Locating and Tracking of Human Faces with Neural Networks

H. Martin Hunke

August 1994
CMU–CS–94–155

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

© H. Martin Hunke, 1994

This research was done in partial fullfillment for a Diplom in Informatik at the University of Karlsruhe (Master of Science in Computer Science). I gratefully acknowledge the collaboration between the University of Karlsruhe and Carnegie Mellon University, that made this research possible.

This research was sponsored by the Department of the Navy, Office of Naval Research under Grant No. N00014–93–1–0806.

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

Keywords: Vision, locating, tracking, multimodal, artificial neural networks, human computer interaction

Abstract

Effective Human–to–Human communication involves both auditory and visual modalities, providing robustness and naturalness in realistic communication situations. Recent efforts at our lab are aimed at providing such multimodal capabilities for human-machine communication as well by introducing gesture, character and speech recognition, eye-tracking and lipreading. Most of the visual modalities require a stable image of a speaker's face.

In this technical report a connectionist face tracker is proposed that manipulates camera orientation and zoom, to keep a person's face located at all times in an image sequence. The system operates in real time and can adapt rapidly to different lighting conditions, different cameras and faces, making it robust against environmental variability. Extensions and integration of the system with a multimodal interface will be presented.

Contents

1 Introduction **1**
- 1.1 Overview on the Face Tracking System . . . 2
- 1.2 Approach and Chapter Overview . . . 3
- 1.3 Acknowledgements . . . 4

2 Related Work **5**

3 Features for Locating Faces **7**
- 3.1 Color . . . 7
  - 3.1.1 Chromatic Diagram . . . 8
  - 3.1.2 Color Distributions . . . 9
  - 3.1.3 Color Shifts . . . 11
  - 3.1.4 Face–Like Color Classifier (FCC) . . . 12
- 3.2 Motion . . . 18
- 3.3 Coherent Objects . . . 19
- 3.4 Combining the Features . . . 21

4 Structure of the Face Tracking System **23**
- 4.1 Searching a Human Face . . . 24
  - 4.1.1 Detecting Faces . . . 24
  - 4.1.2 Selection of an Object . . . 24
- 4.2 Tracking of Faces . . . 25
  - 4.2.1 The Virtual Camera . . . 26
  - 4.2.2 Camera Calibration . . . 27
  - 4.2.3 Using of Motion While Tracking a Face . . . 28
  - 4.2.4 Adjustment of the Color Classifier . . . 29
- 4.3 Structure of the Entire System . . . 29
- 4.4 Switching between the Basic States . . . 30

4.5 Example of a Face Tracking Sequence 31

5 Artificial Neural Networks **36**
5.1 Task and Training of the Networks 36
5.1.1 Representation of the Net Input 39
5.1.2 Representation of the Net Output 39
5.1.3 The Back–Propagation Method 41
5.2 Neural Networks without FCC 43
5.3 Neural Networks with FCC 44

6 Artificial Images and Image Sequences **48**
6.1 Training of Neural Networks with Artificial Images 48
6.1.1 Training Set Requirements 48
6.1.2 Assigning of Net Inputs to Net Outputs 49
6.2 Image Sequences 50
6.3 Creating Artificial Images 51
6.3.1 Creation of the Databases 51
6.3.2 The Blue–Screen Method 53
6.3.3 Creation of an Artificial Image 53

7 Evaluation **57**
7.1 Test Sequences 57
7.1.1 Recording of Test Sequences 57
7.1.2 Manual Marking of the Test Sequences 58
7.1.3 Selection of test sequences 59
7.1.4 Evaluation of a Test Sequence 60
7.2 Results 62
7.2.1 Test Sequence 1 62
7.2.2 Test Sequence 2 64

8 Conclusion **70**

9 Future Directions **72**
9.1 Extensions of the System 72
9.1.1 Stereoscopic Images 72
9.1.2 Locating Eyes and Lips 72
9.1.3 Locating Several Faces 72
9.2 Applications of the System 73
9.2.1 Videophone 73

9.2.2 Identifying Individuals 73
9.2.3 Using the Speaker's Position 73

Chapter 1

Introduction

Today's computers are not capable of most of the communication modalities used by human beings. Artificial speech synthesis, speech, gesture, and hand writing recognition are far beneath human capabilities. Further development in this area could improve efficiency and robustness of human–computer communication.

Recent research deals with integrating visual aspects of communication. Speech recognition systems using only auditory signals can be improved by recognizing lip movements [2] [5]. Similar sounds often correspond to very different lip movements (for example "m" and "n"). Especially in situations providing background noise or several persons talking simultaneously (cocktail party effect), the recognition rate significantly improves by considering visual aspects of communication.

Eye tracking allows the computer to determine the point of the screen, where an individual looks at [1]. In combination with speech recognition text editing systems could offer much more comfortable user interfaces than today's systems. Spoken commands like "delete this word" while focussing the gaze on a word on the screen would make human–computer communication more natural.

Recognizing the identity of an individual [3] by extracting features of a camera image could be used for matters of security or to automatically load an user specific setup. Some additional research projects are described in [20].

Though encouraging results in using visual aspects of communications could be achieved, the developed systems mostly require the individual to be in a given position in front of the camera. Subsequently these systems are

limited to not moving faces. Human beings are always in motion, especially while communicating. A system with the capability to locate faces in camera images could provide other systems using face images with stable images of moving individuals.

1.1 Overview on the Face Tracking System

In this technical report a system to locate faces in camera images for human–computer communication is provided. To allow individuals to move freely in an environment, the system should be capable of adjusting the camera position and zoom lens. The system presented in this report is capable of searching human beings in arbitrary environments, selecting the individual most near to the camera, and maintaining camera position and zoom lens adjustment corresponding to the motion of the observed individual. The subimage covering the face is automatically extracted, scaled, and available for other systems using the visual information. The resolution of the extracted face image is limited only by the used hardware.

The camera used in this system (Sony CCD–TR 101) is mounted on two stepmotors, allowing horizontal and vertical turns. The step motors are controlled by a serial port. The remote control for the zoom lens of the camera has been manipulated to allow access by another serial port. Both the position of the camera and the zoom lens therefore can be adjusted by the computer. The camera images are grabbed by a framegrabber, digitizing the analog video signal into binary RGB–values. The entire computation is done on a single HP 9000/735 workstation.
The system is capable of performing the following operations in real time:

- Locating arbitrary faces in arbitrary environments.
- Selecting the face most near to the camera.
- Maintaining camera position and zoom lens, so that an individual's face is in constant size in the camera image.
- Determining the position and size of the face within the camera image, so that a stable image of the face can be obtained by scaling the extracted subimage covered by the face.
- Learning features of the tracked face to adjust the tracking procedure to the appearance ot the face.

- Adjusting to changing lighting situations.

Selecting one of several faces for tracking could be supported by an microphone–array, so that faces are preferably located in directions, in which a speech signal could be detected [6]. The information about the position of a speaker can be used to stress speech signals in this direction.

1.2 Approach and Chapter Overview

A listing of some related research is given in chapter 2.

Chapter 3 discusses features used for locating faces, mainly motion and color. A color classifier distinguishes between colors occuring in human faces and background colors. For this purpose normalized colors are used which don't consider brightness. The color classifier automatically adjusts to colors occuring in the located face and the lighting situation. All color dependencies can be connected to this classifier, so that exchanging of the camera hardware requires only adjusting this system part but no retraining of neural networks. The theoretical background of chromatic, the construction of the color classifier, and the analysis of coherent moving objects are discussed in this chapter.

In chapter 4 the construction of a complete face tracking system using the features of chapter 3 is discussed. Based on locating moving objects containing skin–color, faces can be located and tracked. A virtual camera is used to avoid a limitation of the system by the inertia of the physical camera, so that new adjustments can be performed virtually without delay.

An additional recognition of form can be performed by using neural networks. The network approach is described in chapter 5. Multi–layer perceptrones are trained with the back–propagation method and determine the position and size of a face in a camera image. Several network structures and preprocessing methods are discussed.

A method to generate an arbitrary number of artifical images and image sequences containing moving faces is provided in chapter 6. These images are used for the network training and evaluation of the entire system.

The performance of the provided methods and the entire system is evaluated in chapter 7. Recorded image sequences containing faces are used to determine the reliability and accuracy of the system. The position and size of a face in each image is therefore manually marked, so that the system output can be evaluated.

Chapter 8 and 9 contain an overview of the provided methods, the obtained results, and the possible future directions.

1.3 Acknowledgements

I greatfully acknowledge my advisor Prof. Alexander Waibel who made this reasearch at the Carnegie Mellon University possible.
I would like to thank Stefan Manke and Michael Finke for their helpful comments.
Many people of the robotics institute supported me in selecting hardware. I would like to thank especially Dean Pomerleau for his helpful feedback.
The readiness to help of the people at the speech group of Carnegie Mellon were a great support. I would like to thank especially Ricky Houghton, Arthur E. McNair, and Torsten Zeppenfeld.
Last but not least I would like to thank the nearly 100 students, who took part in the camera recordings.

Chapter 2

Related Work

Many research projects dealt with the recognition of features in human faces. In early works distances of striking points were used to identify faces [16]. Recent works mostly use artificial neural networks for this purpose. The first twolayered linear networks could store faces, identify them and restore their complete image, given a noisy and uncomplete version of the face [12]. Another auto–associative method uses back–propagation to compress images. The network is trained with identical input and desired output, so that the hidden units are forced to find a representation of the images. A small number of hidden units leads to a high compression rate [3]. The features extracted by these units can be used to identify sex [8], identity [7], and gesture [3]. All these methods only operate on images containing one face in a given position and size.

More similar to the problems considered in this technical report are projects to locate features in human faces. Template matching with an average face was used to locate eyes in imageas of faces [11]. The same publication discusses this problem with multi–layered perceptrons. A window covering a subimage is moved line by line over the entire image and determines the input of a network with three layers. The single output unit of the network determines, if an eye is located in the subimage or not. Another research project uses template matching to track features of faces like eyes and lips, which are marked interactivly [15]. This work shows that template matching is very useful to find known features.

A different approach is used successfully for locating faces in newspaper articles. After detecting edges in the image, a structural model is matched against the located features [9]. Eigenfaces can be used to identify faces [18].

The eigenfaces are determined by performing a principal component analysis on a set of example images with centered faces of the same size. In addition the existence of a face in a given image can be determined. By moving a window covering a subimage over the entire image, faces can be located within the entire image. An conjuntion of this information with motion analysis allows face tracking in near real time given simple non confusing backgrounds.

By using a combination of one of the above slow methods to locate a face with template matching for fast tracking of the located object, the problem of locating and tracking faces could be solved. The task of locating a face and the task of tracking the located face are very similar. Indeed the second task is identical to the first task with additional constraints. Tracking a face is nothing else then to locate a face with known features in following images. Once a face has been located, the shape, skin–color, size and position are known. In the next image tracking of the face can be described as searching a face with similar skin–color and size near to the position of the face in the last image. Therfore it is unsatisfying to use entirely different methods for similar tasks.

The approach developed in this technical report locates faces by searching for features, which occur in all human faces. After locating a face, additional features matching this individual face are extracted. Locating a face in the next image of a sequence with condsideration of these additional features results in fast tracking.

Chapter 3

Features for Locating Faces

Which features do all faces have in common? Are these features easily to extract in low resolution camera images? Human beings loacate faces in high resolution images by locating substructures like eyes, nose and lips. In low resolution images these substructures are not always visible. Using the following features are much more reliable under these conditions:

1. Color
2. Motion
3. Coherence

These features are described detailled in the following in regard to the purpose of tracking faces.

3.1 Color

Using color as a feature results in the following difficulties:

- Neither the analog video signal nor the digitizing of the framegrabber are linear or normalized. Different hardware produces significantly different color values of the same object. Therefore a method to adjust the system in a short time to new hardware environments is described.
- Skin–colors of human beings vary strongly. Even one single person can have different skin–colors depending on the emotional condition (flushing), sun tan, reflections of the environment, etc. In regard to

these problems a method is developed, which automatically adjusts the system to different skin–colors and lighting situations.

- The training of neural networks with color values results in weight vectors, which are dependent on hardware, lighting situation, etc. In the following a method to avoid this dependency is described.

In this chapter methods to normalize colored camera images and color distributions are described. The rgb–values of the video signal are converted to chromatic colors, which contain no brightness information. The different color distributions of backgrounds and faces are used to create face color classifiers (FCC), classifying colors as background or face–like colors. A general FCC (GFCC) is able to classify all skin–colors of human beings and is used to locate unknown faces. After locating a face, the colors actually occuring in the face determine an individual FCC (IFCC), which is used to find the same face again in the following camera images. By frequently adjusting the IFCC the system can deal with changing lighting situations.

3.1.1 Chromatic Diagram

A framegrabber digitizes the analog video signal and transmits digital data to the computer. The camera images are stored as pixel matrices, containing three digital values for each pixel: the basic colors red, green, and blue in a range from $[0, \frac{1}{255}, \frac{2}{255}, \ldots 1]$. Pixels are defined as $Q = (R, G, B)$. The pixels $Q_1 = (0.1, 0.5, 0.1)$ and $Q_2 = (0.2, 1.0, 0.2)$ have the same color but different brightness. Actually the brightness is irrelevant for specifying skin–color. Chromatic colors are defined as colors with normalized brightness, so that the sum of the rgb–values is constant [21]. Because of the nonlinearity of the framegrabber and the CCD–chip in the camera, the determined chromatic colors still have a small dependency on brightness, which can be neglected. The chromatic colors are defined as $f : \mathbb{R}^3 \mapsto \mathbb{R}^2$ with

$$q = (r, g) = f(R, G, B)$$

and

$$r = \frac{R}{R + G + B}$$

$$g = \frac{G}{R + G + B}$$

The normalized value for blue b is redundant, because of $r + g + b = 1$. All chromatic colors $q = (r, g)$ can be sketched in a chromatic diagram (figure 3.1 (a)). In figure (b) an example for a color distribution is shown, to show the connection to the chromatic diagram. Color distributions are explained in the next section.

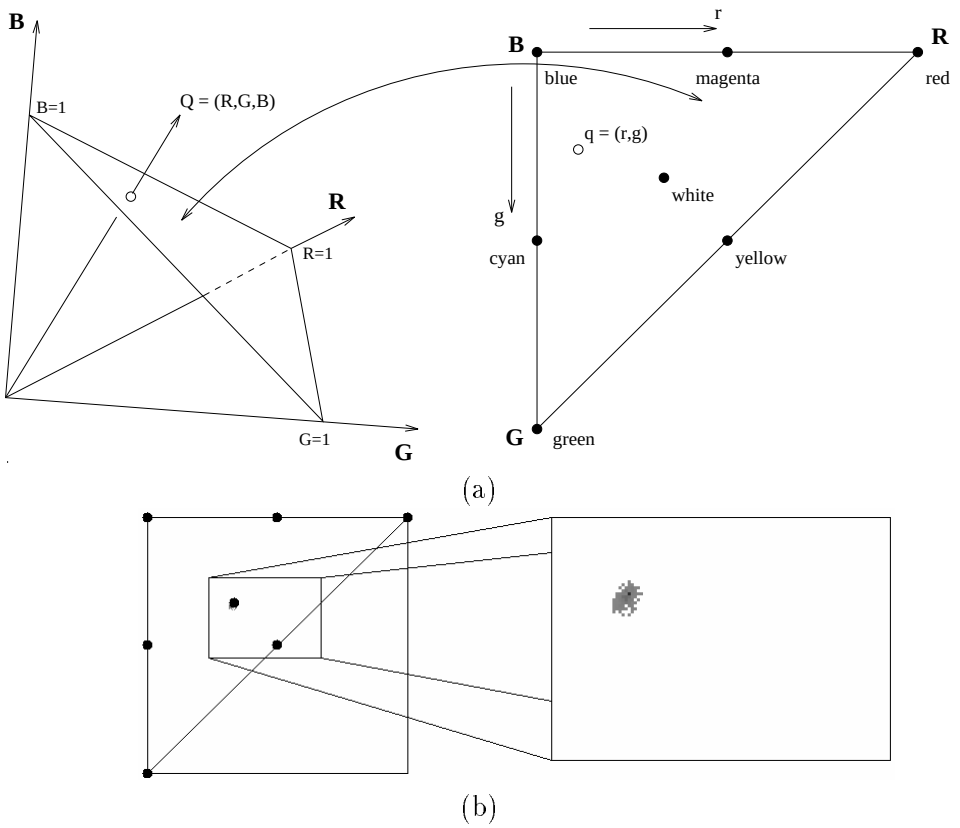


Figure 3.1: Chromatic diagram
(a) Relation between rgb–values and chromatic diagram, **(b)** Example for a color distribution

3.1.2 Color Distributions

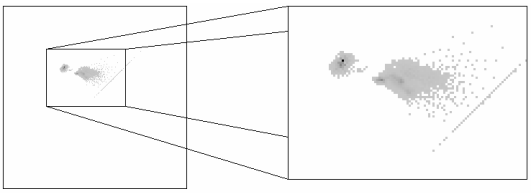
The chromatic diagram contains $\frac{256^2}{2}$ possible colors $q = (r, g)$. The value $N_{r,g}$ is defined as the frequency, how often a color (r, g) occurs in an image. A color distribution is defined as the normalized frequencies $\overline{N}_{r,g}$:

$$\overline{N}_{r,g} = \frac{N_{r,g}}{\max_{i+j \leq 255} N_{i,j}}$$

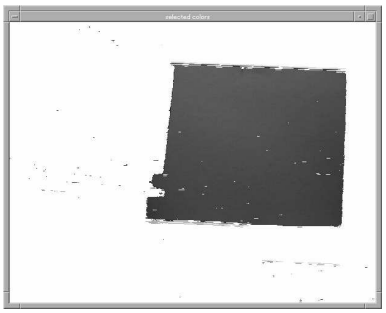
The shown color distributions in this report use the color range gray to black to specify the frequencies. Colors occuring less then 256 times as seldom as the most frequent color ($\overline{N}_{r,g} < \frac{1}{256}$) are represented with white.

The figures 3.2 (a) and (b) show an original camera image and the corresponding color distribution. The diagonal line in the color distribution indicates the existence of colors in the original image, which don't contain the basic color blue. Actually the camera is less sensitive for blue and the indoor lighting conditions support a color shift to the opposite color yellow. The right part of figure (b) magnifies the center of the color distribution.

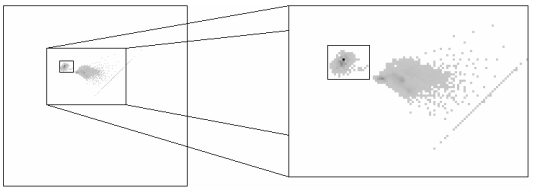
(a)



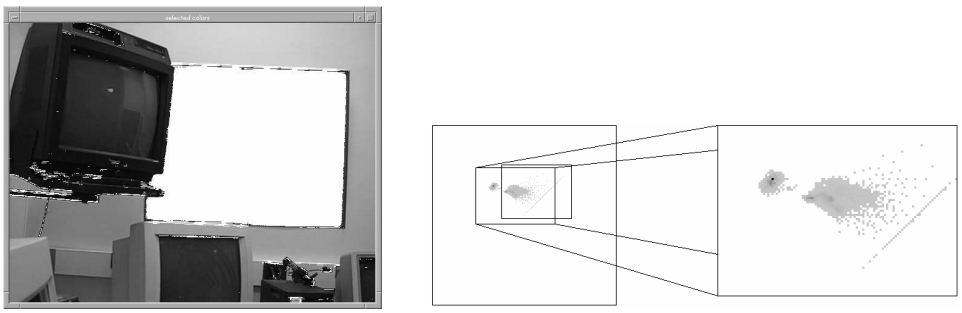
(b)



(c)



(d)



(e) (f)

Figure 3.2: Camera image with corresponding color distribution
(a) and **(b)** Original and color distribution, **(c)** and **(d)** Selection of blue colors, **(e)** and **(f)** Selection of not blue colors

A rectangle in the color distribution in figure 3.2 (d) marks an area of colors. All colors outside this area are drawn white in figure (c). The remaining part of the original image shows a blue colored sheet of paper, the blue–screen (chapter 6.3.2). Comparing the color distribution with the chromatic diagram in figure 3.1 (a), the seleceted color blue is located at the position, where the color white apperas in the chromatic diagram. The reason for this color shift is explained in the next section. To give another example, in figure 3.2 (e) another area of colors (f) is selected.

3.1.3 Color Shifts

The surface of an object determines which colors are reflected and which not. In addition the actual color of an object is dependent on the lighting situation, because only those colors can be reflected, which are part of the lighting. Sunlight contains more blue than flourescent lighting. Color distributions of outdoor recordings therefore are shifted to blue, while distributions of indoor recordings are shifted to yellow. Modern video cameras are supported with an automatic or manual white–balance control to adjust the camera to indoor or outdoor lighting situations, but still significant different colors of the same object can be measured depending on the lighting situation.

Table 3.1 shows average and average deviation of the most frequent occuring color $\overline{p} = (\overline{r}, \overline{g})$ of a recorded face respective to the lighting situation and setting of the white–balance. In all recordings the same face was used as object.

white–balance	lighting situation	average	average deviation	
automatic	indoor	(113,90)	(3.7,2.7)	a_1
automatic	outdoor	(103,78)	(2.9,0.4)	a_2
automatic	indoor and outdoor	(108,84)	(5.2,5.8)	
indoor	indoor	(107,97)	(1.1,0.7)	m_1
indoor	outdoor	(80,80)	(3.1,3.1)	f
outdoor	indoor	(160,88)	(8.9,4.0)	f
outdoor	outdoor	(112,77)	(0.1,0.5)	m_2
corresponding to lighting situation	indoor and outdoor	(109,89)	(2.6,9.9)	

Table 3.1: Color shifts corresponding to lighting situations

The measured values are dependent on the used hardware, so that only the tendency is further examined. Wrong setting of the white–balance (marked with f) results in extreme color shifts. The differences of the average values corresponding to indoor and outdoor recordings using manual setting (m_1 and m_2) are smaller than using automatic setting (a_1 und a_2). To reduce the color shifts, the camera is used only with manual white–balance setting. Still it is not possible to obtain identical colors with indoor and outdoor recordings.

3.1.4 Face–Like Color Classifier (FCC)

Only few of all possible colors actually occur in faces. A strong blue can occur in backgrounds or in eyes, but not as skin–color. In the following a face–like color classifier (FCC) is developed, classifying all colors as face–like colors or background colors. The creation of the FCC is illustrated at figure 3.3.

Figure 3.3: Face

Figure 3.4: Area selection

The color distribution of the selected area in figure 3.4 can be used as an individual FCC (IFCC), classifying colors occuring often in the given example as face–like colors. Figure 3.5 shows the application of the classifier. In (a) the color distribution of the selected area is shown. Figure (b) demonstrates the application of the IFCC on the entire face. Each pixel $q = (r, g)$ is drawn with the gray value $\overline{N}_{r,g}$ of the color distribution. Pixels with a high gray value in (b) correspond to frequently occuring colors, which are drawn dark in the color distribution (a). The actual skin–colors occur in a small area of the chromatic diagram, marked by a rectangle in (a). In figure (c) all pixels are drawn white, which have colors ouside of this rectangle. All other pixels remain their original gray value. The areas of skin–color are stressed and in addition the areas containing the eyes and the lips are emphasized, because they don't contain skin–color. Though the mouth is part of the area which is used to create the IFCC, the colors of the lips occur much more seldom in the image than skin–colors, resulting in classifying lips as not skin–colored.

(a) (b) (c)

Figure 3.5: IFCC
(a) IFCC, **(b)** Application of the IFCC, **(c)** Skin–colored areas

The above method achieved good results on the given example but uses some requirements, which are not met in all situations. The following problems are discussed in the next sections:

1. **Estimation of Color Distributions with a Few Pixels**
   The IFCC could be easily constructed, because a subimage containing mainly skin–colors was availabe. If this subimage exists only in a small resolution, the number of pixels may be too small to estimate a color distribution, which is valid for the entire face.

2. **Unknown Faces**
   The above method requires a subimage of the face, giving the actual color distribution. Faces can't be located, if the skin–color is not known. The development of a general FCC (GFCC) will be illustrated, which is capable of locating unknown faces.

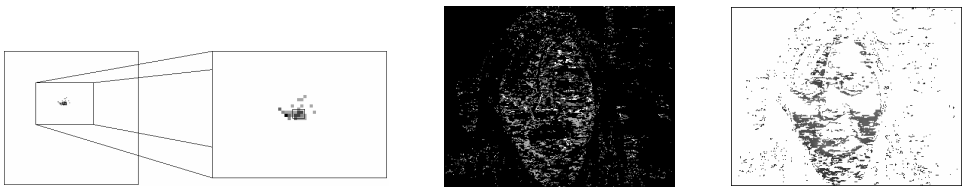
3. **Backgrounds Containing Skin–colors**
   The color classification achieved good results, because the color distributions of face and background were disjunct in the given example. For other situations the colors of the background must be considered to avoid misclassifications.

4. **Changing Lighting Situations**
   The subimage used for the construction of the IFCC didn't contain hair or parts of the background. If the subimages are automatically determined, this can't be guaranteed. In this case it must be prevented, that the IFCC adjusts to hair color or even colors occuring in the background.

Estimation of Color Distributions with a Few Pixels

In the above example a high resolution image containing many representative pixels could be used to determine the corrosponding color distribution. Figure 3.6 shows the same image in a lower resolution, containing only a few different pixels. Applying the same method of creating an IFCC results in figure 3.7.

Figure 3.6: Selected area in low resolution



(a) (b) (c)

Figure 3.7: IFCC mit wenigen Pixeln
(a) IFCC, **(b)** Application of the IFCC, **(c)** Skin–colored areas

Applying this IFCC to the face results in lower brightness of the face (b) than with the other IFCC from figure 3.5. The color distribution obtained by the low resolution subimage doesn't correspond to the entire face. In (a) an area containing the most frequent colors is selected, so that in (c) all colors outside this area are drawn white. The number of colors remaining in their original gray value increase in that area of the face, which was used for creating the IFCC.

For the purpose of creating a color distribution the color of each pixel should be considered as a representant for neighbored colors of the chromatic diagram rather than the exact value. Therefore the neighborhood size is defined as the deviation area of color shifts (chapter 3.1.3). By digitizing the analog video signal areas of colors are projected to discrete values. Figure 3.8 illustrates possible errors. Though all dots are piled up near the line between the two discrete classes, class **a** is considered to contain much more dots than class **b**. Low–pass filtering can perform the task of building neighborhood classes, as shown in figure 3.10. A local operator (figure 3.9) is applied to the color distribution. This method is explained more detailed in [14]. The filtered distribution (a) achieves significantly better results (b) and (c).

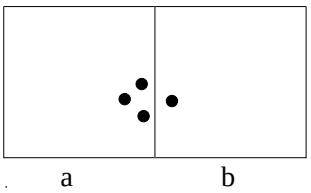


Figure 3.8: Error by digitizing

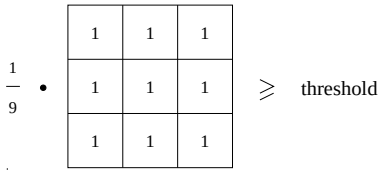
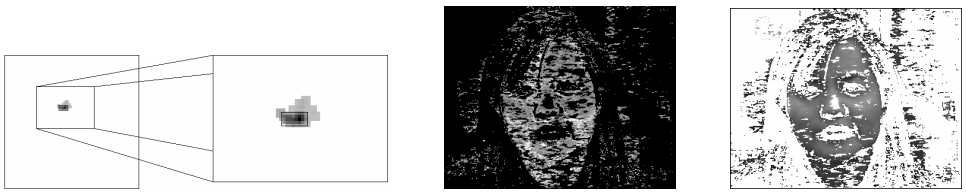


Figure 3.9: Low–pass filter



(a) (b) (c)

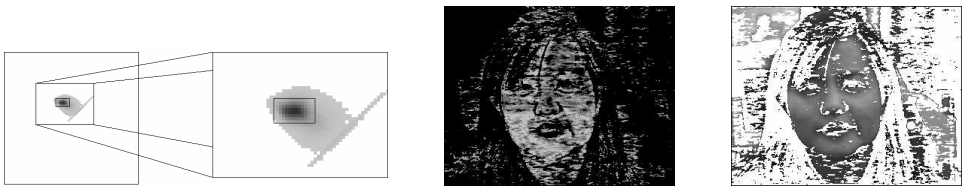
Figure 3.10: Low–pass filtered IFCC
(a) Low–pass filtered IFCC, **(b)** Application of the IFCC, **(c)** Skin–colored areas

Unknown Faces

How can colors be used to locate an unknown face? Skin–colors seem to vary over a wide range, actually they differ much less in color than in brightness. More exactly different skin types reflect nearly the same colors, but in different intensities. Figure 3.11 (a) shows a color distribution of an image containing skin areas from 30 faces with different skin–colors. The intensity of reflections is not considered in the distributions, because the chromatic colors don't contain brightness. In these faces asian, black, and white people are equally represented. Surprisingly the skin–colors occuring most often in these images (the dark blob) can be loacated in one samll area of the chromatic diagram.

For a given hardware configuration and lighting situation, a few example images of faces with different skin–color are sufficient to create a color distribution, which contains all skin-colors that can occur in unknown faces. Similar to the previous section low–pass filtering improves the color distribution, if only a few images are used. This color distribution is used as a general FCC (GFCC) and is the only part of the system, that has to be

determined again after changing hardware or lighting situation (from indoor to outdoor, etc.).



(a) (b) (c)

Figure 3.11: GFCC
(a) GFCC, (b) Application of the GFCC, (c) Skin–colored areas

Applying the GFCC (b) includes a higher probability of classifying parts of the background as skin–colored areas, because more colors are considered as skin–colors. In (c) this is distinctly visible. Therefore this classifier is used only during the search for an arbitrary face. Once a face can be located, the classifier is adjusted to the actual skin–color.

Backgrounds Containing Skin–Colors

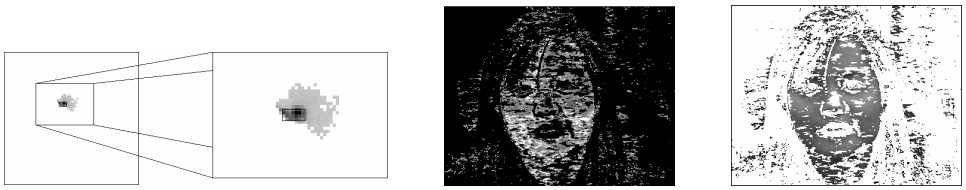
If the background contains many face–like colors, misclassifications can occur. The background of figure 3.4 is shown in figure 3.12 with the corresponding color distribution.

(a) (b)

Figure 3.12: Background
(a) Color distribution, (b) Skin–colored areas

The rectangle in (a) marks an area of the chromatic diagram, which contains skin–colors. The color distributions of the background and the previously used FCC aren't disjunct. This can be seen in the results of the classifier, like the bright areas of the background in figure 3.10 (b). Subtracting the color distribution of the background from the distribution of

the FCC results in a new classifier (figure 3.13), which consideres the actual background. Only those colors are still used for skin–color classification, which don't occur frequently in the background.



(a) (b) (c)

Figure 3.13: Consideration of the background
(a) FCC after subtracting background color distribution, (b) Applying the FCC, (c) Skin–colored areas

Changing Lighting Situations

Because of the object motion the lighting situation changes constantly, for example from direct ligthing zu indirect lighting in shadow areas. The color of the observed object changes corresponding to the lighting. Adjusting the IFCC to these changing conditions is therefore highly desirable. By applying the IFCC to the camera image, each pixel is classified as part of the face or background. The colors of the face area pixels can be used to create a new IFCC classifying all other colors than the actual skin–color as background colors. By combining the old and the new IFCC an adjustment to changing lighting conditions is easily possible. In the next image the combination IFCC is used to locate the face and the found pixles are used for a further adjustment of the classifier.

Assuming the system locates the face area, so that parts of the hair or background are misclassified as part of the face, adjusting the IFCC to these colors may result in even larger parts of the areas containing hair or background in the next image. Therefore only those colors are considered for the classifier adjustment, which are possible skin–colors and consequently occur in the GFCC.

3.2 Motion

To avoid situations, in which the system locates faces in images hanging at a wall, motion is used a necessary feature during the initial search for a face.

Fast motion analysis is possible by determining the difference $\|q_1 - q_2\|$ of gray values of corresponding pixels in following images.

This procedure is much faster then determining the optical flow [19] but has the following disadvantages:

- The camera position and zoom lens adjustment must be constant during the motion analysis. This requirement is met during the initial search of a face.
- If an object has even gray values the motion is detected only at the shape of the object. In the low resolution images faces are adequatly structured.
- The distance of gray values is independent of the order of the image, so that the direction of the motion can't be determined. If an object moves from A to B it is only possible to determine the union of the areas A and B. The information of the actual location of the object in the later image can be determined by combining the motion information with the skin–color information of both images, because only the actual position of the face has skin–colored areas (chapter 3.4).

The object motion is considered as a necessary feature only during the initial search of a face. Once a face is located, motion analysis helps distinguishing between the face and the background, but is not necessary anymore, so that not moving faces can be tracked, too.

3.3 Coherent Objects

With the discussed methods the features skin–color and motion can be determined for each pixel independently. Neighbored pixels can be unified to a coherent object. In the following the coordinates of a pixel p are written as $x(p)$ and $y(p)$.

Def. 1: Two pixels p_1 and p_2 are *neighbored*, if $\|x(p_1) - x(p_2)\| \leq 1 \wedge \|y(p_1) - y(p_2)\| \leq 1$ is valid.

Def. 2: A set $\mathcal{S}$ is *coherent*, if

1. $\mathcal{S} = \{\mathrm{p}\}$ or
2. $\mathcal{S} = \mathcal{T} \cup p$, so that $\mathcal{T}$ is a coherent set and $\exists q \in \mathcal{T} : p$ is neighbored to q.

Def. 3: A pixel p *has the feature*

- *skin–color corresponding to a FCC*, if $\overline{N}_{r,g} \geq \alpha$, with $p = (r, g)$ and N color distribution of the FCC.
- *motion*, if the corresponding pixel q from the previous image meets $\|p - q\| \geq \beta$.

Def. 4: A set $\mathcal{S}$ *has the feature* $\mathcal{C}$, if all pixels $p \in \mathcal{S}$ have this feature.

Def 5.: An *object* $\mathcal{O}$ with the feature $\mathcal{C}$ is a coherent set of pixels with the feature $\mathcal{C}$, so that:
$\forall p \notin \mathcal{O} : \mathcal{O} \cup p$ is not coherent or does not have the feature $\mathcal{C}$.

Def. 6: The *size* of an object $\mathcal{O}$ is defined as
$\max_{p_1,p_2 \in \mathcal{O}}(\|x(p_1) - x(p_2)\| \cdot \|y(p_1) - y(p_2)\|)$.

The thresholds α and β are dependent on the hardware. Searching a face can be easily expressed with the above definitions:

> *Search the object with largest size having the features motion and skin–color.*

Because faces have similar size, selecting the largest object is identical with selecting the object most near to the camera. The definition of an object is very instable corresponding to small changes in the image, which can change a coherent object into a not coherent one. Low–pass filtering of the results of the color classification and motion analysis reduces this instability. Another approach consideres form of an object in addition and is discussed in the chapter about neural networks.

3.4 Combining the Features

The figures 3.14 (a) and (b) show two following images of a sequence. In (c) the difference is illustrated, while (d) stresses motion by drawing all pixels white, which have the feature motion. The actual position of the moving object can't be determined in this motion image. Combining the motion analysis with information about color results in the location of the object. Therefore the classifier GFCC is applied to the image (b), resulting in (e). The background contains large areas which are classified as face–like colors, so that low–pass filtering with the operator in figure 3.9 produces many pixels with feature skin–color (f). Because of the low-pass filtering single pixels with this feature are not considered anymore. The two features are combined in (g) and the largest object having both features is located in (h). All pixels belonging to the located object are replaced by their original gray value in (i).

The set of pixels belonging to the largest object is used to adjust the GFCC to an IFCC by considering the actual skin–color. The improved classifier results in less misclassifications of the background in the next image, so that the reliability of the system improves.

A high disadvantage of considering only motion and color is the neglect of form. Neck, arms, and hands may be interpreted as faces. By using the above definition of object size, faces mostly are larger than other skin–colored objects, but combined objects (for example by taking a hand in front of the face) are considered as one object. Considering the form of an object is possible by using neural networks and is decsribed in the corresponding chapter.

Figure 3.14: Locating a face

(a) und **(b)** Images in a sequence, **(c)** Differnce of the two images, **(d)** Set of pixels with feature motion, **(e)** Applying the GFCC, **(f)** Set of pixels with feature skin–color, **(g)** Objects with features motion and skin–color, **(h)** Largest object, **(i)** Located face

Chapter 4

Structure of the Face Tracking System

The camera control for the purpose of face tracking consists of two basic states with different assumptions:

1. *Searching a face*
    - The face most near to the camera is to be located. The appearance of the face is not known.
    - The position of the face in the camera image is not known.
    - The camera doesn't move and the zoom lens is set to the widest position to get an overview about the environment.
2. *Tracking a face*
    - The face found during state *search face* is to be found again in the actual camera image. The appearance of the face is now known.
    - The position of the face can be assumed to be near to the position in the last image. The maximum distance is dependent on the maximum motion of the person, the distance of the person to the camera, the image frequency, and the adjustment of the zoom lens.
    - The camera and zoom lens are frequently in motion to maintain optimal adjustments.

Both basic states can use the methods described in chapter "Features for Locating Faces" and are explained in the following. Afterwards the criteria used for switching between the states are listed.

4.1 Searching a Human Face

The human eye has only in a small part of the center of the retina (fovea centralis) the ability to distinguish between objects in a high resolution. Only in this area an individual is able to detect colors. Outside the fovea centralis the retina is much more sensitive for motion and light [4]. Therefore interesting objects are brought into the fovea centralis by eye motion for further examination. This method reduces the amount of image data without loosing important information. The face tracking system uses a similar method to reduce computation data. After examining the entire image for faces, a virtual camera representing the fovea centralis is used to focus on one object. The largest of all located faces is assumed to be nearest to the camera and most important for communication. Other possibilities to select one of several located faces for further tracking could include directions of detected speech signals, etc.

4.1.1 Detecting Faces

The used features for locating arbitrary faces are skin–color and motion. The classifier GFCC detects all colors occuring as skin–colors in human beings. Motion is detected by determining the difference between two following images. This simple method is fully sufficient, because the camera is without motion during searching for faces and a stable background can be assumed. All parts of the image containing skin–color and motion are examined for coherent objects (chapter 3.4).

4.1.2 Selection of an Object

Figure 4.1 shows all located objects with motion and skin–color in one image. Because the background of the left part of the image contains many colors classified by the GFCC as skin–colors, the left face is estimated too large. Considering motion can help reducing this error. Because all areas covered by the moving object in the actual image and the previous image are considered as areas with motion, the shape of the motion area is different from the object's shape. After adjusting the color classifier to the colors

occuring in the face, much fewer colors are classified as skin–colors and the mentioned error can be mostly avoided.

Figure 4.1: Loacated Faces

To avoid errors in selecting a face, all faces are observed for several following images. If a constant largest object can be detected over a longer time period, this face is selected and the system continues in state *track face*.

4.2 Tracking of Faces

Searching faces in the entire image is time consuming and unnecessary for tracking a face. If the position of a face is known in the actual image, it can be assumed near to this position in the following image. For this reason the face is searched only in a small area around the last position of the face. The maximum distance of positions of the same face between two following images is determined by the motion speed, the distance of the face to the camera, the image frequency, and the adjustment of the zoom lens. A smaller area used for searching reduces the computation time and results in a higher image frequency. If the images can be grabbed in smaller time steps, the maximum distance between face positions is reduced, so that the area used for searching can be further limited. Good results could be achieved with using an area of double face dimension for searching the face

in the next image. Because this area can me moved and changed in size, it is considered as a "virtual camera".

4.2.1 The Virtual Camera

Figure 4.2 illustrates the function of the virtual camera. While moving a physical camera and adjusting the zoom lens is time consuming, the virtual camera can be adjusted to new settings without delay. If a face attends to move out of the image, the physical camera is adjusted as soon as the virtual camera touches the border of the area called "movement zone" in the figure. The physical camera is then moved to a new position, so that the face becomes the center of the image. In the same way the physical zoom lens is adjusted if the virtual camera exceeds certain dimensions or becomes too small. Using the virtual camera, stable images of the face can be extracted without delay and mechanical movements are reduced as far as possible.

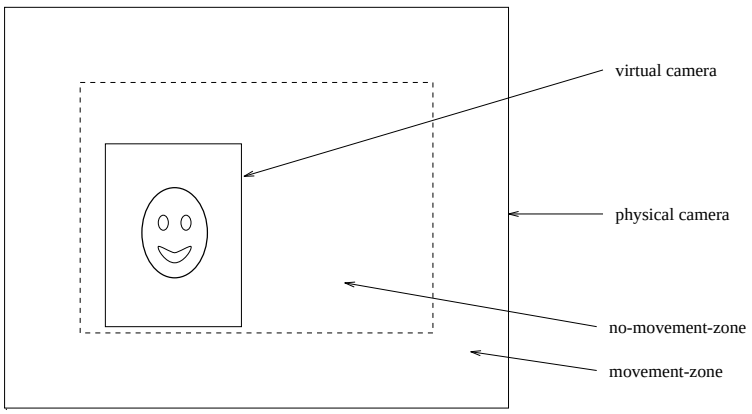


Figure 4.2: Virtual camera

The advantages of using a virtual camera:

- Increased image frequency, because of reduced calculation time.
- Reduced mechanical movements, resulting in a constant background enabling simple motion analysis.
- The center part of the virtual camera contains the face in each image. By using this subimage as output of the entire tracking system, the

stable image of the tracked face is easily available and can be used for example in lip reading systems. If the low resolution of the face is unsatifying, the area covered by the virtual camera can be grabbed with another framegrabber in higher resolution. In this case the output of the system consists only of the coordinates of the virtual camera and are transmitted to another computer. The second computer grabbes a subimage corresponding to the transmitted coordinates in a higher resolution. By interpolating the transmitted coordinates, the images can be grabbed with a higher image frequency.

The size and position of the located face determines the dimensions of the virtual camera. The virtual camera is set around the center of the located face with twice the dimensions of the medians of the last three values for width and height of the face. This way one poor size estimation does not result in a severe error. The positions are neither averaged nor estimated in advance. These methods achieve only improvements with constant motion. In situations of a fast change of direction an estimation of the position in advance would result in searching the face at the wrong position.

For detecting of motion the actual image is compared to the previous image. Instead of reading the entire image only the subimage determined by the virtual camera is grabbed. During changing the settings of the virtual camera, following images don't cover the same area of the entire image and are not suitable for direct comparison. Instead after finishing a physical camera movement an image of the entire scene is taken, which is called background image. This image is assumed to be constant until a new movement of the camera is performed. For motion analysis the subimage of the virtual camera is compared to the corresponding area of the latest background image. The following situations require the background image to be grabbed again:

- The physical camera moves or the adjustment of the zoom lens changes.
- Parts of the background image don't correspond to the actual scene. This happens if the background image contains an moving object. The situation can be detected by an unusually high percentage of changing pixels and a new background image is grabbed.

4.2.2 Camera Calibration

If a camera movement is required, the step motors turning the camera in horizontal and vertical direction are controlled by transmitting the new tar-

get polar coordinates. These polar coordinates must be determined from image coordinates. Assuming a movement of the camera is required, so that a person at the right margin of the image is shifted to the center of the image. Then a camera moving covering half the image width must be performed. To calculate the corresponding polar coordinates for the step motor, the relation between image size and polar coordinates is important. This relation depends on the zoom lens adjustment. The wider the view of the camera, the wider the room angle covered by the image.

The camera calibration therefore consists in determining a list of zoom lens adjustments and the corresponding relations between image size and room angles. Because the camera doesn't support polling the zoom lens adjustment by hardware, a software approximation is used. After initializing the zoom lens by zooming out to the physical limit, the milliseconds zooming in and zooming out are subtracted from each other. This value is always equal to the required time, the zoom lens has to zoom in after initialization to reach the actual adjustment.

The above list of relations can be obtained automatically. An even colored, rectangle sheet of paper is fixed at a wall and is used for the calibration. The color of the paper is automatically measured and used for identifying the paper margins. After zooming out as far as possible, the camera is adjusted to the direction of the paper. While turning to the right, the angle between two points is measured: the point where the right margin of the paper becomes visible in the camera image, and the point where this margin vanishes at the left margin of the image. The same procedure is repeated for obtaining vertical relations. After zooming in one second, the entire process is repeated. For each zoom step the independent horizontal and vertical values are inserted in the relation list until the camera is zoomed in to the maximum adjustment.

4.2.3 Using of Motion While Tracking a Face

Considering the motion of an object helps distinguishing between the object and the background. During tracking a face, the camera frequently moves and the zoom lens adjustment changes, so that the background is not constant anymore. In this case motion is not considered.

If the motion analysis fails in detecting a moving object, motion is not considered as a necessary feature, so that not moving faces can be tracked, too. Motion is a necessary feature only during the initial search of a face, but not during tracking.

4.2.4 Adjustment of the Color Classifier

Once a face is located, additional information about this specific face is available. With a color distribution of the area covered by the face, the color classifier can be adjusted to the colors actually occuring in the face (chapter 3.1.4). By repeating this procedure for every image, the classifier adjusts automatically to changing lighting conditions.

4.3 Structure of the Entire System

Figure 4.3 illustrates the structure of the entire system. A color classifier is applied to the actual system. During the initial search of a face the GFCC is used as classifier, during tracking the adjusted IFCC. The motion analysis is obtained by comparing the actual and the previous image. Color and motion information are combined and used for detecting the largest, coherent moving object with skin–color, which is assumed to be the face. The virtual camera settings are adjusted to the position and size of the located face. If the position or size of the virtual camera exceeds its limits, the physical camera is moved or the zoom lens is adjusted. The pixels belonging to the located face are used to adjust the color classifier to the specific face. Alternative to detecting the largest coherent object, neural networks can be used. These networks and their training are described in chapter “Artificial Neural Networks”.

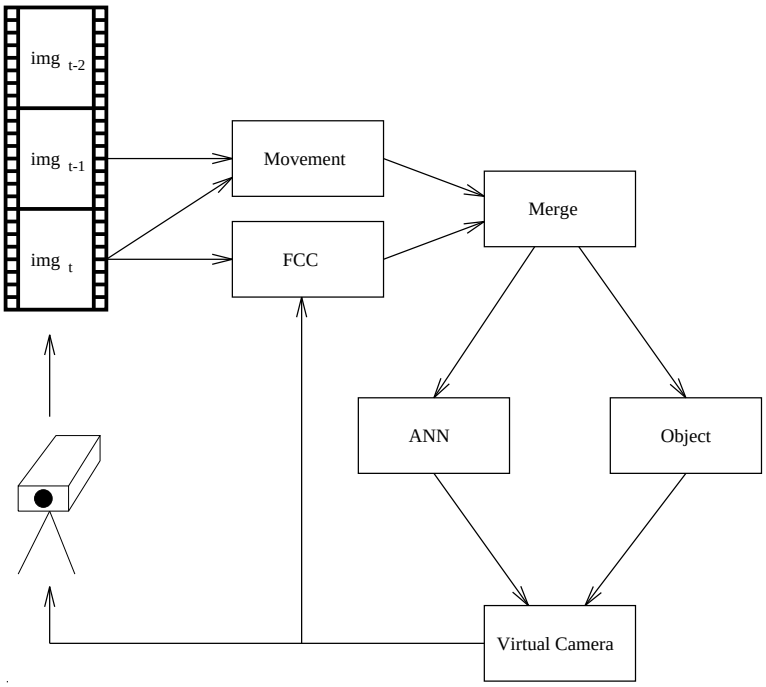


Figure 4.3: Structure of the entire system

4.4 Switching between the Basic States

The system starts in the basic state *search face* and switches to *track face* after successfully locating a face. If the system looses track of a face, it switches back to the first state. Loosing track of a face can be detected in the follwing situations:

- No moving skin–colored object can be detected in the virtual camera image.
- The detected object is very different in size to the object found in the previous image.
- The color distribution of the located object differs significantly from the IFCC.

After meeting at least one of the above requirements for three images the system switches back to the state *search face*.

4.5 Example of a Face Tracking Sequence

The following pages illustrate an example sequence. The first page shows in figure 4.4 (1–4) the state *search face*. The frame with the largest dimensions covers the located object, while the smaller frame is manually drawn to mark the actual face. These markings are used for evaluation and are explained in the chapter "Evaluation". In the first three images another small frame is visible, indicating another found object. Actually this second object is part of the hair containing colors, which are misclassified as skin–colors. After locating the same objects in four images, this object is selected for tracking (figure (5)).

The following images show the state *track face*. The largest frame indicates the area of the virtual camera, the middle frame shows the located object, and the smallest frame indicates the manual markings. Though only the virtual camera image is read, the entire image can be obtained by superimposing the virtual camera image with the constant background image. The position and size of the virtual camera is adjusted in each image. The figures (15–19) show the changing of the background image after several readings of the virtual camera. In figure (20) this error is detected and the entire background image is grabbed again.

The image sequence 4.7 shows the output of the tracking system, the stable image of the observed face. The image numbers correspond to the image numbers of the tracking sequence. As described in chapter 4.2.1, the image resolution can be increased by using a second framegrabber.

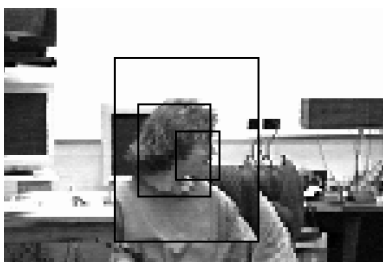
(1) (2)

(3) (4)

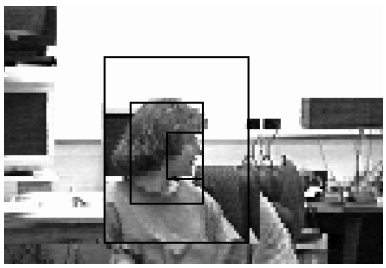
(5) (6)

(7) (8)

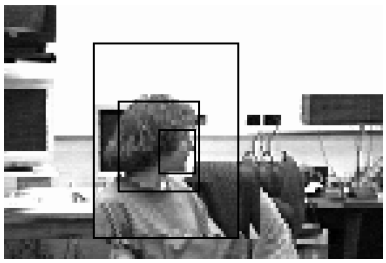
Figure 4.4: Example sequence, part 1



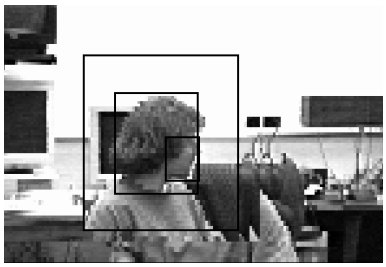
(9)



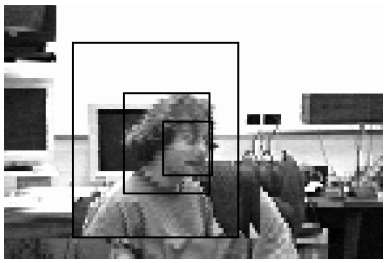
(10)



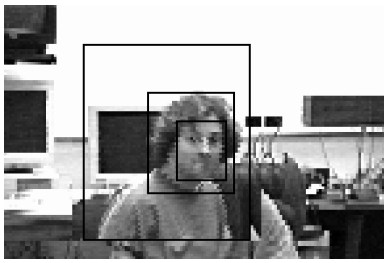
(11)



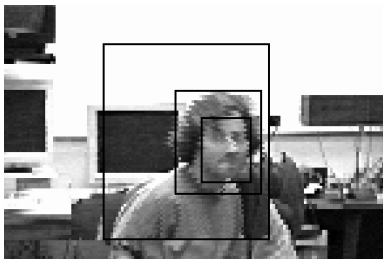
(12)



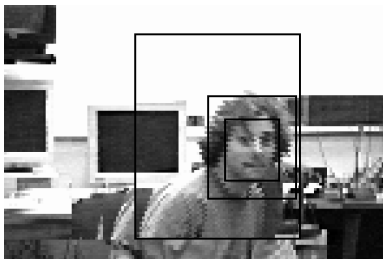
(13)



(14)

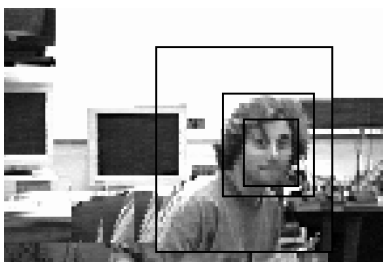


(15)



(16)

Figure 4.5: Example sequence, part 2

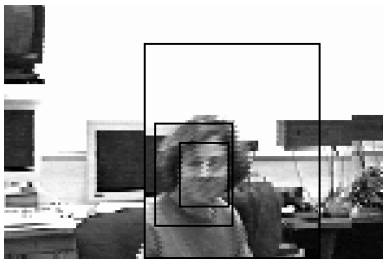


(17)

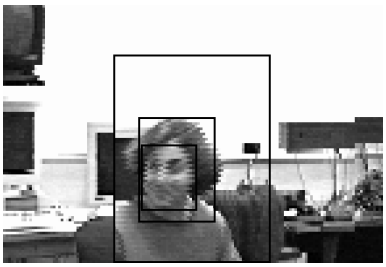
(18)

(19)

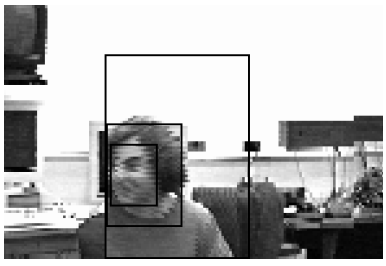
(20)



(21)



(22)



(23)

(24)

Figure 4.6: Example sequence, Part 3

Figure 4.7: Located faces

Chapter 5

Artificial Neural Networks

The procedure *search for the largest coherent object* does not use the shape of an object. All objects with the features movement and skin–color are classified as faces, no matter how the shape of the face looks like. To reduce the number of misclassifications a further examination of the shape could identify arms as long and not round objects, etc. The neural networks presented in this chapter are used for this purpose and replace the simple procedure for detecting coherent objects.

All neural networks used in this system are multilayer perceptrons and are trained with back–propagation. In the following an introduction to the purpose, basic structures, and the training of these networks is presented and the used network topologies are discussed. The networks without any preprocessing except normalizing of the input data achieve results, which are not satisfying for the task of locating human faces. Using the color classifiers from chapter 3.1.4 as a preprocessing procedure leads in a highly improved performance. In addition these networks are independent from lighting situations and the used hardware, so that a retraining of the networks is not necessary, even if these parameters change significantly.

5.1 Task and Training of the Networks

All networks used in this system consist of an input layer (retina), a hidden layer, and an output layer. The input layer receives data from the camera image. The output layer gives informations about the position and size of a located face within the retina, which are used to maintain an optimal position of the camera and adjustment of the zoom lens. In addition the

output layer is able to represent the situation in which no face is visible. This information is used to switch between the two basic states of the system *search face* and *track face*.

A basic structure for a multilayer perceptron designed for this task is illustrated in figure 5.1. The ALVINN–project [13] demonstrated the ability of this network to stear a car automatically.

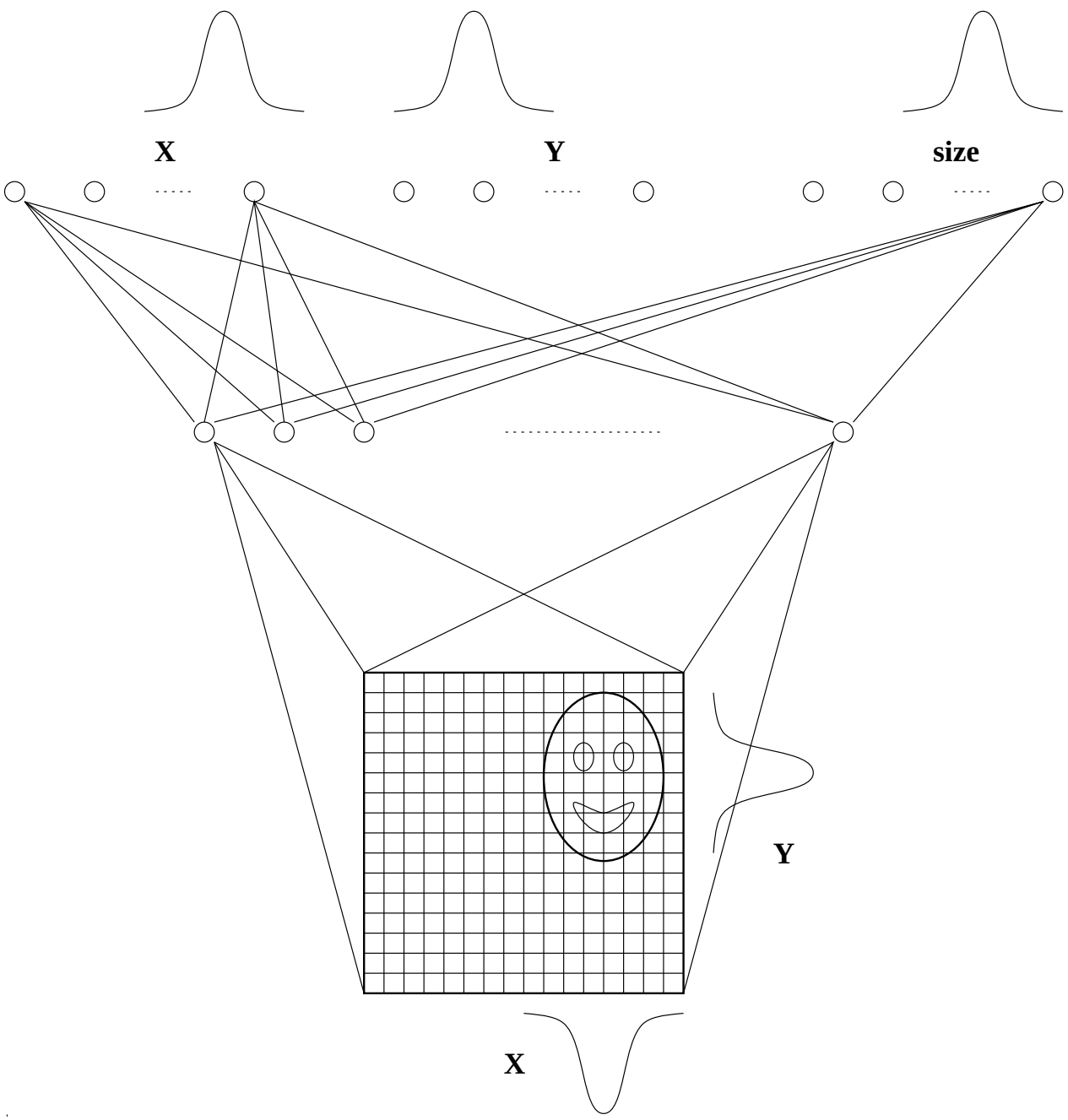


Figure 5.1: Artificial neural network for face locating

The creation of the training set, which is necessary to train the neural network, is described in chapter 6.1.

5.1.1 Representation of the Net Input

The activations of the retina units are determined by the color or gray values of the virtual camera image, which is scaled to the size of the retina. The following preprocessing methods are examined:

- *Normalized gray values:* The averages of the three color values for red, green, and blue are scaled, so that the lowest value becomes -1 and the highest 1. This normalization is necessary, because the gray values of the images are concentrated in a small area.
- *Normalized color values:* The color values are projected to the chromatic diagram, reducing three values into two ($q = (r, g)$). The number of input units must be doubled compared to the usage of gray values.
- *Color classification:* A FCC classifies the color as skin–color or background color. The output of the classifier is scaled between -1 and 1.

5.1.2 Representation of the Net Output

Some basic data representations are examined concerning their suitability for the given task. The following discussions refer to ideal output data, which are used as dersired outputs in the training of the network.

Single output neuron

This representation does not use all available information from the hidden layer and may lead to errors. Assumed one hidden unit extracts features for a face in the left part of the image and another one for a face in the right part. If the output unit gives the horizontal position of a face, the first hidden unit tries to minimize the activation of the output unit and the second hidden unit tries to maximize this value. Altogether the summed activation claims a face in the middle of the picture, instead of deciding for one of the two faces.

Discrete distribution

This representation uses several output units to represent one output value. Each of the units represents one discrete output value. During the training of the network only that output unit is set to 1, which is nearest to the desired output value. All other units are set to 0. To represent the size of a face, three output units could be used, representing the information *too small*, *optimal*, and *too big*. In the application of the network the output unit with the highest activation specifies the entire output value. If all activations don't exceed a threshold, no face has been located. The recognition rate using this representation method is significantly smaller compared to the gaussian distribution, which is described next. The main disadvantage of the discrete distribution is the possibility of strong changes in the output units given only small changes in the input units, which take place at the change of an output class. The network can't adopt to this unsteady behavior.

Gaussian distribution

Instead of using a 1 of N distribution for the desired output units, the desired output values are calculated with a gaussian distribution (figure 5.2). In the application of the network the output unit with the highest activation determines the overall output. This representation allows a smooth changing between distinct output classes.

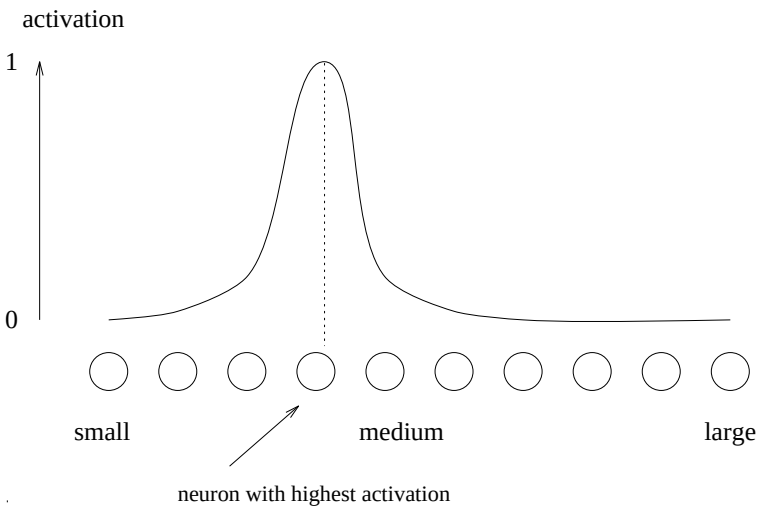


Figure 5.2: Output representation using gaussian distribution

Consequently in all following networks the gaussian distribution is used for output representation.

5.1.3 The Back–Propagation Method

A more detailed description of this material is given in [10]. One training step consists of the selection of a training sample and its application to the network. Each training sample is a pair of given input and desired output data. The input data is copied to the input layer of the network and the activations of the output units are determined and compared to the desired values. Corresponding to the difference between actual output and desired output the weights of the network must be changed, so that the difference gets smaller. In the next training step another training example is used until all examples have been applied to the network in random order.

To verify the ability of the network to generalize the learned knowledge, after each training of all examples an independent test set is applied to the network. When the performance on this test set has reached its maximum, it's a sign to stop training. Continuing the learning process would result in overlearning of the examples. The performance of the network finally must be evaluated with another independet validation set (crossvalidation [17]).

The neural network consists of neurons arranged in layers. The input of each neuron j is coherent with the outputs o_k of all neurons k of the previous layer. The connection from neuron k to neuron j is named w_{kj}. The neuron j receives an activation a_j corresponding to 5.1. The output o_j of this neuron is defined by 5.2.

$$a_j = \sum_k o_k w_{kj} - \Theta_j \tag{5.1}$$

$$o_j = f(a_j) \tag{5.2}$$

Corresponding to the sign of the weight, a connection can be inhibitory or exhibitory. The treshold Θ_j can be considered as an additional unit coherent to the constant value -1. The function f is defined as the the sigmoid function

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{5.3}$$

After copying the training examples to the input neurons of the input layer, the activations and outputs of these neurons can be calculated. The

outputs of each following layer can then be determined by using the outputs of the previous layer multiplied with the corresponding weights as inputs. After comparing the outputs o_j of the last layer to the desired outputs d_j, the error can be calculated.

$$E = \frac{1}{2}\sum_j (o_j - d_j)^2 \tag{5.4}$$

The derivation of the error can be used to change the weights of the output layer, so that the error itself becomes minimized:

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial o_j} \cdot \frac{\partial o_j}{\partial a_j} \cdot \frac{\partial a_j}{\partial w_{kj}} \tag{5.5}$$

The single derivations are easy to determine:

$$\frac{\partial E}{\partial o_j} = o_j - d_j \tag{5.6}$$

$$\frac{\partial o_j}{\partial a_j} = f'(a_j) = o_j \cdot (1 - o_j) \tag{5.7}$$

$$\frac{\partial a_j}{\partial w_{kj}} = o_k \tag{5.8}$$

Hence:

$$\frac{\partial E}{\partial w_{kj}} = (o_j - d_j) \cdot o_j \cdot (1 - o_j) \cdot o_k = \delta_j \cdot o_k \tag{5.9}$$

The correlation between a change of the input activation of neuron j and the total error is given by:

$$\delta_j = \frac{\partial E}{\partial a_j} \tag{5.10}$$

The contribution of an arbitrary weight w_{kj} to the total error is defined by δ_j and the output o_k of the previous neuron. By determining the term 5.6 by the errors of the following layer, the value δ_j can be calculated for each layer:

$$\frac{\partial E}{\partial o_j} = \sum_h \frac{\partial E}{\partial a_h} \cdot \frac{\partial a_h}{\partial o_j} = \sum_h \delta_h w_{jh} \tag{5.11}$$

By applying these equations to all layers beginning with the output layer, the contribution of each neuron to the total error can be determined. The weights must be changed corresponding to the gradient:

$$\triangle w_{kj}(t+1) = -\eta \frac{\partial E}{\partial w_{kj}(t)} \tag{5.12}$$

An additional momentum α adds a part of the last weight change to the actual change.

$$\triangle w_{kj}(t+1) = \alpha \cdot \triangle w_{kj}(t) - \eta \frac{\partial E}{\partial w_{kj}(t)} \tag{5.13}$$

This is especially useful with a small derivation of the error.

During training it's useful to reduce η and to raise α. Because the training can get stuck in a local minimum, it must be repeated several times with different weights in the beginning.

5.2 Neural Networks without FCC

Table 5.1 shows the recognition rates for the network in figure 5.1. The recognition rate is defined by the percentage of correct classified images in the validation set. This set contains 3000 images of faces, none of which were used for the training. Estimates of face position allows 10% deviation of the actual position related to the head size and estimates of face size allows 20% deviation of the actual head size. All networks are trained with faces at all positions and with sizes between $\frac{1}{4} \ldots \frac{1}{2}$ of the retina height. Some examples of gray scale images in this resolution are shown in figure 6.6 (a).

In the following the used networks are enumerated. The best results have been made with 40 hidden neurons.

1. *Net 1:* The network using *normalized gray values* as inputs converged badly.

2. *Net 2:* To reduce the number of weights a smaller retina is used. The network is estimating only the size of the face, not the position. Therefore this task is much easier. For the training of this network only faces are use, which are shifted around the center of the image less than 50% of the head size.

3. *Net 3:* Same as network 2, but using *normalized color values* as inputs.

net	retina	output	number of weights	recognition rate
1	32x32	x, y, size	40000	32%
2	24x24	size	23000	65%
3	2x24x24	size	46000	68%

Table 5.1: Neural networks without FCC

The small improvement by using color values is surprising. Either color is redundant or the network is not able to make use of the additional information. It's unlikely, that color is not useful to locate faces. More likely the network has too many input units and can't extract the important features. This explanation is supported by the following aspects. If the network is trained with less than 10000 examples, it's ability to generalize is rather poor, while it could easily learn the given examples. On the contrary a large training set improves the ability to generalize, but the network converges badly or doesn't converge at all.

5.3 Neural Networks with FCC

Assumed the network using *normalized color values* would achieve a good performance, it still would have a significant disadvantage. The trained weights of the network would have learned the color environment given in the training examples. After changing the lighting situation, the camera or the framegrabber the network would have tobe retrained. Another possiblitily is the usage of several networks, each trained for a special lighting situation. All networks together must be able to handle the range of color shifts illustrated in table 3.1.

These problems can be avoided by using the FCC, which concentrates all dependencies of color values in one central part of the system. The network is integrated in the system by replacing the procedure *"search the largest coherent object"* corresponding figure 4.3. Instead of taking *normalized color values* as input of the network, a combination of the FCC and the motion analysis is used. In addition the normalized gray values are given to the network. The figure 6.6 shows images with gray values (a) and after applying the GFCC (b).

The network considers the form of an object, rather than just finding the largest coherent object. In some situations this behavior results in a highly improved performance (chapter 7.2.2).

Figure 5.3 shows the network structure using a FCC as preprocessing. Actually the network can be considered as two networks. The bottom network locates the x and y position of a face, using a 16x16 retina. Given a located face, a subimage of 12x12 pixels containing the face in the center is used by the upper network to estimate the size of the face. The second network receives only centered faces, so that the task is much easier than to estimate the size of arbitrary positioned faces.

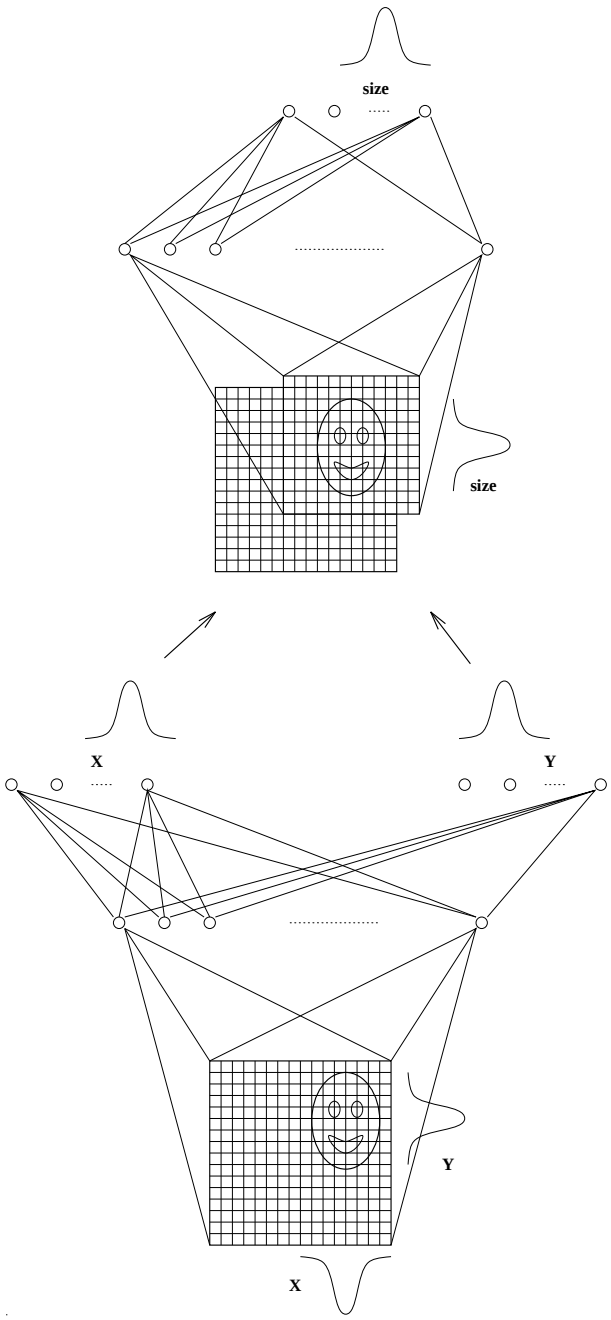


Figure 5.3: Seperate networks for estimation of position and size

This network converged significantly faster than the networks without using a FCC and achieved the best performance with only 5000 training images. Table 5.2 illustrates the recognition rates, which have been achieved with 40 hidden units:

net	retina	output	number of weights	recognition rate
1	16x16	x, y	21000	93.6%
2	12x12	size	12000	95.2%

Table 5.2: Neural network with FCC

Chapter 6

Artificial Images and Image Sequences

In this chapter a method is developed for creating artificial images, so that the position and size of faces in each image are known. These images are used for training neural networks as well as for the evaluation of the performance of the system in special situations.

After explaining the importance of the creation of images for both applications the method will be illustrated.

6.1 Training of Neural Networks with Artificial Images

The used neural networks are trained with a large set of examples which contain each a net input and a desired net output. The net input is organized as a two–dimensional retina. The camera image is preprocessed and projected onto this retina. The net output contains the size and position of a face within the image. The function and structure of the networks are explained in chapter 5.

6.1.1 Training Set Requirements

To enable a good generalization behavior of the network the training set should meet the following requirements:

1. **Number of training examples:** There should be more examples than connections in the entire network. In the used network structures

this leads to a minimum of several thousand examples.

2. **Representative net inputs:** The training set must contain faces with varying age, sex, skin color, hairstyle, line of vision, etc. to avoid the learning of a individual face or a group of faces with common features.

3. **Using all possible net outputs:** The training set must contain representatives for all possible output classes.

6.1.2 Assigning of Net Inputs to Net Outputs

There are two methods to create pairs of net inputs and outputs for examples of the training set:

1. Finding the corresponding net inputs given the net outputs.

2. Finding the corresponding net outputs given the net inputs.

Both methods are evaluated concerning the requirements for the training set.

Net Inputs as Given Data

1. **Number of training examples:** A high number of images can be obtained by recording sequences with a camera. The difficulty is to specify the size and position of faces in these images with a practicable method. Optical markings can't be used because they would be visible on the training images and therefore serve as a feature during the training of the network. Consequently markings must not be optical or filtered out before projecting the image to the net input.

2. **Representative net inputs:** The image sequences must be recorded with a large number of persons and backgrounds. If the method of marking the faces is limited to a special environment, the training examples have undesired common features like lighting situation and background colors.

3. **Using all possible net outputs:** The sequences must contain all possible head positions and sizes with similar frequency. This is difficult to achieve in recordings.

Net Outputs as Given Data

This method requires a possibility to create images containing heads of given size and position. These images can be created based on a database containing example images.

1. **Number of training examples:** Because the images are created artificially, it is possible to obtain an arbitrary large training set.
2. **Representative net inputs:** The complexity of the data base containing example faces determines the complexity of the net inputs. For each face in the data base an arbitrary high number of images containing this face can be created.
3. **Using all possible net outputs:** To each possible output class a given number of images are created.

The above method meets the training set requirements easily.

6.2 Image Sequences

In chapter 6.3 a method is illustrated, which is able to superimpose given faces and backgrounds. This method can be used to create image sequences with

- a desired moving sequence of the head.
- a common foreground but different backgrounds.
- a common moving sequence of the head but different individuals.
- different distances in the moving sequence between following images to simulate different motion speeds.

The sequences allow a variety of possibilities to evaluate the performance under special circumstances. Nevertheless the evaluation with artificial images can only show tendencies of the system behavior, because artificial images have differences to real images from camera recordings. These differences are explained in chapter 6.3.

6.3 Creating Artificial Images

The following method makes it possible to create a training set, which fulfils all requirements of section 6.1.1. The method uses two databases, which contain representative faces and backgrounds. To each given position and size a face selected from one of the databases will be superimposed with a random part of one of the backgrounds.

6.3.1 Creation of the Databases

To scale a face from a database to a given size, the original size of the face must be known. In addition the position of the face must be known to shift the face to a given position. The easiest way to obtain this data is to normalize the size and position of the face during the recording. For this purpose a part of the camera image is displayed on the screen (figure 6.1 (a)). The position of the camera and the zoom lens are adjusted, so that the face is in the middle of the displayed image and the hairline and the chin are touching the horizontal bounds of the subimage. Afterwards an image twice with double dimensions is stored (figure 6.1 (b)). The recording must be done before a special background, the blue–screen, which is decribed in the next section.

(a) (b)

Figure 6.1: Recording of a face

(a) The head is in the middle of the subimage and hairline and chin are touching the horizontal bounds, **(b)** Recorded image

Figure 6.2 shows some faces which have been recorded with the above

method. All faces have consequently the same size (more exactly height) and position. The database for faces contains 24 persons with each three different directions of the head to make the network independent of head directions. Altogether 72 RGB–images have been taken with a resolution of $256 \cdot 256$ pixels.

Figure 6.2: Example faces in a database

An additional database is used containing 10 background images with a resolution of $600 \cdot 460$ pixels. The backgrounds cover the entire walls of the lab and do not contain any faces. Because the network should be optimized for the working environment, which is the lab, no backgrounds from other places are used.

6.3.2 The Blue–Screen Method

The faces in the database must be shifted to a given position and be scaled to a given size. To superimpose the face with a new background it is necessary to know the exact shape of the face. For this purpose the blue–screen method is ideal, which is known as a frequently used technique in television. The database contains faces recorded in front of an even blue background, so that all pixels in the image with color blue are classified as background. The color blue is selected, because the colors red and green appear stronger in faces.

With the techniques developed in chapter "Features for Locating Faces" the background can be easily eliminated (figure 3.2). Figure 6.3 shows the color distribution of the blue background. The black dot indicates the most frequent color, which is the the blue of the background. The neighbored, more seldom colors result from reflexions. All colors appearing in the color distribution in figure 6.3 are classified as background colors. Figure 6.4 shows an example of the database, in which all background colors are marked black and all other colors are marked white.

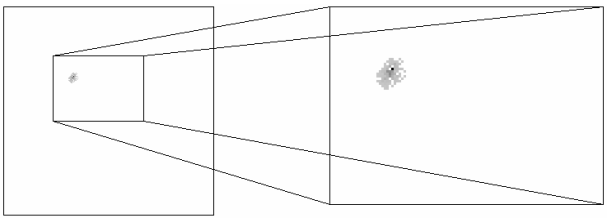


Figure 6.3: Color distribution of background

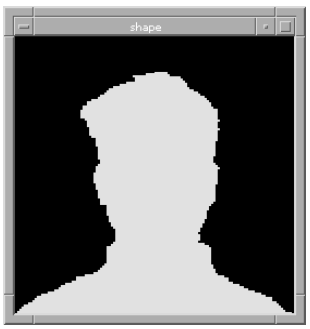


Figure 6.4: Extracting the shape of a face

6.3.3 Creation of an Artificial Image

After extracting the shape of a face with the blue–screen method, the face can be scaled and shifted to the desired size and position. The background of the artificial image is built by a subimage of one of the background images within the second database. The area covered by the subimage is determined by random. After scaling this subimage to the size of the retina of the neural network, the scaled and shifted face can be superimposed. To avoid a sharp

transitions between foregound and background, the shape ges smoothed, so that these transitions can't be used as a feature during the network training. In real camera images it frequently occurs, that only face and shoulders are visible. Especially when a person is sitting behind a table, the real images are very similar to the artificial created images, though these contain only faces and no bodies. In figure 6.5 the method is illustrated.

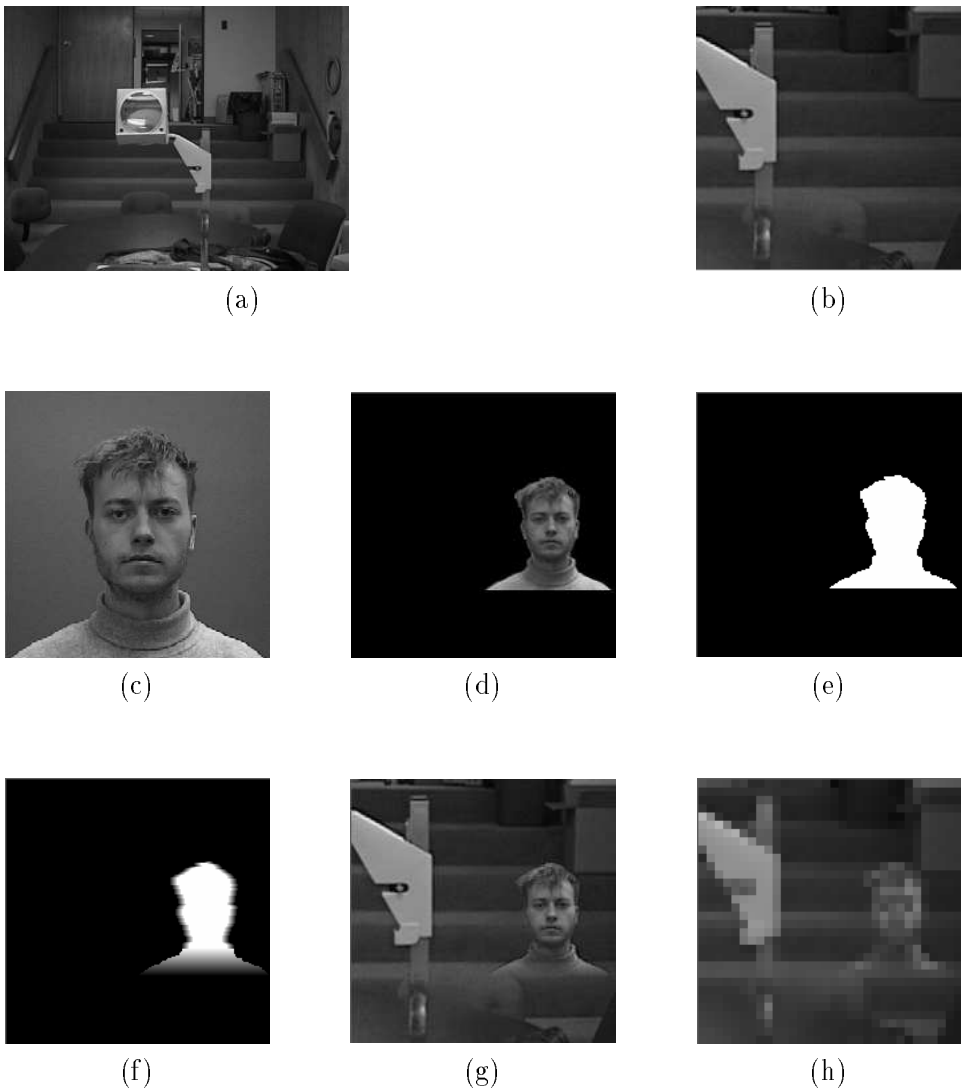


Figure 6.5: Creating an artificial image
(a) Random selection of a background (database 2), **(b)** Random selection of a subimage, **(c)** Face from database 1, **(d)** Extracting of shape, scaling, and shifting corresponding to given data, **(e)** Mask indicating face pixels, **(f)** Smoothing transitions, **(g)** Superimposing (b) and (d) corresponding to the mask, **(h)** Reducing resolution to size of retina

Figure 6.6 (a) shows some example images created with the above method in a resolution of $24 \cdot 24$ pixels. Figure (b) shows the same images after

applying the GFCC. The task of locating faces in these images is much easier, because of the preprocessing of color information turns out to be extremely helpful.

(a) (b)

Figure 6.6: Examples of artificial images
(a) Artificial images, **(b)** Application of the GFCC

Chapter 7

Evaluation

An objective evaluation of the system performance is possible by comparing the actual head position and size with the data produced by the system. The recording of test sequences for this purpose, the evaluation, and an assessment of the neural networks are described in this chapter.

7.1 Test Sequences

In chapter 6.2 a method is described to create artificial image sequences, so that the position and size of faces in each image are known. For several reasons these artificial images are different to camera images:

- The artificial images contain only faces, but no bodies. In real image sequences a complete body may be moving, so that the face is only a small part of the moving area.
- In the artificial images smooth changings of head directions or lighting situations are not simulated.

To have strong evidence in the evaluation only real image sequences are used to measure the system performance.

7.1.1 Recording of Test Sequences

The recording system of the camera records tracking sequences on tape while running the complete tracking system and actually tracking a face. Therefore the recorded sequences show actual applications of the system and are

suited for evalutation. Afterwards the camera plays back the sequences and the images are grabbed 10 times per second and stored on harddisc. To each image the position and size of a face are marked manually. For evaluation the tracking system uses these sequences as input instead of camera images. The position and size of the face estimated by the system are compared for each image to the manually marked data.

Though these sequences contain images from real scenes, their playback can't fully simulate an actual application of the system. Indeed the system achieves better results on real data than on the test sequences, because of the following reasons:

- Recording the images on tape makes worse the image quality. Occuring stripes especially with moving objects lead in lower localization accuracy.
- After loosing track of a face the system switches back to the state *search face*. In an actual application the camera is stopped in this case to use the feature motion to find the face again. In a recorded sequence it is not possible to control the camera. Therefore the system must wait until the background stops moving in the recorded sequence.

When processing real camera images the system grabs the next image immediatly after finishing all calculations on the last one. With the used hardware an average image frequency of 5 images per second could be achieved. A recorded sequence is grabbed with 10 images per second. Only every second image is used for evaluation purposes to simulate the real time image frequency. If all images are used, a higher hardware performance can be simulated to get an idea, how the system would improve with faster hardware.

7.1.2 Manual Marking of the Test Sequences

The manual marking of a face can be done easily by drawing a rectangle with the mouse device around the face. The face area is defined as the skin–colored part between chin and hairline. A difference of these localization data with the data determined by the system must be expected. The manual localizations vary with 10% of the face size if two different persons independently perform the marking procedure. Differences within this range therefore aren't considered as mistakes.

7.1.3 Selection of test sequences

The localization accuracy highly depends on the following complications:

- velocity of head motion,
- frequency of changing backgrounds by moving camera or adjusting zoom lens,
- and the occurence of areas containing colors which are classified as face–like colors by the GFCC or even by the IFCC.

Consequently the evaluation is done on two test sequences with different complications:

- *Test sequence 1:* Six persons with different skin–color were recorded in front of backgrounds, which don't contain skin–like colors. The persons performed long range movements, so that the camera was forced to frequent movements.
- *Test sequence 2:* Another person was recorded in front of a background containing many colors, which are classified by the GFCC as face–like colors. The person performed movements with less speed and shorter range than in the first sequence.

All persons in both test sequences were asked to move freqently and to remain without motion in less than 5% of the time. Each person stood up at least one time in the sequence, brought his/her hand to the chin several times and moved so strongly, that the camera was forced to frequent movements (more than 70% of the time). These strong movements appear seldom in normal communication situations, but were used for evaluation to determine the limitations of the system. The movements occuring in real communications of two sitting persons seldom require camera adjustments and are managed by the system mostly error free.

The average and maximum movement of a face relative to the face size of all sequences is measured to receive assessable values about face motion. The values given in table 7.1 relate to an image frequency of 5 frames/second. The item $y_{max} = 1.88$ for example shows that the maximum movement in vertical direction is 1.88 times the face height.

The head motion relative to the headsize determines if the head is visible in the area of the virtual camera in the next image of a sequence. The virtual camera has twice the dimensions of the face. Therefore the face can

test sequence	number of images	$\overline{x}$	$\overline{y}$	x_{max}	y_{max}
1	1860	0.19	0.06	1.52	1.88
2	460	0.19	0.05	1.26	0.56

Table 7.1: Used test sequences

be expected fully within the virtual camera of the following image only if the horizontal and vertical motion values are less than 0.5. If one of the values is higher than 0.5, a part of the face is invisible. If one of the values is higher than 1.5, the face is completly invisible. The average motion values are far below these limitations. Vertical movements mainly occur while individuals stand up, leading in a high discrepancy between the average and maximum vertical movements. In test sequence 1 several times a movement with value more than 1.5 occurs, so that the system compulsorily looses track of the face.

7.1.4 Evaluation of a Test Sequence

For the purpose of evaluation following statements are important:

1. In how many images the face could be located correctly, ie. how reliable is the system?
2. How large is the systematic error? The systematic error is defined as the average difference between the actual data and the data determined by the system.
3. How large is the standard deviation of these differences?

The above items are discussed in the following.

Reliability

If fast movements occur or the background contains colors which are misclassified as face–like colors, the system can loose track of the face. In this case the system switches back to the basic state *search face*. The value *correct* shows the percentage of time the system is in state *track face* and performs correct data. The remaining time is spent by searching again for a face in the state *search face*. The first search for a face after initializing the system

is not considered. The value *correct* is important for the combination of the tracking system with other systems for extracting visual features. The value specifies how much of the time a stable image of the speaker's face can be expected and therefore the reliability of the system.

Systematic Error

Four systematic errors are determined for each test sequence: s_x and s_y for the position, s_w and s_h for the size of a face. Given the actual position and size of the face (x_{target}, y_{target}, w_{target}, and h_{target}), and the values determined by the system (x_{output}, y_{output}, w_{output}, and h_{output}), the systematic errors are defined as the arithmetic average over the following values of all images in the sequence:

$$s_x : \frac{x_{target} - x_{output}}{w_{target}} \qquad s_w : \frac{w_{output}}{w_{target}}$$

$$s_y : \frac{||y_{target} - y_{output}||}{h_{target}} \qquad s_h : \frac{h_{output}}{h_{target}}$$

Standard deviation

The standard deviation of the locating error is another important aspect of the system's performance. A high, known systematic error can be eliminated; a high standard deviation results in a bad accuracy of the system. The differences of the values for the face size relate to quotients. Therefore the logarithms of the quotiens are calculated. The standard deviation is determined by the average of the following values:

$$\sigma_x : (\frac{||x_{target} - x_{output}||}{w_{target}} - s_x)^2 \qquad \sigma_w : (\log_2 \frac{w_{output}}{w_{target}} - \log_2 s_w)^2$$

$$\sigma_y : (\frac{||y_{target} - y_{output}||}{h_{target}} - y_x)^2 \qquad \sigma_h : (\log_2 \frac{h_{output}}{h_{target}} - \log_2 s_h)^2$$

Assuming a normal distribution of the deviations, the minimum accuracy for a given percentage of time can be determined. In 95.5% of all images a deviation of the position in the area of $[s - 2\sigma, s + 2\sigma]$ can be expected. In 68.3% of all images a deviation will be in the area $[s - \sigma, s + \sigma]$. For the estimation of the size the deviation factor can be expected within $[s2^{-2\sigma}, s2^{2\sigma}]$ for 95.5% and within $[s2^{-\sigma}, s2^{\sigma}]$ for 68.3% of the time.

7.2 Results

The test sequences are evaluated by using several constraints to allow an evaluation of parts of the system. In detail the adjustment of the classificator GFCC to the IFCC, the lowpass filtering of the classificator output, and a comparison of the neural networks with the algorithm *search the largest coherent object* are discussed and evaluated in the following.

7.2.1 Test Sequence 1

The following evaluations have been made on test sequence 1:

1. using the GFCC without adjustment
2. additional lowpass filtering of the output of the GFCC
3. additional adjustment of the GFCC to an IFCC
4. replacement of procedure for object search by neural networks

Table 7.2 shows the results:

mode	correct	s_x	s_y	s_w	s_h	σ_x	σ_y	σ_w	σ_h
1	89.5%	0.11	0.10	1.31	1.23	0.08	0.07	0.13	0.14
2	96.1%	0.09	0.08	1.26	1.27	0.06	0.05	0.13	0.14
3	96.1%	0.07	0.09	1.24	1.29	0.06	0.06	0.13	0.13
4	68.6%	0.16	0.17	1.15	1.18	0.10	0.10	0.38	0.31

Table 7.2: Results for test sequence 1

The lowpass filtering results in a high improvement of the system's reliability. This conclusion can be drawn by the change of the value *correct*. The additional adjustment of the GFCC to the actual face colors doesn't result in a change of the system's performance. The constraint of test sequence 1, not to use skin–colors in the backgrounds, results in the unnecessarity of the classifier adjustment.

Though using all methods in mode 3, the value *correct* is below 100%. Therefore the system loosed track of the face at least once. Considering the motion analysis ot the test sequence, this can't be avoided in this test

sequence. The maximum movements made the face invisible for the system several times.

The average and standard deviations become slightly improved in the first three modes, except for the value s_h. This can be explained by the part of the sequence, in which the indidvidual takes his/her hand to the face. The procedure *search the largest coherent object* is supported by the lowpass filtering to consider arms as part of the face as soon they begin to superimpose, because they have skin–color, too. This results in a higher systematic error for the estimation of the height of the face.

Using neural networks in mode 4 results in different tendencies. The value *correct* seemingly shows a poor reliabilty of the system. One reason for the change for the worse is the fact, that the camera can't be stopped in a recorded sequence after loosing track of the face. Therefore the system must wait until the camera stops accidentally in the sequence. One additional search for a face can therefore be responsible for a significant change of the value *correct*. In a real application the result consequently is better. Another reason is given by the training of the network. The network is trained only with faces, which are fully visible in the retina. Consequently faces which are only partly visible aren't located. The procedure *search the largest coherent object* is able to locate even partly visible faces and achieves better results in these situations.

Table 7.3 shows the maximum deviations, which are to be expected for a given percentage of *time*. The value *time* is determined as the multiplication of *correct* with one of the values 68.3 or 95.5, so that maximum deviations of σ or $2 \cdot \sigma$ are to be expected. For example in 91.7% of the time the x position will be estimated within 19% of the head width from the actual value. It must be considered, that the *actual* value is a manually given value which can vary 10% itself. There deviations of less than 10% or deviation factors between 0.9 and 1.1 are considered as exact estimating of position and size.

To evaluate the influence of the computer performance on the accuracy and reliabilty of the system, the above mode 3 is repeated with different image frequencies (table 7.4).

The accuracy of the system seems to be independent of the computer power, because the average values and standard deviations don't change. The improvement of the value *correct* shows a better reliability of the system by using higher computer power. As long as the face is fully visible within the virtual camera, higher computation power does not result into an improvement. Increasing the computer power therefore results in an improved

mode	time	x	y	w	h
without NN	65.4%	<0.13	<0.15	[1.13,1.36]	[1.18,1.41]
without NN	91.7%	<0.19	<0.21	[1.04,1.48]	[1.08,1.54]
with NN	46.7%	<0.26	<0.27	[0.88,1.50]	[0.95,1.46]
with NN	65.5%	<0.36	<0.37	[0.70,1.89]	[0.77,1.81]

Table 7.3: Accuracy of test sequence 1

frames/sec	correct	s_x	s_y	s_w	s_h	σ_x	σ_y	σ_w	σ_h
2.5	89.5%	0.11	0.09	1.24	1.31	0.07	0.06	0.14	0.14
5	96.1%	0.07	0.09	1.24	1.29	0.06	0.06	0.13	0.13
10	98.8%	0.08	0.1	1.23	1.27	0.05	0.07	0.12	0.14

Table 7.4: Influence of the computer performance

reliability coherent with high speed motions of the tracked indidvidual, but not in a higher accuracy.

7.2.2 Test Sequence 2

Test sequence 2 contains one face in front of a more confusing background but with slower head movements. The sequence has been evaluated with following conditions:

1. using the GFCC without adjusting the classifier
2. using the GFCC with adjusting the classifier to the actual skin–colors of the tracked individual
3. using a classifier, which has been manually adjusted to the actual skin–colors

The third mode uses a classifier, which has been determined with a manual marked area of the face, containing skin–color. The evaluation of this classifier can be considered as a limitation for the automatic adjustment of the GFCC.

Table 7.5 shows the results of the evaluation using the procedure *search the largest coherent object* instead of neural networks. The automatic adjustment of the classifier to the actual skin–colors of the individual results in a significiant improvement of the accuracy and reliability of the system. These results are similar to the results achieved with the manual adjusted classifier, so that the method of automatic adjustment is satisfying.

mode	correct	s_x	s_y	s_w	s_h	σ_x	σ_y	σ_w	σ_h
1	81.7%	0.85	0.38	2.79	2.06	0.67	0.32	0.37	0.4
2	100.0%	0.09	0.1	1.39	2.06	0.05	0.07	0.13	0.14
3	100.0%	0.08	0.12	1.32	2.01	0.04	0.06	0.08	0.14

Table 7.5: Results for test sequence 2 without neural networks

Table 7.6 lists the corresponding results for the system with neural networks. In modus 1 the accuracy and reliability could be improved significantly related to modus of table 7.5 by using neural networks. Especially the average deviation of the estimation error concerning the head size can be lowered. Introducing the manual or automatic adjustment of the classifier, the neural network don't achieve better results than the system without networks. Only the value s_h remains lower than witout using networks. The system is able to handle difficult situations in modus 1 better by using neural networks. This supports the assumption, that the networks are able to perform a form classification rather than finding coherent blobs.

mode	correct	s_x	s_y	s_w	s_h	σ_x	σ_y	σ_w	σ_h
1	94.8%	0.27	0.31	1.43	1.44	0.18	0.24	0.42	0.34
2	100.0%	0.13	0.15	1.21	1.31	0.08	0.07	0.21	0.18
3	100.0%	0.12	0.1	1.13	1.28	0.08	0.07	0.17	0.16

Table 7.6: Results for test sequence 2 with neural networks

Table 7.7 lists the maximum deviations which can be guaranteed for a given percentage of time:

What is the reason for the differnet behavior of s_h? Figure 7.1 shows a situation which is part of the test sequence 2. In this situation the person

mode	time	x	y	w	h
without NN	86.3%	<0.14	<0.17	[1.27,1.52]	[1.87,2.27]
without NN	95.5%	<0.19	<0.24	[1.16,1.66]	[1.70,2.50]
with NN	86.3%	<0.21	<0.22	[1.05,1.40]	[1.16,1.48]
with NN	95.5%	<0.29	<0.29	[0.90,1.62]	[1.02,1.68]

Table 7.7: Accuracy of test sequence 2

takes his hand to his face, so that the procedure *search the largest coherent object* misclassifies the arm as part of the face. Figure 7.3 shows the extracted faces. Because of the arm the face is frequently estimated to high, resulting in a high value s_h.

Figure 7.2 shows the same situation with using neural networks. The extracted faces are given in figure 7.4. The networks are capable of recognizing the form of the arm as not face–like. Therefore the system can perform correct classifications.

The procedure *search the largest coherent object* achieves better reliability and accuracy in situations not requiring a form detection, especially in connection with fast motion. Form detection is not necessary, if the background contains no face–like colors and the face is seperated from other objects with face–like colors. If form detection is required, using neural networks improves the accuracy of the system.

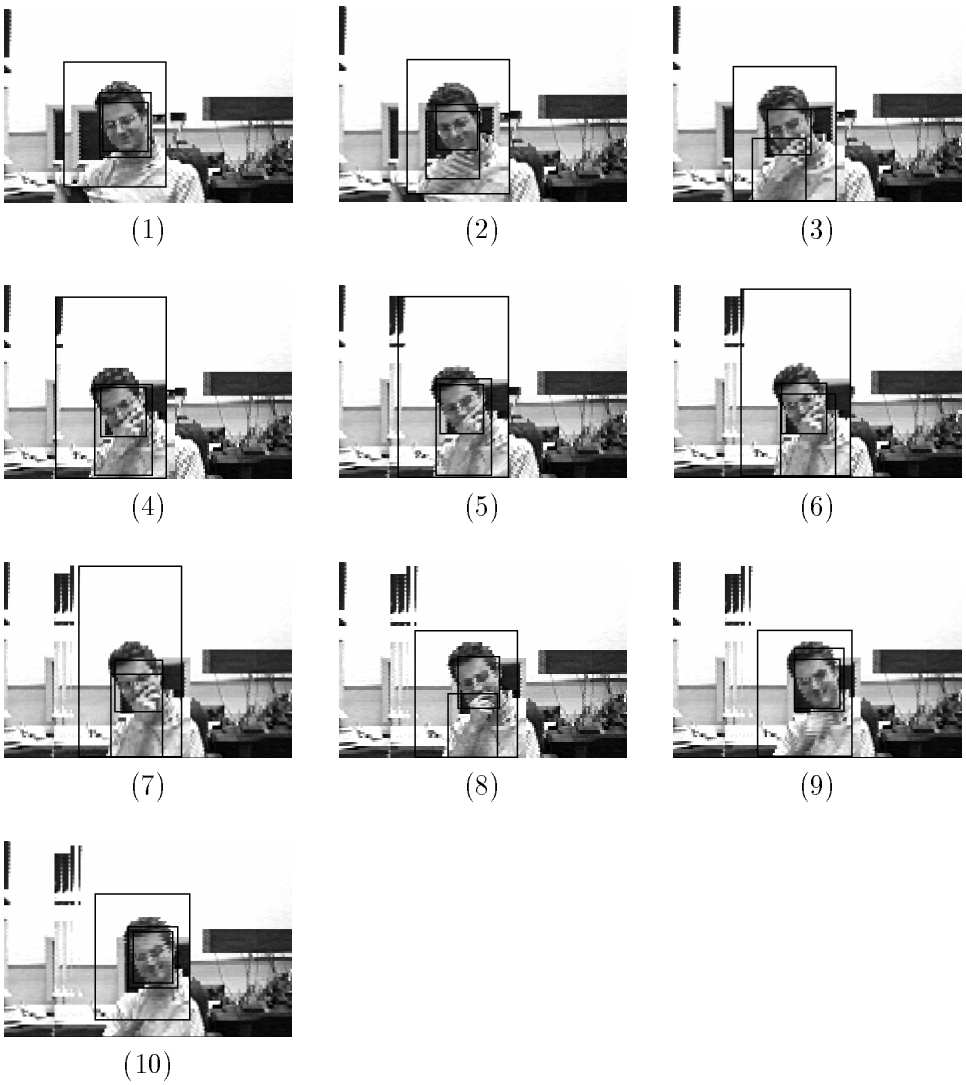


Figure 7.1: Part of test sequence 2 - without using neural networks

Figure 7.2: Part of testsequence 2 - using neural networks

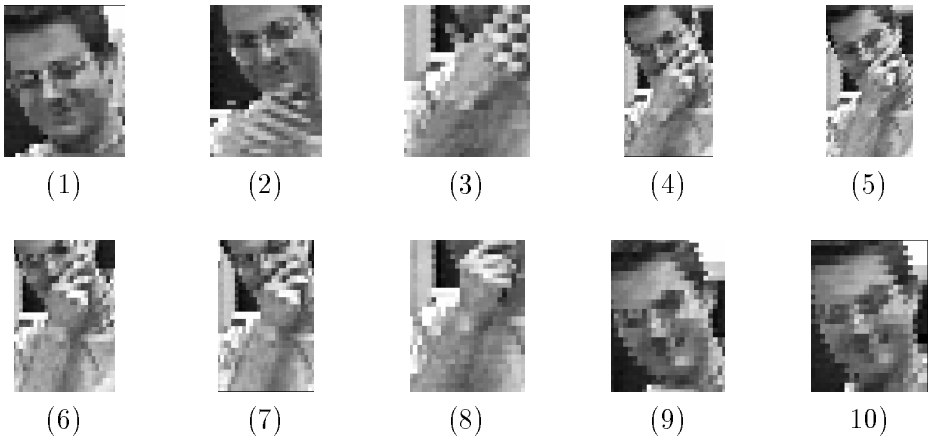


Figure 7.3: Extracted faces without using neural networks

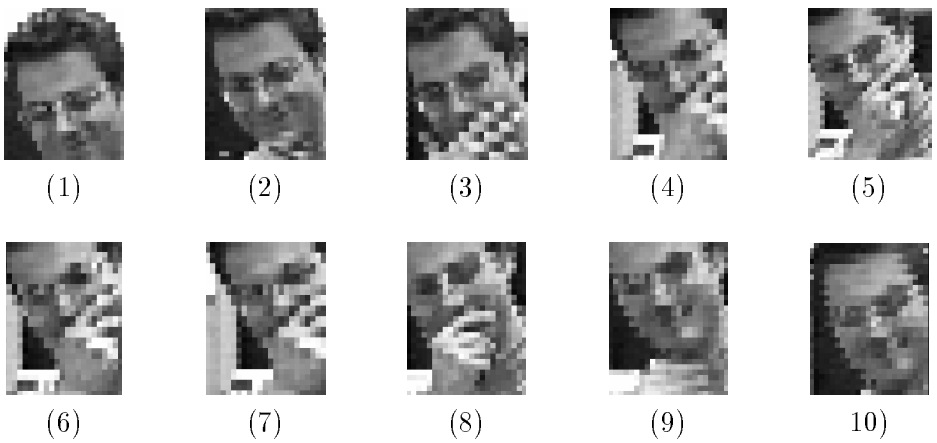


Figure 7.4: Extracted faces with using neural networks

Chapter 8

Conclusion

This technical report proves the possibility of face tracking in real time on workstations. The developed system meets all requirements enumerated in the introduction. The reliability and accuracy of locating faces could be achieved by searching for coherent moving objects with skin–color.

Tracking of faces in realtime requires at least 5 positions per second. To achieve a sufficient image frequency the image size must be limited. It is shown that locating of faces in low resolution images can be highly improved by using color as an additional feature. The invention of a face–color classifier allows the concentration of all color dependencies at a central part of the system, so that all other system parts can be developed without any dependency on lighting situations, skin–colors, hardware, etc. Changing of one of these values influences only the color classifier, not one of the other system parts. Therefore it is not necessary to retrain the neural networks even after exchanging camera, framegrabber and lighting situations, though the networks use color as a feature to locate faces. The presented methods allow an easy automatic adjustment of the color classifier to the environment during the tracking.

With the used methods it's possible to locate arbitrary faces independent of haircut, skin–color or other individual features. Misclassifications may occur only under the following circumstances:

- Several faces with similar skin–color superimpose each other on the image. Using the system without neural networks results in the misclassification of both faces as one large face. The neural networks instead decide for one of the faces. The same problem occurs with bringing up hands or arms in front of the face.

- Misclassifications may occur, if the background contains many colors, which are classified as face–like colors. The neural networks can handle these situations better, because they consider form in addition to color and motion.

- If the motion of the face exceeds a maximum value, the system looses track of the face.

The invention of a virtual camera allows a stable image of the speaker's face without any delay. The inertia of the camera and zoom lens limit the system only in long range motions. In a realistic communication between two individuals sitting on chairs, only small motions occur by moving a chair, leaning back or turning the chair or head. The system performs good results in these situations and seldomly looses track of a face.

Chapter 9

Future Directions

Several possibilities exist to extend or apply the face tracking system developed in this technical report. Most challenging are combinations with systems using visual aspects of communication or the speaker's positions.

9.1 Extensions of the System

9.1.1 Stereoscopic Images

Measuring the distance of a face would be possible by using a second camera. The size of a face is dependent on the distance to the camera and the position of the zoom lens. With the knowledge of these facts an additional feature for a validation of the face size comes into existence. In addition a stereoscopic image improves the distinction between face and background.

9.1.2 Locating Eyes and Lips

The method of color classification can be used for locating eyes and lips. The red of the lips and the white in the eyes represent colors which occur more seldom in faces than skin–colors. The example images in the chapter about the FCC's show the extraction of the named areas.

9.1.3 Locating Several Faces

By using several virtual cameras it is possible to keep track of several faces independently. Only one physical camera is necessary, but higher computer performance will be required for this task.

9.2 Applications of the System

9.2.1 Videophone

The transmission of stable images of the communication partners require the speaker's face at given positions using common videophone systems. The face tracking system can remove this limitation, so that the communication partners can move freely.

9.2.2 Identifying Individuals

A combination of the face tracking system with already existing systems for face identification would highly improve these systems. Individuals are no longer required to get into a given position to enable identification.

9.2.3 Using the Speaker's Position

The face tracking system keeps track of the speaker's position in a room. A microphone array can use these positions to select audio signal from this position by beam–forming. This method can highly improve speech recognition in situations with background noise. In video conferences the camera can automatically move to the direction of the speaker's face, so that always the speaker's face

Bibliography

[1] S. Baluja and D. Pomerleau. Non–Intrusive Gaze Tracking Using Artificial Neural Networks. In *Advances in Neural Information Processing Systems*, volume 6. Morgan Kaufmann, 1993.

[2] C. Bregler, H. Hild, S. Manke, and A. Waibel. Improving Connected Letter Recognition by Lipreading. In *International Conference on Acoustics, Speech and Signal Processing*, 1993.

[3] G. W. Cottrell. Extracting Features from Faces Using Compression Networks: Face, Identity, Emotion, and Gender Recognition Using Holons. In *Connectionist Models: Proceedings of the 1990 Summer School*, pages 328–337, San Mateo, 1990. Morgan Kaufmann.

[4] A. Despopoulos. *Color Atlas of Physiology*. New York, 1991.

[5] P. Duchnowski, U. Meier, and A. Waibel. See Me, Hear Me: Integrating Automatic Speech Recognition And Lip–Reading. In *ICSLP*, 1994.

[6] J. L. Flanagan, J. D. Johnston, R. Zahn, and G. W. Elko. Computer Steered Microphone Arrays for Sound Transduction in Large Rooms. *Journal of the Acoustical Society of America*, 78:236–255, November 1985.

[7] M. K. Fleming and G. W. Cottrell. Categorization of Faces Using Unsupervised Feature Extraction. In *International Joint Conference on Neural Networks*, volume 2, pages 62–70, San Diego, CA., 1990.

[8] B. A. Golomb, D. T. Lawrence, and T. J. Sejnowski. Sexnet: A Neural Network Identifies Sex From Human Faces. In David Touretzky, editor, *Neural Information Processing Systems*, volume 3, pages 572–577, San Mateo, CA., 1991. Morgan Kaufmann.

[9] V. Govindaraju, D. B. Sher, and S. N. Srihari. Locating human faces in newspaper photographs. In *Proc. of IEEE–CS Conf. Computer Vision and Pattern Recognition*, pages 278–285, San Diego, CA, 1989.

[10] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the Theory of Neural Computation*. Addison-Wesley Publishing Company, New York, 1991.

[11] R. A. Hutchinson and W. J. Welch. Comparison of Neural Networks and Conventional Techniques for Feature Location in Facial Images. In *First IEE Int. Conf. on Artificial Neural Networks*, pages 201–205, London, U.K., 1989.

[12] T. Kohonen, P. Lehtio, E. Oja, A. Kortekangas, and K. Makisara. Demonstration of Patter Processing Properties of the Optimal Associative Mappings. In *Proceedings International Confernece on Cybernetics and Society*, Washington, D.C., 1977.

[13] D. Pomerleau. *Neural Network Perception for Mobile Robot Guidance*. PhD thesis, School of Computer Sience, CMU, February 1992.

[14] W. K. Pratt. *Digital Image Processing*. Wiley–Interscience, New York, 1991.

[15] P. W. Rander. Real–Time Image–Based Face Tracking. Master's thesis, Carnegie Mellon University, Pittsburgh, U.S.A., 1993.

[16] T. Sakai, M. Nagao, and T. Kanade. Computer Analysis and Classification of Photographs of Human Faces. In *First USA–Japan computer conference*, Yokyo, Japan, 1972.

[17] M. Stone. Cross–validation: A review. *Math. Operationsforsch. Statist., Ser. Statistics*, 9(1), 1978.

[18] M. A. Turk and A. P. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, March 1991.

[19] S. Uras, F. Girosi, A. Verri, and V. Torre. A Computational Approach to Motion Perception. *Biological Cybernetics*, 60(2):79–87, 1988.

[20] A. Waibel, M. T. Voe, P. Duchnowski, and S. Manke. Multimodal Interfaces. *Artificial Intelligence Review Journal*, 1994.

[21] G. Wyszecki and W. S. Stiles. *Color Science*. John Wiley & Sons, Inc., New York, 1967.