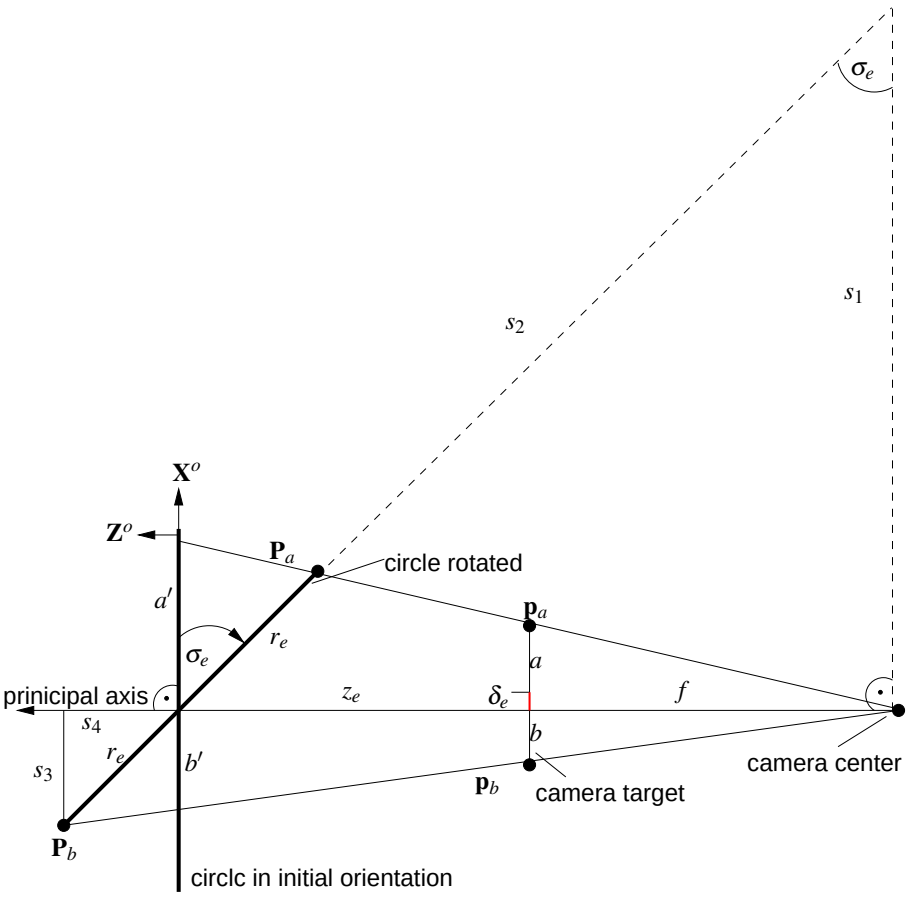

Figure C.1: The geometric relationship of a circle and its perspective projection onto a camera target. Initially, the camera target and circle with the radius r_e are parallel and have the distance z_e. The rotation of the circle is described by a quaternion q, which is located in the object coordinate system $\mathbf{X}^o$ and $\mathbf{Y}^o$ and intersecting the circle center and the rotation angle σ_e. The circle has the radius r_e and is rotated by the angle σ_e. The points $\mathbf{P}_a$ and $\mathbf{P}_b$ are located on the boundary of the circle and enclose all line of sights from the circle to the camera target. The length a' is the distance from the circle center in its initial orientation to the intersection of the line of sight passing through $\mathbf{P}_a$ and analogous the length b' between the circle center to the line of sight passing $\mathbf{P}_b$. The points $\mathbf{P}_a$ and $\mathbf{P}_b$ are projected to $\mathbf{p}_a$ and $\mathbf{p}_b$. The lengths a and b are the distances between the principal point and $\mathbf{p}_a$ and principal point and $\mathbf{p}_b$, respectively. The measurements error δ_e is the distance between the principal point and the middle of the distance between $\mathbf{p}_a$ and $\mathbf{p}_b$. The following auxiliary lengths s_1 to s_4 are only introduced for the derivation.

3.
$$cos(\sigma_e) = \frac{s_1}{s_2 + r_e} \tag{C.6}$$

Hence, we can calculate a' in Equation (C.4) by replacing the auxiliary lengths s_1 and s_2 using Equation (C.5) and (C.6) resulting in

$$\begin{aligned} a' &= \frac{r_e s_1}{s_2} \\ &= \frac{r_e cos(\sigma_e)(z_e + f)}{z_e + f - r_e sin(\sigma_e)}. \end{aligned} \tag{C.7}$$

In an analogous manner we can derive the lengths b and b' by taking the following relationships into account

1.
$$b' = \frac{s_3(z_e + f)}{s_4 + (z_e + f)} \tag{C.8}$$

2.
$$sin(\sigma_e) = \frac{s_3}{r_e} \tag{C.9}$$

3.
$$cos(\sigma_e) = \frac{s_4}{r_e} \tag{C.10}$$

resulting in

$$b' = \frac{r_e cos(\sigma_e)(z_e + f)}{z_e + f + r_e sin(\sigma_e)}. \tag{C.11}$$

Now we are able to replace a' and b' in Equation (C.2) and (C.3), which leads to

$$\begin{aligned} a &= \frac{r_e cos(\sigma_e)(z_e + f)}{z_e + f - r_e sin(\sigma_e)} \frac{f}{f + z_e} \\ &= \frac{f r_e cos(\sigma_e)}{z_e + f - r_e sin(\sigma_e)} \end{aligned} \tag{C.12}$$

$$\tag{C.13}$$

and

$$\begin{aligned} b &= \frac{r_e cos(\sigma_e)(z_e + f)}{z_e + f + r_e sin(\sigma_e)} \frac{f}{f + z_e} \\ &= \frac{f r_e cos(\sigma_e)}{z_e + f + r_e sin(\sigma_e)}. \end{aligned} \tag{C.14}$$

Finally, we can determine the error δ_e by inserting Equation (C.13) and (C.14) in Equation (C.1) resulting in

$$\begin{aligned} \delta_e &= \left| \frac{a-b}{2} \right| \\ &= \left| \frac{f r_e^2 sin(\sigma_e) cos(\sigma_e)}{(f+z_e)^2 - r_e^2 sin^2(\sigma_e)} \right| \end{aligned} \tag{C.15}$$

Bibliography

[1] J. Aitchison and J. Brown. *The Lognormal Distribution*. Cambridge Unviversity Press, 1973.

[2] M. Anisetti, V. Bellandi, E. Damiani, and F. Beverina. 3d expressive face model-based tracking algorithm. In *SPPRA'06: Proceedings of the 24th IASTED international conference on Signal processing, pattern recognition, and applications*, pages 111–116, Anaheim, CA, USA, 2006. ACTA Press.

[3] J. Anliker. Eye movements: On-line measurement, analysis, and control. In R. A. Monty and J. W. Senders, editors, *Eye Movements and Psychological Processes*, pages 185–202. NJ: Lawrence Erlbaum Associates, Hillsdale, 1976.

[4] M. Argyle and M. Cook. *Gaze and Mutual Gaze*. Cambridge University Press, Cambridge, 1976.

[5] B. Arons. Pitch-based emphasis detection for segmenting speech recordings. In *Proc. ICSLP '94*, pages 1931–1934, Yokohama, Japan, 1994.

[6] A. Bahill, D. Adler, and L. Stark. Most naturally occurring human saccaddes have magnitudes of 15 degrees or less. *Investigative Opthal.*, 14:468–469, 1975.

[7] M. Banf and V. Blanz. Example-based rendering of eye movements. *Comput. Graph. Forum*, 28(2):659–666, 2009.

[8] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., New York, NY, USA, 2001.

[9] G. R. Barnes. Vestibulo-ocular function during coordinated head and eye movements to acquire visual targets. *Journal of Physiology*, (287):127–147, 1979.

[10] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12:43–77, 1994.

[11] S. Basu, I. Essa, and A. Pentland. Motion regularization for model-based head tracking. In *In Intl. Conf. on Pattern Recognition (ICPR '96)*, 1996.

[12] W. Becker. Metrics. In D. Kelly, editor, *Neurobiology of Saccadic Eye Movements*, Elsevier, Amsterdam, pages 13–67. In: Wurtz RH, Goldberg ME (eds.), Amsterdam, 1989.

[13] M. E. Beckman and J. Hirschberg. The tobi annotation conventions. http://www.ling.ohio-state.edu/tobi/ame-tobi/annotation-conventions.html, 2007.

[14] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *ECCV '92: Proceedings of the Second European Conference on Computer Vision*, pages 237–252. Springer-Verlag, 1992.

[15] M. Bierling. *Hierarchische Displacementschätzung zur Bewegungskompensation in digitalen Fernsehbildsequenzen*. Dissertation, Universität Hannover, 1991.

[16] E. Bizzi. Central programming and perepheral feedback during eye-head coordination in monkeys. *Bibl Opthal.*, 82:220–232, 1972.

[17] M. J. Black. Robust incremental optical flow. In *Ph.D. thesis, Yale University, New Haven, CT*, 1992.

[18] C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. Proc. ACM SIGGRAPH 97, in Computer Graphics Proceedings, Annual Conference Series, 1997.

[19] I. Bronstein and K. Semendjajew. *Taschenbuch der Mathematik, 25*. Auflage, BG Teubner Verlagsgesellschaft, Stuttgart Leipzig und Verlag Nauka, Moskau, 1991.

[20] M. Buchberger. *Biomechanical Modelling of the Human Eye*. PhD thesis, Johannes Kepler Universität Linz, Österreich, 2004.

[21] R. H. S. Carpenter. *Movements of the eyes*. London: Pion, 1977.

[22] J. Cassell, C. Pelachaud, N. Badler, M. Steedman, B. Achorn, T. Becket, B. Douville, S. Prevost, and M. Stone. Animated conversation: rule-based generation of facial expression, gesture & spoken intonation for multiple conversational agents. *Computer Graphics*, 28(Annual Conference Series):413–420, 1994.

[23] J. Cassell and O. Torres. Turn taking vs. discourse structure: how best to model multimodal conversation. *In Wilks (ed.) Machine Conversations. The Hague: Kluwer.*, 1998.

[24] H.-F. Chen, P. R. Kumar, and J. H. van Schuppen. On kalman filtering for conditionally gaussian systems with random matrices. *Syst. Control Lett.*, 13(5):397–404, 1989.

[25] D. Cleveland. Method and system for accomodating pupil non-concentricity in eyetracker systems. U.S. Patent 0086057, 2003.

[26] D. Cleveland. Focus control system. U.S. Patent 4,974,010, 1990.

[27] D. Cleveland, H. J. Cleveland, and P. L. Norloff. Eye tracking method and apparatus. U.S. Patent 5,231,674, 1993.

[28] A. Colburn, M. Cohen, and S. Drucker. The role of eye gaze in avatar mediated conversational interfaces, 2000.

[29] W. S. Condon and W. D. Ogsten. A segmentation of behavior. In *Journal of psychiatric research*, pages 221–235, Great Britain, 1967. Western Psychiatric Institute and Clinic, School of Medicine, University of Pittsburgh, Pittsburgh, Pennsylvania, Pergamon Press Ltd.

[30] E. Cosatto. *Sample-Based Talking-Head Synthesis*. Phd. thesis, Signal Processing Lab, Swiss Federal Institute of Techology, Lausanne, Switzerland, 2002.

[31] E. Cosatto and H. Graf. Photo-realistic talking heads from image samples. IEEE Trans. on Multimedia, vol. 2, no. 3, pp. 152-163, 2000.

[32] M. von Cranach, R.Schmid, and M. W. Vogel. Über einige Bedingungen des Zusammenhanges von Lidschlag und Blickbewegung. *Psychol. Forsch.*, 33:68–78, 1969.

[33] H. D. Crane. The purkinje image eyetracker, image stabilization, and related forms of stimulus manipulation. In D. Kelly, editor, *Visual Science and Engineering: Models and Applications*, volume 43 of *Optical Engineering*. New York : Marcel Dekker, 1994.

[34] E. B. Dam, M. Koch, and M. Lillholm. Quaternions, interpolation and animation. Technical report, Department of Computer Science, University of Copenhagen, 1998.

[35] H. Davson. *Physiology of the Eye*. London: Macmillan, 1990.

[36] D. DeCarlo and D. N. Metaxas. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision*, 38(2):99–127, 2000.

[37] D. F. Dementhon and L. S. Davis. Model-based object pose in 25 lines of code. *Int. J. Comput. Vision*, 15(1-2):123–141, 1995.

[38] Z. Deng, J. P. Lewis, and U. Neumann. Automated eye motion using texture synthesis. *IEEE Comput. Graph. Appl.*, 25(2):24–30, 2005.

[39] F. Donders. Beitrag zur Lehre von den Bewegungen des menschlichen Auges. *Holländische Beiträge zu den anatomischen und physiologischen Wissenschaften*, 1:104–145, 1848.

[40] F. Dornaika and F. Davoine. Head and facial animation tracking using appearance-adaptive models and particle filters. In *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 10*, page 153, Washington, DC, USA, 2004. IEEE Computer Society.

[41] A. T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[42] G. J. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting face images using active appearance models. In *FG '98: Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, page 300, Washington, DC, USA, 1998. IEEE Computer Society.

[43] P. Eisert and B. Girod. Model-based 3d motion estimation with illumination compensation. In *In Proceedings International Conference on Image Processing and its Applications*, pages 194–198, 1997.

[44] S. Ellyson, J. F. Dovidio, R. L. Corson, and D. L. Vinicur. Visual dominance behavior in female dyads: Situational and personality factors. *Social Psychology Quarterly*, 43:328–336, 1980.

[45] Anatomy of the eye. `http://www.emedicinehealth.com`, 2007.

[46] C. Evinger, K. Manning, and P. Sibony. Eyelid movements, mechanisms and normal data. *Invest. Ophthalmol. Vis. Sci.*, 32:387–400, 1991.

[47] T. Ezzat, G. Geiger, and T. Poggio. Trainable videorealistic speech animation. Proc. ACM SIGGRAPH, pp. 388-397, 2002.

[48] C. Fabian, M. Fuller, B. Guo, X. Lin, and M. Kavanagh. Development of an electro-oculography (eog) measurement system. Technical report, 2002.

[49] O. Faugeras, Q.-T. Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In *European Conference on Computer Vision*, volume 558 of *Lecture Notes in Computer Science*, pages 321–334, 1992.

[50] B. Fehr and R. Exline. Social visual interaction: a conceptual and literature review. In L. Erlbaum, editor, *Nonverbal behavior and communication*, pages 225–326. In: A. Siegman, S. Feldstein (eds.), 1987.

[51] A. Fick. Die Bewegung des menschlichen Augapfels. *Zeitschrift für rationelle Medizin*, 4:109–128, 1847.

[52] G. François, P. Gautron, G. Breton, and K. Bouatouch. Anatomically accurate modeling and rendering of the human eye. In *SIGGRAPH '07: ACM SIGGRAPH 2007 sketches*, page 59, New York, NY, USA, 2007. ACM.

[53] E. G. Freedman and D. L. Sparks. Coordination of the eyes and head: movement kinematics. *Experimental Brain Research*, 131:22–32, 2000.

[54] A. Fukayama, T. Ohno, N. Mukawa, M. Sawaki, and N. Hagita. Messages embedded in gaze of interface agents — impression management with agent's gaze. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 41–48, New York, NY, USA, 2002. ACM Press.

[55] M. Garau, M. Slater, S. Bee, and M. A. Sasse. The impact of eye gaze on communication using humanoid avatars. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 309–316, New York, NY, USA, 2001. ACM Press.

[56] H. Graf, E. Cosatto, V. Strom, and F. Huang. Visual prosody: Facial movements accompanying speech. In *Proc Fifth Int. Conf. Automatic Face and Gesture Recognition*, pages 397–401, 2002.

[57] R. Günther. Detektion und Verfolgung von Merkmalspunkten im Gesicht. Studienarbeit, Leibniz Universität Hannover, 2007.

[58] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, 1998.

[59] W. R. Hamilton. Lectures on quaternions. Hodges and Smith, Dublin, 1853.

[60] D. Harel. Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8(3):231–274, June 1987.

[61] C. Harris, R. Thackray, and R. Shoenberger. Blink rate as a function of induced muscular tension and manifest anxiety. Perc. Mot. Skills, 22, 1966.

[62] F. J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. In *Proceedings of the IEEE*, volume 66, pages 51 – 83, January 1978.

[63] C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[64] T. Haslwanter. Mathematics of three-dimensional eye rotations. *Vision Research*, 35:1727–1739, 1995.

[65] Eye muscles. `http://www.healthopedia.com/pictures/eye-muscles`, 2007.

[66] H. Helmholtz. On the normal movements of the human eye (in german). *Archiv für Ophthalmologie*, IX:153–214, 1863.

[67] D. Heylen, I. van Es, E. van Dijk, and A. Nijholt. Experimenting with the gaze of a conversational agent. In J. van Kuppevelt, L. Dybkjaer, and N. Bernsen, editors, *Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems*. Kluwer Academic Publishers, 2005.

[68] B. K. Horn. *Robot Vision*. McGraw-Hill Higher Education, 1986.

[69] X. Huang and H.-W. Hon. *Spoken Language Processing: A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001. Foreword By-Raj Reddy.

[70] P. J. Huber. *Robust Statistics*. John Wiley and Sons, New York, 1981.

[71] Eyetrace systems. `http://www.iota.se`, 2006.

[72] ITU Telecom. Standardization Sector of ITU. *Methodology for the Subjective Assessment of the Quality of Television Pictures, Recommendation ITU-R BT.500-11*, August 2002.

[73] ITU International Telecom. Union - Telecom. sector. *Subjective video quality assessment methods for multimedia applications, Recommendation ITU-T P.910*, August 1999.

[74] B. Jaehne. *Digital Image Processing*. Springer, New York, USA, 1991.

[75] J. Ahlberg. *Model-based coding-Extraction, coding and evaluation of face model parameters*. Dissertation, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2002.

[76] T. Kam. *Synthesis of Finite State Machines: Functional Optimization*. Kluwer Academic Publishers, Norwell, MA, USA, 1996.

[77] M. Kampmann. *Analyse-Synthese-Codierung basierend auf dem anatomischen Modell einer menschlichen Person*. PhD thesis, Universiät Hannover, Germany, 2002.

[78] H. Kaufmann. Book review. *Strabismus*, 12:51–52(2), March 01, 2004.

[79] A. Kendon. Some functions of gaze-direction in social interaction. pages 22–63. Acta Psychologica 26, 1967.

[80] L. Kennedy and D. Ellis. Pitch-based emphasis detection for the characterization of meeting recordings. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU 2003)*, pages 243–248, St. Thomas, 2003.

[81] J. Kuipers. *Quaternions and Rotation Sequences*. Princeton University Press, Inc., Princeton, NJ, 1998.

[82] T.Kuratate, K.G.Munhall, P.E.Rubin, E.Vatikiotis-Bateson, and H.Yehia. Audio-visual synthesis of talking faces from speech production correlates. *In Proceedings of Eurospeech*, 1999.

[83] M. LaCascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on robust registration of texture-mapped 3D models. In *IEEE Trans. Pattern Analysis and Machine Intelligence*, volume 22(4), April 2000.

[84] Eyegaze systems. `http://www.eyegaze.com`, 2007.

[85] S. P. Lee, J. B. Badler, and N. I. Badler. Eyes alive. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 637–644, New York, NY, USA, 2002. ACM Press.

[86] J. Lee and H. Yang. A simultaneous estimation of rigid and non-rigid face motion. In *ICPR00*, pages Vol I: 1068–1071, 2000.

[87] S. P. Lee. *Facial Animation System with Realistic Eye Movement based on a Cognitive Model for Virtual Agents*. Phd. thesis, University of Pennsylvania, USA, 2002.

[88] A. Lefohn, B. Budge, P. Shirley, R. Caruso, and E. Reinhard. An ocularist's approach to human iris synthesis. *IEEE Comput. Graph. Appl.*, 23(6):70–75, 2003.

[89] V. Lepetit and P. Fua. Monocular model-based 3d tracking of rigid objects: A survey. *Foundations and Trends® in Computer Graphics and Vision*, 1(1).

[90] H. Li, P. Roivainen, and R. Forcheimer. 3-d motion estimation in model-based facial image coding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):545–555, 1993.

[91] E. Limpert, W. A. Stahel, and M. Abbt. Log-normal distributions across the sciences: Keys and clues. *BioScience*, 51(5):341–352, May 2001.

[92] Z. Liu and Z. Zhang. Robust head motion computation by taking advantage of physical properties. In *Workshop on Human Motion*, pages 73–, 2000.

[93] X. Ma and Z. Deng. Natural eye motion synthesis by modeling gaze-head coupling. In *VR '09: Proceedings of the 2009 IEEE Virtual Reality Conference*, pages 143–150, Washington, DC, USA, 2009. IEEE Computer Society.

[94] M. Malciu and F. Prêteux. A robust model-based approach for 3d head tracking in video sequences. In *FG '00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, page 169, Washington, DC, USA, 2000. IEEE Computer Society.

[95] T. K. Marks, J. Hershey, J. C. Roddey, and J. R. Movellan. 3d tracking of morphable objects using conditionally gaussian nonlinear filters. computer vision and image understanding, under review. see also cvpr04 workshop: Generative-model based vision, 2004.

[96] S. Masuko and J. Hoshino. Generating headeye movement for virtual actor. *Syst. Comput. Japan*, 37(12):33–44, 2006.

[97] A. Mehrabian. *Nonverbal Communications*. Aldine-Atherton, New York, 1972.

[98] N. Morgan and E. Fosler-Lussier. Combining multiple estimators of speaking rate, 1998.

[99] J. Mullin, M. Jackson, A. H. Anderson, L. Smallwood, M. A. Sasse, A. Watson, and G. Wilson. Assessment method for assessing audio and video in real-time interactive communications. Technical report, 2002.

[100] S. Njanda. Auswahl und Aktualisierung von Merkmalspunkten zur 3D Bewegungsschaetzung. Diplomarbeit, Leibniz Universität Hannover, 2006.

[101] W. Nogueira, A. Büchner, and B. Edler. Fundamental frequency coding in nofm strategies for cochlear implants. In *118th AES Convention*, volume 0, page Preprint 6515, may 2005.

[102] J. Ostermann. E-cogent: An electronic convincing agent? MPEG-4 Facial Animation: The Standard, Implementation and Applications, Igor S. Pandzic (Editor), Robert Forchheimer (Editor), Wiley, Chichester, England, 2002.

[103] J. Ostermann and A. Weissenfeld. Talking faces - technologies and applications. Proceedings of 11th International Workshop on Systems, Signals and Image Processing, IWSSIP, 2004.

[104] J. Ostermann. *Analyse-Synthese-Codierung basierend auf dem Modell bewegter Dreidimensionaler Objekte*. PhD thesis, Universiät Hannover, Germany, 1995.

[105] I. Pandzic, J. Ostermann, and D. Millen. User evaluation: Synthetic talking faces for interactive services. The Visual Computer, vol. 15, Issue 7/8, 1999.

[106] F. I. Parke. Computer generated animation of faces. In *ACM'72: Proceedings of the ACM annual conference*, pages 451–457, New York, NY, USA, 1972. ACM Press.

[107] I. Poggi, C. Pelachaud, and F. de Rosis. Eye communication in a conversational 3d synthetic agent. *AI Communications*, 13(3):169–182, 2000.

[108] E. Ponder and W. P. Kennedy. ON THE ACT OF BLINKING. *Q J Exp Physiol*, 18(2):89–110, 1927.

[109] S. Priglinger and M. Buchberger. *Computer Assisted Eye Motility Diagnostics*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[110] H. Purnhagen, B. Edler, and C. Ferekidis. Object-based analysis/synthesis audio coder for very low bit rates. In *104th AES Convention*, volume 0, page Preprint 4747, may 1998.

[111] Q. Fan. Prosody-driven eye gaze pattern animation. Master thesis, Leibniz Universität Hannover, 2007.

[112] L. R. Rabiner, M. Cheng, A. Rosenberg, and C. McGonegal. A comparative performance study of several pitch detection algorithms. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24(5):399–418, 1976.

[113] T. Raphan and B. Cohen. Amplitude of human head movements associated with horizontal saccades. *Exp. Brain Res.*, 145(1):1–27, 2002.

[114] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[115] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley and Sons, New York, 1987.

[116] L. Sachs and J. Hedderich. *Angewandte Statistik*. Zwoelfte Auflage, Springer-Verlag, Berlin, 2006.

[117] D. D. Salvucci and J. H. Goldberg. Identifying fixations and saccades in eye-tracking protocols. In *ETRA '00: Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78, New York, NY, USA, 2000. ACM Press.

[118] N. Sarris, D. Makris, and M. G. Strintzis. Three dimensional model based rigid tracking of a human head. In *International Workshop on Intelligent Communication Technologies and Applications with Emphasis on Mobile Communications*, 1999.

[119] H. D. Schworm, J. Ygge, T. Pansell, and G. Lennerstrand. Assessment of Ocular Counterroll during Head Tilt Using Binocular Video Oculography. *Invest. Ophthalmol. Vis. Sci.*, 43(3):662–667, 2002.

[120] K. Shoemakes. Euler angle conversion. *Academic Press Professional, Inc.*, pages 222–229, 1994.

[121] Iris eye-tracker. `http://www.skalar.nl`, 2007.

[122] C. Slama. *Manual of Photogrammetry*. American Society of Photogrammetry and Remote Sensing, 4. edition, 1980.

[123] J. Stahl. Amplitude of human head movements associated with horizontal saccades. *Exp. Brain Res.*, 126(1):41–54, 1999.

[124] C. V. Stewart. Robust parameter estimation in computer vision. *SIAM Reviews*, 41:513–537, 1999.

[125] B. Streefkerk. Acoustical correlates of prominence: A design for research, 1997.

[126] J. Ström. Model-based real-time head tracking. *EURASIP J. Appl. Signal Process.*, 2002(1):1039–1052, 2002.

[127] A. K. Syrdal, J. Hirschberg, J. McGory, and M. Beckman. Automatic tobi prediction and alignment to speed manual labeling of prosody. *Speech Commun.*, 33(1-2):135–151, 2001.

[128] F. Tamburini. Automatic prosodic prominence detection in speech using acoustic features: an unsupervised system. In *In Proceedings of Eurospeech 2003*, pages 129–132, 2003.

[129] M. Tarini, H. Yamauchi, J. Haber, and H.-P. Seidel. Texturing Faces. In *Proc. Graphics Interface*, pages 89–98, May 2002.

[130] J. Terken. Fundamental frequency and perceived prominence of accented syllables. *The Journal of the Acoustical Society of America*, 95:3662–3665, 1994.

[131] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3-d machine vision metrology using off-the-shelf cameras and lenses. *IEEE Transaction on Robotics and Automation*, 3(4):323–344, 1987.

[132] A. M. Turing. Computing machinery and intelligence. *MIND*, 49(236):433–460, 1950.

[133] D. Tweed and T. Vilis. Geometric relations of eye position and velocity vectors during saccades. *Vision Res.*, 30:111–127, 1990.

[134] D. Tweed. Visual-motor optimization in binocular control. *Vision Research*, 37(14):1939–1951, 1996.

[135] L. Vacchetti and V. Lepetit. Stable real-time 3d tracking using online and offline information. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(10):1385–1391, 2004.

[136] J. v. Neumann. Various techniques used in connection with random digits. 3:36–38, 1951. Reprinted in *Collected Works,* Vol. V, pp. 768–770.

[137] T. Warabi. The reaction time of eye-head coordination in man. *Neuurosci. Lett. 6*, pages 45–51, 1977.

[138] Y. Watanabe, T. Fujita, and J. Gyoba. Investigation of the blinking contingent upon sacadic eye movement. *Tohoku Psychol Folia*, 39:121–129, 1980.

[139] S. Weik. *Animierbare Modelle natürlicher Personen für virtuelle Szenen*. PhD thesis, Universiät Hannover, Germany, 2002.

[140] A. Weissenfeld, K. Liu, S. Klomp, and J. Ostermann. Personalized unit selection for an image-based facial animation system. In *MMSP 2005*, volume 0, nov 2005.

[141] A. Weissenfeld, K. Liu, and J. Ostermann. Video-realistic image-based eye animation system. In *EUROGRAPHICS 2009 (Short Paper)*, volume 0, apr 2009.

[142] A. Weissenfeld, O. Urfalioglu, K. Liu, and J. Ostermann. Robust rigid head motion estimation based on differential evolution. In *IEEE International Conference on Multimedia & Expo 2006*, volume 0, pages 225 – 228, jul 2006.

[143] S. Winter. Estimation of facial expressions in an image sequence. Diplomarbeit, Leibniz Universität Hannover, 2005.

[144] J. Xiao, T. Kanade, and J. F. Cohn. Robust full-motion recovery of head by dynamic templates and re-registration techniques. In *FGR '02: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, page 163. IEEE Computer Society, 2002.

[145] S. Yan. Control parameter for photorealistic eye animations. Master thesis, Leibniz Universität Hannover, 2006.

[146] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *HTK Book*. Cambridge University Engineering Department, 2005.

[147] L. Young and D. Sheena. Survey of eye movement recording methods. Behavior Research Methods and Instrumentation 7, 1975.

[148] J. Zheng, H. Franco, F. Weng, A. Sankar, and H. Bratt. Word-level rate of speech modeling using rate-specific phones and pronunciations. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume III, pages 1775–1778, 2000.

Curriculum Vitae

Personal Data

Name	Axel Weißenfeld
Date of birth	August 3rd, 1976
Place of birth	Langenhagen, Germany

School Education

1983 - 1997	Elementary school, Gymnasium Langenhagen, Abitur-matriculation qualification
1993 - 1994	Killeen-High-School, Texas, USA

Academic Education

1998 - 2003	Electrical Engineering, Leibniz Universität Hannover, Germany Field of study: Telecommunication Degree: Diplom-Ingenieur
2003 - 2007	Institute of Information Processing, Leibniz Universität Hannover, Germany Ph.D. study

Work Experience

2003 - 2007	Institute of Information Processing, Leibniz Universität Hannover, Germany Scientific researcher
since 2008	VCS Video Communication Systems, BOSCH Security Systems Firmware Developer