Exemplar-Based Face Image Parsing
with Landmark- and Segment-Based Representations

by

Brandon M. Smith

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2014

Date of final oral examination: 05/12/2014

The dissertation is approved by the following members of the Final Oral Committee:

Li Zhang, Assistant Professor, Computer Sciences
Charles Dyer, Professor, Computer Sciences
Michael Gleicher, Professor, Computer Sciences
Vikas Singh, Associate Professor, Biostatistics and Medical Informatics
Jonathan Brandt, Principal Scientist, Adobe Systems Incorporated

Dedicated to my family.

Acknowledgments

In late August 2007, new to Madison, I went to Michael Gleicher's office for some advice about courses that Fall. I arrived inclined to enroll in *CS 540: Introduction to Artificial Intelligence*. Mike suggested that I get my feet wet and instead take *CS 776: Introduction to Computer Vision*, to be taught by Li Zhang, a new assistant professor. I followed Mike's advice, which turned out to be wise. At the end of the semester, to my surprise, Li offered me a job as his first graduate research assistant. I am very grateful that he took on the significant challenge of guiding me toward a Ph.D. over the past 6+ years. I have learned many lessons from Li, including the surprising amount of time, effort, and scrupulousness required to do good work, from building state-of-the-art systems to preparing for conference talks.

I am thankful to my doctoral committee members: Charles Dyer, Michael Gleicher, Vikas Singh, and Jonathan Brandt. Chuck has always been very kind and helpful, and interested in my work. On two occasions he invited me to talk about my research in his computational photography course, which I appreciate. Mike's dissertation advice, which can be found on his website, was very valuable during the writing process. At a conference in Japan in 2009, Vikas shared a useful sentiment with me about giving conference talks: Don't worry. You are the world's expert on your paper. Years later, I still derive confidence from that sentiment. Vikas also introduced me to Steven Seitz on that trip; Steve was Li's Ph.D. advisor and Chuck's Ph.D. advisee. Jon was my manager at Adobe during my first internship there in 2009, and again in 2013. He and Zhe Lin were my internship mentors in 2013. We have collaborated on multiple projects since mid-2012, and my research is undoubtedly better because of it.

I am grateful to my family for cheering me on and for supporting my ambitions throughout the years. My late mother, Trudy Janssen, was my biggest fan. She encouraged my interests and taught me the value of a good laugh. My late father,

Stefan Smith, like his father, was an avid tinkerer who taught me how to fix and built a multitude of things. My younger sister Alicia has always been astute and easy to talk to about anything. My grandparents, Fannie Smith and Maynard Smith, never went to college, but they made sure that I did. I am especially thankful to my fiancé, Laurie Stephey, who has supported me in countless ways, large and small, and who has tolerated many late nights and work-filled weekends along the way. I owe her a life of favors. I am also thankful to Laurie's family for being so generous and welcoming.

The members of the Visual Computing Lab (a.k.a. the Graphics Lab) have helped make graduate school an enjoyable time. The Lab is a fun and stimulating—if not always efficient—place to work. I'm also thankful to all of my friends for making Madison a wonderful place, especially Chris Hinrichs, Maxwell Collins, and everyone else who took the occasional Saturday evening off for homebrewing.

I am thankful to the National Science Foundation for providing three years of funding from 2009 to 2012 and to Adobe Systems, Inc. for partially funding my work on face image analysis in 2012 and 2013.

Contents

Contents iv

List of Tables viii

List of Figures ix

Nomenclature xi

Abstract xiii

1 Introduction 1
- *1.1 Motivations* . . . 1
- *1.2 An Exemplar-Based Approach* . . . 4
  - 1.2.1 Face Image Parsing with Landmarks . . . 4
  - 1.2.2 Collaborative Face Image Parsing for Transferring Landmark Annotations Across Datasets . . . 5
  - 1.2.3 Face Image Parsing with Soft Segments . . . 6
- *1.3 Thesis Statement* . . . 6

2 Background 8
- *2.1 Terminology* . . . 8
- *2.2 Very Early Approaches* . . . 9
- *2.3 Early Shape Models* . . . 10
- *2.4 Early Holistic Appearance Models* . . . 12
- *2.5 Recent Parametric Holistic Methods* . . . 14
- *2.6 Recent Parametric Local Methods* . . . 16

2.7 State-of-the-Art Approaches . . . 18
2.7.1 Exemplar-Based Approaches . . . 18
2.7.2 Voting-Based Approaches . . . 20
2.7.3 Regression-Based Approaches that Model Shape Nonparametrically . . . 22
2.7.4 Simultaneous Face Detection and Landmark Localization . . . 23
2.7.5 Face Parsing with Segments and Pixel-wise Labels . . . 24
2.8 Summary and Observations . . . 25

3 Face Image Parsing with Landmarks 28
3.1 Introduction . . . 28
3.2 Related Work . . . 31
3.3 Approach . . . 33
3.3.1 Overview . . . 33
3.3.2 Database Construction . . . 35
3.3.3 Step 1: Top Exemplar Retrieval . . . 35
3.3.4 Step 2: Landmark Voting . . . 36
3.3.5 Step 3: Voting Map Selection . . . 37
3.3.6 Step 4: Final Landmark Estimation . . . 37
3.3.7 Computing Exemplar Feature Weights . . . 39
3.3.8 Implementation Details . . . 43
3.4 Results and Discussion . . . 45
3.4.1 Experimental Datasets . . . 45
3.4.2 Comparisons with Recent Work . . . 45
3.4.3 Runtime . . . 50
3.4.4 Impact of Different Algorithm Parts . . . 51

4 Collaborative Face Image Parsing for Transferring Landmark Annotations Across Datasets 52
4.1 Introduction . . . 52
4.2 Related Work . . . 54
4.3 Approach . . . 55
4.3.1 Overview . . . 55
4.3.2 Stage 1: Selection of Top Source Faces . . . 56

4.3.3 Stage 2: Weighted Landmark Voting 57
4.3.4 Stage 3: Shape Regularization 58
4.3.5 Stage 4: Final Landmark Estimation and Integration 58
4.3.6 Implementation Details and Runtime 61
4.4 Results and Discussion 63
4.4.1 Experimental Datasets 63
4.4.2 Comparisons with Recent Work 64
4.4.3 Evaluation with Known Target Landmarks 67

5 Face Image Parsing with Soft Segments 70
5.1 Introduction 70
5.2 Related Work 73
5.3 Approach 75
5.3.1 Overview 75
5.3.2 Step 1: Nonrigid Exemplar Alignment 78
5.3.3 Step 2: Exemplar Aggregation 79
5.3.4 Step 3: Pixel-wise Label Selection 80
5.4 Results and Discussion 82
5.4.1 Experimental Datasets 82
5.4.2 Accuracy vs. Number of Top Exemplars 83
5.4.3 Comparisons on LFW 83
5.4.4 Comparisons on Helen 86
5.4.5 Comparisons with Liu *et al.* [80] 90
5.4.6 Runtime 90
5.4.7 Extensions of Our Approach 92

6 Conclusions and Future Work 96
6.1 Improving Efficiency 97
6.2 A Unified Pipeline 98
6.3 Additional Segment Types 99
6.4 A Semi-Supervised Approach 100
6.5 Better Exemplar Database Characterization 100
6.6 Occlusion Detection 100
6.7 Future Applications 101

6.8 Other Domains 102

A Supplementary Facial Landmark Localization Results 103

B Supplementary Landmark Transfer Results on AFLW Faces 107

C Supplementary Face Segmentation Results 115

D Summary of Face Image Datasets Used in This Dissertation 118

References 124

List of Tables

2.1 Common laboratory and in-the-wild experimental datasets for face alignment and facial landmark localization 19

5.1 Comparison of facial part segmentation accuracy on LFW images via the confusion matrix 85
5.2 Comparison of facial part segmentation accuracy on LFW images via the F-measure 86
5.3 Face part label weights 86
5.4 Comparison of facial part segmentation accuracy on Helen images via the F-measure 88

6.1 Correspondence between steps in each separate algorithm and steps in a unified pipeline 99

List of Figures

1.1 Challenging face images from recent in-the-wild datasets 3

3.1 Selected facial landmark localization result 29
3.2 Overview of facial landmark localization pipeline 34
3.3 Visualization of landmark-specific feature weights 40
3.4 Distribution of landmark localization feature weights 44
3.5 Selected facial landmark localization results on AFW and IBUG datasets 46
3.6 Accuracy of our landmark localization algorithm compared to other recent algorithms 47
3.7 Evaluation of the impact of different parts of our facial landmark localization pipeline on performance 51

4.1 Overview of pipeline for transferring landmark annotations across face datasets 57
4.2 Experimental datasets used to evaluate our pipeline for transferring landmark annotations across face datasets 64
4.3 Cumulative error distribution curves showing a comparison between the accuracy of our core landmark localization approach and other recent algorithms 65
4.4 Results from our landmark transfer algorithm with varying numbers of known landmarks in the target images 68
4.5 Evaluation of our full pipeline for transferring landmark annotations across face datasets 69

5.1 Selected semantic face image segmentation result 71
5.2 Overview of semantic face image segmentation pipeline 77

5.3 Face segmentation accuracy vs. the number of top exemplars used by our algorithm 84
5.4 Example of the problem encountered in facial part segmentation when optimizing the confusion matrix 85
5.5 Selected facial part segmentation results on LFW and Helen images . . . 89
5.6 Comparison of facial part segmentation results with Liu *et al.* [80] . . . 91
5.7 An extension of our facial part segmentation algorithm: contour estimation 93
5.8 An extension of our facial part segmentation algorithm: hair segmentation 94
5.9 An extension of our facial part segmentation algorithm: face synthesis . . 95

A.1 Selected landmark localization results on the AFW face dataset 104
A.2 Selected results on the IBUG face dataset 105
A.3 Landmark localization failure cases 106

B.1 Landmark transfer results on AFLW faces: partial occlusions 109
B.2 Landmark transfer results on AFLW faces: head rotation 110
B.3 Landmark transfer results on AFLW faces: young and elderly 111
B.4 Landmark transfer results on AFLW faces: non-neutral expressions . . . 112
B.5 Landmark transfer results on AFLW faces: miscellaneous challenging conditions 113
B.6 Landmark transfer results on AFLW faces: failure cases 114

C.1 Additional semantic face image segmentation results 116
C.2 Semantic face image segmentation failure cases on mouths 117

Nomenclature

AAM Active appearance model [22]

AFLW Annotated Facial Landmarks in the Wild dataset [60]

AFW Annotated Faces in the Wild dataset [142]

ASM Active shape model [25]

CLM Constrained local model [33]

k-NN k-nearest neighbors

Landmarks Points of interest in the image that carry semantic meaning, *e.g.*, eye centers, mouth corners, nose tip

LFW Labeled Faces in the Wild dataset [51]

LFPW Labeled Face Parts in the Wild dataset [7]

Face alignment The computer vision problem of aligning a model to a face image (or, equivalently, aligning a face image to a reference model), *i.e.*, in order to locate landmarks in the image

Face image parsing The problem of separating a face image into its constituent semantic parts (*e.g.*, eyes, nose, lips), which encompasses face alignment, facial landmark localization, semantic segmentation, and semantic pixel-wise labeling

Facial landmark localization The computer vision problem of locating landmarks in face images

Features Local appearance descriptors (*e.g.*, SIFT), which, unlike landmarks, do not carry semantic meaning

Helen The Helen face dataset [74]

IBUG Intelligent Behavior Understanding Group face dataset [98]

PDM Point distribution model

RANSAC RANdom SAmple Consensus algorithm [39]

SIFT Scale-Invariant Feature Transform [84]

SVM Support vector machine

EXEMPLAR-BASED FACE IMAGE PARSING WITH LANDMARK- AND SEGMENT-BASED REPRESENTATIONS

Brandon M. Smith

Under the supervision of Assistant Professor Li Zhang
At the University of Wisconsin – Madison

Parsing face images into their constituent parts (*e.g.*, eyes, nose, lips) is an important task in many computer vision applications (*e.g.*, face recognition). Unfortunately, automating this task is difficult in practice due to the wide variety of conditions in the real world. For example, current approaches often fail on non-frontal faces and faces with exaggerated expressions. To better highlight and address real-world challenges, researchers have introduced several "in-the-wild" face datasets in recent years. However, each dataset typically has its own set of facial landmark definitions. As a result, models trained on one dataset often cannot be evaluated on others, and inconsistencies between datasets make it difficult to train robust models on more than one dataset. Another limitation of current approaches is that, with few exceptions, they parse face images using landmarks or contours, which have limited representational power.

To address these limitations, this dissertation proposes an exemplar-based approach to face image parsing that models shape and appearance in a nonparametric way. First, a facial landmark localization algorithm is introduced that combines a nonparametric model of the local appearance context of each landmark with an exemplar-based shape regularization technique. Second, this algorithm is extended to address the problem of automatically transferring landmark definitions across different face datasets. The result is a large dataset in which each image includes a union of all landmark types as output. Third, a pixel-wise labeling algorithm is introduced to parse face images into their constituent parts using a soft segmentation. This dissertation will show that all three algorithms are able to parse challenging in-the-wild face images with state-of-the-art accuracy. The results suggest that an exemplar-based approach that models face shape and local appearance in a nonparametric way is (1) flexible enough to parse faces according to landmark- and segment-based representations, (2) can be used to combine different face datasets, and (3) is well-suited to parsing faces depicted in challenging real-world conditions.

Li Zhang

Abstract

Parsing face images into their constituent parts (*e.g.*, eyes, nose, lips) is an important task in many computer vision applications (*e.g.*, face recognition). Unfortunately, automating this task is difficult in practice due to the wide variety of conditions in the real world. For example, current approaches often fail on non-frontal faces and faces with exaggerated expressions. To better highlight and address real-world challenges, researchers have introduced several "in-the-wild" face datasets in recent years. However, each dataset typically has its own set of facial landmark definitions. As a result, models trained on one dataset often cannot be evaluated on others, and inconsistencies between datasets make it difficult to train robust models on more than one dataset. Another limitation of current approaches is that, with few exceptions, they parse face images using landmarks or contours, which have limited representational power.

To address these limitations, this dissertation proposes an exemplar-based approach to face image parsing that models shape and appearance in a nonparametric way. First, a facial landmark localization algorithm is introduced that combines a nonparametric model of the local appearance context of each landmark with an exemplar-based shape regularization technique. Second, this algorithm is extended to address the problem of automatically transferring landmark definitions across different face datasets. The result is a large dataset in which each image includes a union of all landmark types as output. Third, a pixel-wise labeling algorithm is introduced to parse face images into their constituent parts using a soft segmentation. This dissertation will show that all three algorithms are able to parse challenging in-the-wild face images with state-of-the-art accuracy. The results suggest that an exemplar-based approach that models face shape and local appearance in a nonparametric way is (1) flexible enough to parse faces according to landmark- and segment-based representations, (2) can be used to combine different face datasets, and (3) is well-suited to parsing faces depicted in challenging real-world conditions.

Chapter 1

Introduction

This work focuses on the computer vision task of automatically parsing face images into their constituent facial parts, *e.g.*, mouth corner points, left eyebrow region, chin contour. In this dissertation, *face image parsing* is used as a general term that encompasses face alignment, facial landmark localization, and semantic face image segmentation. Face alignment addresses the problem of aligning a landmark or contour model to a face image (or, conversely, aligning a face image to a reference model). Facial landmark localization seeks to automatically locate predefined facial landmarks (*e.g.*, the nose tip, mouth corners, eye centers) in face images. Semantic face image segmentation separates a face image into its constituent parts (*e.g.*, eyebrows, lips, nose) using a pixel-wise labeling.

1.1 Motivations

Autonomous systems that parse face images play a key role in many applications, including face recognition [138], face tracking and performance-driven animation [58, 94, 136], and portrait editing wizards [8, 44, 125]. These applications are especially important and compelling given the explosion of face image data in recent years. Cameras are cheap, ubiquitous, and easy to use, and many websites allow users to conveniently upload thousands of pictures for free. More than 250 billion photos have been uploaded to Facebook since 2004 [37], and a 2008 study of approximately 4 million images randomly downloaded from several popular online sources found that 25.7% contained faces [65].

Unfortunately, face image parsing remains a vexing computer vision problem in practice despite its importance and popularity in the literature. Designing such systems is extremely challenging if the goal is to imitate human visual perception. Humans are remarkably proficient at recognizing many facial traits, even when face images are extremely low-resolution or when faces are partly occluded, for example. Despite this, many mechanisms used by the human visual channel to interpret faces are still not well understood [107].

Historically, most experimental face datasets have been produced in laboratory settings, where different types of variation are limited. For example, the head pose, illumination, and facial expressions in the CMU PIE (Pose Illumination Expression) dataset [106] are carefully controlled. In contrast, recent "in-the-wild" experimental datasets like LFPW (Labeled Face Parts in the Wild) [7], AFW (Annotated Faces in the Wild) [142], and AFLW (Annotated Facial Landmarks in the Wild) [60] incorporate greater variation and highlight many challenges encountered in the real world, as shown in Figure 1.1. In practice, even state-of-the-art algorithms (*e.g.*, [117, 141, 142]) fail on many in-the-wild face images. For example, they often fail to accurately locate landmarks on faces with exaggerated facial expression or non-frontal head pose. Also, for many state-of-the-art algorithms, initialization can be troublesome in practice, which can cause egregious errors in the final result.

To better cope with the many kinds of variations present in face images from the real world, several recent works have abandoned traditional *parametric* models for more *nonparametric* models of face shape [7, 18, 124] and facial landmark appearance [24, 36, 105]. These nonparametric approaches derive their power from large and diverse collections of training faces.

Intuitively, in order for an algorithm to effectively handle faces in the wild, many of the training faces should exhibit real-world variation, *i.e.*, the training images should also come from in-the-wild datasets. Such datasets have become popular since Belhumeur *et al.*'s [7] influential work in 2011. However, most datasets, which each have their own unique strengths and weaknesses, are developed independently with little coordination or common standard between them. As a result, practitioners must choose a single dataset (or a small number of datasets) for training and evaluation. Models trained on one dataset often cannot be evaluated on other datasets, and inconsistencies between datasets make it difficult to train models that combine multiple datasets.

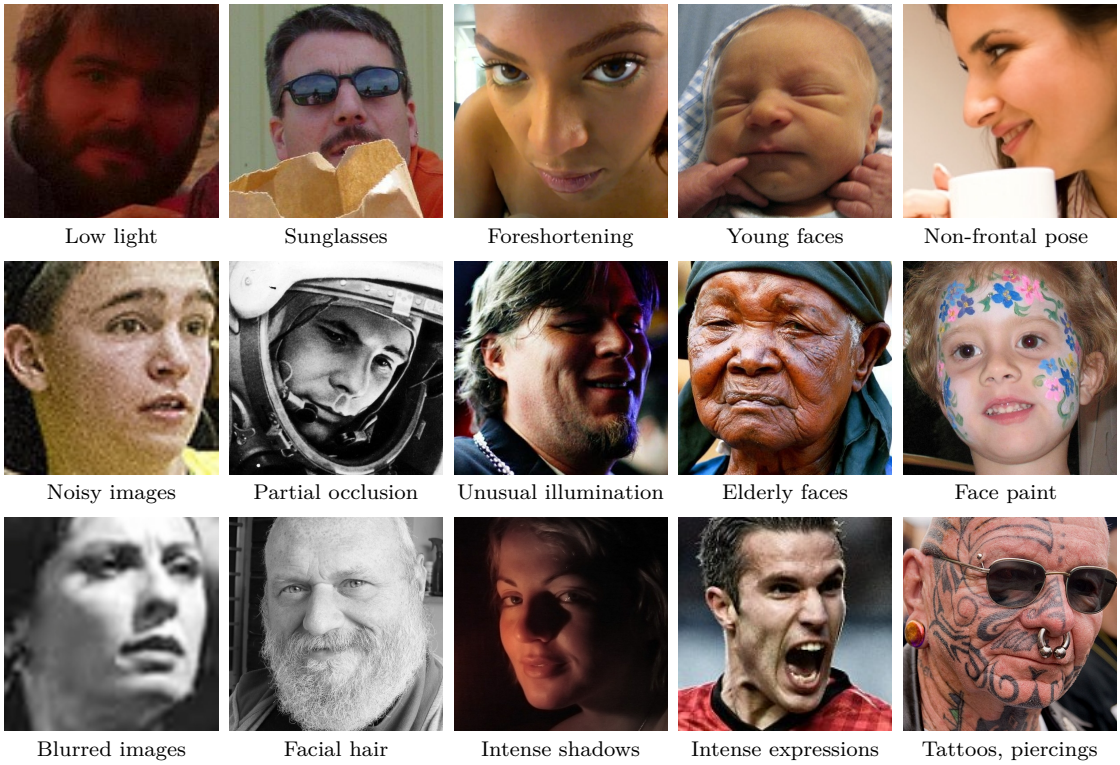


Figure 1.1: The examples above, taken from recent in-the-wild datasets (AFW [142], LFPW [7], IBUG [98], and Helen [74]), highlight many of the challenges present in real-world face images. This work targets these kinds of images.

Ideally, it would be desirable to have a common and unified definition of landmarks and collect datasets following the same definition. However, this goal is challenging in practice because the speed of collecting labels will always lag behind the speed of collecting face data. Furthermore, it is difficult to predict which landmark definitions (*e.g.*, ears) new applications will find useful. Unfortunately, hand annotating is expensive and tedious, which makes large-scale manual efforts impractical. A more practical *automated*, or semi-automated approach is needed to transfer landmark annotations from one dataset to another in a way that uses all of the available information to help guide the transfer, *e.g.*, using the subset of landmark definitions common to multiple datasets.

Landmarks and contours are popular choices for representing face parts because they are simple, efficient to store, and offer computational advantages over other representations. However, their representational power is limited. Landmarks and

contours are not well suited to complex boundaries, *e.g.*, between hair and skin. Some landmark definitions are ambiguous. For example, it is unclear whether a landmark point on the side of the face should be defined relative to the eye, the ear, or by some parameterized position along the chin contour. In addition, the sharp boundaries imposed by contours belie the gradual transition between some face regions, such as the transitions between the upper nose and upper cheeks. A pixel-wise labeling, or soft segmentation, provides a much richer and more general representation of the face. However, few researchers have focused on the problem of accurately and explicitly segmenting faces images into their constituent parts.

1.2 An Exemplar-Based Approach

To address the challenges described above, we propose an exemplar-based approach to face image parsing. We first propose an exemplar-based algorithm that parses face images by estimating facial landmark locations in a nonparametric way. We then extend this algorithm to transfer landmark annotations across different face datasets, which is useful for combining multiple datasets with different landmark definitions into a single dataset, where each face contains a union of all the landmark types. Finally, we propose an exemplar-based algorithm that parses face images via soft semantic segmentation in a nonparametric way.

1.2.1 Face Image Parsing with Landmarks

For facial landmark localization, we generalize and combine a recent exemplar-based approach for shape regularization [7] with an exemplar-based approach for face detection [105] to model the context interactions between facial landmarks and their surrounding local appearance features in a nonparametric way. Specifically, we use a Hough transform-based feature voting scheme to transfer many landmark hypotheses from a large database of exemplar faces to the test image. The votes capture the appearance and geometric correlations between local image features and facial landmarks. Each vote is *weighted*; the weight is precomputed using a data-driven procedure and takes into account each feature's discriminative power. Finally, because

each voting response map may include false peaks, we use an exemplar-based shape regularization technique to select the best arrangement of landmarks.

We show that this approach offers several important advantages over other state-of-the-art approaches, including improved accuracy and robustness on challenging face images (*e.g.*, faces with exaggerated facial expression or extreme head pose). Our algorithm requires only a *rough* bounding rectangle around the face for initialization rather than a full face shape, which is especially important for accurately locating landmarks in unconstrained real-world images. The robustness and flexibility of our approach comes from its ability to efficiently and effectively leverage the information from a large database of face exemplars. Chapter 3 describes our approach to landmark localization in detail.

1.2.2 Collaborative Face Image Parsing for Transferring Landmark Annotations Across Datasets

Researchers introduce new face datasets every year in order to highlight new challenges and advance the state of the art. However, due to lack of coordination between researchers, and because different research projects focus on different problems, each dataset has its own set of landmark definitions. For example, the AFLW [60] dataset includes 21 landmark definitions, whereas the Helen dataset includes 194 contour points. Ideally, we would like to combine multiple datasets to take advantage of the different kinds of information provided by each. This is especially important for exemplar-based approaches like ours, which derive their strength and flexibility from large and diverse collections of exemplar faces.

To address these problems, we make the first effort, to the best of our knowledge, to combine multiple face landmark datasets with different landmark definitions into a single large dataset, in which each image includes a union of all landmark types as output. Specifically, we present a novel pipeline built upon our landmark localization algorithm. Our system labels images in the target dataset collaboratively rather than independently and exploits known landmarks in both the source datasets and the target dataset. Chapter 4 gives a detailed description of this system.

1.2.3 Face Image Parsing with Soft Segments

We argue that, despite the popularity and utility of landmarks and contours in many applications, they have several key limitations compared to segment-based representations. For example,

- Other than eye corners and mouth corners, most landmarks are not well-defined;
- Contour-based representations are not general enough to model some facial parts useful for robust face analysis, including hair strands, teeth, or cheeks; and
- It is difficult to incorporate spatial uncertainty in contour-based representations.

Segment-based representations alleviate these limitations: segments can represent arbitrary facial parts, including hair, teeth, ears, and skin, and soft segments can model uncertain transitions between parts.

We therefore propose an exemplar-based algorithm for parsing face images into their constituent facial parts (*e.g.*, eyes, nose, lips) using a *soft* segmentation. Specifically, our algorithm produces a pixel-wise labeling of face images, in which each pixel is represented by a probability vector of label types. Broadly speaking, our approach to pixel-wise labeling is consistent with our approach to landmark localization (*i.e.*, we propose a exemplar-based approach). Our algorithm is well-suited to segmenting challenging images of faces in real-world conditions. We demonstrate this by producing results with state-of-the-art accuracy on two in-the-wild face datasets. Chapter 5 describes our approach in detail.

1.3 Thesis Statement

This dissertation focuses on three face image parsing problems:

1. Automatically locating landmarks in challenging in-the-wild face images,
2. Combining multiple face datasets with different landmark definitions into a single dataset with a consistent set of landmark definitions, and
3. Using a soft segmentation to separate the face into its constituent parts rather than using contours or landmark points.

We adopt an exemplar-based approach that models face shape *and* local appearance in a nonparametric way to address all three of these problems. For each problem, we evaluate the accuracy of the algorithm on recent in-the-wild face datasets, which include challenging faces with significant yaw, roll, and pitch head rotation, exaggerated expressions, and partial occlusions in images with low illumination, intense shadows, noise, and motion blur, for example.

More succinctly, this dissertation will show that **an exemplar-based approach that models face shape *and* appearance in a nonparametric way can be used to transfer landmarks or segments from an exemplar database to challenging test images (*i.e.*, for facial landmark localization and soft semantic segmentation of facial parts) or between different datasets (*i.e.*, for combining multiple datasets that have different landmark definitions), all with state-of-the-art accuracy.**

Chapter 2

Background

Face image parsing has been studied extensively in the computer vision literature. We forgo a comprehensive survey and instead provide an overview of pioneering, exemplary, and representative works that trace the progression of the state of the art. First, for the sake of clarify, we define some terminology.

2.1 Terminology

The term *scene parsing*, or *image parsing*, typically refers to the problem of assigning a semantic label to each pixel in an image. In this work, we define *face parsing* to refer to the problem of separating a face image into its constituent semantic parts (*e.g.*, eyes, nose, lips) regardless of the representation used to parse the image, *i.e.*, landmarks, contours, pixel-wise labels, or segments.

Most commonly, face parsing algorithms label face parts using landmarks, which carry semantic meaning (*e.g.*, eye centers, mouth corners, nose tip), or contours, which are defined by connected landmarks. Some researchers use *facial features*, *fiducial points* or *keypoints* to refer to landmarks. In this work we reserve the term *feature* to refer to local appearance features (*e.g.*, SIFT [84]) and *landmarks* to refer to semantically meaningful points on the face.

The terms *face alignment* and *facial landmark localization* are sometimes used interchangeably in the literature. Face alignment algorithms iteratively align a model to the face image (or, equivalently, align a face image to a reference model). Once the face image and the face model are aligned, locating the landmarks in the image is

trivial. Face alignment algorithms typically represent the shape of faces using contour landmarks. Facial landmark localization algorithms typically locate fewer landmarks, and they often do not explicitly align a reference model to the face image. However, alignment is trivial given landmark correspondences.

2.2 Very Early Approaches

Computer-aided face recognition started in the mid-1960's with work by Bledsoe on semiautomated face recognition [10, 11, 12, 13]. Like many very early works, Bledsoe sidestepped the difficult task of automatically estimating face shape by requiring manually-entered facial boundaries and landmark coordinates for recognition.

The earliest systems for *automatically* locating facial parts in images were developed in the late 1960s and early 1970s for face recognition. Kelly [57] and later Sakai, Nagao, and Kanade [100] built similar systems that processed images in a top-down manner using part-specific heuristic procedures, *i.e.*, first detect the edges of the head, following by the eyes, nose, and mouth. These very early systems, although pioneering, were very brittle and ad hoc. Nonetheless, improved but substantially similar approaches were used in face recognition systems well into the 1980s, *e.g.*, by Craw *et al.* in 1987 [30].

In 1973, Fischler and Elschlager [40] developed a system for automatically finding a face in an image and simultaneously determining its goodness of fit with a reference model. They modeled the local appearance of face components using ad hoc, manually-coded detector templates. They modeled the global shape of the face by linking the face components with a set of manually-coded 'springs.' Goodness of fit was measured as the sum of the component detector outputs plus the total mount of spring deformation between the reference model and the 'sensed' object. They described a "linear embedding algorithm" for optimizing the goodness of fit. Despite its simplicity and reliance on heuristics rather than statistics derived from training data, their system is noteworthy in that its basic framework (*i.e.*, combining the output of local part detectors with a global shape model) is still used today.

In 1981, Baron [6] described one of the first data-driven approaches for locating face parts in images. Baron built eye templates derived from actual image data rather than manually coding them. At runtime his system detected eyes by exhaustively

scanning each template over the test image and thresholding the cross-correlation value at each location.

In 1985, Nixon [92] created an eye detector based on an analytical shape model of the iris and sclera (white region). His system used the Hough transform [50] to locate the eyes and estimate their shape parameters, which included the iris center and radius and the major and minor axes of the sclera ellipse. Nixon's approach is noteworthy for being one of the first to model face parts *parametrically*.

2.3 Early Shape Models

In the 1980s, elastic models revolutionized the way images were deformed, registered, characterized, and matched. In 1981, Burr developed elastic models for nonrigidly registering pairs of shapes [15, 16]. Several years later, Kass *et al.* proposed Active Contour Models (a.k.a. 'Snakes') [56], which deform and stretch to fit image features to which they are attracted via an energy minimization technique. Active Contour Models are similar to elastic models, except they employ splines for regularization, which improve their robustness to noise and local ambiguities.

One weakness of Active Contour Models is that they do not contain domain-specific structure. Yuille *et al.* [131, 132, 133] addressed this weakness by constructing a separate hand-build deformable template for each face part. Their model for the eye is similar to Nixon's analytic eye model [92] with several additional parameters (*e.g.*, orientation angle, iris offset). These models are limited in that they assume an overly-idealized representations of the face, and they do not generalize well to arbitrary face parts or expressions.

Craw *et al.* [29, 31, 32] proposed using eye templates (similar to Yuille *et al.*) and additional "feature experts" (*ad hoc* procedures) to detect face parts, including the hairline, nose, jawline, *etc.* Most importantly, their system uses statistics (spatial mean and variance) from a collection of training faces to constrain the placement of face parts in the test image.

Wiskott *et al.* [121, 122] later introduced elastic Face Bunch Graphs (FBGs). FBGs are similar in some ways to Craw *et al.*'s system. For example, FBGs employ a graph over the face, which is trained to impose constraints on the range of allowable deformations. However, Wiskott *et al.*'s "local experts" are more principled and

general than Craw *et al.*'s feature experts. The model for each landmark is a "bunch" of Gabor wavelets extracted from different training faces. This approach better adapts to local appearance variation due to head pose, expression, illumination, *etc.*

The aforementioned systems have several key weaknesses. Yuille *et al.*'s deformable templates require hand-built models and individually tailored optimization techniques. Craw *et al.*'s system, although novel for determining the feasibility of solutions by imposing statistical constraints, nonetheless has many heuristic parts and relies on simulated annealing for optimization, which is very slow. Wiskott *et al.*'s system exhaustively searches over the parameter space (scale and aspect ratio) to find the best shape fit, which is slow and suboptimal. Active Contour Models, in addition to not incorporating domain-specific structure, enforce only simple local shape constraints, which do not prevent global errors.

To address these weaknesses, Cootes *et al.* introduced Active Shape Models (ASMs) (a.k.a. "Smart Snakes") [23, 25, 28]. ASMs represent an object's shape X as a vector of N landmark points:

$$X = [\mathbf{x}_1{}^\mathsf{T}, \mathbf{x}_2{}^\mathsf{T}, \cdots, \mathbf{x}_N{}^\mathsf{T}]^\mathsf{T}, \tag{2.1}$$

where $\mathbf{x} = [x, y]^\mathsf{T}$. X is modeled as a mean shape vector X_0 plus a weighted combination of M shape eigenvectors X_m:

$$\begin{aligned} X &= X_0 + \sum_m^M p_m X_m \\ &= X_0 + \mathbf{\Phi}\mathbf{p}, \end{aligned} \tag{2.2}$$

where $\mathbf{p} = [p_1, p_2, \ldots, p_M]^\mathsf{T}$ are the model parameters. The eigenvectors $\mathbf{\Phi} = [X_1, X_2, \cdots, X_M]$ are computed using principle component analysis (PCA) [48, 49, 93] across a set of aligned training shapes. The first few eigenvectors capture the most significant modes of variation among the training shapes, so typically only a few eigenvectors are included[1] (*i.e.*, M is typically much smaller than the total number of training shapes). X_0 and $\mathbf{\Phi}$ act as powerful shape priors. A set of *global* transformation parameters (scale s, rotation R, and translation $\mathbf{t}$) are usually included in addition to the *local* shape deformation parameters $\mathbf{p}$ to account for the placement

[1] The rule of thumb is to keep enough eigenvectors to retain 95% of the variance.

and orientation of the object in the image. In the context of shape modeling, X_0, $\mathbf{\Phi}$, $\mathbf{p}$ and $\{s, R, \mathbf{t}\}$ together form a Point Distribution Model (PDM) [27]. The task of ASMs is to compute the set of shape parameters $\{\mathbf{p}, s, R, \mathbf{t}\}$ in the PDM that best fit the object in the image.

ASMs and PDMs are attractive because they model shapes linearly (which allows for relatively simple mathematics), they represent the modes of shape variation of a class of objects (*e.g.*, faces) compactly, and they prevent 'implausible' shapes from occurring. Although ASMs are often very effective, they are not perfect. For example, ASMs use only local image information near the landmarks for fitting, which can result in mistakes due to local ambiguities, and, to simplify the optimization, they take a first order Taylor expansion of the PDMs, which can lead to imperfect results. Many subsequent works have improved the optimization strategy of ASMs, and the basic PDM framework remains popular, as discussed in Section 2.6.

2.4 Early Holistic Appearance Models

Prior to the late 1980s, most work focused on detecting individual face parts such as the eyes, mouth, and nose, and characterizing the face based on the geometric relationships among these parts. In the late 1980s and early 1990s, *holistic* approaches came to prominence starting with work by Sirovich and Kirby [59, 108] and then Turk and Pentland [113, 116, 114, 115].

Sirovich and Kirby [59, 108] were the first to propose using the Karhunen-Loève expansion [54, 83] (a.k.a. principal component analysis (PCA) [48, 49, 93]) to characterize human face images. Starting with a collection of carefully photographed faces (to ensure scale and position uniformity), they performed PCA on the raster-scanned grayscale pixel values in the collection to compute a mean face image $\boldsymbol{\alpha}_0$ and a set of L face image eigenvectors $\{\boldsymbol{\alpha}_l\}_{l=1}^{L}$. They showed that a novel face image $\boldsymbol{\alpha}$ can be approximately reconstructed using a weighted sum of relatively few eigenvectors:

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \sum_{l=1}^{L} \lambda_l \boldsymbol{\alpha}_l, \tag{2.3}$$

where λ_l is the l-th weight and L is typically much smaller than the number of face images in the collection. Thus, once encoded, holistic face appearance can be stored

using a small (*i.e.*, fewer than 100) collection of weight parameters.

This result inspired Turk and Pentland [113, 114, 115, 116] to extend Sirovich and Kirby's work to face recognition. In their well-known 1991 paper, *Eigenfaces for Recognition* [115], Turk and Pentland showed impressive face recognition results under controlled laboratory conditions.

One of the major weaknesses in these early PCA-based approaches is the lack of shape normalization. The implicit assumption is that pixels with the same image offset correspond to the same semantic face location. Even with careful scale, translation, and rotation alignment, this assumption is easily violated: different faces inherently come in many different shapes, a single face can dramatically change its shape due to facial expression, and, within the 2D image plane, face shape changes significantly with head pose.

As an improvement, Craw and Cameron [29] described an approach similar to Eigenfaces, but with one key difference: they first *manually* normalized the shape of each face. Lanitis, Taylor, and Cootes [71, 72, 70, 73] further extended Craw and Cameron's work by automating the shape normalization procedure. They used ASMs to automatically locate contour landmarks on each test face. ASMs were a natural choice: ASMs use PCA to model shape variation much like Eigenfaces use PCA to model holistic appearance variation.

Lades *et al.* [68] adopted a different approach. To established correspondences between face images, their system deforms each image based on an overlaid deformable rectangular grid. Unlike ASMs, their system uses grayscale information over the *whole* face to deform the grid. Lades *et al.*'s system penalizes deformations, but does not otherwise constrain them (*e.g.*, using PDMs), which means the deformations can produce non-realistic results.

In the late 1990s, Cootes, Edwards, and Taylor [21, 22] introduced Active Appearance Models (AAMs) in a seminal paper by the same name. Like Lanitis *et al.*'s work [73], AAMs use the Eigenfaces framework to model face texture/grayscale appearance holistically, and PDMs to model face shape. Instead of using ASMs to guide the shape fit, AAMs use the grayscale values over the *entire* face.

The central problem that AAMs address is the search for appearance and shape parameters. AAMs model both the overall grayscale appearance $\boldsymbol{\alpha}$ and shape X of

the face linearly:

$$X = X_0 + \sum_{m=1}^{M} p_m X_m \tag{2.4}$$

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \sum_{l=1}^{L} \lambda_l \boldsymbol{\alpha}_l, \tag{2.5}$$

where p_m and λ_l are the m-th shape and l-th appearance parameters, respectively; $\{X_{m=1}\}_{m=1}^{M}$ and $\{\boldsymbol{\alpha}_l\}_{l=1}^{L}$ capture the most significant M shape and L appearance modes, respectively; and X_0 and $\boldsymbol{\alpha}_0$ are the average shape and grayscale appearance vectors, respectively. Note that Eq. (2.4) is identical to the ASM formulation in Eq. (2.2), and Eq. (2.5) is identical to the Eigenfaces formulation in Eq. (2.3). The goal is to minimize the difference between the test face image and the model:

$$\min_{\mathbf{p},\lambda} \left\| W(I,\mathbf{p}) - \left(\boldsymbol{\alpha}_0 + \sum_{l=1}^{L} \lambda_l \boldsymbol{\alpha}_l \right) \right\|_2^2, \tag{2.6}$$

where W warps the face image I in a piecewise linear fashion using the shape parameters $\mathbf{p}$.

The parameters $\mathbf{p}$ and λ are gradually updated using an iterative technique until convergence. The original AAM fitting procedure relies on two approximations in order to simplify the optimization. First, a first order Taylor approximation is used to linearize the problem at each iteration. Second, instead of recomputing the derivative of the error image with respect to the parameters at each iteration, the relationship between the error image and the additive parameter updates is computed once at the beginning using regression, and is assumed to be constant from start to finish. Recent works have focused on improving these approximations.

2.5 Recent Parametric Holistic Methods

There are two key challenges in applying AAMs in practice. First, although AAMs model shape and appearance linearly, the model parameters are nonlinear with respect to the image pixel intensities. Consequently, the (nonlinear) optimization can easily converge to an incorrect local minimum or fail to converge entirely, especially if the

model is not initialized close to the true face shape. Second, it is difficult to adequately synthesize the appearance of a new face using a single linear model. This is known as the *generalization problem*.

Many researchers have extended and improved the original AAM algorithm to better address these challenges. Blanz and Vetter [9] proposed a similar parametric framework to compute and align 3D models to 2D face images. A single 3D linear model can better handle large head pose variation compared to a single 2D linear model. However, like standard AAMs, the algorithm must be initialized carefully and is susceptible to incorrect local minima. Other types of significant variation like exaggerated expressions and dramatic illumination are similarly challenging for holistic 3D models.

Matthews *et al.* [89] improved the optimization procedure of AAMs. Specifically, they proposed a fitting algorithm based on an extension of the inverse compositional image alignment algorithm [4]. Rather than updating the warping parameters *additively*, their algorithm updates the entire warp by *composing* the current warp with the computed incremental warp at each iteration. Rather than a *forward* warp, their algorithm uses an *inverse* warp. They also fit the local shape parameters $\mathbf{p}$ and the global shape transformation parameters $\{s, R, \mathbf{t}\}$ *simultaneously* rather than solving for them independently.

Cong Zhao *et al.* [137] focused on the generalization problem. They proposed a *joint* face alignment technique in which a generic AAM is fit to a batch of similar faces (*i.e.*, faces of the same individual) simultaneously. Specifically, they added a rank term to the optimization, which encourages the grayscale appearance of the input images to be linearly correlated. Intuitively, the rank term favors a solution in which the input images were well aligned with one another (not just the global model).

Xiaowei Zhao *et al.* [139] also focused on the generalization problem. Their algorithm first selects a subset of k-nearest neighbor training faces (with respect to the test face) to construct a target-specific AAM for each test face at runtime. Xiaowei Zhao *et al.*'s algorithm is among those we have evaluated; we show in Sections 3.4 and 4.4 that our approach produces more accurate results on challenging faces.

AAMs are *generative* models that are well-suited to face image synthesis, but not necessarily to fitting. Many works have taken a different approach by learning the parameter fitting procedure in a *discriminative* framework. Such approaches (*e.g.*,

[102, 103]) directly map image appearance to AAM parameter updates. Boosted regressors [41, 42] are commonly used to learn the mapping, *e.g.*, Boosted Appearance Model (BAM) [82], Boosted Ranking Model (BRM) [123], and Tresadern *et al.* [112].

More recently, Tzimiropoulos and Pantic [117] proposed an algorithm for solving the *exact* AAM nonlinear least squares problem (*i.e.*, by using fewer approximations). They showed that AAMs can be trained and fit on in-the-wild faces. Their fitting accuracy is state-of-the-art among AAM-based methods. In Sections 3.4 and 4.4 we compare our fitting accuracy with theirs, and show that our results are significantly more accurate.

Assuming that AAMs can be fit perfectly to the image, they still have inherent limitations. For example, because AAMs model the entire face holistically, they tend to fail on unusual expressions (*e.g.*, asymmetric expressions not represented well in the training set). Similarly, local aberrations, like partial occlusions or intense shadows, are difficult to model holistically.

2.6 Recent Parametric Local Methods

Parametric local methods model the appearance of the face locally via an ensemble of region experts, or local detectors, and model the global shape using a parametric model (*i.e.*, a PDM). Chief among these methods are Constrained Local Models (CLMs) [33, 34, 104]. CLMs overcome many of the problems inherent in AAMs. They have inherent computational advantages (*e.g.*, opportunities for parallelization) [85], reduced modeling complexity and sensitivity to illumination changes [104, 119], and they generalize well to new face images and can be made robust against other confounding effects such as reflectance, image blur, and occlusion [46]. Methods that build on the CLMs framework typically differ along two dimensions: the type of local detectors used and the optimization strategy employed.

A variety of local detectors have been proposed for CLMs. Detectors most commonly use local image patches, SIFT (Scale-Invariant Feature Transform) [84] descriptors, or HOGs (Histograms of Oriented-Gradients) [35] to encode local appearance. Prior to shape fitting, each local detector is exhaustively scanned over each landmark region to produce a local likelihood response map. The local appearance can be modeled parametrically or nonparametrically. Parametric models are more common,

and they can be either generative or discriminative. Generative landmark appearance models include, for example, Gaussian mixture models [46] or eigenvectors computed over collections of small (*e.g.*, 15×15 pixels) training image patches [33, 34] (*i.e.*, much like Eigenfaces, but with each landmark modeled locally and individually). Discriminative landmark appearance models use, for example, linear SVM (support vector machine) classifiers [85, 104, 119] and boosted classifiers [134, 140] like AdaBoost [41].

A variety of iterative optimization strategies have been proposed for CLMs. ASMs [25] use the simplest optimization strategy. They compute the error at each iteration as a weighted least squares difference between the PDM and the coordinates of the peak responses, where each weight is a function of the peak height. The objective function is approximated by a first order Taylor expansion at each iteration. Unfortunately, ASMs are sensitive to false peaks, which commonly occur in practice. Another strategy is to approximate the objective function as a convex quadratic [119], which is less sensitive to isolated false peaks, but can fail if the response maps have multiple strong peaks. Gaussian mixture models provide a better approximation in such cases [46], but are computationally expensive, converge only locally, and the number of mixture components must be chosen *a priori*. Saragih *et al.* [104] proposed Subspace Constrained Mean-Shifts for optimizing the PDM. Specifically, their algorithm uses kernel density estimation (KDE) with an isotropic Gaussian kernel to approximate the objective function. The mean shift algorithm [20] is used to iteratively fit the PDM.

More recently, Asthana *et al.* [3] proposed Discriminative Response Map Fitting (DRMF). DRMF is similar to discriminative approaches to AAM fitting that use boosted regressors [82, 103, 123], but it fits the PDM to local response maps rather than the holistic face appearance.

Several approaches have focused directly on addressing the combinatorial problem of fitting shape models to many landmark candidates. Amberg and Vetter [2] proposed an efficient branch and bound [69] strategy to discard large areas of the search space. In principle, their approach can accommodate different types of shape models, although they limited their experiments to a *fixed* 3D shape model. Zhao *et al.* [140] recently proposed a pruning strategy for efficiently removing incorrect shape configurations based on a discriminative structure classifier. They showed results on nine facial landmarks (eye corners, eye centers, mouth corners, and nose tip), but it is unclear

how well their algorithm performs on more challenging landmarks, *e.g.*, on the upper and lower lips, the face contour, and the eyebrows.

Local methods are perhaps best suited to face tracking, where the face shape in each frame can be initialized using the result from the previous frame. This is because each local detector must be placed near the true landmark location *a priori*. If the initialization is close, then the size of the detector window can be small, which yields fewer ambiguities (false peaks) and greater efficiency. On the other hand, if the landmark locations are less predictable, then the size of each detector window must be large, which can significantly reduce performance.

Because CLMs and AAMs employ the same underlying global shape model (*i.e.*, a PDM), they sometimes do not generalize well to new and unusual face shapes. In practice, they often fail on exaggerated asymmetric facial expressions or non-frontal head pose, for example.

2.7 State-of-the-Art Approaches

Beginning with the influential work of Belhumeur *et al.*'s [7], recent works have shifted away from testing on laboratory datasets like those listed in the left columns of Table 2.1 toward more challenging "in-the-wild" datasets like those listed in the right column of Table 2.1. Recent datasets have introduced new challenges, including greater facial expression and head pose variation, partial occlusion, and a range of real-world imaging conditions, including noise, blur, and poor illumination. To better address these challenges, several recent works [7, 18, 110, 124] have proposed modeling shape *nonparametrically* rather than parametrically.

2.7.1 Exemplar-Based Approaches

Belhumeur *et al.* [7] proposed an exemplar-based approach to shape modeling. Much like previous work, their method starts with an ensemble of SVM-based detectors to compute each landmark response map. The novel part of their method is the shape regularization step. They adopted a generate-and-test approach similar to the RANSAC (RANdom SAmple Consensus) algorithm [39] to select a subset of top face shape exemplars and similarity transformations, which align the face shape exemplars

Laboratory face datasets (early)	In-the-wild face datasets (recent)
• AR (Aleix-Robert) [88] • XM2VTSDB (Extended Multi-Modal Verification for Teleservices and Security Database) [90] • CMU PIE (Pose Illumination Expression) [106] • PUT [55] • CMU Multi-PIE [45] • Yale B [43] • BioID [52]	• LFW (Labeled Faces in the Wild) [51] • AFLW (Annotated Facial Landmarks in the Wild) [60] • LFPW (Labeled Face Parts in the Wild) [7] • AFW (Annotated Faces in the Wild) [142] • Helen [74] • IBUG (Intelligent Behavior Understanding Group) [98]

Table 2.1: Common experimental face datasets for face alignment and facial landmark localization. The left column lists several popular face datasets captured in laboratory environments. They tend to depict faces with less variation (*e.g.*, controlled illumination, limited head pose, neutral expressions, *etc.*). Most early face datasets fall under this category. The right column lists several popular face datasets containing images depicting faces in unconstrained real-world environments. They tend to be more recent, and the faces tend to be much more challenging (*e.g.*, exaggerated facial expression, non-frontal head pose, image noise, *etc.*). In recent years, researchers have been moving away from datasets in the left column toward the more challenging in-the-wild datasets in the right column.

to the test face. Each generated {exemplar shape, similarity transformation} pair is scored based on how well the landmark hypotheses from the aligned exemplar shape coincide with the peaks in the response maps. The landmark hypotheses from the top-scoring shapes are then fused to compute a final set of landmark locations. Belhumeur *et al.* showed impressive results on real-world face images, with landmark localization accuracy similar to human annotators assigned to the same task. We have incorporated their nonparametric approach to shape modeling into our core pipeline, and so we describe their algorithm in more detail in Section 3.3.

Feng Zhou *et al.* [141] recently introduced exemplar-based graph matching (EGM) as an extension of [7]. They noted that a major limitation of [7] is that the final position of each landmark is inferred *independently* in the fusion step using a greedy procedure, which is suboptimal. They formulated the inference problem as a graph matching problem. Specifically, given k sets of landmark candidates and m exemplar faces selected using Belhumeur *et al.*'s algorithm, they find an optimal subset of candidates by (1) computing an affine-invariant shape constraint online from the m exemplar faces, and (2) solving a graph matching problem to find the optimal candidates. We have incorporated a modified version of their algorithm in our pipeline for transferring landmark annotations across multiple face databases. We describe their algorithm in more detail in Section 4.3.

2.7.2 Voting-Based Approaches

There are three main problems with using landmark detectors to compute the local response in each landmark region. First, the initial placement of the detection window depends on a relatively good initialization, which is difficult in practice. Second, each detector must exhaustively scan each local region, which can be inefficient. Third, landmark detectors typically encode the appearance of a landmark using a single small image patch, or a single feature descriptor (*e.g.*, SIFT), which can lead to many local ambiguities that make shape fitting more difficult.

To address these problems, several recent works have used the Generalized Hough Transform [5] framework to 'vote' for landmark locations. The basic idea is to cast a vote from each appearance feature in the test image toward a potential landmark location. The votes are generally noisy because they each depend on only a single local

image feature, but a collection of votes will tend to accumulate at the correct landmark location. Hough voting is attractive because it can capture complex interactions between image features and landmarks, and the interactions can be proximate or distant.

There are at least three key challenges in using Hough voting in practice. First, a feature-to-landmark vote offset must be computed *efficiently* for each {image feature, facial landmark} pair in the test image. Several recent works [24, 36, 126] have proposed using regression forests for this purpose, as described below. Second, because some features are more discriminative and less noisy than others, the weight or confidence of each vote must be computed. Vote confidence is usually computed heuristically. Third, because the raw voting maps tend to by 'spiky', the votes must be spatially aggregated prior to mode selection. The canonical approach is to model the spatial uncertainty of the votes using a 2D Gaussian; this is equivalent to smoothing the voting map with a Gaussian filter after all votes have been accumulated.

Shen *et al.* [105] recently proposed using Hough voting and image retrieval techniques for face detection. Their approach relies on a large database of face exemplars to detect faces in the test image. Specifically, each exemplar consists of a face image, *quantized* SIFT features, and a bounding rectangle around the face. The SIFT features are assigned to one of 100k quantization bins (a.k.a. "visual words") using fast approximate k-means [91], and are stored in an inverted index for fast lookup. At runtime, each SIFT feature in the test image is also quantized, and a list of matching features (*i.e.*, features assigned to the same quantization bin) is retrieved from the database. Each matched feature produces one vote in the test image, which extends from the test feature to the center of a potential face in the test image.

The vote offset is determined as follows. Assume one of the matched features is f_j at location $L(f_j)$ in exemplar j. Exemplar j contains a face centered at location $L(\text{face}_j)$. The vote offset in the test image is then $s \cdot (L(\text{face}_j) - L(f_j))$, where s accounts for scale differences between the test face and the exemplar face. Shen *et al.* demonstrated state-of-the-art detection performance on challenging face images. Although they briefly mentioned face alignment as an extension of their approach, they effectively left it as future work. Our landmark localization approach is inspired by their work, and so we describe their algorithm in more detail in Chapter 3.

Most recent voting-based approaches to facial landmark localization employ re-

gression forests to efficiently compute vote offsets from features in the test image. For example, Dantone *et al.* [36] proposed using conditional regression forests, in which each tree is trained to map image features (input) to displacement vectors (output). Each displacement vector casts a vote from the image feature toward a potential facial landmark location. Each vote is weighted based on the offset distance (*i.e.*, intuitively, nearby votes are usually more accurate than distance votes). Dantone *et al.* imposed no global shape constraint; they simply used mean shift [20] to *independently* select the largest mode in each voting map as the final landmark location.

Cootes *et al.* [24] proposed *Regression-Voting*, which extended Dantone *et al.*'s work by fitting a PDM to the regression voting maps. Their algorithm has many of the benefits of CLMs, but without the need for exhaustive landmark detectors. Hough voting maps can cover a much larger area than local detector windows with little additional overhead.

Yang and Patras [126] proposed 'sieving' regression forests. Each vote is passed through a cascade of sieves, which filter out irrelevant votes and enforce local geometric constraints. The weight of each feature is a function of (1) its ability to accurately vote for the center of the face, and (2) the distance of the vote from the feature to the landmark. Like Dantone *et al.*, Yang and Patras chose not to incorporate a global shape model into their algorithm.

2.7.3 Regression-Based Approaches that Model Shape Nonparametrically

Researchers have recently proposed using regression-based approaches to fit *nonparametric* shape models to the image. Cao *et al.* [18] proposed training a boosted regressor to explicitly minimize the shape alignment error. Their algorithm is iterative; at each iteration a boosted regressor computes an additive shape increment as a function of the image features and the current face shape. Their shape model is nonparametric in the sense that it represents novel shapes as a linear combination of *all* training shapes. However, like PDMs, each combination weight is uniform across the entire face, which means the model may not adapt well to unusual asymmetric expressions, for example.

To address this weakness, Xiong and De la Torre [124] recently proposed a Supervised Descent Method (SDM). They encode local appearance using a single SIFT

feature at each landmark. Their algorithm is iterative. At the k-th iteration the face shape $\mathbf{x}$ is updated as

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \Delta\mathbf{x}_{k-1}, \tag{2.7}$$

where $\Delta\mathbf{x}_{k-1}$ takes the form

$$\Delta\mathbf{x}_{k-1} = \mathbf{R}_{k-1}\phi_{k-1} + \mathbf{b}_{k-1}. \tag{2.8}$$

ϕ_{k-1} is a long vector of all the SIFT descriptors extracted at the current landmark coordinates. $\mathbf{R}_{k-1}$ outputs a descent direction and $\{\mathbf{b}_k\}$ is a bias term. Thus, for any given set of SIFT descriptors, their algorithm produces a face shape update. The sequence of descent directions $\{\mathbf{R}_k\}$ and bias terms $\{\mathbf{b}_k\}$ are trained such that the succession of $\mathbf{x}_k$ converge to the true shape $\mathbf{x}_*$ for all images in the training set.

Xiong and De la Torre's algorithm is very fast and produces excellent results on most faces. However, in practice their method is sensitive to the initialization, *i.e.*, $\mathbf{x}_0$ must be reasonably close to $\mathbf{x}_*$. The initialization is less problematic in face tracking applications, in which the previous video frame generally provides a good initialization of the current frame, but shape initialization is difficult in practice on many static in-the-wild face images, where the initialization is provided by a face detector. We observe in Section 3.4 that most of the errors from their algorithm are due to poor shape initialization when applied to especially challenging face images.

2.7.4 Simultaneous Face Detection and Landmark Localization

Almost all face parsing methods rely on off-the-shelf face detectors for initialization. By far the most popular face detector is Viola-Jones [118] due to its simplicity, speed, satisfactory performance on most near-frontal faces, and the availability of open-source implementations, *e.g.*, via the OpenCV (Open Source Computer Vision) library [1]. Unfortunately, Viola-Jones fails on many challenging face images, including faces with non-frontal head pose and exaggerated facial expression. To address this problem, Zhu and Ramanan [142] proposed a part-based model for coupling face detection *and* facial landmark localization. They used a tree structure (*i.e.*, with no cycles) to model the global face shape, which allowed them to use dynamic programming to efficiently

fit face shapes to the image. They used a mixture of trees to model topological changes due to viewpoint/head pose, with a shared pool of template-based landmark detectors. Zhu and Ramanan showed impressive detection and pose estimation results on faces with a large variety of head poses, from frontal to full-profile. However, the landmark localization accuracy of their method is generally rough, and their model can produce unnatural deformations, which suggests that loopy spatial constraints are necessary. Several recent facial landmark localization methods use their algorithm for initialization [130].

2.7.5 Face Parsing with Segments and Pixel-wise Labels

There are many examples in the literature of algorithms that parse *general* scenes (*e.g.*, with buildings, sky, cars, *etc.*) using segments or pixel-wise labels (*e.g.*, [61, 80, 128]), and there are many examples of methods that parse face images using landmarks. However, few researchers have proposed explicitly parsing face images using semantically-defined segments or pixel-wise labels, and current approaches are limited.

Warall and Prince [120] adapted a general scene parsing algorithm for the specific task of face image labeling. Their work is novel because it is among the first to propose face parsing as an image labeling problem. Unfortunately, their segmentation results were rough and inexact.

Kumar *et al.* [66, 67] proposed separating faces into different semantic regions for face verification. However, because parsing was not the focus of their work, their method is simple and imprecise: each face is first rectified to a common reference using an affine warp, and then a collection of masks in the reference frame select the pixels that lie within each region.

Luo *et al.* [86] proposed a hierarchical deep learning strategy for directly learning a mapping from images to label maps. In their work, they recovered pixel-wise labels for eyes, eyebrows, nose, mouth, and background (including the skin). Unfortunately, the details of their approach are opaque and their component-specific segmentors do not generalize well to larger, less distinct regions of the face, such as the cheeks and the chin. Both [120] and [86] produce a binary classification at each pixel, which does not generalize well to more complicated label interactions, such as those between the

inner mouth region, the lips, and the skin around the lips, for example. In Chapter 5 we describe a nonparametric strategy to face image labeling, which produces a *soft* segmentation that generalizes well to arbitrary label types.

2.8 Summary and Observations

In the early years of face parsing, from the late 1960s through the 1980s, approaches tended to focus on locating facial parts as a way to characterize face images. Starting in the early 1990s with *Eigenfaces*, holistic methods, which model the raster-scanned grayscale pixel intensity values across the *entire* face region, grew in popularity. The Active Appearance Model (AAM) framework was part of this trend and influenced many subsequent face alignment works. To overcome some of the optimization and generalization challenges of AAMs, local methods like CLMs became popular starting in the mid-2000s. Both AAMs and CLMs rely on *parametric* models, which can fail to generalize well to new faces in real-world conditions. For example, they can fail on asymmetric facial expressions or non-frontal head pose. Many approaches have been proposed in the last 10 years to better address the challenges of unconstrained, in-the-wild face images. We can categorize most modern landmark localization and face alignment approaches into five overlapping categories: parametric, nonparametric and exemplar-based, regression-based, voting-based, and graph-based approaches. In addition, a small number of works have focused on computing segment-based face representations rather than landmark- or contour-based representations.

Among these approaches, nonparametric ones have recently demonstrated exceptional promise. For example, using exemplar-based shape constraints, Belhumeur *et al.* [7] demonstrated landmark localization accuracy comparable to human annotators on challenging faces. Both Cao *et al.* [18] and Xiong and De la Torre [124] recently demonstrated even better state-of-the-art accuracy at real-time speeds using shape regression and nonparametric shape models. Voting-based approaches like [36, 105, 126], which model local appearance in a nonparametric way, have recently demonstrated promising results despite imposing no explicit global shape constraints.

Despite the excellent performance of recent nonparametric approaches relative to others, they are not perfect. For example, [7] employs local detectors to measure the local likelihood of each landmark; local detectors are sensitive to large head pose

variation. The iterative regression strategy of [124] depends on a good initialization, which is difficult in practice. Without shape constraints, voting-based approaches like [36, 105, 126] are susceptible to false detections and local aberrations.

Inspired by the recent success of nonparametric approaches, and motivated by their weaknesses, we propose to parse faces using a nonparametric exemplar-based strategy. Our approach models *both* shape and local appearance nonparametrically using a large collection of exemplar faces. A key result is that our approach can locate landmarks with state-of-the-art accuracy on especially challenging face images. For example, our algorithm produces accurate results on faces with significant yaw, pitch, and roll head rotation, and on faces with non-neutral facial expression.

One of the challenges in achieving good results with an exemplar-based approach is that success or failure depends on the existence of similar exemplars in the database. Unfortunately, experimental databases are usually developed independently with no common standards. For example, AFLW faces include up to 21 landmarks, but Helen faces include 194 landmarks. Furthermore, different datasets have different strengths and weaknesses. For example, the LFPW dataset includes many adult celebrities with mostly frontal head pose, while AFW is smaller but includes a wider range of ages and head poses. Because of the different landmark definitions, researchers must choose one database for experimentation and evaluation, which precludes taking advantage of all the different kinds of information from multiple databases. Due to the expense and tediousness of manual annotation, *automated* or semi-automated methods are needed to combined these datasets.

We therefore introduce a novel pipeline that extends our core facial landmark localization algorithm for the task of transferring landmark definitions across different datasets. Specifically, our extended algorithm takes multiple source datasets as input and *collaboratively* fills in the 'missing' labels in a target dataset using a union of landmarks defined in the source datasets. Our algorithm can optionally use known landmarks in the target dataset as constraints. One of the concrete results from this work is 63 supplementary landmarks for faces in the AFLW database [60], which increases the number of landmarks from 21 to 84. AFLW is significant in that, to the best of our knowledge, it is currently the largest publicly available in-the-wild face dataset with 25k annotated faces.

Face parsing is, with few exceptions, performed by facial landmark localization and

face alignment algorithms. Landmarks and contours are convenient for computational and storage reasons, and many experimental datasets exist with ground truth landmark annotations. However, a landmark-based representation is ill-suited for several compelling applications in face image analysis. For example, portrait editing wizards that automatically or interactively touch up faces often require accurate delineations between face parts. Contours are well-suited to delineating the eyes from the cheek, and the lips from the chin, for example. On the other hand, skin-hair boundaries are often too complex to capture with a single contour, and other boundaries, like the sides of the nose, are naturally indistinct, which is lost with sharp boundaries imparted by contours. Facial attribute recognition, too, could benefit from a more descriptive segment-based representation of the face. For example, image texture analysis of skin regions is useful for age recognition, and the shape of the hair can be a useful cue for gender recognition.

Motivated by these examples, we show that an exemplar-based strategy is well-suited to the task of automatic face parsing with landmarks *and* segments. Because our approach avoids parametric models, the collection of exemplar faces and the set of segment types can be easily extended online with no retraining. Our algorithm estimates *soft* segments, which naturally encode the indistinct boundaries between some facial parts. In principle, such a representation is general enough to model any facial part, including hair, ears, teeth, and skin.

Chapter 3

Face Image Parsing with Landmarks

3.1 Introduction

Facial landmark localization is an important research area in computer vision in part because digital face portraits are ubiquitous. Many compelling applications depend on it, including face recognition and retrieval, face animation, and face image editing wizards. At the same time, robust facial landmark localization is very challenging in practice. Real-world images can be cluttered, faces can be partially occluded, and they can exhibit large variations in appearance, shape, expression, and head pose. In particular, when current approaches fail, they often fail on faces with extreme expression and/or head pose.

One primary source for failure is the practical challenge of initialization. Many landmark localization methods rely heavily on a reasonable initialization as a pre-requisite for success, and can fail to find the correct solution if the initial face shape is too far from the true optimum. A popular strategy, even for recent approaches (*e.g.*, [3, 18, 19, 104, 124]), is to first detect the face (*e.g.*, using [118]), and then fit a mean face shape (in which the shape is defined by the facial landmarks) to the detection window. For upright near-frontal faces (*e.g.*, within 30 degrees yaw, pitch, and/or roll head rotation from frontal), detection and initialization is seldom a problem. However, for extreme poses and some expressions, traditional face detectors

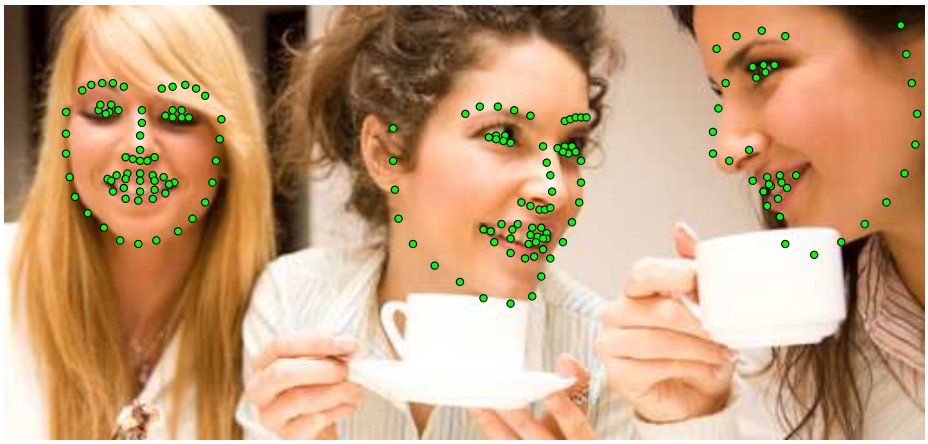


Figure 3.1: Our robust exemplar-based algorithm locates landmarks on challenging faces with extreme head pose.

(*e.g.*, [118]) may fail, or the true shape of the face inside the detection window will differ significantly from the initial shape, making a good initialization unlikely, thereby challenging even recent methods like [124], as we observe in Section 3.4. This problem is often minimized in the literature, where even recent, popular evaluation datasets like LFW [51], LFPW [7], and Helen [74] make initialization relatively easy.

Part-based models [38, 127] can be used to address the initialization problem, but learning an accurate part graph parameterization and inferring part labels from the graph can be challenging. Recent works [130, 142] simplify the graph structure to a tree and produce impressive results. Failure cases suggest that human faces, unlike the human body, still prefer a loopy graph structure. Furthermore, a landmark graph typically only models interactions between landmarks but does not model the interactions between landmarks and non-landmark image patches.

Our goal is to accurately localize landmarks on faces with extreme head pose and/or expression. To achieve this goal, we have developed an nonparametric exemplar-based approach that requires only a weak initialization. More specifically, we generalize and combine a recent exemplar-based approach for shape regularization [7] with an exemplar-based approach for face detection [105] to model context interactions between landmarks and their surrounding local appearance features in a nonparametric way. By "weak initialization" we mean that our algorithm does not require (nor does it

position) a face shape for initialization. Our algorithm relies on a face detector (*i.e.*, [105]) only to establish a *rough* initial size for the face and the region of interest in the image. Our algorithm also searches over multiple face scales and rotations during landmark localization, which allows for a large margin of error in the initial face detection. The end result is a new pipeline that achieves state-of-the-art results on unconstrained face datasets populated with challenging poses and expressions. The robustness and flexibility of our method comes from its ability to efficiently and effectively leverage the information from a large database of face exemplars.

This chapter describes three contributions:

1. We propose a data-driven approach for modeling the correlations between each landmark and its surrounding appearance features. At runtime, each feature casts a *weighted* vote to predict landmark locations, where the weight is precomputed to take into account the feature's discriminative power.

2. We combine nonparametric local appearance modeling with nonparametric shape regularization to build a novel facial landmark localization pipeline that is robust to common types of real-world variation, including scale, rotation, occlusion, expression, and most importantly, extreme head pose.

3. We compare our approach to several recent approaches and show state-of-the-art accuracy on two especially challenging in-the-wild datasets populated by faces with extreme head pose and expression.

3.2 Related Work

Early facial landmark localization and face alignment methods, *e.g.*, Active Shape Models (ASMs) [25] and Active Appearance Models (AAMs) [21], relied on global parametric models for face shape and appearance. Parametric models work well for favorable face images, *e.g.*, where the illumination, pose, and expression do not vary greatly.

To overcome the well-known generalization problem in AAMs, Zhao *et al.* [139] proposed computing a separate AAM for each test face using k-nearest neighbor training faces (w.r.t. the test face) rather than all training faces. Using k-NN exemplars is an important part of our approach (see Section 3.3.3) and others [7, 105, 141], although it is not our main contribution. Like other AAM-based methods, [139] involves a gradient decent-type optimization over the whole face (holistic), which is sensitive to initialization. We use a Hough voting scheme over features to model the appearance context of each landmark, which is robust to poor initialization.

Constrained Local Models (CLMs) [3, 34, 104] handle a wider range of faces than generic AAMs by employing an ensemble of local texture patches or landmark detectors that are constrained by a global shape model. The local appearance models are more robust to a range of challenges including occlusion and global illumination changes, but CLMs still rely on parametric shape models for regularization, which may not generalize well to a broad range of poses.

Belhumeur *et al.* [7], and more recently Zhou *et al.* [141], use nonparametric global shape models, which generalize better to challenging real-world faces. However, these two methods still rely on local landmark detectors (linear support vector machines (SVMs)) to form the landmark response. Because landmark appearance changes dramatically with large head pose variation, these two methods are limited to constrained head poses.

Recent regression-based approaches [18, 124] have also demonstrated increasingly impressive performance on real-world faces without the need for a parametric shape model. Although these methods have some tolerance to inaccurate initialization, for many challenging faces, where pose can easily vary past 30 degrees yaw, pitch, or roll head rotation, a good initialization is much more difficult. As a result, descent-based methods can get stuck in local optima.

Zhu and Ramanan [142] address this initialization problem in their work. They use a tree structured part model of the face, which both detects faces and locates facial landmarks. One of the major advantages of their approach is that it can handle extreme head pose. However, their method only models pairwise landmark interactions on a tree structure. Yu *et al.* [130] speed up [142] by simplifying the mixture of parts for face detection and initial landmark localization, and they extend [142] by adding a two-step local refinement procedure, which resembles the approach in [104] followed by the optimization of several additional constraints via a gradient descent method. Like [142] and [130], we focus on locating facial landmarks without relying on a good initialization, but we model the full interactions between each landmark and its surrounding local features. That is, our context interactions are not limited to a tree structure, and our approach does not involve graphical model inference.

We are most inspired by the recent face detection work of Shen *et al.* [105], who rely on a Hough transform based feature voting scheme to transfer many face hypotheses from a large database of exemplar faces to the test image. The votes capture the appearance and geometric correlations between local image features and the face center. While [105] focuses on face detection, we focus on landmark localization, and we compute a set of sparse feature weights (*i.e.*, most of them are zero) that naturally amplify reliable features and suppress noisy or unreliable features across the database; our weights are tailored to each {feature, landmark, exemplar} combination and are computed in a data-driven way.

Several recent object detection [62, 95], tracking [129], and face landmark localization methods [24, 36, 126] also rely on similar feature voting schemes to generate object/landmark response maps. Among these methods, Yang and Patras [126] is most similar to our work. They use image patches to cast votes for the location of each facial landmark. Our core approach is much simpler than [126]: we do not train regression forests or use SVM classifiers. Instead, we simply use a fast approximate nearest neighbor algorithm [91] for image and feature retrieval, followed by weighted vote accumulation. Unlike [126], we incorporate a shape regularization step in order to avoid false landmark detections.

3.3 Approach

In this section we first give an overview of our pipeline followed by technical details of each step. Figure 3.2 shows a visual synopsis of our pipeline.

3.3.1 Overview

Database Construction Our database is composed of a large collection of exemplars. Each exemplar has four components: a face image, a set of dense quantized SIFT [84] features, a sparse set of semantic facial landmarks corresponding to mouth corners, nose tip, chin contour, *etc.*, and a unique set of weights, one weight per {feature, landmark} pair. Following the approach in [105], we quantize each SIFT descriptor using fast approximate k-means [91], which efficiently maps each descriptor to a *visual word*. The weights are an important aspect of our approach; Section 3.3.7 describes in more detail how they are useful and how we compute them.

Runtime Preprocessing Given a test image, we first use a state-of-the-art face detector [105] to locate the face and roughly estimate its scale. The test image is cropped to the face region, and then rescaled to approximately match the scale of the exemplar faces (scale estimation will be refined in later steps). Dense SIFT descriptors are then extracted over the test face at multiple orientations. Finally, each descriptor is quantized for efficient matching in later steps.

Step 1: Top Exemplar Retrieval Given a detected face region, retrieve a subset of top similar k exemplar faces from the database. The goal is to retrieve exemplars that are similar to the test face in appearance, shape, expression, and pose so that features in the exemplars will produce accurate landmark votes in the test image.

Step 2: Landmark Voting For each type of landmark, generate voting maps using a multi-scale and multi-rotation generalized Hough transform [75]. Each matched feature from the top k exemplars casts a vote (for each scale and rotation) for a possible landmark location in the test image. The result is a table of voting maps for each landmark, where each table row corresponds to an in-plane rotation, and each table column corresponds to a scale estimate.

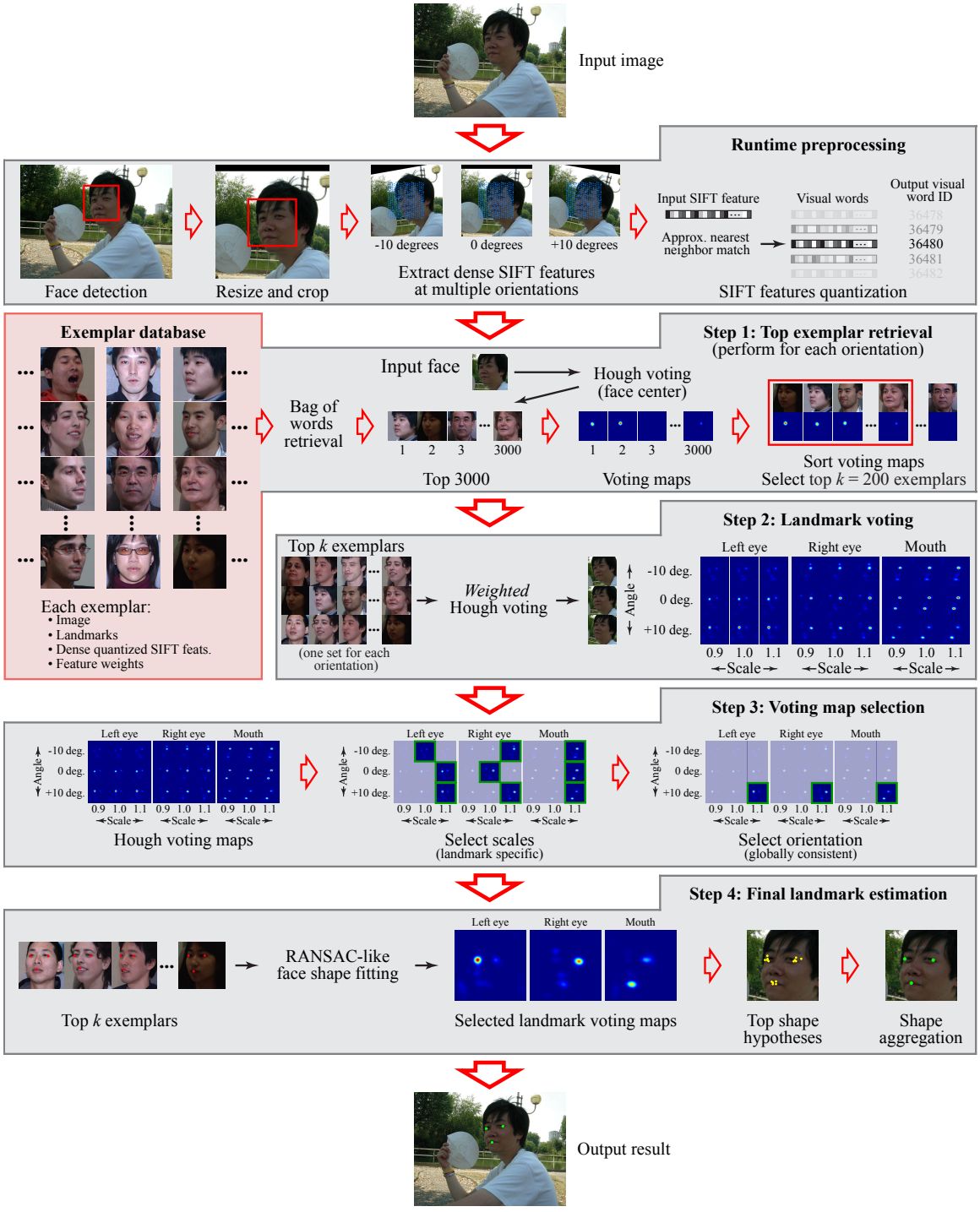


Figure 3.2: Overview of our facial landmark localization pipeline. Above, for simplicity, we search over 3 face orientations, 3 face scales, and we compute 3 landmarks. However, in practice, we search over 7 orientations, 7 scales, and we compute 68 landmarks.

Step 3: Voting Map Selection Select a single voting map for each landmark. We define the score of each voting map as the height of the maximum peak in the voting map. For each landmark, we save the top-scoring voting map in each row of the table of voting maps (each row corresponds to a rotation), which gives a unique scale estimate for each {landmark, rotation} pair. Among these voting maps, we select the single global rotation (and corresponding voting maps) that produce the maximum total score across all landmarks.

Step 4: Final Landmark Estimation Given a single voting map for each landmark, estimate a final set of landmark locations. Due to local ambiguities, noise, occlusions, *etc.* each voting map may contain multiple peaks. We employ a robust nonparametric shape regularization technique [7] that avoids false peaks in the voting maps and estimates a final arrangement of landmarks.

3.3.2 Database Construction

We use 17685 images exclusively from the Multi-PIE Face Database [45] as our exemplars. The Multi-PIE authors annotated 4685 face images, each with 68 landmarks (or 39 landmarks for profile faces). Some Multi-PIE faces have landmarks for one pose (*e.g.*, -30 degrees yaw), but not the opposite (*e.g.*, +30 degrees yaw). For such faces we synthesized the 'missing' landmarks by horizontally flipping the labeled image and its landmarks. 400 additional Multi-PIE images were labeled by [142]; we manually labeled 960 more (mostly non-frontal faces with non-neutral expressions). For each {session, subject, expression, pose} snapshot, Multi-PIE provides 20 images with different lighting directions. Therefore, we used each unique ground-truth face shape on multiple images with the same {session, subject, expression, pose} combination, but different lighting, to obtain 17685 labeled images. Prior to feature extraction and quantization, we used Procrustes analysis to align all exemplar faces.

3.3.3 Step 1: Top Exemplar Retrieval

In order to transfer landmarks from the database to the test image, the shape and appearance of the exemplar faces and the test face should not be drastically different. For example, a left-profile face has a much different shape and appearance than a

right-profile face; there are few feature-landmark correlations between the two. We therefore select a top subset of exemplars for further processing.

Many strategies exist for retrieving similar face images from a database. We use our generalized Hough transform framework to score each exemplar image. First, we use a bag-of-words score to efficiently select the top 3000 exemplars. Next, we use the features on the test face to vote for the center of each exemplar face among the top 3000. The final score for each exemplar is the height of the maximum peak in the voting map associated with each exemplar face. We sort the scores, and select the top $k = 200$. Shen *et al.* [105] adopt a similar strategy for retrieving exemplar faces in the validation step of their face detection algorithm.

3.3.4 Step 2: Landmark Voting

For efficiency, rather than exhaustively sliding each exemplar over the test image, we use quantized features and employ an inverted index file to efficiently retrieve matched features from the top k exemplars. When a feature in the test image is matched with an exemplar feature, the feature-to-landmark offset in the exemplar is transferred to the test image. The offset vector extends from the test feature toward a potential landmark location, and produces a vote. After many such votes, a voting map is formed, where the votes tend to cluster at landmark locations.

This Hough voting strategy is sensitive to scale and rotation differences between the test image and the exemplars. We therefore produce votes at several different scales (0.7 to 1.3 in increments of 0.1) and in several orientations (-30 to +30 degrees roll in increments of 10 degrees) on the test face. For efficiency, we use the same extracted features across multiple scales (this is possible because the scale differences between the test face and the exemplar faces are close to 1); only the vote offset vectors are scaled. For votes at different in-plane rotation angles, we use the corresponding set of orientation-specific features computed during the runtime preprocessing step. This is the same approach taken in [105] for face detection and alignment, except they omit rotation search. In our evaluation, we found that the rotation search is critical to our performance, as Figure 3.7 shows. This is partly due to the fact that our exemplar faces are aligned and thus exhibit very little in-plane rotation variation.

3.3.5 Step 3: Voting Map Selection

In selecting voting maps, we enforce the constraint that all voting maps must come from the same in-plane rotation. The intuition is that faces tend to rotate globally, not locally. On the other hand, the scale of face regions can vary locally due to expression or pose. We therefore select scale separately for each landmark.

3.3.6 Step 4: Final Landmark Estimation

There are many approaches in the literature for enforcing shape constraints, *e.g.*, [25, 34, 104, 130]. We use an exemplar-based approach to shape regularization [7], which fits nicely within our exemplar-based framework. To make this chapter more self-contained, we describe the essence of the shape regularization algorithm here.

Let $X = \{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^n\}$ denote the location of n facial landmarks and let $\mathbf{d}^i$ denote the window of measured detector response values for the i-th landmark. Belhumeur *et al.* [7] use SVM-based detectors to produce $\mathbf{d}^i$, whereas we use the Hough voting strategy described in Step 2. Let j be an index into a set of exemplar face shapes, and let t represent a 2D similarity transformation such that $\mathbf{x}^i_{j,t}$ is the t-transformed location of landmark i from exemplar j. A RANSAC-like procedure first generates and tests a large number of plausible values for j and t:

1. Select a random j.
2. Select two random face parts (*i.e.*, landmark detector windows). Randomly select one of the g highest modes in each detector window. Match each landmark mode to the corresponding landmark in exemplar j (*i.e.*, this creates two pairs of matched points).
3. Compute t such that the exemplar landmarks are aligned with the selected modes in the landmark response map.
4. Evaluate $\prod_{i=1}^{n} P(\mathbf{x}^i_{j,t}|\mathbf{d}^i)$, where $P(\mathbf{x}^i_{j,t}|\mathbf{d}^i)$ is the detector response value in $\mathbf{d}^i$ at location $\mathbf{x}^i_{j,t}$.
5. Repeat Steps 1 to 4 r times.
6. Record the m^* pairs $\{j, t\}$ for which the evaluation in 4 is largest.

We use $r = 40000$, $g = 4$ and $m^* = 100$. In Belhumeur *et al.*'s original work, j searches over the entire collection of exemplar faces. However, we constrain j to only search over the $k = 200$ exemplars retrieved in Step 1 of our pipeline. The top k exemplar shapes tend to be better tailored to the test face than the general set of exemplar shapes, which further aids the optimization.

The final location for each landmark is selected independently using a greedy fusion procedure. Specifically, each location $\mathbf{x}^i_{j,t}$ receives a weight, $w^i_{j,t} = P(\mathbf{x}^i_{j,t}|\mathbf{d}^i)$. The final location for landmark i is selected as the highest mode among all m^* weighted locations. Intuitively, because each $\mathbf{x}^i_{j,t}$ comes from an actual face shape $X_{j,t}$, the possible modes are limited to plausible face shapes. Please see [7] for more detail.

The SVM-based detectors used by Belhumeur *et al.* [7] are limited in that each appearance feature provides information only about its own location, and each SVM attempts to capture all the local appearance variation around each landmark within a single model. This works well on faces with limited head pose variation, but can fail otherwise. In contrast, by aggregating votes from many features, our method takes advantage of the appearance context around each landmark, which provides more robustness to local noise, occlusions, *etc.* We therefore use our voting-based landmark response maps in place of the local detector response maps used in [7].

3.3.7 Computing Exemplar Feature Weights

In this section we describe our approach for computing a unique weight for each {feature, landmark} pair in each exemplar image. Each weight is a score on the appearance and geometric consistency of each {feature, landmark} pair relative to similar pairs in other images.

Previous approaches compute this weight heuristically. For example, Shen *et al.* [105] attempt to down-weight potentially bad votes using a heuristic from object retrieval: $\frac{\mathsf{idf}^2(k)}{\mathsf{tf}_Q(k)\mathsf{tf}_D(k)}$, where $\mathsf{idf}^2(k)$ is the squared inverse document frequency of visual word k, and $\mathsf{tf}_Q(k)$ and $\mathsf{tf}_D(k)$ are the term frequencies of k in the query image and the database image, respectively. Shen *et al.*'s weighting scheme hinges on the heuristic that common and repeated features produce noisier votes.

We rely on a different intuition, and we compute each weight in a data-driven way. Intuitively, if a {feature, landmark} pair is consistent with similar pairs in other images (*i.e.*, the features map to the same visual word and the feature-to-landmark offsets are similar), then the feature is a good predictor of the true landmark location, and its vote should have higher weight. Conversely, if a feature is corrupted due to occlusion, for example, or if it describes an ambiguous local region, then its vote should have lower weight.

For simplicity, the discussion below focuses on a single generic landmark, but the same procedure applies to all landmarks. We first define some new notation. Let $L(f_i^r)$ be the location of feature i in exemplar r, and let $L(l^r)$ be the location of landmark l in exemplar r. The offset vector from $L(f_i^r)$ to $L(l^r)$ is denoted $\Delta L(f_i^r, l^r) = L(l^r) - L(f_i^r)$. Let $w(f)$ denote the mapping from feature f to its visual word, *i.e.* $w(f_i^r) = w(f_j^{r'})$ means that feature i in exemplar r matches feature j in another image r'.

Our goal is to compute the probability that f_i^r will vote for the correct landmark location $L(l^{r'})$ in other face images. Here, we assume that if two features f_i^r and $f_j^{r'}$ in two exemplar images r and r' correspond to the same visual word, *i.e.*, $w(f_i^r) = w(f_j^{r'})$, the relative location offsets from the feature to the landmark should be approximately the same. For "other similar face images" we use the top k retrieved exemplar images from Step 1 of our approach.

We estimate the above probability by counting the number of correct votes (*i.e.*, counting the number of matched offset vectors that are approximately the same) and

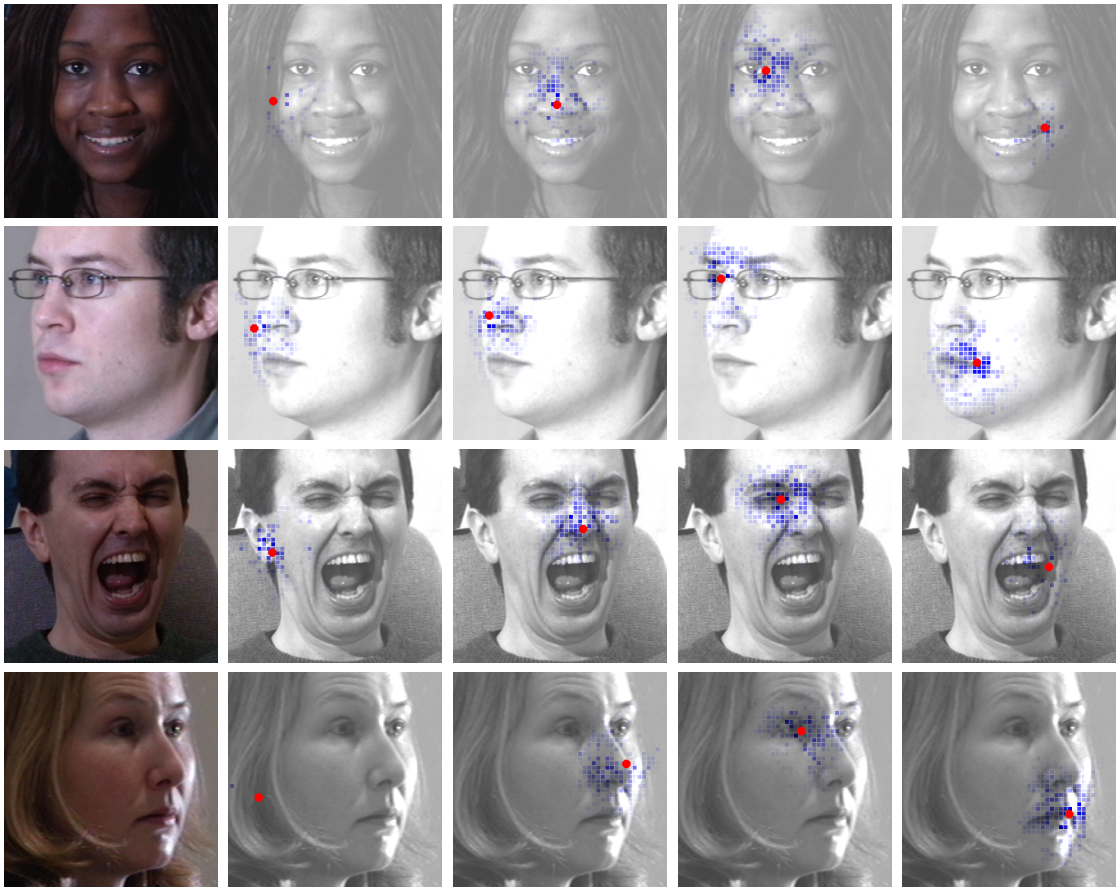


Figure 3.3: Visualization of landmark-specific feature weights. The red dot in each image shows the ground truth landmark location. The intensity of blue is proportional to the weight of the underlying feature. We see that higher weights naturally correspond to more locally discriminative regions and fully visible landmarks, while lower weights occur in uniform regions. For example, for Landmark 31, we see that features near the edges of the nose have higher weight than features on more uniform areas like the cheeks or philtrum. For Landmark 3, which is occluded by hair in the first and fourth rows, and partially occluded due to head pose in the second row, the feature weights are generally lower. We do not use a parametric form to model the weight; the weight values are completely derived from the data.

dividing by the total number of votes. This can be written mathematically as

$$P(\Delta L(f_i^r, l^r)) = \frac{1}{N} \sum_{r' \neq r} \sum_{\substack{j \in r' \text{ s. t.} \\ w(f_j^{r'}) = w(f_i^r)}} \Psi\left(\left\| \Delta L(f_i^r, l^r) - \Delta L(f_j^{r'}, l^{r'}) \right\| \right), \tag{3.1}$$

where the summation is over all features in the other exemplar images that share the same visual word, and N is the total number of votes cast by f_i^r. The function $\Psi(\cdot)$ in Eq. (3.1) quantifies the notion of "approximately the same offset." In our implementation we use $\Psi(x) = \exp\left\{-\frac{x^2}{2\sigma_g^2}\right\}$. We observe in Eq. (3.1) that

$$||\Delta L(f_i^r, l^r) - \Delta L(f_j^{r'}, l^{r'})|| = ||L(l^r) - V||, \tag{3.2}$$

where $V = L(f_j^r) + \Delta L(f_j^{r'}, l^{r'})$. This implies that we can evaluate Eq. (3.1) by first generating a single voting map for f_j^r, where each vote is cast at location V in exemplar r by features in other exemplars; we then count the number of votes near $L(l^r)$ and divide by N to compute $P(\Delta L(f_i^r, l^r))$.

As a kind of regularization, we modulate $P(\Delta L(f_i^r, l^r))$ by a spatial weight that gradually decreases with distance from the landmark,

$$s_j^r = \exp\left\{-\frac{||L(l^r) - L(f_j^r)||^2}{2\sigma_s^2}\right\}. \tag{3.3}$$

Regularization is important because N in Eq. (3.1) can be small, especially for features far from landmarks. Thus, the final weight is

$$v_j^r = s_j^r \cdot P(\Delta L(f_j^r, l^r)). \tag{3.4}$$

Figure 3.3 shows four sets of weights for four different exemplar images. Several qualities emerge naturally from the data. For example, features in uniform regions, such as the cheeks and forehead, receive smaller weights, while features in less ambiguous regions receive larger weights. We see that nose and chin landmarks require wider spatial support, while very distinctive landmarks such as the eye and mouth corners require only nearby support. Because the weights are specific to each feature, landmark, and exemplar, they can adapt to whatever global or local conditions exist, including

different poses, illuminations, facial expressions, occlusions, *etc*.

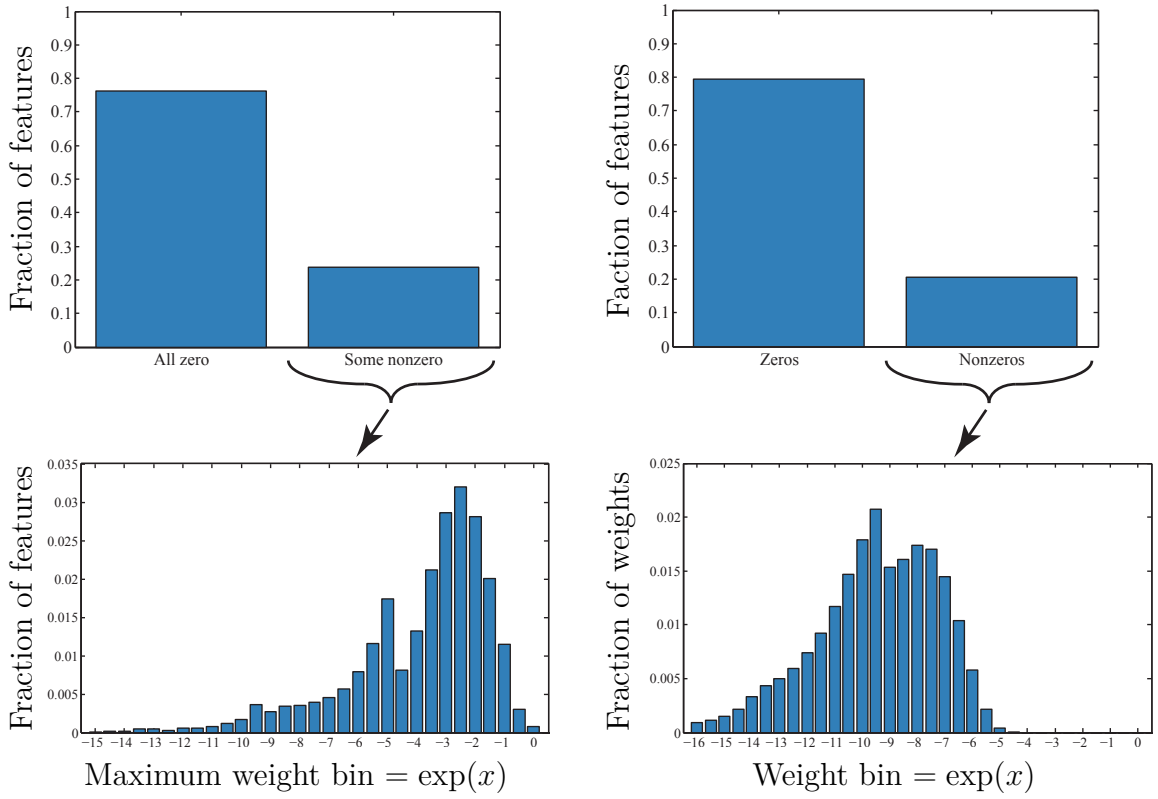
3.3.8 Implementation Details

For a large database of exemplar faces, storing weights v_j^r for all j and r across all types of landmarks could be memory prohibitive. Fortunately, most features have all zero weights, as shown in Figure 3.4 (a), which means we can significantly reduce the size of the database by completely removing them. Additionally, few weights from remaining features are significantly larger than zero, as shown in Figure 3.4 (b), which means they can be efficiently stored in sparse arrays. Thus, even with 68 landmarks, all of our weights fit into a 1.1GB file after 8-bit quantization.

We use $k = 200$ top retrieved exemplars for landmark localization and for training weights. We empirically set $\sigma_s = 0.1 \cdot \text{size}_r$ in Eq. (3.3), where size_r is the size of the r-th exemplar face, defined as the average height and width of the tightest bounding box that encloses all 68 ground truth landmarks. Intuitively, this setting ensures that a feature's influence will be effectively limited to landmarks on only the most nearby face part(s) (*e.g.*, a left eye feature will have negligible influence on right eye landmark estimates).

We empirically set $\sigma_g = 0.03 \cdot \text{size}_r$ in the Gaussian Ψ in Eq. (3.1). In practice, σ_g controls the degree to which the voting maps are smoothed. $\sigma_g = 0.03 \cdot \text{size}_r$ effectively smooths together votes that are within a few pixels of one another.

For face detection, we used our implementation of Shen *et al.*'s face detector [105]. We used their exemplar database to train their algorithm; most of their exemplars come from the AFLW dataset [60]. Please see [105] for more details. For landmark localization, all of our exemplar images come exclusively from the Multi-PIE Face Database [45]; please see Section 3.3.2 for details.



(a) Histogram of *maximum* weight associated with each feature

(b) Histogram of *all* weights among remaining (*i.e.*, not all zero) features

Figure 3.4: Feature weight histograms. (a) The distribution of the *maximum* weight associated with each feature. Because many features have all zero weights (76.3% in our implementation), we can completely remove them from the database. (b) The distribution of *all* weights among remaining (*i.e.*, not all zero) features from (a). Few weights from remaining features are significantly larger than zero (20.5% in our implementation), which means they can be stored efficiently in sparse arrays.

3.4 Results and Discussion

In this section we evaluate the accuracy of our approach and compare with several recent works [3, 7, 124, 130, 139, 142]. We show that our approach produces more accurate landmark estimates on especially challenging faces.

3.4.1 Experimental Datasets

We have evaluated our method on two publicly available datasets: AFW [142] and IBUG [99, 98]. We chose these two datasets because they each contain a large portion of faces with challenging head pose/camera viewpoint and/or facial expression. In contrast, other popular datasets such as LFPW [7], LFW [51], and Helen [74] contain predominantly frontal, and otherwise less widely varying test cases, which are consequently well-addressed by current, less robust methods. For example, the average landmark localization accuracy in [7] was shown to be slightly better than human labelers on LFPW.

For our quantitative results, we compared our landmark estimates with the ground truth annotations provided as part of the 300 Faces In-the-Wild Challenge (300-W) [99, 98]. Specifically, 300-W provides 68 landmarks per face according to the Multi-PIE arrangement [45] for 337 faces in AFW and 135 faces in IBUG. Typical AFW and IBUG faces are shown in Figure 3.5 with landmarks estimated by our algorithm overlaid in green.

3.4.2 Comparisons with Recent Work

We present cumulative error distribution (CED) curves in Figure 3.6 to quantitatively compare the accuracy of our method with six other state-of-the-art methods. For fair comparison in Figure 3.6 we evaluated only a subset of landmarks (49 out of 68: the eye, nose, and mouth landmarks, with inside mouth corners omitted) common to all method shown (*i.e.*, Xiong and De la Torre's [124] publicly available executable outputs only 49 landmarks). Unless otherwise noted, we evaluated each algorithm using the authors' original implementation.

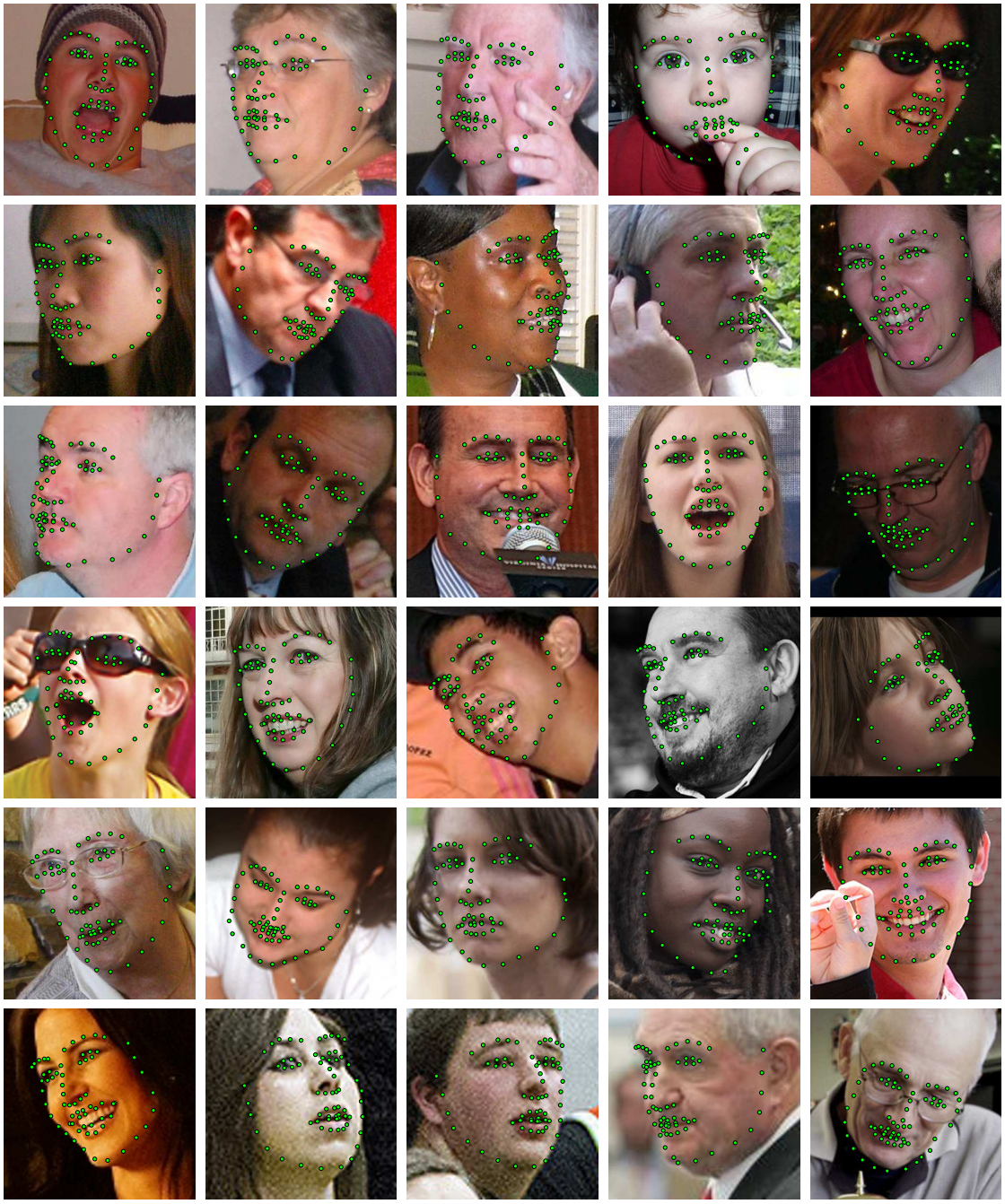
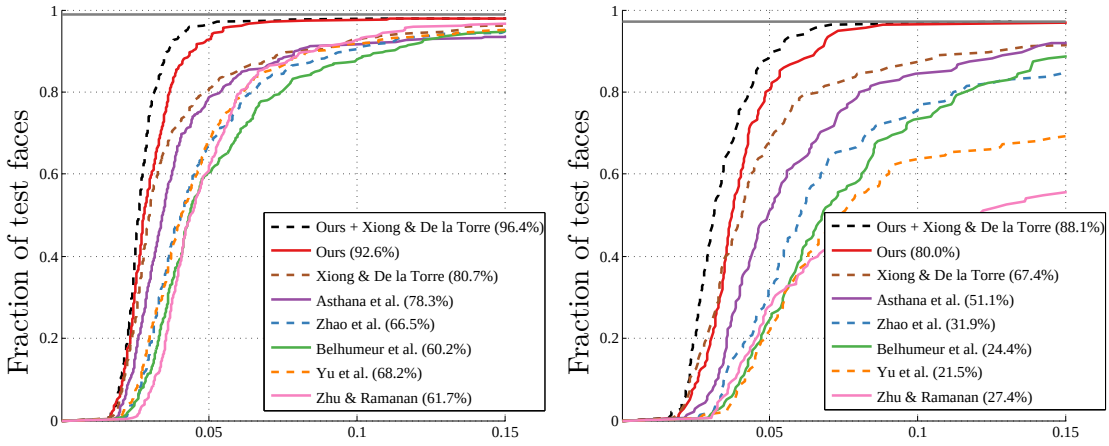


Figure 3.5: Selected qualitative results on two challenging evaluation datasets: AFW [142] (top three rows) and IBUG [99, 98] (bottom three rows). Our method can handle a wide variety of very challenging conditions, including significant image noise and blur, occlusions, and exaggerated expressions and head poses/camera viewpoints. Please see Appendix A for additional results. **Best viewed electronically in color.**



(a) AFW, 49 landmarks (b) IBUG, 49 landmarks

Figure 3.6: Two sets of cumulative error distribution (CED) curves comparing our accuracy with recent methods. In all cases, the average localization error is normalized by the face size, defined as the average height and width of the tightest bounding box that encloses all 68 ground truth landmarks [142]. For fair comparison with other works we evaluated on a subset of 49 landmarks (eye, nose, and mouth contour points, minus the inner mouth corners) common to all methods above (*i.e.*, Xiong and De la Torre output 49 landmarks). We see that our method performs significantly better than Zhu and Ramanan [142] and Yu *et al.* [130] despite the fact that these methods are designed to handle faces with non-frontal pose. Our method performs significantly better than Belhumeur *et al.* [7] and Xiong and De la Torre [124], although most of the errors in [124] are due to the initialization. When we initialize [124] using our landmark estimates, the accuracy is better than either approach in isolation, which suggests that our method is complementary to methods like [124].

Comparisons with Zhu and Ramanan [142] In their original evaluation, Zhu and Ramanan assigned an infinite localization error to the entire face if their algorithm incorrectly estimated the landmark arrangement (*i.e.*, if a frontal face was incorrectly labeled as a profile face with only 39 visible landmarks). In our evaluation, if an incorrect arrangement is given by their algorithm, we simply ignore the missing landmarks, and measure the mean error among the given landmarks.

Zhu and Ramanan's landmark localization algorithm is tied to their detection algorithm, and so we do not provide it with any kind of initialization. For each ground truth face annotation, we select the output face that has the largest bounding

box overlap (the area of intersection divided by the area of union), and we ignore all false positives. We set their detection threshold to $-\infty$ to avoid missing faces. Zhu and Ramanan provide three off-the-shelf models with their implementation, all trained on Multi-PIE. Although it requires the most computation time, we used their *Independent-1050* model for our experiments since it generally performed best.

Comparisons with Yu *et al.* [130] Yu *et al.* rely on a simplified version of Zhu and Ramanan's algorithm for initialization, and so we do not provide a separate initialization. However, the authors' implementation only returns landmark estimates for the highest scoring face in each image, which is a problem for test images with multiple faces. To obtain results for all annotated faces, we isolated each face from the rest of the image. Specifically, we cropped each annotated face using a box centered on the true face location with it's height and width set to be approximately twice the face height and width.

Comparisons with Belhumeur *et al.* [7] We used our own implementation of Belhumeur *et al.*'s algorithm, which we trained on our Multi-PIE exemplar dataset. We endeavored to reproduce their algorithm as faithfully as possible, although some subtle differences are inevitable. As suggested in [7], we placed a mean face shape over each face detection to initialize the location of each landmark detector window. Belhumeur *et al.* set the size of each landmark detector window to approximately 33% of the height/width of the tight face bounding box in their original work. Unfortunately, we found that this size did not always cover the true landmark locations in our experimental datasets, especially for non-frontal faces. As a compromise, we initialized the height and width of each detector window to the larger of: (1) 33% of the size of the tight face bounding box, or (2) large enough to cover the true landmark location plus 5% of the face size. To evaluate [7] on more challenging datasets (*i.e.*, AFW and IBUG), we had to use a more robust face detector, *e.g.* [105], for initialization. The Viola-Jones [118] detector used in [7] works well for near-frontal faces typical in datasets like LFPW and LFW, but misses many faces in AFW and IBUG (*e.g.*, Viola-Jones missed 62 out of 135 faces in IBUG, whereas [105] missed only 4).

Comparisons with Xiong *et al.* [124] We used Xiong and De la Torre's [124] publicly available implementation for evaluation. Because the training code for [124] is not publicly available, we used their off-the-shelf model. According to [124], their model is trained on Multi-PIE and LFW. Therefore, comparing their model to ours is reasonable. By default, Xiong and De la Torre's executable uses the Viola-Jones face detector for initialization. Because Viola-Jones misses so many faces in AFW and IBUG, we instead initialized [124] using the same procedure described above for Belhumeur *et al.*'s algorithm. We used this initialization because it is more realistic than simply using the ground truth face bounding box.

It is possible that the additional faces detected by [105] negatively impacted the performance of [124]. We observe that most of the localization errors from [124] arise when the initialization is far from the true landmark locations (*e.g.*, on faces with extreme head pose and/or expression), which suggests that [124] is sensitive to initialization. In such cases, [124] fails to converge to the correct solution. However, in cases where the initialization is relatively close to the true landmark locations, their algorithm performs slightly better than ours. We demonstrate this in Figure 3.6, where we also show that the accuracy of their algorithm initialized using our estimates (labeled "Ours + Xiong & De la Torre" in each plot), is higher than either approach in isolation. In this way our approach is complementary to [124].

Comparisons with Asthana *et al.* [3] Asthana *et al.*'s implementation provides three modes for initialization: (1) the localization results from [142], (2) MATLAB's Viola-Jones face detector, or (3) a face bounding box. For fair comparison, we elected to use (3) with each bounding box computed by [105] (*i.e.*, the same face detector used in our pipeline).

Comparisons with Zhou *et al.* [139] Zhou *et al.*'s implementation relies on eye center locations (*e.g.*, provided by an eye detector) to initialize the face shape. We note that, like other AAM-based approaches, their algorithm is sensitive to initialization. Therefore, we provided their algorithm with ground truth eye centers. Rather than using a single AAM model, Zhou *et al.* compute a separate AAM for each test image at runtime. For training, we provided their algorithm with our Multi-PIE exemplar dataset.

3.4.3 Runtime

On a 900×600-pixel image with one face, the overall runtime of our MATLAB implementation of Shen *et al.*'s face detector [105] is 42.8 seconds on an Intel Xeon E5-2670 workstation; our landmark localization algorithm, also implemented in MATLAB, requires an additional 25.5 seconds. This is similar to the runtime of [142] (using their *Independent-1050* model) and our MATLAB implementation of [7], but is much slower than several recent methods (*e.g.*, [3, 124, 130]) that are designed to run in real time. However, we note that many strategies exist to speed up our implementation. For example, although the stages of our pipeline must run sequentially, each stage represents an embarrassingly parallel workload (in the parlance of parallel computing), and the size of each landmark voting map, which currently span the entire face, could be reduced significantly by employing a multi-resolution image pyramid.

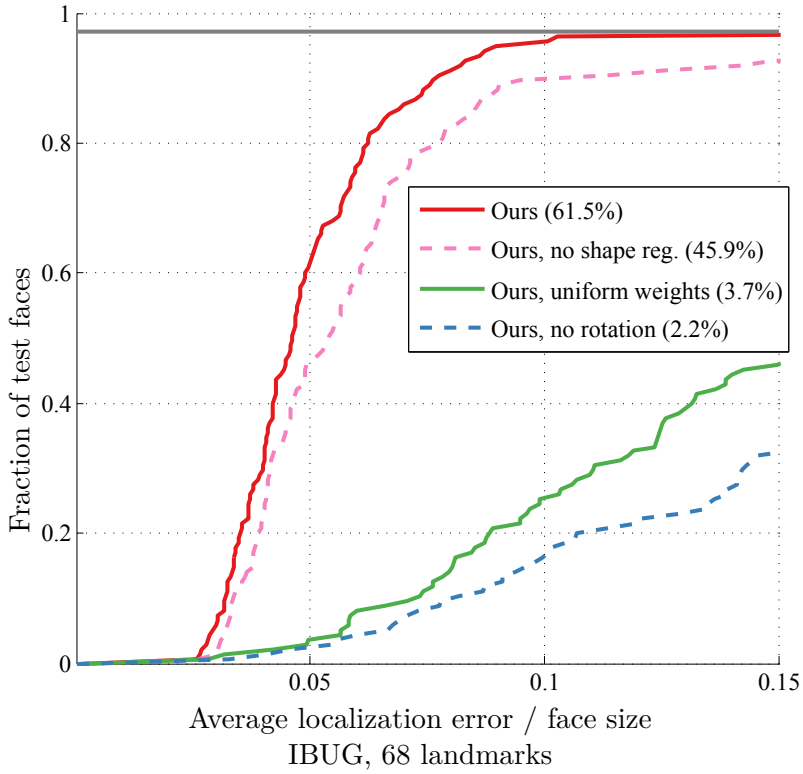


Figure 3.7: Cumulative error distribution (CED) curves showing the impact of different parts of our pipeline on performance. The average localization error is normalized by the face size, defined as the average height and width of the tightest bounding box that encloses all 68 ground truth landmarks [142].

3.4.4 Impact of Different Algorithm Parts

Figure 3.7 shows the relative impact of different parts of our pipeline on accuracy. We see that shape regularization noticeably boosts our accuracy. Key to the success of our method is a data-driven approach for weighting the feature votes. We see in Figure 3.7 that performance drops dramatically without these weights. We also see in Figure 3.7 that our voting-based approach is sensitive to rotation misalignment between the test face and the exemplar faces, thus necessitating a rotation search. This is partly a function of our exemplar faces, which are rotation normalized. Another strategy would be to expand our exemplar database to include rotated versions of faces, and handle in-plane rotation nonparametrically. Other methods don't perform an explicit grid search over rotations.

Chapter 4

Collaborative Face Image Parsing for Transferring Landmark Annotations Across Datasets

4.1 Introduction

Many datasets have also been proposed to evaluate face alignment and landmark localization methods, from early datasets collected in the lab like CMU Pose Illumination Expression (PIE) [106], Multi-PIE [45], AR [88], and eXtended Multi Modal Verification for Teleservices and Security applications DataBase (XM2VTSDB) [90], to more recent in-the-wild datasets like Labeled Face Parts in the Wild (LFPW) [7], Annotated Facial Landmarks in the Wild (AFLW) [60], Annotated Faces in the Wild (AFW) [142], Helen [74], and IBUG [99, 98].

On one hand, new datasets pose new challenges to the research community and foster new ideas. On the other hand, as researchers, we always have to choose specific datasets for evaluation to publish our work, which becomes increasingly difficult because different datasets have different landmark definitions (for example, AFLW uses a 21-landmark markup, while Helen uses 194 contour points). As a result, models trained on one dataset often cannot be evaluated on other datasets. Furthermore, inconsistencies between datasets make it difficult to train robust landmark localization models that combine all the different datasets.

Ideally, it would be desirable to have a common and unified definition of landmarks and collect datasets following the same definition. However, this goal is challenging in practice because the speed of collecting labels will always lag behind the speed of collecting face data. Furthermore, it is difficult to predict which landmark definitions (*e.g.*, ears) new applications will find useful.

We make the first effort, to the best of our knowledge, to combine multiple face landmark datasets with different landmark definitions into a super dataset, with a union of all landmark types computed in each image as output. Specifically, we present a novel pipeline built as an extension of the state-of-the-art facial landmark localization method described in Chapter 3. Our system labels images in the target dataset jointly rather than independently and exploits exemplars from both the source datasets and the target dataset. This approach allows us to integrate nonparametric appearance and shape modeling and graph matching together to transfer annotations across datasets. Toward this goal, this chapter makes the following contributions:

1. A pipeline that transfers landmark annotations from multiple source datasets to never-before-labeled datasets.

2. An algorithm that takes multiple source datasets as input and labels a partially labeled target dataset using a union of landmarks defined in the source datasets. Our system can optionally use known landmarks in the target dataset as constraints.

3. 63 supplementary landmarks for faces in the AFLW database [60], for a total of 84 landmarks. AFLW is significant in that, to the best of our knowledge, it is currently the largest publicly available in-the-wild face dataset with 25k annotated faces.

4.2 Related Work

To the best of our knowledge, we know of no other systems that try to solve the same problem considered here: that is, *given multiple source datasets, estimate a union of landmarks for a given target dataset.* However, components of our system are inspired by and/or are built upon existing methods in the literature, which we summarize below.

Like [24, 36, 105, 126], we use a Hough voting approach to generate landmark response maps in Stage 2 of our system. Yang and Patras [126] use several 'sieves' to filter out votes that are not relevant. Cootes *et al.* [24], Dantone *et al.* [36], and Shen *et al.* [105] weight each vote based on heuristics. In our approach, we adjust the weight of each vote by considering how well it agrees with other votes from matched features in other images.

Our landmark detection algorithm optionally uses known landmarks in the target image as constraints. Cootes and Taylor [26] proposed a constrained AAM that utilizes some known landmarks in the target image; AAMs are parametric models, while our approach is nonparametric and exemplar-based. Sagonas *et al.* [99] proposed a semi-automatic method for creating facial landmark annotations using person-specific models. Their process is iterative: users label results as 'good' or 'bad', and good results are used in later iterations as training data. In contrast, our method is fully automatic; our system has the ability to take user input, but we do not consider it in our experiments.

Exemplar-based approaches have been popular since Belhumeur *et al.*'s pioneering work [7]. Zhao *et al.* [139] use grayscale pixel values and HOG features to select k-nearest neighbor training faces, from which they construct a target-specific AAM at runtime. Shen *et al.* [105] perform Hough voting using k-NN exemplar faces; we use the same basic approach in our system. Finally, Zhou *et al.* [141] combine an exemplar-based approach with graph matching for robust facial landmark localization. We extend Zhou *et al.*'s approach to integrate different landmarks from multiple source datasets.

4.3 Approach

In this section we first give a brief overview of our system followed by a more detailed explanation of each stage in subsequent sections.

4.3.1 Overview

The input to our system is one or more *source* face datasets, and one *target* face dataset. We assume that each source dataset consists of a set of face images, in which each image is labeled with a set of facial landmarks, *e.g.*, eye centers, mouth corners, nose tip. Importantly, we do not require the landmark definitions to be consistent between source datasets. Optionally, each target image can have known landmarks, which our system uses as additional constraints. The output of our system is a combined set of landmark estimates (*i.e.*, the union set of landmark types from all source datasets) for each target face.

Stage 0: Preprocessing

We first rotate and scale all faces such that the eyes are level and the size is the same across all face instances. We then extract dense SIFT [84] features across each face at multiple scales. Following the approach in [105], we quantize each SIFT descriptor using fast approximate k-means [91], which efficiently maps each descriptor to a visual word.

Stage 1: Selection of Top Source Faces

For each target face, retrieve a separate subset of top k similar faces from each source dataset. The goal is to retrieve faces that are similar to the target face in appearance, shape, expression, and pose so that features in the source images will produce accurate landmark votes in the target image.

Stage 2: Weighted Landmark Voting

For each target face, independently compute a separate voting map for each landmark type from each source dataset using a generalized Hough transform [75]. Each feature

from the top k source faces casts a vote for a possible landmark locations in the target image.

Stage 3: Shape Regularization

For each target face, compute a separate set of landmark estimates from each source dataset. Due to local ambiguities, occlusions, noise, *etc*. each voting map may contain multiple peaks. We employ a robust nonparametric shape regularization technique [7] that avoids false peaks and estimates a globally optimized set of landmark estimates from each source dataset.

Stage 4: Final Landmark Estimation and Integration

For each target face, retrieve the top m most similar faces from the *target* dataset. The goal is to exploit the correlation between landmark estimates from Stage 3 among similar target faces to consistently label all target images. We combine estimates for landmarks common to multiple source datasets, and we optionally use known landmarks in each target image to constrain the optimization. We extend the graph matching technique in [141] for landmark integration from multiple source datasets. The final output for each target face is a full set of landmark estimates; by "full" we mean the union of landmark types from all source datasets.

4.3.2 Stage 1: Selection of Top Source Faces

To transfer landmarks from each source dataset to the target image, the shape and appearance of the source faces and the target face should not be too different. For example, a frontal face has much different appearance and shape than a profile face; there are few geometric feature-landmark correlations between the two. We therefore select a top subset of source faces for further processing.

Many strategies exist for retrieving similar face images from a database. In our system, we use a generalized Hough transform framework to score each source face. Specifically, we use the features on the target face to vote for the center of the face in each source image. The final score for each source face is the height of the maximum peak in the voting map associated with each source image. The intuition is that source faces with many shared features in similar geometric layouts with the target image

Input: One or more *source* face datasets, and one *target* dataset
Output: A combined set of landmark estimates for each target face

Stage 0: Preprocessing
for all target faces **do**
  Stage 1: Selection of Top Source Faces
  Stage 2: Weighted Landmark Voting
  Stage 3: Shape Regularization
end
for all target faces **do**
  Stage 4: Final Landmark Estimation and Integration
end

Figure 4.1: Overview of our pipeline for transferring landmark annotations across face datasets. Stage 4 is in a separate loop because it uses all the target face results from Stage 3 to help constrain and consistently estimate the final landmark results.

will produce many consistent votes for the center of the face. We sort the scores and select the top $k = 200$. This is the same procedure described in Step 1 of our facial landmark localization algorithm (Chapter 3).

4.3.3 Stage 2: Weighted Landmark Voting

For efficiency, rather than exhaustively sliding each source landmark region over the target image, we use quantized features and employ an inverted index file to efficiently retrieve matched features (*i.e.*, features in the same quantization bin) from the top k source images. When a feature in the target image is matched to a feature in a source image, the feature-to-landmark offset in the source image is transferred to the target image. The offset vector extends from the feature in the target image toward a potential landmark location and produces a vote. After many such votes, a voting map is formed, where the votes tend to cluster at landmark locations.

In practice, due to errors in the feature quantization step, image noise, occlusions, uniform and locally ambiguous image regions, *etc.*, many of the votes are incorrect, which can significantly impact overall voting accuracy. In Section 3.3.7 we described an *offline* procedure for precomputing feature weights for landmark localization. This was necessary to speed up the execution of our algorithm at test time.

However, because we are concerned with annotating whole *datasets* rather than test images in this chapter (*i.e.*, for future use as a training dataset), the 'online' speed is not the main priority. Therefore, we instead compute a weight for each vote *online* as follows. For a given feature in the target image, retrieve all features in the top k source images that share the same quantization bin. For each of these features, compute their offset from landmark l. After rejecting outlier votes (*i.e.*, by measuring the distribution and rejecting vote offsets outside the inter-quartile range), we compute the variance σ_v^2 of the remaining offsets. We then cast a 'fuzzy' vote from each offset using a 2D Gaussian $\mathcal{N}(v; \sigma^2)$ centered on the vote location v. Intuitively, this rewards matched features that produce consistent voting offsets and suppresses features that disagree.

We note that the Hough voting strategy is sensitive to scale and rotation differences between source and target faces. Shen *et al.* [105] addressed this problem by performing Hough voting over multiple scales, and we addressed it in Chapter 3 by searching over multiple scales and rotations. Here, we instead normalize the scale and orientation of each face in Stage 0, which eliminates the need to search for scale and rotation parameters.

4.3.4 Stage 3: Shape Regularization

There are many strategies for enforcing shape constraints (*e.g.*, [7, 104, 124, 142, 141]). However, in our case, we use an exemplar-based approach to shape regularization [7], which fits nicely with our exemplar-based Hough voting strategy for generating landmark response maps. We use the same procedure here as described in Step 4 of our landmark localization pipeline in Chapter 3.

4.3.5 Stage 4: Final Landmark Estimation and Integration

Stages 0 through 3 are very similar to the steps in our core landmark localization algorithm, as described in Chapter 3. This stage embodies the main extensions our original algorithm. The goal here is to combine the individual landmark estimates from each source dataset into a single result for each target image. We incorporate several constraints into the optimization:

1. We model each landmark location as a linear combination of the other landmarks, which provides an affine-invariant shape constraint [141].

2. If available, known landmarks in the target image are fixed and help steer nearby landmark estimates to their correct locations.

3. Only one estimate is allowed for each landmark type irrespective of the number of source datasets contributing estimates for each type.

We address the shape constraint first. Let $\mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_{N_\mathbf{P}}]$ be a face shape composed of $N_\mathbf{P}$ landmarks. Following [141], we assume that the c-th landmark location $\mathbf{p}_c$ can be reconstructed by a linear combination of neighboring landmarks: $\mathbf{p}_c = \mathbf{P}\mathbf{w}_c$, where $\mathbf{w}_c \in \mathbb{R}^{N_\mathbf{P}}$ is a vector of weights for the other $N_\mathbf{P} - 1$ landmarks (the c-th entry of $\mathbf{w}_c$ is zero).

Suppose we have N_S source datasets and therefore N_S {target image t, source dataset s} pairs. Each pair has a union set of landmark types, $L_{ts} = \{L_t \cup L_s\}$, composed of landmark types L_t defined in the target image[1] or landmark types L_s defined in the source dataset. In other words, each L_{ts} contains all landmark types either known *a priori* in target image t, or estimated from source s or both. We aim to compute an optimal $\mathbf{w}_{tsc}$ for $c \in L_{ts}$ for each {target image t, source dataset s} pair (we subsequently omit t and s subscripts in $\mathbf{w}_{tsc}$ for simplicity).

To accomplish this task we need a set of example shapes that include all landmark types in L_{ts}. Given a target image t, we retrieve the m most similar face shapes among the target face images; the face shapes for the target images come from the regularized landmark localization results from Stage 3. As a distance metric, we simply use the mean Euclidean error between shapes after similarity transformation alignment. Using [141], we compute the $\mathbf{w}_c$ for image t that minimizes the sum of reconstruction errors among the top m most similar shape results from Stage 3:

$$\begin{aligned} \min_{\mathbf{w}_c} \sum_{j}^{m} ||\mathbf{P}^j\mathbf{w}_c - \mathbf{p}_c^j||_2^2 \;\; &+ \;\; \eta||\mathbf{w}_c||_2^2 \\ \text{s.t.} \;\; \mathbf{w}_c{}^\mathsf{T}\mathbf{1}_{N_\mathbf{P}} &= 1, \\ w_{cc} &= 0, \\ w_{cr} &= 0 \;\; \forall r \notin L_{ts}, \end{aligned} \tag{4.1}$$

[1] L_t can be empty, in which case the target dataset has no known landmarks.

where $\mathbf{P}^j$ is the j-th most similar face shape relative to t among other results from Stage 3; the constraint $w_{cr} = 0\ \forall r \notin L_{ts}$ means that we force weights to zero if the r-th landmark is undefined in L_{ts}; and $\eta||\mathbf{w}_c||_2^2$ is a regularization term that penalizes the sparsity of the weight vector, *i.e.*, it promotes more uniformity in the weights, which means that non-local landmarks can also carry importance in determining the c-th landmark location. Eq. (4.1) is a small convex quadratic problem, which we solve independently for each $\mathbf{w}_c$. The formulation of Eq. (4.1) is the same as [141] except for our added third constraint.

We compose the joint weight matrix as $\mathbf{W}_s = [\mathbf{w}_1, \ldots, \mathbf{w}_{N_\mathbf{P}}]$, and we repeat the process for each source dataset s to create a set of N_S joint weight matrices $\mathbf{W}_1, \ldots, \mathbf{W}_{N_S}$ specific to target image t. Note that undefined columns in each $\mathbf{W}_s$ (corresponding to landmarks not defined in L_{ts}) are set to zero.

Following [141], let us now define a global coordinate matrix $\mathbf{Q} = [\mathbf{Q}_1, \ldots, \mathbf{Q}_{N_\mathbf{P}}] \in \mathbb{R}^{2\times N}$, where $\mathbf{Q}_c \in \mathbb{R}^{2\times N_c}$ denotes candidate locations for the c-th landmark and $N = \sum_c N_c$. Let $\mathbf{G} \in \{0,1\}^{N_\mathbf{P}\times N}$ be a binary association matrix, where $g_{ci} = 1$ if the i-th point belongs to the c-th landmark. Note that the candidate locations are the locations of the local peaks in the landmark response maps in Stage 3. When a landmark is common in multiple source datasets, we average the response maps from different source datasets before finding the local peaks. Let $\mathbf{A} \in \mathbb{R}^{N_\mathbf{P}\times N}$ denote the assignment cost matrix, *i.e.*, $a_{ci} = -\log(R_c(\mathbf{q}_i))$, where $R_c(\mathbf{q}_i)$ is the height value in the c-th voting map at $\mathbf{q}_i$ after the voting map is normalized to sum to 1.

Given the candidates $\mathbf{Q}$, $\mathbf{G}$, $\mathbf{A}$ and the shape constraints $\mathbf{W}_1, \ldots, \mathbf{W}_{N_S}$, the problem consists of finding the optimal correspondence $\mathbf{X}$ that minimized the following error:

$$\begin{aligned} \min_{\mathbf{X}} \quad \lambda \mathsf{tr}(\mathbf{A}\mathbf{X}^\mathsf{T}) \ &+ \ \sum_s^{N_S} \|\mathbf{Q}\mathbf{X}^\mathsf{T}(\mathbf{I}_s - \mathbf{W}_s)\|_1 \\ \mathsf{s.t.}\ \mathbf{X}\mathbf{1}_N \ &= \ \mathbf{1}_{N_\mathbf{P}}, \\ \mathbf{X} \ &\in \ \{0,1\}^{N_\mathbf{P}\times N}, \\ x_{ci} \ &= \ 0, \quad [c,i] \in \{[c,i] \mid g_{ci} = 0\}, \end{aligned} \tag{4.2}$$

where $\mathbf{I}_s$ is an $N_\mathbf{P} \times N_\mathbf{P}$ identity matrix except that we set $\mathbf{I}_s(r,r) = 0\ \forall r \notin L_{ts}$ (*i.e.*, the r-th diagonal element is set to zero if landmark r is not defined in the target

image or in dataset s). Eq. (4.2) is inspired by [141] except here we sum over multiple shape constraint terms instead of just one. λ is a regularization weight that balances feature cost and reconstruction error. The first constraint, $\mathbf{X}\mathbf{1}_N = \mathbf{1}_{N_\mathbf{P}}$, enforces a many-to-one mapping in $\mathbf{X}$. The remaining constraints forces $\mathbf{X}$ to select only a single landmark from the candidates in $\mathbf{G}$.

Due to the integer constraint on $\mathbf{X}$, optimizing Eq. (4.2) is NP-hard. Like [141], we solve Eq. (4.2) by relaxing the integer constraint with a continuous one, $\mathbf{X} \in [0,1]^{N_\mathbf{P} \times N}$, and then we reformulate the problem using the trick in [53, 79] by incorporating auxiliary variables $\mathbf{U}_s, \mathbf{V}_s \in \mathbb{R}^{2 \times N_\mathbf{P}}$ that replace the non-smooth ℓ_1 norm with a smooth term and a linear constraint:

$$\begin{aligned} \min_{\mathbf{X},\mathbf{U},\mathbf{V}} \quad \lambda \mathsf{tr}(\mathbf{A}\mathbf{X}^\mathsf{T}) \;&+\; \sum_s^{N_S} \mathbf{1}_2{}^\mathsf{T}(\mathbf{U}_s + \mathbf{V}_s)\mathbf{1}_{N_\mathbf{P}} \\ \text{s.t. } \mathbf{Q}\mathbf{X}^\mathsf{T}(\mathbf{I}_s - \mathbf{W}_s) &= \mathbf{U}_s - \mathbf{V}_s, \\ \mathbf{U}_s &\geq \mathbf{0}_{2 \times N_\mathbf{P}}, \\ \mathbf{V}_s &\geq \mathbf{0}_{2 \times N_\mathbf{P}}, \\ \mathbf{X} &\in [0,1]^{N_\mathbf{P} \times N}, \\ x_{ci} &= 0, \quad [c,i] \in \{[c,i] \mid g_{ci} = 0\}. \end{aligned} \tag{4.3}$$

MATLAB's LP solver can be used to find the optimal solution of the above minimization problem. Please see [141] for more details.

Incorporating known landmarks in the target image as constraints in Eq. (4.2) is straightforward. We simply provide only a single candidate location for each of the known landmarks via the matrices $\mathbf{Q}$ and $\mathbf{G}$.

Because we use the same correspondence matrix $\mathbf{X}$ for all terms in Eq. (4.2), we will obtain only one estimate for each landmark type, regardless of how many source datasets contribute to the estimate.

4.3.6 Implementation Details and Runtime

For quantizing SIFT features we use fast approximate k-means [91] with $k = 10^5$ clusters. For efficiency, we quantize the spatial variance σ_v^2 measurement of each vote cluster in Section 4.3.3 and convolve each voting map after all voting is complete using

a set of precomputed Gaussian kernels. We also threshold σ_v to prevent erroneous spikes in the voting maps: we do not allow σ_v to fall below 3 pixels. In Stage 4, we set $\eta = 1000$, $\lambda = 100$, and we use about 200 candidates for each landmark.

Because our system operates on face datasets, we consider our pipeline to be entirely 'offline.' However, it is not prohibitively slow despite the number of steps involved. All tests were conducted on an Intel Xeon E5-2670 workstation. For each 480×480 image in our evaluation set, feature extraction and quantization takes less than a second. For each {target image, source dataset} pair, top exemplar selection (Stage 1) takes approximately 2.5 seconds, landmark voting (Stage 2) across 84 landmarks takes approximately 15 seconds, and shape regularization (Stage 3) takes approximately 10 seconds using our MATLAB implementation. The final stage is the most expensive (approximately 30 seconds per image) in part because MATLAB's linear program solver is relatively slow with many landmarks and candidate locations. We remark that most stages in our pipeline can be easily parallelized.

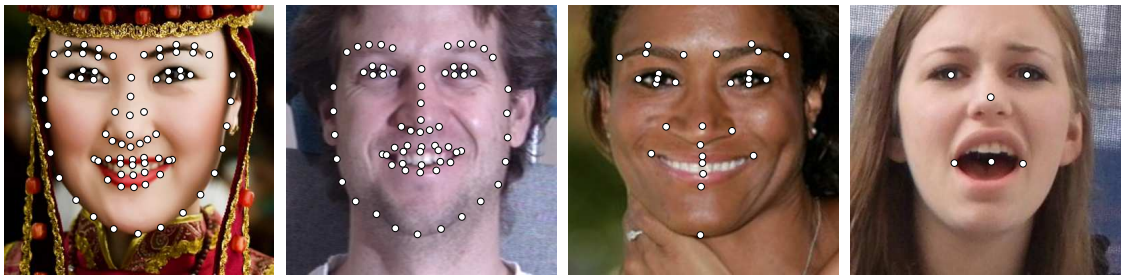
4.4 Results and Discussion

In this section we present two groups of experiments to evaluate the accuracy of our approach. In the first group, the source dataset consists of several subsets, each with a different set of landmark definitions. Each target image has no known landmarks, and our algorithm estimates a union of landmarks for each target image from the source datasets. This group of experiments allows us to compare our results with recent landmark localization works [3, 7, 117, 124, 130, 139, 141, 142]. We show state-of-the-art accuracy on especially challenging "in-the-wild" faces. In the second group of experiments we show that our approach can simultaneously (1) integrate landmark annotations from multiple source datasets and (2) effectively make use of known landmarks in the target dataset to significantly reduce errors among the estimated landmarks. These experiments show that our approach is well-suited to automatically supplementing an existing dataset with additional landmarks from other source datasets.

4.4.1 Experimental Datasets

We used five face datasets for our quantitative evaluation: Multi-PIE [45], Helen [74], LFPW [7], AFW [142], and IBUG [98]. In the literature, there are two versions of landmark annotations for Helen, LFPW, and AFW: (1) the annotations provided when the datasets were originally released, which we refer to as "original" hereafter; and (2) the recent annotations provided as part of the 300 Faces in-the-wild Challenge (300-W) [98], which we refer to as "300-W" hereafter. We use both versions of the landmarks; the details are described in the context of individual experiments.

As in [142, 117, 130], we measure the size of the face as the average of the height and width of the rectangular hull around the ground truth landmarks. We favor this size measurement over inter-ocular distance (IOD) because it is more robust to yaw head rotation. Prior to testing, we rescaled all faces to a canonical size (200 pixels) and rotated them to make the eyes level.



(a) Helen (b) Multi-PIE (c) LFPW (d) IBUG, AFW

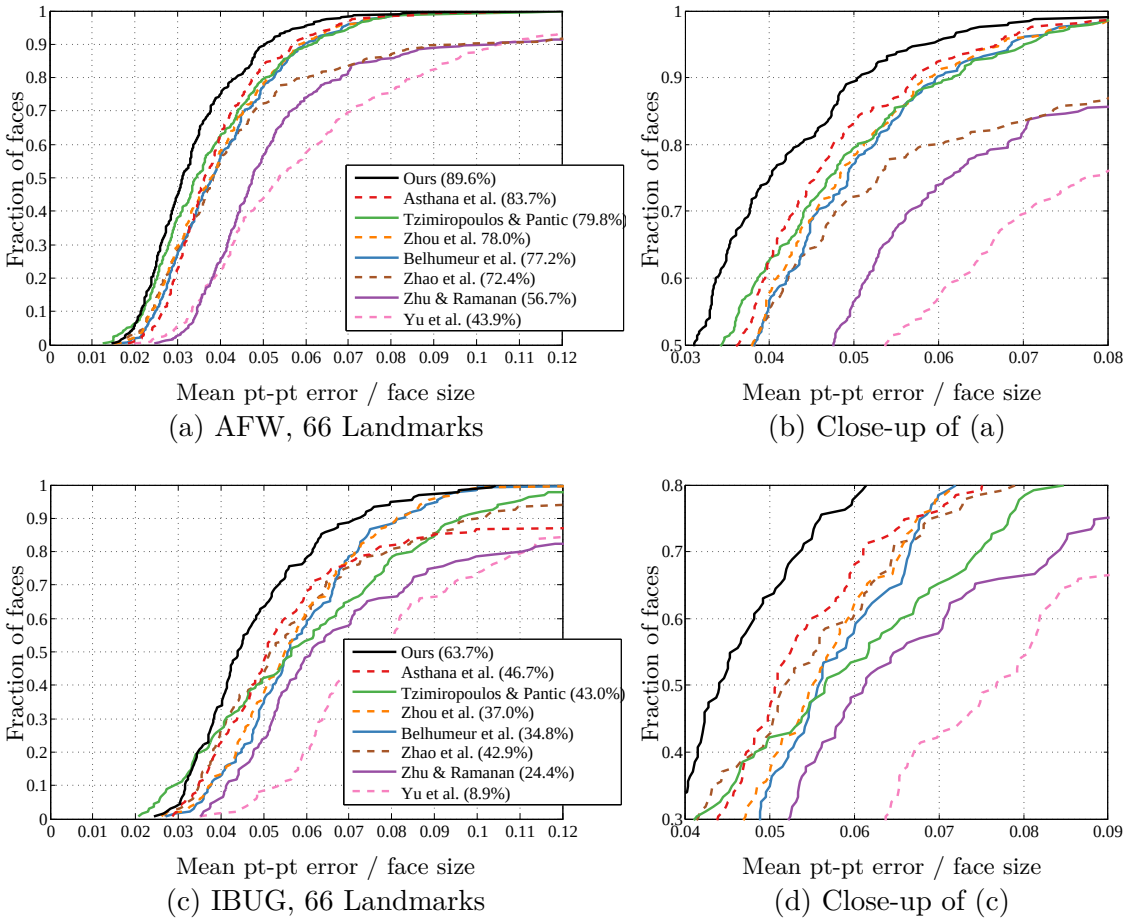
Figure 4.2: Experimental datasets. When each dataset is acting as a source, we use the landmark annotations shown above. Across all source datasets there are 85 unique landmark types.

4.4.2 Comparisons with Recent Work

In this section we quantitatively compare our algorithm with recent works [7, 142, 141, 3, 117, 130, 139]. The source datasets for training consist of Multi-PIE, Helen, and LFPW. For our algorithm, we used the ground truth landmark annotations shown in Figure 4.2 for Multi-PIE, Helen, and LFPW as training. The ground truth landmarks come from both the original annotations and the 300-W annotations (300-W annotations are favored in cases of redundant definitions). We note that there are 85 unique landmark types across all source datasets.

Our target datasets for testing are AFW [142] and IBUG [98]. We use these two datasets for evaluation because they are particularly challenging; they include a large percentage of faces with extreme facial expression and/or head pose. In contrast, other popular datasets like BioID [52], Helen [74], LFW [51], and LFPW [7] contain faces with less challenging variations, which are consequently well-addressed by current methods. For example, among recent methods (*e.g.*, [7, 141, 3]) the average point-to-point error for estimated landmarks is less than 10% of the IOD for more than 95% of LFPW faces.

We made every effort to implement Belhumeur *et al.* [7] and Zhou *et al.* [141] algorithms faithfully; we trained them on the source dataset (Multi-PIE, Helen, and LFPW) using only 300-W annotations. For all other algorithms, we used the original authors' implementations. We used the off-the-shelf models provided with each implementation, with the exception of Zhao *et al.* [139]. Zhao *et al.* compute



(a) AFW, 66 Landmarks (b) Close-up of (a)

(c) IBUG, 66 Landmarks (d) Close-up of (c)

Figure 4.3: Two sets of cumulative error distribution (CED) curves on AFW and IBUG face datasets. In all cases, the average localization error is normalized by the face size as defined in [142]. Here we compare the accuracy of our approach with seven recent works: Asthana *et al.* [3], Tzimiropoulos and Pantic [117], Zhou *et al.* [141], Belhumeur *et al.* [7], Zhao *et al.* [139], Zhu and Ramanan [142], and Yu *et al.* [130]. We see that our approach generally produces more accurate results among those evaluated above. **Best viewed in color.**

target face-specific models online from a given training database; for their algorithm, like Belhumeur *et al.* [7] and Zhou *et al.* [141], we provided Multi-PIE, Helen, and LFPW faces as training data using only 300-W annotations.

Initialization

For Belhumeur *et al.* [7] and Zhou *et al.* [141] we initialized the position of each landmark detector using a mean face shape aligned to the face. The diameter of each detector window was set to the larger of 33% of the face size or large enough to overlap the true landmark location. Zhao *et al.*'s implementation [139] is initialized via eye center detectors; we therefore provided their algorithm with ground truth eye centers. Tzimiropoulos and Pantic's [117] algorithm takes a face bounding box as input; for this we provided the ground truth bounding box.

Zhu and Ramanan's [142] algorithm is tied to their detection algorithm, and so we do not provide it with an initialization. We set their detection threshold to $-\infty$ to avoid missing faces. For each ground truth face annotation, we select the output face that has the largest bounding box overlap (the area of intersection divided by the area of union), and we ignore all false positives. Zhu and Ramanan provide three models with their implementation. Although it requires the most computation time, we used their Independent-1050 model for all of our experiments since it generally performs best.

Asthana *et al.* [3] and Yu *et al.* [130] each rely on a version of Zhu and Ramanan's [142] algorithm for initialization, and so we do not provide a separate initialization. However, since Yu *et al.*'s implementation only returns landmark estimates for the highest scoring face in each image, we isolated the true face by cropped it out (the crop window was centered on the true face and set to approximately twice the face height/width).

Quantitative Results

Figure 4.3 shows two sets of cumulative error distribution (CED) curves, which compare the accuracy of our approach with others. Using Multi-PIE, Helen, and LFPW as source datasets, our algorithm produces 84 landmark estimates (a union of both 300-W and original annotations from the three source datasets).[2] We evaluated the accuracy

[2]We supplemented the 300-W annotations on Helen with 10 landmarks from the original annotations (3 on each eyebrow, and 4 on the nose). When we use LFPW as a source dataset, we use only the 29 landmarks from the 300-W annotations that coincide with the original annotations. When we use AFW and IBUG as source datasets, we use only the 6 300-W annotations that coincide with the original AFW annotation definitions. Figure 4.2 shows the layout of landmarks for each source dataset.

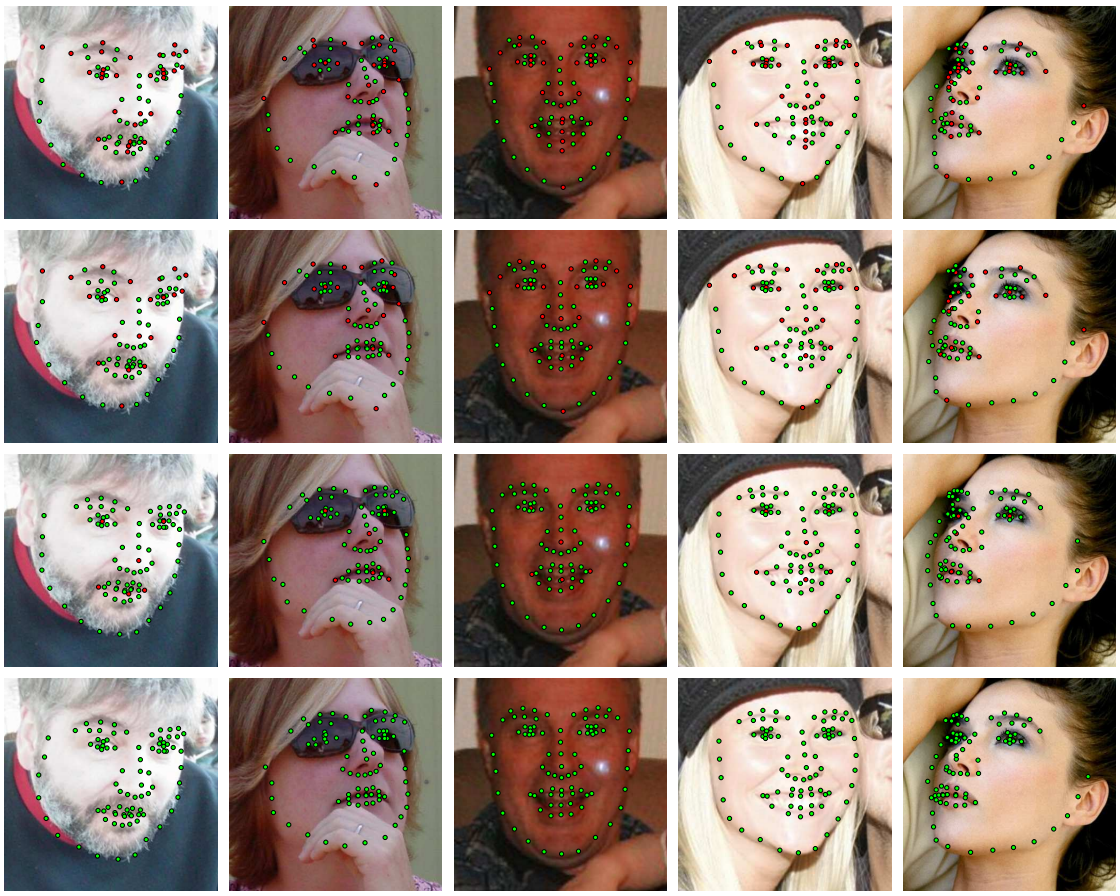
of all algorithms on 66 landmarks in Figure 4.3 because [3] estimates 66 landmarks. Errors are computed relative to the 300-W ground truth landmarks as the mean point-to-point error normalized by the face size. We see that our approach (assuming no known landmarks in the target dataset) produces state-of-the-art accuracy on AFW and IBUG faces.

4.4.3 Evaluation with Known Target Landmarks

We have quantitatively evaluated our full pipeline using 1035 images from LFPW as the target dataset, and using Multi-PIE, Helen, AFW, and IBUG as source datasets. The union of the different annotation definitions results in a total of 85 landmarks. For each target image, our algorithm estimates 85 landmarks. We performed six different trials, with each trial assuming a different number of known landmarks in the target dataset: 0, 6, 9, 15, 21, and 32. Among these different numbers, we chose 6 because it corresponds to the original AFW annotations; we chose 21 because it corresponds to annotations provided in the AFLW dataset [60]; and we chose 32 because it closely resembles the original annotations in LFPW. The top of Figure 4.5 shows the arrangement of known landmarks for each trial. For each face, we measure the accuracy of 79 landmarks (out of 85 estimated) relative to ground truth annotations from 300-W and the original LFPW dataset; the ground truth of the remaining 6 landmarks are not available for LFPW.

The CED curves in Figure 4.5 show the accuracy of our algorithm on each of these trials across 79 face landmarks. We see that the accuracy of our algorithm is high with 0 known landmarks (96.5% at 0.05 average overall), and the accuracy continues to improve with additional known landmarks.

A prime target dataset for our approach is AFLW [60], which contains 25k in-the-wild face images from Flickr, each manually annotated with up to 21 sparse landmarks. Our approach is well-suited to automatically supplementing AFLW with additional landmarks from source datasets like Multi-PIE [45] and Helen [74]. Please see Appendix B for selected qualitative results.



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

Figure 4.4: Qualitative results on AFW faces (a)-(d) and IBUG faces (e) with varying numbers of known landmarks in the target images. Green points are estimated landmark locations, and red points are known landmark locations. From the top row to the bottom row, results were computed with 32, 21, 6, and 0 known landmarks. We see that errors are corrected with additional known landmarks, *e.g.*, the eyebrows in (a) and (d), and the lips in (c) and (e). Even with no known landmarks (bottom row), our algorithm performs well on challenging faces, including those with significant head pitch rotation (a and e), head yaw rotation (a, b, d, and e), occlusion (b), and facial hair (a). Figure 4.3 shows a quantitative evaluation with no known landmarks in the target images. **Best viewed electronically in color.**

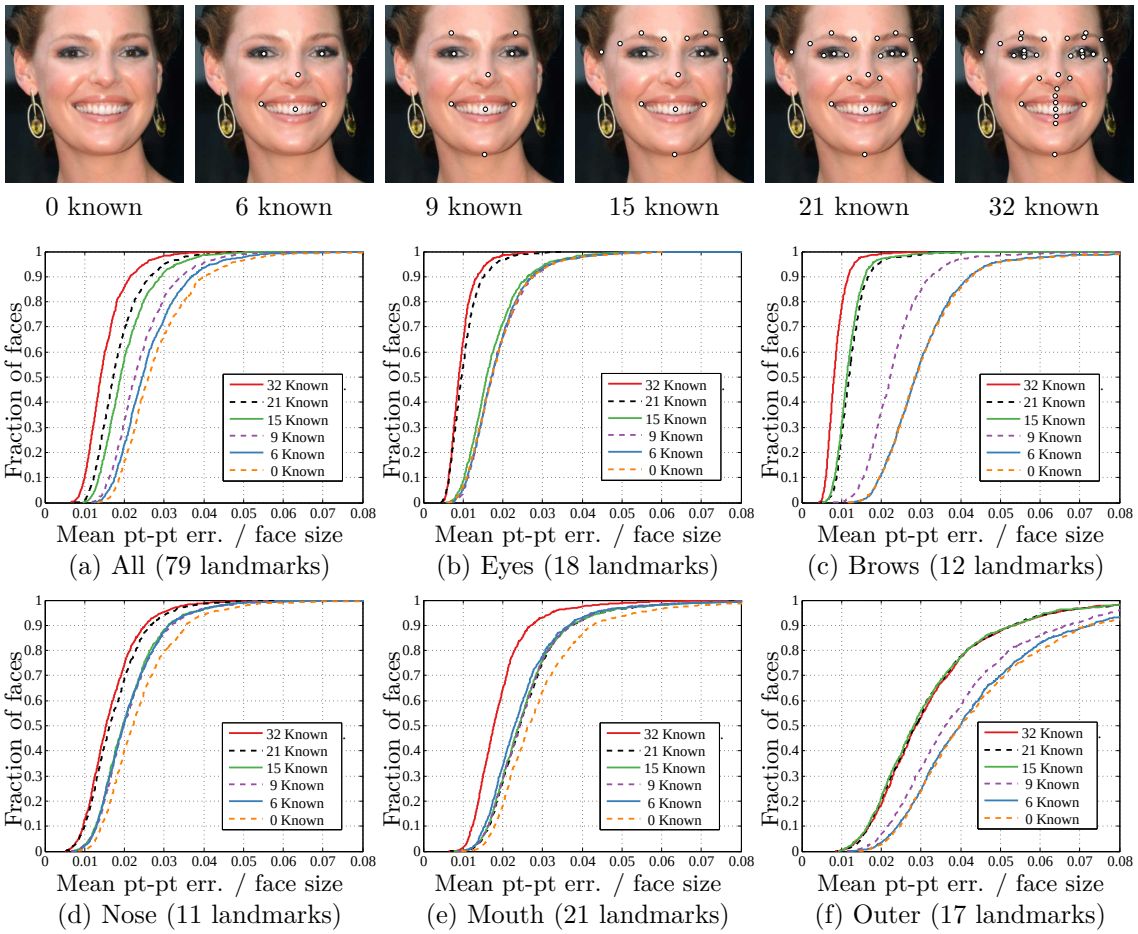


Figure 4.5: Quantitative evaluation of our full pipeline with six different trials, each assuming a different number of known landmarks. The top row shows the arrangement of known landmarks for each trial. "6 known" corresponds to the original AFW layout; "21 known" corresponds to AFLW; and "32 known" closely resembles the original LFPW annotations. In (a) we see the overall mean accuracy is high with 0 known landmarks (96.5% at 0.05), and the accuracy continues to improve significantly as additional landmarks become known. For reference, Belhumeur *et al.* [7] showed that their algorithm surpasses the average accuracy of human labelers on most landmarks, and our algorithm further improves Belhumeur *et al.*'s localization accuracy (see Figure 4.3) even with 0 known landmarks. We note that inherent ambiguities exist on the face, especially on longer contours such as the lips and the outer face contour. For example, a landmark estimate on the outer contour may be qualitatively correct, but in disagreement with "ground truth" in terms of its location along the contour. This phenomenon partly explains the lower CED curves in (c), (e), and (f). In general, we see that our approach correctly estimates landmarks on a large majority of faces, especially with 21 or 32 known landmarks. This suggests that our approach is well-suited for automatically supplementing the landmarks in large, sparsely annotated datasets like AFLW. **Best viewed in color.**

Chapter 5

Face Image Parsing with Soft Segments

5.1 Introduction

In face image analysis, one common task is to parse an input face image into facial parts, *e.g.*, left eye and upper lip. Most previous methods accomplish this task by marking a few landmarks [7, 142] or a few contours [46, 104] on the input face image. In this chapter, we seek to mark each pixel on the face with its semantic part label; that is, our algorithm parses a face image into its constituent facial parts.

Compared to segment-based representations, we argue that landmark- and contour-based representations have several key limitations.

- Other than eye corners and mouth corners, most landmarks are *not* well-defined. For example, it is unclear how many landmarks should be defined on the chin line, or how noses should be represented: should there be a line segment along the nose ridge, or a contour around the nostrils? Due to the lack of agreement, different datasets have different contour models. This creates difficulty for practitioners interested in unifying different datasets for robust algorithm development.
- Contour-based representations are not general enough to model several facial parts useful for robust face analysis. For example, teeth are important cues for analyzing open-mouth expressions; ears are important cues for analyzing profile faces; strands of hair are often confused by algorithms as occluders. It would be

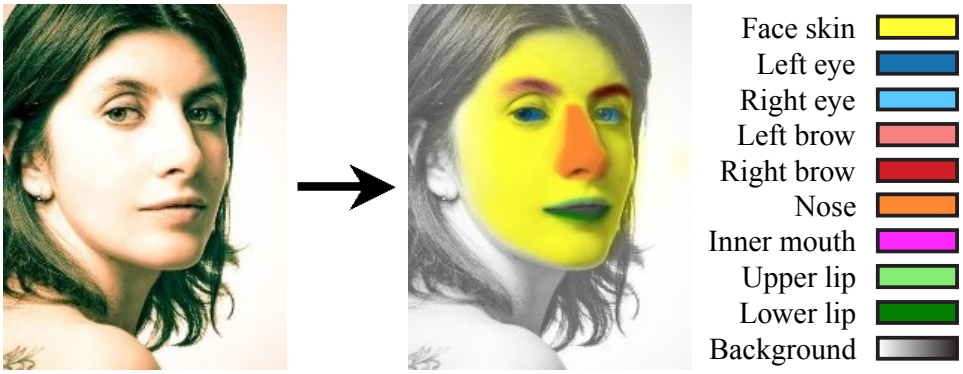


Figure 5.1: Our exemplar-based algorithm parses a face image into its constituent facial parts using a soft segmentation.

difficult to model these important parts using contours, and it is unclear how many landmark points should be used.

- It is difficult to encode uncertainty in contour-based representations. For example, the precise location of the tip of an eyebrow or the contour of a nose ridge are difficult to determine, even for human labelers. Such uncertainty leads to errors in human labeled face data that are used in both the training and evaluation of algorithms.

Segment-based representations alleviate the aforementioned limitations: segments can represent any facial part, be they hair or teeth, and soft segmentation can model uncertain transitions between parts. Although semantic segmentation for general scenes has received tremendous attention in recent years [61, 63, 80, 111], there has been relatively little attention given specifically to face part segmentation, with the exception of [86, 120]. Since facial parts have special geometric configurations compared to general indoor and outdoor scenes, we propose an exemplar-based face image segmentation algorithm, taking inspiration from previous work in image parsing for general scenes.

Specifically, our approach assumes a database of face images, each of which is associated with a hand-labeled segmentation map and a set of sparse keypoint descriptors. We emphasize that these keypoints need not correspond between different

images in the database; we extract keypoints and their descriptors independently from each image using SIFT [84]. Given a test image, our algorithm first employs Belhumeur *et al.* [7] to select m top exemplar images from the database as input. Our algorithm then computes a nonrigid warp for each top exemplar; each nonrigid warp aligns the exemplar image to the test image by matching the set of sparse precomputed exemplar keypoints to the test image. Finally, we propagate labels from the exemplar images to the test image in a pixel-wise manner, using trained weights that modulate and combine label maps differently for each part type.

We evaluate our method on two challenging datasets [51, 74] and compare with two face parsing algorithms [86, 120] and a general scene parsing algorithm [80]. We also compare our segmentation results with contour-based face alignment results: that is, we first run the alignment algorithms [46, 104, 142] to extract contour points and then derive segments from the contours. Our algorithm compares favorably with these previous works on all datasets tested. As a byproduct, our algorithm can recover contours of facial parts as well, although we do not focus on this representation. In summary, this chapter makes the following contributions.

1. *A novel algorithm for robustly segmenting face parts.* We recover a soft segmentation, which naturally encodes the segment class uncertainty in the image. Rather than artificially placing a hard boundary between, *e.g.*, skin and eyebrow regions, we recover label probabilities at each pixel. Our algorithm is exemplar-based; exemplars can also encode continuous, probabilistic segment labels.

2. *A learning algorithm for finding optimal parameters for calibrating exemplar label types.* Some part labels occur more frequently than others (*e.g.*, "skin" labels occur more frequently than "eye" labels), and tend to dominate adjacent labels that occur less frequently. To correct these biases, we train a set of label weights that adjust the relative importance of each label type.

3. *A rich training dataset for face segmentation.* Our dataset is built as an extension of the recent Helen Facial Feature Dataset [74]. The images are high resolution, and our dataset features segments that are created from densely-sampled, hand-labeled contours. Hair mattes are also included for future work in hair segmentation.

5.2 Related Work

We are inspired by the recent work of Luo *et al.* [86], who proposed a hierarchical technique for face parsing. In their work, they recover pixel-wise labels for eyes, eyebrows, nose, mouth, and background, which includes the skin. Their approach can be divided into three stages:

1. detect face parts (*e.g.*, upper face, lower face),
2. detect face components (*e.g.*, mouth, eyes), and
3. use component-specific segmentors on each component to estimate pixel-wise labels.

There are two aspects of this approach that can be improved. First, because they operate at the lowest level, the component-specific segmentors do not generalize well to labeling larger and/or less distinct regions of the face, such as the cheeks or the chin. Second, because they produce only a binary classification, the component-specific segmentors do not generalize well to more complicated label interactions, such as those that exist between the inner mouth region, the lips, and the skin around the lips, for example. Instead, we propose a nonparametric approach that naturally extends to these difficult cases.

Previously, Warrell and Prince [120] introduced *LabelFaces* based on a scene parsing approach applied to faces. Warrell and Prince argued that the scene parsing approach is advantageous because it is general enough to handle unconstrained face images, where the shape and appearance of features vary widely and relatively rare semantic label classes exist, such as mustaches and hats. As part of their contribution, they introduced priors to loosely model the topological structure of face images (so that mouth labels do not appear in the forehead, for example). However, the labels they generate are often coarse and inaccurate, especially for small face components like eyes and eyebrows. We show in this work that our approach produces accurate, fine-scale label estimates in unconstrained face images.

There are several recent scene parsing approaches in the literature that do not target faces, but nonetheless do share some aspects with our approach ([61, 63, 80, 111] to name just a few). A full review is outside the scope of this chapter. Of these

approaches, Liu *et al.* [80] is particularly relevant. Like our approach, they propose a nonparametric system that transfers labels from exemplars in a database to annotate a test image. At its core, Liu *et al.*'s system relies on the SIFT Flow [81] algorithm to densely transfer labels from exemplars to the test image. Unfortunately, SIFT Flow is slow, even with a coarse-to-fine strategy. For example, [81] reported a runtime of 31 seconds for 256×256-pixel image pairs with a C++ implementation running on modern hardware.

Targeting face image matching, we employ an efficient sparse matching approach that does not require global optimization, and therefore requires much less computation for each image pair. This savings allows us to use a large set of top exemplar images for label transfer (in [80] they use $m \leq 9$ top exemplar images; we use $m = 100$), which is important in our approach for two reasons. First, by aggregating label votes from many exemplars, our approach is robust to outliers and noise. Second, it partially explains why we can avoid global optimization of the flow field: by using a large number of top exemplars, the sparse keypoint matches cover the test image well and good matches occur almost everywhere.

Furthermore, our algorithm produces a soft segmentation while [80] produces a hard segmentation; specifically, our algorithm assigns each pixel a probability value for each label type. To this end, we propose a training algorithm for estimating a set of weights that convert label maps from exemplars to label probabilities on the test image. We remark that soft segmentation is useful for future work on hair segmentation, among other applications.

5.3 Approach

In this section we first give an overview of our approach and then present technical details of each step. Figure 5.2 includes a supplementary visual synopsis of our algorithm.

We start with some notation. Let $\mathbf{p}_i$ be a probability vector for pixel i:

$$\begin{aligned} \mathbf{p}_i &= [p_{i,1}, p_{i,2}, \ldots, p_{i,K}]^\mathsf{T} \\ &\text{s.t.} \quad \textstyle\sum_k^K p_{i,k} = 1, \qquad 0 \leq p_{i,k} \leq 1, \end{aligned} \tag{5.1}$$

where $p_{i,k}$ is the probability that pixel i belongs to segment class k. Each $\mathbf{p}_i$ encodes label uncertainty at the pixel level, which reflects the natural indistinctness of some facial features (*e.g.*, light eyebrows, chin line). We seek to estimate $\mathbf{p}_i$ for all pixels $i = 1, 2, \ldots, N$ in the test image I.

5.3.1 Overview

Database Construction Our database $\mathcal{M}$ is composed of a set of exemplars $\{M_j\}_{j=1}^J$. Each exemplar M_j has four parts: an image, a label map, a very sparse set of facial landmark points, and a sparse set of SIFT [84] keypoint descriptors. Each label map can either be soft (*i.e.*, so that each $\mathbf{p}_i$ has several non-zero components), or hard (*i.e.*, so that each $\mathbf{p}_i$ has exactly one nonzero component). We use 12 landmark points: 2 mouth corners, 4 eye corners, 2 points on the eyebrows (each centered on the top edge), 2 points on the mouth (one on the top edge of the upper lip and one on the bottom edge of the bottom lip), 1 point between the nostrils, and 1 chin point. About 150 SIFT keypoints are automatically extracted from each image independently in the database. We make a distinction here between *landmarks*, which are defined consistently across images (*e.g.*, the mouth corners), and *keypoints*, which are not necessarily consistent across images. Each SIFT keypoint descriptor is computed using a window radius of approximately 1/4 the inter-ocular distance (IOD).

Runtime Preprocessing Given a test image, we first use a face detector (*i.e.*, [118]) to roughly locate the face and estimate its scale. The test image is then rescaled so that the face has an IOD of approximately 55 pixels, which is the size of the

exemplar faces. Second, *dense* SIFT descriptors [84] are extracted using the same window size (1/4 IOD) across all pixels. To search for a subset of m top exemplar faces in the database, we use Belhumeur *et al.* [7] on our 12 landmark points. The output of the preprocessing is a set of m exemplars, each of which is associated with a similarity transformation that aligns the exemplar to the face in the test image.

Step 1: Nonrigid exemplar alignment For each keypoint in each of the top m exemplars, search within a small window in the test image to find the best match; record the matching score and the location offset of the best match for each keypoint. Warp the label map of each top exemplar nonrigidly using a displacement field interpolated from the location offsets.

Step 2: Exemplar label map aggregation Aggregate warped label maps using weights derived from the keypoint matching scores in Step 1. The weights are spatially varying among exemplar pixel locations and favor exemplar pixels near keypoints that are matched well with the test image.

Step 3: Pixel-wise label selection Produce a label probability vector at each pixel by first attenuating each channel in the aggregated label map and then normalizing it. The attenuating weights are trained offline in order to correct for label population biases and maximize labeling accuracy. Hard segmentation can be generated by selecting the highest probability label channel.

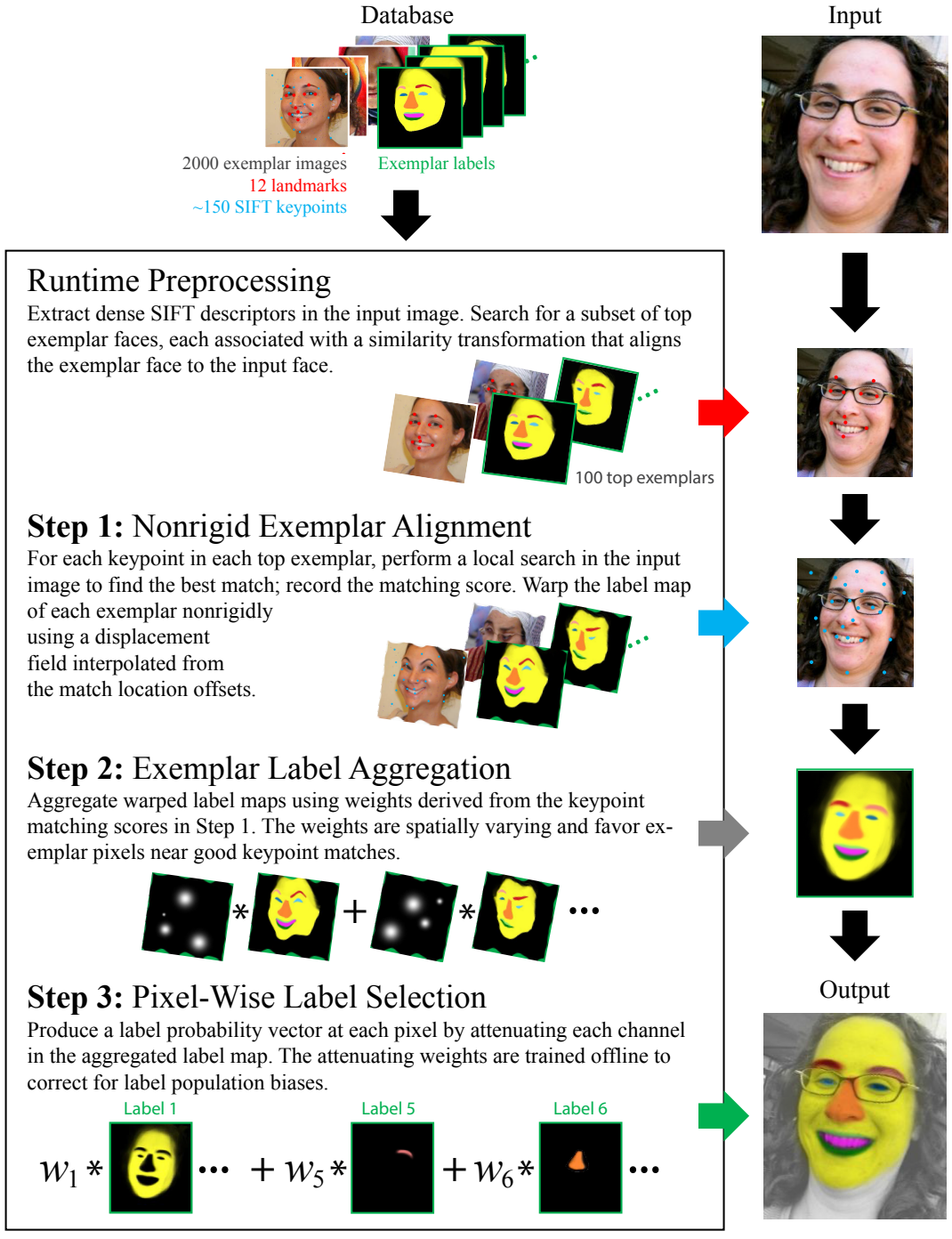


Figure 5.2: Overview of our semantic face image segmentation pipeline.

5.3.2 Step 1: Nonrigid Exemplar Alignment

Due to local deformation, a similarity transformation is not sufficient to align an exemplar with the testing image. The goal of Step 1 is to refine the registration using a nonrigid warp between each top exemplar label map and the test image. Dense per-pixel correspondence algorithms like SIFT Flow [81] are one strategy for this purpose. However, per-pixel correspondence algorithms often perform global optimization, which is computationally intensive and does not scale well to the many exemplars we use for our task. In particular, we would like to aggregate a large number of top exemplars (we use 100) in order to be robust against outliers in one or a small number of exemplars. Therefore, for efficiency reasons, we instead rely on about 150 SIFT keypoints to compute the nonrigid warp between each exemplar and the test image.

Given the similarity transformation estimated in the preprocessing step, for each keypoint in one top exemplar, its true correspondence in the test image is usually within a small window centered at the location predicted by the similarity transformation. Therefore, we adopt a local search within the window to find its best match. To make the search robust to untextured regions, we encourage the best match to be close to the window center. Specifically, for each keypoint f with SIFT descriptor $\mathbf{s}_f$, we search for the location offset $\Delta\mathbf{x}_f$ that produces the best match in the window by using the following objective function:

$$r(\Delta\mathbf{x}_f) = g_{\mathsf{spatial}}(\tfrac{\Delta\mathbf{x}_f}{\sigma_{\mathsf{spatial}}}) \cdot g_{\mathsf{desc}}(\tfrac{\mathbf{s}(\Delta\mathbf{x}_f)-\mathbf{s}_f}{\sigma_{\mathsf{desc}}}), \tag{5.2}$$

where g_{spatial} and g_{desc} are Gaussian functions and $\mathbf{s}(\Delta\mathbf{x}_f)$ is the SIFT descriptor at the offset location. In our implementation, we set $\sigma_{\mathsf{spatial}} = 10$ to be the same as the search window radius and $\sigma_{\mathsf{desc}} = a \cdot b$, where a is the length of the SIFT descriptor and b is the scale of the descriptor elements.

Our algorithm computes a nonrigid warp for each exemplar label map by interpolating the displacements $\{\Delta\mathbf{x}_f\}_{f=1}^{F}$, where F is the number of SIFT keypoints in the exemplar. The interpolation is implemented using a linear combination of Gaussian Radial Basis Functions (RBF) [97] centered at each keypoint, where each RBF bandwidth is proportional to the distance to the nearest neighboring keypoint.

5.3.3 Step 2: Exemplar Aggregation

For each exemplar label map, we interpolate the matching scores $r(\Delta\mathbf{x}_f)$ in Eq. (5.2) between its keypoints to generate a matching score for each pixel; the interpolation uses the same Gaussian RBFs as we use at the end of Step 1 to generate the nonrigid warping field. Note that $r(\Delta\mathbf{x}_f) \in [0, 1]$, where higher values suggest better matches. Now, each nonrigidly warped exemplar label map is associated with a per-pixel matching score map. We aggregate these label maps by taking a weighted sum as follows:

$$\mathbf{p}_i \propto \sum_j^m r_{i,j} \cdot \mathbf{p}_{i,j}^{\mathsf{exemplar}}, \tag{5.3}$$

where $\mathbf{p}_{i,j}^{\mathsf{exemplar}}$ is the label probability vector at pixel i in exemplar j, and $r_{i,j}$ is the corresponding matching score.

5.3.4 Step 3: Pixel-wise Label Selection

The results from the previous stage are imperfect. Near smaller regions, like the eyes, eyebrows, and lips, we observe that, if the aggregated label probabilities are incorrect, they tend to be incorrect in the direction of the larger surrounding regions, namely the face skin and background. Consider, for example, a region near the eyebrow edge. Assuming noise prevents perfect correspondences, "skin" label correspondences will occur more frequently than "eyebrow" label correspondence simply because there are many more skin labels than eyebrow labels. A common symptom of this label bias is that estimated eyebrow regions (and other small regions of the face) tend to be too small.

We compensate for this bias by re-weighting each component of the aggregated label probability vector, and then re-normalizing each pixel's label probability vector afterward. Given a tuning set with ground truth label probabilities, we find label component weights $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_K]$ by minimizing the following function Θ:

$$\begin{aligned} \Theta(\boldsymbol{\alpha}) \;&=\; \sum_k \theta_k(\boldsymbol{\alpha}) + \lambda \sum_{k'>k} \left\| \theta_k(\boldsymbol{\alpha}) - \theta_{k'}(\boldsymbol{\alpha}) \right\|^2 \\ &\text{s.t.} \quad \textstyle\sum_k^K \alpha_k = 1, \qquad \forall k: \;\; \alpha_k > 0, \end{aligned} \tag{5.4}$$

where

$$\theta_k(\boldsymbol{\alpha}) = \frac{1}{C_k} \sum_j^J \sum_{i \in \phi_{j,k}} \left\| p_{i,j,k}^{\mathtt{gt}} - \frac{\alpha_k p_{i,j,k}}{\sum_k \alpha_k p_{i,j,k}} \right\|^2, \tag{5.5}$$

$p_{i,j,k}$ and $p_{i,j,k}^{\mathtt{gt}}$ are estimated and the ground truth label probabilities for image j, pixel i, and class k, respectively; $\phi_{j,k}$ is the set of pixels in image j for which $p_{i,j,k}^{\mathtt{gt}} > 0$; C_k is a normalization parameter; and λ is a scalar regularization parameter.

Minimizing Eq. (5.4) is complicated by the normalization in $\theta_k(\boldsymbol{\alpha})$, which makes it nonlinear, and the sum over images and pixels, which makes it large. However, Eq. (5.4) only needs to be minimized offline once. In our implementation, we minimize Eq. (5.4) using MATLAB's fmincon function, which uses the interior point method for large-scale constrained nonlinear optimization problems [17].

We can use Eq. (5.4) to find optimal weights to maximize different evaluation metrics. For example, setting C_k to the total number of pixels in all images with label k according to ground truth and $\lambda = 0.5$, we can maximize accuracy with respect to a

confusion matrix; setting $C_k = 1$ and $\lambda = 0$, we can maximize accuracy with respect to F-measures. We will discuss the pros and cons of different settings in Section 5.4.3.

After the label component weights have been found, we adjust each label probability vector. Optionally, we can select the component with the largest probability value to generate hard segmentation results. A hard segmentation is used in our quantitative experiments for accuracy evaluation.

5.4 Results and Discussion

We have evaluated our method on two different datasets, and we show that it clearly improves upon a recent general scene parsing approach and existing face parsing approaches. Additionally, we adapt a recent landmark localization method and two face alignment algorithms to produce segmentation results, and show that our method is more accurate.

5.4.1 Experimental Datasets

Our first experimental dataset is LFW [51]. Luo *et al.* [86] showed segmentation results on 300 randomly selected images from LFW. To compare with their results, we use the same subset of images in our experiments. Following their procedure to evaluate accuracy, we generated ground truth by annotating each face with contour points around each segment.

Our second (primary) dataset is Helen [74], which is composed of 2330 face images with densely-sampled, manually-annotated contours around the eyes, eyebrows, nose, outer lips, inner lips, and jawline. We use Helen because it features high-quality, real-world photographs of people with a more balanced proportion of genders, ages, and ethnic backgrounds than other face datasets. We separated Helen into three parts for our experiments: exemplar, tuning, and test sets. Our exemplar set was used for all experiments, including experiments on LFW images. Our Helen tuning and test sets were formed by taking the first 330 images in the dataset; they include no subjects from the exemplar set. Our Helen test set is composed of 100 randomly selected images from the first 330, and our tuning set comes from the remaining images.

We generated ground truth eye, eyebrow, nose, inside mouth, upper lip, and lower lip segments automatically by using the manually-annotated contours as segment boundaries. For face skin, we used the jawline contour as the lower boundary; for the upper boundary, we separated the forehead from the hair by manually annotating forehead and hair scribbles and running an automatic matting algorithm [77] on each image. Although we do not focus on hair segmentation in this work, we also recovered "ground truth" hair regions using this approach. The hair mattes from [77] are usually accurate, but mistakes are inevitable. Therefore, to ensure fair accuracy measurements, we manually annotated the face skin in all test images.

5.4.2 Accuracy vs. Number of Top Exemplars

The number of top exemplars m effects labeling accuracy, the quality of the soft segmentation, and runtime. Up to a point, greater m results in more accurate hard segments and higher quality transitions between soft segments, but at the cost of runtime, which is linear in m. Figure 5.3 shows a plot of the F-measure[1] versus m. We see that accuracy plateaus near $m = 50$.

5.4.3 Comparisons on LFW

We first show results on LFW, and compare with two recent face parsing techniques [86, 120], two recent face alignment algorithms [46, 104] and a face landmark localization algorithm [142], which we have adapted for face parsing; that is, we derived segmentation results from the estimated contour points using the same approach used to generate ground truth segments in Helen. Warrell and Prince [120] also showed results on LFW, but their 150-image test subset and source code are no longer available. We therefore simply report their numbers, and acknowledge that their results were computed on a different (but qualitatively similar) set of images.

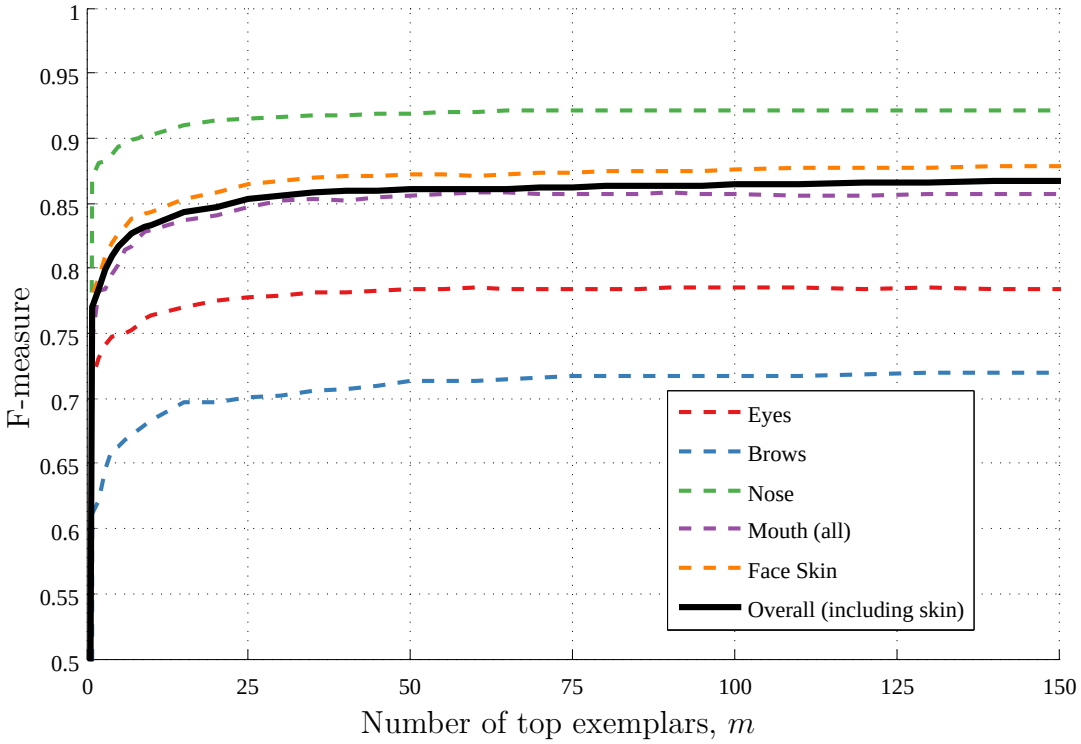
Luo *et al.* [86] presented the accuracy of their segmentors using a confusion matrix. We repeated their experiment using our method. First, we computed results on our Helen tuning set, and used our continuous results to train the label weights according to Section 5.3.4 (we matched the LFW segment representation by grouping the Helen mouth components, and treating face skin as background). Table 5.1 shows a comparison of our results using our trained label weights, and the results reported in [86].

Using the confusion matrix for comparison, our results look much more accurate than [86]. However, Figure 5.4 shows a result that exemplifies a problem with the label weights found by minimizing Eq. (5.4). Effectively, if we use $C_k = \sum_j |\phi_{j,k}|$ and $\lambda = 0.5$, Eq. (5.4) finds label weights that maximize the recall rates of eye, eyebrow, nose, and mouth pixels, which are relatively few and sensitive to errors, by sacrificing the recall rate of background pixels, which are numerous and insensitive to errors. The confusion matrix in Table 5.1 (b) reflects maximized recall rates, but does not give any indication of the problem exemplified in Figure 5.4 (a). We therefore instead

[1] The F-measure is the harmonic mean of precision and recall: $F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$.

$m = 1$ $m = 2$ $m = 5$ $m = 25$ $m = 50$

(a) Results generated using different numbers of top exemplars



(b) Pixel-wise label accuracy vs. the number of top exemplars

Figure 5.3: Face segmentation accuracy vs. the number of top exemplars m used by our algorithm. (a) Qualitative results with different m. (b) Pixel-wise label accuracy, as measured by the F-measure, which is the harmonic mean of precision and recall. We see that accuracy increases with greater m until approximately $m = 25$, where it begins to plateau. The accuracy does not improve significantly beyond $m = 50$.

	left eye	right eye	nose	left brow	mouth	right brow	background
left eye	**.90**			.01			.09
right eye		**.93**				.01	.06
nose			**.88**		.01		.11
left brow	.03			**.91**			.06
mouth					**.90**		.10
right brow		.02				**.89**	.09
background			.01		.04		**.95**

(a) Results from [86]

	left eye	right eye	nose	left brow	mouth	right brow	background
left eye	**.990**			.003			.007
right eye		**.990**				.002	.008
nose			**.992**	.001		.001	.006
left brow	.002			**.988**			.010
mouth			.001		**.983**		.016
right brow		.003				**.982**	.015
background	.002	.002	.004	.006	.005	.006	**.975**

(b) Our results

Table 5.1: Luo *et al.* [86] presented the accuracy of their segmentors using the confusion matrix shown in (a). We repeated their experiment using our method; our results are shown in (b) for comparison. Based on the confusion matrix, our results look much more accurate. However, this metric can be deceiving, as discussed in Section 5.4.3.

(a) Estimated (b) Ground truth

Figure 5.4: The result on the left exemplifies the problem with the label weights found by minimizing Eq. (5.4) using $C_k = \sum_j |\phi_{j,k}|$ and $\lambda = 0.5$: the eye, eyebrow, nose, and mouth regions are too dilated compared to the ground truth shown on the right despite maximizing the diagonal of the confusion matrix in Table 5.1 (b). Based on this problem with the confusion matrix, we instead show accuracy using the F-measure, and set $C_k = 1$ and $\lambda = 0$ in Eq. (5.4).

F-Measures for LFW Images					
Method	Eyes	Brows	Nose	Mouth	Overall
Warrell & Prince [120]	0.443	0.273	0.733	0.653	n/a
Zhu & Ramanan [142]	0.520	n/a	n/a	0.635	n/a
Saragih *et al.* [104]	0.684	0.651	0.903	0.753	0.793
Gu & Kanade [46]	0.735	0.722	0.900	0.801	0.820
Ours	**0.765**	**0.752**	**0.914**	**0.881**	**0.863**

Table 5.2: The top row is copied from [120]. Zhu & Ramanan [142] is a landmark localization method, Saragih *et al.* [104], and Gu & Kanade [46] are face alignment methods. Segments were derived from the contours generated by these methods. We used the model provided with Zhu & Ramanan's implementation, which was trained on the Multi-PIE face database [45]. We used our implementations of Saragih *et al.* [104] and Gu & Kanade [46], both trained on Helen. The "overall" values are computed using all eye, eyebrow, nose, and mouth pixels. We see that our algorithm compares favorably to all previous works on LFW.

Label Weights			
Face skin	0.0765	Nose	0.1000
Left eye	0.0925	Inner mouth	0.2132
Right eye	0.0925	Upper lip	0.1114
Left brow	0.0615	Lower lip	0.1067
Right brow	0.0615	Background	0.0841

Table 5.3: Label weights computed using Eq. (5.4) with $C_k = 1$ and $\lambda = 0$.

show accuracy in Table 5.2 using the F-measure, which is the harmonic mean of both recall *and* precision. We find that setting $C_k = 1$ and $\lambda = 0$ in Eq. (5.4) results in a set of label weights that give good performance with respect to the F-measure, shown in Table 5.3.

We compare the accuracy of our method with several other face parsing and alignment methods [46, 104, 120, 142] in Table 5.2. Our algorithm compares favorably to these works on LFW.

5.4.4 Comparisons on Helen

We repeated the LFW experiment on the Helen test set. Table 5.4 shows a comparison of the accuracy from three variants of our algorithm, Liu *et al.* [80], which is a general

scene parsing method, and the adapted output from Zhu & Ramanan [142], Saragih *et al.* [104], and Gu and Kanade [46], all ordered by overall F-measure.

A large disparity in accuracy can be seen between the results from Zhu & Ramanan [142] and the other methods. We conjecture that this is due in part to the fact Zhu & Ramanan's provided model was trained on Multi-PIE, whereas the other methods were trained on Helen, which includes a much richer set of landmarks and faces. Regardless, we see that our approach improves upon the segments generated by recent face alignment algorithms.

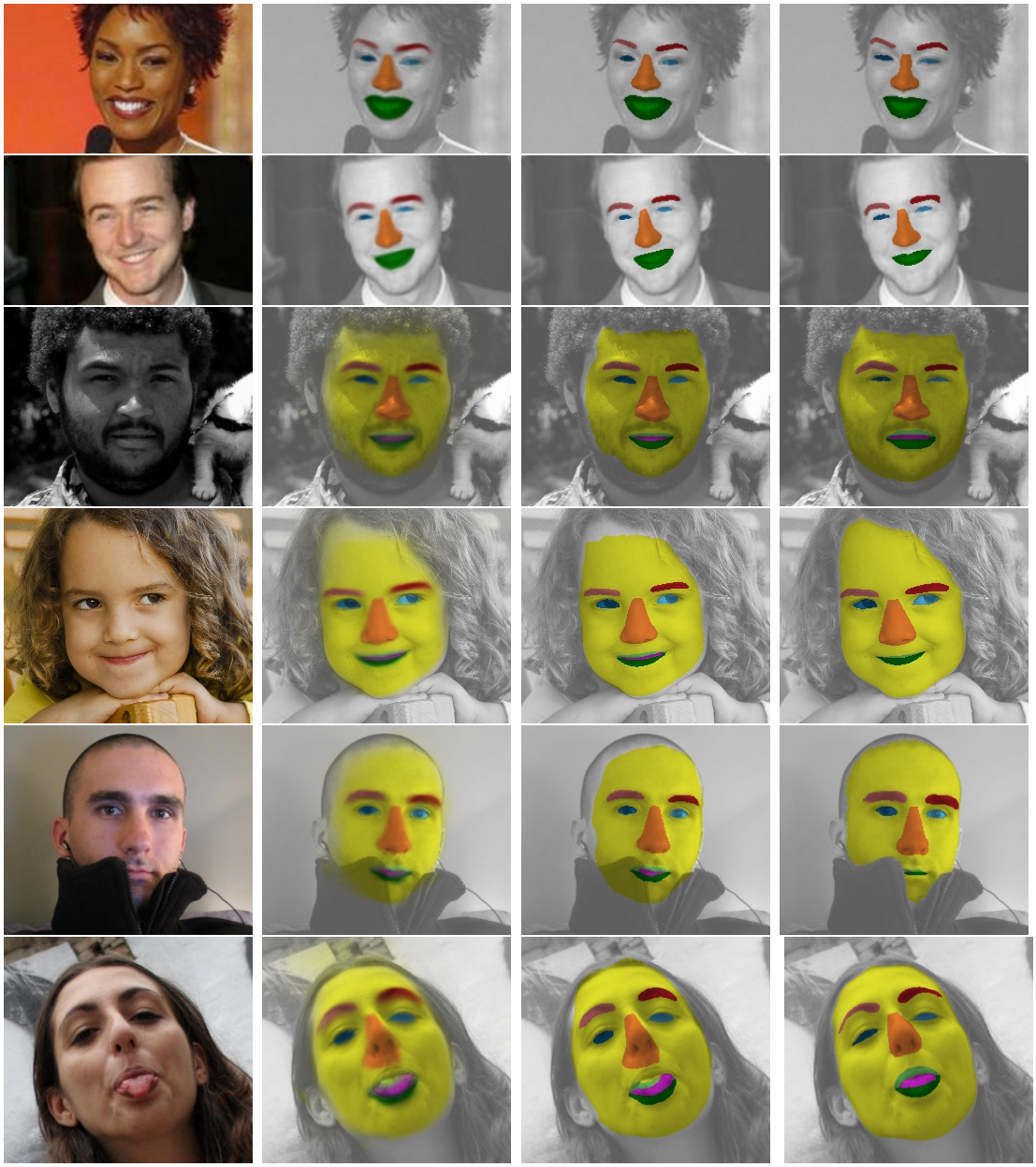
We used Liu *et al.*'s code trained on Helen to generate the values in Table 5.4. Liu *et al.*'s algorithm requires four parameters. We set $\alpha = 0.06$ (spatial prior weight) and $\beta = 1$ (smoothness weight) by performing a parameter sweep and selecting the values that maximized the overall F-measure. In [80] they suggest using $K = 85$ nearest neighbors and $M = 9$ voting candidates. However, for a fairer comparison, we set $K = 100$ and $M = 100$ to match the number of m top exemplars used by our method. To ensure that this change of K and M did not artificially reduce the performance of their method, we verified that $K = M = 100$ produced more accurate results than $K = 85$ and $M = 9$. We see in Table 5.4 that our algorithm is much more accurate than Liu *et al.* in general. Qualitatively, the segments generated by Liu *et al.* are also less accurate; please see our supplementary material for a visual comparison.

By comparing the fifth and sixth rows of Table 5.4, we observe that the local search and nonrigid exemplar alignment from Step 1 of our algorithm modestly improves the quantitative accuracy of our results. However, the improvement from Step 1 is mostly visible in our continuous, probabilistic results, which we cannot adequately judge quantitatively (*i.e.*, to compute the F-measures, we must first quantize our results).

We see a noticeable improvement from row six to row seven in Table 5.4, especially in the inner mouth region, due to the label weights. In our view, the mouth is the most challenging region of the face to segment. The shape and appearance of lips vary widely between subjects, mouths deform significantly, and the overall appearance of the mouth region changes depending on whether the inside of the mouth is visible or not. Unusual mouth expressions, like the one shown in the bottom row of Figure 5.5 and in Appendix C, Figure C.2 are not represented well in the exemplar images, which results in poor label transfer from the top exemplars to the test image. Despite these challenges, our algorithm generally performs well on the mouth, with large

F-Measures for Helen Images									
Method	Eyes	Brows	Nose	In Mouth	Top Lip	Bot. Lip	Mouth(all)	Skin	Overall
Zhu & Ramanan [142]	0.533	n/a	n/a	0.425	0.472	0.455	0.687	n/a	n/a
Saragih *et al.* [104]	0.679	0.598	0.890	0.600	0.579	0.579	0.769	n/a	0.733
Liu *et al.* [80]	0.770	0.640	0.843	0.601	0.650	0.618	0.742	**0.886**	0.738
Gu & Kanade [46]	0.743	0.681	0.889	0.545	0.568	0.599	0.789	n/a	0.746
Ours, Steps 1, 3 omitted	0.766	0.687	0.896	0.678	0.637	**0.703**	0.853	0.861	0.779
Ours, Step 3 omitted	0.772	0.708	0.914	0.659	0.639	0.697	0.850	0.872	0.790
Ours, full pipeline	**0.785**	**0.722**	**0.922**	**0.713**	**0.651**	0.700	**0.857**	0.882	**0.804**

Table 5.4: Zhu & Ramanan [142], Liu *et al.* [80], Saragih *et al.* [104], and Gu & Kanade [46] were trained as described in Section 5.4.3. In this case, the "overall" measure is computed over eye, eyebrow, nose, inner mouth, upper lip, and lower lip segments; face skin is excluded in the overall measure, as it cannot be computed for Zhu & Ramanan, Saragih *et al.*, or Gu & Kanade. The only area where Liu *et al.*'s system is more accurate than ours is on the face skin. The difference is minimal and is primarily due to our algorithm incorrectly "hallucinates" skin in hair regions, while Liu *et al.*'s system does not. In general, we see that our algorithm compares favorably to all previous works on this dataset, and our full pipeline performs best overall.



Original image | Our method, *soft* segmentation | Our method, *hard* segmentation | Ground truth

Figure 5.5: Selected results on LFW images (top two rows) and Helen images (bottom four rows). The inside of the mouth is not given as ground truth for the LFW images, and so we show only the entire mouth segment. We observe that our algorithm works well on grayscale images (third column) and images of young faces (fourth column). Our algorithm does not currently detect occlusions, and instead 'hallucinates' occluded parts of the face (fifth column). The last column shows a failure case due to the unusual expression. We note that our algorithm generally produces excellent results. **Best viewed in color.**

segmentation errors occurring infrequently. We show qualitative results in Figure 5.5 and in Appendix C, Figure C.1. Our improvement over other algorithms demonstrates the advantages of using segments to parse face parts. For example, the inside of the mouth is not well modeled using a classical contour based representation.

5.4.5 Comparisons with Liu *et al.* [80]

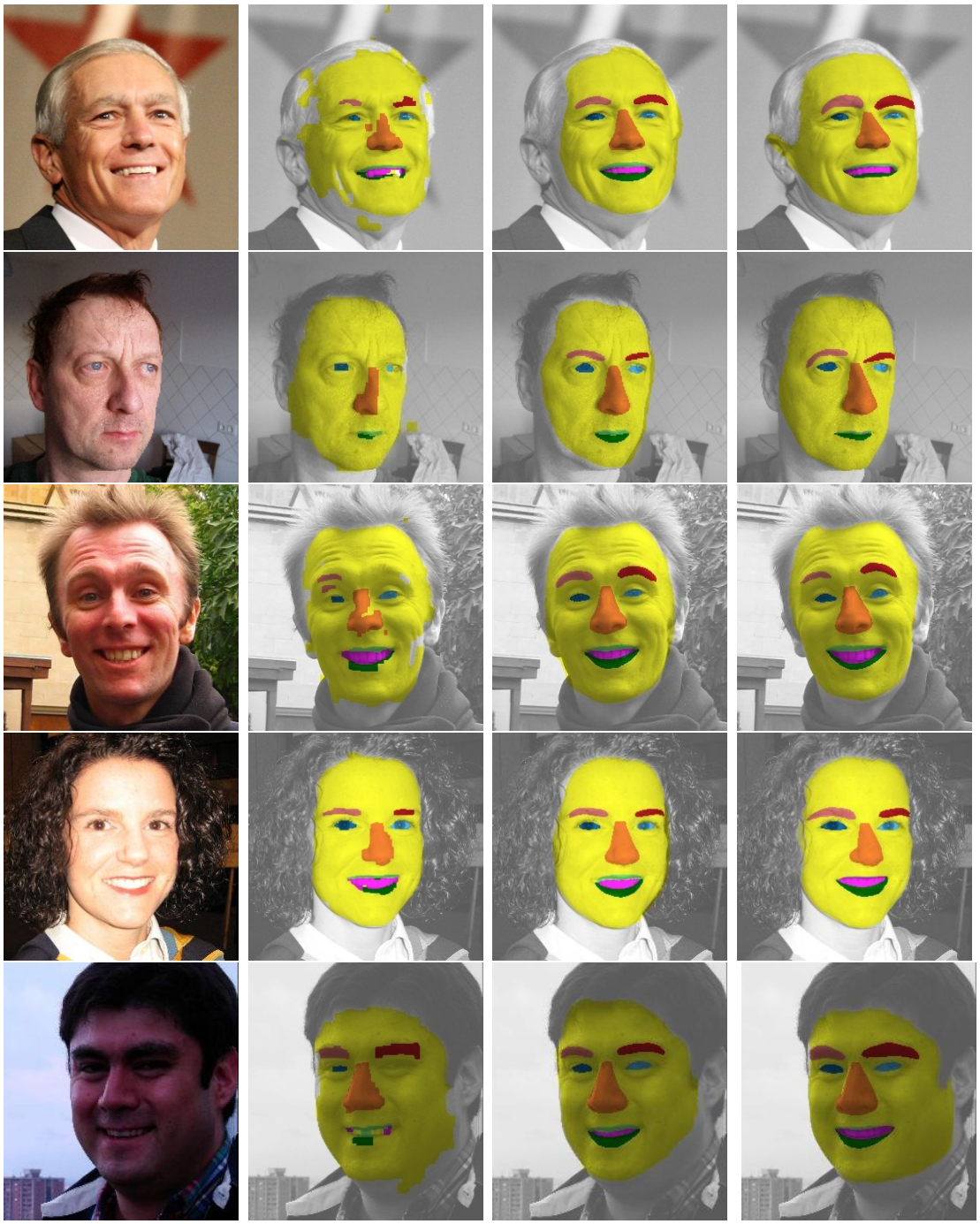
The scene parsing approach by Liu *et al.* [80] shares several similarities with our work. Like our approach, they propose a nonparametric system that transfers labels from exemplars in a database to annotate a test image. This begs the question, *Why not simply apply the approach from Liu* et al. *to face images?*

To help answer this question, we used the code provided by Liu *et al.* on our Helen [74] images; our exemplar set is used for training their system, and our test set is used for testing. Figure 5.6 shows several selected results for qualitative comparison. In general, our algorithm performs much better than Liu *et al.*'s algorithm.

5.4.6 Runtime

Our experimental implementation was written in MATLAB, which makes it difficult to judge the true runtime of our approach. However, we can roughly estimate it as follows. Belhumeur *et al.* [7] report a runtime of less than one second per fiducial and they note that most of the time is spent evaluating the local detectors. For 12 landmarks, their algorithm should take approximately 12 seconds. Our MATLAB implementation of their algorithm represents 56% of the total computation in our pipeline. We therefore estimate that the true runtime of our algorithm would be approximately 21 seconds with a C++ implementation. Furthermore, much of the computation is devoted to interpolation and image warping, which can be made very fast on the GPU.

To be more concrete, we can compare the actual runtime of our current implementation with that of Liu *et al.* [80]. Liu *et al.* use global optimization, which makes their approach very slow. On a workstation with two quad-core 3.00 GHz Intel Xeon CPUs and 32 GB memory, their implementation took an average of 18.5 minutes per test image with $M = K = 100$, whereas our implementation took an average of 1.1



Original image | Liu *et al.* [80] | Our method | Ground truth

Figure 5.6: Qualitative comparison with Liu *et al.* [80]. We see that the segments generated by Liu *et al.* are visibly less accurate, especially in the mouth region. This suggests that a general scene parsing approach is not well suited for faces. **Best viewed in color.**

minutes per test image with $m = 50$ after the runtime pre-processing step (*i.e.*, after Belhumeur *et al.*'s algorithm).

5.4.7 Extensions of Our Approach

Here we also show preliminary results for several extensions of our approach.

Contour Estimation Estimating contours is not the focus of this work. However, we can recover contours by treating contour points in the exemplars in almost the same way that we treat segment labels. That is, in Step 1, we warp the contour point from each exemplar in the same way that we warp the exemplar label maps. Then, in Step 2, we can aggregate the contour points using an approach similar to Belhumeur *et al.* [7]. Specifically, each contour point is found by computing the weighted average location of the warped exemplar contour points; each weight j is given by the match scores in R_j closest to the contour point. Figure 5.7 shows two selected results using this approach.

Hair Segmentation Several approaches for hair segmentation start by estimating a set of hair / not hair seed pixels in the image, and then refine the hair region using a matting algorithm ([96] is one example). We can also generate seeds by counting the votes from hair / not hair labels from the top exemplars, and thresholding the counts. Figure 5.8 shows seeds generating using this approach, and hair mattes computed from these seeds using [77].

Face Image Reconstruction and Synthesis Exemplar-based face image reconstruction/synthesis is applicable for various face image editing tasks, including grayscale image colorization [76] and automatic face image retouching [47]. We can create a synthetic version of the input face by propagating color and intensity information from the exemplar images to the input image; this can be easily accomplished by replacing the label vectors with the color (or intensity) channels of the exemplar images. Figure 5.9 shows two examples.

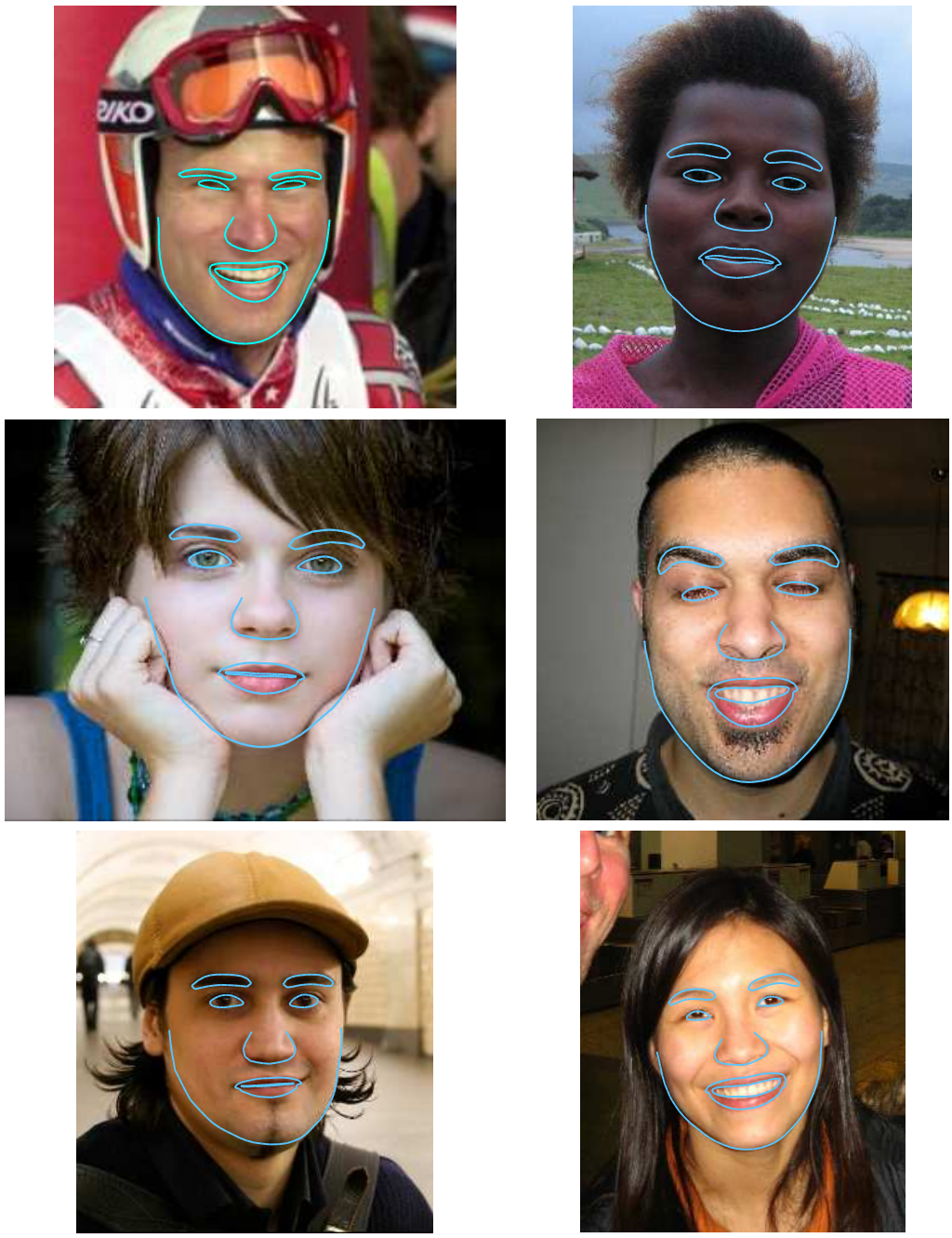


Figure 5.7: A simple extension of our approach is contour estimation. Please see Section 5.4.7 for details.

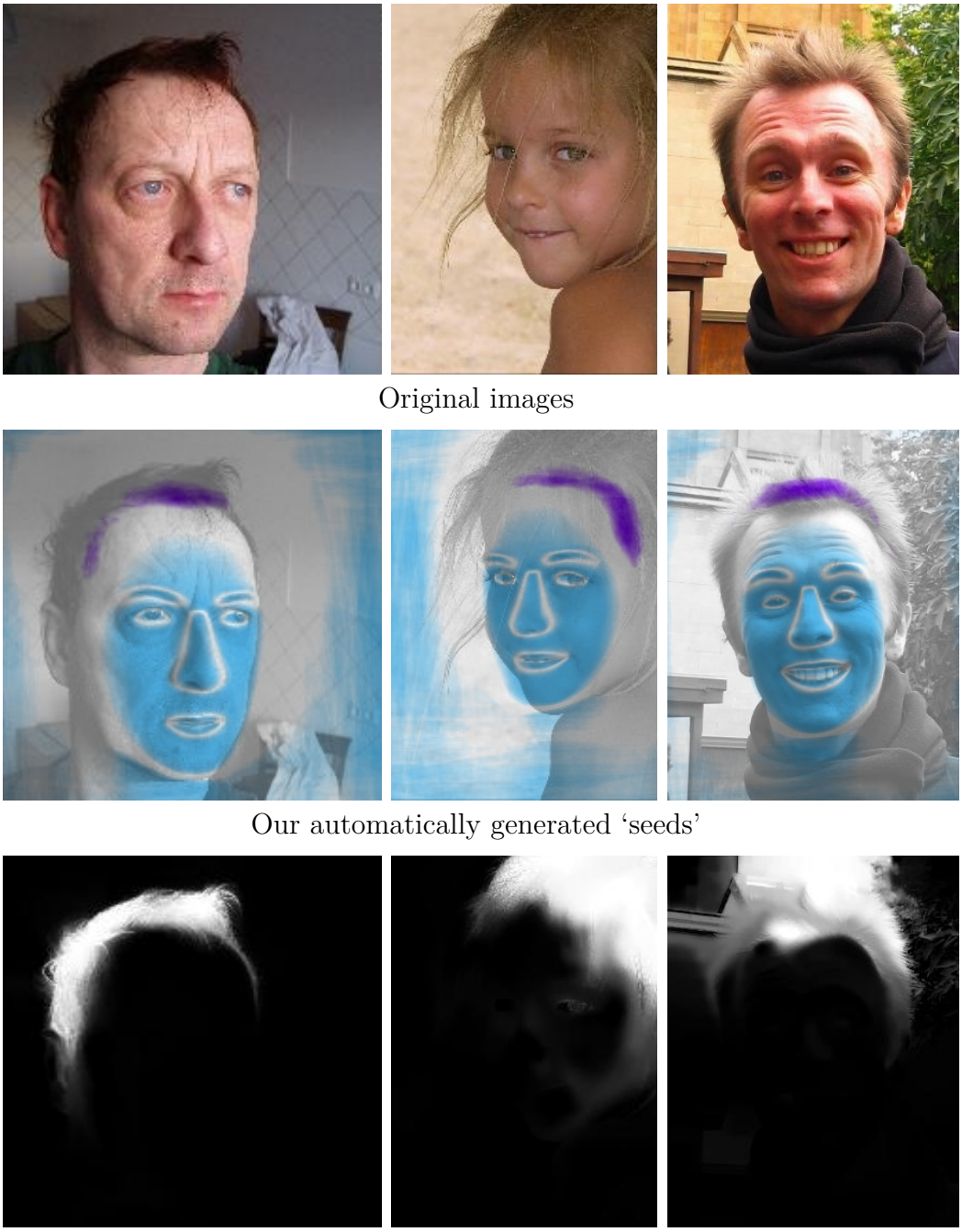


Figure 5.8: Preliminary results using an extension of our approach for hair segmentation. Top row: input. Middle row: our automatically generated ‘seeds’ for hair (purple) and background (blue). Bottom row: automatic matting results from [77]. We can recover accurate hair mattes in many cases (first two columns), but the procedure often fails on difficult cases (third column). **Best viewed in color.**

Original image Synthesized result

Figure 5.9: We can synthesize the input face by replacing the exemplar label vectors with the color channels from the exemplar images. See Section 5.4.7 for more details.

Chapter 6

Conclusions and Future Work

This dissertation has described three related and complementary algorithms with associated qualitative and quantitative results that strongly support the thesis that: *an exemplar-based approach that models face shape and appearance in a nonparametric way can be used to transfer landmarks or segments from an exemplar database to challenging test images (*i.e.*, for facial landmark localization and soft semantic segmentation of facial parts) or between different datasets (*i.e.*, for combining multiple datasets that have different landmark definitions), all with state-of-the-art accuracy.*

Chapter 3 described an exemplar-based approach to facial landmark localization that models the local appearance context of each landmark and global face shape in a nonparametric way. Using this approach, we built an algorithm that produces results with state-of-the-art accuracy on challenging in-the-wild face datasets, including faces with non-neutral expression, non-frontal head pose/camera viewpoint, and partial occlusion.

Chapter 4 described a novel pipeline built as an extension of the facial landmark localization algorithm in Chapter 3 for the task of combining multiple face landmark datasets that have different landmark definitions into a super dataset, with a union of all landmark types computed in each image as output. Our pipeline transfers landmark annotations from multiple source datasets to a target dataset in a collaborative way (*i.e.*, the information from multiple datasets is used jointly in the final landmark localization step). One of the concrete results from this work is a larger supplementary set of landmark definitions for the AFLW face dataset (increasing the original 21 landmarks to 84 landmarks), which is significant because AFLW is probably the

largest publicly available in-the-wild landmark database. We anticipate that our supplementary landmarks will be useful as a richer training and evaluation dataset in future work.

Chapter 5 described an exemplar-based approach to semantic face image segmentation that also models shape and appearance in a nonparametric way. Using this approach, we built an algorithm that produces segmentation results with better accuracy than other methods on challenging in-the-wild images. Our algorithm produces a probability vector of label types at each pixel. Unlike landmarks or contours, this representation is general enough to model any facial part, including facial hair, teeth, forehead, *etc.*

This work extends the state-of-the-art of face image parsing but only begins to address the problem of creating commercially viable systems that surpass human accuracy irrespective of the common types of real-world challenges encountered. Many disparities exist between the academic prototypes developed here and successful products, which must be *fast* (*i.e.*, near real-time) in addition to being accurate and robust against real-world challenges.

6.1 Improving Efficiency

Several strategies exist for improving efficiency without sacrificing accuracy.

1. A careful C++ implementation of each algorithm (as opposed to the MATLAB implementations used for experimentation) would greatly improve runtime.

2. Although the steps of each algorithm must be executed sequentially, each step is an embarrassingly parallel workload (in the parlance of parallel computing). For example, each Hough vote is completely independent of all other Hough votes; similarly, each loop of the RANSAC-like shape regularization algorithm in Section 3.3 can be executed independently in parallel.

3. The current implementation of the landmark localization algorithm computes voting maps across the whole face for each landmark, which is inefficient. We could use an image pyramid to consider fewer features and effectively reduce the size of each landmark's search region.

4. Many exemplar faces provide very little additional information relative to other exemplar faces. The recent work of Li *et al.* [78] suggests that we can eliminate this redundancy to significantly improve efficiency. That is, by selecting the most unique and discriminative faces in the database and discarding the rest (*e.g.*, using AdaBoost [41]), each step of the algorithm would involve less data, which would result in less computation. This is applicable to both landmark localization and soft segmentation.

The above strategies have more to do with improving the implementations rather than the fundamental algorithms. Greater efficiency could be achieved by modifying the algorithms as well. This could be achieved by combining steps in some way: voting for landmarks while simultaneously enforcing an implicit or explicit shape constraint, or removing the need to bootstrap the face segmentation algorithm with an initial set of landmark estimates, for example.

6.2 A Unified Pipeline

Chapter 3, Chapter 4, and Chapter 5 describe different but related algorithms. We can unify all three algorithms in a straightforward way as follows:

Database Construction: Rectify each source/exemplar face; extract features
Runtime Preprocessing: Detect and rescale each target face; extract features
Step A: Top exemplar retrieval
Step B: Weighted landmark voting
Step C: Shape regularization
Step D: Final landmark estimation and integration
Step E: Nonrigid exemplar alignment
Step F: Exemplar label aggregation
Step G: Pixel-wise label selection

Table 6.1 shows the correspondence between steps in each separate algorithm and steps in the unified pipeline above. Step D can be optionally added to the landmark localization algorithm (*i.e.*, by assuming that no landmarks are known in the target face). In the runtime preprocessing step in Chapter 5, sparse landmark localization serves as

Landmark Localization (Chapter 3)	Combining Datasets (Chapter 4)	Soft Segmentation (Chapter 5)	Unified Algorithm
Database Construction	Preprocessing	Database Construction	Database Construction
Runtime Preprocessing	Preprocessing	Runtime Preprocessing	Runtime Preprocessing
Step 1	Stage 1	Runtime Preprocessing	Step A
Step 2	Stage 2	Runtime Preprocessing	Step B
Step 3	Stage 3	Runtime Preprocessing	Step C
—	Stage 4	—	Step D
—	—	Step 1	Step E
—	—	Step 2	Step F
—	—	Step 3	Step G

Table 6.1: Correspondence between steps in each separate algorithm and steps in the unified pipeline given in Section 6.2.

an initialization for soft segmentation. We used a straightforward implementation of Belhumeur *et al.*'s algorithm [7] for this purpose. However, we can instead use Steps A through C (and optionally Step D) for this purpose. If Steps A through D involve only a sparse set of landmarks, then Steps E through F could be used to localize a more dense set of landmarks or contours (*e.g.*, Helen's 194 contour landmarks), which would be less computationally expensive than computing a large set of voting response maps. Step G is an optional step that only applies to soft segmentation.

6.3 Additional Segment Types

Hand labeling face images is time-consuming and tedious, which partially explains the lack of good face datasets that include semantically-defined pixel-wise labels. Chapter 4 addressed the problem of transferring landmark definitions across multiple face datasets. Interesting future work would include extending this approach to other representations, including segments and pixel-wise labels, with the goal of efficiently creating large exemplar databases with additional label types, including ears, teeth, eye pupils, hair, and neck.

For especially challenging images, we could also extend the original pipeline by bringing users into the loop as part of a crowdsourcing effort. Users could vet results for accuracy and help correct failure cases, *i.e.*, by clicking on one or a few landmarks to add additional constraints.

6.4 A Semi-Supervised Approach

A semi-supervised approach to face image parsing could be used to transfer annotations from a relatively small source dataset, or datasets, to much larger target datasets. An important component of such an approach would be an iterative joint optimization of landmark or segment estimates among the target faces, with the goal of making the landmarks or segments more shape- and appearance-consistent. By *appearance-consistent* we mean that the local appearance at each landmark or segment estimate is more similar across images, and by *shape-consistent* we mean the spatial arrangement of landmarks or segments are more consistent. This could also be used to clean up large collections of imprecisely annotated face images. A semi-supervised approach could also be combined with our algorithm for combining multiple face datasets.

6.5 Better Exemplar Database Characterization

Although our approach generally works well, we observe that it occasionally fails on faces with exaggerated mouth expressions. We assume this is due to the fact that some exaggerated facial expressions are not well-represented among the exemplars. We currently do not have a concrete idea of how well different exaggerated expressions are characterized among the exemplar faces. A better characterization of the distribution of exemplar facial expressions would allow us to more accurately determine the range of facial expressions that our algorithms can handle. It would also allow us to compensate for attribute biases among the exemplars. For example, near-frontal faces with neutral facial expression are more common than non-frontal, non-neutral faces in real-world face datasets. This bias can negatively impact the accuracy of our algorithm on head poses and expressions that exist in the long 'tail' of the distribution.

6.6 Occlusion Detection

One limitation of our current approach (and almost all other approaches) is that it assumes face parts are always visible. An artifact of this assumption is that occluded landmarks and segments are incorrectly hallucinated in our results. This behavior is acceptable for some applications, such as face tracking. However, for other applications,

such as automated portrait retouching and, to some extent, face recognition, occlusion detection is necessary in order to generate reasonable results.

Occlusion detection is challenging for many reasons. For example, occlusions can take any form (*i.e.*, generative models are impractical), they often resemble noisy but visible face regions, and the strategies that make algorithms robust to common types of real-world variation are often antithetical to occlusion detection (*i.e.*, algorithms are robust *because* they ignore local aberrations).

One strategy is to decouple occlusion detection from face parsing. That is, first perform face parsing, and then examine each landmark or region to determine whether or not it is occluded. Another strategy is to incorporate occlusion detection directly into the parsing pipeline. Burgos-Artizzu *et al.* [14] recently proposed such a strategy. They incorporated occlusion information directly during learning to improve facial landmark localization. Interesting future work would include pixel-wise occlusion detection (*i.e.*, to explicitly label occluding objects), which is much more challenging than landmark occlusion detection.

6.7 Future Applications

Face image retrieval is an incidental part of the exemplar-based strategy described in this work. However, with more development, the face image retrieval part of the pipeline could be used more prominently for a variety of compelling applications, including face image search [109], face attribute recognition (*e.g.*, age, head pose, facial expression) [66, 67], automatic example-based grayscale portrait colorization, example-based face portrait restoration and superresolution, and personal portrait photo collection management.

In the behavioral sciences it is often necessary to analyze the head pose and facial expression of human subjects. In some situations, it is also necessary to maintain privacy or anonymity. For example, the Federal Highway Administration (FHWA) announced a grant[1] in 2013 for anonymizing naturalistic driving data. The FHWA is interested in better understanding driver behavior leading up to vehicle crashes, and they want to release data to more researchers for analysis, but existing privacy

[1]Federal Highway Administration Broad Agency Announcement No. DTFH61-13-R-00011: "Exploratory Advanced Research Program"

issues make this very difficult. The segment-based face parsing approach described in Chapter 5 is well-suited to the task of automatically masking faces in driver videos, while retaining information useful for behavioral analysis. For example, even when a face is digitally masked to conceal identity, it is possible to infer head pose, eye state, facial expression, *etc.*

6.8 Other Domains

Faces were the focus of this work for several reasons, including:

- Compared to other types of objects, faces are especially compelling to humans.
- Many interesting computer vision applications (*e.g.*, face recognition, automated portrait editing) depend on the accuracy of face parsing algorithms.
- Face datasets are readily available.

However, the general approach described in this dissertation is applicable to other domains. The broader problem addressed here is finding correspondences between multiple image instances of objects that share the same class. For example, in principle, we could apply the same techniques to parse car images for vehicle identification [135], leaf images for plant identification [64] (*i.e.*, to automatically identify edible or poisonous plants), and images of assembled products for quality inspections in factories [87].

Appendix A

Supplementary Facial Landmark Localization Results

Figures A.1, A.2, and A.3 supplement the results in Chapter 3. The input images in Figure A.1 are from the AFW dataset [142], and the input images in Figure A.2 are from the IBUG dataset [99, 98]. We note that our algorithm generally produces accurate results, even on faces with extreme head poses and expression. However, our algorithm is not perfect and makes mistakes on some especially challenging faces, as shown in Figure A.3.

In our judgment, faces with unusual mouth shapes and/or significant yaw+pitch head rotation are the most challenging for our implementation. We speculate that this is largely due to the limited head pitch rotation variation and mouth shape variation in the Multi-PIE Face Database [45], which we use as our sole exemplar database. Head yaw and pitch combinations like those shown in the left two columns of Figure A.3, and the mouth shapes shown in the right two columns of Figure A.3, are not well-represented in our database, which results in poor landmark votes from the top exemplar images to the test image. Despite these challenges, our algorithm generally performs well on mouths, and on faces with significant yaw+pitch head rotation.

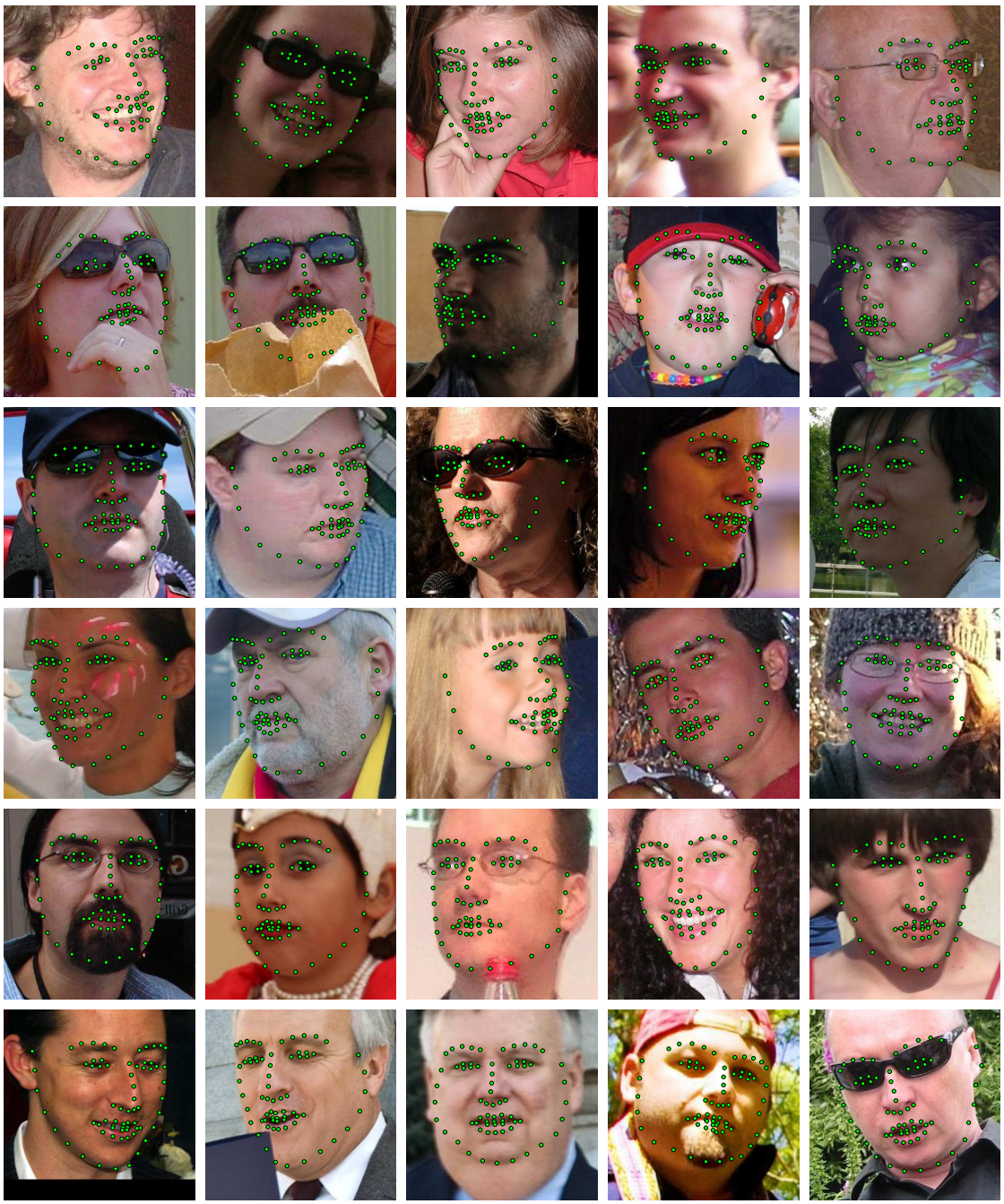


Figure A.1: **Selected results on AFW [142].** Our method can handle a wide variety of very challenging conditions, including significant image noise and blur, occlusions, and extreme expressions and head poses. **Best viewed electronically in color.**

Figure A.2: **Selected results on IBUG [99, 98].** Our method can handle a wide variety of very challenging conditions, including significant image noise and blur, occlusions, and extreme expressions and head poses. **Best viewed electronically in color.**

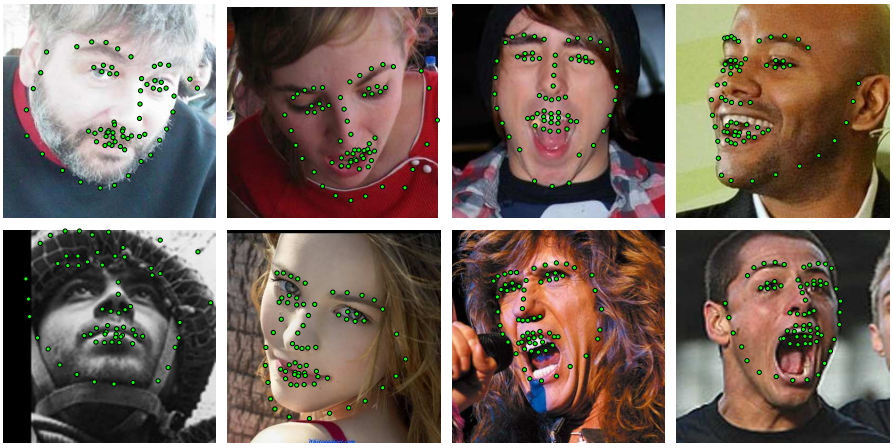


Figure A.3: **Failure cases on faces with significant yaw+pitch rotation (left two columns) and on mouths (right two columns).** The input images on the top row are from AFW [142], and the input images on the bottom row are from IBUG [99, 98]. Large errors occur infrequently, but when they do occur they are usually localized to the eyes (in the case of significant yaw and pitch head rotation) and the mouth (in the case of unusual mouth expressions). Yaw and pitch combinations and unusual mouth expressions like those shown above are not well-represented in our exemplar database, which results in poor landmark votes from the top exemplar images to the test image. **Best viewed electronically in color.**

Appendix B

Supplementary Landmark Transfer Results on AFLW Faces

The Annotated Facial Landmarks in the Wild (AFLW) face database [60] is an ideal target for our system. AFLW includes 25k real-world faces with up to 21 landmark annotations on each face. Our system uses these known landmarks as constraints to help localize 85 landmarks from multiple source datasets (Multi-PIE [45], Helen [74], LFPW [7], AFW [142], and IBUG [98]).

A quantitative evaluation on AFLW is impractical because (1) our system uses all 21 known landmarks as constraints, and (2) no additional ground truth is provided beyond the 21 landmarks. Therefore, we instead show a large selection of qualitative results on challenging faces from AFLW, including:

Figure B.1: partially occluded faces,

Figure B.2: faces with significant yaw and/or pitch head rotation,

Figure B.3: young and old faces,

Figure B.4: faces with non-neutral expressions,

Figure B.5: bearded, painted, blurred, bespectacled, and dramatically illuminated faces, and

Figure B.6: problem cases on especially difficult cases.

We see that our system generally performs very well despite these challenges. Our future work will include detecting landmark occlusions (*i.e.*, due to extreme head pose), and introducing humans in the loop as part of a crowd sourcing platform to efficiently correct mistakes; users will typically only need to manually click on one or two landmarks to correct problematic contours.

Figure B.1: Results on partially occluded faces due to sunglasses (first column), hands (second column), hair and clothing (third and fourth columns), and objects (fifth column). Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. **Best viewed electronically in color.**

Figure B.2: Results on faces with significant yaw and/or pitch head rotation. Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. **Best viewed electronically in color.**

Figure B.3: Results on young faces (first three columns) and older faces (last two columns). Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. **Best viewed electronically in color.**

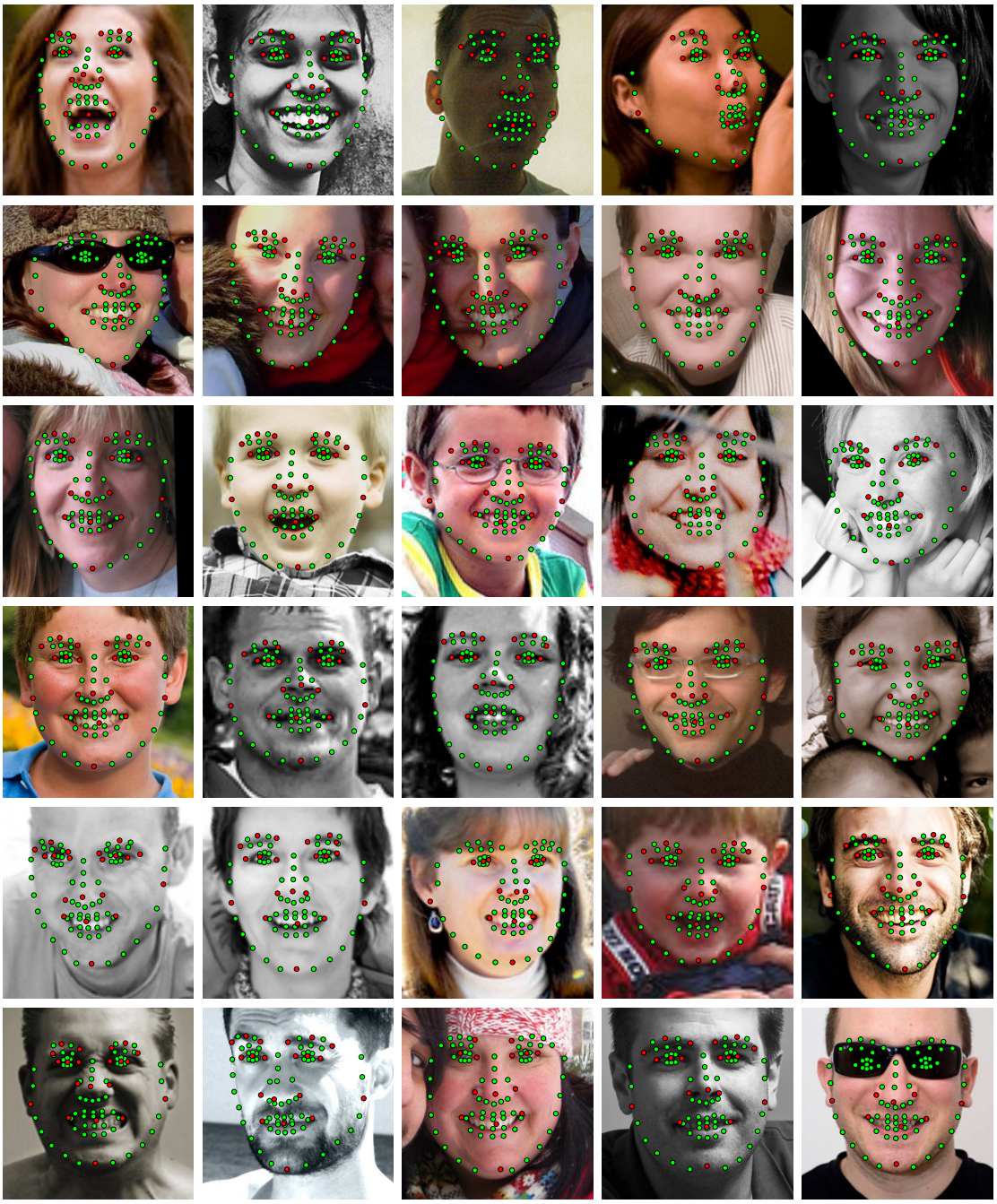


Figure B.4: Results on faces with non-neutral expressions. Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. **Best viewed electronically in color.**

Figure B.5: Results on faces with beards (first column), face paint (second column), eyeglasses (third column), image blur (fourth column), and challenging illumination (fifth column). Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. **Best viewed electronically in color.**

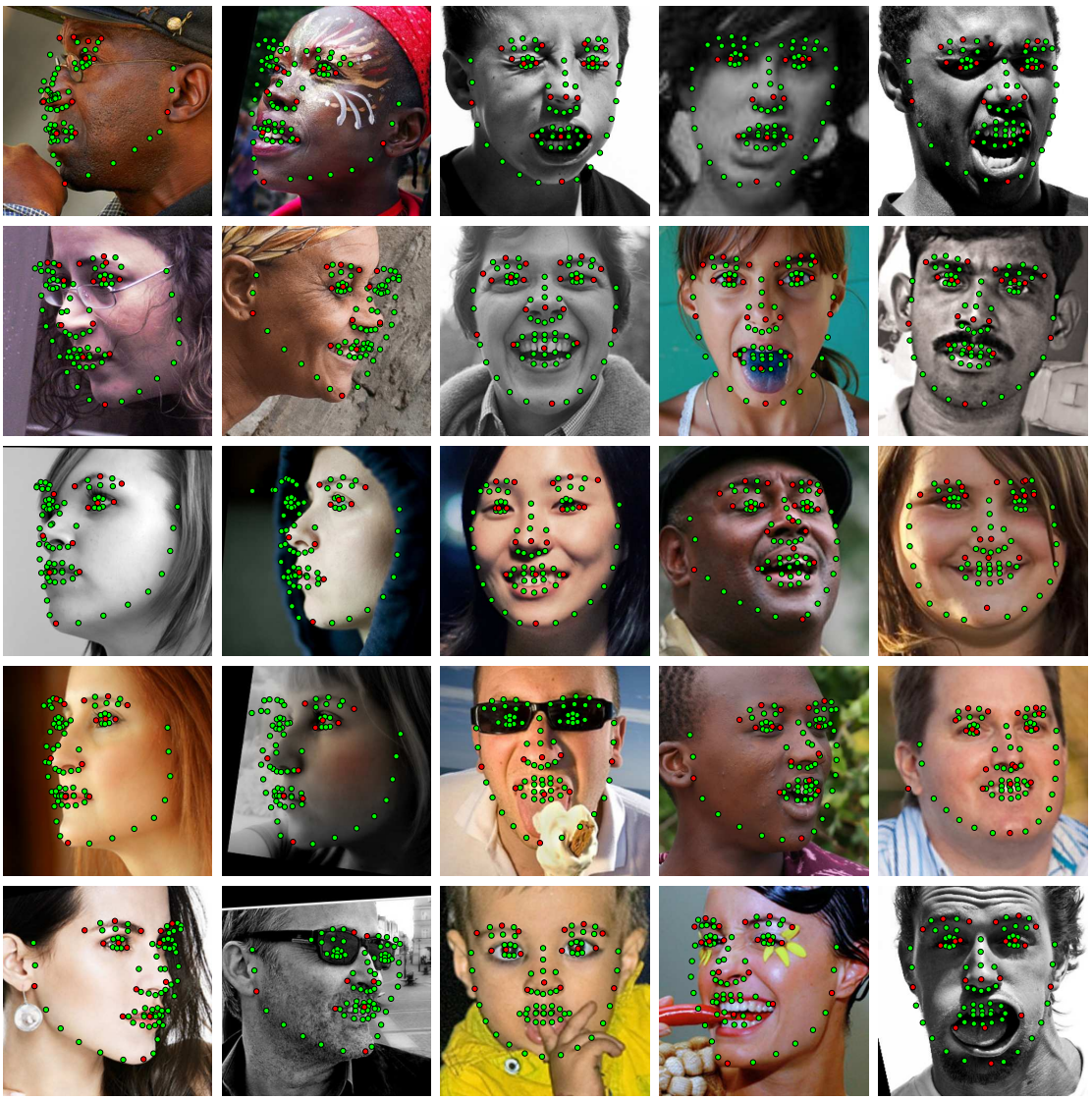


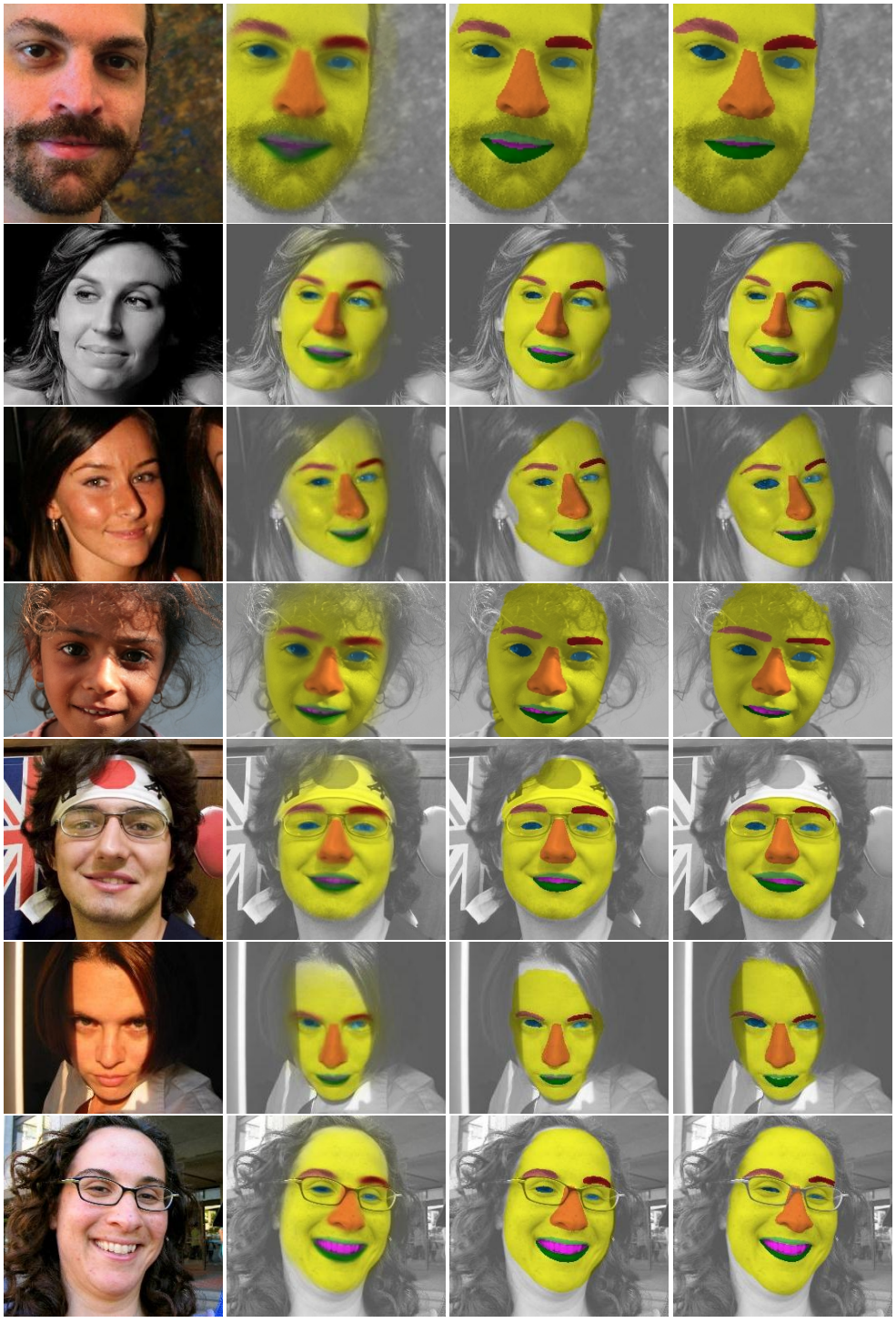
Figure B.6: Our system generally performs well, but occasionally makes mistakes. Localization errors sometimes occur due to: (1) self-occluded landmarks on full profile faces, which our algorithm incorrectly hallucinates (first two columns), (2) unusual mouth expressions not well-covered by the source datasets (third and fourth columns), and (3) errors among the 'known' landmark annotations in AFLW (fifth column). Red dots denote 'known' landmarks provided by AFLW, and green dots denote landmarks estimated by our system. Future work could include detecting landmark occlusions (*i.e.*, due to extreme head pose), and introducing humans in the loop as part of a crowd sourcing platform for efficiently correcting mistakes; users will typically only need to manually click on one or two landmarks to correct problematic contours. **Best viewed electronically in color.**

Appendix C

Supplementary Face Segmentation Results

Figures C.1 and C.2 supplement the results in Chapter 5. In all cases, the input images come from our Helen [74] test set. We note that our algorithm generally produces accurate results, as shown in Figures C.1. However, our algorithm is not perfect and makes mistakes on especially challenging input images, as shown in Figure C.2.

In our view, the mouth is the most challenging region of the face to segment: the shape and appearance of the lips vary widely from subject to subject, mouths deform significantly, and the overall appearance of the mouth region changes depending on whether the inside of the mouth is visible or not. Unusual mouth expressions, like those shown in Figure C.2, are not well-represented in the exemplar images, which results in poor label transfer from the top exemplars to the test image. Despite these challenges, our algorithm generally performs well on the mouth, with large segmentation errors occurring infrequently.



Original image | Our method, *soft* segmentation | Our method, *hard* segmentation | Ground truth

Figure C.1: Selected qualitative results. We note that our algorithm generally produces accurate results. **Best viewed in color.**

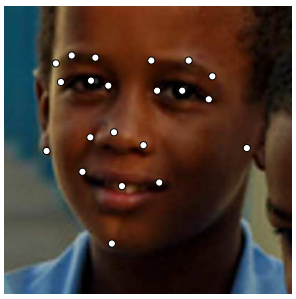
Original image | Our method, *soft* segmentation | Our method, *hard* segmentation | Ground truth

Figure C.2: Failure cases on mouths. Large segmentation errors occur infrequently, but when they do occur, errors are almost always localized to the mouth region. Unusual mouth expressions like those shown above are not well-represented in the exemplar images, which results in poor label transfer from the top exemplars to the test image. **Best viewed in color.**

Appendix D

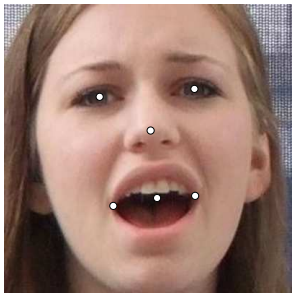
Summary of Face Image Datasets Used in This Dissertation

AFLW (Annotated Facial Landmarks in the Wild) [60]



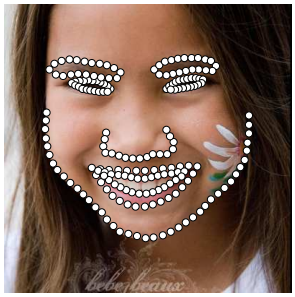
Type	In-the-wild
Year released	2011
Resolution	Various, high resolution
Number of images	21,997
Number of faces	25,993
Number of subjects	Unknown, but few duplicates
Number of landmarks	Up to 21 (many missing)
Head pose variation	$\pm 90^\circ$ yaw, $\pm 30^\circ$ roll, $\pm 30^\circ$ pitch
Facial expressions	Various, mostly neutral or smiling
Demographics	Wide variety of ages and ethnic backgrounds
Annotation details	Each face was hand-annotated using a tool developed at the Institute for Computer Graphics and Vision at Graz University of Technology. The landmark annotations are sometimes inaccurate.

AFW (Annotated Faces in the Wild) [142]



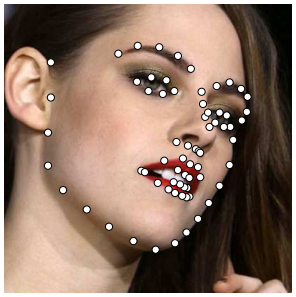
Type	In-the-wild
Year released	2012
Resolution	Various, high resolution
Number of images	205
Number of faces	468
Number of subjects	Unknown, but few duplicates
Number of landmarks	6
Head pose variation	$\pm 90°$ yaw, $\pm 15°$ roll, $\pm 30°$ pitch
Facial expressions	Various, mostly neutral or smiling
Demographics	Wide variety of ages and ethnic backgrounds
Annotation details	Each face was hand-annotated by Xiangxin Zhu or Deva Ramanan. 337 faces were separately annotated with 68 landmarks according to the Multi-PIE arrangement [45] as part of the 300 Faces in the-wild Challenge [98].

Helen [74]



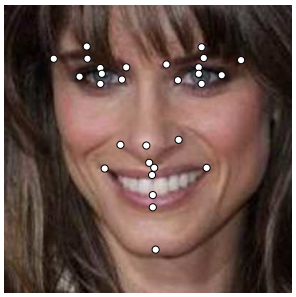
Type	In-the-wild
Year released	2012
Resolution	Various, high resolution
Number of images	2,330
Number of faces	2,330
Number of subjects	Unknown, but some duplicates
Number of landmarks	194
Head pose variation	$\pm 45^\circ$ yaw, $\pm 15^\circ$ roll, $\pm 30^\circ$ pitch
Facial expressions	Large variety
Demographics	Wide variety of ages and ethnic backgrounds
Annotation details	Pre-screened Amazon Mechanical Turk workers hand-annotated each face (one worker per face) using an interactive tool. The contours were reviewed manually by the authors to ensure high quality annotations. All faces were separately annotated with 68 landmarks according to the Multi-PIE arrangement [45] as part of the 300 Faces in the-wild Challenge [98].
Additional comments	Landmarks are spaced evenly along each contour. Therefore, landmarks with the same ID do not necessarily correspond to exactly the same semantic point on the face.

IBUG (Intelligent Behavior Understanding Group) [98]



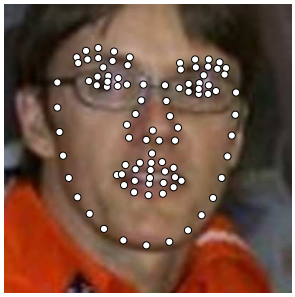
Type	In-the-wild
Year released	2013
Resolution	Various, high resolution
Number of images	135
Number of faces	135
Number of subjects	Unknown, but some duplicates
Number of landmarks	68
Head pose variation	±60° yaw, ±30° roll, ±45° pitch
Facial expressions	Large variety
Demographics	Almost all white adults; many are celebrities
Annotation source	The authors used a semi-automatic tool [99] to annotate each face. The landmark annotations are very accurate.
Comments	All images depict non-frontal head pose and/or exaggerated facial expression.

LFPW (Labeled Face Parts in the Wild) [7]



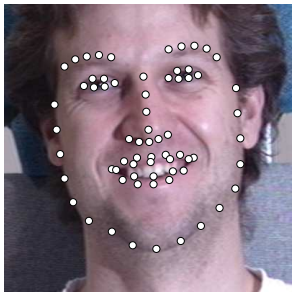
Type	In-the-wild
Year released	2011
Resolution	Various, high resolution
Number of images	1,432
Number of faces	1,432
Number of subjects	Unknown, but some duplicates
Number of landmarks	29
Head pose variation	±45° yaw, ±15° roll, ±15° pitch
Facial expressions	Large variety
Demographics	Almost all white adults; many are celebrities
Annotation details	Three Amazon Mechanical Turk workers hand-annotated each face. The authors used the average location of each landmark as ground truth. 1,035 faces were separately annotated with 68 landmarks according to the Multi-PIE arrangement [45] as part of the 300 Faces in the-wild Challenge [98].

LFW (Labeled Faces in the Wild) [51]



Type	In-the-wild
Year released	2007
Resolution	Consistent, 250×250 pixels
Number of images	13,233
Number of faces	13,233
Number of subjects	5,749
Number of landmarks	Varies[†]
Head pose variation	$\pm 45°$ yaw, $\pm 15°$ roll, $\pm 15°$ pitch
Facial expressions	Large variety
Demographics	Almost all adult celebrities; ethnically diverse
Annotation details	[†]Luo *et al.* [86] independently hand-annotated 300 faces with 94 landmarks each, as shown above right. Saragih [101] independently hand-annotated 1,000 faces with 66 landmarks according to the Multi-PIE landmark arrangement [45] minus the inner mouth corners.

Multi-PIE (CMU Multiple Pose Illumination Expression) [45]



Type	Laboratory
Year released	2008
Resolution	Consistent, 480×640 pixels
Number of images	754,202
Number of faces	754,202
Number of subjects	337
Number of landmarks	68
Head pose variation	$-90°$ to $+90°$ yaw in increments of $15°$; almost no pitch or roll variation
Facial expressions	Approximately 46% Neutral, 18% smile, 9% surprise, 9% squint, 9% disgust, and 9% scream
Demographics	Representative of staff and students at CMU
Annotation details	The landmark locations for 4,685 images were "manually established" by the authors. We observe that the landmark locations are very accurate. Zhu and Ramanan [142] separately hand-annotated an additional 400 (mostly non-frontal) faces. We hand-annotated 960 more (mostly non-frontal faces with non-neutral expressions).
Additional comments	Each {session, subject, head pose, facial expression} snapshot includes 19 different flash illuminations. Therefore, the number of landmark-annotated images is effectively $6048 \times 19 = 114855$.

References

[1] OpenCV (Open Source Computer Vision) library, http://code.opencv.org/.

[2] Brian Amberg and Thomas Vetter. Optimal landmark detection using shape models and branch and bound. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[3] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Robust discriminative response map fitting with constrained local models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[4] Simon Baker and Iain Matthews. Lucas-Kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, February 2004.

[5] Dana H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[6] Robert J. Baron. Mechanisms of human facial recognition. *International Journal of Man-Machine Studies*, 15(2):137–178, August 1981.

[7] Peter N. Belhumeur, David W. Jacobs, David J. Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[8] Dmitri Bitouk, Neeraj Kumar, Samreen Dhillon, Peter N. Belhumeur, and Shree K. Nayar. Face Swapping: Automatically Replacing Faces in Photographs. *ACM Transactions on Graphics*, 27(3):39:1–39:8, August 2008.

[9] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 187–194, 1999.

[10] Woodrow Wilson Bledsoe. The model method in facial recognition. Technical Report PRI 15, Panoramic Research, Inc., Palo Alto, California, 1964.

[11] Woodrow Wilson Bledsoe. Man-machine facial recognition: Report on a large-scale experiment. Technical Report PRI 22, Panoramic Research, Inc., Palo Alto, California, 1966.

[12] Woodrow Wilson Bledsoe. Semiautomatic facial recognition. Technical Report SRI Project 6693, Stanford Research Institute, Menlo Park, California, 1968.

[13] Woodrow Wilson Bledsoe and Helen Chan. A man-machine facial recognition system—some preliminary results. Technical Report PRI 19A, Panoramic Research, Inc., Palo Alto, California, 1965.

[14] Xavier P. Burgos-Artizzu, Pietro Perona, and Piotr Dollár. Robust face landmark estimation under occlusion. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[15] D. J. Burr. A dynamic model for image registration. *Computer Graphics and Image Processing*, 15:102–112, 1981.

[16] D. J. Burr. Elastic matching of line drawings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3(6):708–713, 1981.

[17] Richard H. Byrd, Mary E. Hribar, and Jorge Nocedal. An interior point algorithm for large-scale nonlinear programming. *Society for Industrial and Applied Mathematics (SIAM) Journal on Optimization*, 9(4):877–900, 1999.

[18] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. Face alignment by explicit shape regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[19] Angela Caunce, David Cristinacce, Christopher J. Taylor, and Timothy F. Cootes. Locating facial features and pose estimation using a 3D shape model.

In *Advances in Visual Computing*, volume 5875 of *Lecture Notes in Computer Science*, pages 750–761. Springer-Verlag, 2009.

[20] Yizong Cheng. Mean shift, mode seeking, and clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(8):790–799, August 1995.

[21] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. In *European Conference on Computer Vision (ECCV)*, 1998.

[22] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001.

[23] Timothy F. Cootes, A. Hill, Christopher J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. *Information Processing in Medical Imaging*, 687:33–47, 1993.

[24] Timothy F. Cootes, Mircea C. Ionita, Claudia Lindner, and Patrick Sauer. Robust and accurate shape model fitting using random forest regression voting. In *European Conference on Computer Vision (ECCV)*, 2012.

[25] Timothy F. Cootes and Christopher J. Taylor. Active shape models – 'smart snakes'. In *British Machine Vision Conference (BMVC)*, 1992.

[26] Timothy F. Cootes and Christopher J. Taylor. Constrained active appearance models. In *IEEE International Conference on Computer Vision (ICCV)*, 2001.

[27] Timothy F. Cootes, Christopher J. Taylor, D. H. Cooper, and J. Graham. Training models of shape from sets of examples. In *British Machine Vision Conference (BMVC)*, 1992.

[28] Timothy F. Cootes, Christopher J. Taylor, D. H. Cooper, and J. Graham. Active shape models—their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995.

[29] Ian Craw and Peter Cameron. Race recognition by computer. In *British Machine Vision Conference (BMVC)*, 1992.

[30] Ian Craw, Hadyn D. Ellis, and J. Rowland Lishman. Automatic extraction of face-features. *Pattern Recognition Letters*, 5:183–187, February 1987.

[31] Ian Craw, David Tock, and Alan Bennett. Finding face features. In *European Conference on Computer Vision (ECCV)*, 1992.

[32] Ian Craw, Davit Tock, and Alan Bennett. Finding face features. Technical report, Department of Mathematical Sciences, University of Aberdeen, Aberdeen, UK, 1991.

[33] David Cristinacce and Timothy F. Cootes. Feature detection and tracking with constrained local models. In *British Machine Vision Conference (BMVC)*, 2006.

[34] David Cristinacce and Timothy F. Cootes. Automatic feature localisation with constrained local models. *Pattern Recognition*, 41(10):3054–3067, October 2008.

[35] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[36] Matthias Dantone, Juergen Gall, Gabriele Fanelli, and Luc Vn Gool. Real-time facial feature detection using conditional regression forests. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[37] Facebook, Ericsson, and Qualcomm. *A Focus on Efficiency, http://internet.org/efficiencypaper*, September 2013.

[38] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[39] Martin A. Fischler and Robert C. Bolles. Random Sample Consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.

[40] Martin A. Fischler and Robert A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, C-22(1):67–92, January 1973.

[41] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

[42] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28(2):337–407, 2000.

[43] Athinodoros S. Georghiades, Peter N. Belhumeur, and David J. Kriegman. From few to many: Generative models for recognition under variable pose and illumination. In *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.

[44] Floraine Grabler, Maneesh Agrawala, Wilmot Li, Mira Dontcheva, and Takeo Igarashi. Generating photo manipulation tutorials by demonstration. *ACM Transactions on Graphics*, 28(3):66:1–66:9, July 2009.

[45] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, May 2010.

[46] Leon Gu and Takeo Kanade. A generative shape regularization model for robust face alignment. In *European Conference on Computer Vision (ECCV)*, 2008.

[47] Dong Guo and Terence Sim. Digital face makeup by example. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[48] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 498–520, 1933.

[49] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3-4):321–377, 1936.

[50] Paul V. C. Hough. Method and means for recognizing complex patterns. Patent US 3069654, 1962.

[51] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[52] Oliver Jesorsky, Klaus J. Kirchberg, and Robert W. Frischholz. Robust face detection using the hausdorff distance. In *Third International Conference on Audio- and Video-Based Biometric Person Authentication*, 2001.

[53] Hao Jiang, Mark S. Drew, and Ze-Nia nLi. Matching by linear programming and successive convexification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):959–975, 2007.

[54] Kari Karhunen. Über lineare methoden in der wahrscheinlichkeitsrechnung (English: "On linear methods in probability and statistics"). *Annales Academiæ-Scientarum FennicæMathematica*, 37:1–79, 1947.

[55] Andrzej Kasiński, Andrzej Florek, and Adam Schmidt. The PUT face database. *Image Processing and Communications*, 13(3-4):59–64, 2008.

[56] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision (IJCV)*, 1(4):321–331, 1988.

[57] Michael David Kelly. *Visual Identification of People by Computer*. PhD thesis, Stanford University, 1971.

[58] Ira Kemelmacher-Shlizerman, Aditya Sankar, Eli Shechtman, and Steven M. Seitz. Being John Malkovich. In *European Conference on Computer Vision (ECCV)*, 2010.

[59] Michael Kirby and Lawrence Sirovich. Application of the Karhunen-Loève procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, January 1990.

[60] Martin Koestinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[61] Peter Kontschieder, Samuel Rota Bulò, Horst Bischof, and Marcello Pelillo. Structured class-labels in random forests for semantic image labeling. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[62] Peter Kontschieder, Samuel Rota Bulò, Michael Donoser, Marcello Pelillo, and Horst Bischof. Evolutionary hough games for coherent object detection. *Computer Vision and Image Understanding*, 116(11):1149–1158, November 2012.

[63] Daniel Kuettel and Vittorio Ferrari. Figure-ground segmentation by transferring window masks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[64] Neeraj Kumar, Peter N. Belhumeur, Arijit Biswas, David W. Jacobs, W. John Kress, Ida Lopez, and Jo ao V. B. Soares. Leafsnap: A computer vision system for automatic plant species identification. In *European Conference on Computer Vision (ECCV)*, 2012.

[65] Neeraj Kumar, Peter N. Belhumeur, and Shree K. Nayar. FaceTracer: A search engine for large collections of images with faces. In *European Conference on Computer Vision (ECCV)*, 2008.

[66] Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. Attribute and simile classifiers for face verification. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[67] Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. Describable visual attributes for face verification and image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1962–1977, October 2011.

[68] Martin Lades, Jan C. Vorbrüggen, Joachim Buhmann, Jörg Lange, Christoph Von Der Malsburg, Rolf P. Würtz, and Wolfgang Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, March 1993.

[69] Ailsa H. Land and Alison G. Doig. An automatic method for solving discrete programming problems. In Michael Jünger, Thomas M. Liebling, Denis Naddef, George L. Nemhauser, William R. Pulleyblank, Gerhard Reinelt, Giovanni Rinaldi, and Laurence A. Wolsey, editors, *50 Years of Integer Programming 1958-2008*, pages 105–132. Springer Berlin Heidelberg, 2010.

[70] Andreas Lanitis, Timothy F. Cootes, and Christopher J. Taylor. Automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13(5):393–401, June 1995.

[71] Andreas Lanitis, Christopher J. Taylor, and Timothy F. Cootes. An automatic face identification system using flexible appearance models. In *British Machine Vision Conference*, 1994.

[72] Andreas Lanitis, Christopher J. Taylor, and Timothy F. Cootes. A unified approach to coding and interpreting face images. In *IEEE International Conference on Computer Vision (ICCV)*, pages 368–373, June 1995.

[73] Andreas Lanitis, Christopher J. Taylor, and Timothy F. Cootes. Automatic interpretation and coding of face images using flexible models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):743–756, July 1997.

[74] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S. Huang. Interactive facial feature localization. In *European Conference on Computer Vision (ECCV)*, 2012.

[75] Bestian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *European Conference on Computer Vision (ECCV) Workshop on Statistical Learning in Computer Vision*, 2004.

[76] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. *ACM Transactions on Graphics*, 23(3):689–694, August 2004.

[77] Anat Levin, Alex Rav-Acha, and Dani Lischinski. Spectral matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1699–1712, October 2008.

[78] Haoxiang Li, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Gang Hua. Efficient boosted exemplar-based face detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[79] Hongsheng Li, Xiaolei Huang, and Lei He. Object matching using a locally affine invariant and linear programming techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):411–424, 2013.

[80] Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing via label transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2368–2382, December 2011.

[81] Ce Liu, Jenny Yuen, and Antonio Torralba. SIFT Flow: Dense correspondence across scenes and its applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):978–994, May 2011.

[82] Xiaoming Liu. Generic face alignment using boosted appearance model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[83] Michel Lo'eve. *Probability Theory, Volume II, 4th edition (Graduate Texts in Mathematics)*. Springer-Verlag, 1978.

[84] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[85] Simon Lucey, Yang Wang, Jason Saragih, and Jeffrey F. Cohn. Non-rigid face tracking with enforced convexity and local appearance consistency constraint. *Image and Vision Computing*, 28(5):781–789, May 2010.

[86] Ping Luo, Xiaogang Wang, and Xiaoou Tang. Hierarchical face parsing via deep learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[87] Elias N. Malamas, Euripides G. M. Petrakis, Michalis Zervakis, Laurent Petit, and Jean-Didier Lagat. A survey on industrial vision systems, applications and tools. *Image and Vision Computing*, 21(2):171–188, February 2003.

[88] Aleix M. Martinez and Robert Benavente. The AR Face Database. Technical Report 24, Computer Vision Center (CVC) at the Autonomous University of Barcelona (UAB), June 1998.

[89] Iain Matthews and Simon Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, November 2004.

[90] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended m2vts database. In *Second International Conference on Audio and Video-based Biometric Person Authentication*, 1999.

[91] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *VISAPP International Conference on Computer Vision Theory and Applications*, 2009.

[92] Mark Nixon. Eye spacing measurement for facial recognition. In *Applications of Digital Image Processing VIII*, December 1985.

[93] Karl Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(11):559–572, 1901.

[94] Frédéric Pighin, Jamie Hecker, Dani Lischinski, Richard Szeliski, and David H. Salesin. Synthesizing realistic facial expressions from photographs. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 75–84, 1998.

[95] Nima Razavi, Juergen Gall, Pushmeet Kohli, and Luc van Gool. Latent Hough transform for object detection. In *European Conference on Computer Vision (ECCV)*, 2012.

[96] Cédric Rousset and Pierre-Yves Coulon. Frequential and color analysis for hair mask segmentation. In *IEEE International Conference on Image Processing (ICIP)*, 2008.

[97] Detlef Ruprecht and Heinrich Müller. Image warping with scattered data interpolation. *IEEE Computer Graphics and Applications*, 15(2):37–43, March 1995.

[98] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 Faces in-the-Wild Challenge: The first facial landmark localization challenge. In *IEEE International Conference on Computer Vision (ICCV), 300 Faces in-the-Wild Challenge Workshop (300-W)*, 2013.

[99] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. A semi-automatic methodology for facial landmark annotation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 5th Workshop on Analysis and Modeling of Faces and Gestures (AMFG)*, 2013.

[100] Toshiyuki Sakai, Makoto Nagao, and Takeo Kanade. Computer analysis and classification of photographs of human faces. In *First USA-Japan Computer Conference*, 1972.

[101] Jason Saragih. Principal regression analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011.

[102] Jason Saragih and Roland Göecke. A nonlinear discriminative approach to aam fitting. In *IEEE International Conference on Computer Vision (ICCV)*, 2007.

[103] Jason Saragih and Roland Göecke. Learning aam fitting through simulation. *Pattern Recognition*, 42(11):2628–2636, 2009.

[104] Jason M. Saragih, Simon Lucey, and Jeffrey F. Cohn. Face alignment through subspace constrained mean-shifts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[105] Xiaohui Shen, Zhe Lin, Jonathan Brandt, and Ying Wu. Detecting and aligning faces by image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[106] Terence Sim, Simon Baker, and Maan Bsat. The CMU pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1615âŁ“–1618, December 2003.

[107] Pawan Sinha, Benjamin Balas, Yuri Ostrovsky, and Richard Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11):1948–1962, November 2006.

[108] Lawrence Sirovich and Michael Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519–524, March 1987.

[109] Brandon M. Smith, Shengqi Zhu, and Li Zhang. Face image retrieval by shape manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[110] Birgi Tamersoy, Changbo Hu, and J.K. Aggarwal. Nonparametric facial feature localization. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures (AMFG) in conjunction with the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[111] Joseph Tighe and Svetlana Lazebnik. Superparsing: Scalable nonparametric image parsing with superpixels. In *European Conference on Computer Vision (ECCV)*, 2010.

[112] Philip A. Tresadern, Patrick Sauer, and Tim F. Cootes. Additive update predictors in active appearance models. In *British Machine Vision Conference (BMVC)*, 2010.

[113] Matthew A. Turk. *Interactive-Time Vision: Face Recognition as a Visual Behavior*. PhD thesis, Massachusetts Institute of Technology, 1991.

[114] Matthew A. Turk and Alex P. Pentland. Face recognition without features. In *International Association of Pattern Recognition (IAPR) Workshop on Machine Vision Applications (MVA)*, pages 267–270, November 1990.

[115] Matthew A. Turk and Alex P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[116] Matthew A. Turk and Alex P. Pentland. Face recognition using Eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1991.

[117] Georgios Tzimiropoulos and Maja Pantic. Optimization problems for fast AAM fitting in-the-wild. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[118] Paul Viola and Michael J. Jones. Robust real-time face detection. *International Journal of Computer Vision (IJCV)*, 57(2):137–154, May 2004.

[119] Yang Wang, Simon Lucey, and Jeffrey F. Cohn. Enforcing convexity for improved alignment with constrained local models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[120] Jonathan Warrell and Simon J. D. Prince. Labelfaces: Parsing facial features by multiclass labeling with an epitome prior. In *IEEE International Conference on Image Processing (ICIP)*, 2009.

[121] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Technical Report IR-INI 96-08, Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany, 1996.

[122] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, July 1997.

[123] Hao Wu, Xiaoming Liu, and Gianfranco Doretto. Face alignment via boosted ranking model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[124] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[125] Fei Yang, Jue Wang, Eli Shechtman, Lubamir Bourdev, and Dimitri Metaxas. Expression flow for 3D-aware face component transfer. *ACM Transactions on Graphics*, 30(4):60:1–60:10, July 2011.

[126] Heng Yang and Ioannis Patras. Sieving regression forest votes for facial feature detection in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[127] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[128] Yiqing Yang, Zhouyuan Li, Li Zhang, and Christopher Murphy. Local label descriptor for example based semantic image labeling. In *European Conference on Computer Vision (ECCV)*, 2012.

[129] Kwang Moo Yi, Hawook Jeong, Byeongho Heo, Hyung Jin Chang, and Jin Young Choi. Initialization-insensitive visual tracking through voting with salient local features. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[130] Xiang Yu, Junzhou Huang, Shaoting Zhang, Wang Yan, and Dimitris N. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[131] Alan L. Yuille. Deformable templates for face recognition. *Journal of Cognitive Neuroscience*, 3(1):59–70, 1991.

[132] Alan L. Yuille, David S. Cohen, and Peter W. Hallinan. Feature extraction from faces using deformable templates. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1989.

[133] Alan L. Yuille, Peter W. Hallinan, and David S. Cohen. Feature extraction from faces using deformable templates. *International Journal of Computer Vision (IJCV)*, 8(2):99–111, August 1992.

[134] Li Zhang, Haizhou Ai, Shengjun Xin, Chang Huang, Shuichiro Tsukiji, and Shihong Lao. Robust face alignment based on local texture classifiers. In *IEEE International Conference on Image Processing*, 2005.

[135] Zhaoxiang Zhang, Tieniu Tan, Kaiqi Huang, and Yunhong Wang. Three-dimensional deformable-model-based localization and recognition of road vehicles. *IEEE Transactions on Image Processing*, 21(1):1–13, January 2012.

[136] Zhengyou Zhang, Zicheng Liu, Dennis Adler, Michael F. Cohen, Erik Hanson, and Ying Shan. Robust and rapid generation of animated faces from video images: A model-based modeling approach. *International Journal of Computer Vision (IJCV)*, 58(2):93–119, July 2004.

[137] Cong Zhao, Wai-Kuen Cham, and Xiaogang Wang. Joint face alignment with a generic deformable face model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[138] Wen-Yi Zhao, Rama Chellappa, P. Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, December 2003.

[139] Xiaowei Zhao, Shiguang Shan, Xiujuan Chai, and Xilin Chen. Locality-constrained active appearance model. In *Asian Conference on Computer Vision (ACCV)*, 2012.

[140] Xiaowei Zhao, Shiguang Shan, Xiujuan Chai, and Xilin Chen. Cascaded shape space pruning for robust facial landmark detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[141] Feng Zhou, Jonathan Brandt, and Zhe Lin. Exemplar-based graph matching for robust facial landmark localization. In *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[142] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation and landmark localization in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.