Department of Computing

Model Based Methods for Locating, Enhancing and Recognising Low Resolution Objects in Video

Annika Kramer

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University of Technology

December 2009

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university. To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgement has been made.

Annika Kramer

Date

Model Based Methods for Locating, Enhancing and Recognising Low Resolution Objects in Video

by
Annika Kramer

Submitted to the Department of Computing
in December, 2009 in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Visual perception is our most important sense which enables us to detect and recognise objects even in low detail video scenes. While humans are able to perform such object detection and recognition tasks reliably, most computer vision algorithms struggle with wide angle surveillance videos that make automatic processing difficult due to low resolution and poor detail objects. Additional problems arise from varying pose and lighting conditions as well as non-cooperative subjects. All these constraints pose problems for automatic scene interpretation of surveillance video, including object detection, tracking and object recognition.

Therefore, the aim of this thesis is to detect, enhance and recognise objects by incorporating a priori information and by using model based approaches. Motivated by the increasing demand for automatic methods for object detection, enhancement and recognition in video surveillance, different aspects of the video processing task are investigated with a focus on human faces. In particular, the challenge of fully automatic face pose and shape estimation by fitting a deformable 3D generic face model under varying pose and lighting conditions is tackled. Principal Component Analysis (PCA) is utilised to build an appearance model that is then used within a particle filter based approach to fit the 3D face mask to the image. This recovers face pose and person-specific shape information simultaneously. Experiments demonstrate the use in different resolution and under varying pose and lighting conditions. Following that, a combined tracking and super resolution approach enhances the quality of poor detail video objects. A 3D object mask is subdivided such that every mask triangle is smaller than a pixel when projected into the image and then used for model based tracking. The mask subdivision then allows for super resolution of the object by combining several video frames. This approach achieves better results than traditional super resolution methods without the use of interpolation or deblurring.

Lastly, object recognition is performed in two different ways. The first recognition method is applied to characters and used for license plate recognition. A novel character model is proposed to create different appearances which are then matched with the image of unknown characters for recognition. This allows for simultaneous character segmentation and recognition and high recognition rates are achieved for low resolution characters down to only five pixels in size. While this approach is only feasible for objects with a limited number of different appearances, like characters, the second recognition method is applicable to any object, including human faces. Therefore, a generic 3D face model is automatically fitted to an image of a human face and recognition is performed on a mask level rather than image level. This approach does not require an initial pose estimation nor the selection of feature points, the face alignment is provided implicitly by the mask fitting process.

To Ian

For your unconditional love and support throughout the last three and a half years

To my parents

For your encouragement and being my foundation from the other side of the world

Acknowledgements

This dissertation would not have been possible without the following people:

My supervisors, Professor Svetha Venkatesh and Associate Professor Tele Tan for their support and expert guidance throughout the course of this project and for giving me the opportunity to study.

All postgraduate students of the Department of Computing for their camaraderie and their invaluable discussions and especially Patrick for his endless seminars.

All the people who made their software available online and thus contributing to this thesis. In particular Jörgen Ahlberg at Linköping University in Sweden for providing the parametrised 3D face model CANDIDE-3, Ghassan Hamarneh at Simon Fraser University in Canada for his Matlab-based multi-resolution Active Shape Models, Deng Cai at Zhejiang University in China for providing Matlab codes of subspace learning algorithms and Anke for providing the German number plate font.

Last but not least a big thanks to everyone who had to proof read my thesis :-)

Copyright

I would like to thank the following authors and publishers for permission of copyright:

Figure 2.1(a) Figure 2.1(b)	Reprinted with kind permission of Tim Cootes.
Figure 2.1(c)	ACM, SIGGRAPH 1999 CONFERENCE PROCEEDINGS: COMPUTER GRAPHICA ANNUAL CONFERENCE SERIES, "A Morphable Model for the Synthesis of 3D faces" pp. 187-194, 1999. Reprinted by permission of Pearson Education, Inc.
Table 2.1(b)	With kind permission from Springer Science+Business Media: International Journal of Computer Vision, Robust and Rapid Generation of Animated Faces from Video Images: A Model-Based Modeling Approach, 58, Jul 1, 2004, 93-119, Z. Zhang, Z. Liu, D. Adler, M.F. Cohen, E. Hanson and Y. Shan, Figure 1
Table 2.1(c)	With kind permission of IEEE: Conference on Computer Vision and Pattern Recognition, 1999, Explanation-based facial motion tracking using a piecewise bezier volume deformation model, H. Tao and T. Huang, Figure 4.
Table 2.2(a)	With kind permission of Springer Science+Business Media: Computer Analysis of Images and Patterns, volume 3691, 2005, pages 854-861, A hierarchical face behavior model for a 3D face tracking without markers, R. Roussel and A. Gagalowic, Figure 8
Table 2.2(b)	With kind permission of IEEE: Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004, Image-based photorealistic 3-d face modeling, I.K. Park, H. Zhang, V. Vezhnevets, H.-K. Choh, Figure 2.
Table 2.2(c)	With kind permission of IEEE: Brazilian Symposium on Computer Graphics and Image Processing, 2004, Adaptive deformable models, S. Goldenstein, C. Vogler and L. Velho, Figure 1
Figure 3.2	With kind permission of IEEE: Conference on Pattern Recognition, 2008, Model-based combined tracking and resolution enhancement, A. Kuhl, T. Tan and S. Venkatesh, Figure 1
Figure 3.4 Figure 3.7 Figure 3.8 Figure 3.13 Figure 3.15 Figure 3.16	With kind permission of Springer Science+Business Media: MIRAGE Computer Vision / Computer Graphics Collaboration Techniques and Applications, 2009, Automatic fitting of a deformable face mask using a single image, A. Kuhl, T. Tan and S. Venkatesh, Figure 2 to 7.

Published Work

The original work presented in Chapters 3, 4 and 5 of this thesis is based upon works that have been previously published in refereed conference papers.

The model based method for fitting a deformable 3D face mask to a single image, proposed in Chapter 3, is based on the following paper:

> Kuhl, A., Tan, T., and Venkatesh, S. (2009a). Automatic Fitting of a Deformable Face Mask Using a Single Image. In *MIRAGE Computer Vision / Computer Graphics Collaboration Techniques and Applications*. Springer.

Chapter 4 introduces a combined tracking and super resolution approach for enhancing low resolution video objects and this method has previously been published in:

> Kuhl, A., Tan, T., and Venkatesh, S. (2008b). Model-based combined tracking and resolution enhancement. In *19th International Conference on Pattern Recognition ICPR*. IEEE Computer Society.

> Kuhl, A., Tan, T., and Venkatesh, S. (2009b). *A Model-based Approach for Combined Tracking and Resolution Enhancement of Faces in Low Resolution Video*, chapter 9, pages 173–194. IN-TECH.

The model based character recognition approach for objects in low resolution, presented in Chapter 5, is based on the following conference paper:

> Kuhl, A., Tan, T., and Venkatesh, S. (2008a). Model-based character recognition in low resolution. In *15th IEEE International Conference on Image Processing ICIP*, pages 1001–1004. IEEE Computer Society.

I married on the 23rd of May 2009 and changed my surname from Kuhl to Kramer.

Notations

A	degradation matrix
C_c^r	character template r for character c
D_m	3D coordinate of the centre of the mth mask triangle
E	mean error
E^f_{colour}	mean colour error for frame f
F	number of frame
H	height of an image
I	2D image
I^{high}	high resolution image
I^{low}	low resolution image
$\mathbf{J}$	vector of concatenated colour values
$\widehat{\mathbf{J}}$	reconstructed image
$\mathbf{J}_{SR}$	super resolved vector of concatenated colour values
L	number of mask vertices
M	number of mask triangles
$\mathcal{N}$	Normal distribution
O	column matrix of warping templates
$\mathcal{P}$	projection from 3D coordinates to 2D image coordinates
P_l	3D coordinate of the lth mask vertex
$\mathcal{Q}$	creates vector of concatenated colour values
R	number of templates
$\mathbf{S}$	each column is a displacement vector that controls the shape
$\mathcal{T}$	template creation function
T_0	initial camera parameters
T^{app}	camera parameters for appearance based tracking
T^{geo}	camera parameters for geometric based tracking
T^{int}	intrinsic camera parameters
T^{ext}	extrinsic camera parameters
V_h	function that returns the hth harmonic image
W	width of an image
X	principal components
Z_c^r	best 2D image position for template r of character c

b	harmonic image
c	character
d	distance in feature space
d^o	transformation parameter displacement
d_x	displacement along x-axis
d_y	displacement along y-axis
e	reconstruction error in feature space
$\mathbf{g}$	deformed face mask
$\overline{\mathbf{g}}$	neutral face mask
h	error term
n_i	transformation parameter displacement
l	mask vertex point number
o	warping template
p	mask vertex point in 2D
q	vector of coefficients
$\overline{x}$	average face
$x_t^{(i)}$	Monte Carlo sample i at time t
$\widetilde{w}_t^{(i)}$	weight of particle i at time t
Σ	diagonal covariance matrix
α	linear coefficient
β	linear coefficient
γ_k	parameter that controls the kth mask deformation
ϵ	threshold
η	surface normal
κ	threshold for adding characters
λ	annealing factor
ξ	pixel size
ρ	albedo
υ	annealing factor
ϕ_x	rotation around x-axis
ϕ_y	rotation around y-axis
ϕ_z	rotation around z-axis

Contents

Abstract ii

Acknowledgements iv

Published Work vi

Notations viii

1 Introduction 1
- 1.1 Aims and Approach . . . 2
- 1.2 Contributions and Significance . . . 3
  - 1.2.1 Model Based Object Super Resolution During Tracking . . . 4
  - 1.2.2 Low Resolution Character Recognition Using Parametrised Model . 4
  - 1.2.3 Automatic Face Recognition Using a Single Training Image . . . 5
- 1.3 Structure of the Thesis . . . 6

2 Background 8
- 2.1 Model-Based Fitting . . . 9
  - 2.1.1 Principal Component Analysis . . . 9
    - 2.1.1.1 Active Shape Models . . . 10
    - 2.1.1.2 Active Appearance Models . . . 12
    - 2.1.1.3 3D Morphable Models . . . 13
  - 2.1.2 Deformable 3D Face Mesh Models . . . 14
    - 2.1.2.1 Predefined Metrics and MPEG-4 . . . 14
    - 2.1.2.2 Complex Deformable Models . . . 17
- 2.2 Model Based Tracking . . . 20
  - 2.2.1 Feature Based Tracking . . . 21
  - 2.2.2 Appearance Tracking . . . 21
  - 2.2.3 Particle Filters . . . 23
- 2.3 Super Resolution . . . 24
  - 2.3.1 The Inverse Problem . . . 24
  - 2.3.2 Image and Video Super Resolution . . . 25
  - 2.3.3 Super Resolution Optical Flow . . . 27
  - 2.3.4 Super Resolution via a 3D Model . . . 28
  - 2.3.5 Hallucinating - Face Super Resolution . . . 29

2.3.6 Limits on Super Resolution 29
2.4 Character Recognition 30
2.4.1 Text Document Recognition 30
2.4.1.1 Low Resolution Text Recognition 32
2.4.2 Number Plate Recognition 33
2.4.2.1 Low Resolution Number Plate Detection and Recognition . 34
2.5 Face Recognition 35
2.5.1 2D Face Detection and Recognition 35
2.5.2 3D Face Recognition 36
2.5.3 Multi-modal 2D + 3D Face Recognition 37
2.5.4 Face Recognition From A Single Image 38
2.5.5 Modelling Illumination for Face Recognition 38
2.6 Conclusion 41

3 Automatic Fitting of a Deformable Face Mask 44

3.1 Automatic 3D Face Mask Fitting 45
3.1.1 3D Appearance Model Generation 46
3.1.1.1 3D Face Mask 46
3.1.1.2 The 3D Appearance Model 49
3.1.2 Light-Invariance through Harmonic Images 51
3.1.3 Particle Filter Refinement 52
3.1.4 Automatic Fitting Algorithm 54
3.2 Experiments 55
3.2.1 IMM Face Database 56
3.2.2 Yale Face Database 57
3.2.3 The 3D Appearance Model 57
3.2.4 Fitting Performance 58
3.2.4.1 Accuracy versus Speed 59
3.2.4.2 Low Resolution Fitting 60
3.2.4.3 Comparison with Active Shape Models 63
3.2.4.4 Different Pose and Lighting Conditions 68
3.2.5 Experiments Summary 70
3.3 Conclusion 72

4 Combined Tracking and Super Resolution 73

4.1 The Image Formation Process 74
4.2 Method Overview 76
4.3 3D Object Tracking 78
4.3.1 Appearance Based Tracking 78

4.3.2 Geometric Based Approach 79
4.3.3 Combined Appearance and Geometric Tracking 80
4.4 3D Model Based super resolution 81
4.5 Extension to Non-planar and Non-rigid Objects 81
4.6 Experiments 83
4.6.1 Tracking Accuracy 83
4.6.1.1 Combined Appearance and Geometric Tracking 84
4.6.1.2 Expression Tracking 86
4.6.1.3 Mask Size vs. Object Size 87
4.6.2 3D Model Based super resolution 89
4.6.2.1 Planar Object - A Cube 89
4.6.2.2 Non-planar Objects - Faces 92
4.6.3 Comparison with super resolution Optical Flow 98
4.6.3.1 Super resolution of Faces in Surveillance Footage 101
4.7 Conclusion 104

5 Model Based Low Resolution Character Recognition 106
5.1 Model Based Character Recognition 107
5.1.1 Template Generation 108
5.1.2 Low-Resolution Word Recognition 111
5.2 Experiments 114
5.2.1 Suitability and Interchangeability of Different Fonts 115
5.2.1.1 Medium Resolution - 1m Distance 116
5.2.1.2 Low Resolution - 2m Distance 118
5.2.2 Number Plate Recognition 119
5.2.2.1 Indoor Number Plate Recognition 120
5.2.2.2 Outdoor Number Plate Recognition 122
5.2.3 Text Recognition 126
5.3 Conclusion 129

6 Face Recognition 131
6.1 Multi-Modal Face Recognition from a Single Image 132
6.1.1 Creating the Face Database 132
6.1.2 Pose and Lighting Invariant Recognition 134
6.2 Experiments 135
6.2.1 Recognition under Different Resolutions 136
6.2.2 Recognition Under Different Pose and Lighting Conditions 141
6.2.3 Automatic Face Recognition 143
6.2.4 Comparison with Tensor based method 144

6.2.5 Comparison of Recognition Results 146
6.2.6 Experiments Summary 149
6.3 Conclusion 149

7 Conclusion 151
7.1 Future Work 152

A Face Mask Subdivision 165

List of Figures

2.1 Different PCA Based Face Models 11
2.2 Model based Tracking 20
2.3 Basic steps of optical character recognition methods 31
2.4 Basic steps of number plate recognition algorithms 34
2.5 Different face recognition approaches 35
2.6 Face under different illumination 39

3.1 Method Overview 45
3.2 CANDIDE-3 face mask 47
3.3 Face Mask Shape Deformations 48
3.4 Different mean faces 50
3.5 Harmonic Images 52
3.6 Estimating the lighting with harmonic images. 53
3.7 Different steps of the proposed fitting algorithm 54
3.8 Point correspondences face mask and landmarked images 56
3.9 Subsets of the Yale Face Database 57
3.10 Different Poses of the Yale Face Database 58
3.11 First seven Eigenfaces 59
3.12 Accuracy versus speed 60
3.13 Result for different image resolutions and mask sizes. 61
3.14 Different size masks fitted to an image 62
3.15 Results for testing on training 63
3.16 Results for light variation 63
3.17 ASM fitting results for different resolutions 65
3.18 ASM fitting results for different image resolutions 66
3.19 Fitting results on the IMM Face Database - Leave one out test 67
3.20 ASM fitting results for different lighting conditions 68
3.21 Fitting results for different pose and lighting conditions 71

4.1 The Image Formation Process 75
4.2 Different low resolution appearances of a high resolution cube. 75
4.3 Method Overview 77
4.4 CANDIDE-3 face mask 82
4.5 Cropped faces for each image resolution used 84
4.6 Tracking results for different size faces 84

4.7 Sample frame shots for both tracking approaches used. 85
4.8 Results of the expression tracking. 87
4.9 Mean tracking RMSE across 100 frames for a planar objects. 88
4.10 Example frames and the bilinear interpolated frames. 89
4.11 Super resolution results of a cube of resolution of 24×24 pixels. 90
4.12 Super resolution results of a cube of resolution 12×12 pixels. 91
4.13 Example images of all six persons at 160×120 pixels. 92
4.14 Example images of all six persons at 80×60 pixels. 93
4.15 Quality of the super-resolved 3D face mask using different resolutions. . . . 94
4.16 Results of the combined tracking and super resolution approach for Person 1. 95
4.17 Results of the combined tracking and super resolution approach for Person 2. 95
4.18 Results of the combined tracking and super resolution approach for Person 3. 96
4.19 Results of the combined tracking and super resolution approach for Person 4. 96
4.20 Results of the combined tracking and super resolution approach for Person 5. 97
4.21 Results of the combined tracking and super resolution approach for Person 6. 97
4.22 Result of the proposed super resolution approach. 98
4.23 Result of the super resolution optical flow algorithm. 99
4.24 Final comparison with optical flow. 100
4.25 super resolution of Person 3 in surveillance footage 102
4.26 super resolution of Person 7 in surveillance footage 102
4.27 super resolution of Person 8 in surveillance footage 102
4.28 super resolution of Person 5 in surveillance footage 103
4.29 super resolution of Person 9 in surveillance footage 103
4.30 super resolution of Person 10 in surveillance footage 103

5.1 Model based character recognition - overview 108
5.2 Template generation process 109
5.3 The Effect of Image Enhancement 110
5.4 Low-Resolution Word Recognition Algorithm 112
5.5 Low-Resolution Word Recognition Algorithm continued 113
5.6 Experimental setup 115
5.7 Example images of different fonts at 1m distance 116
5.8 Example images for character 'A' at distance 1m 116
5.9 Example images of different fonts at 2m distance. 118
5.10 Example images for character 'A' at distance 2m. 118
5.11 Number plate recognition in an indoor environment. 120
5.12 Cropped number plates for each distance. 120
5.13 Example images of Western Australian number plates. 122
5.14 Cropped example Western Australian number plates. 123

5.15 Cropped example number plates used for recognition. 123
5.16 Example images of a German number plate. 125
5.17 Cropped example German number plates at 2m distance. 125
5.18 Cropped example German number plates at 3m distance. 126
5.19 Cropped document images used for recognition 127
5.20 Text recognition results at 20cm and 25cm. 127
5.21 Text recognition results at 30cm. 128

6.1 Face recognition system overview 133
6.2 Face Registration for Training 134
6.3 Lighting Correction Example 135
6.4 Subsets of the Yale Face Database 136
6.5 Different image resolutions and mask sizes used for recognition 137
6.6 Recognition results for different image resolutions and mask sizes during testing 138
6.7 Recognition results for different image resolutions and mask sizes during training and testing 139
6.8 Detailed recognition rates for all ten individuals 140
6.9 Persons of the Yale Face Database B 141
6.10 Recognition rates for the Yale Face Database B for different pose and lighting conditions 141
6.11 Different Poses of the Yale Face Database 142
6.12 Automatic Face Recognition Results 143

A.1 Different Subdivision Schemes Applied to CANDIDE-3 166
A.2 The combined subdivision schemes applied to CANDIDE-3 167

List of Tables

2.1 Comparison of different 3D face mesh models using predefined metrics . . . 15
2.2 Comparison of different 3D face mesh models 18

3.1 Shape parameters of the CANDIDE-3 face mask 47
3.2 Results comparison with ASM 65
3.3 Fitting results for different pose and lighting conditions 70

4.1 Results of the expression tracking. 86
4.2 Mean RMSE across 100 frames 88

5.1 Font recognition results at distance 1m 117
5.2 Font recognition results at distance 2m. 119
5.3 Summary of the number plate recognition result in an indoor environment. 121
5.4 Number plate recognition results 126

6.1 Tensor based face recognition results. 144
6.2 Face recognition results for a single training image. 145
6.3 Face recognition results for different methods proposed in literature. 146
6.4 Comparison of different face recognition methods 148

Chapter 1

Introduction

The increasing interest in public and private wide area surveillance and the continuously growing number of CCTV cameras has been followed by an increasing demand for automatic image and video analytic methods. Most surveillance cameras are used only for recording and storing surveillance video whilst the actual video processing is done manually. However, watching surveillance footage to recognise suspects and to identify events is laborious and time consuming. Thus, there is a high demand for automatic video processing methods for scene interpretation and analysis.

The usability of most existing video processing approaches is curtailed by the poor quality of surveillance videos. Wide area surveillance situations usually require a large number of sensors, thus making the use of high resolution cameras prohibitive because of high cost and exponential growth in storage capacity. However, small and low price CCTV cameras usually use CMOS technology, which produces poorer quality video compared to the more expensive CCD cameras. However, CCD cameras in wide area surveillance can still yield low resolution images of the object of interest, due to large distances from the camera. Thus, low resolution is the main problem faced by video processing methods because of poor detail in scenes that do not provide reliable features for subsequent processing.

A large number of approaches have been proposed in the literature to address the problem of automatic video processing but most methods rely on accurate low level feature extraction to perform object detection and recognition tasks. However, such bottom-up approaches are difficult in surveillance scenes that only provide poor detail in objects due to low resolution. Model based methods approach the problem in a top down way by assuming a priori knowledge of the scene or the object. A generic object model provides ex ante information that can be used to detect, enhance and recognise the appearance of an object as a whole without trying to find low level object features first.

All methods developed in this thesis use a model based approach for automatic video processing in low resolution and thus avoid the use of unreliable detection of low level image features. The 3D face pose and shape is estimated by automatically fitting a 3D generic deformable face model to an image under different pose and lighting conditions. Furthermore, this thesis explores the use of a 3D face mask for combined tracking and super resolution. The face mask is subdivided into a fine mesh and several video frames are combined to increase the resolution of the mask texture without the need for interpolation. Following that, the appearance of planar objects is modelled and a large number of different appearances, including different resolutions, is created and then used for recognition. Finally, a multi-model face recognition method is proposed that requires a single training image per subject only. A 3D deformable face mask is fitted to an image and recognition is performed directly on the mask texture.

1.1 Aims and Approach

The focus of this thesis is the development of model based methods that work automatically with low resolution images as well as under varying pose and lighting conditions. Therefore, the four objectives of this thesis are:

1. The development of a model based method for accurate pose and shape estimation in low resolution for different pose and illumination using only a single image,
2. The implementation of a super resolution approach that uses a 3D object model and avoids the use of interpolation to combine several video frames to create a single high resolution image,
3. The development of a recognition method for planar objects that uses model information to create different object appearances, and
4. The development of a recognition method for non-planar objects under different pose and lighting conditions, using a single training image.

Each of these aims is addressed in a separate chapter of this thesis. The first specific aim is achieved by automatically fitting a generic deformable 3D face mask to a previously unseen image of an object. The pose and person-specific shape parameters are recovered during the fitting process. The 3D face mask is used to model different illumination conditions and thus allows for automatic fitting under varying illumination conditions.

The same face mask is furthermore used for object tracking and super resolution. Resolution enhancement is achieved by subdividing the object mask into a fine mesh and projecting it into each video frame using the tracking results. A novel texture mapping approach assigns a single colour value to each mask triangle and by combining several frames, a super resolved texture is created without image interpolation. The key is the use of a fine mask mesh where each triangle is smaller than a pixel when projected into the image, making super resolution possible.

The third aim is achieved by creating a large number of different object appearances to perform template matching based recognition of planar objects. The image formation process of the camera is parameterised and different templates representing different object appearances in different resolution are created. Planar objects are recognised directly in very low resolution images without the use of image segmentation, enhancement or super resolution techniques.

To tackle the problem of non-planar object recognition a template based approach is not feasible. Therefore, the fourth aim is achieved by deploying a generic 3D face mask to recover pose and person-specific shape parameters first. Instead of cropped 2D images, the texture of the object mask is directly used for recognition. By using the mask texture no image alignment is necessary and different lighting conditions are accounted for by the 3D object model.

1.2 Contributions and Significance

This thesis makes three main contributions in the area of computer vision:

1. The use of subdivided object models together with a novel texture mapping technique for combined tracking and super resolution,
2. The development of a parameterised character model for recognition in low resolution images, and
3. The deployment of a generic 3D deformable face model for automatic face detection and recognition

The contributions and their significance are detailed in the following sections.

1.2.1 Model Based Object Super Resolution During Tracking

The first major contribution of this thesis is the development of a combined tracking and super resolution method using a 3D object model. A model based tracking approach is deployed to estimate the pose parameters of the 3D object model in each frame of a video sequence. A novel texture mapping approach then assigns a single colour value to each vertex of the 3D object model mask. This is contrary to existing texture mapping techniques that rotate and scale an image area to texture each mask vertex. To make super resolution possible all mask vertices need to be smaller than a pixel when projected into the image, therefore subdivision schemes commonly used in computer graphics are applied to subdivide the 3D object mask into a fine mesh. The super resolved object texture is then computed as the mean colour value of each vertex across several frames. This super resolution technique has several advantages over traditional video super resolution:

- The resolution is increased at mask level and only for the selected object instead of enhancing the entire scene of the image. This reduces computation time and makes the approach applicable to any non-planar and non-rigid object that can be tracked using a deformable 3D object model.
- No interpolation is needed to increase the resolution, instead the resolution increase is determined by the number of subdivisions of the object mask. The finer the mask mesh the higher the possible increase in resolution. The achieved resolution is equal or higher compared to traditional methods without the need for deblurring.
- The resolution increases simultaneously during the tracking process and improves with every frame. A threshold guarantees that only frames with small tracking errors are used for super resolution to ensure best results.
- The subdivided mask is necessary for super resolution but also improves tracking. Experiments show that a fine mask mesh improves the tracking accuracy compared to a coarser mesh.

1.2.2 Low Resolution Character Recognition Using Parametrised Model

The second main contribution of this thesis is the development of a parameterised camera model to create character templates for low resolution character recognition. The image formation process of the camera is modelled and parameterised, and then applied to each character of the alphabet. By varying the parameters, different character templates showing different appearances and resolutions are created. The proposed character recognition method applies template matching and clustering techniques to recognise single characters. This approach has several advantages:

- The simple camera model is parameterised to allow for the efficient generation of character templates by varying three parameters. The resulting templates precisely match character images taken by a camera and achieve high correlation.
- The template matching based character recognition approach allows for simultaneous character segmentation and recognition. Single characters are separated and recognised despite merged character edges that hinder character separation in low resolution. Thus, the recognition of low resolution characters of down to five pixels in height is possible.
- No image enhancement or super resolution is required to increase the resolution or enhance the image quality. Each character is recognised by its low resolution grey scale appearance only.
- Recognition is performed on grey scale images instead of binarising each character image. The additional information stored in the aliased grey scale character edges is utilised by the proposed approach for improved recognition in low resolution.
- Due to the small size of each character template, recognition based on a template matching approach is performed efficiently.

1.2.3 Automatic Face Recognition Using a Single Training Image

The third main contribution of this thesis is the deployment of a generic 3D deformable face model for automatic face pose and shape detection and recognition using only a single training image. A model based mask fitting method is developed to automatically fit a deformable 3D face mask to an image of a face. The fitting recovers the 3D pose and person-specific shape parameters while the 3D shape of the mask is used to account for different lighting conditions. Recognition is then performed directly on the extracted face mask texture instead of the cropped image area and only a single image is required for training. This method has several benefits:

- By using the mask texture for recognition, no additional image alignment is required and pose variations are accounted for automatically during the mask fit. Thus, recognition is possible without the need for feature selection to align test and training images.

- The 3D person-specific shape of the deformable face mask is used to compensate for different pose and lighting conditions during fitting and recognition.
- Image alignment and scaling is unnecessary because recognition is performed on the mask texture by comparing the colour values of each mask vertex instead of comparing pixels. The mask also crops the face area and thus voids the influence of the background on the recognition performance.
- The model based fitting as well as the recognition methods are suitable for different resolutions. The number of mask vertices and not the resolution of the image determines the resolution of the mask texture.

1.3 Structure of the Thesis

The structure of this thesis is as follows:

In Chapter 2 a review of the current state of the art in video processing, including object detection, model based fitting, object tracking, image super-resolution and object recognition is given. Different approaches for fitting a 3D deformable model to an image are reviewed and different 3D face masks are compared. This is followed by an introduction to model based approaches for object tracking in low resolution, inclusive of both feature and appearance based methods. Next, the basic problem of super resolution is formulated and existing super resolution approaches are compared. Finally, object recognition methods for character recognition as well as 2D, 3D and multi-modal face recognition are elaborated.

Chapter 3 presents a method for automatic object detection and pose and shape estimation, the first step towards automatic video processing. Here, a deformable 3D face model is fitted to a single image of a human face using an appearance model previously built from a set of training images. Experiments on two face databases demonstrate the use of the model based fitting approach under different pose and lighting conditions as well as in low resolution.

Once the object is detected and its pose is estimated, object tracking can be used to estimate the pose parameters for every frame of a video sequence. Thus, in Chapter 4 a method for combined tracking and super resolution is developed. The result of a model based tracking approach is used to combine several video frames to increase the resolution of the tracked object. But unlike traditional super resolution approaches only the resolution of the object is increased on mask level rather than enhancing the image of the entire scene. Experiments on planar and non-planar, including different faces are conducted and evaluated.

After object detection, object tracking and super resolution, object recognition is usually the last step for processing surveillance video. Chapters 5 and 6 present two model based object recognition methods for planar and non-planar objects respectively. The planar objects used for recognition in Chapter 5 are characters. Each character of the alphabet is modelled and a large number of different templates covering different character appearances in different resolutions are created. The recognition approach is based on template matching and allows for simultaneous character separation and recognition on low resolution images. Experiments on car license plates and text documents show the ability of the proposed approach to recognise small characters of down to five pixels in size.

Such a template based recognition approach is only feasible for planar objects with a limited number of parameters that change the appearance of the object. Therefore, the recognition method presented in Chapter 6 uses a deformable 3D object model to recognise non-planar objects, namely human faces. Only a single training image is used and recognition is performed directly on the mask texture. The 3D deformable face model is fitted to a new image of a person's face and recognition is performed without the need for additional pose estimation or image alignment. The proposed face recognition approach is compared with existing methods and experiments on a face database demonstrate its use for face recognition under different pose and lighting conditions.

Finally, Chapter 7 summarises the findings of this thesis and provides possible future directions for model based video processing methods.

Chapter 2

Background

The research in this thesis involves different fields of computer vision, including model-based fitting, object tracking, image super-resolution as well as object recognition. An overview of the current state of the art in each of the respective fields is given in the following sections. Firstly, for model-based fitting of a 3D mesh model Principal Component Analysis (PCA) and PCA based methods are reviewed. Different 3D face mesh models and their respective fitting approaches are then compared with respect to their advantages and disadvantages for low resolution image processing. Following that, model based approaches for object tracking in low resolution video are introduced. Feature based and appearance based tracking methods are presented and the theory of particle filters is explained. The resolution of low resolution images and videos is enhanced by so called super resolution methods. The basic problem formulation of super resolution methods is given and the super resolution optical flow approach is introduced. Subsequently, object recognition methods for images of characters and images of human faces are presented. Existing approaches for text document recognition and number plate recognition in low resolution images are compared and an overview of 2D, 3D and multi-modal 2D and 3D face recognition methods are given. Finally, the challenges and open problems of approaches for model based low resolution image processing are summarised.

The background chapter is organised as follows: Section 2.1 gives an overview of methods for model-based fitting of a face model to an image, which is the main scope of Chapter 3. Section 2.2 introduces methods commonly used for tracking objects through video sequences, followed by traditional super resolution algorithms in Section 2.3. Both, tracking and super resolution methods are combined in Chapter 4 of this thesis. Section 2.4 and Section 2.5 familiarise the reader with the state of the art in object recognition, in particular character recognition and face recognition, which are the main topics of Chapter 5 and Chapter 6.

2.1 Model-Based Fitting

Model-based fitting methods adjust a deformable 3D model of an object to fit an image of a particular type of object. The object types used within this thesis are human faces. Principal Component Analysis (PCA) is a popular method commonly used in model-based fitting approaches, its details are described in Subsection 2.1.1 followed by three different model-based fitting approaches based on PCA. Section 2.1.2 then compares different 3D face models and reviews methods that use deformable 3D meshes for model-based fitting in the context of human faces.

2.1.1 Principal Component Analysis

Principal Component Analysis (PCA) is a fundamental method in data processing. It is a mathematical algorithm that calculates the eigenvalue decomposition, given a square data matrix A. Image processing methods usually use PCA for dimensionality reduction or the construction of new images. Therefore, 2D images are vectorised to J_i and the data matrix $\bar{A}$ is constructed from these vectors as $\bar{A} = [J_1^T - \bar{x}, J_2^T - \bar{x}, .., J_n^T - \bar{x}]$. The data is centred around the mean and multiplied with its transpose to result in a squared data matrix as:

$$A = \bar{A}\bar{A}^T \tag{2.1}$$

where the columns of $\bar{A}$ are vectorised 2D images J_i and $\bar{x}$ is the mean of $\bar{A}$. The matrix A can then be decomposed into its eigenvectors and eigenvalues as:

$$A = X\Lambda X^{-1} \tag{2.2}$$

where the columns of X contain the eigenvectors and Λ is a diagonal matrix containing the corresponding eigenvalues. If the matrix A is constructed from 2D images of faces only, the eigenvectors are also called Eigenfaces. These eigenvectors span a vector space for all the data points J_i. The dimensionality of this space is reduced by keeping only the eigenvectors that correspond to the largest eigenvalues and neglecting the eigenvectors for which the corresponding eigenvalues are below a defined threshold as:

$$A_{reduced} = (\bar{A} - \bar{x})X_{reduced} \tag{2.3}$$

where $X_{reduced}$ contains a reduced set of eigenvectors that correspond to the largest eigenvalues. These eigenvectors span a smaller vector space and thus, reduce the dimensionality of the data matrix A.

Additionally, each data point can be represented as a linear combination of eigenvectors and any new data point can then be reconstructed from these eigenvectors as:

$$\hat{\mathbf{J}} = \bar{\mathbf{x}} + \mathbf{X}\mathbf{X}^T(\mathbf{J}_{new} - \bar{\mathbf{x}}) \tag{2.4}$$

where $\mathbf{J}_{new}$ is a new vectorised 2D image and $\hat{\mathbf{J}}$ represents the same image reconstructed as a linear combination of eigenvectors X. The reconstruction error or the distance from feature space is then defined as:

$$e = ||\mathbf{J}_{new} - \hat{\mathbf{J}}||_2 \tag{2.5}$$

PCA was first applied to images of faces in the late 80's (Sirovich and Kirby, 1987) and later used for face recognition (Turk and Pentland, 1991). In the context of model based fitting PCA is often used for training, i.e. model building. A set of training images of a particular object is used for calculating the eigenvectors, which are then used for reconstructing a new instance of an object that was not included in the training set.

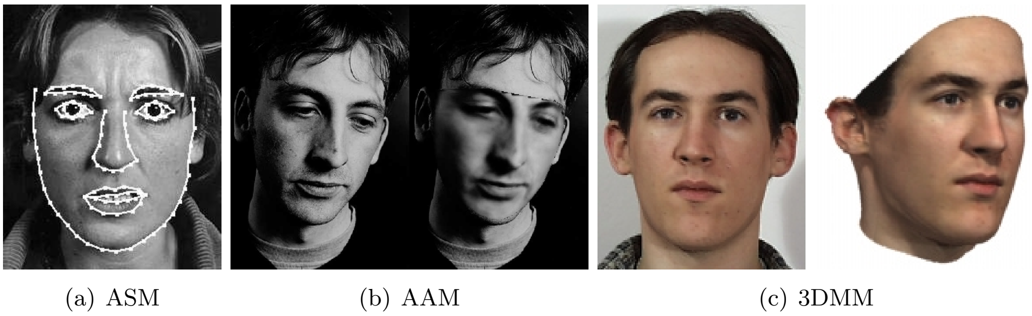
The disadvantage, however, is that in order to represent a large variety of object appearances, the training set needs to cover all these instances. For example, a training set consisting of male faces only is not sufficient to represent female faces. Thus, the training set is crucial for the performance of PCA when reconstructing previously unseen images.

The approaches presented in the following sections use PCA to reduce the dimensionality of the training data and thus new object instances can be represented and compared efficiently.

2.1.1.1 Active Shape Models

Active Shape Models (ASM) are a statistical model that have first been proposed in Cootes *et al.* (1995). They are mainly used to detect feature points along contour lines in images. These contour lines are learnt from a set of training images with labelled object boundaries. Figure 2.1(a) shows such an example training image with labelled contours.

Each image of the training set is labelled with a set of feature points connected through contour lines. These images are aligned to a common coordinate system and PCA is used to reduce the dimensionality for a concise and efficient representation. Only the eigenvectors with the largest eigenvalues are kept to model the distribution of the training


(a) ASM

(b) AAM

(c) 3DMM

Figure 2.1: Different face models based on learning the vector space from a set of labelled training images using PCA. (a) Active Shape Models (ASM) model only the intensity values along object contour lines. (b) Active Appearance Models (AAM) learn the shape as well as the appearance of a particular type of object, like a human face, from a set of training images. (c) 3D Morphable Models (3DMM) calculate the eigenvectors from a set of 3D laser scans and thus, represent 3D shape and texture.

data. Each training contour c can then be approximated as:

$$c \approx \bar{c} + \mathbf{X}b \tag{2.6}$$

where $\bar{c}$ is the mean of all training contours, $\mathbf{X}$ contains the eigenvectors corresponding to the largest eigenvalues and b is a vector of coefficients. Thus, varying b will change the shape of the contour c and to ensure similarity to the training set, b is usually limited by the eigenvalues λ as $\pm 3\sqrt{\lambda_i}$ (Cootes and Taylor, 1999).

When presented with a previously unseen image, the ASM algorithm iteratively improves the fit of the learnt object shape model. The parameter vector b is initially set to zero and the position of each feature point along the contour is optimised separately in the local surrounding area. The parameters of the shape model, i.e. the vector of coefficients b, a scaling factor and orientation and translation parameters are then updated to fit these new contour lines. This process is repeated until convergence.

Since the shape model is optimised only locally, the performance is highly dependent on the initialisation. A poor starting point may result in an unacceptable final fit. Furthermore, since PCA is used to calculate a shape model from a set of training images, this set needs to cover all possible object deformations to allow for a sufficient generalisation.

The basic ASM algorithm has been extended for example by Tu *et al.* (2004) using a hierarchical CONDENSATION (particle filter) framework and by Yan *et al.* (2003), where the shape estimation problem is formulated in a Bayesian framework.

2.1.1.2 Active Appearance Models

While Active Shape Models only optimise around the contour lines of the object, Active Appearance Models (AAM) (Edwards *et al.*, 1998) extend this approach and represent both, shape and texture of a given class of object. During the training process a generative model is built such that both shape and texture are controlled by a set of parameters. Again, PCA is applied to a set of labelled and normalised training images. The parameter vector b of the resulting appearance model then controls both the shape s and texture a of a new object as:

$$s = \bar{s} + \mathbf{X}_s b \qquad a = \bar{a} + \mathbf{X}_a b \tag{2.7}$$

where $\mathbf{X}_s$ and $\mathbf{X}_a$ are matrices derived from the eigenvectors of the shape model and the appearance model respectively. The vector of coefficients b controls both shape and appearance, since they are linearly dependent.

Finding the optimum model parameters b for a previously unseen image then equals to deforming the shape s and varying the appearance a of the model to fit the image. Therefore the image residual δI is calculated as the difference between the new image I_{new} and the image I_b created with the current appearance model parameter b as $\delta I = I_{new} - I_b$. Minimising δI then requires the calculation of the first derivatives with respect to b, i.e. the Jacobian. In practice it is sufficient to estimate the Jacobian from the training set using numeric differentiation by perturbing each of the coefficients in b and observing the resulting error (Cootes and Taylor, 1999). Figure 2.1(b) shows an example of a new image of a face (left) and the face shape and texture modelled by the AAM (right).

This approach improves the ASM algorithm in Section 2.1.1.1 as it takes into account not only the shape of an object but also its appearance. However this increases the complexity and decreases the computational performance of the fitting algorithm. Furthermore, the quality of the fit is only as good as the generalisation ability of the appearance model, again a diverse training set is needed to achieve best results.

Since it was first proposed, the original AAM algorithm has been extended and modified. For example the authors of Dedeoglu *et al.* (2006) include the down sampling of the camera into the fitting process, and thus fit an AAM model to faces in low resolution images. In Ayala-Raggi *et al.* (2008), AAMs have been extended to incorporate lighting changes using harmonic images. In Xiao *et al.* (2004), AAMs were combined with 3D Morphable Models to allow for a three-dimensional fitting.

2.1.1.3 3D Morphable Models

The previous two methods used PCA to build 2D shape and 2D appearance models from a set of training images. This concept has been extended to 3D laser scans of human faces. The authors of Blanz and Vetter (1999, 2003) generate a so called 3D Morphable Face Model from a set of accurate 3D laser scans. They acquired 200 face scans of 100 male and 100 female persons and similar to the AAM approach in Section 2.1.1.2 PCA is used to reduce the dimensionality of the 3D scans. The eigenvectors of the 3D shape S and the texture T are calculated from the training data set and a new 3D face, including its shape and texture, can then be represented as:

$$S = \bar{s} + \sum_{i=1}^{m-1} \alpha_i s_i, \qquad T = \bar{t} + \sum_{i=1}^{m-1} \beta_i t_i \tag{2.8}$$

where $\bar{s}$ and $\bar{t}$ are the average shape and texture vectors respectively and s_i and t_i are the eigenvectors of the shape and texture respectively that form an orthogonal basis. An arbitrary 3D face is then modelled as a linear combination of eigenvectors by varying the linear coefficients α_i and β_i.

In order to fit the model to a previously unseen image, the model parameters α_i and β_i are refined in an analysis-by-synthesis loop. The model is initialised manually and the residual between the deformed model and the image is calculated. The calculated error is then used to refine the model such that it converges to its optimum. In Blanz and Vetter (1999), this optimisation is defined in a Bayesian framework and a stochastic gradient descent is used to find the optimum.

Morphable models are now widely used, for example, in combination with model-based bundle adjustment of selected facial feature points (Dimitrijevic *et al.*, 2004) or in combination with shape-from-silhouette (Wang *et al.*, 2005b). Blanz *et al.* (2005) extended the 3D Morphable Model approach to accomplish face recognition using the estimated model parameters. The authors of Zhao *et al.* (2006) increased the speed of the reconstruction process by optimising the shape parameters only and extracting the weighted texture information from different images. In Wang *et al.* (2005a), an Expectation-Maximisation (EM) approach is applied to infer shape and texture parameters from their Syncretized Shape and Syncretized Texture Model respectively.

2.1.2 Deformable 3D Face Mesh Models

The methods in the previous sections applied PCA to create a face model from a set of labelled training images or 3D scans. The following approaches use predefined deformable 3D face models to fit images of human faces. These face models are usually 3D polygonal representations consisting of vertices and edges like in Ahlberg (2001). Such a deformable model can either be designed by an artist (Zhang *et al.*, 2004) or hand-crafted to fit a specific face (Goldenstein *et al.*, 2004b). It can also be derived from human facial muscles (Roussel and Gagalowicz, 2005) or include the MPEG-4 points to deform the facial mesh (Tang and Huang, 2008). The shape of these models can either be controlled by predefined metrics like the MPEG-4 standard or use physics based functions like deformable fields or radial basis functions to control the person-specific shape and expressions of the model.

The following Section 2.1.2.1 introduces a number of different 3D face models based on predefined metrics and more complex models are presented in Section 2.1.2.2. Each model is compared with respect to its size, the deformability of the mask and selected fitting methods.

2.1.2.1 Predefined Metrics and MPEG-4

The MPEG-4 standard defines a set of facial feature points and facial animations (Pakstas, 2002). These deformations are defined as fractions of distances between feature points and thus, the MPEG-4 standard can be used to control any 3D face model by selecting the feature points of the particular 3D model. This standard is often used to animate avatars to allow for low bit rate video transmissions (Tang and Huang, 2008).

Similar to the MPEG-4 standard other 3D face models with their own predefined metrics have been proposed in literature as shown in Table 2.1. Each of these models is described in more detail.

Directed Search

The CANDIDE-3 face model in Ahlberg (2001) is defined by 104 vertices and 184 triangles and controlled by 14 shape and 65 animation parameters. The neutral face mask $\bar{g}$ is linearly deformed as:

$$g = \bar{g} + \mathbf{S}\sigma + \mathbf{T}\alpha \tag{2.9}$$

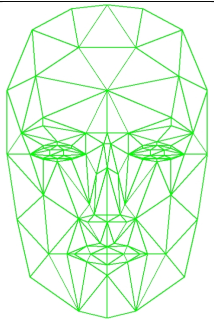
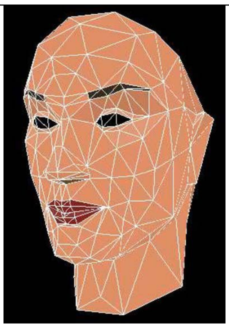
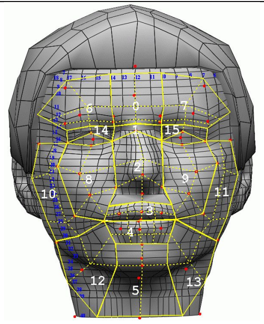
	 © 2008 IEEE.		 6 0 7 14 15 2 8 9 10 11 3 4 12 5 13 © 1999 IEEE.
Vertices	184	194	n\a
Triangles	104	360	16 Bézier volumes
Deformability	14 shape and 64 expression parameters	65 metrics	23 visemes and 6 universal expressions
Fitting Method	Locally Exhaustive & Directed Search	Model-Based Bundle-Adjustment, Non-Linear LMS Optimisation	manual selected feature points
Reference	(Ahlberg, 2001; Dornaika and Ahlberg, 2006)	(Zhang *et al.*, 2004; Lu *et al.*, 2001)	(Tao and Huang, 1999)

Table 2.1: Comparison of different 3D face mesh models using predefined metrics with respect to their size, deformability and methods for fitting the 3D face model to an image.

where **S** and **T** are matrices containing the shape and animation metrics respectively which are controlled by the linear coefficients contained in vectors σ and α. The conversion from shape and animation parameters to the MPEG-4 standard is given in Ahlberg (2001).

For fitting this model to an image of a known face a locally exhaustive and directed search is proposed in Dornaika and Ahlberg (2006). The best parameter vector $\mathbf{b} = [dx, dy, dz, \phi_x, \phi_y, \phi_z, \alpha]$ containing rotations $[\phi_x, \phi_y, \phi_z]$, translations $[dx, dy, dz]$ and the animation parameter α is calculated by minimising the reconstruction error in feature space (Equation 2.5). The values of $\mathbf{b}_i$ are systematically altered to locally explore the error function until a local minimum is found. The search along the direction of this local minimum then yields the final $\mathbf{b}$. This heuristic method calculates the best model parameters without calculating the derivatives of the error function. However, only the animation parameters α, but not the person-specific shape parameters σ, are optimised.

Model-Based Bundle-Adjustment

Similar to the previous model, the face model in Zhang *et al.* (2004) is defined by 194 vertices and 360 triangles and 65 metrics control its deformations. The proposed fitting approach is based on model-based bundle-adjustment. Given a video sequence of a face, a large number of image features p_{ij} are selected in two images of a video sequence and 3D points P_j are reconstructed using bundle-adjustment as:

$$\min_{P_j,\mathcal{P}_i} \sum_i \sum_j \mathcal{P}_i(P_j) - p_{ij} \tag{2.10}$$

where P_j is the jth feature point in 3D, $\mathcal{P}_i$ is the projection into the ith frame and p_{ij} is the jth 2D feature point in the ith image. This error function is minimised with respect to the 3D coordinates of all points P_j and the projection parameters for each frame i.

The face model is then fitted to the resulting 3D point cloud. This initial fit is further refined by utilising model-based bundle adjustment. The original error function is constrained by penalty terms based on the face model and all video frames are taken into account. The penalty terms reduce the search space and allow for a more efficient calculation. The disadvantage is the need for manually selected features and even though model-based bundle-adjustment is more computational efficiently than bundle adjustment, it is not applicable for real time applications.

Non-Linear Least Mean Square Optimisation

The authors of (Lu *et al.*, 2001) use the same face model as Zhang *et al.* (2004) but instead of model-based bundle-adjustment a feature point driven non-linear least mean square optimisation is used. Only a single image of a person's face is used to deform and fit the model. A trained ASM (Section 2.1.1.1) is used to automatically detect facial features along contour lines. These feature points p_j match predefined 3D points P_j on the generic face model. The assigned point-to-point and point-to-contour correspondences are then used to optimise for the initial pose parameters, i.e. three translation and three rotation parameters, as:

$$\min_{\mathcal{P}} \sum_j \mathcal{P}(P_j) - p_j \tag{2.11}$$

where P_j is the jth feature point in 3D, $\mathcal{P}$ is the projection into the image and p_j is the jth 2D feature point.

This is similar to the bundle adjustment approach except that only a single image is used for optimisation and the quality of the fit depends on the accuracy of the detected image features. Since the mask fitting is solely dependent on feature points, an erroneous feature detection will inevitably result in an inaccurate fit.

Bézier volumes

The face model in Tao and Huang (1999) is defined as a piecewise Bézier volume. The mesh is surrounded by a top and a bottom Bézier volume and deformations are modelled by predefined metrics as:

$$\mathbf{V} = B\mathbf{D} \tag{2.12}$$

where $\mathbf{D}$ controls the displacement of the Bézier volumes and $\mathbf{V}$ contains the resulting displacements of each mesh point. The matrix B is the mapping function of Bernstein polynomials. Assuming that $\mathbf{V}_0$ represents the neutral face model, expressions can be modelled as:

$$\mathbf{V}_0 + B[\mathbf{D}_0, \mathbf{D}_1 ... \mathbf{D}_M]P \tag{2.13}$$

where each $\mathbf{D}_m$ deforms the Bézier volume according to a specific expression and the vector of coefficients P controls the intensity of these expressions.

This model is used in Tang and Huang (2008) to track head movements and facial expressions of a person through a video sequence. The deformation parameters of the model are then converted into the MPEG-4 standard to animate an avatar. Thus, the result of the expression tracking needs to be expressed in terms of MPEG-4 facial animation parameters (FAP) as:

$$argmin||\Delta\hat{v}_i - \sum_{k=1}^{K} \Delta\hat{v}_i^k||_2 \tag{2.14}$$

where $\Delta\hat{v}_i$ is the movement of the i^{th} facial feature point resulting from the tracking and $\Delta\hat{v}_i^k$ is the k^{th} facial animation parameter (FAP) defined by the MPEG-4 standard. The MPEG-4 standard allows for standardised and comparable deformations across different face models but only allows for basic expressions sufficient to animate avatars, however more realistic human facial expressions require more complex models.

2.1.2.2 Complex Deformable Models

The complex deformable 3D face models presented in this section use either a multiple layer architecture (Roussel and Gagalowicz, 2005) or define the model deformation as radial basis functions (Park *et al.*, 2004) or deformable fields (Goldenstein *et al.*, 2004b). Because of their complexity these models are in general more flexible and can model slight shape differences better than models with fewer deformation parameters. However, adjusting these models to fit a specific face may also be more computationally expensive and images of high resolution are usually needed for fitting. The three models presented in this section are summarised in Table 2.2 and each of them are described in more detail.

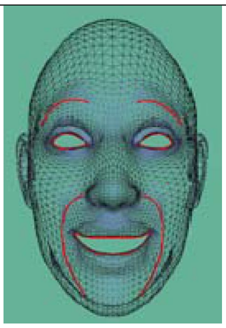
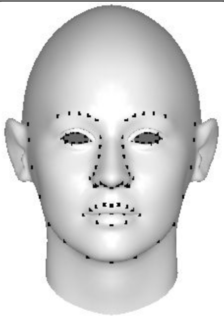
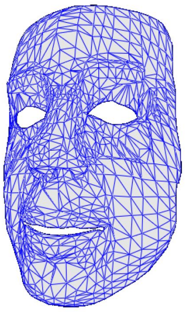
		 © 2004 IEEE.	 © 2004 IEEE.
Vertices	ca. 5000	n\a	1,101
Triangles	ca. 10,000	ca. 20,000	2,000
Deformability	4-layer architecture including MPEG-4 and Bézier curves	Radial Basis Functions for 160 key feature points	11 parameters
Fitting Method	manual selected feature points	feature points along contour lines	manual fitting using Deformable Fields
Reference	(Roussel and Gagalowicz, 2005)	(Park *et al.*, 2004)	(Goldenstein *et al.*, 2004b, 2003)

Table 2.2: Comparison of different 3D face mesh models with respect to their size, their deformability and methods for fitting the 3D face model to an image. All images are taken from the respective references.

Hierarchical Anatomy Driven Face Model

The face model in Roussel and Gagalowicz (2005) is inspired by the anatomy of the human face. The MPEG-4 animation standard is extended to include facial muscles for more flexible and realistic deformations of the face. The deformations of the face model are controlled by four layers. The first layer defines radial basis functions for smooth deformations of selected 3D points, while the second layer uses the MPEG-4 standard to control specific feature points as well as Bézier curves to deform predefined curves along the face mesh. The two highest layers then use these predefined deformations to model specific facial movements like 'lower lip' or 'left eyelid' and to construct specific facial expressions like 'happy' or 'sad', at level three and four respectively. Due to the layered structure, this model is very flexible but requires high resolution images for accurate fitting and tracking. The layered architecture can be adapted to any face mesh model if desired.

Radial Basis Functions

In Park *et al.* (2004), a generic 3D head model is created as the mean across a number of 3D head scans. To accurately adjust this generic head model to a specific person two or more high resolution 2D images are required. Radial basis functions are then defined to fit the generic head to a frontal and side view image of the new person. A number of image feature points are located along contour lines and the corresponding 3D head model points are assigned.

A function $f(\bar{p})$ then deforms every 3D point $\bar{p}$ of the head model to fit these 2D-3D feature point correspondences. Deformations are driven by radial basis functions which are defined around 160 key feature points selected from the 3D head model. The distance between a 3D point $\bar{p}$ and a feature point determines the strength of the deformation: the further away the weaker the deformation. Using constraint optimisation the parameters of the function $f(\bar{p})$, i.e. the deformation forces, are estimated. Similar to the previous approach such a realistic 3D head model can only be created from high resolution images, like in this example a frontal and a side view.

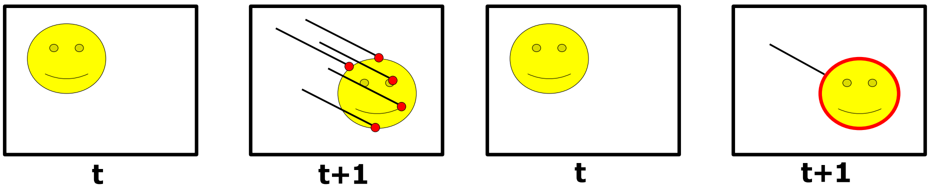
Deformable Fields

Instead of using a rigid mesh with pre-defined metrics the authors of Goldenstein *et al.* (2004b, 2003) use a dynamic face mesh model and deformable fields are used to adjust the model to a specific face. The deformable face mask consists of 1,101 vertices that are connected by edges to form 2,000 triangles, but instead of metrics that deform selected vertices in a pre-defined way, a deformable field is used for modelling deformations. The position of every 3D model point p_i is calculated by a function F_i as:

$$p_i = F_i(q, u_i, v_i) \tag{2.15}$$

where q is a vector of deformation parameters and (u_i, v_i) is a point in (u, v) space. This vector field applies the deformation in 2D space and then returns the resulting 3D vertices of the mesh. The deformation parameters affect the vertices close to the centre of the deformation most, while smaller forces are applied to vertices farther away.

This type of mesh is also defined as a multi-resolution structure. Starting from a coarse base mesh, binary multi-triangulation is used to generate a higher resolution mesh. The deformations are decoupled from this as they are defined in (u, v) space. This model has only been applied to high-resolution images of faces where a sufficient number of feature points is available for fitting or tracking (Goldenstein *et al.*, 2004a; Metaxas *et al.*, 2004).



(a) Feature Based Tracking

(b) Appearance Tracking

Figure 2.2: (a) Single features are tracked from frame t to frame $t+1$ (b) The appearance of the entire object is used for tracking.

2.2 Model Based Tracking

Tracking describes the process of locating an object in successive frames of a video sequence, where the location of the object is usually defined in respect to the camera coordinate system. In 2D tracking the transformation between video frames can be described with affine transformations or projective transformation, e.g. homography, while 3D tracking requires the calculation of the 3D pose and orientation of the object. The pose parameters T^{ext} are then defined as translations $[dx, dy, dz]$ and rotations $[\phi_x, \phi_y, \phi_z]$ around the camera axis.

Most 2D tracking approaches work well for planar objects and under constant lighting conditions. However, when tracking non-planar objects, large view point changes may alter the appearance of the object and cause most 2D based trackers to fail. Model based tracking tries to overcome these limitations by incorporating the 3D model of an object to help find the new pose and orientation in the next frame. By knowing the 3D geometry of the object, changes in pose and lighting can be modelled and thus, better accounted for.

Model based tracking can either be feature based or appearance based as shown in Figure 2.2. Feature based methods track single features from the previous frame to the current frame and then estimate the pose parameters and the deformation parameters of the object model by fitting it to retrieved feature points in the current frame. Appearance based tracking methods used the entire texture of the face, rather than single feature points, to find the new position in the next frame. Representatives of both methods are described in the following sections.

2.2.1 Feature Based Tracking

Feature based tracking methods usually calculate a motion field first and then deform the object model to fit the new feature points. Optical flow (Gautama and van Hulle, 2002) is commonly used for calculating a two-dimensional motion field based on image intensities. Given a feature at image location (x, y) in frame i the optical flow assumes that the intensity of this feature is constant in the next frame $i + 1$ as:

$$I(x, y, i) = I(x + \Delta x, y + \Delta y, i + 1) \tag{2.16}$$

where $I(x, y, i)$ is the intensity of the image feature at location (x, y) in frame i and $(\Delta x, \Delta y)$ is the feature point displacement in image coordinates.

The position of each feature $(x + \Delta x, y + \Delta y)$ in frame $i + 1$ can be calculated using block-matching algorithms (Shi and Tomasi, 1994) or differential methods, for example. A good overview including a performance evaluation is given by Barron *et al.* (1994). The accuracy of the motion field is highly dependent on the selected features. Feature points lacking a rich local texture are more likely to be mismatched, i.e. their position may not be recovered accurately.

The facial tracking method proposed in Tao and Huang (1999) uses a template matching approach to estimate a motion field in two consecutive frames. The 3D motion of the entire mask, i.e. translation and rotation as well as the expression parameters are then calculated using a least square estimator by linearising the derivatives. This way single mismatched image features are corrected by using the deformability of the mask as constraint. This tracking approach is extended to learn new facial expressions that are not yet represented by the face model. After each frame the resulting tracking error is analysed and the predefined deformation parameters are adjusted, if required.

2.2.2 Appearance Tracking

Appearance based tracking methods use a textured 3D model of the object to find the location of that object in the next frame. Instead of an accurate 3D face model the method proposed by Cascia *et al.* (2000) uses a cylindrical head model to track the motion of a person's head. It is argued that a more complex head model does not improve the quality of the track as it requires more parameters and is less robust to perturbations in the initial position. The cylindrical model is initialised automatically using the result of a face detector. The authors reported that it was not possible to automatically initialise a complex human head model.

The cylindrical model is projected into the first frame of the tracking sequence and a reference texture J_0 is extracted. This reference texture is used to calculate so called warping templates by perturbing the initial position. A displacement matrix $N_a = [n_1, n_2, ..., n_K]$ stores the displacements of each pose parameter $T = [dx, dy, dz, \phi_x, \phi_y, \phi_z,]$. Each warping template o_k is then calculated as:

$$J_0 = \mathcal{P}(I_0, T_0)$$
$$o_k = J_0 - \mathcal{P}(I_0, T_0 + n_k)$$

where $\mathcal{P}$ projects the cylindrical head model into image I_0 using parameters T_0 and J_0 is the resulting reference texture. In practise, four displacements per parameter are sufficient.

The illumination is modelled similarly as a linear system using illumination templates. These templates U are calculated by applying Singular Value Decomposition (SVD) to a large set of training images of different persons under different lighting conditions. The difference in illumination between two frames can then be approximated as:

$$J - J_0 \approx Uc \tag{2.17}$$

where the columns of U are the illumination templates and c is a vector of coefficients.

Tracking is then realised by calculating the difference between the reference texture J_0 and the texture of the current frame J. Using the warping templates and the illumination templates this difference is approximated as:

$$J - J_0 \approx Oq + Uc \tag{2.18}$$

where the columns of O contain the warping templates. The linear coefficients q and c are estimated as a weighted and regularised least square solution.

According to Wen and Huang (2005) the best model-based tracking results in low resolution video are obtained by combining feature based and appearance based methods. They initialised the face mask manually in the first frame and for each new frame two texture residuals are calculated using an appearance based and a feature based tracking approach. The method that results in the smallest texture residual is then chosen for the current frame.

2.2.3 Particle Filters

Particle filters (Kitagawa, 1987; Doucet *et al.*, 2000) are statistical tools to model non-linear discrete time series and are used for feature based tracking as well as appearance based tracking. They are a special variant of the Kalman Filter that allows the state variables to be non-Gaussian distributed. The theory of Monte Carlo samples is applied to estimate the posterior probability distribution of the current state x_t. When used for tracking, each state represents the location and/or pose parameters of the tracked object in the current video frame t. The posterior of each state is factorised as follows:

$$P(x_t|y_{1:t}) \propto P(x_1)\prod_{t=1}^{T} P(y_t|x_t)\prod_{t=2}^{T} P(x_t|x_{t-1}) \tag{2.19}$$

where x_t is the state at time $t, t = 1, 2, 3, ...T$ and y_t are the observations at time t, which can be low level image features, for example. Each state usually describes the location and/or pose parameters of frame t and all observations are conditionally independent given the state.

The conditional state density $P(x_t|y_{1:t})$ at time t is represented by a set of samples, i.e. particles, $x_t^{(i)}$ with corresponding weights $w_t^{(i)}$ representing the sample probability. Given a set of particles at time $t-1$, new samples at time t are drawn depending on the sampling scheme (MacKay, 1998). The bootstrap particle filter uses importance sampling and Monte Carlo samples $x_t^{(i)}$ are drawn as:

$$x_t^{(i)} \sim P(x_t|x_{t-1}) \tag{2.20}$$

$$\widetilde{w}_t^{(i)} = P(y_t|x_t), \qquad w_t^{(i)} = \frac{\widetilde{w}_t^{(i)}}{\sum\limits_i \widetilde{w}_t^{(i)}} \tag{2.21}$$

where $\sim$ means “sample from” and $w_t^{(i)}$ is the weight associated with sample i. After each time step t all particles are resampled (Doucet *et al.*, 2000).

There are a number of variations to this particle filter approach which differ in the way the samples are drawn, or the resampling rate. The annealing particle filter (Deutscher and Reid, 2005) for example decreases the spread of sampling distribution in order to ensure convergence.

Such an annealing particle filter tracking approach, combined with incremental weighted Principal Component Analysis (PCA), is used in Tu *et al.* (2006) to track faces in low resolution. They use a face model that only covers the top part of the face as this part is mostly undisturbed by facial expressions. The samples are used to represent translation and rotation parameters in 3D space and incremental PCA is applied to model appearance variations and thus, track the face.

2.3 Super Resolution

Many computer vision algorithms require images of high resolution. The higher the resolution, the more detailed the scene and the better the performance of the image processing method. Unfortunately such high resolution images and videos are rarely available, especially in surveillance tasks. Thus, super resolution offers a cost-effective way to increase the resolution of videos and images and has been well studied in the last decades (Chiang and Boult, 2000; Lin and Shum, 2004a; Park *et al.*, 2003; Tanaka and Okutomi, 2005).

In general super resolution methods increase the resolution of a single image or a whole video sequence and can be formally treated as single frame and multi-frame approaches using spatial and frequency information (Huang and Tsai, 1984; Borman and Stevenson, 1998). Park *et al.* (2003) give a good introduction to the technical side of super resolution. The theory of super resolving low resolution images is introduced in the next sections.

2.3.1 The Inverse Problem

The Observation Model is based on the assumption that every image is warped, blurred and down sampled by the imaging system. Therefore each high resolution image I^{high} can be transformed into a low resolution image I^{low}, under the assumption that I^{low} remains constant during the super resolution process, as:

$$I^{low} = MI^{high} + z \quad \text{with} \quad \mathrm{M} = \mathrm{SBW} \tag{2.22}$$

where M is the system matrix that represents the imaging system consisting of warping W, blurring B as well as sub-sampling S and z is additive random noise (Park *et al.*, 2003). When dealing with colour images the sub-sampling matrix S can be further divided into $S = DA$ according to Farsiu *et al.* (2004), where D represents the generic down-sampling by a constant factor and A specifies the colour filter effects on colour images.

As part of the system matrix M, the warping matrix W contains the transformations from the world coordinate system into the camera coordinate system, i.e. global and local translations and rotations. The estimation of this matrix is crucial as the warping describes the sub-pixel shifts of the low resolution images with respect to the high resolution image. The blur which is described by the blurring matrix B is the result of the point spread function (PSF) of the camera, motion blur or other effects caused by the optical system, e.g. out of focus or aberrations. The PSF of an optical system is usually approximated as a Gaussian filter. A more realistic model for the blurring caused by the camera is the Bessel function.

The problem of finding the high-resolution image is now equivalent to inverting the system matrix M, e.g. solving the inverse problem. However inverting the system matrix M is computationally complex because M might be ill conditioned or even singular. Thus, most approaches approximate I^{high} by defining a cost function and minimising:

$$\hat{I}^{high} = \underset{I^{high}}{\operatorname{argmin}} \, ||I^{low} - MI^{high}||_2^2. \tag{2.23}$$

A commonly used cost function is the L_2 norm. However Farsiu *et al.* (2004) report that using the L_1 norm instead of the least mean squares approach is more tolerant to outliers. Additionally, constraints in the form of Lagrangian multipliers or regularisation terms can be added to control the smoothness (Baker and Kanade, 2002).

2.3.2 Image and Video Super Resolution

The simplest way of increasing the resolution of a single image is by interpolation using nearest neighbour or spline interpolation. However, interpolation alone is unable to recover high-frequency details of the image or video and is therefore not truly regarded as 'formal' super resolution (Park *et al.*, 2003).

Super resolution methods that reconstruct high resolution images from video sequences apply the image formation process of Equation 2.22 to several frames (Farsiu *et al.*, 2004) or use a Bayesian approach to estimate images of higher resolution (Baker and Kanade, 2002). The four main steps for increasing the resolution of a video sequence according to Chiang and Boult (1996) are:

1. Finding Correspondences
2. Warping into Coordinate Systems

3. Fusion of Frames
4. Deblurring

The first step finds pixel correspondences in all images of the sequence in order to detect sub-pixel movements. Common techniques are block-matching algorithms like the sum of absolute differences (SAD), optical flow (Gautama and van Hulle, 2002) or KLT (Shi and Tomasi, 1994). Accurate image registration is important and affects the quality of the super resolution result as shown in Zhao and Sawhney (2002). The more precise the motion estimation, the better the resulting high-resolution images. However the precise detection of sub-pixel movements is particularly hard in low resolution images (Barreto *et al.*, 2005), especially when dealing with non-planar and non-rigid objects under changing lighting conditions. According to Baker and Kanade (1999) most existing super resolution algorithms are therefore not suitable for video sequences of non-planar and non-rigid objects, like faces.

After the estimation of pixel correspondences and sub-pixel movements, all frames are warped into one coordinate system. This warping usually involves interpolating the low resolution images using standard techniques like nearest-neighbour interpolation or spline interpolation (Chiang and Boult, 1996).

In the third step the different frames are combined to result in a single super resolved image. One of the simplest techniques is calculating the mean or the median. More complex methods include non-uniform interpolation, weighted nearest neighbour interpolation or wavelet interpolation.

The deblurring of the resulting high resolution image is an optional last step. Common deconvolution techniques are the Wiener filter or the Lucy-Richardson algorithm, both algorithms require an initial guess of the underlying point spread function.

Pre-requisite for increasing the resolution of video sequences by combining several frames are sub-pixel shifts between consecutive frames. Typical super resolution techniques assume that the camera is moving as the scene is recorded, but a moving camera results in motion blur which decreases the quality of the images, so according to Ben-Ezra *et al.* (2005) traditional cameras should avoid motion blur as much as possible.

Recent studies involve the use of special cameras to capture super resolved video sequences directly. The so called jitter camera is used in Ben-Ezra *et al.* (2005) and creates sub-pixel offsets between frames during recording. They also show that motion blur degrades the result of super resolution algorithms, even if the motion blur itself is known, and should therefore be avoided. The authors of Agrawal and Raskar (2007) use a special, so called, flutter shutter camera. This camera preserves the high frequencies by opening and closing the shutter frequently during exposure. A single camera is extended to capture super resolved stereo images in Gao and Ahuja (2006) by recording the scene through a transparent rotating plate in front of the camera.

Other existing approaches attempt to obtain a super resolved image directly during the tracking process. For example the method proposed by Dellaert *et al.* (1998) presented a Kalman Filter approach for model-based motion estimation and tracking by simultaneously increasing the resolution. Instead of conventional methods for motion estimation like the sum of squared differences (SSD) or sum of absolute differences (SAD) a Kalman Filter approach is applied. The novelty is the use of the texture map within the measurement model of the Kalman Filter. The state variables represent the 3D pose and the texture of an image patch. The resolution of the image is improved by assuming a high-resolution image that is then projected into each frame of the sequence using the camera model and additive noise.

2.3.3 Super Resolution Optical Flow

Super resolution optical flow (Baker and Kanade, 1999) is a common technique to increase the resolution of a video sequence. It typically comprises the following five main steps:

1. *Image Interpolation* - interpolate each frame to twice its size
2. *Image Registration* - estimate the optical flow between consecutive frames
3. *Image Warping* - warp images into a reference coordinate system
4. *Image Fusing* - fuse images using mean, median or robust mean
5. *Deblurring* - apply standard deconvolution algorithms to super-resolved image

After each frame is increased to twice its size, optical flow is used to register consecutive frame which are then warped into a reference coordinate system and finally fused. While this approach is well suited to increase the resolution of images of rigid and planar scenes, image registration is more difficult for non-rigid and non-planar low-resolution objects. The first step of the super resolution optical flow algorithm interpolates each frame to twice its size using standard interpolation techniques such as nearest neighbour or bilinear. However, interpolation cannot recover high-frequency details in images. In addition it introduces artificial random noise that is difficult to remove in the deblurring step. Warping images, the third step, also involves the interpolation of pixels which introduces further noise.

In Yu and Bhanu (2006) super resolution optical flow is extended and planar patches are used to track different parts of the face individually to account for the non-rigidity of the face. The resolution of the face is increased for these different facial parts individually.

2.3.4 Super Resolution via a 3D Model

Using a 3D model of a particular object can help calculating super resolved images of this object. A Bayesian approach based on model-based surface reconstruction is proposed in Smelyanskiy *et al.* (2000). Their approach estimates the 3D surface together with the lighting conditions to synthetically render high resolution images given a set of low resolution observations of the scene.

They assume a user supplied parameterised surface-model, which includes a triangular mesh representing the 3D surface and a reflectance model with the albedo of each triangle. A synthetic image is rendered using the model parameters as well as the parameters from the camera (position and orientation). The resulting images, one for each camera view, are then compared with the actual observed low-resolution images and the difference is used to update the parameters.

The key idea and the difference to traditional rendering in terms of computer graphics is the different use of pixels and triangles. Traditional computer vision rendering techniques map each triangle of the 3D model to more than one pixel in the image. The texture of each triangle consists of several image pixels because each surface triangle is usually much larger then a pixel. However super resolution is only possible if each triangle is smaller than a pixel. Therefore, the approach proposed by Smelyanskiy *et al.* (2000) uses a 3D mesh with triangles smaller than pixels, when projected into the image.

The approach proposed by Dedeoglu *et al.* (2006) improves the fitting of Active Appearance Models (AAM) to low resolution video sequences. A so called Resolution Aware Formulation (RAF) is incorporated into the error function of the AAM. The error function usually ranges over all pixels in the face appearance template and thus require the input image to be of the same size as the template. In case of resolution differences interpolation is applied which affects and decreases the result of the fitting algorithm.

The RAF algorithm models the camera and the appropriate point spread function (PSF) and the error function therefore minimises the AAM parameters with respect to the camera model. The error is calculated as the squared difference between the low resolution input frame and the simulated low resolution image depending on the AAM parameters and the PSF. Once fitted correctly the AAM corresponds to the super resolved image of the object represented by the AAM.

2.3.5 Hallucinating - Face Super Resolution

A learning based super resolution approach that is commonly applied to images of faces is called Hallucinating. A prior is learnt from a set of training face images on the spatial distribution of the image gradient (Baker and Kanade, 2000). In a super resolution approach this prior increases the resolution of the input image by up to eight times in size.

The approach has been further improved by dividing each image into a set of overlapping patches from which the prior is learnt (Liu *et al.*, 2005). The high resolution image is then reconstructed from these patches. A further improvement includes tensors to represent facial expressions (Jia and Gong, 2006).

However, all these approaches are class based and work exclusively on objects represented within the training set. The applied super resolution methods reconstruct the high resolution image from the learnt prior rather than inverting the system matrix in Equation 2.22, which is the more general approach.

2.3.6 Limits on Super Resolution

The first investigations of possible limits of super resolution techniques are conducted by Kosarev (1990). He showed that there is an absolute limit for resolution enhancement by Shannon's theorem. The maximum Shannon resolution limit is:

$$\frac{1}{3}\log_2\left(1+\frac{P_s}{P_n}\right) \tag{2.24}$$

where P_s is the signal energy and P_n is the signal noise. The ratio between P_s and P_n is usually called the signal to noise ratio (SNR or S/N).

A more recent study on super resolution limits is done by Lin and Shum (2004b). They examined reconstruction-based super resolution algorithms. According to them the theoretical magnification factor limit is 5.7 whereas the the practical limit is 1.6 or 2.5. They also determined the number of low resolution images needed for achieving these magnification factors M. If M is not an integer the sufficient number of low resolution images is $[2\lceil N_h - M \rceil]^2$, where N_h is the square root of the number of high resolution pixels.

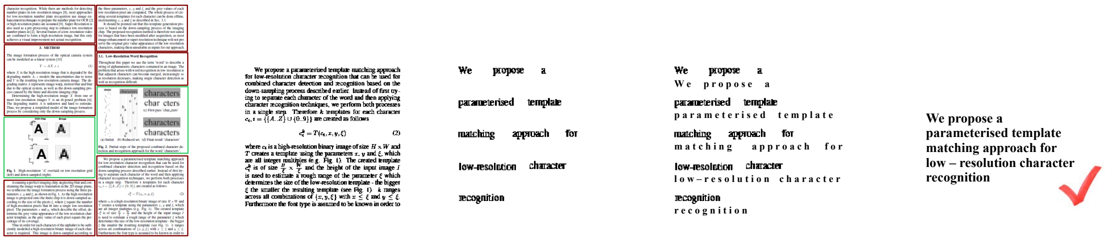
2.4 Character Recognition

The problem of low resolution character recognition arises in many situations such as number plate recognition in large scale surveillance situations or the recognition of street signs in surveillance video. Also text documents captured by web cameras or mobile phones require algorithms that detect and recognise characters in low resolution images and videos. This section gives an overview of existing approaches for character recognition, in particular text document recognition and number plate recognition.

2.4.1 Text Document Recognition

Optical character recognition (OCR) is a method widely used for text document recognition with a large number of commercial software available. It is mainly used to convert scanned text documents into machine readable text files. In general, OCR algorithms comprise the following five main steps as illustrated in Figure 2.3:

1. Layout detection
2. Binarisation
3. Segmentation
4. Recognition
5. Spell check


We propose a
parameterised template
matching approach for
low-resolution character
recognition
We propose a parameterised template matching approach for low – resolution character recognition

1) Layout Detection 2) Binarisation 3) Segmentation 4) Character Recognition 5) Spell Check

Figure 2.3: The basic steps of common optical character recognition methods include layout detection to separate text paragraphs from images and figures, followed by binarisation to convert the grey scale image into black and white. Each paragraph is then segmented into single words and characters, which are then recognised individually followed by an optional spell check.

The first step is to detect the layout of the provided image document and to separate images and other non-text areas from the text. A good overview of existing text extraction techniques is given by Jung *et al.* (2004). Once the text sections are detected they may be normalised to increase the contrast for better recognition results before being converted into binary images. These binary text images are then segmented into single words and each word is segmented into single characters. The cropped images of binary characters are then compared against a database of characters or recognised by a trained classifier. A spell checker can then be used to correct single characters and words based on a dictionary.

The training of classifiers usually requires the extraction of features from a set of training images of different characters. A good overview of different feature extraction methods for recognising segmented single characters is given by Trier *et al.* (1996). Existing methods are compared and tested with respect to their suitability for different applications.

Even though OCR is the standard method for recognising text documents it cannot be applied to low resolution images of text. Most existing OCR approaches require a resolution of at least 300dpi for a A4 page of font size ten which corresponds to a character height of at least 30 pixels (S. Rice and Nartker, 1996). However low resolution characters can be as small as five pixels in height.

Most OCR algorithms work on binary images only, but characters of less than 20 pixels in height do not show clear edges but appear instead as amalgamates of aliased pixels. Binarising these text images will result in degraded and concatenated characters which are hard to separate into single letters, causing most OCR methods to fail.

2.4.1.1 Low Resolution Text Recognition

Character and text recognition in low resolution images is a challenging task due to merging characters and unclear character edges. Existing approaches that tackle these problems can be separated into approaches that enhance the quality and resolution of the text images and approaches that recognise the low resolution text directly.

A super resolution based approach is proposed by Dalley *et al.* (2004) to enhance the quality of low resolution text images. Pairs of grey scale low and high resolution image patches of characters are used to train a Bayesian framework. During testing the most likely super resolved patch is inferred from a given low resolution image patch. This approach produces a visual improvement only but no recognition results are reported.

Another approach that improves the visual appearance of low resolution text images uses resolution expansion (Thouin and Chang, 2000). The problem of estimating the high resolution image is formulated as a constrained non linear optimisation problem that is solved iteratively. The resulting images of enhanced text achieve better recognition results than images improved by standard resolution expansion methods.

Enhancement and super resolution methods improve the visual appearance of low resolution text images and can improve the recognition accuracy of OCR methods. However, most enhancement methods modify the image in a way that may create artefacts that adversely reduce the performance of standard OCR methods. Thus, the following approaches recognise low resolution words without prior image enhancement.

A method based on dual eigenspace decomposition for low resolution character recognition is proposed in Sun *et al.* (2005). A large set of degraded low resolution characters is generated synthetically from binary character images of higher resolution. These images are down-sampled, zoom interpolated and result in blurred and degraded characters that are then used for training. A heuristic approach is chosen to separate low resolution words into single characters which are then classified by a dual eigenspace classifier. The disadvantage of this method is the need for segmented single characters.

The approach proposed by Einsele *et al.* (2008) recognises low resolution words directly without prior character segmentation. They specialise on web images with anti-aliased small size text and utilise a Hidden Markov Model (HMM). A sliding window technique is used to extract features, i.e. slices, from a set of training images. The HMM is then trained on these features with states symbolising characters. To recognise a new image of a word the HMM is then used to model the entire word.

Another similar algorithm is proposed by Jacobs *et al.* (2005). Again features are extracted from a set of training images and a neural network-based character recogniser is trained. In the recognition step the low resolution word is cut into slices and the neural network recogniser returns the most likely character for each slice. Dynamic programming is then used to recognise the entire word given the recognition result of each image slice.

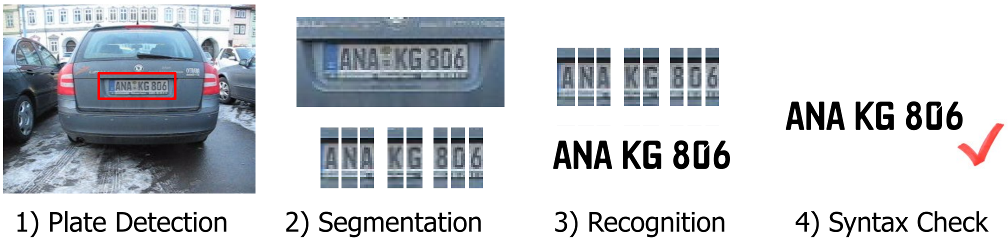
2.4.2 Number Plate Recognition

Number plate or license plate recognition methods are widely used for identifying vehicles for access authorisation or traffic infringements. A large number of commercial software solutions are available and number plate recognition methods usually consist of the following four steps as illustrated in Figure 2.4

1. Plate Detection
2. Character Segmentation
3. Character Recognition
4. Regional Syntax Check (optional)

The detection of the number plate within the current frame or image is the first step. Once the plate is detected it is normalised to adjust for orientation, size and skew as well as brightness and contrast of the image. In the next step the number plate is segmented into single characters and standard optical character recognition (OCR) methods are applied to recognise each character individually. A regional syntax check to improve the recognition result is optional.

A good overview of different detection, segmentation and recognition approaches is given by Anagnostopoulos *et al.* (2008). Most approaches presented in the literature (Anagnostopoulos *et al.*, 2006; Lee *et al.*, 2004; Chang *et al.*, 2004; Jiao *et al.*, 2009) achieve recognition rates of 90% and above when tested on high resolution images of number plates. Images are considered of high resolution when the characters are at least 20 pixels in height. This is usually realised by a relatively short distance between the camera and the vehicle's number plate. Smaller character heights may result in merged characters that are hard to separate and recognise using OCR methods.



1) Plate Detection 2) Segmentation 3) Recognition 4) Syntax Check

Figure 2.4: The basic steps of number plate recognition methods include plate detection to localise the plate within the image. The cropped plate is then segmented into single characters which are then recognised separately. A syntax check is optional.

2.4.2.1 Low Resolution Number Plate Detection and Recognition

Large scale surveillance situations often result in low resolution images of number plates due to low resolution cameras or large distances between the camera and the plate. Methods for detecting number plates in low resolution images have been proposed in the literature (Wu *et al.*, 2006), however the following problems arise for character segmentation and recognition in low resolution images. Low resolution characters tend to merge along their edges with the next character, making segmentation difficult, but without character separation standard OCR methods will fail.

Most methods for detecting number plates in low resolution images use low level image features. The method proposed by Wu *et al.* (2006) uses the morphological operation 'bottom-hat' to detect possible number plate candidates. Heuristic criteria like size and shape and number of zero crossings are used to validate the plate candidates. No further recognition is applied to the detection results.

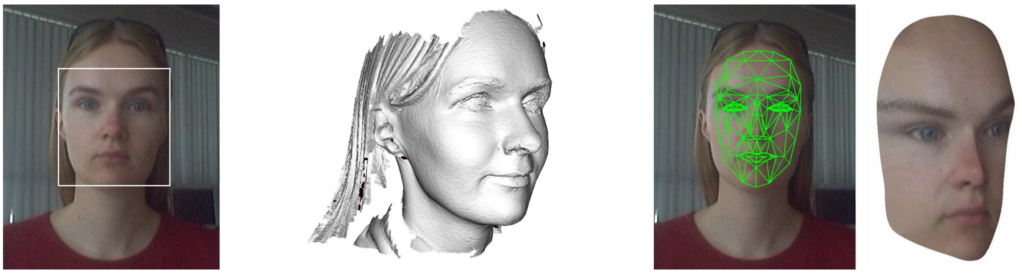
In Zhang *et al.* (2006), a cascading AdaBoost like classifier is built from local and global low level image features. Firstly, global features are used to eliminate the background, followed by classifiers trained with local Haar-like features to accurately detect the license plates. Detection rates of 93.5% are reported but again no recognition is performed on the results.

Image enhancement and super resolution methods are also applied to increase the quality of low resolution number plate images. The method proposed by Suresh *et al.* (2007) uses a maximum a posteriori (MAP) based super resolution method to increase the resolution of low resolution number plates. Several low resolution frames of a video sequence are fused to result in a high resolution number plate image. By using a different cost function the MAP based super resolution method is capable of real-time processing (Yuan *et al.*, 2008).

However, the applied super resolution is for visual improvement only and no recognition has been applied on the results.

2.5 Face Recognition

Face recognition methods are divided into 2D face recognition, 3D face recognition and multi-modal approaches. Classical 2D face recognition approaches use a 2D intensity image of a person's face to recognise its identity. With the availability of 3D scanners, 3D face recognition approaches became popular in the late 1980s (Bowyer *et al.*, 2004). Instead of using 2D intensity images 3D face recognition methods use the 3D shape of the face for recognition. Multi-modal face recognition approaches combine the advantages of both 2D and 2D face recognition and are assumed to achieve better results than 2D or 3D face recognition alone (Bowyer *et al.*, 2005).



(a) 2D Face Recognition

(b) 3D Face Recognition

(c) Multi-Modal Face Recognition

Figure 2.5: Different approaches for face recognition. (a) 2D face recognition uses 2D intensity images for recognising faces, while (b) 3D face recognition is applied to 3D data of the face shape. (c) Multi-modal face recognition approaches combine the advantages of both methods by fusing 2D intensity and 3D shape information.

2.5.1 2D Face Detection and Recognition

Face detection determines the location of a person's face within an image whereas face recognition identifies the actual person. Most approaches for face detection use low level 2D image features to train a classifier. A commonly used face detector is proposed by Viola and Jones (2001). Haar-like low level image features are used to train a cascade of different classifiers to obtain a robust face detector. A good survey of face detection techniques is presented by Yang *et al.* (2002).

Face recognition on 2D intensity images dates back to the mid 1960s (Bledsoe, 1966) and a large number of different algorithms have been proposed since. Common methods include Principal Component Analysis (PCA) (Turk and Pentland, 1991), Independent Component Analysis (ICA) (Draper *et al.*, 2003), Linear Discriminant Analysis (LDA) (Etemad and Chellappa, 1996), Kernel Methods (Kim *et al.*, 2002), Active Appearance Models (AAM) (Cootes and Taylor, 1999) or Bayesian Methods (Moghaddam *et al.*, 2000) with two comprehensive overviews given by Chellappa *et al.* (1995) and Zhao *et al.* (2003).

Two standard face recognition methods commonly used for comparison as described in (Georghiades *et al.*, 2001) are correlation and PCA. Correlation is the simplest recognition method. A new image is recognised by calculating the distance in image space to all training images stored in the face database. The image with the highest correlation, ie. the nearest neighbour is chosen. Depending on the size of the face database this approach requires a large amount of storage and can be computationally expensive.

In order to decrease the amount of storage needed PCA is commonly used for dimensionality reduction. The eigenvectors, also called eigenfaces, of a set of training images are calculated and only the eigenvectors that correspond to the largest eigenvalues are kept in order to reduce the dimensionality. A new image is recognised by projecting it into the reduced feature space and calculating the distance to all other training images. The image that results in the smallest distance is chosen.

However most 2D face recognition approaches are unable to satisfactorily solve the problem arising from different pose and illumination conditions as well as facial expressions, occlusions or ageing. 2D image intensity information alone is not sufficient to unambiguously identify a person's face under such a variety of environmental conditions.

2.5.2 3D Face Recognition

3D face recognition approaches use the 3D shape of person's face instead of intensity images for recognition and are therefore assumed to overcome the problems of 2D face recognition (Bowyer *et al.*, 2004). The main advantage of using the 3D face shape for recognition is its robustness against changing lighting conditions, different head positions or varying facial expressions.

Different methods for acquiring the 3D shape of a person's face include laser scanners or range sensors. The main drawback however is, that most 3D acquisition techniques do not operate in real time and require the person's cooperation, and even though the 3D face shape is illumination-independent, most acquisition methods are not. The lighting conditions do effect the acquisition process and may result in poorly reconstructed 3D shapes (Bowyer *et al.*, 2005). Thus, most 3D face recognition approaches to date are not suitable for large scale surveillance tasks with uncooperative subjects.

2.5.3 Multi-modal 2D + 3D Face Recognition

Recent studies (Bowyer *et al.*, 2005; Husken *et al.*, 2005; Abate *et al.*, 2007) have shown that in order to improve existing face recognition methods a multi-modal approach combining 2D and 3D face recognition outperforms 2D or 3D face recognition alone. Multi-modal face recognition uses either 3D sensors for acquiring the 3D shape of the face or calculate the 3D shape from a set of 2D images. The third type of multi-modal approaches incorporates previously acquired 3D information to fit a 3D face model to the 2D image of the face. Common to all methods is the use of 3D shape and 2D intensity information for multi-model face recognition.

Different methods for acquiring the 3D shape from a set of 2D images include depth-from-stereo, photometric stereo, structured light or shape-from-motion. A good survey is given by Chan *et al.* (2002). Different methods are examined with respect to their advantages and disadvantages for the use of face recognition.

Multi-modal approaches that use pre-recorded 3D information to recognise faces in 2D images are most promising since they combine the advantages of both methods by minimising the disadvantages of 3D sensors. Well known methods like Active Appearance Models (Xiao *et al.*, 2004) or Graph Matching (Husken *et al.*, 2005) are improved by incorporating 3D information, and new approaches like 3D Morphable Models (Blanz and Vetter, 2003) are introduced.

3D Morphable Models (3DMM) have recently been used for face recognition (Blanz and Vetter, 2003). Using an analysis-by-synthesis loop for estimating 3D shape, pose, texture and lighting from a single image, this method is very time intensive and has been applied to high-resolution images only. In order to improve the run time of 3DMM, in Xiao *et al.* (2004) 3DMM have been combined with Active Appearance Models (AAM). This method optimises 2D texture as well as 3D shape and is less computationally expensive, but again requires high-resolution images with a sufficient amount of facial features.

Similarly to the combined AAM and 3DMM approach the authors of Husken *et al.* (2005) extend Hierarchical Graph Matching (HGM) to include the 3D shape. The resulting recognition rates strengthen the assumption that a multi-modal face recognition approach is more powerful than 2D or 3D face recognition alone.

2.5.4 Face Recognition From A Single Image

Face recognition approaches commonly use a number of different images per person for training. These images may be taken under different illumination conditions and under different poses to ensure a diverse training set and thus, accurate and robust recognition results. However, this requires the person to willingly volunteer and to participate in the training process, but in large scale surveillance situations no more than a single image per suspect may be available for training.

Several face recognition methods that use a single training image per person only have been proposed, with a good survey given by Tan *et al.* (2006). However most of these methods modify 2D face recognition approaches (Yang *et al.*, 2004) or utilise the 3D shape of the person's face to increase the training set by creating novel views under different illumination (Hu *et al.*, 2004; Lu *et al.*, 2006).

Recently a face recognition approach based on 3D Morphable Models and spherical harmonics has been proposed (Zhang and Samaras, 2006). Using only a single training image the 3D Morphable Model is fitted to the person's face image on a analysis-by-synthesis loop and the illumination is expressed as a linear combination of spherical harmonic images created from the 3D Morphable Model. The model parameters are then used for recognitions. This fitting method requires either a manual initialisation or the assignment of facial feature points and high resolution images.

2.5.5 Modelling Illumination for Face Recognition

Different lighting conditions can change the appearance of a person's face drastically, presenting a challenge to most face recognition algorithms as illustrated in Figure 2.6. Different methods for modelling the illumination have been proposed in the past. These methods are either based on a sub-space representation of 2D images under different lighting conditions or use the 3D shape of a face for modelling illumination.

Most 2D face recognition approaches need a large number of training images to sufficiently model illumination changes. These training images are then used to construct a low-dimensional person-independent subspace that is suitable for modelling varying lighting conditions. Since the first attempts in the mid 1990s, a number of approaches using Principal Component Analysis (PCA) or other dimensionality reduction techniques have been proposed (Hallinan, 1994; Belhumeur and Kriegman, 1998). In order to increase the performance of PCA based recognition approaches the first three eigenfaces with the largest eigenvalues are discarded. In practise this has been proven to increase recognition under varying lighting conditions (Belhumeur *et al.*, 1997). The Linear Subspace method (Georghiades *et al.*, 2001) uses a number of training images for each subject under different illumination to construct a three-dimensional linear subspace. New images are then recognised by calculating the distance to each linear subspace. The subspace of the subject that is closest is assigned to the new image.

The gradient angles method proposed in Chen *et al.* (2000) is a 2D face recognition method. Changes in lighting are accounted for by illumination invariant features - image gradients. During the pre-training phase the joint probability density function is calculated from a set of 1280 images of 20 objects, no faces are amongst the objects. Then, only a single frontal image of each subject under frontal illumination is used for training and recognition requires the manual location and alignment of the face images.

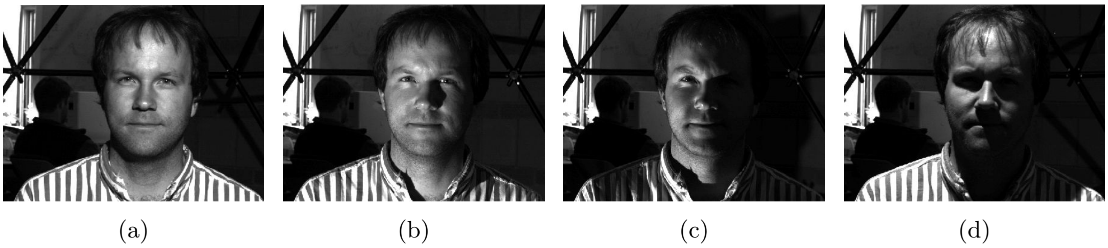

(a) (b) (c) (d)

Figure 2.6: The appearance of a person's face changes drastically under different illumination conditions presenting a challenge to most face recognition methods. Images taken from the Yale Face Database (Georghiades *et al.*, 2001).

In Shashua and Riklin-Raviv (2001) an illumination invariant representation of face images, the so called quotient images are presented. Given a single image of a person and a database of a number of different people under different lighting conditions, the quotient image can be used to re-render or recognise a new person even under new illumination conditions. However the disadvantage of this method, as well as PCA based methods, is the large number of training images needed to construct the subspace.

To overcome this problem, a method has been proposed in recent years that uses only a small number of images to construct the illumination subspace. According to (Georghiades *et al.*, 2001) only a small number of training images is required to construct a face illumination subspace. This so called illumination cone representation is based on the theory that the images of convex shaped objects with Lambertian reflectance under different illumination form convex cone. A small set of training images under different lighting conditions is used to construct the shape and the albedo of the person's face using a variant of photometric stereo. The reconstructed face shape and albedo is then used to synthesise images of the subject under different poses and illumination conditions and these images are then used to create a low dimensional illumination cone for each pose. During recognition the identity of the illumination cone with the smallest Euclidean distance is assigned to the new image. The Cones - cast method constructs the illumination cone with cast shadows whereas the Cones - attached method only allows for attached shadows and shading. Another approach that uses photometric stereo is proposed by Zhou *et al.* (2004). They use only a single image under unknown illumination to extract the albedo and surface normals of the face. All appearances of human faces are handled in a single class and Lambertian reflectance is used to model the illumination.

The illumination cone representation is further investigated by Lee *et al.* (2001, 2005). Instead of obtaining or synthesising a large number of images and applying dimensionality reduction methods to create the illumination cone, the authors of Lee *et al.* (2005) show that such a subspace is also spanned by only five to nine images of the subject. Point light sources are arranged in such a way that a small number of images taken of the subject under specific illumination settings is sufficient for representing its illumination cone. No dimensionality reduction methods need to be applied. Real images as well as synthesised images are used.

The theoretical explanation of the dimensionality reduction was first given by Basri and Jacobs (2003) and Ramamoorthi (2002), who independently applied the spherical harmonic representation to images of faces. They showed that the illumination of a convex Lambertian object can be represented by nine harmonic images which are derived from lighting functions defined on the surface of a sphere. Each harmonic image only depends on the surface normals and the albedo of the object. So given a 3D model of an object, i.e. a human face, the harmonic images can be calculated and the linear combination of the first nine harmonic images is sufficient to model most illumination conditions, including multiple light sources.

Instead of obtaining or synthesising a large number of images or taking a few images under specific lighting conditions, these so called basis images that span the illumination cone of a subject can also be generated from a 3D shape model according to (Basri and Jacobs, 2003). Thus, 3D Morphable Models (3DMM) are used together with spherical harmonics to recognise faces in Zhang and Samaras (2006). Only a single training image of each subject is required and by semi-automatic fitting a 3DMM the face shape is recovered and then used for generating nine basis images for each pose. Recognition is then performed by calculating the distance to each subspace spanned by the basis images. The identity the corresponds to the smallest distance is assigned to the new image.

2.6 Conclusion

The previous sections reviewed existing model based approaches for processing, enhancing and recognising objects in low resolution images and video. The first subsection compared existing methods for fitting a 3D face model to an image of a person's face. All of the reviewed methods require high resolution images for accurate model fitting and a large number also requires manually selected facial feature points. A fully automatic approach is proposed in Lu *et al.* (2001). Their method automatically detects feature points and then fits a deformable face model to these points. However, the fitting of the deformable mask is only as accurate as the feature point detection and might fail in low resolution images.

Six different deformable face mesh models are compared with respect to their size, deformability and fitting methods. In general, a deformable mesh based on a small set of predefined metrics can be fitted more efficiently than complex meshes containing several deformation layers or a large number of possible deformations. However high resolution images are required for fitting a complex and very flexible model which then results in an accurate, person-specific 3D face shape approximation.

The review continued by investigating feature based and appearance based tracking methods. While feature based methods track single features only and may therefore suffer from accumulated tracking errors, appearance based tracking methods try to match the entire appearance of the object to find the pose parameters in the next frame. Again low resolution images pose problems to most tracking approaches due to a lack of prominent image features. The method by Wen and Huang (2005) combined feature based and appearance based methods and showed that a combined approach achieves best results especially in low resolution images.

Super resolution methods enhance the resolution of a single low resolution image or combine several video frames to create an image of higher resolution. Traditionally the image formation process is inverted to enhance the resolution of a single image, but this requires the estimation of the system matrix M in Equation 2.22 that might be ill conditioned or even singular. To overcome this problem video super resolution uses optical flow to combine several video frames based on tracked image features (Baker and Kanade, 1999). However this approach will fail for non-planar and non-rigid objects, because changes in pose and object deformations will void the underlying assumptions and may result in distorted super resolution images.

Continuing, different methods for object recognition are reviewed, namely character recognition and face recognition. Standard optical character recognition methods work on binary images of single characters and thus require a minimum image resolution. Character images of less than 20 pixels will result in distorted binary images depending on the applied threshold, and may result in recognition errors. Additionally, characters start to merge with decreasing image resolution and show no clear edges which impedes character separation as well as recognition. Existing approaches that recognise low resolution characters without prior character separation are based on Hidden Markov Models (Einsele *et al.*, 2008) or dynamic programming (Jacobs *et al.*, 2005).

Lastly, face recognition methods based on 2D images, 3D face models and multi-model approaches combining 2D and 3D are reviewed. The main problems of 2D face recognition algorithms are changes in pose and lighting. 3D face recognition methods try to overcome these problems by using the 3D shape of the face. However, acquiring the 3D face shape requires the person to voluntarily participate which makes these methods unsuitable for wide area surveillance applications. Multi-modal approaches combine the advantages of both 2D and 3D face recognition and are believed to outperform both (Bowyer *et al.*, 2005). The 3D Morphable Model approach is such a method that works on 2D images but includes the 3D shape information implicitly by fitting the 3D Morphable Model. A face recognition approach using only a single training image has been proposed (Zhang and Samaras, 2006). However, their method and a large number of existing face recognition methods are not automatic, they usually require a number of feature points to align the faces for training and for recognition. However the precise detection of facial feature points is difficult especially under different pose and illumination conditions.

The background chapter reviewed a number of different approaches from various fields of computer vision, including model-based fitting, object tracking, image super resolution as well as object recognition. In wide area surveillance situations all of the above methods are faced with poor, low resolution video images. However, most of the presented techniques require high resolution images or manual intervention to achieve best results. Furthermore, the revision showed a need for fully automatic methods especially for model-based fitting and object recognition. Conclusively, the review identified open problems in computer vision and the potency to improve the performance of methods in low resolution images. The following chapters explore methods for model-based fitting, object tracking, image super resolution and object recognition to handle automatic processing of low resolution video images.

Chapter 3

Automatic Fitting of a Deformable Face Mask

Detecting a person's face and precisely calculating its pose and shape parameters is indispensable for a large number of image processing algorithms such as model-based tracking or face recognition. However, the appearance of the face can be affected by a variety of factors, such as different lighting conditions, different camera viewing angles, facial expressions or the resolution of the recorded image. All these factors make accurate face pose and shape detection a challenging task, especially in uncontrolled environments.

Most existing approaches require controlled indoor environments (e.g. access control system (Messer *et al.*, 2003)) or recover only a rough estimate of the face position (Viola and Jones (2001)). However, the height and width of the face alone are insufficient for accurate initialisation or alignment tasks. More precise information like the 3D coordinates of the face and its shape are needed for tasks like model-based tracking or facial recognition applications.

Previous methods address the problem of fitting a deformable face model to a new image by first finding facial features and then fitting the face mask to these points (Lu *et al.*, 2001). Active Shape Models (ASM) is a common method for detecting such facial features (Tu *et al.*, 2004) whereas Active Appearance Models (AAM) are used to model the facial shape and texture, thereby detecting the face as a whole (Edwards *et al.*, 1998).

This chapter proposes a new method for fitting a 3D deformable face model to a single image of a person's face. Using Principal Component Analysis (PCA) together with a novel texture mapping method, an appearance model is built which is then used within a particle filter based fitting algorithm. Only a single image of a previously unseen person is needed for accurate 3D model fitting. Lighting invariance is achieved by incorporating the work of Basri and Jacobs (2003) into the error function of the particle filter. Thus, the proposed approach tackles the problem of automatic 3D mask fitting, lighting invariance and unlike most ASM and AAM methods, is also suitable for low resolution images.

The proposed approach differs from the AAM in that the facial shape is already implicitly given by the deformable face mask model. Only the facial texture is learnt from a set of training images using PCA and the deformation parameters are directly taken from the face model. Furthermore, unlike 3D Morphable Models (Xin *et al.*, 2005) the proposed approach does not aim to generate a 3D model of a person's face. The deformable face model utilised is not flexible enough to model subtle shape differences.

This chapter is organised as follows: The proposed automatic mask fitting algorithm is described in Section 3.1, including the generation of the appearance model in Section 3.1.1, the lighting invariance in Section 3.1.2, the particle filter refinement in Section 3.1.3 and the automatic fitting algorithm in Section 3.1.4. The experiments are outlined in Section 3.2.

3.1 Automatic 3D Face Mask Fitting

The proposed method automatically fits a 3D deformable face mask to a single image of a previously unseen person under different lighting conditions and in low resolution. An overview of the proposed algorithm is shown in Figure 3.1.

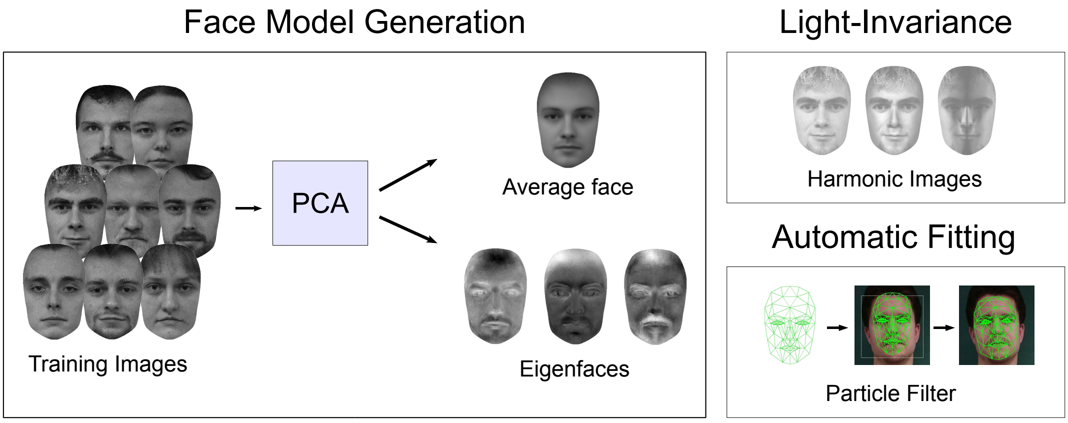


Figure 3.1: Overview of the automatic mask fitting approach. The 3D appearance model is built from a set of training images using Principal Component Analysis (PCA). The lighting invariance is achieved through harmonic images (Basri and Jacobs, 2003), generated from the 3D face mask. Lastly, a particle filter based fitting algorithm combines the 3D appearance model and the harmonic images to fit a deformable face mask to a single image.

The pre-requisite for the fitting algorithm is a 3D appearance model, which is built only once in the first step of the proposed approach. This 3D appearance model consists of a mean face and a set of principal components, also called Eigenfaces, which are generated from a set of training images using Principal Component Analysis (PCA) (Turk and Pentland, 1991).

The deformable 3D face mask is also used to generate so called harmonic images, based on the work of (Basri and Jacobs, 2003), in order to achieve lighting invariance. Finally, a particle filter based automatic fitting algorithm combines the 3D appearance model together with the harmonic images to fit the deformable face mask to a single image of a previously unseen person. Each of these modules is described in detail in the following sections.

3.1.1 3D Appearance Model Generation

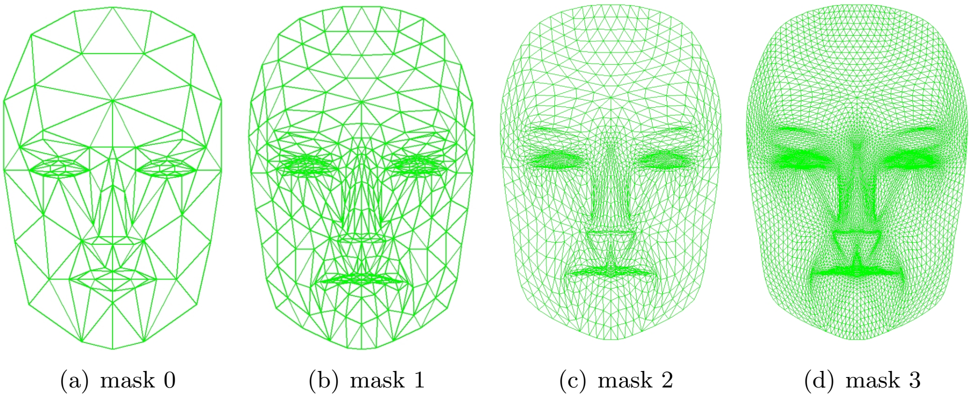
The 3D appearance model used within the proposed fitting algorithm is built off-line and only once. Therefore, the deformable 3D face mask CANDIDE-3 (Ahlberg, 2001) is fitted semi-automatically to a set of training images. PCA is then used to calculate the 3D appearance model, consisting of a mean face and a set of Eigenfaces. The following sections describe the deformable CANDIDE-3 face mask, as well as the model generation process in detail.

3.1.1.1 3D Face Mask

The deformable 3D face mask CANDIDE-3 (Ahlberg, 2001) is used for generating the 3D appearance model. This 3D mask is defined by 104 vertices and 184 triangles and a set of 14 shape and 65 animation parameters control its appearance. Each shape parameter determines a person-specific face shape, whereas the animation parameters deform the neutral face to allow for expressions. Thus the deformable mask mesh is described as follows:

$$\mathbf{g} = \overline{\mathbf{g}} + \mathbf{S}\gamma \tag{3.1}$$

where $\overline{\mathbf{g}}$ is the neutral face mask as shown in Figure 3.2, the columns of $\mathbf{S}$ are the shape parameters and the linear coefficient $\gamma_k \in [-1..1]$ controls the k^{th} shape deformation; expressions are neglected. Each shape parameter is a list of vertices and the displacement needed to achieve the particular deformation.



(a) mask 0 (b) mask 1 (c) mask 2 (d) mask 3

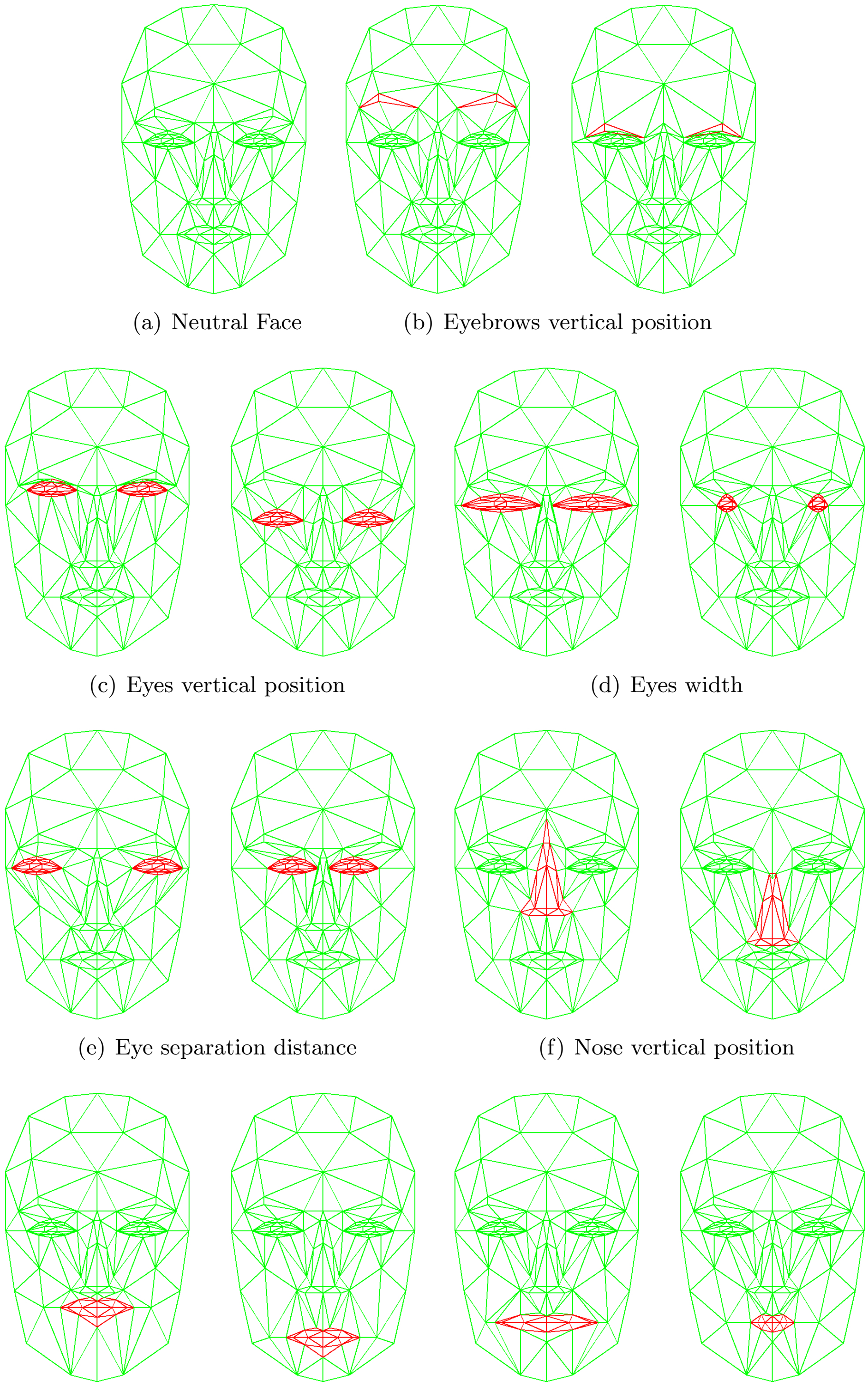
Figure 3.2: (a) Original CANDIDE-3 face mask with 184 triangle, (b), (c) and (d) are subdivided masks after 1, 2 and 3 subdivision steps resulting in 736, 2944 and 11776 triangles.© 2008 IEEE.

Table 3.1 lists all shape parameters of the CANDIDE-3 face mask. This list is reduced to seven shape parameters (left column) which model the most significant person-specific shape deformations, neglecting the shape parameters (right column) that change the neutral mask only marginally. The reduced set of shape parameters is sufficient to adjust the deformable mask to fit different faces, the main aim of this approach. The effect of these shape parameters for $\gamma_k = 1$ and $\gamma_k = -1$ is shown in Figure 3.3. The deformability of the mask does not allow for a precise modelling of a person's face shape, like the exact shape of the nose or the chin region for example. The proposed approach aims to automatically detect the location of facial features by fitting a generic deformable 3D face mask to an image of a face. The experiments in Section 3.2 will show that this is sufficient in dealing with low resolution images under different lighting conditions.

In addition to the original CANDIDE-3 face mask (Figure 3.2(a)) the proposed fitting algorithm also uses a finer sampled mesh. Therefore, the original face mask is subdivided three times using the Modified Butterfly algorithm and the Loop subdivision as described

Used parameters	**Additional parameters**
Eyebrows vertical position	Head height
Eyes vertical position	Eyes height
Eyes width	Cheeks z-extension
Eye separation distance	Nose z-extension
Nose vertical position	Nose pointing up
Mouth vertical position	Eyes vertical difference
Mouth width	Chin width

Table 3.1: All 14 shape parameters of the CANDIDE-3 face mask. Only the shape parameters in the left column are used by the proposed approach.



(a) Neutral Face

(b) Eyebrows vertical position

(c) Eyes vertical position

(d) Eyes width

(e) Eye separation distance

(f) Nose vertical position

(g) Mouth vertical position

(h) Mouth width

Figure 3.3: Extreme shape deformations for different shape parameters for $\gamma_k = 1$ (left) and $\gamma_k = -1$ (right). For comparison the neutral generic face mask $\overline{\mathbf{g}}$ is shown in (a).

in Appendix A. This subdivision algorithm divides each mask triangle into four new triangles by adding new vertices and keeping the position of the old vertices mainly unchanged. The result after one, two and three subdivision steps is shown in Figure 3.2(b,c and d). The finest sampled mask consists of 5984 vertices and 11776 triangles.

3.1.1.2 The 3D Appearance Model

The CANDIDE-3 face mask is used to generate the 3D appearance model. This deformable mask is fitted semi-automatically to a set of training images of faces. A number of key features are manually selected and the mask is automatically fitted using the Levenberg-Marquardt algorithm (Moré, 1977) for optimisation. The sum of the squared distances between the selected image feature points and the associated mask vertices is minimised in order to calculate the face-specific shape parameters γ and pose parameters $T^{ext} = [t_x, t_y, t_z, \phi_x, \phi_y, \phi_z]$ containing the rotation $[\phi_x, \phi_y, \phi_z]$ and translation $[t_x, t_y, t_z]$ of the mask with respect to the camera.

After the deformable mask is fitted to an image of a face, it is assigned its texture, but instead of traditionally mapping the texture of an area within the image to a mask triangle, each mask triangle is assigned with a single colour value only. The centre of each triangle is projected into the image as:

$$J = \mathcal{Q}(\mathcal{P}(\mathbf{g}, T)) \quad \text{with} \quad \mathrm{T} = [\mathrm{T}^{\mathrm{int}}, \mathrm{T}^{\mathrm{ext}}] \tag{3.2}$$

where $\mathbf{g}$ is the deformed face mask, $\mathcal{P}$ projects the centre of each mask triangle into the 2D image using camera parameters T. The intrinsic T^{int} camera parameters are determined by standard camera calibration techniques (Zhang, 2000). Note that a rough camera calibration is sufficient, since the actual size of the face is not important for the proposed approach. $\mathcal{Q}$ then creates a vector of concatenated colour values from the list of textured mask vertices.

The method of assigning each mask triangle with a single colour value has several advantages over traditional texture mapping techniques. Each mask triangle is assigned with a single colour value only. No image warping or interpolation is needed for texture mapping the image area to the triangle, only a single projection from the triangle centre to image coordinates is sufficient. Secondly, the number of triangles in the subdivided mask determines the resolution of the mask texture. The original CANDIDE-3 mask consists of only 184 triangles and thus 184 colour values. By subdividing this mask into a fine mesh the resolution of the mask texture increases with the number of triangles and the more triangles, the finer the mesh and the higher the resolution of the mask texture.

More importantly, by using a vector of concatenated colour values J instead of a traditional textured mask or a 2D image, no image normalisation is necessary to align different faces. The alignment is already implicit in the vector representation. Thus, Principal Component Analysis (PCA) (Turk and Pentland, 1991) can be applied to the vectors directly without the need for additional face alignment or normalisation.

PCA is applied to a set of vectors of concatenated grey values $\mathbf{J}$, generated from training images of faces. The result is an average face $\bar{\mathbf{x}}$ and a set of principal components $\mathbf{X}$, called Eigenfaces. Once calculated, a new face $\mathbf{J}_{new}$ can be represented as a combination of these principal components $\mathbf{X}$ as:

$$\hat{\mathbf{J}} = \bar{\mathbf{x}} + \mathbf{X}\mathbf{X}^T(\mathbf{J}_{new} - \bar{\mathbf{x}}) \tag{3.3}$$

where $\hat{\mathbf{J}}$ is image reconstructed from $\mathbf{J}_{new}$.

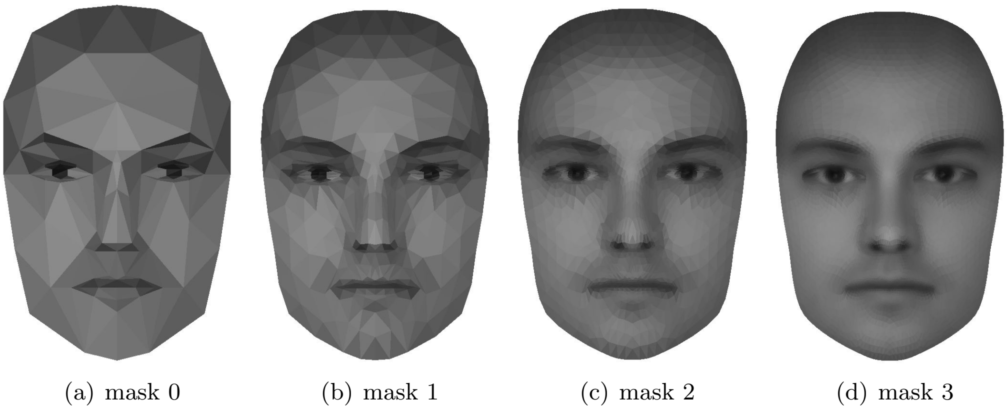

(a) mask 0 (b) mask 1 (c) mask 2 (d) mask 3

Figure 3.4: The mean face generated from a set of training images for different mask resolutions. The finer the mask mesh the higher the resolution of the mask texture.

An example of a mean face $\bar{\mathbf{x}}$ for different mask resolutions is shown in Figure 3.4. The finer the mask mesh, i.e. the more triangles, the higher the resolution of the mask texture. The mean face in Figure 3.4 is created as the average across all 40 persons of the IMM Face Database (Nordstrøm *et al.*, 2004), one image per person.

3.1.2 Light-Invariance through Harmonic Images

The lighting invariance is achieved through harmonic images (Basri and Jacobs, 2003). They are derived from spherical harmonics, a set of functions that form an orthonormal basis for functions defined on the surface of a sphere. All lighting functions that illuminate the surface of a sphere can be expressed as a linear combination with spherical harmonics. Furthermore, the authors of Basri and Jacobs (2003) show that given a 3D model and the albedo of a convex object, any image of this object under different lighting conditions can be approximated by a linear combination of harmonic images b_{nm} as:

$$I_i = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \alpha_{nm} b_{nm}(P_i) \tag{3.4}$$

where α_{nm} are linear coefficients and each harmonic image b_{nm} depends only on the 3D surface point P_i. Every image can now be represented as a linear combination of harmonic images and for simplicity Equation 3.4 is rewritten as

$$I = \sum_{h=1}^{9} \beta_h V_h(\eta, \rho) \tag{3.5}$$

where β_h is a linear coefficient of the h^{th} harmonic image and V_h returns the h^{th} harmonic image given the surface normal η and the albedo ρ of all surface points of the 3D model. As in Basri and Jacobs (2003) the proposed approach uses the first nine harmonic images, where each harmonic image is dependent on the albedo ρ and the surface normal η of each 3D surface point P. The first nine harmonic images are:

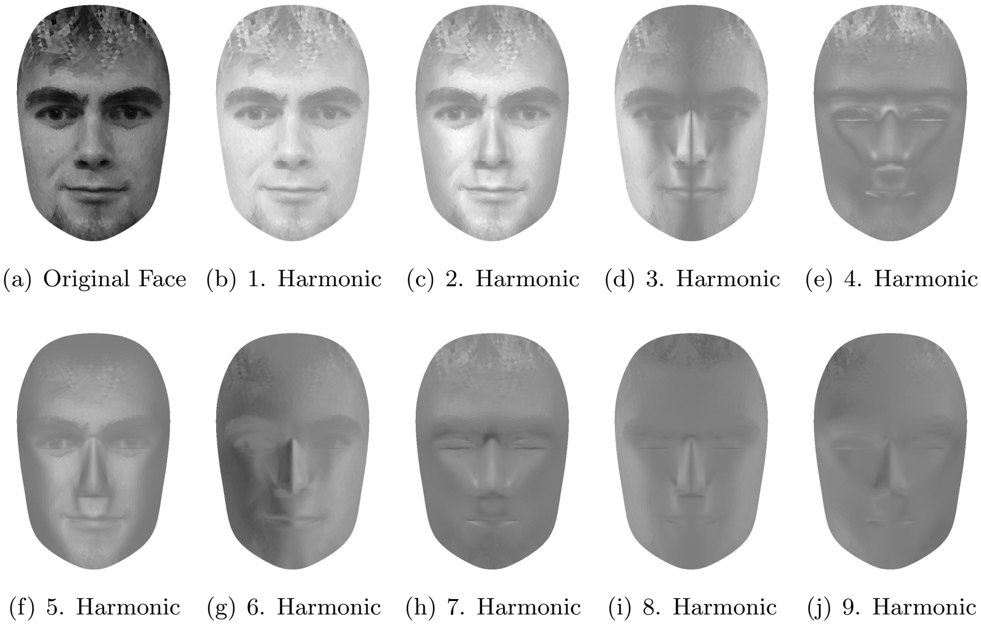
$$V_1(\eta,\rho) = \rho\ \frac{\pi}{\sqrt{4\pi}} \qquad V_2(\eta,\rho) = \rho\frac{2\pi}{3}\sqrt{\frac{3}{4\pi}}z \qquad V_3(\eta,\rho) = \rho\frac{2\pi}{3}\sqrt{\frac{3}{4\pi}}x \tag{3.6}$$

$$V_4(\eta,\rho) = \rho\frac{2\pi}{3}\sqrt{\frac{3}{4\pi}}y \qquad V_5(\eta,\rho) = \rho\frac{\pi}{8}\sqrt{\frac{5}{4\pi}}(3z^2-1) \qquad V_6(\eta,\rho) = \rho\frac{\pi 3}{4}\sqrt{\frac{5}{12\pi}}xz \tag{3.7}$$

$$V_7(\eta,\rho) = \rho\frac{\pi 3}{4}\sqrt{\frac{5}{12\pi}}xy \qquad V_8(\eta,\rho) = \rho\frac{3\pi}{8}\sqrt{\frac{5}{12\pi}}(x^2-y^2) \qquad V_9(\eta,\rho) = \rho\frac{3\pi}{4}\sqrt{\frac{5}{12\pi}}xy \tag{3.8}$$

where x, y and z denote the 3D coordinates of the surface normals η.

Given the 3D model of an object and its texture, i.e. albedo, Equation 3.5 is then used to generate the first nine harmonic images. An example of a textured 3D face mask and the first nine harmonic images is shown in Figure 3.5. The colour scheme of this figure ranges from -1 to +1; bright areas denote positive values whereas dark values denote negative values.



(a) Original Face (b) 1. Harmonic (c) 2. Harmonic (d) 3. Harmonic (e) 4. Harmonic

(f) 5. Harmonic (g) 6. Harmonic (h) 7. Harmonic (i) 8. Harmonic (j) 9. Harmonic

Figure 3.5: The original face mask and the first nine harmonic images. The colour map ranges from -1 (black) to +1 (white). The original image is taken from the IMM Face Database (Nordstrøm *et al.*, 2004)

These harmonic images can then be used to estimate the lighting conditions of a new image as shown in Figure 3.6. The 3D model, including its albedo, is shown in Figure 3.6(a) and the lighting conditions of the new image $\mathbf{J}_{new}$ in Figure 3.6(b) can be estimated as:

$$\min_{\beta} ||\mathbf{J}_{new} - \beta V(\eta, \mathbf{J})||_2 \tag{3.9}$$

where $\mathbf{J}$ is the original mask texture, $\mathbf{J}_{new}$ is a textured mask with new and unknown lighting conditions and V returns the harmonic images given the surface normals of the 3D model η and the albedo ρ. The result of the light estimation is shown in Figure 3.6(c).

3.1.3 Particle Filter Refinement

Once the 3D appearance model is built according to Section 3.1.1 and the lighting invariance is achieved as described in Section 3.1.2, both approaches are included in a particle filter based method to estimate the best fit of a deformable face mask to a previously unseen image.

As described in Section 2.2.3, particle filters are statistical models commonly used for

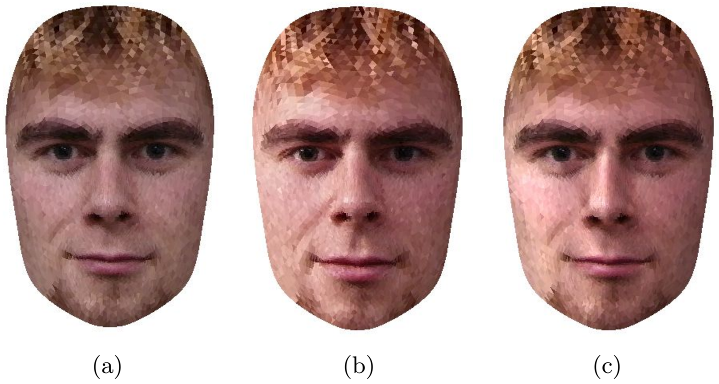


Figure 3.6: (a) Given a 3D model of a face and its texture, $\mathbf{J}$, (b) a new lighting condition, $\mathbf{J}_{new}$, can be estimated using harmonic images. (c) shows the result of this optimisation which equals to $\beta V(\eta, \mathbf{J})$.

tracking objects across several frames. Based on the state in the previous frame, Monte Carlo samples are drawn, evaluated and weighted in order to estimate the state in the current frame. In this section, however, particle filters are used to estimate the pose parameters T_{ext} and shape parameters γ of a deformable mask given a single image, so no tracking is performed.

Instead, Monte Carlo samples $x_t = \{T^{ext}, \gamma\}$ are drawn from a normal distribution $\mathcal{N}(x_t; x_{t-1}, \sum)$ in the neighbourhood of the previous state x_{t-1}. Assuming independence, $\sum$ is a diagonal covariance matrix with values set heuristically. The standard deviation of the three translation parameters is set to equal a shift of about one sixth of the face mask size and to about 15° for the three rotation parameters. The standard deviation for all shape parameters γ_k is set to 0.3.

Each sample $x_t^{(i)}$ is then evaluated by deforming the neutral face mask $\overline{\mathbf{g}}$ according to Equation 3.1 using the sampled shape parameters γ. The deformed face mask is then projected into the image by applying the sampled pose parameters T^{ext}, i.e. three translation and three rotation parameters, according to Equation 3.2. The resulting face image vector of concatenated grey values $\mathbf{J}$ is then used to calculate the distance in feature space, given the 3D appearance model as:

$$d(x_t) = \underset{\beta}{\operatorname{argmin}} ||\mathbf{J} - \beta V(\eta, \hat{\mathbf{J}})||_2 \tag{3.10}$$

where $\hat{\mathbf{J}}$ is calculated from the 3D appearance model according to Equation 3.3 and η are the surface normals of the deformed face mask $\mathbf{g}$. From this distance d, the weighting function for the particle filter is defined as a normalised vector as:

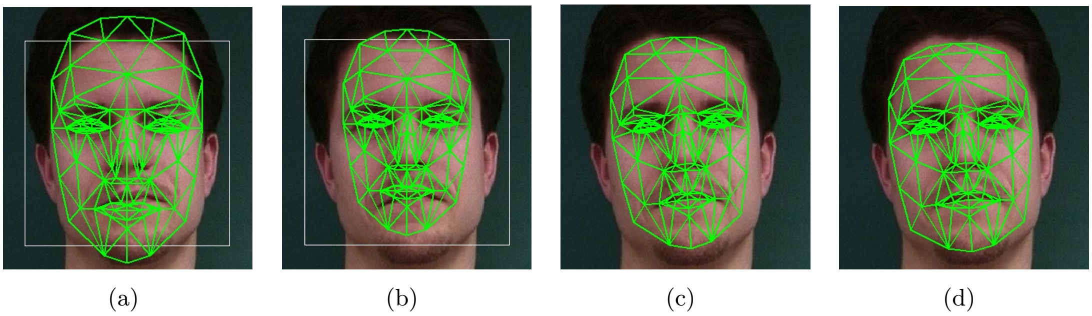
$$\widetilde{w}_t^{(i)} = [\upsilon - d(x_t^{(i)})]^{\lambda} \tag{3.11}$$

where $\upsilon = \max_i(d(x_t^{(i)}))$ and λ are annealing factors to increase the spread of the particle weights (Deutscher and Reid, 2005) and λ is empirically set to λ=4.

This particle filter based approach "iterates" in the *same* image for each time step t and thus performs an incremental refinement of the face mask pose and shape parameters, rather than tracking the face mask. Therefore, the genetic-algorithm like nature of the particle filter is utilised (Deutscher and Reid, 2005). Convergence is ensured by adjusting $\sum$ after each time step as:

$$\Sigma(t) = 0.8 \cdot \Sigma(t-1). \tag{3.12}$$

3.1.4 Automatic Fitting Algorithm



(a) (b) (c) (d)

Figure 3.7: (a) Result after face detection [white rectangle] and mask initialisation, (b) result after refined initialisation using grid-search, (c) final result of the proposed approach, (d) ground truth mask that was fitted to the labelled landmarks.

The proposed face mask fitting approach is based on the generated 3D appearance model (Section 3.1.1) and includes harmonic images to model lighting changes (Section 3.1.2). A particle filter based refinement (Section 3.1.3) is applied for optimisation. Figure 3.7 illustrates the proposed automatic face mask fitting approach. Given a previously unseen image of a face, the algorithm developed by Viola and Jones (2001) is used first to detect a near frontal face. The detected face coordinates are then used as a bounding box and the deformable face mask is initialised within this box. The neutral face mask is centred and aligned such that it fills the whole bounding box as shown in Figure 3.7(a).

Since the detected face coordinates are only a rough estimation, this first initialisation is further refined by a fast grid-search in order to improve the approximation of the z-part of the translation parameters $[x, y, z]$. The parameter z determines the distance of the face from the camera which also equals the size of the face mask. Therefore, z is assigned with 14 different values ranging from +5% to -5% from the initial z-value. A locally exhaustive and direct search as proposed by Dornaika and Ahlberg (2006) is used to solve for the best x and y values for each chosen z. The error function e, that guides this search is defined as the reconstruction error in feature space as:

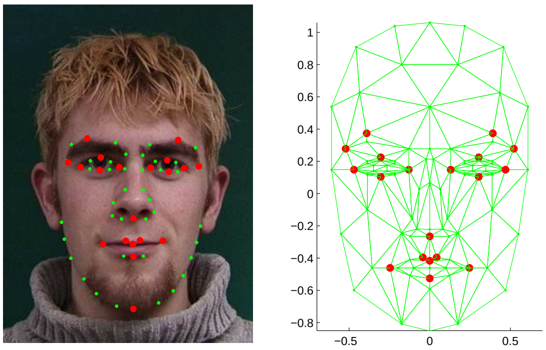
$$e(T^{ext}, \gamma) = ||\mathbf{J} - \hat{\mathbf{J}}||_2 \tag{3.13}$$

where $\mathbf{J}$ is the face image vector and $\hat{\mathbf{J}}$ is the face image vector reconstructed from the 3D appearance model according to Equation 3.3. The shape parameters γ as well as the rotation parameters within T^{ext} are set to zero and remain unchanged. The set of pose parameters (x, y, z) that results in the smallest error e is then used to initialise the particle filter. This results in a better mask fit around the eye and the nose area as shown in Figure 3.7(b). Since the person-specific shape parameters γ are kept constant in this step, the size of the mask will vary depending on the distance between the eyes, for example narrow eyes will result in a smaller mask.

The initialisation refinement is followed by the particle filter based fitting approach. In the first time step of the particle filter refinement, the shape parameters γ and the rotation parameters $[\phi_x, \phi_y, \phi_z]$ are set to zero. The next time step t uses the result of the previous step as initialisation. The particle filter then converges to the correct pose (T^{ext}) and person-specific shape parameters γ by repeatedly iterating on the same image, thus performing an incremental randomised search for the global maximum. The final mask fit after six iterations is shown in Figure 3.7(c). For comparison the manual fit is shown in Figure 3.7(d).

3.2 Experiments

The performance of the proposed mask fitting approach is tested on two databases, the IMM Face Database (Nordstrøm *et al.*, 2004) and the Extended Yale Face Database B (Georghiades *et al.*, 2001). Both databases are publicly available and details are described in the following subsections.



(a) (b)

Figure 3.8: (a) Sample image of the IMM face database with annotated facial landmarks, (b) CANDIDE-3 mask, red dots indicate point-to-point correspondences.

3.2.1 IMM Face Database

The IMM Face Database (Nordstrøm *et al.*, 2004) consists of 240 images of 40 individuals, 23 men and 7 women and six different images per individual. Those six images include variations in pose, lighting and expression. The resolution of each image is 640×480 pixels. Additionally, each image of the database is labelled with a set of 58 different landmarks, depicting facial feature points. An example image with annotated landmarks is shown in Figure 3.8(a).

These pre-labelled feature points are the main reason for choosing this dataset as they can be used for building the ground truth for the proposed mask fitting algorithm. For calculating the ground truth, point-to-point correspondences between the landmarks and the corresponding CANDIDE-3 mask vertices are assigned. The big red dots in Figure 3.8(a) and Figure 3.8(b) depict these correspondences.

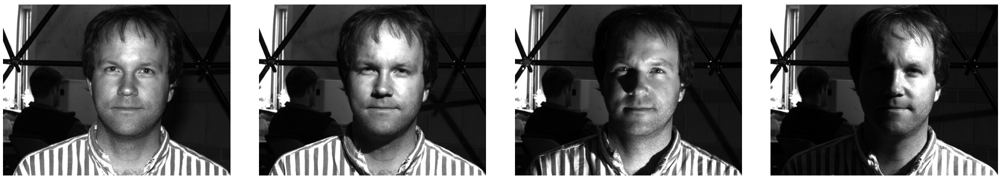
The CANDIDE-3 mask is then automatically fitted to the images of the IMM Face Database by minimising the Euclidean distance between the landmark points and corresponding mask vertices as:

$$\min_{T^{ext},\gamma} \sum_{l=1}^{L} \mathcal{P}(\mathbf{g}, T, l) - p_l \tag{3.14}$$

where $\mathcal{P}$ projects the l^{th} mask vertex into the image using $T = [T^{int}, T^{ext}]$ and p_l are the 2D coordinates of the l^{th} facial landmark. Levenberg-Marquardt is then utilised to estimate the pose parameters T^{ext} and shape parameters γ of the deformable face mask. Images with facial expressions are left out.

3.2.2 Yale Face Database

The subset of the Yale Face Database B (Georghiades *et al.*, 2001) that is used in the experiment section consists of 10 individuals under 9 different pose and 45 different lighting conditions. The Extended Yale Face Database B contains additional 28 individuals under the same pose and lighting conditions. Example images of each pose are shown in Figure 3.10. The combined set of 38 individuals is used in the experiments. The images of each person under different lighting conditions are further divided into four subsets according to Georghiades *et al.* (2001).



(a) Subset 1 up to 12° (b) Subset 2 up to 25° (c) Subset 3 up to 50° (d) Subset 4 up to 77°

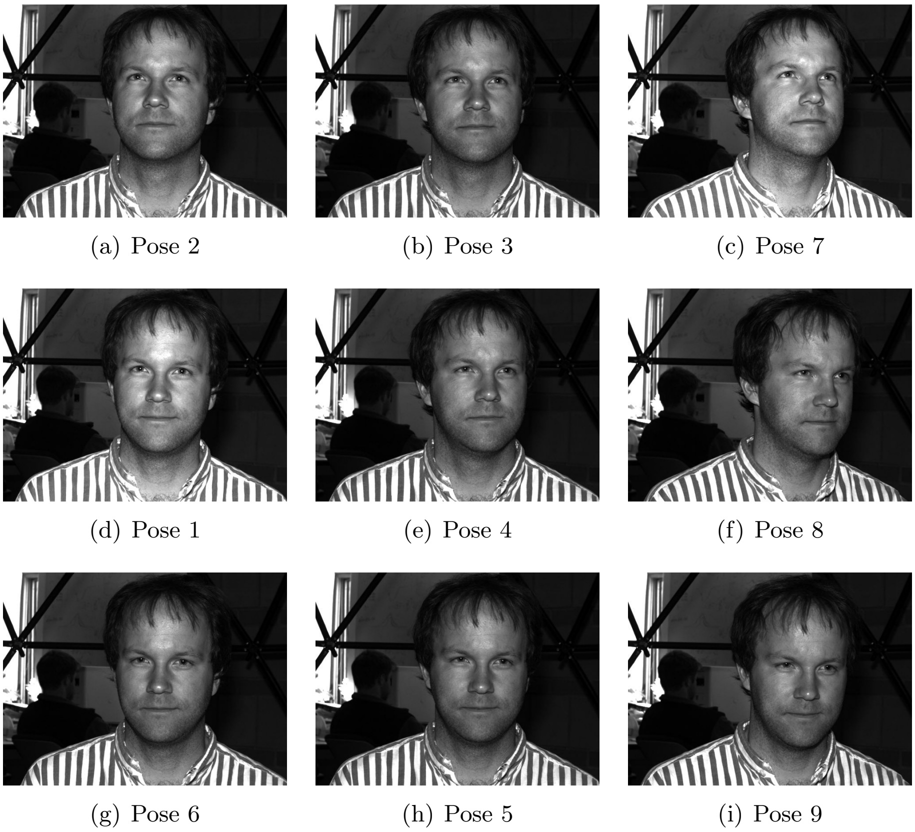
Figure 3.9: The Yale Face Database is divided into four subsets depending on the angle between between the camera axis and the light source. Figures (a)-(d) are example images of each subset.

The four subsets are created according to the angle between the light source and the camera axis. Subset 1 contains images with near frontal lighting (up to 12°), Subset 2 (up to 25°), Subset 3 (up to 50°) and Subset 4 contains the least illuminated images (up to 77°). An example image of each subset is shown in Figure 3.9. The ground truth for this dataset is acquired by deforming the CANDIDE-3 face mask manually to fit each individual. It is assumed that the pose and shape parameters of each individual remain constant across different lighting conditions.

3.2.3 The 3D Appearance Model

Prerequisite for the proposed mask fitting algorithm is a 3D appearance model, as explained in Section 3.1.1. This 3D appearance model is built from the first image of each individual of the IMM Face Database and is used throughout the entire experimental section, unless otherwise stated.

Using the point-to-point correspondences as shown in Figure 3.8, the CANDIDE-3 face mask is fitted automatically to the first image of each person in the IMM Face Database as explained in Section 3.2.1. The first image shows a full frontal face with neutral expression



(a) Pose 2

(b) Pose 3

(c) Pose 7

(d) Pose 1

(e) Pose 4

(f) Pose 8

(g) Pose 6

(h) Pose 5

(i) Pose 9

Figure 3.10: The Yale Face Database B contains face images under nine different poses, ranging from frontal (Pose 1) to looking up and far left (Pose 7) and looking down far left (Pose 9).

under diffuse lighting. Once the mask is fitted, the centre of each mask triangle is then projected into the image according to Equation 3.2 and the image vector **J** is extracted for each of the 40 individuals. PCA is then used to calculate the 3D appearance model, i.e. the mean face and a set of Eigenfaces from these image vectors as described in Section 3.1.1. The top 70% of all Eigenfaces are kept, using the Matlab code provided by Cai (2007).

The 3D appearance model is created for each of the four mask sizes. The mean face $\overline{\mathbf{x}}$ for each mask size is shown earlier in Figure 3.4 and the first seven Eigenfaces for the finest mask 3 are shown in Figure 3.11.

3.2.4 Fitting Performance

The following subsections describe the different experiments on the two face databases, namely the IMM Face Database (Section 3.2.1) and the Extended Yale Face Database B

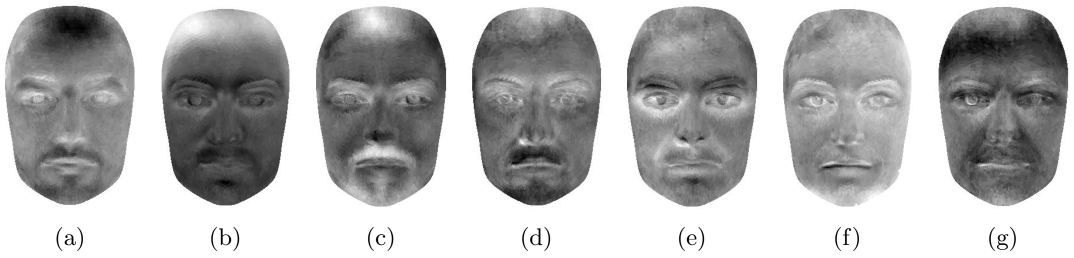


Figure 3.11: The first seven Eigenfaces generated from the first image of each of the 40 individuals in the IMM Face Database. Figure (a)-(g) show Eigenfaces 1 to 7 respectively.

(Section 3.2.2) and discuss their results. In Section 3.2.4.1 the performance of the proposed fitting approach with respect to accuracy and speed is evaluated. Therefore, the number of particles is varied to find the best trade off between fitting accuracy and run time. The IMM Face Database is used for the experiments in Section 3.2.4.2 and Section 3.2.4.3. Section 3.2.4.2 evaluates the fitting performance of the proposed approach on images of different resolutions as well as under different lighting conditions. In Section 3.2.4.2 these results are compared with another fitting method based on Active Shape Models. Lastly, Section 3.2.4.4 uses the Extended Yale Face Database B to test the proposed approach on images under different pose and lighting conditions.

3.2.4.1 Accuracy versus Speed

The proposed automatic mask fitting approach uses a particle filter to estimate the pose parameters T^{ext} and the person-specific shape parameters γ. The performance in terms of accuracy and speed is directly dependant on the number of particles used. The more particles the higher the achieved accuracy, since more of the parameter space is likely to be explored or the higher the chance of finding the global maximum. However, an increased number of particles will also increase the computation time.

For the following experiment the first image of each of the 40 individuals in the IMM dataset is used. These are the same images that were used to build the 3D appearance model in the previous Section 3.2.3. Using this 3D appearance model the proposed deformable mask fitting algorithm as described in Section 3.1.4 is employed to automatically fit the CANDIDE-3 mask to each of the 40 images, estimating the pose parameters T^{ext} and the person-specific shape parameters γ. This is also called testing on training data since the testing images are also used for building the model.

The testing on the training data experiment is used and the number of particles is varied from 100 to 50,000. The resulting mean vertex difference is used for comparison and the time needed to evaluate the number of particles is measured on an Intel Core 2 Quad 2.33GHz PC, using only a single core.

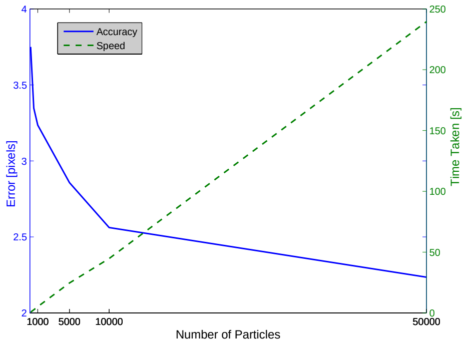


Figure 3.12: The number of particles and the resulting mean vertex point difference (accuracy) versus the run time (speed).

The diagram in Figure 3.12 shows the result of the performance test. The evaluation of 100 particles takes less then 1s but the result is the least accurate with a mean vertex difference of 3.76 pixels. The evaluation run time increases linearly with the number of particles, whereas the mean vertex difference drops exponentially. The number of particles closest to the intersection of these two graphs is chosen as the best trade off between accuracy and speed. Thus, 10000 particles are chosen for all further experiments since this maximises the accuracy given a limited computation time.

3.2.4.2 Low Resolution Fitting

Within this experiment, different mask sizes and different image resolutions are used. The original image resolution of 640×480 pixels is halved three times resulting in images of size 320×240, 160×120 and 80×60 pixels with average face sizes of 250×160, 125×80, 60×40 and 30×20 pixels respectively. Also, each of the four different mask sizes as shown earlier in Figure 3.2 are used for automatic fitting.

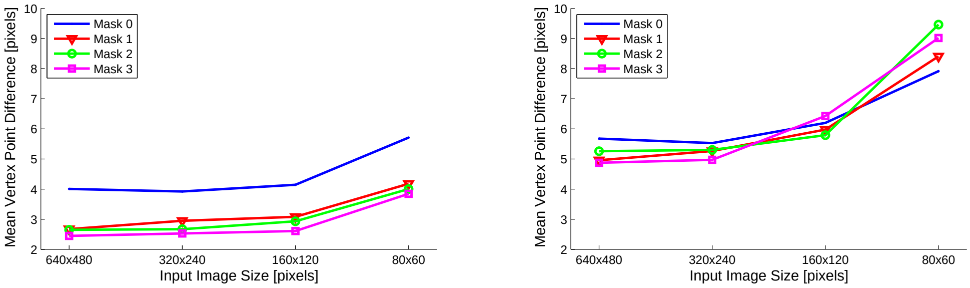
In a second experiment, the CANDIDE-3 mask is automatically fitted to the fifth image of each individual in the IMM Face Database. These images are similar to the first image in that they show full frontal faces with neutral expressions, but instead of diffuse lighting a spot light is added to illuminate the person's left side. Again, the original image size is halved three times and all four different mask sizes are used for automatic fitting.

To evaluate the results of the automatic fitting experiments in this section, the mean Euclidean distance between the estimated vertex points and the ground truth vertex points is calculated and the result is shown in Figure 3.13. Each result is averaged over 40 images, one image for each individual. Both graphs show that the original CANDIDE-3 mask 0 is too coarse, resulting in the least accurate fit. The main reason for this is the applied texture mapping technique assigning only a single grey value to each mask vertex, and as shown in Figure 3.14 this is not detailed enough to precisely represent different faces.

The fitting accuracy increases with the increasing number of mask vertices. The finer the mask the more detailed the texture and the more accurate the fitting. However, an increasing number of mask vertices will result in an increase in run time as the resolution of the mask texture increases. The mask 2 hereby is found as the best trade off between accuracy and speed. The increase in accuracy of mask 3 is minimal compared to the increase in run time and does therefore not justify its use.

Furthermore, the fitting accuracy decreases with decreasing image resolution as shown in Figure 3.13. For easier comparison of the results, the mean vertex difference is calculated with respect to an image size of 640×480 pixels, since the mean vertex difference depends on the initial image resolution. For images of size 320×240 pixels and below, the estimated pose and shape parameters are used to project the mask into the image of size 640×480 pixels.

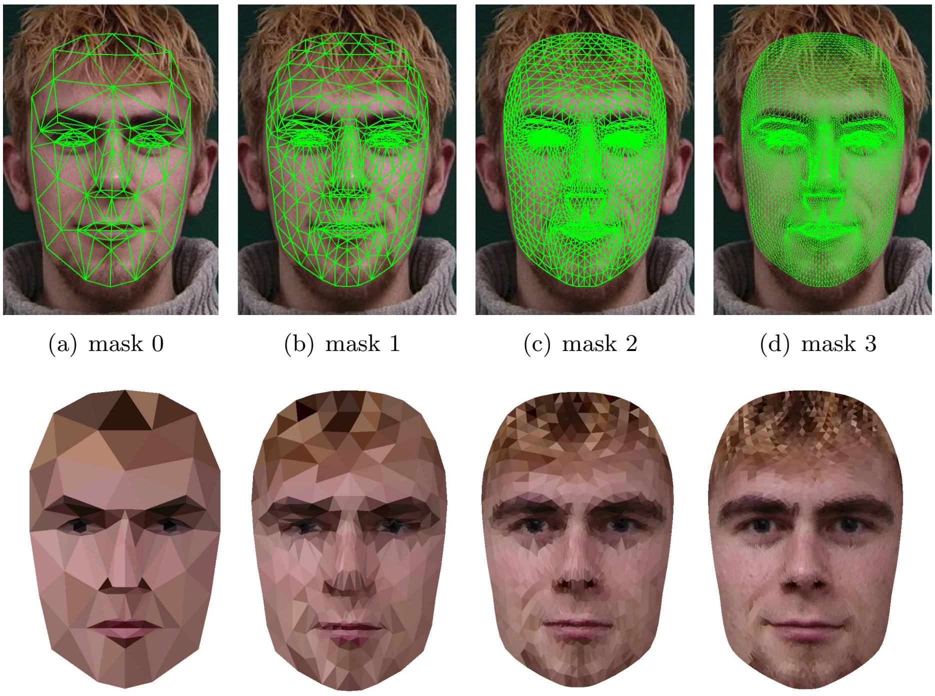
Both graphs in Figure 3.13 show clearly that the fitting accuracy depends on the image resolution as well as on the mask size used. The coarser the mask, the higher the fitting error. The same is true for the image resolution; i.e. the smaller the resolution, the least accurate the fit.



(a)

(b)

Figure 3.13: Mean vertex point difference for different input resolutions and different mask sizes used for (a) testing on training and (b) testing in a different lighting condition.



(a) mask 0 (b) mask 1 (c) mask 2 (d) mask 3

(e) mask 0 (f) mask 1 (g) mask 2 (h) mask 3

Figure 3.14: Different masks sizes fitted to an image (top row) and the resulting mask texture (bottom row). The finer the mask the higher the resolution of the mask texture.

Figure 3.15 shows the result of the testing on training data for different image resolutions of the same person. Using mask 2 for fitting, the mean vertex differences are 2.67, 2.95, 3.08 and 4.18 pixels for image resolutions of 640×480, 320×240, 160×120 and 80×60 pixels respectively. Note that the 3D appearance model is built from images of resolution 640×480 pixels, but by using different mask sizes and by applying the described texture mapping technique, the same mask can also be applied to any image resolution smaller than 640×480 pixels.

In comparison, Figure 3.16 shows the result of the automatic mask fitting approach tested on the fifth image of the IMM Face Database using mask size 0. These images are taken under different lighting conditions compared to the images used for creating the 3D appearance model. Again, mask 0 is too coarse and achieves the least accurate fitting results. The best results are achieved for images of resolution 640×480 and 320×240 pixels. Different lighting conditions are particularly difficult to estimate at lower resolutions, which is shown by the decrease in accuracy in Figure 3.13(b). On average, the mean vertex difference increases by two pixels when fitting the deformable mask to images of lighting conditions that differ from the training set.

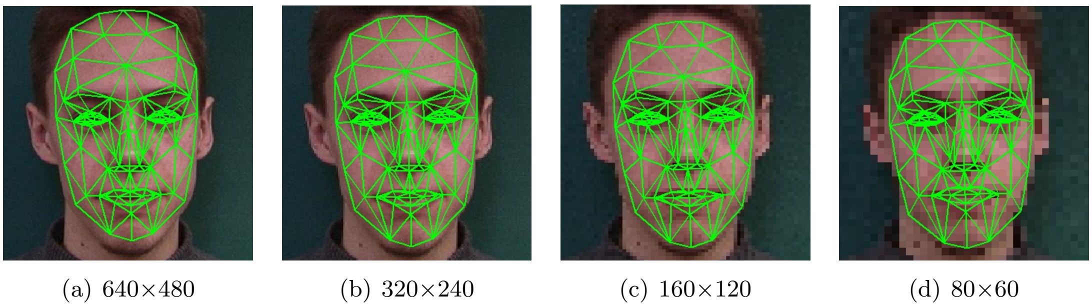

(a) 640×480 (b) 320×240 (c) 160×120 (d) 80×60

Figure 3.15: Example result of the testing on training for different image resolutions.

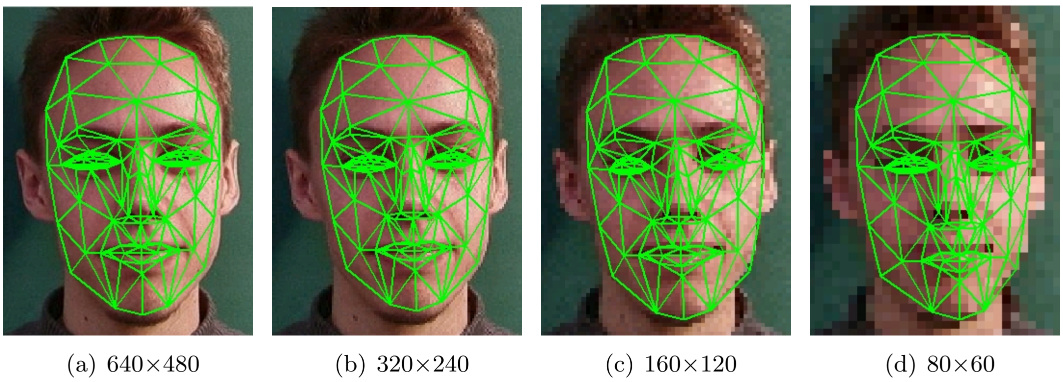

(a) 640×480 (b) 320×240 (c) 160×120 (d) 80×60

Figure 3.16: Example result of the testing in different lighting conditions for different image resolutions.

Furthermore the accuracy of the person-specific shape parameters γ varies for each of the seven different parameters listed in Table 3.1. Shape parameters that control the position of the eyes, nose or the mouth are easier to estimate and result in higher fitting accuracies than shape parameters that only control a small number of triangles, like *'Eyebrows vertical position'*, *'Eyes width'* or *'Mouth width'*. Their accuracy drops, especially in low resolution images where slight parameter variations may not effect the fitting error.

3.2.4.3 Comparison with Active Shape Models

Active Shape Models (ASM) are commonly used for detecting facial feature points as described in Section 2.1.1.1. A pre-trained ASM model is fitted to a new image by matching the position of landmark points along the contour lines of a pre-defined shape. Unlike the proposed fitting approach, only the image area in direct circumference of the contour lines of the object is used within the optimisation.

In the following experiment, the proposed mask fitting approach is compared with the method proposed by Lu *et al.* (2001). Their approach automatically fits a deformable 3D face mask to an image using an ASM. First a set of significant facial feature points is detected using the ASM approach and then the 3D mask model is fitted to these points. This two step approach differs from the proposed algorithm that directly fits a deformable face mask to an image without detecting facial feature points first.

For comparing both approaches, the ASM implementation of Hamarneh (2008) is used for training the ASM with the 58 landmarks of the first image of each individual of the IMM Face Database. Again the face detector (Viola and Jones, 2001) is used to locate the face within the image and the ASM is initialised to fill the detected face bounding box. The pre-trained ASM face model is then used to detect the 58 feature points of the face. In the final step, the CANDIDE-3 mask is fitted to these landmarks by minimising the Euclidean distance between the landmarks and the corresponding mask vertices as described in Section 3.2.1.

This method is compared with the proposed fitting approach by performing the following experiments on the IMM Face Database:

1. Testing on training data,
2. Leave one out test,
3. Testing with lighting variation.

The testing on the training data experiment from the previous section is repeated by automatically fitting the mask to all 40 images that are also used to build the appearance model. The second experiment, using the leave one out test method, is set up to test the generalisation of both approaches. Therefore, the 3D appearance model as well as the ASM face model is built from only 39 of the first 40 images of the IMM Face Database. This model is then used to automatically fit the face mask to the image that was left out in the model building process. For the last experiment, the fifth image of the IMM Face Database is used for fitting to examine the robustness against lighting changes. The 3D appearance model and the ASM face model are again built from the first image of each individual.

The results of all three experiments are summarised in Table 3.2. They show that the proposed approach achieves consistently better results. The ASM requires a large set of parameters for initialisation and is also very sensitive to the chosen set of parameters.

	Proposed Approach	**ASM Approach**
(1) Testing on training	2.6 pixels	6.2 pixels (26/40)
(2) Leave one out test	4.2 pixels	6.4 pixels (26/40)
(3) Lighting variation	4.8 pixels	8.1 pixels (9/40)

Table 3.2: The fitting results of the proposed approach compared to the ASM based approach on the IMM Face Database.

Choosing a fixed set of initialisation parameters, the ASM was only able to detect the facial feature points of 26 individuals out of 40 individuals in the testing on training data test. In 14 cases, the ASM drifted completely off the face region in images of size 640×320 pixels. The mean vertex difference amounts to 6.2 pixels for 26 correctly detected individuals, compared to 2.6 pixels achieved by the proposed approach.

Furthermore, the ASM failed when trying to detect facial landmark points in images of resolution 320×240 pixels and lower. This is because the ASM was trained on images of resolution 640×480 pixels and will fail on any image that differs greatly from this resolution. This is contrary to the proposed approach that uses different mask resolutions and assigns a single colour value to each vertex. Using this technique the deformable face mask can be fitted to any image resolution that is the same or smaller compared to the training images.

Therefore, different ASM face models are created from images of resolution 320×240 pixels, 160×120 pixels and 80×60 pixels in addition to the ASM face model build from images of resolution 640×480 pixels. These models are then used to recover the facial features in images of corresponding resolutions. Again the CANDIDE-3 mask is fitted to these points and the mean vertex point difference with respect to the ground truth is calculated. Similar to the previous experiment the mean vertex error is calculated with respect to an image resolution of 640×480 pixels for better comparison. The result is shown in Figure 3.17.

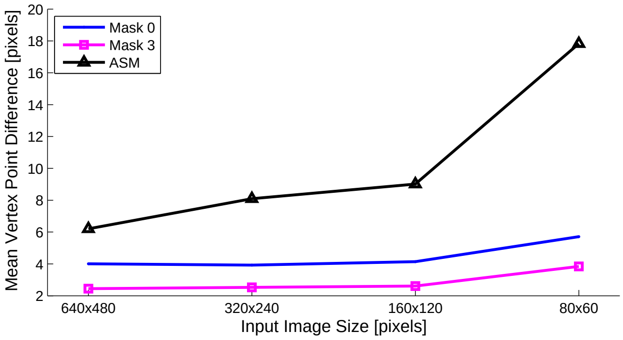
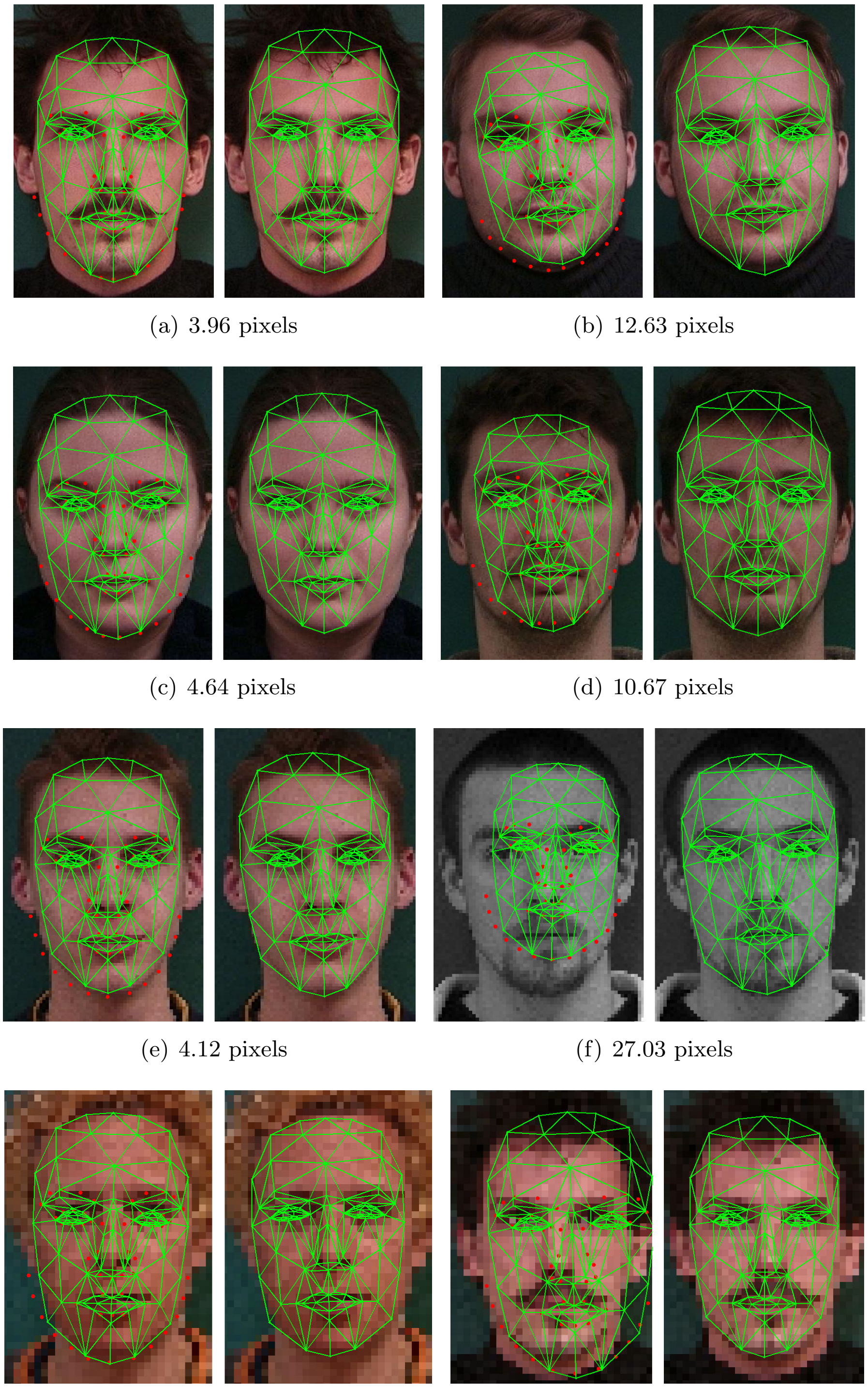


Figure 3.17: ASM fitting results for different image resolutions



(a) 3.96 pixels

(b) 12.63 pixels

(c) 4.64 pixels

(d) 10.67 pixels

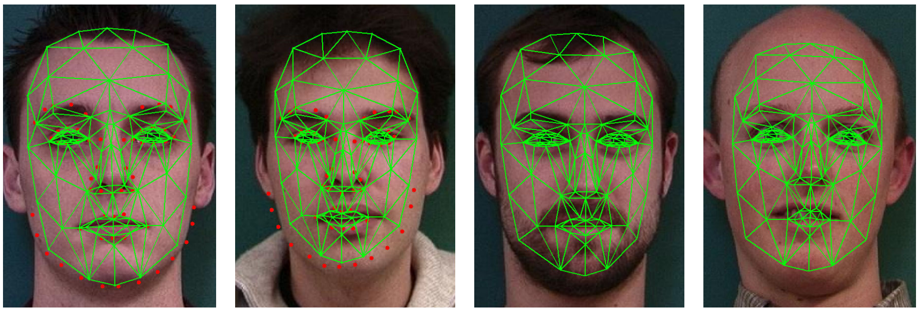
(e) 4.12 pixels

(f) 27.03 pixels

(g) 7.56 pixels

(h) 25.85 pixels

Figure 3.18: Fitting results of the ASM based approach in the testing on training data for images of resolution 640×480 pixels (top row), 320×240 pixels (second row), 160×120 pixels (third row) and 80×60 pixels (bottom row). The caption of each figure states the mean vertex distance in pixels. The best fit with the lowest values (a),(c),(e),(g) and the worst fit corresponding to the highest value (b),(d),(f),(h) are shown for each resolution. The image on the right of each figure shows the ground truth.



(a) 3.55 pixels (b) 7.50 pixels (c) 2.29 pixels (d) 6.98 pixels

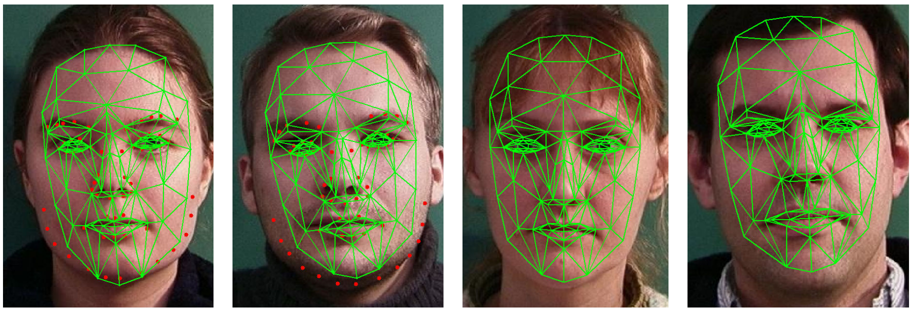
Figure 3.19: Fitting results of the ASM based approach (a),(b) and the proposed approach (c),(d) in the leave one out test. The caption of each figure states the mean vertex distance in pixels. The best fit with the lowest values (a),(c) and the worst fit corresponding to the highest value (b),(d) are shown.

Even though the training and testing image resolutions are the same, the ASM approach was only able to detect 37 faces in images of resolution 80×60 pixels, 40 faces in resolution 160×120 pixels, 17 faces in resolution 320×240 pixels and 26 faces in images of resolution 320×240 pixels, out of 40 images. The mean vertex error is only calculated for the detected faces and not for images where the ASM face model drifted completely off the face.

As shown in Figure 3.17, the fitting error differs between 2 and 12 pixels compared to the fitting result of the proposed approach using the coarsest mask 0. This increases to about 4 to 14 pixels when using mask 3. These fitting results for different image resolutions are visualised in Figure 3.18. The best fit with the lowest mean vertex difference and the worst fit with the highest error are shown for image resolutions of 640×480 pixels, 320×240 pixels, 160×120 pixels and 80×60 pixels.

The second experiment, leave one out test, aims to evaluate the generalisation of both fitting approaches. The mean vertex point difference across all 40 images increased to 6.4 pixels from 6.2 pixels for the ASM approach. Again only 26 out of 40 faces were detected. In comparison, the proposed approach results in a mean vertex point difference of 4.2 pixels across 40 images of size 640×480 pixels using a mask 2 and 10,000 particles. For an average face size of 250×160 pixels this equals an error increase of about 2% compared to the testing on training data result.

Figure 3.19 shows the best fit with the smallest mean vertex difference and the worst fit resulting in largest error for both fitting approaches. The proposed approach generalises very well with only a small error increase. The achieved mean vertex difference across 40 images is 2.2 pixels less compared to the ASM approach. However, the ASM approach



(a) 5.16 pixel (b) 12.14 pixel (c) 2.80 pixel (d) 8.12 pixel

Figure 3.20: Fitting results of the ASM based approach in the lighting variation test. The caption of each figure states the mean vertex distance in pixel. The best fit with the lowest values (left) and the worst fit corresponding to the highest value (right) are shown.

optimises the position of the facial feature points based on intensity values in the local circumference as explained in Section 2.1.1.1 and it is therefore less affected by the leave one out test. In comparison, the proposed approach uses the entire appearance of the face and, thus depends on the pre-trained 3D appearance model. A large and diverse training set will result in a 3D appearance model that generalises well.

The last experiment uses the fifth image of the IMM Face Database for fitting in order to test the effect of different lighting conditions. Since the ASM approach does not allow for lighting changes the face was only detected in 9 out of 40 images. The mean vertex difference for these 9 images amounts to 8.1 pixels, with the best and the worst fit shown in Figure 3.20(a) and Figure 3.20(b) respectively.

In comparison, the proposed approach uses harmonic images to model different lighting conditions and is therefore superior to the ASM approach in this experiment. The resulting mean vertex difference across 40 images amounts to 4.8 pixels which is 41% less compared to the ASM approach. Again the best and the worst fit of the proposed fitting approaches are shown in Figures 3.20(c) and 3.20(d) respectively.

3.2.4.4 Different Pose and Lighting Conditions

The experiments in this section are aimed at testing the proposed approach on a large number of different lighting conditions as well as under different poses. Therefore, a subset of the Extended Yale Face Database is used, which contains images of nine different pose and 45 different lighting conditions of 38 different individuals divided into four subsets according to Section 3.2.2. For the first experiment the 3D appearance model is built from the IMM Face Database as described in Section 3.2.3. The second experiment then uses a 3D appearance model that also includes one image for each of the individuals in the Yale Face Database B under frontal lighting, i.e. zero degrees azimuth and elevation.

The first experiment uses the proposed mask fitting approach to automatically fit the deformable face mask to the first subset, containing 266 images of size 640×480 pixels, seven images for each of the 38 individuals. The mean vertex point difference across all 266 images amounts to 6.5 pixels. This equals a mean fitting error of 3.3% with respect to the face size. Note that none of the 38 individuals are included in the training set used for building the 3D appearance model.

Furthermore the proposed approach is used to automatically fit the face mask to all 456 images of the second subset, 12 for each individual. The images are 640×480 pixels in size and the proposed approach was able to fit the mask correctly to 433 out of 456 (95%) using mask 2 and 10,000 particles. In 23 cases, the lighting condition could not correctly be modelled resulting in false particle weights (Equation 3.11). The mean vertex point difference across all 433 correctly fitted images amounts to 6.2 pixels, which is similar to the first subset. These results worsen with decreasing image resolution and more extreme lighting conditions.

The images in subset 3 and 4 of the Yale Face Database exhibit extreme lighting conditions as shown earlier in Figure 3.9. However the CANDIDE-3 face mask is only able to model a limited number of person-specific facial deformations (Table 3.1). Since the harmonic images, used for modelling the illumination depend on the exact person-specific face shape, the proposed approach using the CANDIDE-3 face mask, will fail when trying to fit the mask to these images. The deformability of the face mask is not sufficient to model extreme lighting conditions.

The second experiment uses a subset of the Yale Face Database B containing a total of 4050 images of 10 individuals under nine different pose and 45 different illumination conditions. Again, the proposed approach is used to automatically fit the mask to each of 4050 images of size 640×480 pixels using mask 2 and 10,000 particles. However, one image of each individual of the Yale Face Database is also used to build the 3D appearance model.

The mean vertex difference in pixels for each pose and each subset containing different illumination conditions are shown in Table 3.3. Additionally, Figure 3.21 shows example images of the results for different persons under different pose and illumination conditions. Similar to the first experiment, Subset 1 and 2 achieve best results across all poses.

Pose	Subset 1	Subset 2	Subset 3	Subset 4
1 - frontal	6.48	7.70	25.08	59.81
2 - up	10.52	10.80	30.15	53.77
3 - up left	10.06	10.72	25.67	56.29
4 - left	8.48	10.05	22.47	66.27
5 - down left	6.09	6.87	16.22	51.06
6 - down	5.50	6.06	17.22	52.75
7 - up far left	14.46	19.31	37.27	61.13
8 - far left	11.06	13.25	39.39	63.30
9 - down far left	6.95	7.36	15.97	54.69

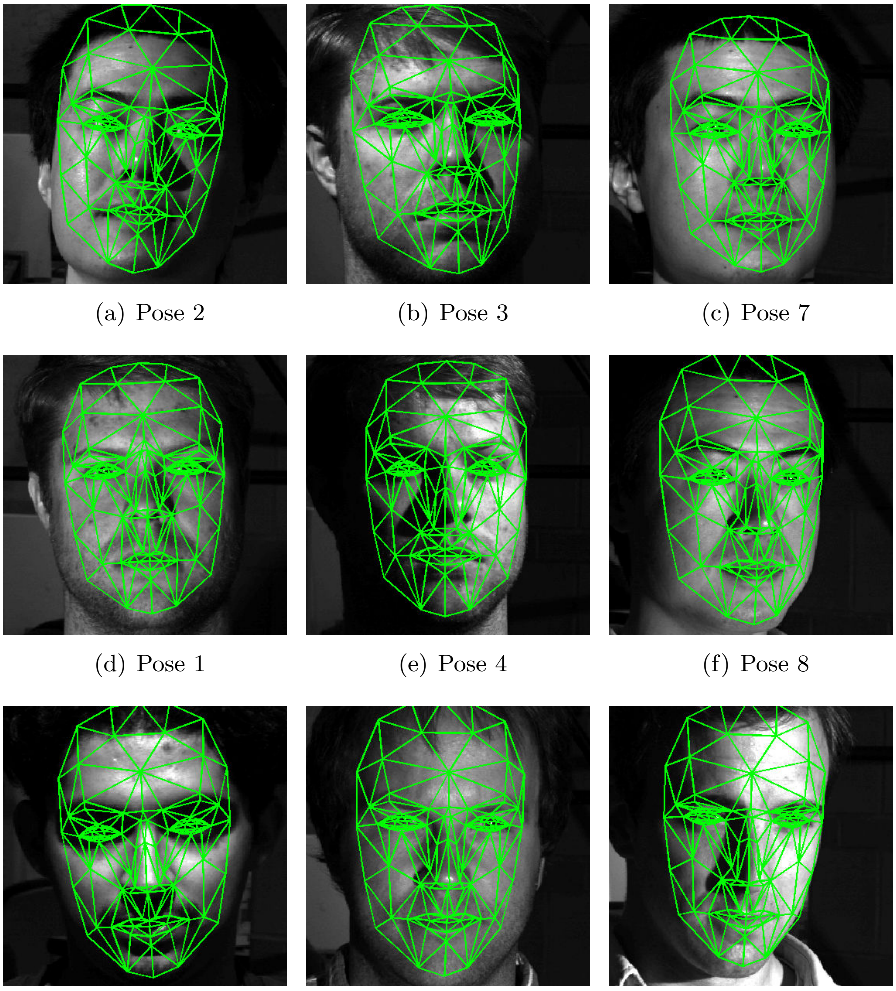
Table 3.3: The result of the proposed fitting approach for 10 individuals under nine different poses and for different lighting conditions in Subset 1 (seven images), Subset 2 (twelve images), Subset 3 (twelve images) and Subset 4 (fourteen images). A total of 4050 images are used. The results for each pose and subset show the mean vertex difference in pixels.

However in the second experiment the mask is fitted to all images contained in Subset 1 and 2 due to a larger training set that contains one image for each individual of the Yale Face Database. However, the CANDIDE-3 mask is still unable to model the illumination conditions of Subset 3 and 4 sufficiently, resulting in large fitting errors.

The quality of the mask fit varies across different poses for each subset, but in general, the best fitting results are achieved by near frontal poses. The farther left the face is turned, the worse the mask fit. There is also a great difference between poses that 'look up' compared to poses that 'look down'. The mean vertex error for Pose 5 (down left), 6 (down) and 9 (down far left) is up to 50% smaller in comparison to Pose 2 (up), 3 (up left) and 7 (up far left) for all subsets. The appearance of faces that 'look down' changes only marginally compared to the frontal pose. However, when 'looking up', the nostrils become more visible as well as a larger part of the eyelids, for example. These larger changes in appearance decreases the fitting performance because only frontal images are used to build the 3D appearance model.

3.2.5 Experiments Summary

The results on the IMM Face Database show that the proposed approach is able to correctly and fully automatically recover the pose parameters T_{ext} and the person-specific face shape parameters γ for different persons. The leave one out experiment showed its generalisation ability and the experiments with lighting variation show that the proposed approach works well under unknown lighting conditions.



(a) Pose 2 (b) Pose 3 (c) Pose 7

(d) Pose 1 (e) Pose 4 (f) Pose 8

(g) Pose 6 (h) Pose 5 (i) Pose 9

Figure 3.21: Fitting results for different persons of the Yale Face Database under different pose and different lighting conditions.

The experiments on the Yale Face Database B show the performance of the proposed 3D face mask fitting approach for different pose and illumination conditions. The 3D appearance model for the first experiment is built entirely from the IMM Face Database, thus these results confirm the generalisation ability of the proposed approach. Furthermore, this data set contains 45 different lighting conditions of which about 40% are suitable for fitting using the CANDIDE-3 mask. Extreme lighting conditions, such as in Subset 3 and 4 of the Yale Face Database, cannot be modelled acurately with this face mask.

3.3 Conclusion

This chapter proposes a new method for fitting a deformable 3D face mask to a single image, detecting person-specific facial features as well as the overall 3D pose of the face is performed simultaneously. The CANDIDE-3 mask is used for automatic fitting and light-invariance is achieved through spherical harmonic images that are created directly from the 3D face mask. Using a set of training images, Principal Component Analysis (PCA) is applied for creating a 3D appearance model. A particle filter based approach then includes the harmonic images as well as the 3D appearance model into an error function to estimate the best shape and pose parameters for a given image of a face.

The CANDIDE-3 face mesh is subdivided, resulting in a number of different mask mesh resolutions, ranging from coarse to fine. By applying a new texture mapping function the same mask is used to fit images of high and low resolution. Using a certain image resolution for training, the resulting 3D appearance model is suitable for the same size testing images as well as any input image of smaller resolution, unlike Active Shape Models (ASM) methods.

Experiments on the IMM Face Database confirm the suitability for different image resolutions and different lighting conditions. Generally the fitting accuracy increases with increasing image resolution as well as with increasing mask resolution. The finer the mask, i.e. the more triangles, the better the fit.

Further experiments on the IMM Face Database Nordstrøm *et al.* (2004) and the Yale Database B Georghiades *et al.* (2001), both publicly available, showed that the proposed method generalises well and is suitable for a large variety of different illuminations. However, the CANDIDE-3 face mask, which is used for all experiments, allows only for non-extreme lighting conditions, such as Subset 1 and 2 of the Yale Face Database B. Additionally, the Yale Face Database B is used to evaluate the fitting performance under different pose. The experiments show that faces that 'look down' result in smaller fitting errors compared to faces that 'look up', when the 3D appearance model is built from frontal images only.

Summarising, the proposed approach is best suited for detecting pose and person-specific shape parameters of a previously unseen image under unknown non-extreme lighting conditions. The position of facial features is estimated and the mask is fitted for successive applications like object tracking in the following Chapter 4 or face recognition in Chapter 6.

Chapter 4

Combined Tracking and Super Resolution

Wide area surveillance tasks require high resolution images of the object of interest while only acquiring low resolution video of the scene. This chapter proposes a new method for combining model-based tracking and super resolution in the context of large scale surveillance. The key idea is the use of a deformable 3D object model for both tracking and super resolution. Unlike most existing super resolution techniques, the proposed method increases only the resolution of the object rather than the entire scene without using interpolation techniques.

A common super resolution algorithm is the super resolution optical flow (Baker and Kanade, 1999). This method interpolates each frame to twice its size and optical flow is used to register previous and consecutive frames, which are then warped into a reference coordinate system. The super-resolved image is calculated as the average across these warped frames. However, the first step of interpolation introduces artificial random noise which is difficult to remove. Secondly, the optical flow is calculated between previous and consecutive frames preventing its use for an online stream processing algorithm. Also, accurate image registration requires precise motion estimation (Barreto *et al.*, 2005), which in turn affects the quality of the super-resolved image (Zhao and Sawhney, 2002). Most optical flow methods fail in low textured areas and cannot be used to register non-planar and non-rigid objects. A recent technique proposed by Gautama and van Hulle (2002) calculates sub-pixel optical flow between several consecutive frames (with non-planar and non-rigid moving objects), however it is unable to estimate an accurate dense flow field, which is needed for accurate image warping.

Solving all the issues in a general case is difficult, as the general problem of super resolution is numerically ill-posed and computationally complex (Farsiu *et al.*, 2004). A specific issue is addressed here: Simultaneous tracking and increased super resolution of known object types (i.e. faces, license plates, etc.) acquired by low resolution video. The use of an object-specific 3D mesh overcomes the issues with optic flow failures in low textured images. Interpolation is avoided and a 3D mesh is used to track, register and warp the object of interest. Using the 3D object mask to estimate translation and rotation parameters between two frames is equivalent to calculating a dense sub-pixel accurate optical flow field and subsequent warping into a reference coordinate system. The 3D mesh is subdivided, such that each triangle is smaller than a pixel when projected into the image, which makes super resolution possible (Smelyanskiy *et al.*, 2000) and allows for sub-pixel accurate image registration and warping. In addition, such a fine mesh improves the tracking performance of low-resolution objects. Each triangle then accumulates the average colour values across several registered images and a high resolution 3D model is created online during tracking. This approach differs from classical super resolution techniques as the resolution is increased at the model level rather than at the image level. Furthermore, only the object of interest is tracked and super-resolved rather than the entire scene, which reduces computation costs. Lastly, the use of a deformable mask mesh allows for tracking of non-rigid objects, like human faces.

This chapter is organised as follows: The image formation process is described in Section 4.1 and based on this the proposed method is outlined in Section 4.2. Next, the 3D tracking approach and the model based super resolution method are introduced in Sections 4.3 and 4.4 respectively. An extension to non-planar and non-rigid objects is proposed in Section 4.5. The experimental evaluations of the combined tracking and super resolution method are demonstrated in Section 4.6.

4.1 The Image Formation Process

The image formation process is important for understanding the necessity for super resolution. When taking a picture with a digital camera, the resulting image is captured from the (high resolution) 3D world and projected onto the CCD chip - the image plane. During this process the high resolution image I^{high} is sub-sampled, warped and blurred resulting in a degraded low resolution image I^{low}:

$$I^{low} = AI^{high} + h \tag{4.1}$$

where I^{high} and I^{low} are the high and low-resolution images respectively. The degrading matrix A represents image warp, blur and image sampling; h models the uncertainties due to noise.

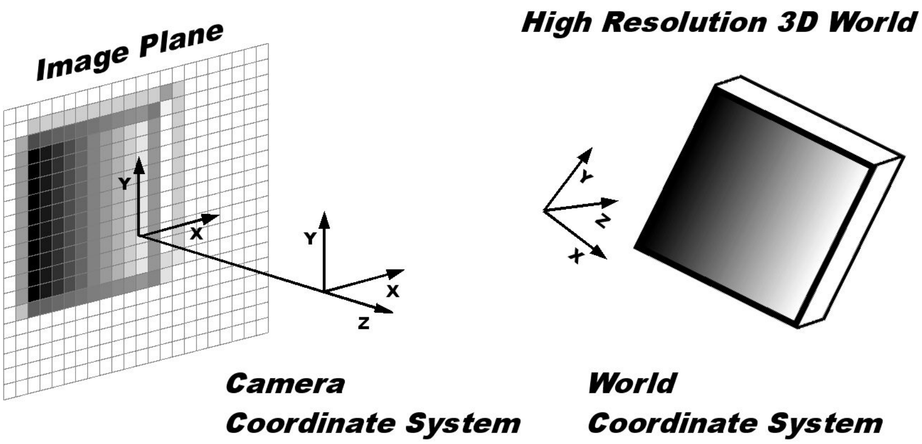


Figure 4.1: The basic principles of the image formation process: The high resolution 3D object is warped from the world coordinate system into the camera coordinate system and is finally projected onto the image plane. After this process the resulting low resolution image is warped, blurred and sub-sampled.

This image formation process (Faugera, 1993) is illustrated in Figure 4.1. The 3D object is warped from the world coordinate system into the camera coordinate system and then projected into the 2D image plane. The resulting image is sub-sampled as an effect of the finite number of pixels on the imaging chip. Furthermore this image is degraded by blurring, which is caused by the optical system of the camera, motion and additional random noise. This image formation process can be described by Equation 4.1, where the degrading matrix A is used to model all possible degradations. However, the matrix A is unknown and hard to estimate.

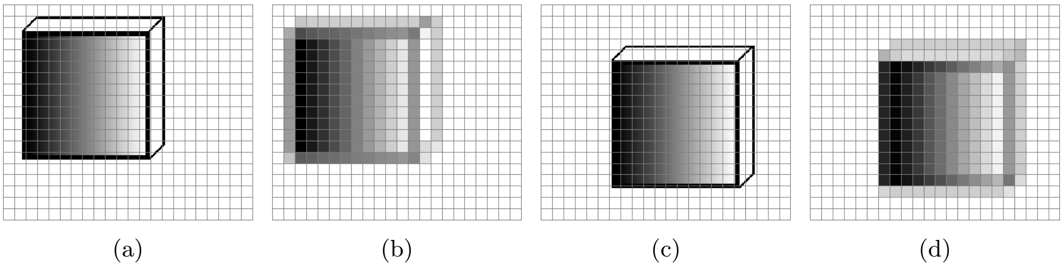


Figure 4.2: (a) and (c) show high resolution images of a cube being projected on a low resolution grid representing the image plane. Depending on the position on the grid different low resolution appearances result as shown in (b) and (d).

Another effect of the sub-sampling process that occurs when a 3D object is projected onto the 2D imaging chip is shown in Figure 4.2. The low resolution image depends on the number of pixels on the imaging chip, the size and position of the 3D object in front of the camera. An imaging chip with a smaller number of pixels results in a lower resolved image compared to an imaging chip with more pixels. The position on the imaging chip on which the object is projected will also change the appearance of the low resolution image, as illustrated in Figure 4.2.

The high resolution cubes in Figures 4.2(a) and 4.2(c) are projected onto different parts of the imaging chip. The image formation process is modelled by averaging over the number of high resolution points that fall within each pixel. The resulting effect is most prominent along the edges of the cube. Depending on whether the black edge falls in between pixels and depending on the colour of adjacent pixels, different shades of grey result in the low resolution image. Using this effect, the key idea of the proposed approach is to assume that the 3D shape of the high resolution object is known. The 3D object model is projected back into the image, low resolution images are created and then used to reconstruct the appearance of the high resolution 3D object.

4.2 Method Overview

Using the effects of the image formation process as outlined in Section 4.1, the proposed method reconstructs the high resolution appearance of a known 3D object as illustrated in Figure 4.3.

It is assumed that the 3D object is known and that a 3D model of that object is available. This is a realistic assumption given that a large number of different 3D object models are freely available on the internet or can be created with 3D software tools like Google SktechUp[1]. Such a 3D model is used within a model-based tracking approach to estimate translation and rotation parameters between consecutive frames. The model based tracking approach allows for accurate tracking of non-planar and non-rigid objects. Once the pose parameters of the current frame are estimated, the 3D object is projected back into the image and instead of using traditional texture mapping techniques, the 3D model is textured by projecting every mask triangle into the image and assigning it with a single colour value.

In order to achieve a 3D object with a high resolution texture, every mask triangle has to be smaller than a pixel when projected into the image (Smelyanskiy *et al.*, 2000). This is achieved by subdividing the 3D object model using standard computer graphics methods. Depending on the size of the object and resolution of the image, the 3D object mask

[1] http://sketchup.google.com

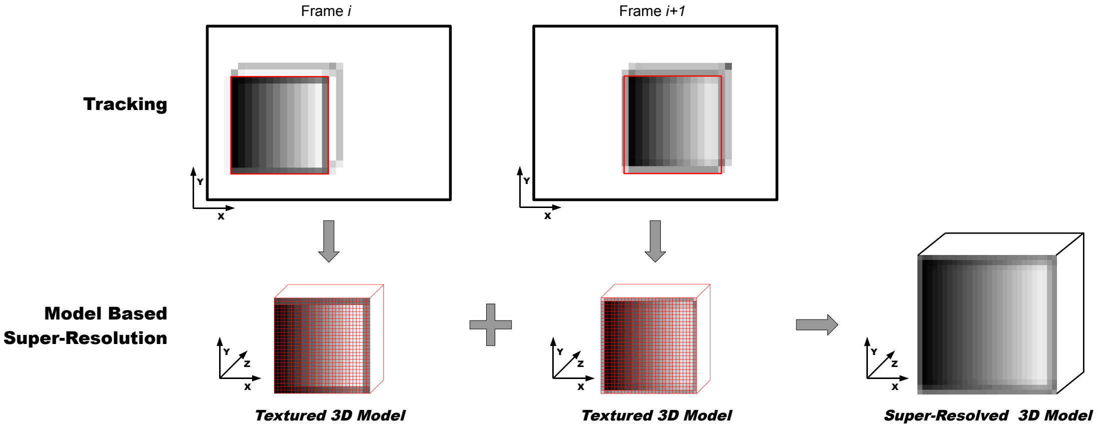


Figure 4.3: The basic outline of the proposed tracking and super resolution approach. The object of interest is tracked across several frames. Assuming that the type of object is known, the 3D model of the object is projected back into every image and every quad or triangle of the 3D model is assigned a single colour value. The super-resolved texture is then calculated as the mean across several frames.

triangles are subdivided until they are smaller than a pixel when projected into the image.

Following the previous Section 4.1, the number of pixels that the object covers within the whole image depends on the size of the imaging chip, the optical lens, the size of the object itself and the distance between object and camera. As the object or the camera moves, it may be projected onto different pixels of the imaging chip in different frames.

In Figure 4.3, the black edges surrounding the gradient on the front side of the cube are projected nearly exactly into pixel centres resulting in 14 black pixels on either side of the cube in the image plane in frame i. The movement of the cube in front of the camera results in sub-pixel movements on the image plane. The black edges of the cube now fall between pixels of the imaging chip, resulting in grey edge pixels in frame $i+1$. As a result, the two 3D models of the cube in Figure 4.3 are textured differently for each frame. Over time each model mask triangle will accumulate different colour values and thus, the super-resolved 3D model is then calculated as the mean colour value of each triangle. Without loss of generality, Figure 4.3 only shows the projection and super resolution of one side of the cube; the same is true for non-planar and/or non-rigid objects.

The super-resolved 3D model is created online during tracking and improves with every frame, whereas super resolution optical flow incorporates consecutive and previous frames which prohibits its usage as an online stream processing algorithm. Furthermore, using an object-specific 3D model in a combined tracking and super resolution approach inverses the image formation process in Equation 4.1. The subdivided 3D mesh represents the high-resolution object I^{high} that is down-sampled by projection into the image plane. The finer the mesh, the higher the resolution of I^{high} and the higher the possible increase in resolution. Thus, interpolation, the first step of the optical flow algorithm, is unnecessary and the resulting super-resolved 3D model is less blurred whilst maintaining the same resolution increase. This in turn makes deblurring (the last step of the optical flow algorithm) unnecessary. Lastly, using the 3D mesh for tracking equals image registration, warping and the estimation of a dense flow field, comprising steps 2 and 3 of the optical flow algorithm.

4.3 3D Object Tracking

For tracking low-resolution 3D objects, an object-specific mask and a combined geometric and appearance based tracking approach similar to Wen and Huang (2005) is used. They apply a geometric based tracking approach and an appearance based tracking approach for each new frame. The method that achieves best results is then used for the current frame. The 3D object tracking approach proposed here uses the same concept but differs in that different geometric and appearance based tracking algorithms are used.

The tracking of various objects requires the initialisation of the object-specific mask in the first frame, which is done either automatically (see Chapter 3) or manually. Once initialised, the tracking runs automatically. For each frame two different tracking methods are applied, each of which is described in detail:

- Appearance based tracking and,
- Geometric based tracking

4.3.1 Appearance Based Tracking

The appearance based tracking approach that is used here was first proposed by Cascia *et al.* (2000). The main difference to the proposed method lies in the use of a subdivided mask mesh and the texture mapping technique. Once the 3D object mesh model is initialised, it is subdivided into a fine mesh such that every triangle is smaller than a pixel when projected into the image. This achieves better tracking results than a mesh that is coarser with respect to the pixel size. Instead of using traditional texture mapping techniques, each triangle is projected into the image and is assigned a single colour value, instead of texture mapping a section of the image onto a single triangle.

After initialisation, the 3D object mask is projected onto the image, and each triangle is assigned a colour value. These colour values are reorganised, resulting in J_0, a vector of concatenated colour values of the initial frame. During tracking it is assumed that the difference between the reference vector J_0 and a new vector J is small (Cascia *et al.*, 2000) and thus approximated as:

$$J - J_0 \approx Oq \tag{4.2}$$

where the columns of O are called warping templates and q is a vector of coefficients.

Each warping template o_i contains the pixel value changes due to translation and rotation variations with respect to a particular transformation n_i. They are created by altering the initial pose of the 3D object mask as:

$$o_i = J_0 - \mathcal{Q}(\mathcal{P}(D, T_0 + n_i)) \text{ where } \mathrm{J}_0 = \mathcal{Q}(\mathcal{P}(\mathrm{D}, \mathrm{T}_0)) \tag{4.3}$$

where $\mathcal{P}$ is the projection of 3D object points D to image coordinates using the initial transformation T_0. $\mathcal{Q}$ creates a vector of concatenated RGB-values from the textured 3D object mask. D is a vector containing the 3D coordinates of the centre of each mask triangle, n_i is the transformation parameter displacement and J_0 is a vector of the concatenated RGB-values of each projected triangle. The intrinsic camera transformation parameters are obtained using standard camera calibration techniques (Zhang, 2000). A rough camera calibration is sufficient as the actual size of the object is not important.

The warping templates are calculated only once after initialisation. The object is then tracked by using the pose parameters of the previous frame as initialisation and solving for q for each frame f as:

$$J_0 - \mathcal{Q}(\mathcal{P}(D, T_f^{app})) \approx Oq, \tag{4.4}$$

where the columns of O are the warping templates o_i and T_f^{app} contains the transformation parameters for the appearance-based tracking at frame f.

4.3.2 Geometric Based Approach

The geometric based tracking uses a standard template matching approach restricted by the object-specific mask. Unlike the method proposed in Wen and Huang (2005) no action units are used to track facial movements. The extension to non-planar and non-rigid objects for tracking facial expressions is proposed in Section 4.5.

Objects are tracked by projecting each vertex P of the 3D mask mesh into the previous image using perspective projection. Around each projected vertex in the image, a rectangular template is cropped and matched with the current frame. The size of the patch is set to $\frac{1}{6}$th of the whole object. Normalised cross-correlation is used to match this patch in the current frame within a window that is double the size of the template. In order to minimise the effect of outliers, the entire mask is fitted to the new vertex points p_l in the current frame f as:

$$T_f^{geo} = \min_{T^{geo}} \sum_{l=1}^{L} (\mathcal{P}(P_l, T_f^{geo}) - p_l)^2, \tag{4.5}$$

where L is the number of mask vertices P and T_i^{geo} contains the transformation parameters of the geometric-based approach at frame f. Again the transformation parameters are initialised with the previous frame and Levenberg-Marquardt is utilised for solving for T^{geo}.

4.3.3 Combined Appearance and Geometric Tracking

During the tracking process each method is applied individually and a texture residual as the root mean squared error (RMSE) for the current frame f, with respect to the first frame 0 is calculated as:

$$RMSE(T_f) = \sqrt{\frac{1}{M} \sum_{m=1}^{M} (\mathcal{Q}(\mathcal{P}(M_m, T_0)) - \mathcal{Q}(\mathcal{P}(M_m, T_f)))^2}, \tag{4.6}$$

where M is the number of mask triangles M. The pose parameters T_f of the method with the smallest RMSE will be used for the current frame f as:

$$T_f = \min_{T_f} \left(\left[RMSE(T_f^{app}), RMSE(T_f^{geo}) \right] \right), \tag{4.7}$$

where $RMSE(T_f^{app})$ and $RMSE(T_f^{geo})$ are the texture residuals of the appearance-based and the geometric-based approach respectively for frame f. The tracking runs automatically once the mesh mask is initialised in the first frame of the sequence.

4.4 3D Model Based super resolution

During tracking, the resolution of the low-detail object is gradually increased. To achieve this, every triangle of the object-specific mask is projected into the video using perspective projection. However in order to increase the resolution of the object, every triangle needs to be smaller than a pixel when projected into the image (Smelyanskiy *et al.*, 2000).

As each mask triangle is projected into different frames of the sequence, it is eventually assigned with different colour values for each frame as shown in Figure 4.3. Therefore the super-resolved mask J_{SR} is calculated as the mean of the last r frames that result in an RMSE below a certain threshold ϵ:

$$J_{SR} = \frac{1}{r}\sum_{f=1}^{r}(\mathcal{Q}(\mathcal{P}(M, T_f))) \quad with \quad RMSE(T_f) < \epsilon \tag{4.8}$$

Small tracking errors (RMSE) allow for an exact alignment of the 3D mask across frames, whereas high RMSE result in blurring and distortion. The threshold ϵ depends on the initial object resolution. Low-resolution objects usually result in higher RMSE during tracking as image pixels are more likely to change due to the down-sampling process of the imaging chip. Furthermore, the quality of the super-resolved mask J_{SR} also depends on the total number of frames r. However, as the number of frames increases, the probability of introducing noise increases as frames might not be aligned perfectly. The issue of choosing the appropriate number of frames r versus the quality of the super-resolved mask is evaluated empirically in Section 4.6.2.

The combined tracking and super resolution method uses a 3D object specific mask and thus only increases the resolution of that object and not of the whole scene. However many applications, like face recognition or number plate enhancement, only require the object of interest to be super-resolved. Furthermore this high-resolution 3D object mask can then be used for a number of applications like generating different high-resolution views of the object from different viewpoints and lighting conditions.

4.5 Extension to Non-planar and Non-rigid Objects

In order to increase the resolution of non-planar and non-rigid objects the tracking algorithm needs to allow for deformations, i.e. the mask mesh representing the 3D object needs to be deformable. This is especially an issue when tracking non-rigid objects like faces.

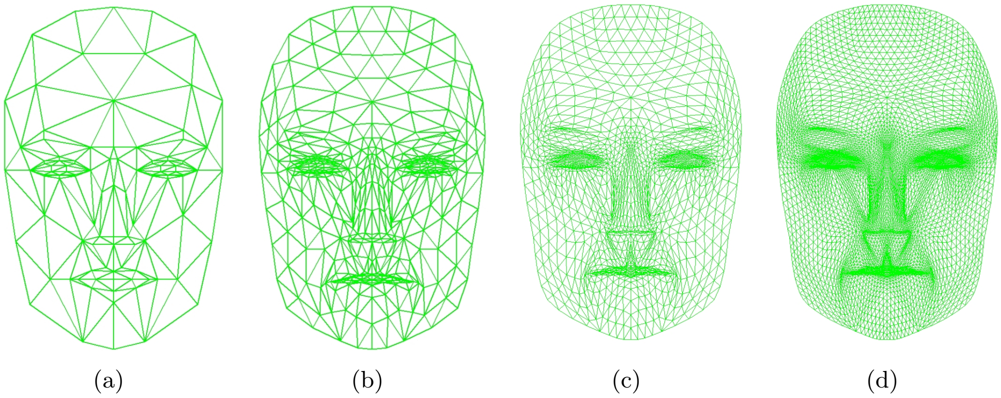


Figure 4.4: (a) CANDIDE-3 face mask with 184 triangles, (b), (c) and (d) are subdivided masks after 1, 2 and 3 subdivision steps resulting in 736, 2944 and 11776 triangles.

Therefore, the proposed combined tracking and super resolution approach is extended to non-planar and non-rigid objects and is applied to faces, in particular human faces.

For tracking faces, the CANDIDE-3 face model as proposed by Ahlberg (2001) is used. As shown in Figure 4.4 this triangular mesh consists of 104 vertices and 184 triangles and is subdivided using the Modified Butterfly algorithm and the Loop subdivision scheme as described in Appendix A. To allow for the non-rigidity of faces the CANDIDE-3 expression parameters for tracking mouth and eyebrow movements in low-detailed faces are used. More complex facial expressions often require a more detailed face model as well as high-resolution images (Goldenstein *et al.*, 2003; Roussel and Gagalowicz, 2005; Wang *et al.*, 2005c).

Expression tracking is performed after the actual tracking for each frame. Using the expression parameters of the last frame, the combined geometric and appearance based tracking approach is used to determine the position of the mesh model in the current frame. Next a global random search (Zhigljavsky, 1991) is performed to improve the RMSE around the mouth and the eyebrow regions. This probabilistic search assumes a Gaussian distribution of the expression parameters and independence. For the expression parameters of the mouth and the eyebrow region, this is a suitable assumption. The mean of this normal distribution is initialised with zero in the first frame and is set to the expression value of the previous frame after that. The variance is assumed to be constant and empirically set to 0.2.

This normal distribution is then used to randomly sample 10 to 20 different expression values for each expression parameter. These samples are evaluated against the observed image and a RMSE according to Equation 4.6 is calculated. The sample that results in the smallest RMSE is chosen for the current frame. Experiments in Section 4.6.1.2 show that the proposed method for expression tracking reduces the mean tracking error and thus allows for a better alignment of consecutive frames.

4.6 Experiments

The following subsections describe the experiments and their evaluation to demonstrate the capability of the proposed combined tracking and super resolution approach. Most experiments use a video sequence of rigid objects (a cube) or non-rigid objects (human faces) that has been recorded in the lab or in a real world surveillance environment. In the lab environment the video sequences are recorded with 15 fps and with a resolutions of 640×480 pixels or 320×240 pixels unless otherwise stated. In the surveillance environment, the video resolution is 640×480 pixels recorded at 23 fps on average.

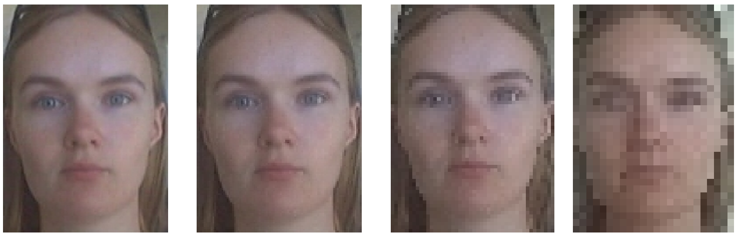
The proposed approach is systematically tested starting with the evaluation of the tracking accuracy in Section 4.6.1, including tracking of non-rigid facial expressions and tracking with different mask sizes. In the following Section 4.6.2, the proposed approach is tested on non-rigid and rigid objects at different image resolutions. Lastly, the combined tracking and super resolution method is compared with an optical flow approach in Section 4.6.3 and applied to real world footage in Section 4.6.3.1.

4.6.1 Tracking Accuracy

The following three sections examine the proposed tracking approach with respect to

- Combined Appearance and Geometric Tracking
- Expression Tracking
- Mask Mesh Size vs. Object Size

The first section examines the use of an appearance tracker and a geometric tracker versus the use of a combined tracking approach. Next the expression tracking is compared to tracking without allowing for non-rigid deformations. Lastly, the effect of the 3D mesh size with respect to the object size on the tracking result is examined.

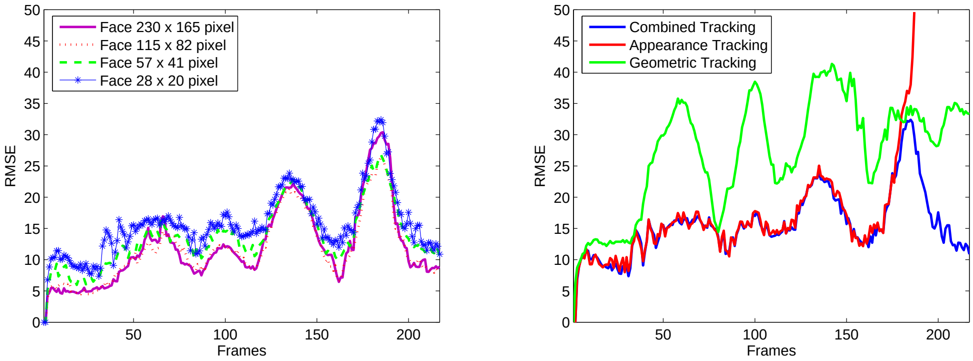


(a) 230×165 (b) 115×82 (c) 57×41 (d) 28×20

Figure 4.5: Cropped faces for each image resolution used. The captions indicate the size of the face in pixels.

4.6.1.1 Combined Appearance and Geometric Tracking

In order to examine the tracking accuracy of the combined geometric and appearance based tracking algorithm, one video sequence of a face with translation and rotation movements is recorded at 15 frames per second and an initial resolution of 640×480 pixels. The face within one frame has an average size of 230×165 pixels. This resolution is divided into halves three times, resulting in face sizes of 115×82, 57×41 and 28×20 pixels with corresponding frame sizes of 320×240, 160×120 and 80×60 respectively. A cropped face for each face size used is shown in Figure 4.5.



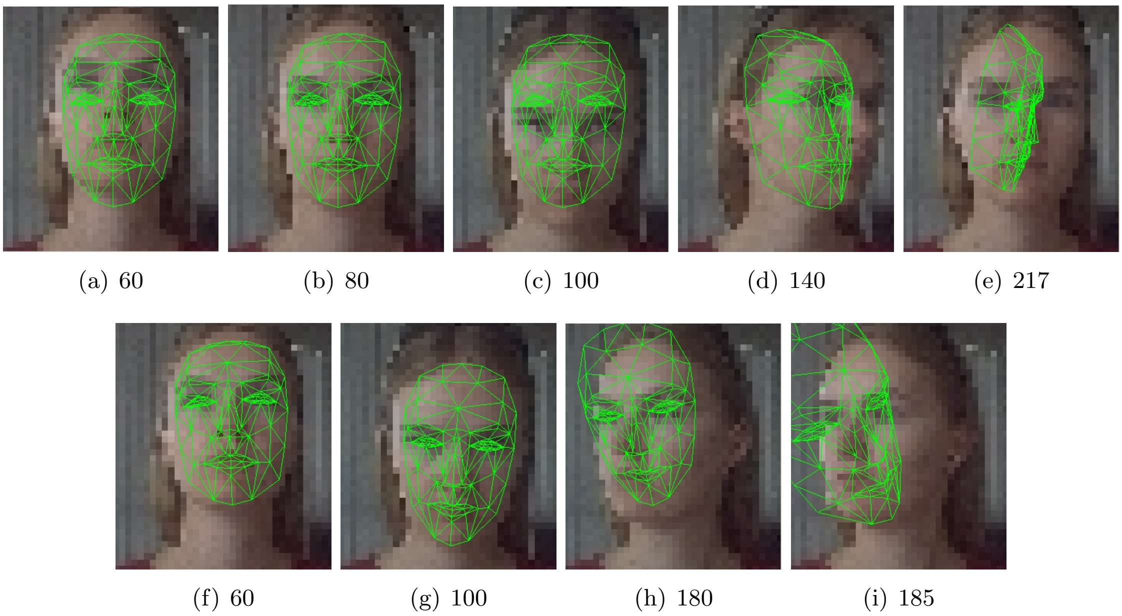
(a) (b)

Figure 4.6: Tracking results for different size faces (a) and each tracking method applied individually to the video of resolution 80×60 pixels with a face of size 28×20 pixels (b).

For tracking faces, the CANDIDE-3 face model, as shown in Figure 4.4, is used. In order to initialise this mask in the first frame of the sequence, the shape parameters of the CANDIDE-3 model are adjusted to the face manually. After this initialisation, the mask is tracked automatically over more than 200 frames using the combined geometric and appearance-based approach described in Section 4.3.

The result of the combined tracking algorithm applied to different face sizes is shown in Figure 4.6(a). The RMSE for measuring the tracking accuracy with respect to the first frame is defined in Equation 4.6. The variation in RMSE in frames 1 to 100 are due to translation and rotation around the horizontal x-axis, whereas the peaks at frames 130 and 175 respectively are mainly due to rotation around the vertical y-axis.

Faces between 230×165 and 115×82 pixels in size result in similar RMSE, whereas faces with a resolution down to 57×41 pixels result in a slightly increased tracking error, that is 22% larger on average. Even though the RMSE increases by 41% when tracking faces with a resolution of 28×20 pixels, the algorithm is still able to qualitatively track the face to the end of the sequence.



(a) 60 (b) 80 (c) 100 (d) 140 (e) 217

(f) 60 (g) 100 (h) 180 (i) 185

Figure 4.7: Cropped sample frames of the geometric based tracking (a)-(e) and the appearance based tracking (f)-(i). Numbers denote different frames.

In comparison, Figure 4.6(b) shows the result of the geometric and appearance based tracking approach operating individually on the same video sequence, with the smallest face size of 28×20 pixels, as this face size is the most difficult to track. The geometric approach loses track after 40 frames, and as shown in Figure 4.7(a), this is due to small inter frame movements that cause the mask to stay in the initial position instead of following the face. The mask then recovers in frame 80 and loses track immediately afterwards as shown in Figures 4.7(b) and 4.7(c). The geometric approach finally loses track around frame 140, from which it cannot recover, as shown in Figures 4.7(d) and 4.7(e). This shows that the geometric approach is not able to handle small inter frame movements due to image noise and low resolution.

The appearance based approach, on the other hand, results in small tracking errors from frame 1 through to frame 170 as shown in Figures 4.7(f) and 4.7(g). However, as the face turns, the appearance based approach loses track from which it cannot recover, as shown in Figure 4.7(h) and 4.7(i). This is mainly due to large inter frame movements and the rotation of the face, resulting in partial occlusion of the face.

Figure 4.6(b) shows that by combining the geometric and appearance based approach, tracking is improved. Both approaches complement one another resulting in smaller RMSE than either of them individually. While the appearance based method tends to be more precise for small inter-frame movements, the geometric method is better for larger displacements. Furthermore the geometric approach applies template matching between the current and previous frames, while the appearance approach is based on the comparison of the current frame with the first frame. Thus, the combination is more stable and precise and able to track even small size faces, down to 28×20 pixels in size.

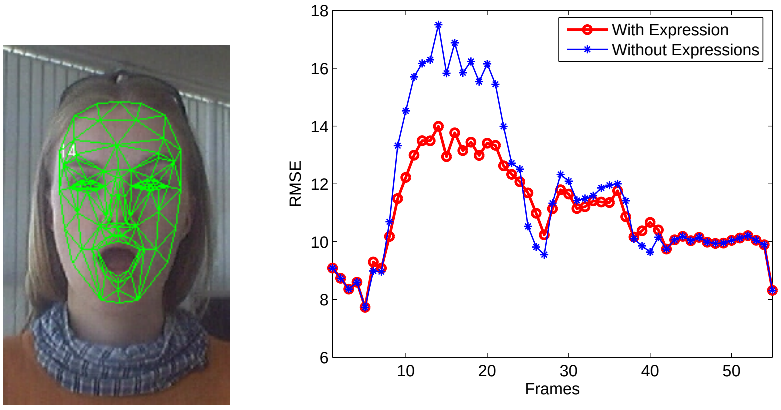
4.6.1.2 Expression Tracking

In order to evaluate the performance of the expression tracking approach, one video of a face with mouth and eyebrow movements is recorded. The face within each frame is about 230×165 pixels. One frame of this sequence is shown in Figure 4.8(a). The graph in Figure 4.8(b) compares the result of expression tracking with the combined geometric and appearance based tracking approach without expression tracking. Frames 10 to 22 contain mouth openings and frames 28 to 38 contain eyebrow movements. The graph shows clearly that expression tracking improves the result of the combined tracking approach by reducing the RMSE for each frame.

Face Resolution	With Expressions	Without Expressions	Improvement
210×145	10.89	11.50	5.3 %
105×73	10.96	11.60	5.5 %
52×37	13.92	14.51	4.1 %
26×18	16.23	16.40	1.0 %

Table 4.1: The mean tracking RMSE of 56 frames for different face resolutions with and without expression tracking.

Furthermore, the resolution of the video is cut in half three times resulting in face resolutions of 210×145, 105×73, 52×37 and 26×18 pixels in size. The mean tracking error across 56 frames for each resolution is shown in Table 4.1. While the difference in RMSE amounts to about 0.60 (which equals an improvement of 5.5% to 4.1%) for the first three



(a)

(b)

Figure 4.8: Results of the expression tracking. Subfigure (a) shows a single frame of the middle of the sequence and (b) compares the tracking RMSE with and without expression tracking.

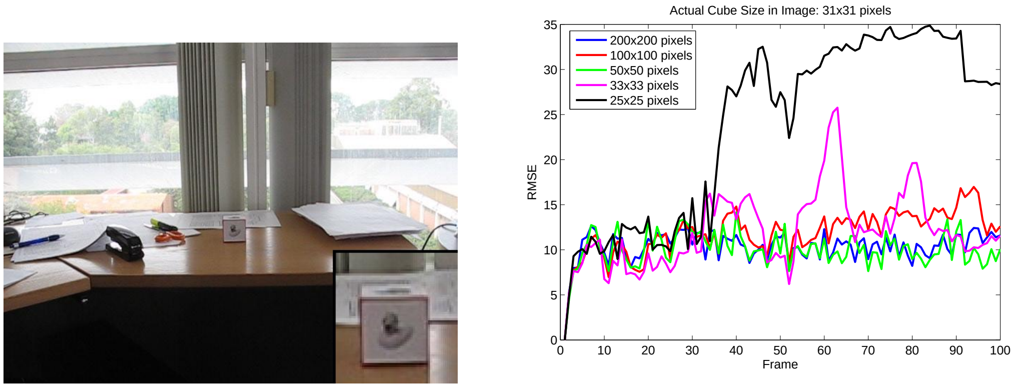
resolution levels, a face of size 26×18 pixels only results in a RMSE difference of 0.17 compared to the tracking approach without expressions. This translates to an improvement of only 1.0%.

The smaller the resolution of the face, the higher the RMSE as shown in Figure 4.6(a) and Table 4.1. A face that is captured at high-resolution results in a large number of pixels that represent the face. However the smaller the number of representative pixels, the more likely they are to change over time. Due to the discretisation of the imaging chip certain face regions (e.g. the eyes) result in only a small number of pixels. The colour value of these pixels is most likely to change over time as the camera or the face moves. Therefore, tracking expressions of low-resolution faces does not improve the overall RMSE significantly.

4.6.1.3 Mask Size vs. Object Size

The performance of the combined geometric and appearance-based tracking algorithm is tested with different mask sizes. As described in Section 4.2, super resolution is only possible when the mask mesh is subdivided such that every mask triangle is smaller than a pixel when projected into the image. The following experiment evaluates the effect of the mesh size on the tracking performance.

The front side of a cube as shown in Figure 4.9(a) is tracked across 100 frames. This



(a) Example frame

(b) Object Size 31x31

Figure 4.9: Mean tracking RMSE across 100 frames for a planar object, the front side of a cube. The bottom right corner shows a cropped and enlarged image of the cube (a). Using a object mask mesh that is finer than the actual object results in smaller tracking errors (b).

planar patch covers an area of 31×31 pixels within the image. For tracking this patch a 3D model mesh similar to the one in Figure 4.1 is used. This mesh is equally subdivided into 25×25, 33×33, 50×50, 100×100 and 200×200 quads. Figure 4.9(b) shows the result of the combined tracking approach when different mesh sizes are used.

For tracking a planar patch of size 31×31, the best result is achieved with mesh sizes larger than 33×33, whereas further subdivision does not improve the tracking. Using a mesh that is coarser with respect to the pixel size loses track easily and results in higher RMSE during tracking as shown in Figure 4.9(b). Such a coarse mesh is an under representation of the object, resulting in higher tracking errors. By using a larger number of quads, the mesh is able to better account for appearance changes due to sub-pixels movements.

	Object Size	
Mesh Size	31×31	17×17
13×13	-	20.84
16×16	-	23.49
20×20	-	18.78
25×25	23.88	19.09
33×33	12.06	17.78
50×50	10.03	15.62
100×100	11.88	16.25
200×200	10.40	-

Table 4.2: Mean RMSE across 100 frames for different object sizes and different mesh sizes used.

Another video of the same object was recorded at greater distances, resulting in a cube of size 17×17 pixels. The mask mesh used for tracking consists of 13×13, 16×16, 20×20, 25×25, 33×33, 50×50 or 100×100 quads. The corresponding mean tracking errors are shown in Table 4.2. Again the best tracking results are achieved by a mask mesh that contains a larger number of quads than the pixels covered by the object within the image.

Summarising, the combined geometric and appearance-based tracking approach achieves best results when a fine model mesh is used. In practice a mesh that is double the size of the object has proven to be the best trade-off between accuracy and speed.

4.6.2 3D Model Based super resolution

In order to increase the resolution of the object of interest, the object-specific 3D mesh model must be subdivided into a fine mesh. Each quad or triangle must be smaller than a pixel when projected into the image to make super resolution possible as illustrated earlier in Figure 4.1. The possible increase in resolution depends on the size of the 3D model mesh. The finer the mesh, the higher the possible increase in resolution, however more frames are needed. The following experiments quantitatively evaluate the number of frames needed to achieve different resolution increases. Therefore planar and non-planar objects are examined in Sections 4.6.2.1 and 4.6.2.2 respectively.

4.6.2.1 Planar Object - A Cube

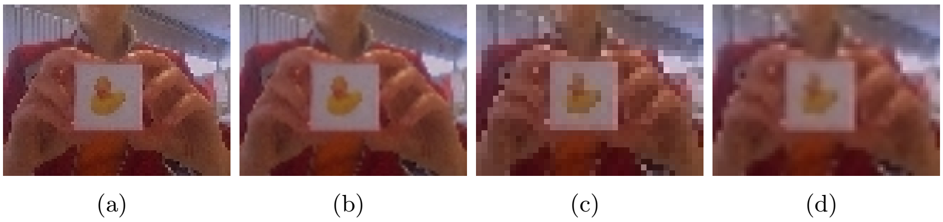


Figure 4.10: Example frame of resolution (a) 80×60 pixels and (c) 40×30 pixels with cube sizes of 24×24 pixels and 12×12 pixels respectively. Figure (b) and (d) show these frames after they have been doubled in size using bilinear interpolation.

A small paper cube, as shown in Figure 4.10, is chosen as a representative for planar objects. This cube is tracked for more than 200 frames of one video sequence, recorded at a resolution of 320×240 pixels with the cube of size 95×95 pixels. This video is sub-sampled three times, resulting in resolutions of 160×120, 80×60 and 40×30 pixels and corresponding cube sizes of 48×48, 24×24 and 12×12 pixels respectively. In order to diminish the effect of tracking errors, the cube is tracked at the highest resolution of 95×95 pixels. The estimated pose parameters are then used for the cube of size 24×24 pixels and 12×12 pixels. An example frame for both these resolutions is shown in Figures 4.10(a) and 4.10(c) respectively. For comparison Figures 4.10(b) and 4.10(d) show these frames after their resolution is doubled using bilinear interpolation.

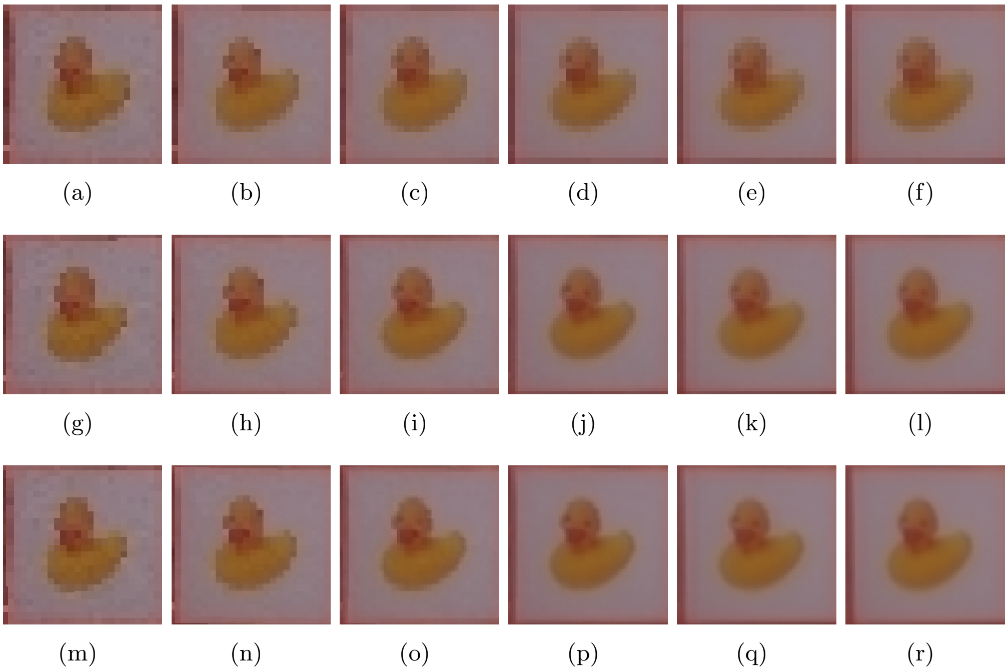

(a) (b) (c) (d) (e) (f)
(g) (h) (i) (j) (k) (l)
(m) (n) (o) (p) (q) (r)

Figure 4.11: Super resolution results of a cube of size 24×24 pixels using mesh consisting of 25×25 (top row), 50×50 (middle row) and 100×100 quads (bottom row) after 1, 10, 20, 50, 100 and 200 frames respectively.

For increasing the resolution of the cube, a 3D model as shown in Figure 4.3, is used and subdivided into 25×25, 50×50 and 100×100 equal size quads. Using the video sequence of a cube of size 24×24 pixels, this mesh is projected onto every frame of the sequence and the super-resolved 3D model J_{SR} is created according to Equation 4.8 by combining 1, 10, 20, 50, 100 or 200 frames. The results is shown in Figure 4.11.

A mesh with only 25×25 quads cannot increase the resolution of a cube of size 24×24 pixels. Calculating the mean according to Equation 4.8 across 20 frames removes the camera noise and partially recovers the eyes of the duck, which are not visible in the first frame (Figure 4.11(c)). Using a mesh that is double in size (50×50 quads) results in a more detailed image, but after about 20 to 50 frames the maximal possible resolution is achieved and adding more frames does not improve the resolution further as shown in Figures 4.11(i) and 4.11(j) respectively.

Using a mesh with 100×100 quads equals to increasing the resolution four times in each dimension. However, to achieve this increase more than 50 frames are needed as shown in Figure 4.11(p), but taking the mean across such a large number of frames also introduces noise as shown in Figure 4.11(r). This noise results from slightly miss-aligned frames and from the averaging process itself.

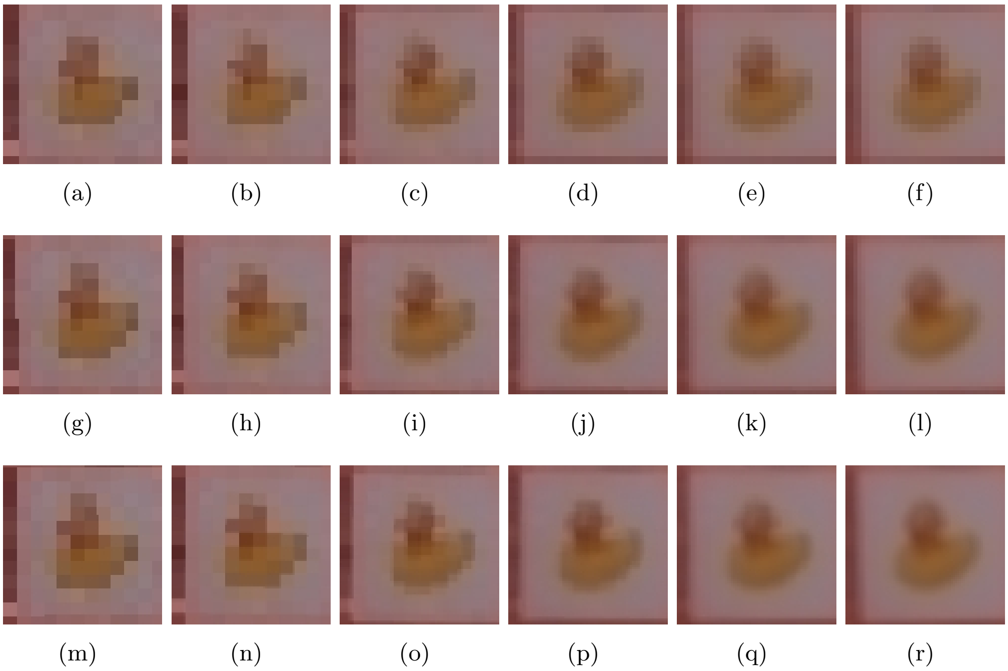


Figure 4.12: Super resolution results of a cube of size 12×12 pixels using a mesh with 20×20 quads (top row), 40×40 quads (middle row) and 80×80 quads (bottom row) after 1, 10, 20, 50, 100 and 200 frames respectively.

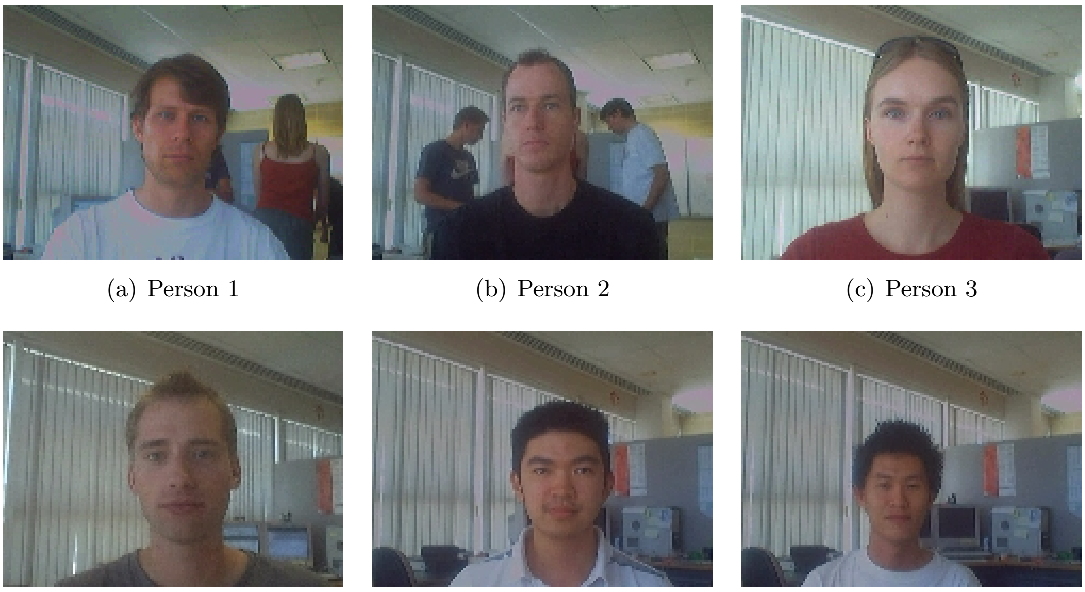
The same experiment is performed on a cube of size 12×12 pixels using a mesh with 20×20, 40×40 and 80×80 quads. The result after combining 1, 10, 20, 50, 100 and 200 frames is shown in Figure 4.12. Using a mesh with 20×20 quads, which equals a resolution increase of 166%, requires about 20 to 50 frames (Figures 4.12(c) and 4.12(d)). Adding more frames does not improve the result further.

Using a mesh with 40×40 quads translates to a possible resolution increase of 3.3 times in each dimension and requires about 100 frames as shown in Figure 4.12(k). Trying to increase the resolution 6.6 times requires a mesh with 80×80 quads and about 200 frames as shown in Figure 4.12(r). The shape of the duck is recovered in greater detail, however the overall result is noisy and blurred as a result of taking the mean across 200 frames.

In practice, it is therefore not recommended to increase the resolution of an object by more than 2 to 3 times. The higher the increase in resolution, the more frames are needed to achieve this resolution, which in turn results in more noise. It is therefore a trade-off between the possible resolution increase and the number of frames. Furthermore, the tracking error ϵ in Equation 4.8 influences the resulting super resolution image. Large tracking errors lead to a misalignment of frames resulting in noisy and blurred super-resolved images.

4.6.2.2 Non-planar Objects - Faces

Simple objects, like a cube, allow for an equal subdivision of the 3D model mesh. More complex objects require a 3D model that consists of different size quads or triangles, thus resulting in a varying resolution increase across the mask mesh. For increasing the resolution of faces, the CANDIDE-3 mask as shown in Figure 4.4 is used. This mask is finely sampled around the eyes, the mouth and the nose region. These areas are also the most important parts of the face and therefore a finer sampled mask with smaller triangles allows for a larger increase in resolution in these areas.



(a) Person 1

(b) Person 2

(c) Person 3

(d) Person 4

(e) Person 5

(f) Person 6

Figure 4.13: Example images of each of the six sequences, i.e. persons, used for the results in Figure 4.15. The image resolution is 160×120 pixels.

Evaluating the number of frames needed to achieve a certain resolution is more difficult using such a complex mask as the resolution increase varies across the entire mask due to different size triangles. In order to evaluate the minimum number of frames needed to create a face mask of higher resolution, faces are tracked in videos with minimal head movements in order to keep the tracking error to a minimum. The average size of the faces are 60×40 pixels and 30×20 pixels using videos of resolution 160×120 pixels and 80×60 pixels respectively. These resolutions are sub-sampled from the original video sequence recorded at 640×480 pixels. The mask mesh is manually fitted to the first frame of each sequence and then tracked fully automatically across more than 200 frames. Example frames of each sequence are shown in Figures 4.13 and 4.14.

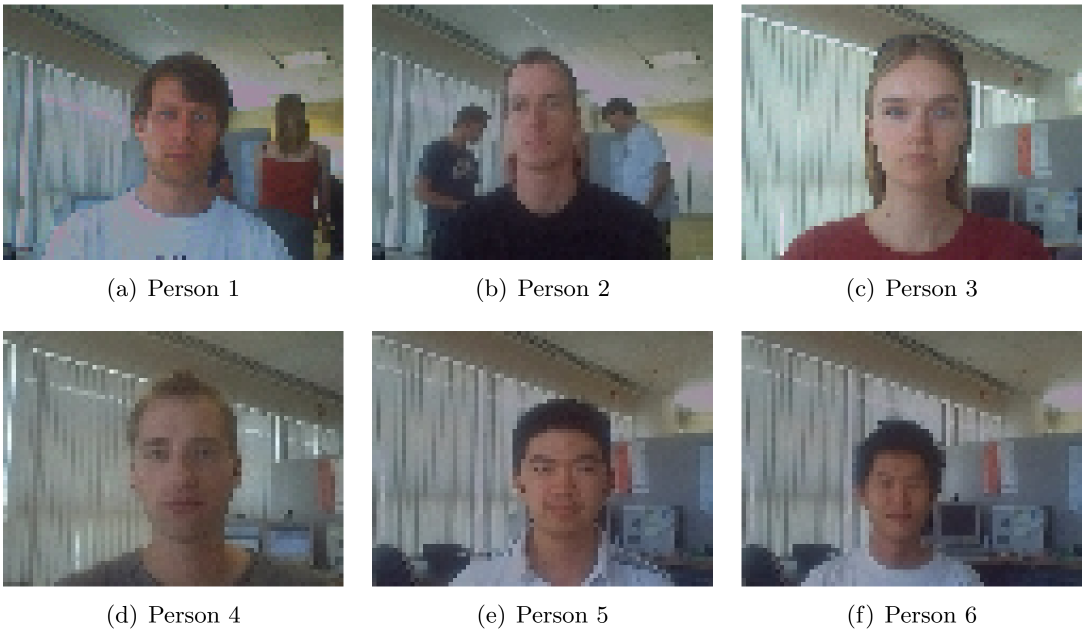

(a) Person 1
(b) Person 2
(c) Person 3
(d) Person 4
(e) Person 5
(f) Person 6

Figure 4.14: Example images of each of the six sequences, i.e. persons, used for the results in Figure 4.15. The image resolution is 80×60 pixels.

The super-resolved mask J_{SR} is calculated by using between 1 to 200 frames with the smallest tracking RMSE according to Equation 4.8. The mean tracking RMSE is 10.9 and 14.9 for faces of size 60×40 pixels and 30×20 pixels respectively. The CANDIDE-3 mask that is subdivided three times, as shown in Figure 4.4(d), is used for both face sizes. The high-resolution mask created from faces of size 30×20 pixels is then compared with a single frame of double the resolution (60×40 pixels) and the mask created from faces of size 60×40 pixels is compared to the face of size 120×80 pixels respectively. In each case, the mean colour difference E_{colour} is used to compare two face masks J_1 and J_2 consisting of M triangles each as:

$$E_{colour} = \frac{1}{M}\sum_{m=1}^{M} |J_1(m) - J_2(m)|_2 \tag{4.9}$$

One short video sequence of six different persons, as shown in Figures 4.13 and 4.14, is used and the results for these six persons is shown in Figure 4.15. Common to all persons and face sizes is the strong error decrease within the first 20 to 30 frames. Within the first 20 frames, faces of size 60×40 pixels increase most significantly with respect to a face of double the size. Faces of size 30×20 pixels are smaller and, therefore, more frames are needed to achieve the same resolution increase using a mask mesh with the same number of triangles. After about 20 to 30 frames the resolution increases significantly. These results are comparable to the results of the cube shown in Figures 4.11 and 4.12.

For a qualitative comparison the super resolution results for Person 1 to 6 are shown in Figures 4.18 to 4.21. The faces on the left show the facial mask that is textured with a single frame of a face of size 60×40 pixels (Figures 4.18(a) to 4.21(a)) and 30×20 pixels (Figures 4.18(f) to 4.21(f)) respectively. The second, third and fourth mask show the increase in resolution after 20 (Figures 4.18(b) and (g) to 4.21(b) and (g)), 50 (Figures 4.18 (c) and (h) to 4.21(c) and (h)) and 200 (Figures 4.18 (d) and (i) to 4.21(d) and (i)) frames have been added to the super-resolved mask.

The first step of the super resolution optical flow algorithm doubles the size of the input images using interpolation techniques (Section 2.3.3). Therefore, bilinear interpolation is used to increase the size of the video of each person. The result after combing 200 frames of the interpolated input frames is shown in Figures 4.18 (e) and (j) to 4.21(e) and (j) for face sizes of 60×40 pixels and 30×20 pixels respectively. Even though the input images are double in size, the resulting super-resolved faces show less detail and are more blurred. Interpolation does not recover high-frequencies, and on the contrary, introduces further noise, and should therefore be avoided during the super resolution process.

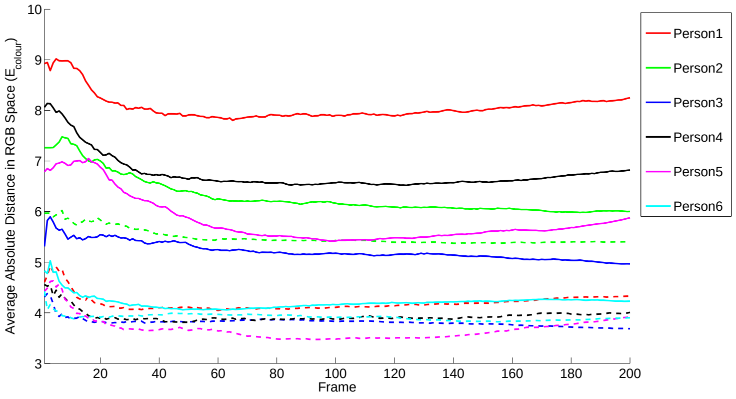


Figure 4.15: Quality of the super-resolved 3D face mask using different resolutions (60×40 pixels, dotted line and 30×20 pixels solid line) and number of frames (x-axis).

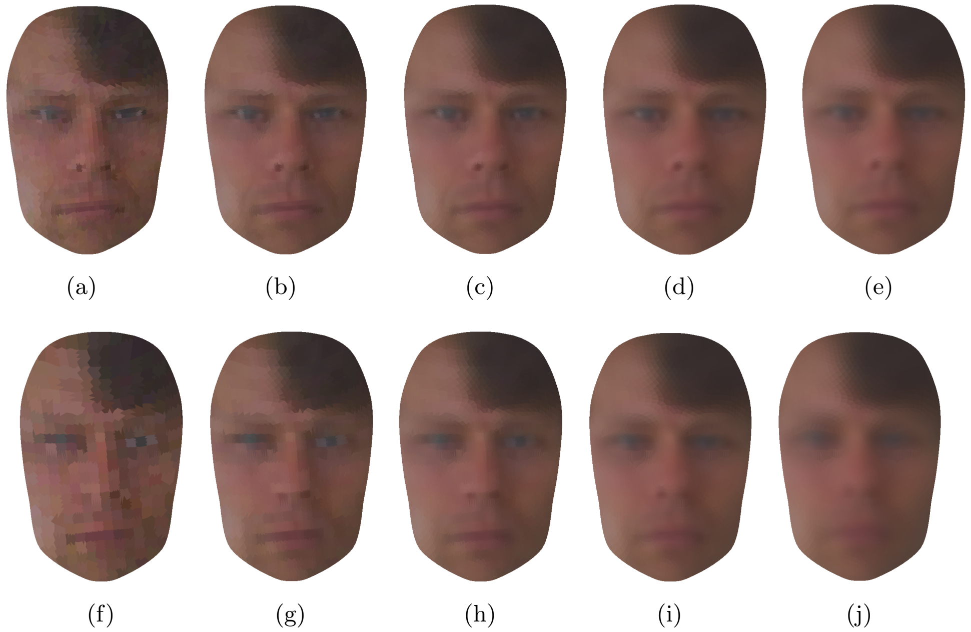


Figure 4.16: Results of the combined tracking and super resolution approach for Person 1 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.

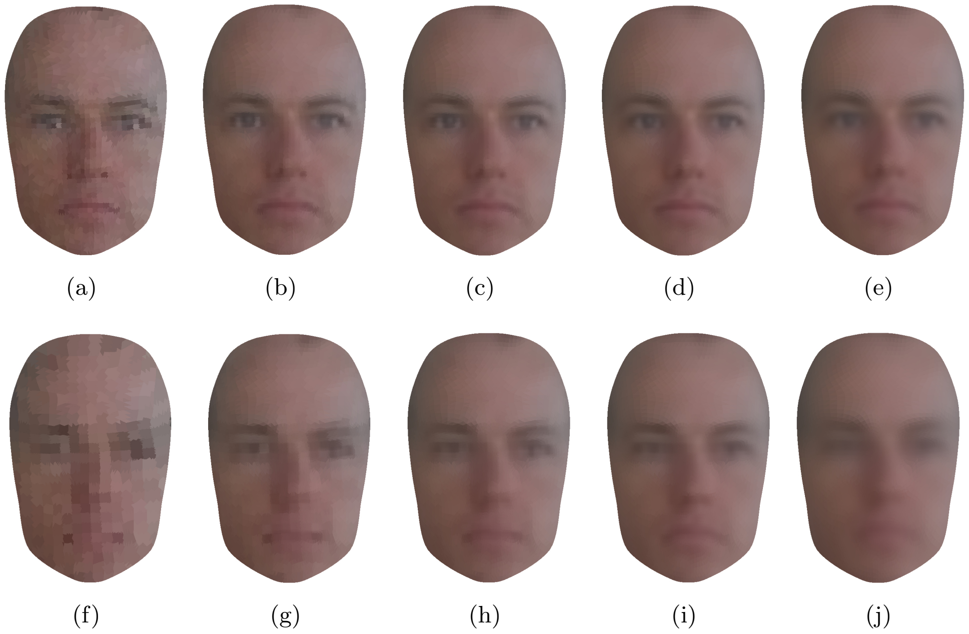
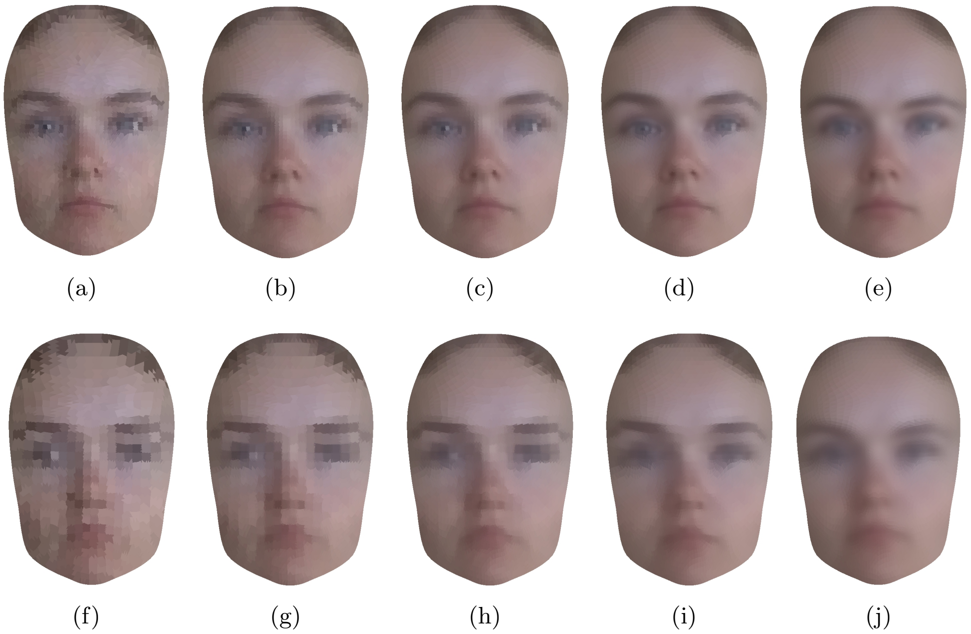
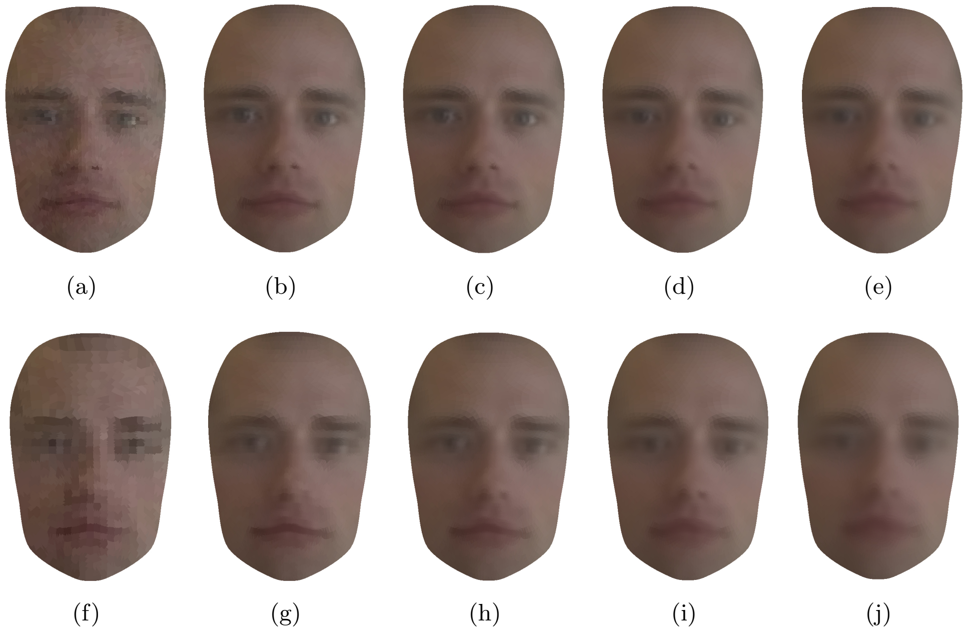


Figure 4.17: Results of the combined tracking and super resolution approach for Person 2 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 4.18: Results of the combined tracking and super resolution approach for Person 3 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.


(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 4.19: Results of the combined tracking and super resolution approach for Person 4 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.

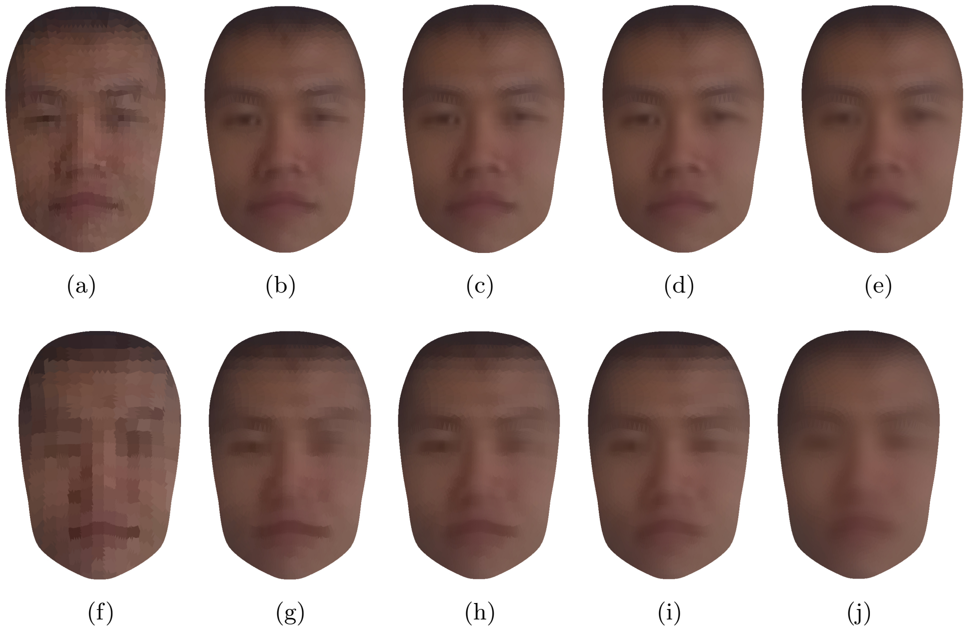


Figure 4.20: Results of the combined tracking and super resolution approach for Person 5 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.

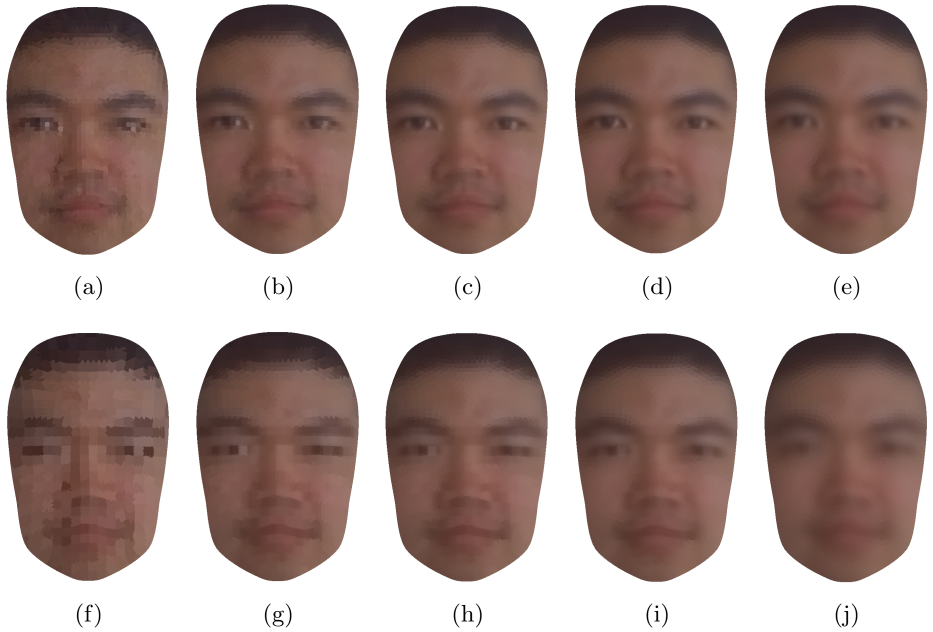


Figure 4.21: Results of the combined tracking and super resolution approach for Person 6 with a face of size 60×40 pixels (top) and 30×20 pixels (bottom) after 1, 20, 50 and 200 frames respectively. The last column ((e) and (j)) shows the result after 200 frames when the input images are interpolated to double the size using bilinear interpolation.

4.6.3 Comparison with super resolution Optical Flow

The proposed approach is compared with the super resolution optical flow technique, using video sequences recorded in a lab environment as well as surveillance video of faces. The implementation of the super resolution optical flow follows the first four steps as outlined in Section 2.3.3. No deconvolution techniques are used as this is an additional option for both the proposed approach and the optical flow method to further increase the quality of the super-resolved images. The optical flow between consecutive frames is calculated using Gautama and van Hulle (2002) and the mean is used to calculate the super-resolved image.

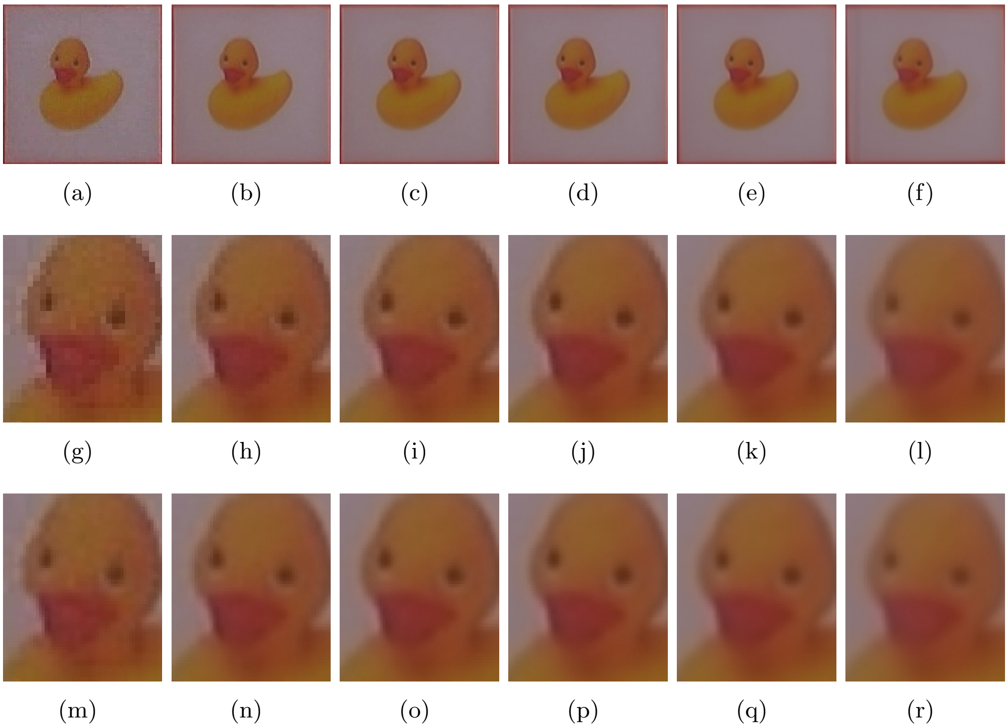


Figure 4.22: Result of the proposed super resolution approach after combining (a) 1, (b) 5, (c) 10, (d) 20, (e) 50, (f) 100 frames. Figures (g)-(l) show cropped parts of every Figure (a)-(f) respectively and Figures (m)-(r) show the result after each input image was doubled in size using bilinear interpolation.

A video sequence of a cube with a resolution 320×240 pixels is used. The actual cube covers about 100×100 pixels in each frame. The combined geometric and appearance based approach is used to track this cube across more than 100 frames resulting in a mean tracking error of 9.16. The pose parameters are then used to project the 3D model into every image and the top k frames with the smallest tracking error are used to calculate the super-resolved image J_{SR} according to Equation 4.8.

The 3D model of the cube is subdivided into 400×400 triangles before being projected into the image which, under ideal conditions, equals a resolution increase of 400%. The result is shown in Figure 4.22. After 20 frames (Figures 4.22(d) and 4.22(j)) the maximum resolution increase is reached and further added frames result in increased blur due to tracking errors. For comparison, the size of each frame is doubled using bilinear interpolation before creating the super resolution image. Again a cube with 400×400 triangles is used and the result is shown in Figure 4.22(m) to Figure 4.22(r). Even though the input images are doubled in size, the resulting super-resolved images after 20 frames (Figure 4.22(p)) do not show a significant resolution increase compared to the result without initial interpolation (Figure 4.22(j)). On the contrary, the resulting super-resolved images show a greater amount of blur as interpolation cannot recover high-frequency details (Figure 4.18).

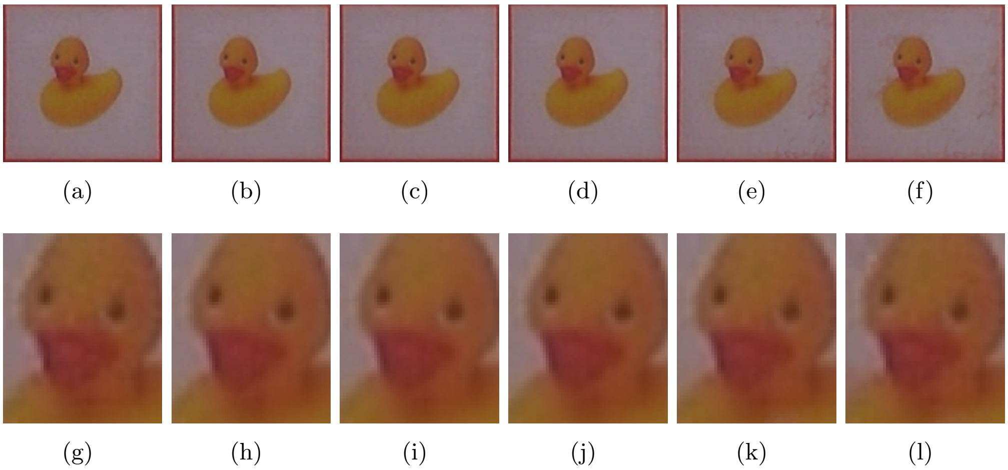


Figure 4.23: Result of the super resolution optical flow algorithm after calculating the mean across (a) 1, (b) 5, (c) 10, (d) 20, (e) 50 and (f) 100 frames. A cropped part of each image is shown in (g)-(l) respectively.

Super resolution optical flow increases the resolution of each frame by interpolation. Several frames are then combined to enhance the quality of these interpolated images, but no further increase in resolution is possible, the resolution increase is fixed at 200%. This is contrary to the proposed approach as it avoids interpolation to allow for less blurred images. However, optical flow based methods require less frames as shown in Figure 4.23. The quality of the optical flow super-resolved image increases most significantly within the first 5 to 10 frames as shown in Figures 4.23(b) and 4.23(c). This corresponds to the number of frames most optical flow based super resolution methods use (Baker and Kanade, 1999). The addition of more frames results in more blurred and noisy images, as estimating an accurate dense flow field across a large number of frames is difficult and erroneous, especially in low-resolution images. This is clearly visible in Figures 4.23(e) and 4.23(f). Estimating a dense optical flow field across 50 to 100 frames is erroneous and the calculation of the mean across such a large number of frames results in artefacts and distortion.

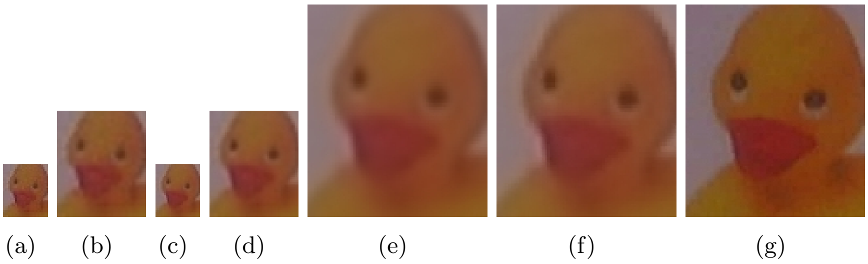


Figure 4.24: (a) Cropped original frame with a cube of 100×100 pixels, (b) result after applying bilinear interpolation, (c) result of the optical flow after 10 frames without interpolation, (d) result of the optical flow after 10 frames with interpolation, (e) result of the proposed approach after 20 frames with interpolation (f) result of the proposed approach after 20 frames without interpolation, and (g) cropped cube of size 400×400 pixels.

Figure 4.24 summarises the results of both methods. The original video is recorded at a resolution of 320×240 pixels and the cube consisting of 100×100 quads. A single cropped frame is shown in Figure 4.24(a). The simplest way of increasing the resolution is by interpolation and the result of bilinear interpolation is shown in Figure 4.24(b). However, interpolation cannot recover high-frequency details and in addition introduces artificial random noise. Therefore, the super resolution optical flow algorithm without an initial resolution increase by interpolation is applied. The result after taking the mean across 10 frames is shown in Figure 4.24(c). The quality of the image is improved, i.e. the image noise is reduced, but the resolution remains unchanged because no interpolation is applied. Interpolation is needed to increase the resolution, thus Figure 4.24(d) shows the result of the optical flow using bilinear interpolation to double the resolution of the input frames.

The proposed combined tracking and super resolution approach is applied to the same video sequence that has been doubled in size using interpolation. The resulting super resolved image (Figure 4.24(e)) is more blurred and shows less detail as a result of the interpolation, compared to the super-resolved image in Figure 4.24(f) that is calculated from the original input sequence. Furthermore, the subdivision of the 3D model into a fine mesh allows for a greater increase in resolution compared to optical flow based methods. Figure 4.24(g) shows a cropped image of the cube of size 400×400 pixels, which equals a resolution increase of 400% compared to Figure 4.24(a).

4.6.3.1 Super resolution of Faces in Surveillance Footage

The proposed approach is tested on non-planar and non-rigid objects, in this case that of surveillance video of six people entering a bus. One video per persons is recorded with a resolution of 640×480 pixels at 23 frames per second (due to dropped frames). Each frame is sub-sampled to half the resolution resulting in 320×240 pixels. The face within one frame is about 32×25 pixels, a single cropped frame of each person is shown in Figures 4.25(b) to 4.30(b). The combined geometric and appearance based approach is used to track these faces across 30 to 50 frames.

The use of the extended expression tracking approach is omitted because it is assumed that people have a neutral expression when entering the bus. Furthermore, the low resolution of the face does not justify the runtime overhead of the proposed expression tracking as shown in Section 4.6.1.2. If expressions occur during tracking the tracking error will increase, but as the super resolved image is created using only frames below a threshold ϵ (Equation 4.8), these expression frames will not affect the result.

Optical flow is feasible for tracking planar objects, like the front side of a cube in the last experiment, but tracking non-planar and non-rigid objects like faces poses challenges, especially when trying to estimate a dense flow field across a large number of frames. The result of the optical flow combining only 5 frames is shown in Figures 4.25(d) to 4.30(d). In most cases the feature tracker (Shi and Tomasi, 1994) was not able to estimate a precise flow field, especially for Persons 3 and 9 in Figures 4.25 and 4.29 who walk quickly towards the camera. The same is true for Person 7 in Figure 4.26, who turns while approaching the camera. A false flow field then results in the erroneous warping of frames into a reference coordinate system which causes artefacts in the super resolved image as visable in Figure 4.25(d), 4.29(d) and 4.26(d). Figures 4.25(c) to 4.30(c) show the initial frames after they have been doubled in size using bilinear interpolation.

A comparative result of the combined tracking and super resolution method after combining 20 frames is shown in Figures 4.25(e) to 4.30(e). The super-resolved faces are less blurred and show more detail compared to the result of the optical flow. The optical flow based approach uses interpolation to increase the resolution but this introduces artificial random noise. However, the proposed method achieves the same or a slightly higher resolution increase after combining 20 frames without interpolation.

Another advantage of the proposed super resolution approach is the further use of the created super-resolved 3D model. In the case of faces, these models can be used to generate

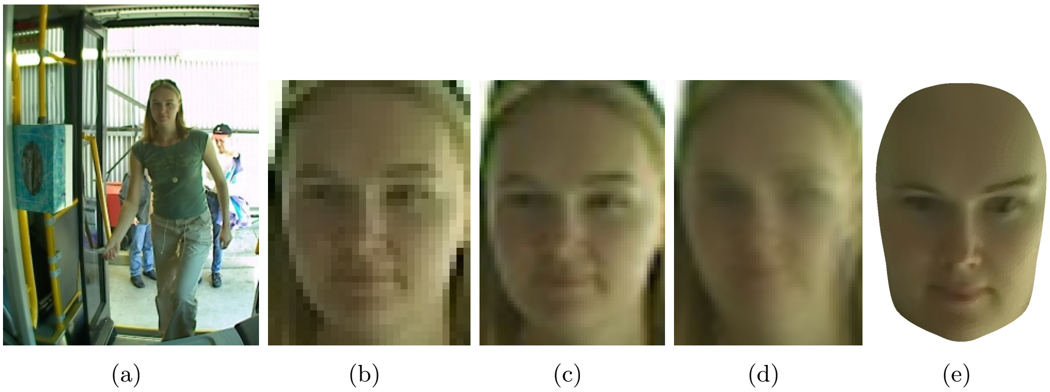


Figure 4.25: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

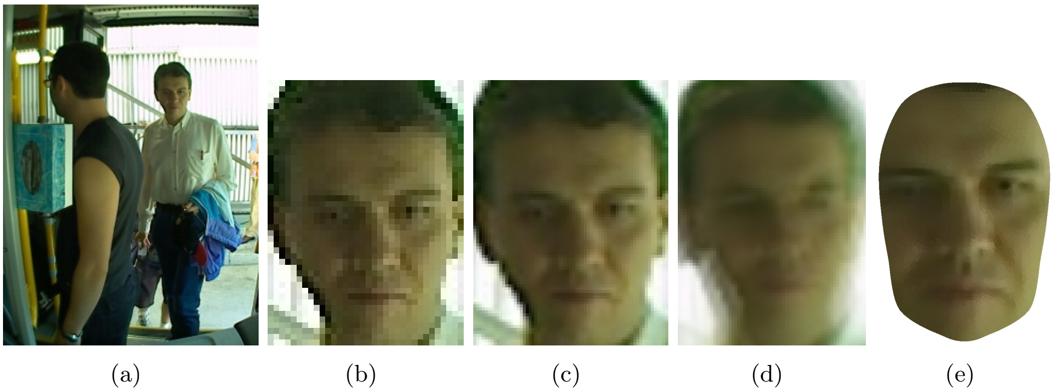


Figure 4.26: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

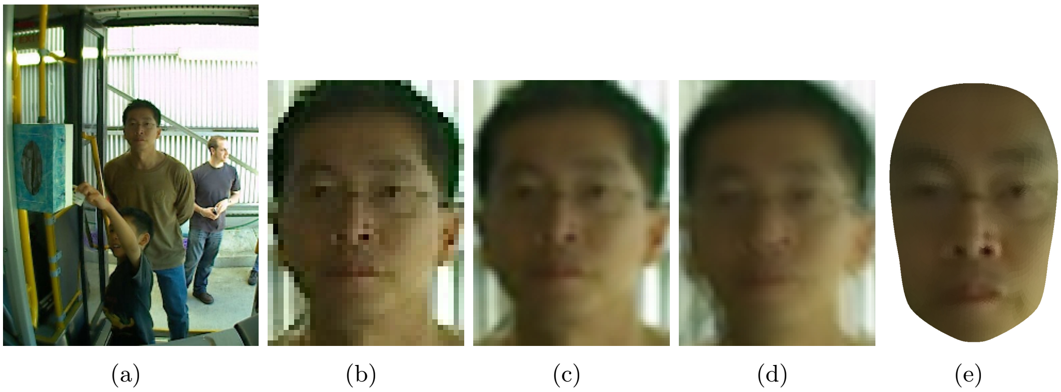


Figure 4.27: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

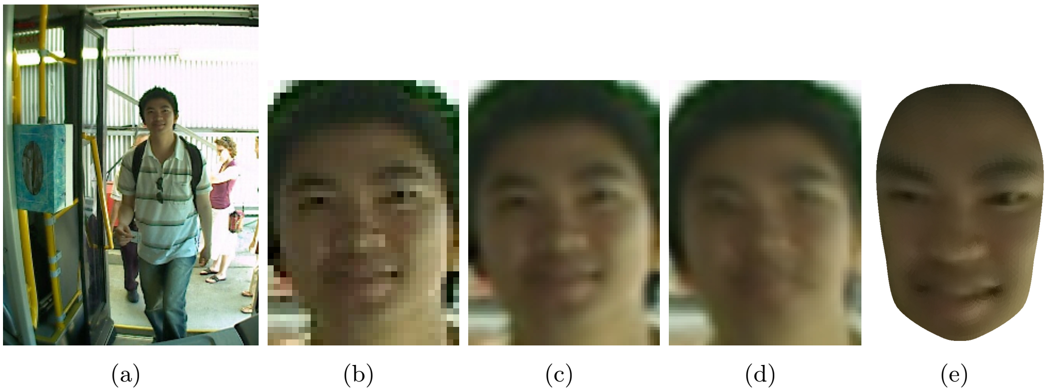


Figure 4.28: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

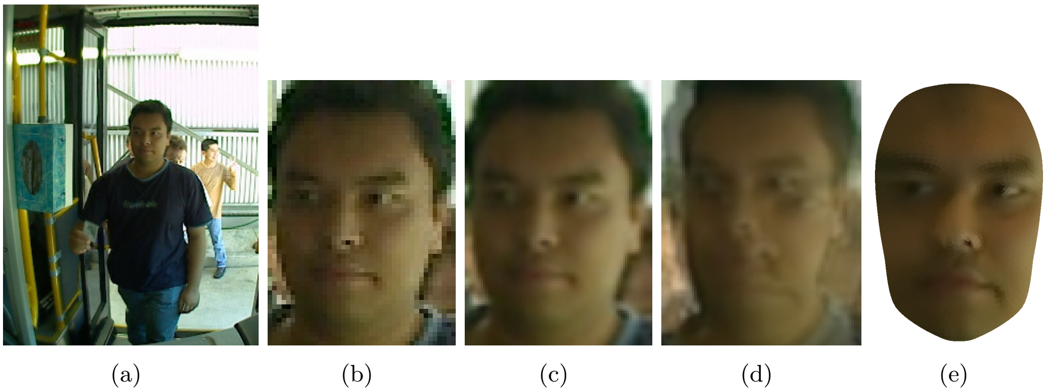


Figure 4.29: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

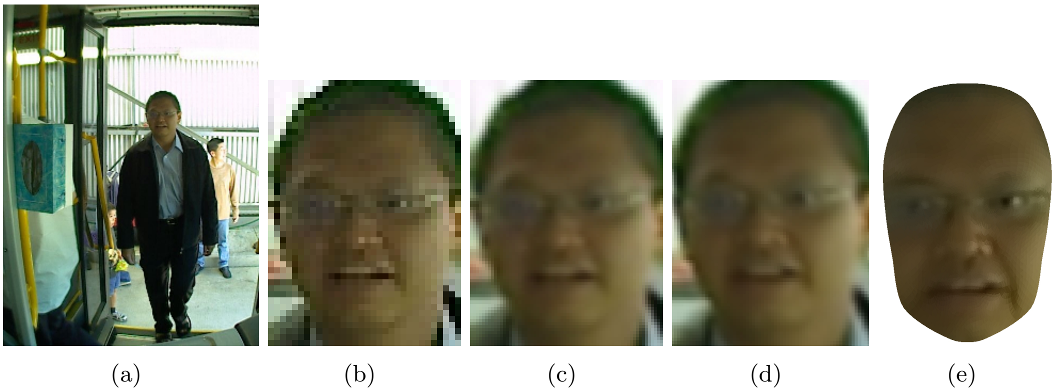


Figure 4.30: (a) Original image, (b) cropped face, (c) bilinear interpolation, (d) optical flow result after combining 5 frames and (e) the result of the proposed tracking and super resolution approach after combining 20 frames.

various face images under different pose and lighting in order to improve the training of classifiers (Hu *et al.*, 2004; Lu *et al.*, 2006) or the super resolved 3D model itself can be used for 3D face recognition (Scheenstra *et al.*, 2005).

4.7 Conclusion

This chapter proposed a combined tracking and super resolution algorithm that increases the resolution simultaneously during the tracking process. An object-specific 3D mask mesh is used to track non-planar and non-rigid objects. This mask mesh is then subdivided such that every quad or triangle is smaller than a pixel when projected into the image. This makes super resolution possible and in addition improves tracking performance. This approach varies from traditional super resolution as the resolution is increased at the mask level and only for the object of interest, rather than on an image level and for the entire scene.

Experiments on sequences of different size faces demonstrate that the combined geometric and appearance based tracking approach is able to track faces down to 28×20 pixels in size. The combination of these two tracking algorithms achieves better results than each method alone. The appearance-based method tends to be more precise for small inter-frame movements, whereas the geometric method is better for larger displacements. Furthermore the tracking performance increases with the number of quads or triangles per mask mesh. This way of tracking is not only necessary for creating the super resolution image, but also benefits from the super resolution process.

The proposed 3D model based super resolution algorithm allows for a high increase in resolution, the finer the 3D mesh, the higher is the possible increase in resolution. Therefore, the number of frames needed to achieve a certain resolution increase is empirically estimated. In practice about 20 to 30 frames are needed to double the resolution. Increasing the resolution further is limited by the number of frames needed and the tracking error. Large tracking errors as well as averaging across a large number of frames introduces noise that is difficult to remove subsequently.

The proposed tracking and super resolution method is tested on low resolution video of faces that are acquired both in the lab and in real surveillance situations. It is shown that it outperforms the optical flow based method, and performs consistently better for longer tracking durations in video that contain non-planar and non-rigid low-resolution objects. The resolution increases significantly within the first 20 to 30 frames and gives excellent super-resolved images. This differs from the use of a smaller number of frames (say 5-9) by optical flow methods, which are unable to take more frames because of large motion deviation.

The proposed approach needs three to four times more images to achieve the same resolution increase compared to optical flow based methods. Super resolution optical flow methods increase the resolution through interpolation in the first step unlike the proposed approach, which avoids interpolation. The resulting super-resolved 3D model is less blurred by achieving the same or a higher resolution increase. This in turn makes deblurring, the last step of the optical flow algorithm unnecessary. Furthermore the super-resolved 3D model is created online during tracking and improves with every frame, whereas super resolution optical flow incorporates consecutive and previous frames which prohibits its usage as an online stream processing algorithm. However, the resulting super resolved image depends on the quality of the tracking, large tracking errors will result in additional noise or even artefacts that decrease the super resolution quality.

The novel super resolution approach proposed in this chapter is based on the effect of the image formation process as described in Section 4.1. The appearance of the low resolution images depends on the position on the imaging chip and varies with movement. This effect is further utilised and used for recognition in the following chapter.

Chapter 5

Model Based Low Resolution Character Recognition

Low resolution character recognition is a problem especially in large scale surveillance situations where the distance between the camera is large and the resolution of the video is poor. The most common examples are car license plate recognition or text recognition in surveillance footage. Additionally, text documents captured by mobile devices require methods for character recognition at low resolution.

Based on the effect of the image formation process described in the previous chapter and assuming a simple camera model, the key idea of this chapter is to use this effect for recognition of objects in low-resolution images given a priori knowledge of the particular object. The objects this chapter focuses on are low resolution characters with applications to number plate and text recognition.

Existing methods for detecting number plates in low resolution images use low level image features (Zheng *et al.*, 2005) or morphological operations (Wu *et al.*, 2006). However, when trying to segment and recognise low resolution characters the following problems arise. Low resolution characters tend to merge along their edges with the next character, making it very hard to separate them. Instead of clear sharp character edges only an amalgamation of aliased pixels is visible. Text document recognition faces similar problems when recognising words in low resolution images. Merging characters make it hard to separate the word into single letters and standard Optical Character Recognition (OCR) methods will fail.

Most existing OCR methods need a minimum image resolution of 300dpi (S. Rice and Nartker, 1996), which means that a scanned A4 page with font size 10 will result in characters of at least 30 pixels in height. Unlike current OCR methods, the proposed approach is suitable for character heights as short as five pixels. However, low resolution characters of less than 20 pixels in height do not possess clear edges and thus make low resolution character separation and subsequent successful recognition a challenging task.

Another disadvantage of standard OCR methods is their limitation to binary images. After each character is separated, it is converted into a binary image, but low resolution images of characters are highly dependent on the threshold used to generate the binary image. Depending on the chosen threshold, they may result in degraded binary images that are hard to recognise. Furthermore binary images lose the grey scale information which is necessary for low-resolution character recognition.

This chapter proposes a new method for recognising characters in low resolution with characters down to five pixels in height. The image formation process is synthesised assuming a simple camera model and given a priori information about each character, this model is used to generate several low resolution images, i.e. templates of each character. Instead of segmenting the image into single characters, the proposed approach recognises the word as a whole by matching the aforementioned templates. Normalised cross-correlation is used to find the best position of each character template within the word and thus a separate character segmentation step is unnecessary. The proposed method works directly on grey scale images and thus avoids any information loss through binarisation, which is required for OCR methods. Also image enhancement methods like super resolution are unnecessary since they may create artefacts that could decrease the recognition performance. Recognition experiments are conducted on low resolution number plates down to 30×8 pixels in size as well as text documents with characters down to five pixels in height.

This chapter is organised as follows: The proposed method for recognising low resolution characters is introduced in Section 5.1, including the template generation process in Section 5.1.1 and the word recognition algorithm in Section 5.1.2. Experimental evaluation are presented in Section 5.2. Recognition is performed on number plates in Section 5.2.2 and on text documents on Section 5.2.3. The chapter concludes with Section 5.3 including a discussion about model based face recognition.

5.1 Model Based Character Recognition

The image formation process is important for understanding low resolution images. As outlined previously in Chapter 4.1, the appearance of the low resolution image depends on the size of the object, the distance between the object and the camera, the optical system and the number and size of image pixels. The appearance also changes with the movement of either the camera or the object itself. This effect is used by the proposed recognition algorithm and the basic overview is shown in Figure 5.1.

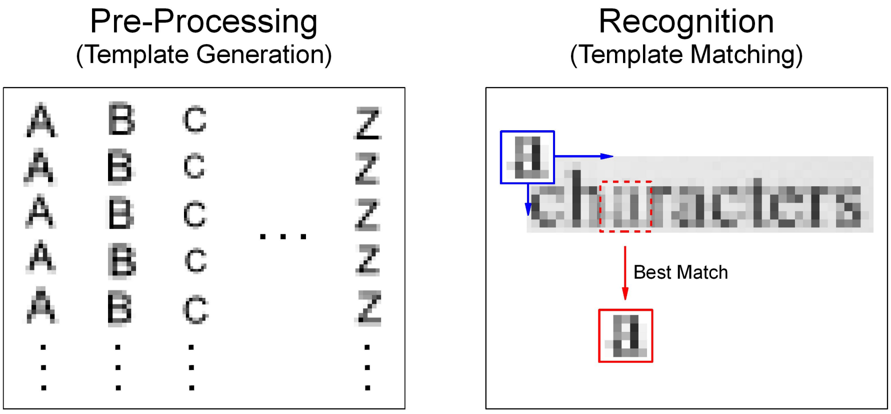


Figure 5.1: Model based character recognition. A set of templates for each character is generated in the pre-processing step. The recognition step uses these templates to identify the best matching character for each position within the word, thus character separation is unnecessary.

By modelling a simple camera and assuming a priori knowledge of the characters and the font type, a large set of degraded low resolution appearance for each character is created. The template generation method is based on the fact that the appearance of the low resolution character image changes with certain parameters.

This pre-processing step creates a set of different templates for each character before the actual recognition. A template matching approach is used to identify characters and calculate their position within the word simultaneously. The proposed method does not require separated single characters; instead normalised cross-correlation is used to find the best matching template for each position. Both the pre-processing and the recognition steps are explained in detail in Section 5.1.1 and Section 5.1.2 respectively.

5.1.1 Template Generation

The image formation process as described in Section 4.1 is used to model the formation of low resolution images by the camera. When the high resolution object is captured by the camera, its image is warped, blurred due to the optical system of the camera as well as motion blur and it is down sampled as a result of the finite imaging chip.

The proposed template generation method assumes a perfect imaging chip, possible motion blur is neglected and the image warp is constrained to translations in the 2D plane. These are realistic constraints for recognising characters in images parallel to the camera, i.e.

the image plane. If blur, in particular motion blur is an issue, existing methods can be used for deblurring. For example, the authors of Agrawal and Raskar (2007) use a special camera to create a high resolution image from a single motion blurred image.

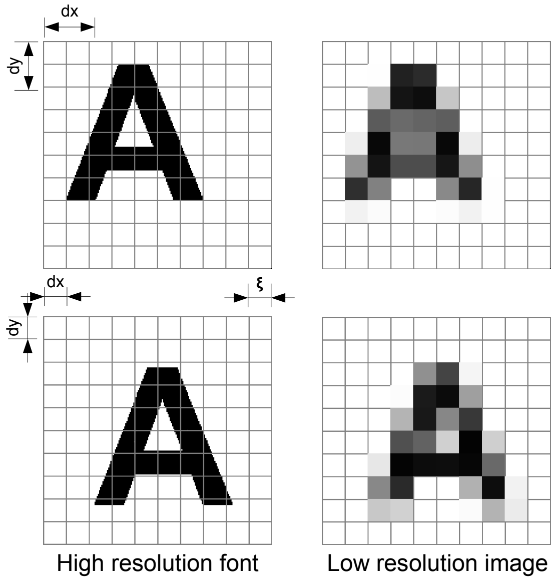


Figure 5.2: The template generation process. For each character a number of different low resolution appearances are generated by varying the parameters, dx and dy, the position on the image plane and ξ, the size of each pixel.

The template generation process is illustrated in Figure 5.2. For each character of each font type a high resolution image is acquired by using the associated TrueType definition[1]. Using the TrueType, each character can be rendered at any required high resolution.

For each character $c = \{\{A..Z, a..z\} \cup \{0..9\}\}$ of a particular font type, one high resolution image is created. The size of these high resolution images is H×W, where H defines the height and W the width of the image. Each high resolution character image c is used to create a number of different low resolution templates C_c^r by down sampling the high resolution image c as:

$$C_c^r = \mathcal{T}(c, dx, dy, \xi) \tag{5.1}$$

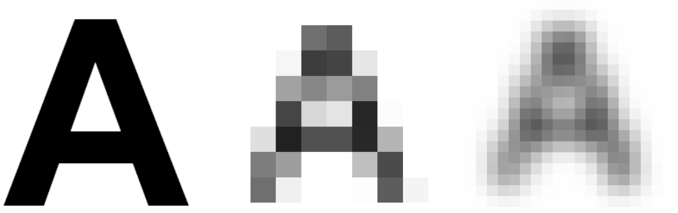
where the function T generates the low resolution template C_c^r from the high resolution image of character c by using the parameters dx, dy and ξ. The integer number r iterates over all low resolution templates of the same character c.

The parameters dx and dy describe the translation displacement along the horizontal x-

[1] TrueType is a standard for fonts that describes the outline of each character as a vector graphic.

axis and the vertical y-axis of the imaging chip respectively, image rotations are neglected to reduce the number of possible low resolution templates. The parameter ξ defines the pixel size, which is the number of high resolution points that fit into a single low resolution pixel. It also indirectly determines the size of the resulting template. A high resolution image of size H×W will result in an template of size $\frac{H}{\xi} \times \frac{W}{\xi}$ for a particular ξ. The parameters dx, dy and ξ are all integer multiples and r indexes all possible combination of these three parameters, with $dx \leq \xi$ and $dy \leq \xi$.

The function $\mathcal{T}$ is realised by overlaying a high resolution image of a character c with a grid representing the imaging chip as shown in Figure 5.2. The size of the grid is set by ξ and the parameters dx and dy are used to position the character image on the simulated imaging chip. Each square of the grid represents one low resolution pixel. The grey value of each pixel is then defined as the percentage of its coverage. Using these three parameters the appearance of the resulting low resolution character template can be altered as shown in Figure 5.2. These templates are generated off-line only once for different characters and font types.



(a) High Resolution (b) Low Resolution (c) Interpolated

Figure 5.3: Image enhancing methods may alter the appearance of low resolution images to something that is not modelled by the proposed approach. (a) shows the high resolution image of character 'A', (b) shows the low resolution image of character 'A' and (c) shows the enhanced low resolution image after B-spline interpolation.

The proposed template generation process models the down sampling process of the imaging chip. These templates are therefore only suited for images of characters that have not been further modified after capturing. Most image enhancing methods, like super resolution techniques, will change the appearance of the low resolution character to something that is not modelled in the proposed template generation process. Such modified images are not suitable for this approach as shown in Figure 5.3.

5.1.2 Low-Resolution Word Recognition

The main problem of low resolution word[2] recognition are merged characters. As the resolution decreases, the characters of each word begin to merge and instead of clear separable white gaps between black characters, the whole word connects, making single character separation an almost impossible task.

The proposed low resolution word recognition approach, therefore uses a parameterised template matching based algorithm for combined character detection and recognition. Instead of separating the characters first and applying optical character recognition methods on the separated characters, the proposed method performs recognition on the whole word in a single step.

Given an input image I that contains a low resolution word, R different possible low resolution templates C_c^r are created for each character c according to Equation 5.1. It is assumed that the input image is cropped using existing pre-processing methods, such that the height of the input image can be used to estimate a rough range of the parameter ξ which determines the size of the low-resolution template, the bigger ξ the smaller the resulting template. It is also assumed that the font type of the word is known, and this further reduces the number of possible matching character templates.

The generated templates as described in the previous section are then used to perform character separation and recognition simultaneously. Therefore, each character template C_c^r is matched at every possible image position $\{x, y\}$ using normalised cross-correlation:

$$z_c^r = \underset{\{x,y\}}{\operatorname{argmax}}\{ncorr(C_c^r, \{x, y\}, I)\}, \tag{5.2}$$

where I is the cropped image containing a single word and $ncorr$ calculates the normalised cross-correlation of template C_c^r and the image I at every 2D image position $\{x, y\}$. Therefore z_c^r is the 2D image position of character template r for character c that achieves the highest correlation value.

After the image position with the highest correlation value for each character template C_c^r is calculated according to Equation 5.2, a global order amongst all these templates is created with respect to their correlation values. This set of possible character templates is reduced by deleting variations with correlation values below a certain threshold, empirically defined to be 0.85.

[2] The term 'word' is used to describe a string of alphanumeric characters contained in an image.

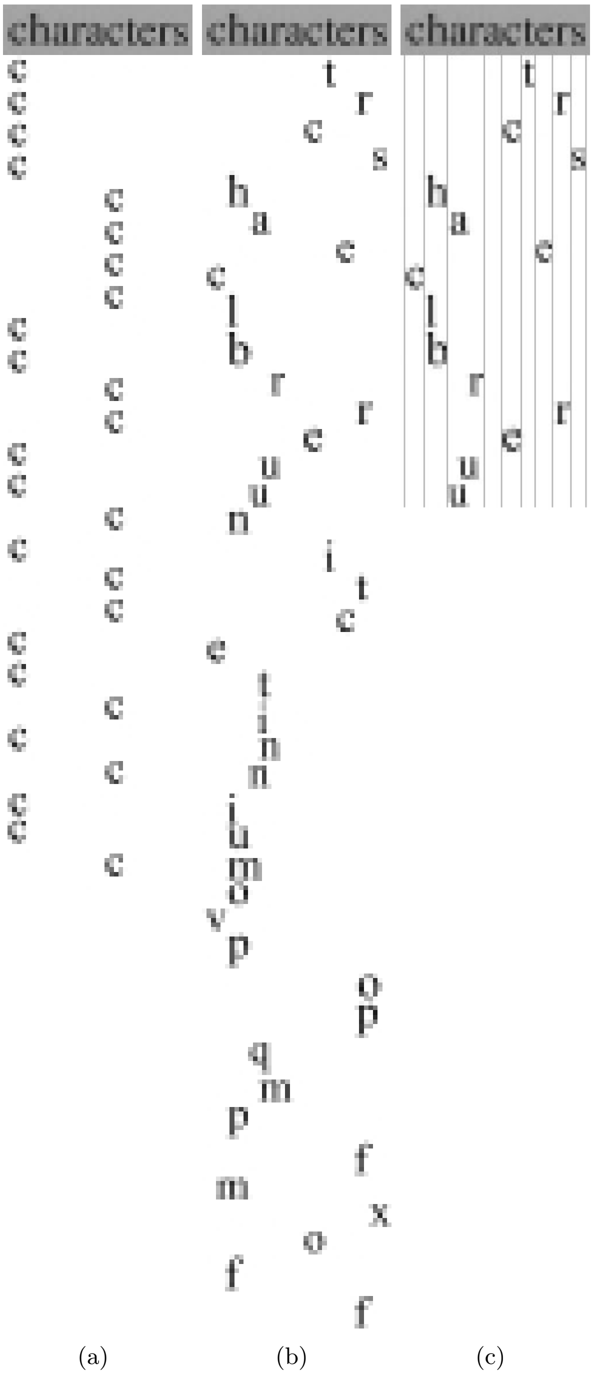


Figure 5.4: (a) Initial result after the best position is calculated for each low resolution character template according to Equation 5.2. The templates are ordered with high correlation values on top. (b) Reduced set after clustering leaves only a single template for each character *c* at one image position. (c) The top characters are chosen and again clustered to identify the best matching character for each image position

Figure 5.4(a) shows the result after these initial steps for the word 'characters' in low resolution. For all character templates C_c^r with $c =' c'$ and $r = 1..26$, the best position within the image is shown. These templates are also ordered according to their correlation values, with the highest value at the top. This redundancy of character templates at identical image positions is reduced by applying clustering techniques. The MATLAB function 'clusterdata' is used to cluster all matching low resolution templates of a particular character c with respect to their position along the horizontal image x-axis. Once the clusters are found, only the character template with the highest correlation value in each cluster is considered for further processing. This means that at each image position x, only a single instance of the character c is kept and all other appearances of the same character at the same position are deleted as shown in Figure 5.4(b).

From the remaining character set, only the character templates that correspond to the highest correlation values are considered for further processing. Therefore, starting from the top, character templates with high correlation values are chosen until all image columns are covered by a character template. In Figure 5.4(c), the character templates 't', 'r', 'c', 's', 'h', 'a', 'e' and 'c' are chosen in descending order. This selection stops after the last κ added templates do not cover additional image columns. In the example in Figure 5.4(c), $\kappa = 4$. The last four character templates that are selected additional to the characters mentioned before are 'l', 'b', 'r' and 'r'. The image columns covered by these four characters are already covered by previous ones, thus no more character templates are added.

In the last step of the proposed algorithm the selected character templates are clustered into non-overlapping areas. Hierarchical clustering is used to find column sets consisting of one or more characters. These sets are then used to select the character that best fits each image position. The cluster bounds are shown in the example in Figure 5.4(c)(vertical lines), with the cropped low resolution image on top and the synthesised character templates below.

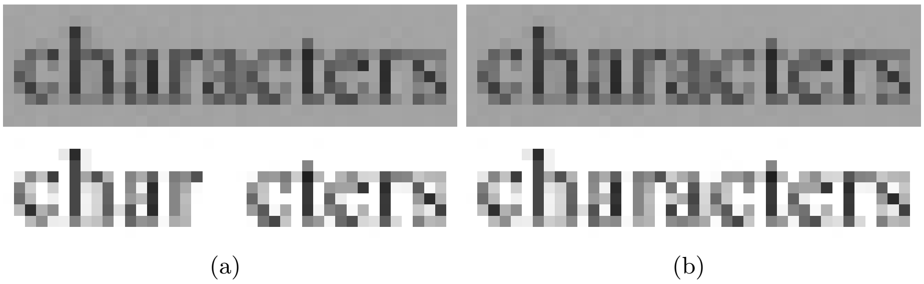

(a) (b)

Figure 5.5: The result of the proposed character recognition approach. (a) shows the result after the first run. (b) the final result after all characters within the image are recognised.

After the column sets are bounded, the average correlation values for all valid character permutations are calculated. A possible permutation is a set of one or more character templates that cover all image columns in the column set. In the example in Figure 5.4(c), the third column set consists of the character templates {'a', 'r', 'u', 'u'} and possible permutations for this set are {'a', 'r'} or {'u', 'r'} while the permutation {'u', 'u'} does not cover all image columns of this cluster. For each column set, the character templates of the permutation with the maximum correlation value are selected. The result is shown in Figure 5.5(a). Please note the gap with the missing character 'a'.

In a post-processing step, image gaps of characters that are not yet identified are detected and cropped. These cropped image gaps are used as a new input image I for the proposed character recognition algorithm and recognition starts again until all gaps are filled with a matching character. The final result is shown in Figure 5.5(b). These gaps are not common and are most likely to occur in words with several identical characters where all templates of this particular character are initially matched at the same image position. These gaps can also result from cutting off too many character templates in the reduced set (Figure 5.4(c)).

This heuristic template matching approach requires a large set of templates for each possible character and font type. However, this set is reduced by estimating the rough range of ξ using the input image height and by assuming that the font type is known. Furthermore, the use of normalised cross-correlation is practicable since each template is only a few pixels in size and so is the low resolution input image I. Parallel programming can also be used to implement the template matching step of the proposed character recognition algorithm and thus several templates can be matched simultaneously to improve the runtime. A further speed up in performance could be achieved by using a Field-programmable gate array (FPGA) hardware implementation.

5.2 Experiments

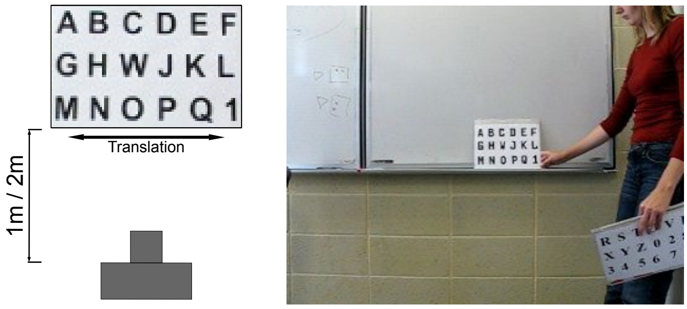
The following subsections describe the results of the experiments that are conducted in order to demonstrate the performance of the proposed recognition algorithm for low resolution characters. Most experiments use images or videos recorded with a compact digital camera of resolution 320×240 pixels unless otherwise stated. The camera automatically compresses and stores them as JPEG images. These images and video frames are cropped and used directly as input images without any pre-processing.

The first experiments in Section 5.2.1 analyse the suitability and interchangeability of different font types for the use with the proposed character recognition algorithm. Following this, the proposed approach is applied to number plate recognition, both indoors and outdoors. Section 5.2.3 demonstrates the use of the proposed approach for text recognition.

5.2.1 Suitability and Interchangeability of Different Fonts

The proposed character recognition approach as described in Section 5.1 uses a high resolution image of each character that is down-sampled according to three parameters dx, dy and ξ and subsequently used for template matching. The following experiment measures the suitability and interchangeability of different fonts when used for recognition, namely Arial, Times New Roman, and the fonts used for standard Western Australian number plates and standard German number plates.

The high resolution images for the template generation of the Arial and Times New Roman font are created from their TrueFont definitions available on any Windows machine. For the Western Australian number plate font, high resolution images of each character are captured manually from different licence plates. These images are normalised in size and converted into binary images. The high resolution images of the German number plate font are created from a TrueType definition created by an artist and kindly made available at Anke-Art[3].



(a) (b)

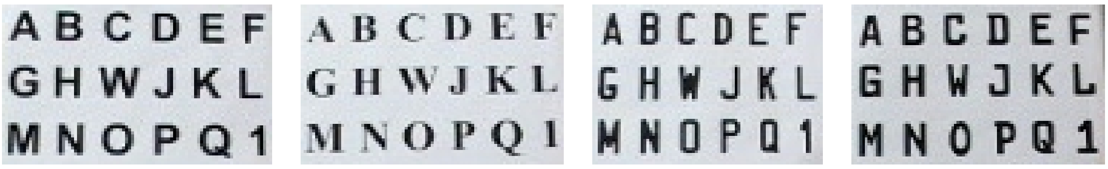
Figure 5.6: The experimental setup. The camera is placed about 1m and later about 2m in front of A4 sheet of paper with printed characters. Several images are recorded while the paper is translated horizontally.

The experimental set up for this experiment is shown in Figure 5.6. For each font type, each of the 36 characters (26 letters and 10 numbers) are printed on a piece of cardboard

[3] http://www.anke-art.de/home/?p=62\&lang=en

using a bold font size of 130 for Arial and Times New Roman and equivalent font sizes for the Western Australian and German number plate font. These cardboards are placed in front of a camera, such that the cardboard is parallel to the imaging chip of the camera. To achieve variations in the grey value appearance of each character, the position of the cardboard is manually translated, while several images are recorded at a resolution of 320×240 pixels. This experiment is carried out at two different distances (1m and 2m), resulting in different character sizes. The following two subsections describe the results of each experiment.

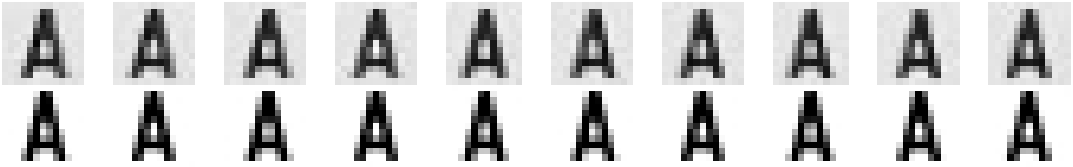
5.2.1.1 Medium Resolution - 1m Distance



(a) Arial (b) Times (c) Australian (d) German

Figure 5.7: Example images of (a) Arial, (b) Times New Roman, (c) Western Australian number plate font and (d) German number plate font printed on cardboard which is placed at a distance of 1m in front of a camera with a resolution of 320×240 pixels.

The four different character fonts are printed on cardboard and placed 1m in front of the camera. The recorded images have a resolution of 320×240 pixels that results in an average character height of about nine pixels for each character. Example images for each font, Arial, Times New Roman, Western Australian number plate font and German number plate font are shown in Figure 5.7.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 5.8: Cropped frames of the sequence showing Western Australian number plate fonts (Figure5.7(c)). Cropped letter 'A' (top) and the artificially created template that fit best (bottom). The distance to the camera is 1m.

As the cardboard is translated in front of the camera the resulting character images change their appearance according to the proposed model introduced in Section 5.1.1. Several different cropped example images of the letter 'A' of the Western Australian number plate

	Arial	**Australian**	**German**	**Times**
Arial	709 (**98.47%**)	276 (38.33%)	351 (48.75%)	561 (77.92%)
Australian	282 (39.16%)	715 (**99.31%**)	491 (68.19%)	251 (34.86%)
German	490 (68.05%)	440 (61.11%)	720 (**100%**)	361 (50.14%)
Times	432 (60.00%)	120 (16.66%)	203 (28.19%)	684 (**95.00%**)

Table 5.1: Recognition results of four different font types, using four different fonts for recognition respectively. Each row represents one particular model font used for recognition and each column denotes the set of images to the test font. The results are given in percentage of the number of correctly recognised images, where 720 is the maximum number (20 frames with 36 characters each).

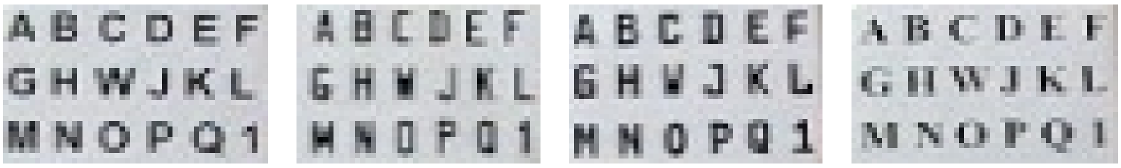
font are shown in Figure 5.8(top row). Note that all these images differ from each other slightly, because the cardboard is translated slightly during the recording.

The algorithm as described in Section 5.1.2 is then used to determine the character template that fits these images best. Therefore, each character image is cropped manually before it is used as input image I for the proposed algorithm. The result is shown in Figure 5.8, where the bottom row shows the generated character templates that matched the above image best. Even though the recorded images of the letter 'A' are effected by image noise and compression from the camera, the matched templates in Figure 5.8(bottom row) model the original appearance very well, achieving an average correlation value of 0.98.

For comparing the suitability and interchangeability of different font types, 20 images of each character of each font type are recorded and cropped. Each of the four font types, Arial, Times New Roman, standard Western Australian number plate font and standard German number plate font, are then used to recognise these fonts. For example, Arial font templates are used to recognise Arial font as well as the Times New Roman font and the two number plate fonts. This experiment is set up to identify how the choice of model font type impacts on the recognition performance over these fonts. The result is shown in Table 5.1, where each row represents one particular type of model font used for recognition and each column denotes the set of 720 cropped images (20 images of each of the 36 characters) belonging to the test fonts.

The main diagonal of this correlation matrix achieves maximal recognition rates as every character is best recognised using the matching font type for recognition. Times New Roman appears to be the font that is most difficult to recognise, achieving only 95 %. This is mainly due to its serif style that results in characters that appear less bold compared to sans-serif font types of the same size. As a result serif fonts usually consists of less black and grey pixels compared to a similar non-serif font of same the size, making them harder to recognise. This is also reflected in the very low recognition rates for Western Australian (16.66%) and German (28.19%) number plate fonts using Times New Roman templates for recognition, as Times New Roman is the only serif font type used. In general all sans-serif fonts (Arial, Western Australian and German number plate font) achieve better recognition rates when sans-serif fonts are used for recognition, compared to serif fonts (Times New Roman). Furthermore, fonts used for number plates are usually more narrow than standard computer fonts as shown in Figure 5.7, resulting in better recognition rates for the pair of Western Australia and German number plate fonts compared to using Arial and Times New Roman, which are both wider.

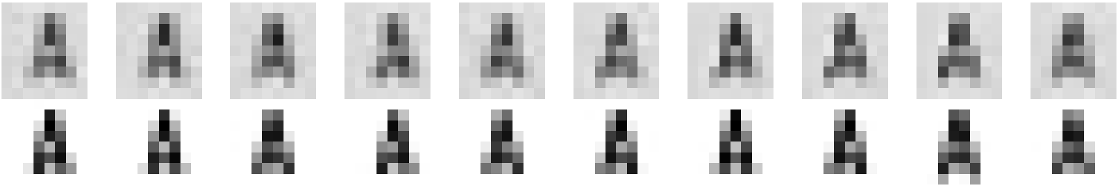
5.2.1.2 Low Resolution - 2m Distance



(a) Arial (b) Australian (c) German (d) Times

Figure 5.9: Characters of (a) Arial, (b) Times New Roman, (c) Western Australian number plate font and (d) German number plate font printed on cardboard which is placed in a distance of 2m in front of a camera with a resolution of 320×240 pixels.

In the second experiment, the distance between the camera and the cardboard is increased to 2m. Using the same image resolution of 320×240 pixels, the resulting characters are smaller in size with an average of six to seven pixels in height. An example image of each font type is shown in Figure 5.9.



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 5.10: Cropped frames of the sequence showing Western Australian number plate fonts (Figure 5.7(c)). Cropped letter ’A’ (top) and the artificially created template that fit best (bottom). The distance to the camera is 2m.

Figure 5.10(top) shows ten different cropped images of the letter ’A’ of the Western Australian number plate font. The bottom row in Figure 5.10 shows the artificially created template that fit best. Note that these images appear more blurred compared to the ones

	Arial	**Australian**	**German**	**Times**
Arial	706 (**98.05%**)	278 (38.61%)	342 (47.50%)	337 (46.81%)
Australian	305 (42.36%)	669 (**92.92%**)	434 (60.28%)	165 (22.92%)
German	382 (53.05%)	480 (66.66%)	702 (**97.50%**)	243 (33.75%)
Times	360 (50.00%)	140 (19.44%)	159 (22.08%)	641 (**89.03%**)

Table 5.2: Recognition results of four different font types, using four different fonts for recognition respectively. Each row represents one particular model font used for recognition and each column denotes the set of images to the test font. The results are given in percentage of the number of correctly recognised images, where 720 is the maximum number (20 frames with 36 characters each).

recorded at a distance of 1m in Figure 5.8. Seeing only a single cropped image of the letter 'A', it is very hard for the human eye to recognise the character since it is merely seven pixels in height.

Even though each character is no more than seven pixels in height, the recognition rates are still high as shown in Table 5.2. Only the recognition rate for Times New Roman, the only serif type, drops below 90%. Arial and the German number plate fonts seem to be best suited for low-resolution character recognition. Arial is a sans-serif font whose characters are all quite distinct, unlike the standard Western Australian number plate font, where the characters 'Q' and 'O' or '1' and 'I' look very similar in low resolution. The standard German number plate font is a so called FE-Schrift (short for 'fälschungserschwerende Schrift' - falsification-hindering script). The characters of this font are proportioned such that falsification, for example making an 'E' out of a 'F', is hindered. It also eases automatic number plate recognition as these proportions result in more distinct characters (Wikipedia, 2009).

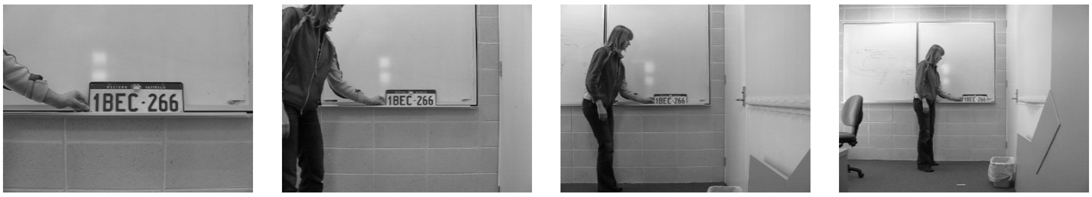
Summarising the results, the font type of the characters should be known beforehand in order to choose matching templates and achieve best recognition results especially in low resolution with character heights down to six pixels. Experiments show that non-matching fonts decrease recognition rates significantly especially at low-resolution (i.e. the non-diagonal results in Table 5.2). In cases where the exact font is not available for creating character templates, a very similar font should be used to achieve good recognition rates.

5.2.2 Number Plate Recognition

After the two different number plate fonts, namely the Western Australian number plate font and German number plate font, were tested with respect to their interchangeablity and suitability within the proposed character recognition approach, the following sections evaluate their usage for low resolution number plate recognition.

Section 5.2.2.1 presents number plate recognition in an indoor lab environment in order to test the low resolution limits of the proposed character recognition approach. The following Section 5.2.2.2 demonstrates the suitability of the approach for low-resolution number plate recognition in an outdoor setting.

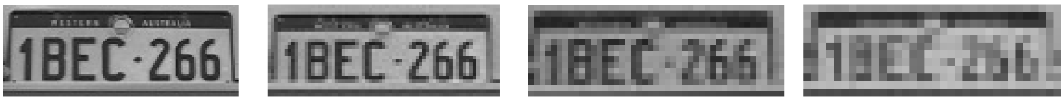
5.2.2.1 Indoor Number Plate Recognition



(a) 1m (b) 2m (c) 3m (d) 4m

Figure 5.11: Number plate recognition in an indoor environment. The distance between the plate and the camera varies from 1m to 4m. The resolution of the images is 320×240 pixels. Figures (a) to (d) show example images for each distance.

The experiments in this section are aimed to test the limits of the low resolution character recognition approach. Therefore, an Australian number plate is placed in front of a camera at various distances. An increasing distance between the camera and the plate will result in an decrease in resolution. The characters on the plate will start merging, making recognition very difficult. Example images are shown in Figure 5.11.



(a) 1m (b) 2m (c) 3m (d) 4m

Figure 5.12: Number plate recognition in an indoor environment. The distance between the plate and the camera varies from 1m to 4m. Figures (a) - (d) show cropped images for each distance.

The distance between the camera and the number plate is set to 1m, 2m, 3m and 4m. The resolution of the captured images is kept constant at 320×240 pixels resulting in average character heights of 23, 11, 7 and 5 pixels respectively. A cropped sample image of a number plate at each distance is shown in Figures 5.12(a) to 5.12(d). Note that the number plate at a distance of 4m in front of the camera is hardly recognisable by the human eye as shown in Figure 5.12(d).

	1m	2m	3m	4m
Number plate size [pixels]	120x40	64x22	41x13	30x8
Character height [pixels]	23	12	8	6
Recognised number plates	50 (100%)	50 (100%)	42 (84%)	26 (52%)
Recognised characters	350 (100%)	350 (100%)	340 (97.14%)	327 (93.40%)

Table 5.3: Summary of the number plate recognition results in an indoor environment. 50 images are recorded at each distance, resulting in 350 characters as the number plate consists of 3 letters and 4 numbers.

At each distance from the camera, 50 different images are captured by varying the horizontal position of the plate manually after each image. This allows for different grey scale appearances of each character in each image. The position of the number plate within the image is obtained manually and each image is cropped accordingly. The proposed character recognition approach for low resolution characters is then used to recognise the number plate without separating the individual characters beforehand. The recognition results are summarised in Table 5.3.

The number plates of size 120×40 pixels and 64×22 pixels, recorded at a distance of 1m and 2m in front of the camera respectively, are correctly recognised in each of the 50 images, consisting of three letters and four numbers and resulting in 350 characters in total. Thus, the character recognition rate as well as the number plate recognition rate is 100%. A number plate is considered recognised when every character on the plate is correctly recognised. The recognition rate drops down to 97.14% for the number plate of size 41×13 pixels at a distance of 3m from the camera and only 340 are recognised correctly. As a result the number plate recognition drops down to 84% as only 42 plates are recognised correctly.

The images recorded at 4m show characters with an average height of only six pixels (Figure 5.12(d)), making recognition with the human eye very difficult. However, the proposed approach is still able to recognise 93.40% of all characters across all 50 images correctly. The plate recognition rate drops down to 52% for number plates of size 30×8 pixels, mainly due to merging and blurred characters. The letter 'B' was often recognised as an '8' and the number '2' was mixed up with a 'Z'. However, such false recognitions can be avoided by using a regional syntax check after recognition. The syntax check for the Australian number plate for example would involve checking that the plate consists of seven characters of which the first character needs to be a number, followed by three letters and followed again by three numbers. Thus, regional syntax checks can avoid mix ups between letters and numbers.

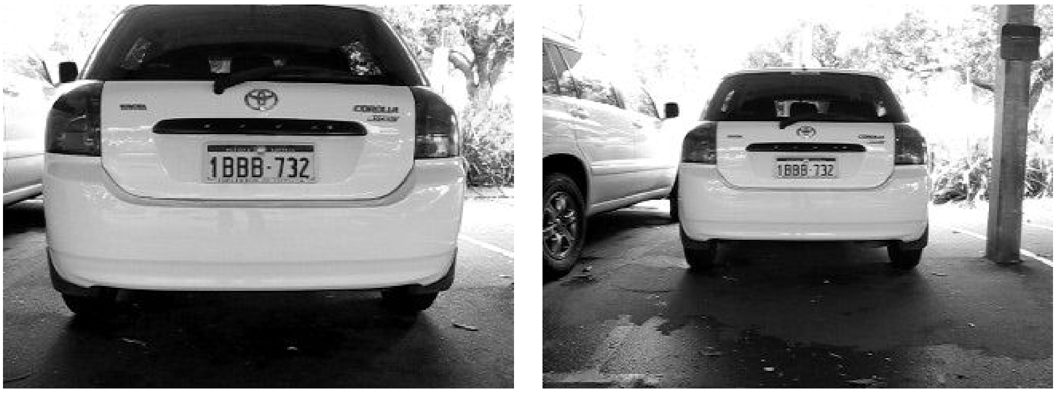
Increasing the distance between the camera and the number plate even further to 5m results in characters of only three to four pixels in height. Additionally, the blue edges of the number plate start to merge with the characters. These images are unrecognisable by the proposed method.

5.2.2.2 Outdoor Number Plate Recognition

After testing the limitations of the proposed character recognition approach in an indoor scenario, this section evaluates the number plate recognition performance on images recorded outdoor. However, the performance evaluation of number plate recognition approaches in the literature lacks uniformity due to missing ground truth datasets (Jung *et al.*, 2004). A first attempt towards a public dataset is made by Anagnostopoulos *et al.* (2008). They collected images of Greek number plates under different illumination conditions and under partial occlusion. However, their homepage (Anagnostopoulos *et al.*, 2009) is still under construction and no ground truth or recognition results are given.

In order to evaluate the proposed approach, a dataset is collected containing images of 35 cars with Western Australian number plates and 35 cars with German number plates, randomly picked in an outdoor car park. The following paragraphs describe the results of both experiments.

Western Australian Number Plate Recognition

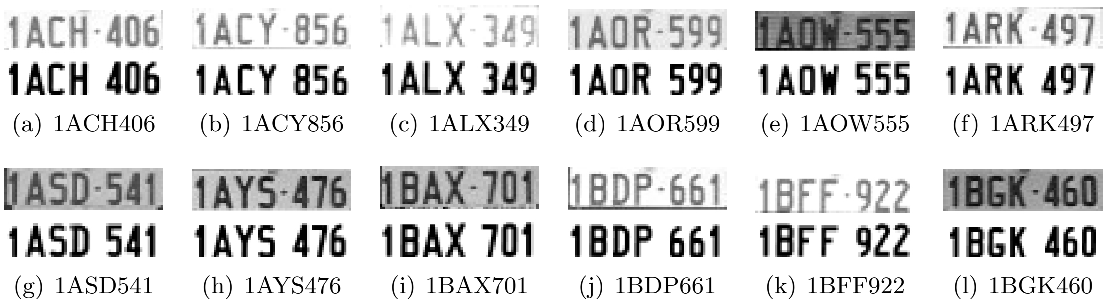


(a) 1.5m

(b) 3m

Figure 5.13: Example images of Western Australian number plates. The distance between the camera and the car is about (a) 1.5m and (b) 3m respectively. The resolution of the images is 320×240 pixels.

For capturing images of Western Australian number plates, a standard compact digital camera with a resolution of 320×240 pixels is used. The distance between the camera and the number plate is set to about 1.5m and 3m and the resulting average character heights for the Australian number plates amount to thirteen pixels and seven pixels respectively. An example image at each distance is shown in Figure 5.13. For each number plate, 20 slightly different images were recorded. These slight changes in appearance are achieved by moving the camera slightly during the recording.

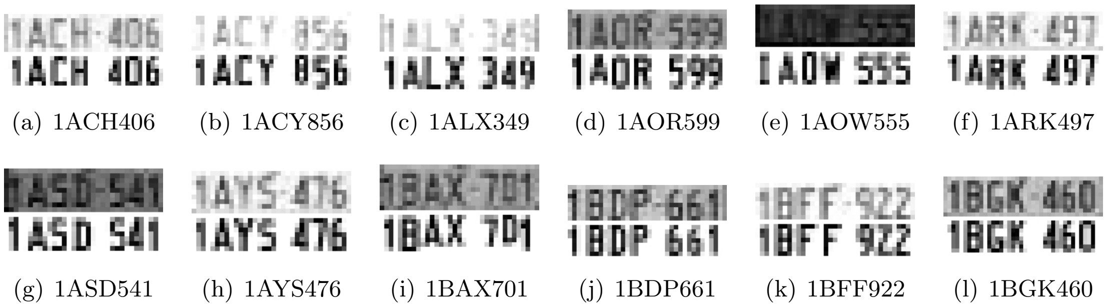


(a) 1ACH406 (b) 1ACY856 (c) 1ALX349 (d) 1AOR599 (e) 1AOW555 (f) 1ARK497

(g) 1ASD541 (h) 1AYS476 (i) 1BAX701 (j) 1BDP661 (k) 1BFF922 (l) 1BGK460

Figure 5.14: Cropped example images of Western Australian number plates used for recognition at a distance of 1.5m. Each sub figure shows the original image (top) with the result (below).

After recording, the position of each number plate in each image is determined manually. The total number of license plate images is 720, which is made up of 35 different number plates and 20 different images per plate. Each Western Australian number plate consists of seven characters (four numbers and and three letters), so the total number of characters is 5040. Figures 5.14 and 5.15 show cropped example plates at a distance of 1.5m and 3m respectively. Each sub figure shows the original image on top with the result below, that are the generated characters that best match each position within the image. Note that the cropped number plates differ in contrast and saturation depending on the lighting conditions as well as the plate background.



(a) 1ACH406 (b) 1ACY856 (c) 1ALX349 (d) 1AOR599 (e) 1AOW555 (f) 1ARK497

(g) 1ASD541 (h) 1AYS476 (i) 1BAX701 (j) 1BDP661 (k) 1BFF922 (l) 1BGK460

Figure 5.15: Cropped example number plates used for recognition at a distance of 3m. Each sub figure shows the original image (top) with the result (below).

Once the number plates are cropped, the proposed character recognition algorithm (Section 5.1.2) is used to recognise each character on the number plate without previous character separation. The position of each character is recovered simultaneously during recognition.

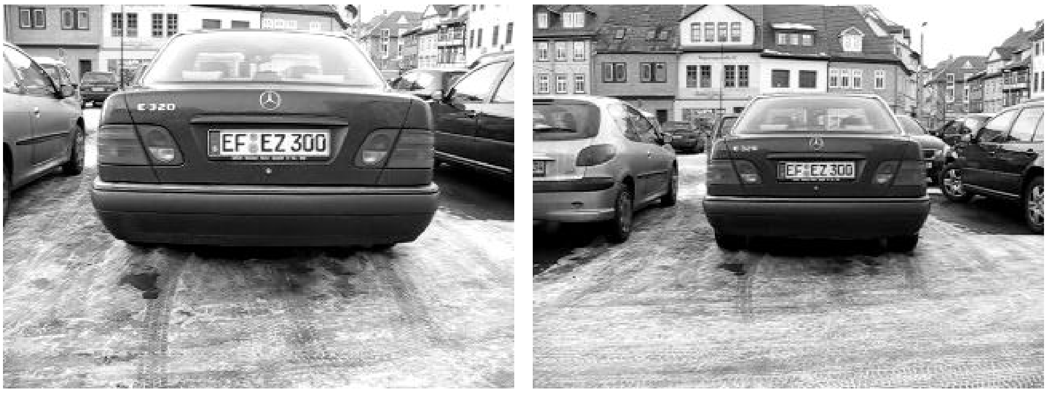
At a distance of 1.5m and a resulting character height of 13 pixels, 690 Western Australian plates out of 720 possible are recognised correctly, which equals 95.83%. A number plate is considered to be recognised if all characters on the plate are recognised correctly. From all possible 5040 characters (four numbers and three letters per plate), 5010 were identified correctly which equals 99.41%. By increasing the distance to 3m, the number of correctly recognised plates drops to 365, which equals 50.69%. Out of 5040 possible characters 4562 (90.52%) were identified correctly. Within the character recognition rate false positives are neglected. Especially in lower resolution, the dark surrounding edges of the number plate can be falsely recognised as the letter 'I' or the number '1'. These additional false positives are not taken into account.

The main reason for a false recognition are characters that fit into characters, like the 'U' fits the letter 'O', but both characters are of same height and width. The correlation value for the letter 'U' might therefore be slightly better than the correlation of the letter 'O', leading to false recognition. Furthermore, increasing similarities of the characters '0' and 'O' or '2' and 'Z' with decreasing resolution affect recognition, but such false classifications could be resolved by incorporating a regional syntax check. Lastly, as shown in Figure 5.14 and 5.15, the number plates also vary in contrast due to different lighting conditions in the car park. However, by using normalised cross-correlation to match the character templates the proposed method is robust against such lighting changes.

German Number Plate Recognition

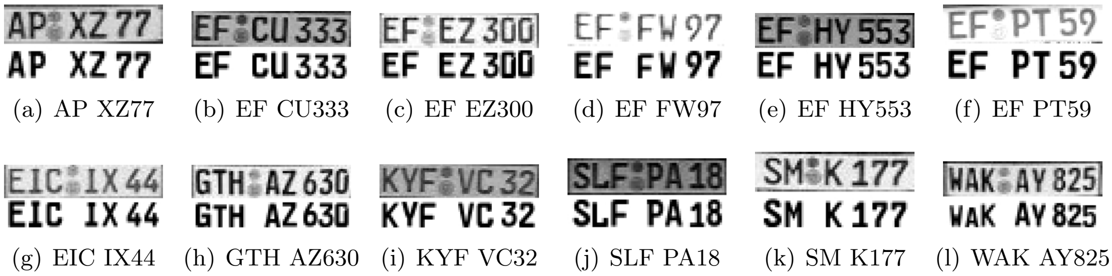
The same number plate recognition experiment was performed on 35 different German number plates, randomly picked in an outdoor car park. Again 20 different images of each number plate are recorded at 320×240 pixels by moving the camera slightly. The distance between the camera and the number plates is about 2m and 3m, with resulting average character heights of eleven and seven pixels respectively. An example image at each distance is shown in Figure 5.16.

Again each number plate is manually cropped and 12 example images are shown in Figure 5.17 and 5.18 for distances of 2m and 3m respectively. The number of characters on each German number plate varies between six and eight characters per plate, with at least two numbers and at least two letters. The total number of characters across all 35 different plates and 20 images per plate is 4800.



(a) 2m (b) 3m

Figure 5.16: Example images of German number plates recorded in an outside car park. The distance between the camera and the car is about (a) 2m and (b) 3m respectively. The resolution is 320×240 pixels.



(a) AP XZ77 (b) EF CU333 (c) EF EZ300 (d) EF FW97 (e) EF HY553 (f) EF PT59

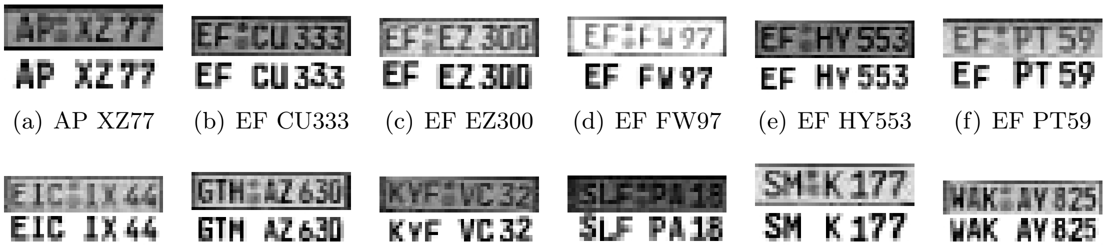
(g) EIC IX44 (h) GTH AZ630 (i) KYF VC32 (j) SLF PA18 (k) SM K177 (l) WAK AY825

Figure 5.17: Cropped example images of German number plates used for recognition at a distance of 2m. Each sub figure shows the original image (top) with the result (below).

The plate recognition rate at a distance of 2m for German number plates is 77.57%, which drops down to 55.57% for characters with an average height of 7 pixels at a distance of 3m. However, in most cases only a single character per number plate is not recognised correctly, resulting in character recognition rates of 96.34% and 91.13% for both distances respectively. Again, the main reason for a false recognition, especially in lower resolution, are similar characters like 'B' and '8' or '0' and 'O'.

Table 5.4 summarises the results of the outdoor number plate recognition of Western Australian and German number plates. Both number plate fonts have been tested in a lab environment in Section 5.2.1 for their suitability. In that test the German number plate font resulted in slightly higher recognition rates compared to the Western Australian number plate font. However, when tested on real number plates the difference in recognition rate is only marginal.

The German number plate font is called FE-Schrift (short for 'fälschungserschwerende



(a) AP XZ77 (b) EF CU333 (c) EF EZ300 (d) EF FW97 (e) EF HY553 (f) EF PT59

(g) EIC IX44 (h) GTH AZ630 (i) KYF VC32 (j) SLF PA18 (k) SM K177 (l) WAK AY825

Figure 5.18: Cropped example images of German number plates used for recognition at a distance of 3m. Each sub figure shows the original image (top) with the result (below).

	Western Australian NP		German NP	
Distance	1.5m	3m	2m	3m
Character height	13 pixel	7 pixel	11 pixel	7 pixel
Plate accuracy	95.83%	50.69%	77.57%	55.57%
Character accuracy	99.41%	90.52%	96.34%	91.13%

Table 5.4: Recognition Results. Plate accuracy is the number of correct recognised plates in percentage of 720 possible frames (20 frames for each of the 35 number plates) and character accuracy is the number of correctly recognised characters, out of 5040 and 4800 possible for the Western Australian and the German number plate font respectively.

Schrift' - falsification-hindering script), which should make this font less prone to false recognition due to similar letters. However, the high resolution German characters are created directly from the associated TrueType definition, which was created by a designer, whereas the Western Australian high resolution characters were acquired directly from the original number plates by taking high resolution images of each character. The high resolution characters of the Western Australian number plate font are therefore a more accurate model of the characters.

This experiment showed that the proposed character recognition algorithm for low resolution characters is able to recognise and simultaneously separate number plate characters down to seven pixels in height. The result is comparable with the number plate recognition results in ideal lab conditions in Section 5.2.2.1. In that experiment characters as small as 6 pixels in height were recognised with 93.40% accuracy.

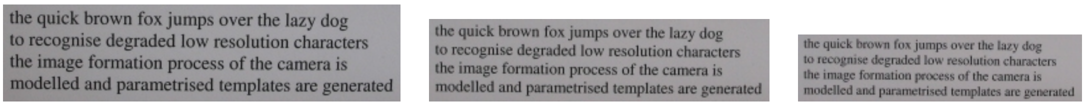
5.2.3 Text Recognition

The last experiment applies the proposed character recognition approach to text documents. Therefore, a short text with Times New Roman font size twelve is printed on a sheet of paper and captured with a compact digital camera. The resolution of the image is 640×480 pixels and the distance between the text and the camera is set to 20cm, 25cm and 30cm resulting in average characters of about eight, six and five pixels in height respectively.

The printed text is as follows

> the quick brown fox jumps over the lazy dog
> to recognise degraded low resolution characters
> the image formation process of the camera is
> modelled and parameterised templates are generated

The sentence 'the quick brown fox jumps over the lazy dog' is chosen for recognition because it is a pangram, it contains every letter of the alphabet at least once. It is therefore ideally suited for testing the proposed character recognition approach.



the quick brown fox jumps over the lazy dog
to recognise degraded low resolution characters
the image formation process of the camera is
modelled and parametrised templates are generated

(a) 20cm

the quick brown fox jumps over the lazy dog
to recognise degraded low resolution characters
the image formation process of the camera is
modelled and parametrised templates are generated

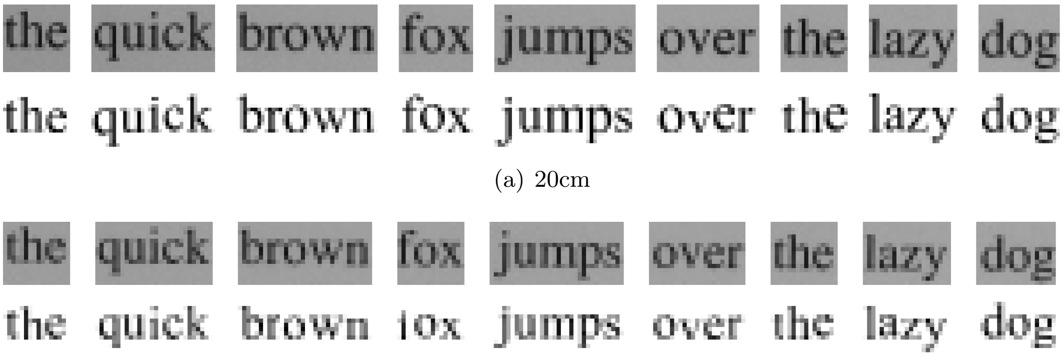
(b) 25cm

the quick brown fox jumps over the lazy dog
to recognise degraded low resolution characters
the image formation process of the camera is
modelled and parametrised templates are generated

(c) 30cm

Figure 5.19: Cropped document images used for recognition. The distance between the text document and the camera is set to 20cm, 25cm and 30cm in (a), (b) and (c) respectively.

After the is image captured, the text is cropped out of the entire image as shown in Figure 5.19. The individual text lines and the words in each line are separated by using adaptive thresholding and assuming that the text is straight. The proposed algorithm is then applied to each word separately without previous character separation.



the quick brown fox jumps over the lazy dog
the quick brown fox jumps over the lazy dog

(a) 20cm

the quick brown fox jumps over the lazy dog
the quick brown iox jumps over the lazy dog

(b) 25cm

Figure 5.20: Text recognition results for distances of 20cm (a) and 25cm (b). Each sub figure shows the original image (top) with the result (below).

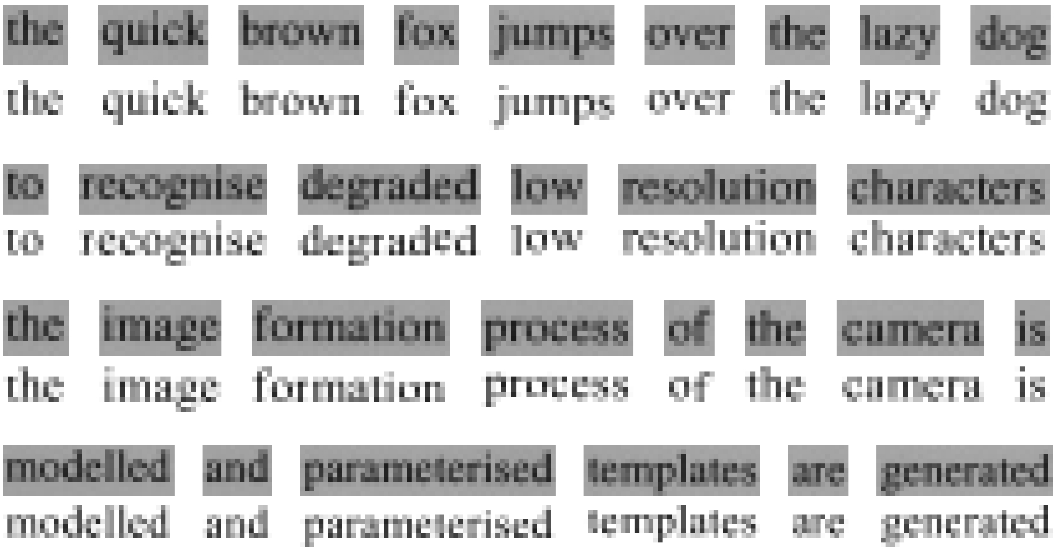

the quick brown fox jumps over the lazy dog
the quick brown fox jumps over the lazy dog
to recognise degraded low resolution characters
to recognise degraded low resolution characters
the image formation process of the camera is
the image formation process of the camera is
modelled and parameterised templates are generated
modelled and parameterised templates are generated

Figure 5.21: Text recognition results for a distances of 30cm. Each sub figure shows the original image (top) with the result (below).

The text captured at a distance of 20cm with an average character height of about eight pixels is recognised without errors achieving a recognition rate of 100%. All 29 words and 187 characters are identified correctly. The image of the first text line is shown in Figure 5.20(a), with the recognised character templates below each word.

The same text is captured at a distance of 25cm and results in average character heights of about six pixels. Again, the image of the first text line is shown in Figure 5.20(b). Only one character out of 187 is not recognised correctly; the word 'fox' is recognised as 'tox'. The reason is the overlap of the letter 'f' with the following letter 'o' in this resolution, leading to a higher correlation value for the letter 't' compared to the letter 'f' at this position.

In general, the lower case characters 'f', 't', 'i' and 'l' are most difficult to recognise especially in lower resolution. As the resolution decreases to only a few pixels per character, the appearance of these characters become more alike, as shown in Figure 5.21.

Increasing the distance between the text document and the camera even further to 30cm results in average lower case characters of about five pixels. With so few pixels single characters merge and become inseparable for standard methods, as shown in Figure 5.21. The proposed low resolution character recognition algorithm is still able to identify all 187 characters correctly.

Even though the proposed algorithm is applied to text documents it is still only a method for recognising characters at low resolution. Unlike text recognition approaches, no dictionary is used during recognition. This is an optional step that would increase the recognition performance of the proposed approach even further.

5.3 Conclusion

This chapter proposes a new method for character recognition in low resolution images by modelling the down-sampling process of the optical camera system. Using this model different low resolution templates for each character of each font are generated. A template based matching approach then uses these templates for recognition without the need for an initial character segmentation. Without using image enhancement or super-resolution techniques, each character is recognised by its low resolution grey scale appearance only. Even though a large number of templates is used, the proposed approach is practical, as the size of the low resolution input image as well as the templates themselves are quite small.

The proposed method is best suited for applications in which the font type is known beforehand, like number plate recognition or the transcript of a book. Experiments show that the chosen font type influences the recognition result. Using character templates that differ greatly from the actual font will result in low recognition rates, especially at low resolutions. Best results are achieved when using the same font type for template generation and recognition.

The performance of the proposed character recognition method is tested on number plates of Western Australian and German cars captured in outdoor environments. For characters of only seven pixels in height the resulting character recognition rate amounts to over 90% for both number plate fonts. The character recognition approach can also be applied to text documents.

The proposed method uses a priori information about an object for recognition. The high resolution appearance of the object is used to generate a number of possible low resolution appearances by modelling a simple camera. The objects in this chapter are limited to 2D characters only. Furthermore, in the template generation process in Section 5.1.1 only translation movements are considered and any rotations are neglected to reduce the number of possible templates. In order to allow for possible skew or different camera angles, three additional parameters are required to model adequate templates. Alternatively, the rotation can be detected and corrected before recognition.

The proposed template matching based approach is well suitable for low resolution character recognition. The generated templates are very small in size and the unknown character image is cropped before recognition which reduces its size and makes cross correlation feasible. Even though this approach is theoretically applicable to 3D objects given their 3D model and appearance, it is not feasible nor efficient for complex 3D objects like faces. The number of templates needed to cover all possible appearances as well as the template size prohibits an efficient usage.

Therefore, the following chapter proposes a model based face recognition method that is not based on templates but uses the results of Chapter 3. A deformable face model mask is fitted to an image of a previously unknown person and is subsequently used for recognition.

Chapter 6

Face Recognition

Automatic face recognition in large scale surveillance is still an open problem due to low resolution video and large changes in face appearance caused by different pose and lighting conditions, expressions as well as occlusion. This challenging task is further impeded by uncooperative subjects, the availability of only a single training image for each individual and the absence of 3D face shape information.

Multi-modal methods combining 2D and 3D face recognition have the potential to achieve better recognition results than either of the two alone (Abate *et al.*, 2007; Bowyer *et al.*, 2005). By incorporating the 3D shape of the face, the limitations of most 2D face recognition systems with respect to changing pose and lighting conditions can be overcome. However, to acquire the 3D face shape most sensors require the subject to cooperate which is not feasible in large scale surveillance situations.

While most existing multi-modal face recognition methods use 3D sensors to acquire the person specific face shape, the authors of Zhang and Samaras (2006) propose a multi-modal approach using only a single training image without additional 3D shape information. Instead 3D Morphable Models (3DMM) are used in combination with harmonic images (Basri and Jacobs, 2003) to perform face recognition under different pose and illumination. However, this approach requires the manual selection of facial features for accurate image alignment as well as high resolution images for fitting the 3DMM.

Based on the findings in the previous chapter, that a template based recognition approach is infeasible for complex 3D objects, this chapter proposes a model based approach for automatic multi-modal face recognition in low resolution using only a single training image per subject. A deformable 3D face model is utilised to extract 3D shape information from a single 2D image for training rather than using a 3D face scan acquired by expensive equipment. The resulting textured 3D face model is then used directly for face recognition instead of using cropped and aligned 2D images. Pose invariance is achieved indirectly by fitting the 3D face model and different lighting conditions are modelled by incorporating the findings of Basri and Jacobs (2003) into the recognition process. This method does not require the manual selection of facial feature points and works in low resolution images which makes it suitable for automatic non-intrusive surveillance and identification.

This chapter is organised as follows: The proposed method is explained in Section 6.1, including the creation of the face database, the integration of the lighting model and the face recognition approach. The experimental evaluation is outlined in the subsections of Section 6.2, followed by the conclusion in Section 6.3

6.1 Multi-Modal Face Recognition from a Single Image

The proposed multi-modal face recognition approach uses a single training image per subject and does not require any manual feature selection or image alignment. The basic outline is shown in Figure 6.1. The face database is built from a single image per individual by manually or automatically fitting a deformable 3D face mask and extracting the face texture. During the recognition step the automatic face mask fitting method developed in Chapter 3 is utilised to estimate the 3D pose and person-specific shape parameters of the person's face. The mask is textured and the lighting conditions are corrected using harmonic images. For recognition, the weighted combination of harmonic images that is closest to the test face is chosen from the database. Each step is described in detail in the following sections.

6.1.1 Creating the Face Database

The face database is created by registering a single image of each subject under neutral lighting conditions. It is assumed that the image shows a near frontal face which is uniformly illuminated without the presence of cast or attached shadows. A passport photo is a typical example of such lighting conditions. The face registration process for training is shown in Figure 6.2.

Given an image of a new person, a deformable 3D face mask is used to estimate the 3D pose, the person-specific shape parameters and to extract the face texture. Therefore, automatic or manual 3D mask fitting methods can be utilised since the face database is created offline. For automatic face registration the method proposed in Chapter 3 is applied to detect the face and to fit a deformable 3D face model to the 2D image as shown in Figure 6.2.

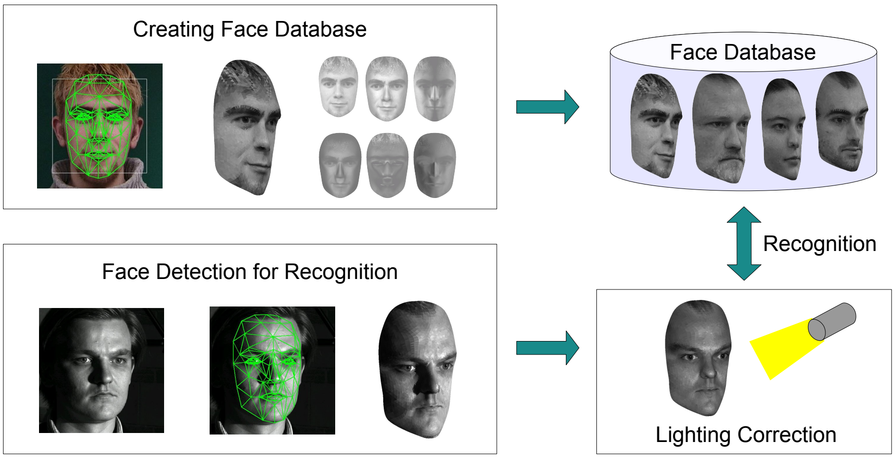


Figure 6.1: System Overview of the proposed automatic face recognition system. For creating the face database a deformable face mask is fitted to a single image per person. The resulting textured 3D mask is used to calculate harmonic images and then stored in the face database. During recognition the deformable 3D mask is automatically fitted to an image and the 3D face mask together with the harmonic images are used to adjust the lighting conditions and to match the mask texture with the face database.

After the mask is fitted to the 2D image the face texture is extracted by projecting the centre of each triangle into the image and assigning each mask triangle with a single colour value as:

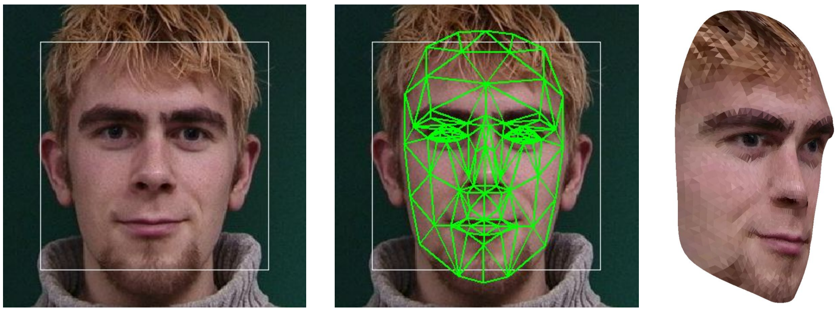
$$J = \mathcal{Q}(\mathcal{P}(\mathbf{g}, T)) \quad \text{with} \quad \mathrm{T} = [\mathrm{T}^{\mathrm{int}}, \mathrm{T}^{\mathrm{ext}}] \tag{6.1}$$

where $\mathbf{g}$ is the deformed face mask and $\mathcal{P}$ projects the centre of each mask triangle into the 2D image using the intrinsic and extrinsic camera parameters T^{int} and T^{ext} (Equation 3.1 and 3.2). $\mathcal{Q}$ then creates a vector of concatenated colour values from the textured mask vertices.

To achieve lighting invariance during recognition, nine harmonic images are constructed from each fitted and textured 3D face mask. The authors of Basri and Jacobs (2003) showed that any image I of a convex Lambertian object can be approximated as a linear combination of so called harmonic images by using spherical harmonics to model lighting conditions. These harmonic images are dependent on the surface normal η and the albedo ρ of each surface point of the 3D model as:

$$I = \sum_{h=1}^{9} \beta_h V_h(\eta, \rho) \tag{6.2}$$

where I is the intensity image, β_h are coefficients and the function V_h returns the h^{th} harmonic image given a set of surface normals η and their albedo ρ. The first nine



(a) Face Detection (b) Mask Fit (c) 3D Model

Figure 6.2: Face Registration for Training. To register a new person into the face database, the face is automatically detected (a), the deformable face mask is automatically fitted (b) and the 3D mask is textured (c).

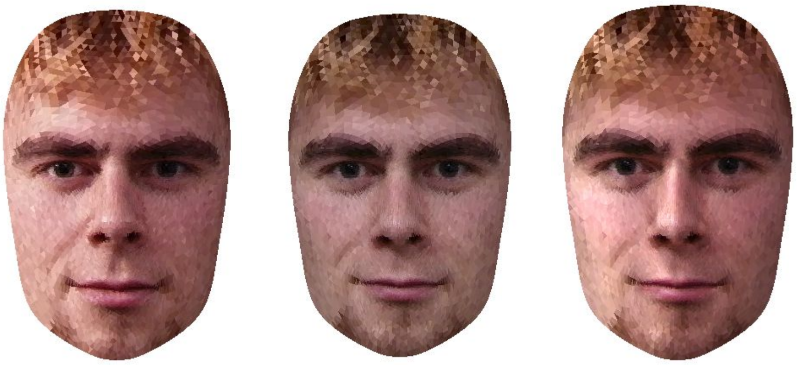
harmonic images are a sufficient approximation of the illumination cone according to Basri and Jacobs (2003) and are used within the proposed face recognition method.

The function V_h is then used to calculate the first nine harmonic images given the albedo ρ, i.e. the mask texture J, and the surface normals η of each triangle of the deformed 3D face mask $\mathbf{g}$. The texture vector J, the shape parameters γ which are necessary to deform the mask and the nine harmonic images are then stored in the face database for recognition.

6.1.2 Pose and Lighting Invariant Recognition

For automatic face recognition of a previously unseen person, the mask fitting method as proposed in Chapter 3 is applied first to recover the 3D pose and person-specific shape of the face. This pre-processing step, necessary for automatic face recognition, includes 2D image based face detection and the automatic fitting of a deformable 3D face mask to the image. This overcomes the first limitation of traditional 2D face recognition - variance in pose - since the mask automatically and implicitly recovers the 3D pose of the face.

Most traditional 2D face recognition approaches require the face images to be normalised such that the spatial position of the facial features (like eyes and mouth) are aligned throughout the dataset. However, by using a textured mask instead of a 2D image for recognition, all facial features are aligned implicitly. After the mask is fitted to the 2D image, it is subdivided and textured by assigning a colour value to each mask triangle. The centre of each triangle is projected onto the image according to Equation 6.1 and the texture vector $\mathbf{J}_{new}$ is extracted.



(a) $\mathbf{J}_{new}$ (b) $\mathbf{J}$ (c) $\min_\beta ||\mathbf{J} - \beta V(\eta, \mathbf{J})||_2$

Figure 6.3: Given a new, unknown face texture $\mathbf{J}_{new}$, the first nine harmonic images are used to recover the lighting conditions and the face texture $\mathbf{J}$ is adjusted accordingly. The result after minimising $\min_\beta ||\mathbf{J}_{new} - \beta V(\eta, \mathbf{J})||_2$ is shown in 6.3(c).

Lighting invariance, the second main limitation of most 2D face recognition methods, is achieved by integrating spherical harmonics in the form of harmonic images during recognition. Given the mask texture of a new face, $\mathbf{J}_{new}$ a weighted combination of harmonic images is recovered for every individual in the face database and the distance to the unknown texture vector $\mathbf{J}_{new}$ is minimised as:

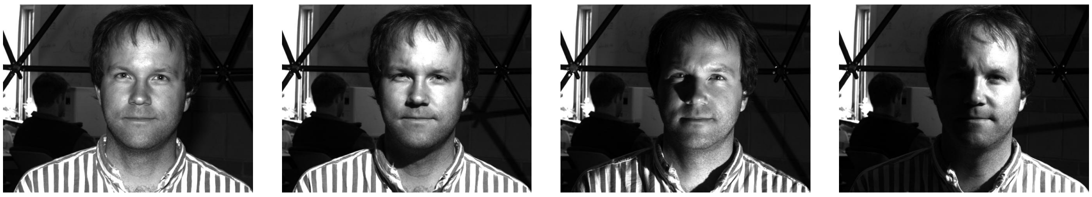
$$\min_\beta ||\mathbf{J}_{new} - \beta V(\eta, \mathbf{J})||_2 \tag{6.3}$$

where $\mathbf{J}_{new}$ is the texture vector of the unknown face, $\mathbf{J}$ is a face in the database and V returns nine harmonic images given the surface normals of the 3D model η and its texture $\mathbf{J}$. QR decomposition with pivoting is used to find the parameter vector β that minimises this equation. The weighted combination of harmonic images in the database that is closest to the unknown face $\mathbf{J}_{new}$ is chosen. An example is shown in Figure 6.3.

6.2 Experiments

The Yale Face Database B (Georghiades *et al.*, 2001) is used throughout this section. This dataset contains ten individuals and over 400 images for each subject which sample nine different poses under 45 different illumination conditions. Despite its relatively small size it has become a standard for comparing face recognition methods for variable lighting conditions as well as different poses.

This data set is subdivided into four subsets depending on the angle between the light



(a) Subset 1 up to 12° (b) Subset 2 up to 25° (c) Subset 3 up to 50° (d) Subset 4 up to 77°

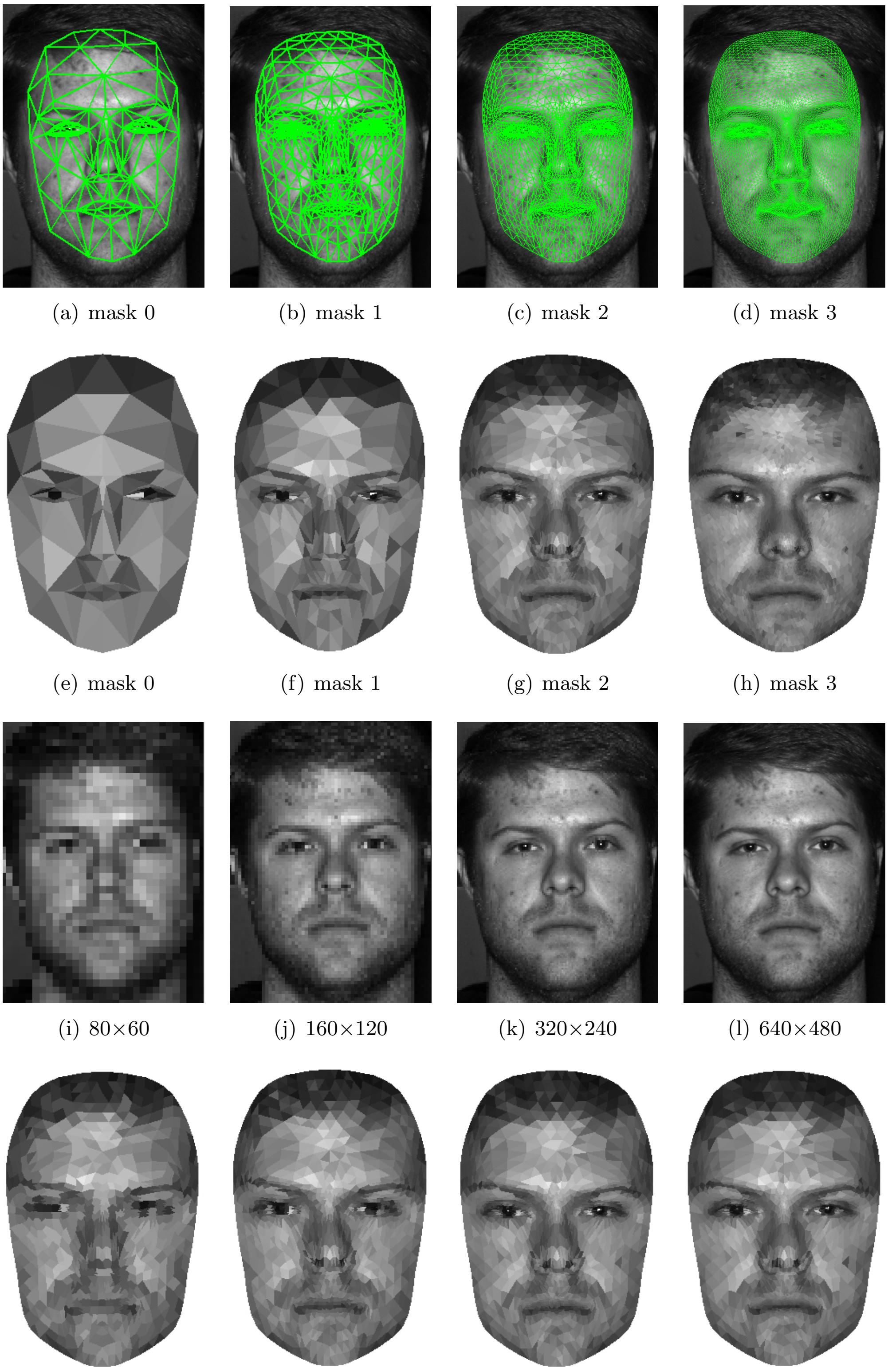
Figure 6.4: The Yale Face Database is divided into subsets depending on the angle between between the camera axis and the light source. Figures (a)-(d) show example images of each subset.

source and the camera axis according to Georghiades *et al.* (2001). The resulting Subset 1 contains seven images of each subject under near frontal illumination (up to 12°). The lighting conditions become more extreme towards Subset 4 with up to 70° between the light source and the camera axis. Sample images of each subset are shown in Figure 6.4 with Subsets 2 and 3 both containing twelve and Subset 4 containing fourteen different images of each individual.

The face recognition experiments on the Yale Face Database B described in the following sections are evaluated with respect to different image resolutions, pose and lighting conditions and fully automatic mask fitting. The results are compared with a 2D face recognition method based on image tensors (Rana, 2009) as well as with a large number of different approaches reported in literature (Georghiades *et al.*, 2001; Lee *et al.*, 2005; Chen *et al.*, 2000; Zhang and Samaras, 2006).

6.2.1 Recognition under Different Resolutions

The experiments in this section are designed to evaluate the face recognition performance for different image resolutions as well as for different mask sizes, i.e. for different number of mask subdivisions. The Yale Face Database B contains images of size 640×480 pixels, this resolution is cut into half three times successively to result in images of size 320×240, 160×120 and 80×60 pixels with average face sizes of 280×200, 140×100, 70×50 and 35×25 pixels respectively. Furthermore, the original coarse CANDIDE-3 mask is subdivided three times to result in a fine mesh. Example images of different image resolutions and mask sizes are shown in Figure 6.5.



(a) mask 0 (b) mask 1 (c) mask 2 (d) mask 3

(e) mask 0 (f) mask 1 (g) mask 2 (h) mask 3

(i) 80×60 (j) 160×120 (k) 320×240 (l) 640×480

(m) 80×60 (n) 160×120 (o) 320×240 (p) 640×480

Figure 6.5: Different mask sizes (a)-(d) and the resulting mask textures (e)-(h) and different image resolutions (i)-(l) and the resulting textured mask 2 (m)-(p) used for recognition.

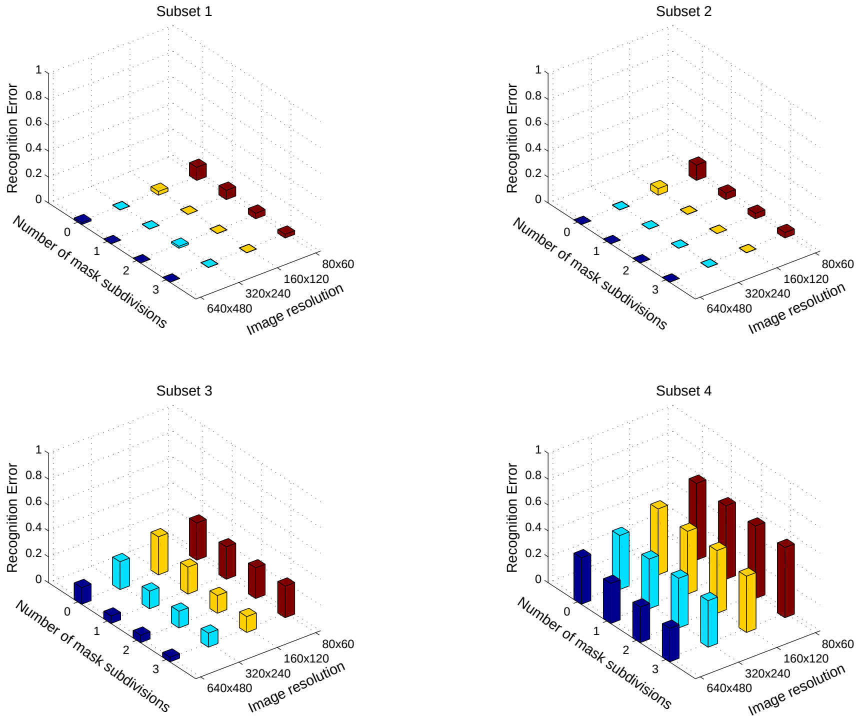


Figure 6.6: Recognition rates for different image resolutions and mask sizes used for testing. Each graph shows the results for one particular subset of the Yale Face Database B. A single image of size 640×480 pixels under frontal illumination is used for training.

The proposed face recognition approach as described in Section 6.1 is used for recognising the individuals in the Yale Face Database B under different lighting conditions in Subsets 1 to 4 by using different image resolutions and mask sizes. For the first experiment a single frontal image of size 640×480 pixels of each subject under frontal illumination is used for training and the 3D face mask is fitted manually during training and testing.

The mean recognition error across ten individuals for each subset are shown in Figure 6.6. Best recognition rates are achieved for Subsets 1 and 2 containing seven and twelve images respectively. The recognition error increases for all image resolutions and mask sizes for Subsets 3 and 4 that contain more extreme lighting conditions with twelve and fourteen images respectively.

Common across all subsets is the increase in error rates for low resolution images of size 80×60 pixels and a coarse mask 0 (Figure 6.5(e)). The error rate increases with both, decreasing image resolution and decreasing number of mask triangles, i.e. a coarser mask. The highest recognition rate is achieved by images of resolution 640×480 pixel and a

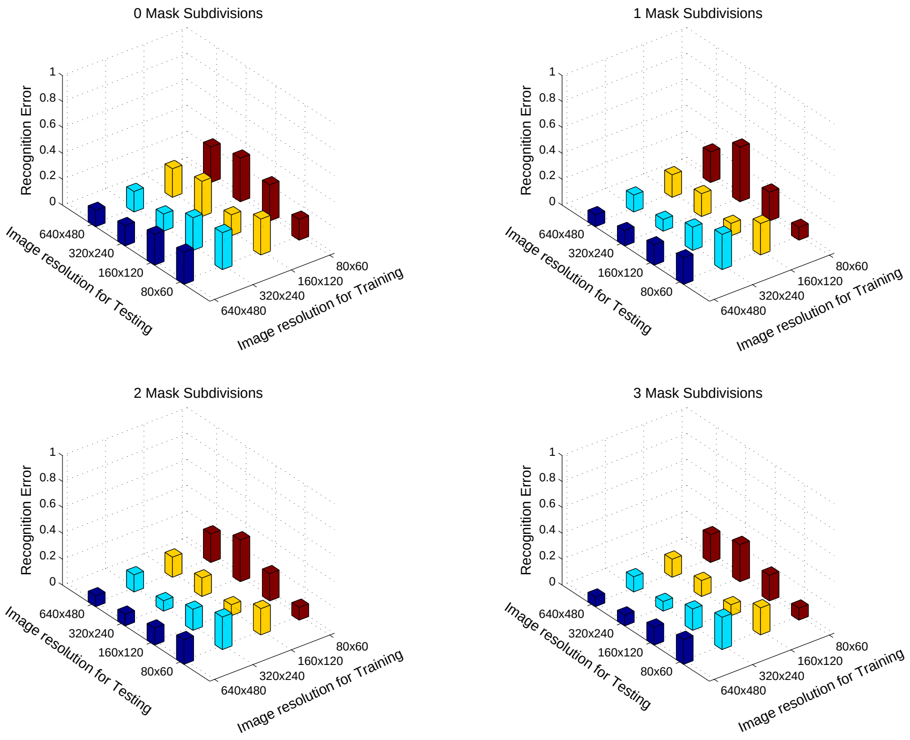


Figure 6.7: Recognition rates for different image resolutions and mask sizes used for training and testing. Each graph shows the results of one particular mask size as the average recognition error across all persons and subsets of the Yale Face Database B.

mask 3 (Figure 6.5(h)) resulting in error rates of 0, 0, 0.035 and 0.137 for Subsets 1 to 4 respectively and corresponding false alarm rates of 0, 0, 0.03 and 0.238.

The second experiment uses different image resolutions for training and testing. Again different mask sizes are used and the resulting recognition errors as the mean across all four subsets for all individuals (450 images) are shown in Figure 6.7. Each graph shows the result for a particular mask size used for training and testing.

Noticeable in all four graphs is that the lowest recognition error are along the main diagonals. This means that the best recognition results are achieved by using the same image resolution for training and testing. However, the off-diagonal results do not differ greatly from the error rates along the diagonals, especially for training images of size 640×480 pixels. As a result, a high resolution passport photo for example can be used to texture different mask sizes. These masks are then sufficient for recognising images of lower resolution without the need for image resizing or scaling because the resolution is determined by the size of the mask or the image resolution, whichever is smallest.

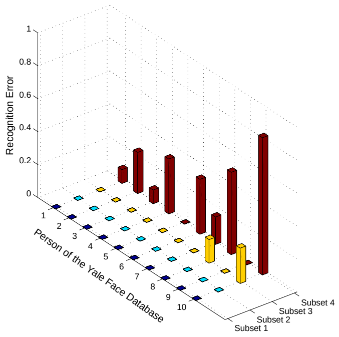
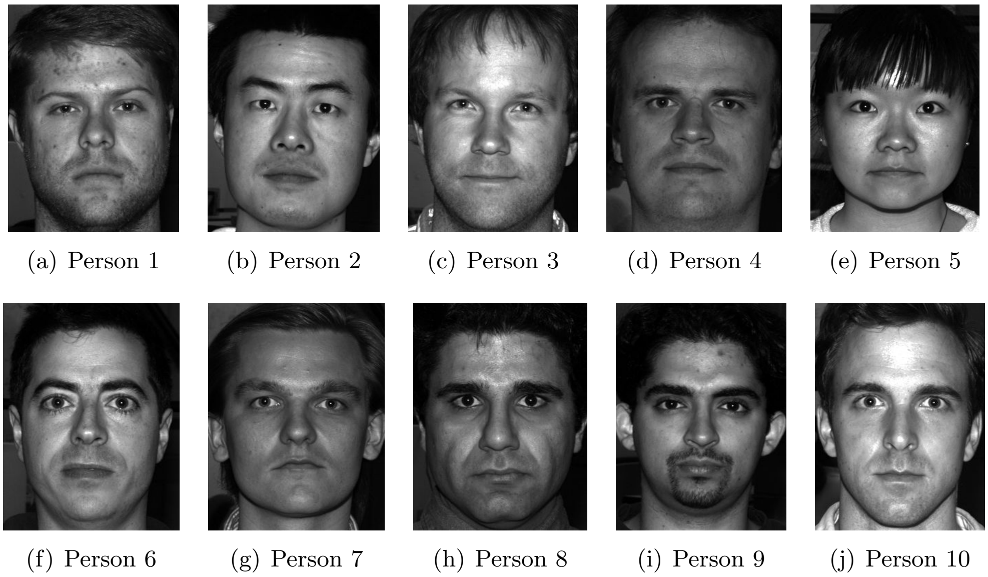


Figure 6.8: Detailed recognition errors for all ten individuals of the Yale Face Database B for different subsets.

Furthermore, the mask size affects the resulting error rates only slightly. The original CANDIDE-3 mask 0 is too coarse, showing only little detail (Figure 6.5(e)) and therefore the resulting error rates are highest. The recognition error decreases with the number of subdivisions, the finer the mask, the smaller the recognition error. However the error difference between mask 1 to 3 amounts to only 0.02%. The reason for this are differences in the triangle size for different parts of the 3D face mask. After one or two subdivisions the eye and mouth area are already represented by a fine mesh compared to areas like the cheeks or the forehand as shown in Figure 6.5. Since the eyes and the mouth areas are most important for recognising subjects, these parts are already sufficiently represented by a fine mesh such that the recognition rates improve only slightly.

Lastly, the detailed recognition rates for all ten individuals and different subsets are shown in Figure 6.8. The recognition rates are calculated by using mask 3 (Figure 6.5(h)) and images of resolution 640×480 pixels for training and testing. The graph shows that all images in Subset 1 and 2 are recognised without errors. Persons 8 and 10 were not recognised in two and three images respectively in Subset 3 and the error rates for nearly all subjects are highest for Subset 4 that contain images with extreme lighting conditions.

As mentioned previously in Chapter 3, the deformations of the CANDIDE-3 face mask do not allow for a precise modelling of a person's face shape. As a result it can only be used to model non-extreme lighting conditions like in subsets 1, 2 and partially 3. Furthermore, the example images of each subject in the Yale Face Database B in Figure 6.9 show that the face shape of Person 8 differs from most others and is not accurately represented by the CANDIDE-3 mask. The training image used for Person 10 in Figure 6.9(j) shows a bright reflection on his forehand which also hinders recognition.



(a) Person 1 (b) Person 2 (c) Person 3 (d) Person 4 (e) Person 5

(f) Person 6 (g) Person 7 (h) Person 8 (i) Person 9 (j) Person 10

Figure 6.9: The image of each subject of the Yale Face Database B with frontal lighting that is used for training.

6.2.2 Recognition Under Different Pose and Lighting Conditions

The following experiment demonstrates the ability of the proposed face recognition algorithm under varying pose and lighting conditions. Again, the Yale Face Database B is used which contains images of nine different poses and 45 different illumination conditions for ten individuals. An example image of each pose is shown in Figure 6.11.

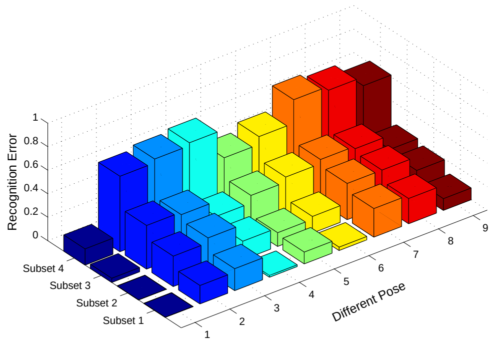
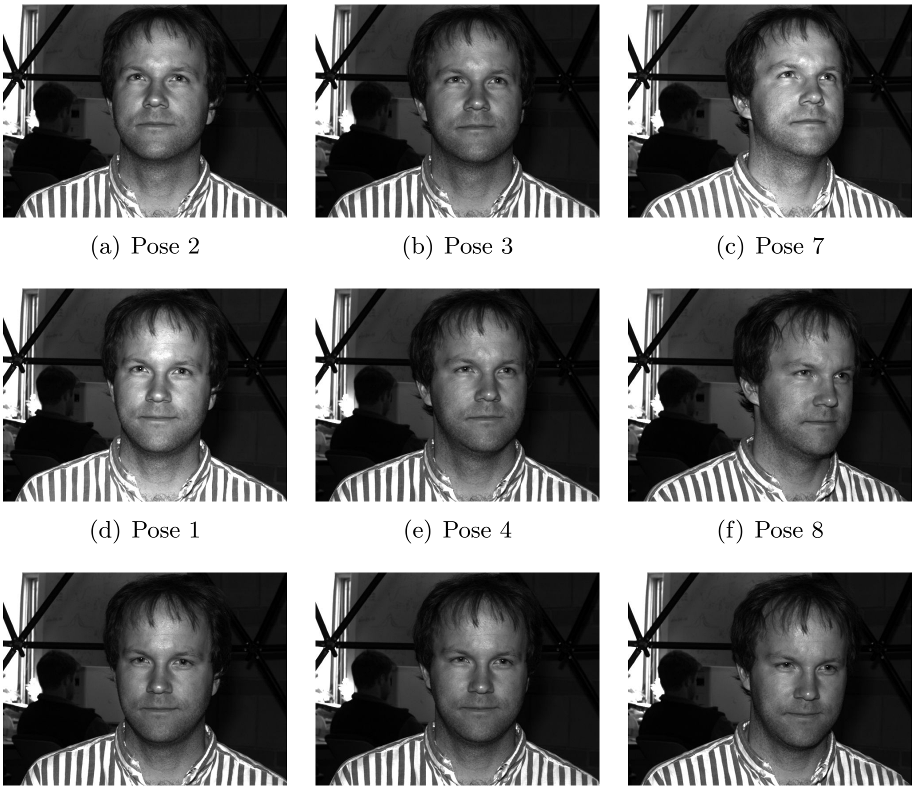


Figure 6.10: Recognition rates for the Yale Face Database B for different pose and different lighting conditions. The recognition rates shown are the average over ten persons and 7, 12, 12 and 14 different lighting conditions for Subset 1, 2, 3 and 4 respectively.

For the following experiment the mask is fitted to each image manually. The mask is then subdivided three times, each mask triangle is assigned a colour value and the illumination is


(a) Pose 2 (b) Pose 3 (c) Pose 7

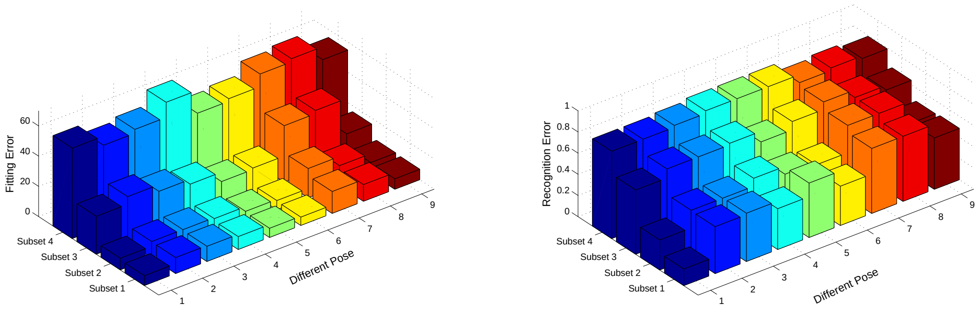
(d) Pose 1 (e) Pose 4 (f) Pose 8

(g) Pose 6 (h) Pose 5 (i) Pose 9

Figure 6.11: The Yale Face Database B contains face images under nine different poses, ranging from frontal (Pose 1) to looking up and far left (Pose 7) and looking down and far left (Pose 9).

adjusted according to Section 6.1.2 using a single frontal image of each subject as reference. The face alignment as well as the correction for pose is implicitly done by fitting the mask to the person's face. The identity of a new face then equals the weighted combination of harmonic images that is closest to the new image.

The average recognition error across ten persons and 45 different lighting conditions for each pose are detailed in Figure 6.10. The recognition rates for each pose and each subset are shown separately. As expected the recognition rates are best for Subset 1, with only slight illumination changes. These rates decrease for Subsets 2, 3 and 4 with extreme lighting conditions, the darker and less illuminated the face, the more challenging the recognition task. Furthermore, frontal images achieve best results while poses diverting from the frontal view show increased errors. As the person's face turns away from the camera, its face as well as the fitted mask become more and more occluded, resulting in decreasing recognition rates.



(a) Automatic Fitting

(b) Automatic Recognition

Figure 6.12: The results of the automatic fitting approach (a) are used for automatic face recognition (b). The fitting error is shown as the mean vertex point different and the recognition error is the mean across ten individuals in each subset and for each pose.

These results show the limitations of the proposed approach. The images in Subset 3 and 4 show extreme lighting conditions with large shadows but the generic face shape provided by the CANDIDE-3 face mask is not sufficient to model these conditions. Accurate 3D scans are needed for such illumination conditions. Similar to the pose and shape estimation results in Section 3.2.4.4, near frontal poses as well as 'looking down' poses are recognised best. Poses that differ most from the frontal pose, where large parts of the face are occluded result in the highest recognition errors.

6.2.3 Automatic Face Recognition

The experiments in this section all apply the automatic 3D mask fitting approach as proposed in Chapter 3 to recover the 3D pose and the person-specific shape parameters during recognition. Again, only a single training image of each subject with frontal lighting is used for training. The CANDIDE-3 face mask is automatically fitted, subdivided three times and recognition is performed on images of size 640×480 pixels using the recognition approach outlined in Section 6.1.2.

For comparison the results of the automatic fitting are shown Figure 6.12(a) as the mean vertex point difference in pixels. Using these fitting results, the resulting recognition errors are shown in Figure 6.12(b). The two graphs show that the recognition errors depend on the fitting accuracy, larger fitting errors result in larger recognition errors due to the misalignment of training and testing images. Furthermore, the limitations of the CANDIDE-3 face mask are shown in the high recognition errors for Subset 3 and 4 as well as for non frontal poses. These results show that accurate alignment of training and testing images is necessary for high recognition rates and that the fitting accuracy of the proposed approach is not sufficient for accurate face recognition.

6.2.4 Comparison with Tensor based method

The proposed recognition approach is compared with a tensor based method for 2D face recognition using only a small number of training images. The authors of Rana (2009) propose a method for incomplete and unbalanced datasets and assume a complete (in terms of different pose and varying lighting conditions) dataset of friendly and cooperative people is available for training. Using this complete dataset a tensor is trained and person-identity vectors are calculated for friendly people as well as for hostile people with only a few training images available.

This tensor based algorithm is used for the following facial recognition task on the Extended Yale Face Database B, which includes images of 38 people in nine different poses and under 45 different lighting conditions. For this experiment only the frontal images of each person are used. For this experiment the dataset is divided into a set of friendly people, with all images available, and hostile people, with only a single image available. The frontal image under frontal lighting, i.e. zero azimuth and zero elevation, is chosen as the only image available for each hostile person in the training set. The complete set of images of the friendly people is used for training using methods in Rana (2009) and the single image for each hostile person is used to calculate the person-identity vector. The remaining images in Subset 1 to 4 for each hostile person are used for recognition. Thus, the recognition of hostile persons requires only a single training image similar to the proposed recognition method based on a 3D face model.

Number of Hostile Persons	11	14	17	20	23	26
Subset 1	0.0	6.12	4.2	4.29	6.83	6.59
Subset 2	0.0	0.0	0.0	2.92	7.61	15.71
Subset 3	14.29	18.37	23.53	30.36	39.13	47.8
Subset 4	43.94	53.57	52.94	61.25	66.67	75.32

Table 6.1: Tensor based face recognition based on Rana (2009). The recognition error for different subsets and varying number of hostile people.

The results are shown in Table 6.1. Each column contains the recognition error for each of the four subsets for different numbers of hostile people. In the last column for example, the first 12 people of the Extended Yale Face Database B are used for learning the tensor structure and the remaining 26 hostile people are used for recognition. Recognition rates are best for the first subset with only slight lighting changes. Large angles between the light source and the camera axis cause dark shadows on the face, which in turn decrease the recognition performance. Furthermore, with the number of friendly people decreasing and the number of hostile people increasing at the same time, the recognition error increases as shown in Table 6.1.

Number of Hostile Persons	11	14	17	20	23	26	34
Subset 1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Subset 2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Subset 3	6.3	9.25	13.24	14.29	15.13	17.65	25.0
Subset 4	14.46	19.61	25.74	29.41	35.54	40.2	62.0

Table 6.2: The proposed face recognition approach. The recognition error for different subsets and varying number of hostile people.

For comparison the same experiment is repeated using the proposed method for face recognition based on a deformable 3D face mask. The face mask is manually fitted to all individuals of the Extended Yale Face Database B, the texture is extracted and the proposed recognition approach is applied.

The proposed method does not require any learning unlike the tensor based face recognition method. During the recognition, the textured mask of each hostile person is compared against the mask texture of the unknown person as described in Section 6.1.2. The recognition errors for different number of hostile persons and different lighting conditions are shown in Table 6.2.

The proposed face recognition method uses only a single image for each hostile person and outperforms the tensor based approach for each subset and for each number of hostile people. The generic 3D face mask is able to model the different lighting conditions more accurately than the tensor method in Rana (2009) leading to better recognition rates when only a single image is available for training. The mask only approximates the shape of the person's face because the number and amount of deformations is limited. However, these additional 3D shape information are sufficient to model different lighting conditions which are necessary for accurate face recognition.

6.2.5 Comparison of Recognition Results

The Yale Face Database B, despite its relatively small size, is commonly used for comparing different face recognition methods. As mentioned earlier, the face images contained in this dataset sample sufficiently the whole illumination space and have therefore become a testing standard. Following is a summary of different face recognition methods proposed in the literature and their results in comparison to the results of the proposed approach. Table 6.3 gives a quick overview of the different methods and their recognition results for ten different persons under seven, twelve, twelve and fourteen different lighting conditions in Subset 1, 2, 3 and 4 respectively. Each of these methods is described in detail in Section 2.5.

Methods	**Subset 1**	**Subset 2**	**Subset 3**	**Subset 4**
Correlation (Georghiades *et al.*, 2001)	0.0	0.0	23.3	73.6
Eigenfaces (Georghiades *et al.*, 2001)	0.0	0.0	25.8	75.7
Eigenfaces w/o 1^{st} 3 (Georghiades *et al.*, 2001)	0.0	0.0	19.2	66.4
Linear Subspace (Georghiades *et al.*, 2001)	0.0	0.0	0.0	15.0
Cones - attaches (Georghiades *et al.*, 2001)	0.0	0.0	0.0	8.6
Cones - cast (Georghiades *et al.*, 2001)	0.0	0.0	0.0	0.0
9PL (Lee *et al.*, 2005)	0.0	0.0	0.0	2.8
9PL (real images) (Lee *et al.*, 2005)	0.0	0.0	0.0	0.0
Gradient Angles (Chen *et al.*, 2000)	0.0	0.0	0.0	1.4
3DMM + harmonic images (Zhang and Samaras, 2006)	0.0	0.0	0.0	2.8
CANDIDE + harmonic images	0.0	0.0	3.57	13.73

Table 6.3: Face recognition results for different face recognition methods proposed in literature. Recognition errors are taken from the respective references and are shown as the average over ten individuals for each subset.

For easier comparison of the face recognition methods a detailed summary is given in Table 6.4. The first three methods, namely Correlation, Eigenfaces and Eigenfaces without the first three largest eigenvectors, are standard face recognition methods commonly used for comparison. These methods do not specifically model different illumination conditions and thus result in the largest recognition errors especially for Subset 3 and 4. The linear subspace method constructs a three-dimensional illumination subspace for each subject but does not allow for shading, thus resulting in an error rate of 15% for Subset 4.

The first four methods in Table 6.3 use all seven images in Subset 1 for training, therefore the resulting recognition errors for Subset 1 are almost by definition zero. The two illumination cone methods also use all images in Subset 1 for each subject to reconstruct the surface of the face. This 3D surface is then used to render 121 images under different illumination conditions from which the cone is calculated for each person. These 121 images cover almost the entire illumination space resulting in very low recognition errors. Similarly, the nine points of light methods (9PL) uses specific point light source configurations to acquire the basis images directly instead of using a large number of training images and the resulting recognition errors are equally small.

The last two methods, namely the Gradient Angles and 3D Morphable Models (3DMM) with harmonic images, use only a single image of each subject for testing and training. While the Gradient Angles method is a 2D face recognition approach that requires the manual alignment of all images for training and testing the 3DMM method fits a 3DMM to the training image to recover its shape. However, the fitting of the 3DMM also requires the manual assignment of facial feature points.

The proposed method is similar to the 3DMM based method in Zhao *et al.* (2006) in that the 3D shape of the face is used to generate nine harmonic images, also called basis images. This differs from the illumination cone and 9PL method that use a large set of training images or specific point light source configurations to acquiring these basis images. The advantage of using a 3D face model is that only a single training image is sufficient for estimating the 3D shape which can then be used to render the harmonic images. However, the precise 3D face shape is needed for accurate illumination modelling.

Using only a single training image of Subset 1 for each individual, the proposed face recognition approach is able to recognise all images of Subset 1 and 2 without errors, provided the mask is fitted manually. The recognition error increases to 3.57% and 13.73% for Subset 3 and 4 respectively due to inadequate face shape representation.

Methods	**training images per subject**	**Training**	**Pre-processing**
Correlation (Georghiades *et al.*, 2001)	7 (Subset 1)	n/a	faces manually located and aligned
Eigenfaces (Georghiades *et al.*, 2001)	7 (Subset 1)	calculate Eigenfaces of feature space	faces manually located and aligned
Eigenfaces w/o 1^{st} 3 (Georghiades *et al.*, 2001)	7 (Subset 1)	calculate Eigenfaces of feature space	faces manually located and aligned
Linear Subspace (Georghiades *et al.*, 2001)	7 (Subset 1)	calculate linear subspace for each subject	faces manually located and aligned
Cones - attaches (Georghiades *et al.*, 2001)	7 (Subset 1)	reconstruct surface, generate 121 images from which cone is formed	faces manually located and aligned
Cones - cast (Georghiades *et al.*, 2001)	7 (Subset 1)	reconstruct surface, generate 121 images from which cone is formed	faces manually located and aligned
9PL (Lee *et al.*, 2005)	9 (mostly Subset 4)	Acquire nine images under specific lighting conditions	faces manually located and aligned
9PL (real images) (Lee *et al.*, 2005)	9 (mostly Subset 4)	Acquire nine images under specific lighting conditions	faces manually located and aligned
Gradient Angles (Chen *et al.*, 2000)	1	joint probability density from 1280 images of objects (non faces)	faces manually located and aligned
3DMM + harmonic images (Zhang and Samaras, 2006)	1	rendered basis images for each person	3DMM manually fitted
CANDIDE + harmonic images	1	appearance model for fitting	manual or automatic fitting

Table 6.4: Comparison of different face recognition methods proposed in literatur.

6.2.6 Experiments Summary

The experiments in this chapter show that by using a subdivided mask, face recognition can be performed independent of the image resolution. The number of subdivisions determines the size of the mask triangles and thus the resolution of the mask texture. A finer mask results in a higher resolved mask texture and in turn leads to higher recognition rates compared to a coarse mask.

The automatic recognition results indicate the limitations of the proposed face recognition approach. Precise image alignment is important for accurate face recognition, however the fitting accuracy of the proposed approach is not accurate enough and thus its use for face recognition is limited. However, the CANDIDE-3 face mask is sufficient for modelling non-extreme lighting conditions without the use of an accurate 3D face scan and good recognition results are achieved when the mask is fitted manually.

Comparative experiments with a tensor based 2D face recognition approach in Section 6.2.4 show that face recognition methods which include the 3D face shape for modelling illumination are superior to 2D based methods, especially when only a small number of training images is available for each subject.

Compared to a number of different face recognition approaches, the proposed approach achieves good recognition results for non-extreme lighting conditions without the need for precise 3D shape recovery. The CANDIDE-3 face mask can be used for modelling different illumination conditions without accurate person-specific 3D face scans.

6.3 Conclusion

The face recognition method proposed in this chapter utilises a deformable 3D face mask to recover the 3D pose and person-specific shape parameters given a 2D image. Only a single frontal image under uniform illumination is used for training and lighting invariance is achieved by incorporating spherical harmonics into the recognition approach.

The harmonic images are chosen for modelling the lighting conditions because they only require a 3D model of the object along with its albedo. Instead of an exact 3D scan of a person's face, which to this date can not be obtained in a non-cooperative large scale surveillance situation, the proposed face recognition algorithm uses the result of the face mask fitting approach in Chapter 3. The fitted, subdivided, person-specific deformed face mask provides a good approximation of the person's face and is used instead of a more accurate 3D face scan.

Fitting a deformable 3D face mask recovers the spatial position of all facial features and thus makes normalisation and alignment redundant. The vectors of concatenated triangle colour values are used for recognition, which makes the proposed approach independent of the input image resolution of the face. Cropping and adjusting the size of the image is unnecessary.

However, experiments showed that the accurate alignment of testing and training images is indispensable for accurate face recognition. The fitting accuracy achieved by the automatic fitting approach in Chapter 3 is not sufficient for accurate face recognition. A manual mask fit is required to achieve good recognition rates for non-extreme lighting conditions (as in Subset 1 and 2 and in parts of Subset 3). The CANDIDE-3 face mask represents a generic face shape that does not allow for precise face shape modelling however, this generic face is sufficient to model non-extreme lighting conditions.

Chapter 7

Conclusion

This thesis has investigated model based methods for automatic processing of low resolution video objects under different pose and illumination conditions where standard feature based methods are particularly difficult to apply. The developed methods include face pose and shape estimation, object tracking, super resolution and low resolution character and face recognition.

The method for automatic pose and shape estimation, the first step towards automatic video processing, is described in Chapter 3. The deformable 3D face mask CANDIDE-3 is utilised within a particle filter based fitting approach to recover the 3D pose and the person-specific shape parameters. Harmonic images (Basri and Jacobs, 2003) are included into the error function of the particle filter to allow for accurate mask fitting under different illumination conditions. However, only non-extreme lighting conditions can be modelled accurately due to the limited deformability of the CANDIDE-3 face mask. The precise 3D shape of a person's face is necessary for precise modelling of extreme lighting conditions. Using computer graphics subdivision schemes, the face mask is subdivided into a fine mesh to allow for accurate fitting in different image resolutions, as demonstrated in experiments on the IMM Face Database (Nordstrøm *et al.*, 2004). A comparative evaluation of the proposed method and an Active Shape Models based approach shows that the particle filter based fitting works consistently better especially in low resolution and under different lighting conditions.

A novel approach combining tracking and super resolution is presented in Chapter 4. Only the resolution of the object is increased during tracking while the rest of the image scene remains unchanged. This form of super resolution is made possible by subdividing the object mask into a fine mesh, such that every triangle is smaller than a pixel when projected in to the image. Experiments show that such a fine mesh is not only necessary for super resolution it also benefits the tracking. Furthermore, the combined appearance and geometric tracking approach achieves better results than either of the tracking methods alone. When applied to non-planar and non-rigid objects like human faces, the combined tracking and super resolution approach outperforms traditional super resolution optical flow. Unlike optical flow based methods no interpolation is needed to increase the resolution, resulting in less blurred images. The larger the number of mask triangles, the higher the possible increase in resolution, however more frames are required to achieve this increase. In contrast to optical flow based methods, more frames are needed to achieve the same resolution increase because interpolation is omitted for less blurred results.

Based on the effect of the image formation process utilised in the previous chapter, a character recognition method is developed in Chapter 5. For recognising low resolution characters, the simple camera model is parameterised with three different parameters for an efficient generation of low resolution character templates. Experiments on four different font types show that this model generates character templates that accurately match character images taken by a compact digital camera, achieving average correlation values of 0.98. Best recognition results are achieved by using identical fonts for template generation and character recognition. The small size of the character templates allow for an efficient recognition using template matching and clustering techniques. Characters are recognised without prior character separation and without binarising the images, unlike traditional optical character recognition (OCR) methods. In contrast, the grey scale edges of each character contain important information and benefit the recognition. Experiments on German and Western Australian car license plates demonstrate the practicability of the approach for automatic plate recognition in low resolution images, typically taken by wide area surveillance cameras.

Based on the automatic fitting of a deformable 3D face mask proposed in Chapter 3, a face recognition method is developed in Chapter 6. The 3D face mask is used to implicitly recover the 3D pose of the face as well as person-specific shape parameters. The fitted, subdivided and textured mask is then used for recognition instead of cropped 2D images, making image alignment and resizing unnecessary. Provided with a precise fit the generic shape of the mask is sufficient to model non-extreme lighting conditions and good recognition results are achieved. However, precise image alignment is indispensable for high recognition rates and the automatic mask fitting approach is not accurate enough for the use within a face recognition approach.

7.1 Future Work

The methods developed in this thesis aim to improve automatic video processing for automatic scene interpretation and analysis. However, there are certain limits that are worth investigating in the future. Foremost is the deformable 3D face model CANDIDE-3 used in nearly all chapters. The fourteen hand defined shape parameters allow for an easy deformation of the face mask. However, this face mask is too rigid to model the exact shape of a person's face, like the form of the cheek bones or the nose shape. As a result, harmonic images can only be applied in conjunction with this mask to model non-extreme lighting conditions for mask fitting and face recognition.

Further research should investigate different 3D deformable face masks that accurately model a specific face like Tao and Huang (1999); Roussel and Gagalowicz (2005) but allow for efficient and accurate fitting, preferably in real time for surveillance applications. A first attempt that uses 3D Morphable Models together with harmonic images is proposed by (Zhang and Samaras, 2006). However, the fitting requires the manual assignment of feature points which prohibits automatic face recognition.

Another area that can be readily improved is the character recognition method proposed in Chapter 5. As mentioned earlier in Section 5.3, the transformations allowed for in the template generation process are limited to translations parallel to the image plane and scaling in order to keep the number of possible templates low and to allow for efficient recognition. However, to be applicable in real world surveillance scenarios, the algorithm should also include rotation and skew by either detection and correction of the image or by including it into the template generation process. However, including additional parameters into the template generation process would increase the run time of the template matching algorithm, thus detecting and image rectifying would be favourable.

Bibliography

Abate, A. F., Nappi, M., Riccio, D., and Sabatino, G. (2007). 2d and 3d face recognition: A survey. *Pattern Recognition Letters*, **28**(14), 1885–1906.

Agrawal, A. and Raskar, R. (2007). Resolving objects at higher resolution from a single motion-blurred image. *IEEE Conference on Computer Vision and Pattern Recognition*.

Ahlberg, J. (2001). CANDIDE-3 - an updated parameterized face. Technical Report LiTH-ISY-R-2326, Dept. of Electrical Engineering, Linkping University, Sweden.

Anagnostopoulos, C., Anagnostopoulos, I., Loumos, V., and Kayafas, E. (2006). A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems*, **7**(3), 377–392.

Anagnostopoulos, C., Anagnostopoulos, I., Psoroulas, I., Loumos, V., and Kayafas, E. (2008). License plate recognition from still images and video sequences: A survey. *Journal of Intelligent Transportation Systems*, **9**(3), 377–391.

Anagnostopoulos, C., Anagnostopoulos, I., Psoroulas, I., Loumos, V., and Kayafas, E. (As retrieved in October 2009). Medialab LPR database. http://www.medialab.ntua.gr/research/LPRdatabase.html.

Ayala-Raggi, S. E., Altamirano-Robles, L., and Cruz-Enriquez, J. (2008). Towards an illumination-based 3D active appearance model for fast face alignment. In *Progress in Pattern Recognition, Image Analysis and Applications*, pages 568–575.

Baker, S. and Kanade, T. (1999). Super-resolution optical flow. Technical Report CMU-RI-TR-99-36, CMU.

Baker, S. and Kanade, T. (2000). Hallucinating faces. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, page 83.

Baker, S. and Kanade, T. (2002). Limits on super-resolution and how to break them. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(9), 1167 – 1183.

Barreto, D., Alvarez, L. D., and Abad, J. (2005). Motion estimation techniques in super-resolution image reconstruction. a performance evaluation. *COST A283 MC meeting & Workshop VIRTUAL OBSERVATORY: "Plate Content Digitization, Archive Mining and Image Sequence Processing"*.

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

Basri, R. and Jacobs, D. W. (2003). Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(2), 218–233.

Belhumeur, P. N. and Kriegman, D. J. (1998). What is the set of images of an object under all possible illumination conditions? *International Journal of Computer Vision*, **28**(3), 245–260.

Belhumeur, P. N., Hespanha, Jo a. P., and Kriegman, D. J. (1997). Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**(7), 711–720.

Ben-Ezra, M., Zomet, A., and Nayar, S. K. (2005). Video super-resolution using controlled subpixel detector shifts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(6), 977–987.

Blanz, V. and Vetter, T. (1999). A morphable model for the synthesis of 3D faces. In A. Rockwood, editor, *Siggraph 1999, Computer Graphics Proceedings*, pages 187–194, Los Angeles. Addison Wesley Longman.

Blanz, V. and Vetter, T. (2003). Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(9), 1063–1074.

Blanz, V., Grother, P., Phillips, P. J., and Vetter, T. (2005). Face recognition based on frontal views generated from non-frontal images. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 454–461, Washington, DC, USA. IEEE Computer Society.

Bledsoe, W. W. (1966). Man-machine facial recognition. Technical report, Panoramic Res. Inc., CA.

Borman, S. and Stevenson, R. L. (1998). Super-resolution from image sequences - a review. In *MWSCAS '98: Proceedings of the 1998 Midwest Symposium on Systems and Circuits*, pages 374–378, Washington, DC, USA. IEEE Computer Society.

Bowyer, K. W., Chang, K., and Flynn, P. (2004). A survey of approaches to three-dimensional face recognition. *International Conference on Pattern Recognition*, **01**, 358–361.

Bowyer, K. W., Chang, K., and Flynn, P. (2005). A survey of approaches and challenges in 3d and multi-modal 3d + 2d face recognition. *Computer Vision and Image Understanding*, **101**, 1–15.

Cai, D. (As retrieved in October 2007). Codes and datasets for subspace learning. http://www.cs.uiuc.edu/homes/dengcai2/Data/data.html.

Cascia, M. L., Sclaroff, S., and Athitsos, V. (2000). Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(4), 322–336.

Chan, M., Delmas, P., Gimelfarb, G., and Leclercq, P. (2002). Comparative study of 3d face acquisition techniques. Technical Report CITR-TR-159, CITR, The University of Auckland, 2002.

Chang, S., Chen, L., Chung, Y., and Chen, S. (2004). Automatic license plate recognition. *Intelligent Transportation Systems*, **5**(1), 42–53.

Chellappa, R., Wilson, C., and Sirohey, S. (1995). Human and machine recognition of faces: a survey. *Proceedings of the IEEE*, **83**(5), 705–741.

Chen, H., Belhumeur, P., and Jacobs, D. (2000). In search of illumination invariants. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 254–261.

Chiang, M. and Boult, T. (2000). Efficient super-resolution via image warping. *Image and Vision Computing*, **18**(10), 761–771.

Chiang, M.-C. and Boult, T. E. (1996). Efficient image warping and super-resolution. In *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*, page 56, Washington, DC, USA. IEEE Computer Society.

Cootes, T. and Taylor, C. (1999). Statistical models of appearance for computer vision. Technical report, University of Manchester, Wolfson Image Analysis Unit, Imaging Science and Biomedical Engineering, Manchester M13 9PT, United Kingdom.

Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. (1995). Active shape models—their training and application. *Computer Vision and Image Understanding*, **61**(1), 38–59.

Dalley, G., Freeman, B., and Marks, J. (2004). Single-frame text super-resolution: a Bayesian approach. In *IEEE International Conference on Image Processing*, pages 3295–3298.

Dedeoglu, G., Baker, S., and Kanade, T. (2006). Resolution-aware fitting of active appearance models to low-resolution images. In *European Conference on Computer Vision*, volume 2, pages 83–97. Springer.

Dellaert, F., Thorpe, C., and Thrun, S. (1998). Super-resolved texture tracking of planar surface patches. In *IEEE/RSJ International Conference on Intelligent Robotic Systems*.

Deutscher, J. and Reid, I. (2005). Articulated body motion capture by stochastic search. *International Journal of Computer Vision*, **61**(2), 185–205.

Dimitrijevic, M., Ilic, S., and Fua, P. (2004). Accurate face models from uncalibrated and ill-lit video sequences. Technical report, EPFL I&C, 1015 Lausanne.

Dornaika, F. and Ahlberg, J. (2006). Fitting 3D face models for tracking and Active Appearance Model training. *Image and Vision Computing*, **24**(9), 1010–1024.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, **10**(3), 197–208.

Draper, B., Baek, K., Bartlett, M., and Beveridge, J. (2003). Recognizing faces with PCA and ICA. *Computer Vision and Image Understanding*, **91**(1-2), 115–137.

Edwards, G. J., Taylor, C. J., and Cootes, T. F. (1998). Interpreting face images using Active Appearance Models. In *FG '98: Proceedings of the 3rd. International Conference on Face & Gesture Recognition*, page 300, Washington, DC, USA. IEEE Computer Society.

Einsele, F., Ingold, R., and Hennebert, J. (2008). Recognition of ultra low resolution word images using HMMs. In *Computer Recognition Systems 2*, pages 429–436. Springer Verlag.

Etemad, K. and Chellappa, R. (1996). Face recognition using discriminant eigenvectors. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 2148–2151.

Farsiu, S., Robinson, D., Elad, M., and Milanfar, P. (2004). Advances and challenges in super-resolution. *International Journal of Imaging Systems and Technology*, **14**(2), 47–57.

Faugera, O. (1993). *Three-Dimensional Computer Vision - A Geometric Viewpoint.* MIT Press, Cambridge, MA, USA.

Gao, C. and Ahuja, N. (2006). A refractive camera for acquiring stereo and super-resolution images. *IEEE Conference on Computer Vision and Pattern Recognition*, **2**, 2316–2323.

Gautama, T. and van Hulle, M. (2002). A phase-based approach to the estimation of the optical flow field using spatial filtering. *IEEE Transactions on Neural Networks*, **13**(5), 1127–1136.

Georghiades, A., Belhumeur, P., and Kriegman, D. (2001). From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(6), 643–660.

Goldenstein, S., Vogler, C., and Metaxas, D. (2004a). 3d facial tracking from corrupted movie sequences. *IEEE Conference on Computer Vision and Pattern Recognition*, **1**, 880–885.

Goldenstein, S., Vogler, C., and Velho, L. (2004b). Adaptive deformable models. *Brazilian Symposium on Computer Graphics and Image Processing*, **0**, 380–387.

Goldenstein, S. K., Vogler, C., and Metaxas, D. (2003). Statistical cue integration in DAG deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**, 801–813.

Hallinan, P. W. (1994). A low-dimensional representation of human faces for arbitrary lighting conditions. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 995–999.

Hamarneh, G. (As retrieved in February 2008). Multi-resolution active shape models. http://www.cs.sfu.ca/~hamarneh/software/asm/index.html.

Hu, Y., Jiang, D., Yan, S., Zhang, L., and Zhang, H. (2004). Automatic 3d reconstruction for face recognition. *IEEE International Conference on Automatic Face and Gesture Recognition*, page 843.

Huang, T. S. and Tsai, R. Y. (1984). Multi-frame image restoration and registration. In *Advances in Computer Vision and Image Processing*, pages 317–339.

Husken, M., Brauckmann, M., Gehlen, S., and der Malsburg, C. V. (2005). Strategies and benefits of fusion of 2D and 3D face recognition. *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, **0**, 174.

Jacobs, C., Simard, P. Y., Viola, P., and Rinker, J. (2005). Text recognition of low-resolution document images. In *ICDAR '05: Proceedings of the Eighth International Conference on Document Analysis and Recognition*, pages 695–699, Washington, DC, USA. IEEE Computer Society.

Jia, K. and Gong, S. (2006). Multi-resolution patch tensor for facial expression hallucination. *IEEE Conference on Computer Vision and Pattern Recognition*, **1**, 395–402.

Jiao, J., Ye, Q., and Huang, Q. (2009). A configurable method for multi-style license plate recognition. *Pattern Recognition*, **42**(3), 358 – 369.

Jung, K., Kim, K., and Jain, A. (2004). Text information extraction in images and video: a survey. *Pattern Recognition*, **37**(5), 977–997.

Kim, K., Jung, K., and Kim, H. (2002). Face recognition using kernel principal component analysis. *IEEE Signal Processing Letters*, **9**(2), 40–42.

Kitagawa, G. (1987). Non-Gaussian state-space modeling of nonstationary time series. *Journal of the American Statistical Association*, **82**(400), 1032–1041.

Kosarev, E. L. (1990). Shannon's superresolution limit for signal recovery. *Inverse Problems*, **6**(1), 55–76.

Lee, H., Chen, S., and Wang, S. (2004). Extraction and recognition of license plates of motorcycles and vehicles on highways. In *International Conference on Pattern Recognition*, pages IV: 356–359.

Lee, K., Ho, J., and Kriegman, D. (2001). Nine points of light: Acquiring subspaces for face recognition under variable lighting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 519–526.

Lee, K.-C., Ho, J., and Kriegman, D. J. (2005). Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(5), 684–698.

Lin, Z. and Shum, H.-Y. (2004a). Fundamental limits of reconstruction-based superresolution algorithms under local translation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(1), 83–97.

Lin, Z. and Shum, H.-Y. (2004b). Fundamental limits of reconstruction-based superresolution algorithms under local translation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(1), 83–97.

Liu, W., Lin, D., and Tang, X. (2005). Hallucinating faces: Tensor patch super-resolution and coupled residue compensation. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 478–484. IEEE Comp. Society.

Lu, L., Zhang, Z., Shum, H.-Y., Liu, Z., and Chen, H. (2001). Model- and exemplar-based robust head pose tracking under occlusion and varying expression. In *In Proceedings of the IEEE Workshop on Models versus Exemplars in Computer Vision*, Kauai, Hawaii.

Lu, X., Jain, A. K., and Colbry, D. (2006). Matching 2.5d face scans to 3d models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**, 31–43.

MacKay, D. J. C. (1998). Introduction to Monte Carlo methods. In *Proceedings of the NATO Advanced Study Institute on Learning in graphical models*, pages 175–204, Norwell, MA, USA. Kluwer Academic Publishers.

Messer, K., Kittler, J., Sadeghi, M., Marcel, S., Marcel, C., Bengio, S., Cardinaux, F., Sanderson, C., Czyz, J., Vandendorpe, L., Srisuk, S., Petrou, M., Kurutach, W., Kadyrov, A., Paredes, R., Kadyrov, E., Kepenekci, B., Tek, F., Akar, G. B., Mavity, N., and Deravi, F. (2003). Face verification competition on the XM2VTS database. In *In 4th International Conference on Audio and Video Based Biometric Person Authentication*, pages 964–974.

Metaxas, D. N., Venkataraman, S., and Vogler, C. (2004). Image-based stress recognition using a model-based dynamic face tracking system. *International Conference on Computational Science*, **1**.

Moghaddam, B., Jebara, T., and Pentland, A. (2000). Bayesian face recognition. *Pattern Recognition*, **33**(11), 1771 – 1782.

Moré, J. J. (1977). The Levenberg-Marquardt algorithm: Implementation and theory. In G. A. Watson, editor, *Numerical Analysis*, pages 105–116. Springer, Berlin.

Nordstrøm, M. M., Larsen, M., Sierakowski, J., and Stegmann, M. B. (2004). The IMM face database - an annotated dataset of 240 face images. Technical report, Technical University of Denmark, DTU.

Pakstas, A. (2002). *MPEG-4 Facial Animation: The Standard, Implementation and Applications*. John Wiley & Sons, Inc., New York, NY, USA.

Park, I. K., Zhang, H., Vezhnevets, V., and Choh, H.-K. (2004). Image-based photorealistic 3-d face modeling. In *Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 49–54.

Park, S. C., Park, M. K., and Kang, M. G. (2003). Super-resolution image reconstruction: a technical overview. *IEEE Signal Processing Magazine*, **20**, 21–36.

Ramamoorthi, R. (2002). Analytic PCA construction for theoretical analysis of lighting variability in images of a Lambertian object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(10), 1322–1333.

Rana, S. (2009). *Multilinear Analysis of Face Image Ensembles*. Ph.D. thesis, Department of Computing, Curtin University of Technology. to be published.

Roussel, R. and Gagalowicz, A. (2005). A hierarchical face behavior model for a 3D face tracking without markers. In *Computer Analysis of Images and Patterns*, volume 3691, pages 854–861. Springer.

S. Rice, F. J. and Nartker, T. (1996). The fifth annual test of OCR accuracy. Technical report, UNLV Information Science Research Inst.

Scheenstra, A., Ruifrok, A., and Veltkamp, R. C. (2005). A survey of 3d face recognition methods. In *Audio- and Video-Based Biometric Person Authentication*, pages 891–899.

Shashua, A. and Riklin-Raviv, T. (2001). The quotient image: Class-based re-rendering and recognition with varying illuminations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(2), 129–139.

Shi, J. and Tomasi, C. (1994). Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 593 – 600, Seattle.

Sirovich, L. and Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America. A, Optics and image science*, **4**(3), 519–524.

Smelyanskiy, V., Cheeseman, P., Maluf, D., and Morris, R. (2000). Bayesian super-resolved surface reconstruction from images. *IEEE Conference on Computer Vision and Pattern Recognition*, **1**, 375–382.

Sun, J., Hotta, Y., Katsuyama, Y., and Naoi, S. (2005). Camera based degraded text recognition using grayscale feature. In *Eighth International Conference on Document Analysis and Recognition*, volume 1, pages 182–186.

Suresh, K., Kumar, G., and Rajagopalan, A. (2007). Superresolution of license plates in real traffic videos. *IEEE Transactions on Intelligent Transportation Systems*, **8**(2), 321–331.

Tan, X., Chen, S., Zhou, Z.-H., and Zhang, F. (2006). Face recognition from a single image per person: A survey. *Pattern Recognition*, **39**(9), 1725–1745.

Tanaka, M. and Okutomi, M. (2005). Theoretical analysis on reconstruction-based super-resolution for an arbitrary PSF. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 947–954, Washington, DC, USA. IEEE Computer Society.

Tang, H. and Huang, T. S. (2008). Mpeg4 performance-driven avatar via robust facial motion tracking. In *IEEE International Conference on Image Processing (ICIP'08).*

Tao, H. and Huang, T. (1999). Explanation-based facial motion tracking using a piecewise bezier volume deformation model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages I: 611–617.

Thouin, P. D. and Chang, C.-I. (2000). A method for restoration of low-resolution document images. *International Journal on Document Analysis and Recognition*, **2**(4), 200–210.

Trier, O., Jain, A., and Taxt, T. (1996). Feature-extraction methods for character-recognition: A survey. *Pattern Recognition*, **29**(4), 641–662.

Tu, J., Zhang, Z., Zeng, Z., and Huang, T. (2004). Face localization via hierarchical Condensation with Fisher boosting feature selection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages II: 719–724.

Tu, J., Huang, T., and Tao, H. (2006). Accurate head pose tracking in low resolution video. In *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, pages 573–578, Washington, DC, USA. IEEE Computer Society.

Turk, M. and Pentland, A. (1991). Face recognition using eigenfaces. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–518.

Wang, C., Yan, S., Zhang, H., and Ma, W. (2005a). Realistic 3d face modeling by fusing multiple 2d images. *International Multimedia Modelling Conference (MMM'05)*, **00**, 139–146.

Wang, S., Zhang, L., and Samaras, D. (2005b). Face reconstruction across different poses and arbitrary illumination conditions. In *Audio- and Video-Based Biometric Person Authentication*, pages 91–101.

Wang, Y., Gupta, M., 0002, S. Z., Wang, S., Gu, X., Samaras, D., and Huang, P. (2005c). High resolution tracking of non-rigid 3d motion of densely sampled data using harmonic maps. In *Tenth IEEE International Conference on Computer Vision*, volume 1, pages 388–395.

Wen, Z. and Huang, T. (2005). Enhanced 3D geometric-model-based face tracking in low resolution with appearance model. *International Conference on Image Processing*, **2**, II– 350–3.

Wikipedia (As retrieved in October 2009). Vehicle registration plates of germany. http://en.wikipedia.org/wiki/German_car_number_plates.

Wu, H.-H. P., Chen, H.-H., Wu, R.-J., and Shen, D.-F. (2006). License plate extraction in low resolution video. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 824–827, Washington, DC, USA. IEEE Computer Society.

Xiao, J., Baker, S., Matthews, I., and Kanade, T. (2004). Real-time combined 2D+3D active appearance models. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 535 – 542.

Xin, L., Wang, Q., Tao, J., Tang, X., Tan, T., and Shum, H. (2005). Automatic 3d face modeling from video. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*, pages 1193–1199, Washington, DC, USA. IEEE Computer Society.

Yan, S., Li, M., Zhang, H., and Cheng, Q. (2003). Ranking prior likelihood distributions for bayesian shape localization framework. In *International Conference on Computer Vision*. IEEE.

Yang, J., Zhang, D., Frangi, A. F., and Yang, J.-y. (2004). Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**(1), 131–137.

Yang, M., Kriegman, D., and Ahuja, N. (2002). Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(1), 34–58.

Yu, J. and Bhanu, B. (2006). Super-resolution restoration of facial images in video. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 342–345, Washington, DC, USA. IEEE Computer Society.

Yuan, J., Du, S., and Zhu, X. (2008). Fast super-resolution for license plate image reconstruction. In *International Conference on Pattern Recognition*, pages 1–4.

Zhang, H., Jia, W., He, X., and Wu, Q. (2006). Learning-based license plate detection using global and local features. In *International Conference on Pattern Recognition*, pages 1102–1105.

Zhang, L. and Samaras, D. (2006). Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**(3), 351–363.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(11), 1330–1334.

Zhang, Z., Liu, Z., Adler, D., Cohen, M. F., Hanson, E., and Shan, Y. (2004). Robust and rapid generation of animated faces from video images: A model-based modeling approach. *International Journal Computer Vision*, **58**, 93–119.

Zhao, M., Chua, T.-S., and Sim, T. (2006). Morphable face reconstruction with multiple images. In *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, pages 597–602, Washington, DC, USA. IEEE Computer Society.

Zhao, W. and Sawhney, H. S. (2002). Is super-resolution with optical flow feasible? In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 599–613, London, UK. Springer-Verlag.

Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys*, **35**(4), 399–458.

Zheng, D., Zhao, Y., and Wang, J. (2005). An efficient method of license plate location. *Pattern Recognition Letters*, **26**(15), 2431–2438.

Zhigljavsky, A. A. (1991). *Theory of global random search.* Dordrecht Netherlands : Kluwer Academic Publishers.

Zhou, S., Chellappa, R., and Jacobs, D. (2004). Characterization of human faces under illumination variations using rank, integrability, and symmetry constraints. In *European Conference on Computer Vision*, pages 588–601.

Zorin, D. and Schroder, P. (2000). Subdivision for modeling and animation. In *Subdivision for modeling and animation*, number 36 in Computer Graphics and Interactive Techniques. ACM Siggraph.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear from any copyright owner who has been omitted or incorrectly acknowledged.

Appendix A

Face Mask Subdivision

A number of different subdivision algorithms have been proposed in the literature (Zorin and Schroder, 2000). The following two have been selected for the use within this thesis:

- Loop Subdivision
- Modified Butterfly Subdivision

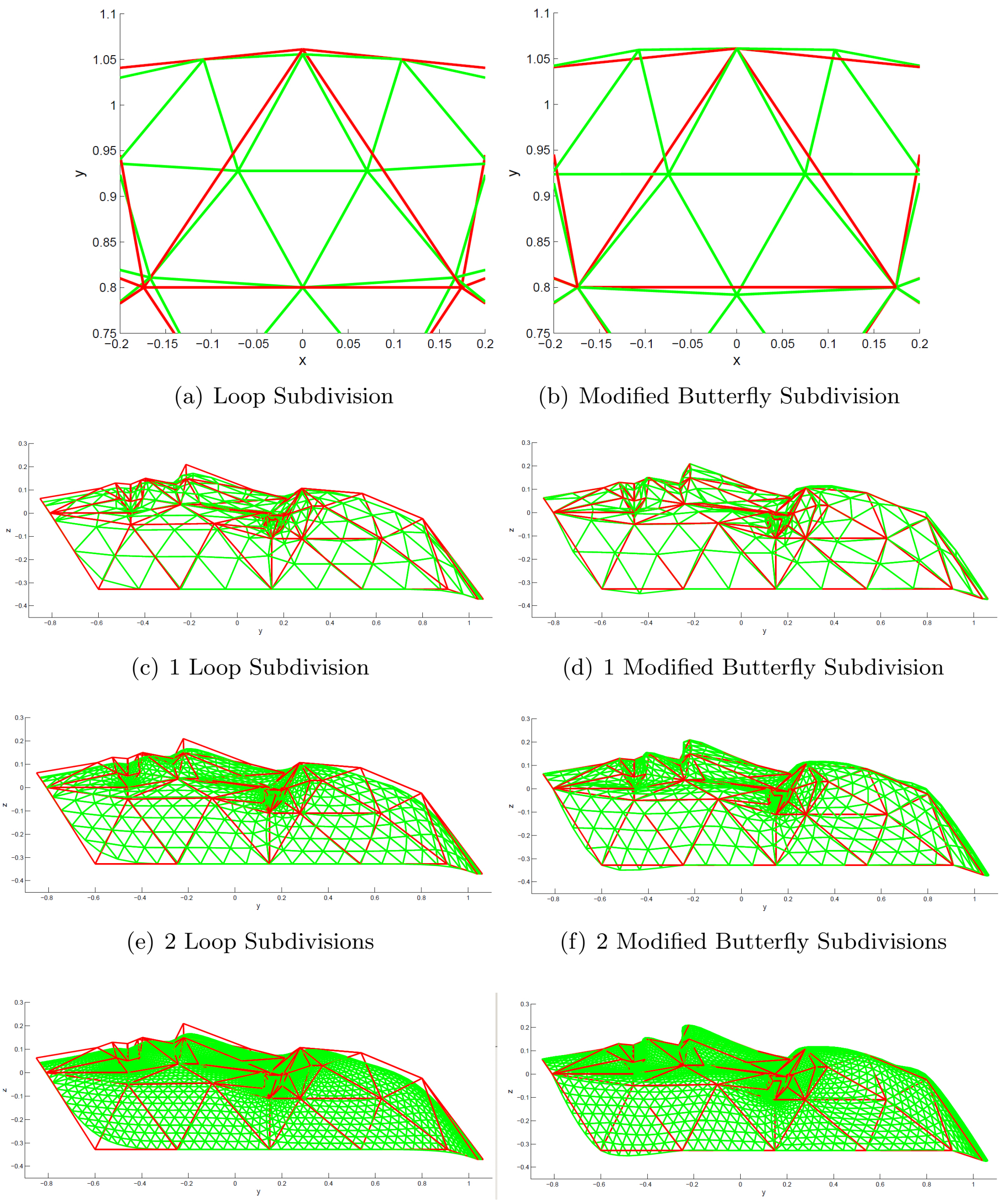
Both methods are face-split schemes for triangular meshes. Each existing surface triangle is subdivided into four smaller triangles by inserting three new vertex points in the centre of each triangle edge. Special rules apply for boundary and extraordinary vertices (Zorin and Schroder, 2000). Figures A.1(a) and A.1(b) show the top section of the CANDIDE-3 mesh (green) and the result after one subdivision (red) using the Loop and the Modified Butterfly scheme respectively.

The advantage that both subdivision schemes offer is that the deformability of the original CANDIDE-3 mask can be transferred onto the subdivided model. The deformation matrix **S** in Equation 3.1 contains a list of vertices and their displacement which are controlled by the parameter γ. During the subdivision each triangle is split into four triangles and new vertex points are added. These new vertex points may then be added to the deformation matrix **S** to ensure the deformability of the new subdivided and finer mask.

The main difference between the two subdivision schemes is the calculation of the new vertex points after each triangle is split into four. The Loop subdivision scheme is an approximating scheme based on splines and, thus produces piecewise polynomial surfaces, whereas the Modified Butterfly Subdivision scheme is interpolating which means that the position of the old vertex points remains unchanged.

The result after one, two and three subdivisions of the CANDIDE-3 face mask using the Loop and the Modified Butterfly subdivision scheme is shown in Figure A.1. The Modified Butterfly scheme preserves the location of all mask vertices and only interpolates the position of the inserted vertex points, whereas the Loop subdivision smooths the entire mask using spline approximation and none of the original vertex points are preserved.

To achieve a subdivided CANDIDE-3 mask with a smooth surface while keeping close to the position of the original CANDIDE-3 vertex points both subdivision schemes are used. The original mask is subdivided using the Modified Butterfly Subdivision scheme first and then subdivided twice using Loop Subdivision scheme. The result is shown in Figure A.2.



(a) Loop Subdivision

(b) Modified Butterfly Subdivision

(c) 1 Loop Subdivision

(d) 1 Modified Butterfly Subdivision

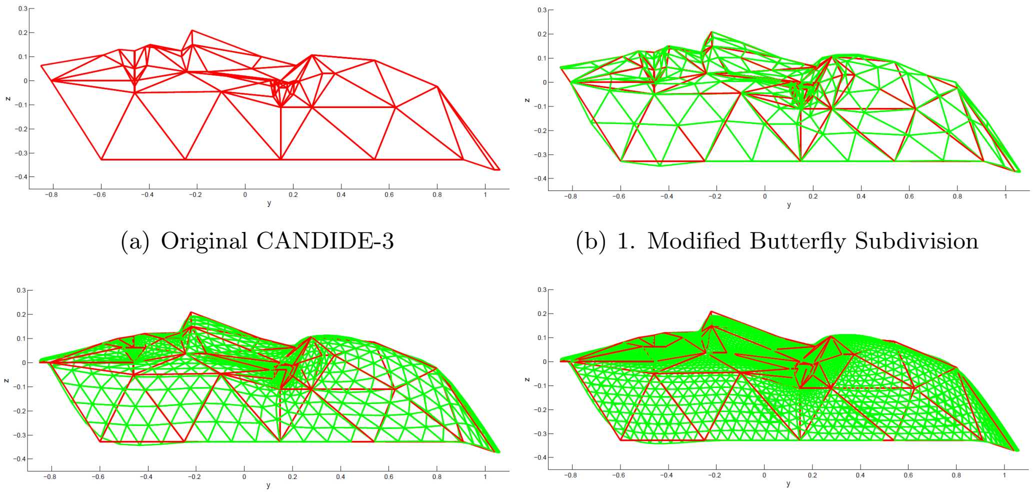
(e) 2 Loop Subdivisions

(f) 2 Modified Butterfly Subdivisions

(g) 3 Loop Subdivisions

(h) 3 Modified Butterfly Subdivisions

Figure A.1: The Loop subdivision scheme and the Modified Butterfly subdivision scheme are used to subdivide the CANDIDE-3 mask. The result after one, two and three subdivisions is shown in figures (c)-(h). Figure (a) and (b) show the top section of the original CANDIDE-3 mask (red) as well as the result after one subdivision (green) to visualise the different between the two subdivision schemes.



(a) Original CANDIDE-3

(b) 1. Modified Butterfly Subdivision

(c) 2. Loop Subdivision

(d) 3. Loop Subdivision

Figure A.2: For this thesis the original CANDIDE-3 face mask (a) is subdivided using the Modified Butterfly Subdivision scheme (b), followed by two Loop subdivisions (c),(d).