Discriminative Appearance Models for Face Alignment

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

der Fakultät für Informatik
des Karlsruher Institut für Technologie (KIT)

genehmigte

Dissertation

von

Hua Gao

aus Jiangsu, China

Tag der mündlichen Prüfung:	**19.06.2013**
Erster Gutachter:	**Prof. Dr. R. Stiefelhagen**
Zweiter Gutachter:	**Prof. Dr. T. F. Cootes**
Dritter Gutachter:	**Dr.-Ing. H. K. Ekenel**, **Forschungsgruppenleiter**

Abstract

Facial image analysis has found its application in various fields including security, entertainment, multimedia indexing, human-computer interaction, etc. Essentially, as an early step in facial image analysis, face alignment has a crucial impact on the robustness and quality of the later processes. Face alignment has been studied for several decades, yet it is still a difficult problem, which suffers from the confounding factors of intrinsic and extrinsic imaging conditions. Due to these challenges, it is still an interesting research problem and receives increasing attention. In particular, deformable model based face alignment has become very popular, since the invention of the Active Shape Model (ASM) and the Active Appearance Model (AAM). Numerous successful application systems have been developed based on deformable models. However, it has been shown in many works that the AAM suffers from generalization problems due to generative appearance modeling and least square minimization. In this thesis, we address on solving two main problems in deformable model-based face alignment, *i.e.*, appearance variations due to illumination, occlusion, image quality etc.; and generalization ability due to appearance modeling and definition of an alignment objective function. We aim at developing a face alignment algorithm framework that is reliable enough for real-world applications.

In the proposed face alignment algorithm, local gradient-based features are used as our appearance representation. The gradient features are obtained by pixel value comparison, which provide robustness against changes in illumination. Due to the locality, the local gradient features are less sensitive to appearance variations in local regions caused by partial face occlusion or facial expression. The features used in this thesis include Pseudo Census Transform (PCT) for deriving analytical alignment algorithms, Modified Census Transform (MCT) and RAndom Pixel Intensity Difference (RAPID) for direct search based alignment algorithms. More importantly, the adopted features are modeled in three discriminative methods, namely, classification, ranking, and regression, which correspond to different alignment cost functions. The alignment cost functions indicate (relative) correctness of alignments. Aligning a face image is equivalent to optimizing the cost function with respect to alignment parameters. The discriminative appearance modeling and the cost function learning alleviate the generalization problem to some extent.

We evaluate the alignment performance of the proposed discriminative appearance models at different levels on four publicly available face databases, namely, on the FERET, FRGC, IMM, and LFW face databases. Extensive experiments are carried out to analyze the effects of the model parameters on the alignment performance. Experimental results show that the proposed discriminative

appearance models achieve higher alignment convergence rates than the generative AAM, especially on unseen data. This provides evidence of superiority in generalization ability of the proposed models. Other observations from the experiments are that the discriminative appearance models based on ranking and regression are superior to the classification based models due to the increased smoothness in the score function. The regression-based appearance model with the RAPID feature achieves the best performance in this study. In addition, we show in the experiments that it outperforms two state-of-the-art discriminative face alignment models.

To evaluate the performance of the proposed alignment algorithms under difficult conditions, we systematically analyze the alignment robustness under different levels of image noise, synthetic occlusion, and illumination. Experiments show that the RAPID-based appearance model is the most reliable one against image noise and occlusion. However, it is interesting to observe that this representation has a generalization problem under extreme illumination conditions. While the binarized local structural feature, namely, the MCT, is less sensitive to such variations. We end this thesis with an application of the proposed alignment algorithm in cross-pose face recognition. The experiments demonstrate improved recognition performance compared with the AAM-based alignment. In addition, we propose a view-based pose normalization method, which solves the pose mismatch problem in regression-based cross-pose recognition.

Zusammenfassung

Die Analyse von Gesichtsaufnahmen ist ein bedeutendes Forschungsgebiet im Bereich digitaler Bild-und Mustererkennung. Dies ist im zunehmenden Bedarf einer Reihe von Anwendungsfeldern wie Sicherheit, Mensch-Maschine-Interaktion oder Multimedia-Indexierung begründet. Gesichtsausrichtung (d.h. Registrierung von Gesichtsaufnahmen) ist ein grundlegender aber essentieller Baustein im Rahmen der Gesichtsanalyse und hat große Auswirkungen auf die Robustheit und Qualität der darauf aufbauenden Folgeprozesse. Das Ziel der Gesichtsausrichtung ist es, hervorstechende Merkmale wie Augenwinkel oder Nasenspitzen in Gesichtsaufnahmen zu lokalisieren. Diese lokalisierten Merkmale können verwendet werden, um weitere aussagekräftige örtliche Merkmale in der unmittelbaren Umgebung zu extrahieren oder um Gesichtsaufnahmen zu Vergleichszwecken in ein standardisiertes Koordinatensystem zu übertragen.

Mit der Erfindung von Active shape model (ASM) [CT92] und Active appearance model (AAM) [CET98a] begann die Ära der Gesichtsausrichtung mittels statistisch verformbarer Modelle. Diese Modelle interpretieren Gesichtsaufnahmen mittels je eines Form- und eines Texturmodells, welche jeweils ein gewisses Maß an Verformbarkeit zulassen. Die durch Modellanpassung gewonnenen Informationen über die geometrischen Formen können in verschiedenen Anwendungen genutzt werden, so z.B. Gesichtsausdruckerkennug [LCK$^+$10] oder Mensch-Computer-Interaktion [AZC$^+$08]. Durch Bewegungen der Gesichtmuskeln zur Erzeugung von Gesichtsausdrücken, wie z.B. einem Lächeln, bieten relative Verformungen einer Gesichtspartie eindeutige, charakteristische Hinweise, aus denen sich Schlussfolgerungen über die Art des Gesichtsausdrucks ableiten lassen. Strukturen um Augen und Nase herum können zusätzliche Informationen (z.B. über Hautfalten) liefern, was die Leistung einer Ausdruckerkennung weiter verbessert. Die somit gewonnene Gesichtsform kann auch zur Bestimmung der dreidimensionalen Kopfdrehungen aus dem 2D-Bild mittels des POSIT-Algorithmus [DD95] verwendet werden, wobei ein generisches 3D-Gesichtsmodell für iterative Berechnungen der Parameter von Kopfdrehungen benötigt wird. Die Information von Kopforientierung in Kombination mit bestimmten Gesichtsbewegungen kann als effektive Interaktionsschnittstelle zur Handhabung von Viewport und Figuren in Computerspielen dienen.

Registrierung von Gesichtsaufnahmen gilt als besonders heikler Aspekt in der Gesichtserkennung [ES09]. Frühe Untersuchungen bauen auf eine lineare Transformation, indem sie von den Koordinaten der Augenmitte ausgehen um sicherzustellen, dass die Augen innerhalb des ausgerichteten Rahmens an den standardisierten Stellen positioniert werden. Diese Registrierungsmethode wird

üblicherweise in der frontalen oder der ansichtsbasierten Gesichtserkennung verwendet. Für Cross-Pose-Gesichtserkennung konnte Gao *et al.* [GES09] jedoch zeigen, dass Registrierung, die auf linearer Transformation basiert, nur dürftige Ergebnisse erzielt. Vorgeschlagen wird daher eine nichtlineare, verzerrungsbasierte Registrierung um die Kopfdrehung standardkonform zu normalisieren, wobei die Verzerrung anhand der mittels AAM-Anpassung lokalisierten Gesichtsmerkmale definiert wird. Die Cross-Pose-Gesichtserkennungsrate wird durch Nutzung dieser Normalisierung von Kopfdrehungen signifikant erhöht.

Trotz des breiten Anwendungsfeldes stellt die Gesichterkennung nach wie vor eine grosse Herausforderung dar. Zusätzlich zu den üblichen Aspekten im Bereich Computer-Vision wie Lichtbedingungen, Verdeckung oder Bildqualität kommen erschwerende Schlüsselfaktoren wie Ausdrucksverformung und geometrische Strukturen verschiedener Individuen hinzu. Aus dem Internet gesammelte Beispielbilder in Abbildung 1.1 demonstrieren die Bandbreite möglicher Erscheinungsformen tatsächlicher Gesichtsaufnahmen.

Um eine ausreichenden Anzahl an Variationen tatsächlich vorkommender Gesichtsaufnahmen bewältigen zu können, bedürfen generative Erscheinungsmodelle zunächst einer ausreichenden Menge an Trainingsdaten um die Bandbreite möglicher Variationen abdecken zu können. Dies führt wiederum zu einer übergrossen Anzahl von Variationsmodi, was sich in einem Modell mit einer enormen Menge an Parametern widerspiegelt. Innerhalb eines grossen Parameterraumes nach optimalen Parametern zu suchen ist keineswegs trivial, da es rechnerisch ineffizient und anfällig für lokale Optima ist. Des Weiteren kann nicht garantiert werden, dass wenn das Ziel der Modellanpassung im Sinne kleinster Quadrate definiert ist, das globale Optimum der Zielfunktion genau mit den tatsächlichen Formparametern übereinstimmt. Andererseits sind die durch Poseveränderungen und lokale Verformungen verursachten Variationen des Erscheinungsbildes nichtlinear, weshalb die Modellierung der Variationen mit einem linearen Modell nicht ausreicht. Es kommt häufig vor, dass das Modell nicht in der Lage ist, bestimmte fotometrische Signale zu erklären, was einen hohen Rekonstruktionsrückstand verursacht. In [RPG00] werden Lösungen mittels nichtlinearer generativer Modelle vorgeschlagen, deren Komplexität allerdings signifikant erhöht ist, was wiederum die Effizienz der Modellanpassung zunichte macht.

Die Untersuchung in dieser Arbeit versucht robuste und effektive Modellrepräsentationen zu entwickeln, welche die Nichtlinearität möglicher Variationen im Erscheinungsbild abdeckt. Die diskriminative Modellierung von Gesichtsabbildungen ist generativen Modellen aufgrund folgender Faktoren überlegen: (a) Anders als bei der Anpassung mittels Synthesestrategie beim AAM, wo hochdimensionale Erscheinungsparameter auf optimale Synthese durchsucht werden, schliesst das diskriminative Erscheinungsmodell nicht Parameterwiederherstellung von Texturen zum Zwecke der Texturinstanzgenerierung ein. Stattdessen

wird das gelernte Erscheinungsmodell als Kostenfunktion zum Ausrichten von Gesichtsaufnahmen angesehen, in welcher lediglich die niedrigdimensionalen Formparameter durchsucht werden. Daraus resultiert eine effiziente Modellanpassung. (b) Die Nichtlinearität der Erscheinungsvariationen kann in diskriminativen Modellen in den Kostenfunktionen modelliert werden. Obwohl dies den Berechnungsaufwand erhöhen kann, bleibt die Modellkomplexität gleich. (c) Das Hauptproblem im Zusammenhang mit diskriminativer Erscheinungsmodellierung ist, eine Kostenfunktion für die Bewertung der Ausrichtung zu erlernen. Im Idealfall stellt der Lernprozess sicher, dass das globale Optimum sich an der gewünschten Stelle befindet, also bei den tatsächlichen Formparametern. Diese Bewertungsfunktion wird erlernt durch Minimierung einer geeigneten Verlustfunktion, die über die Trainingsdaten definiert wurde. Optional kann die Bewertungsfunktion auch beschränkt werden, um Konvexität zu erzwingen, was zu einer glätteren Ergebniskarte mit einer geringeren Anzahl von lokalen Extrema führt. Ein lokaler optimiererbasierter Modellanpassungsalgorithmus kann letztendlich von diesen vorteilhaften Eigenschaften der erlernten Kostenfunktion profitieren. (d) Grundsätzlich haben diskriminative Modelle eine bessere Generalisierungsfähigkeit bei ungesichteten Daten als generative Modelle, da sie sich nicht auf Vermutungen der Datenverteilung stützen, welche bei ungesichteten Daten naturgemäss nicht vorherzusagen ist.

Es folgt eine kurze Erklärung der vorgeschlagenen diskriminativen Erscheinungsmodelle für robuste Gesichtsausrichtung.

Die Verwendung lokaler gradientenbasierter Merkmale für robuste Erscheinungsmodellierung wird vorgeschlagen. Diese Merkmale vergleichen Pixelwerte sowohl in der Umgebung als auch weiter entfernt. Im Falle des Vergleichs mit der unmittelbaren Umgebung werden lokale Strukturinformationen erfasst. Diese stellen die Existenz von Merkmalen wie Ecken, Rändern oder Strukturen dar, welche wiederum bestimmte Gesichtregionen repräsentieren können, so z.B. Talstrukturen für Pupillen oder Nasenlöcher. Um analytische Algorithmen zur Modellanpassung abzuleiten, wird eine nichtbinarisierte Census-Transformation für die Extrahierung derartiger lokaler Strukturinformationen vorgeschlagen. Ein Merkmalsvektor für eine Pixelposition wird extrahiert, indem man von dessen Intensitätswert den Durchschnittswert der Umgebung subtrahiert. Der Vergleich lokaler Intensität gewährleistet eine beleuchtungsinvariante Repräsentation von Merkmalen, da die lokale Texturstruktur unter verschiedensten Lichtbedingungen bei lokaler mittelwertfreier Normalisierung erhalten bleibt. Zur Unterscheidbarkeit von der herkömmlichen Census-Transformation wird der Begriff Pseudo-Census-Transformation gewählt.

Aus Effizienzgründen wird ausserdem das Binärmustermerkmal Modified Census Transform (MCT) verwendet. Ein analytischer Anpassungsalgorithmus für MCT-basierte Erscheinungsmodelle ist nicht ableitbar. MCT, auch als beleuch-

tungsinvariantes Merkmal bekannt, ist allerdings weniger informativ aufgrund des binarisierungsbedingten Informationsverlustes.

Das lokale Strukturmerkmal ist effizient; es mangelt ihr aber an semantischer Bedeutung um die Qualität der Ausrichtung zu bewerten. Das Merkmal berechnet die Pixelintensitätsunterschiede aus der Entfernung und wählt sinnvolle Merkmale aufgrund der Korrelation aus. Die semantische Bedeutung der ausgewählten Merkmale kann interpretiert werden als "der Intensitätsunterschied zwischen den Augenmitten ist niedrig" oder "die Pixelintensität der Augenbraue ist niedriger als die der Wange". Um die Auswirkungen von Bildrauschen zu reduzieren, kommt ein Quantisierungsprozess für das Merkmal der Intensitätsunterschiede zur Anwendung.

Verschiedene Maschinenlerntechniken für das Erlernen diskriminativer Erscheinungsmodelle wurden untersucht. Diese sind Klassifizierungsmodell, Rankingmodell, und Regressionsmodell.

Das klassifizierungsbasierte Modell betrachtet Gesichtsausrichtung als binäres Klassifikationsproblem. Es unterscheidet zwischen korrekten und falschen Ausrichtungen. Während des Trainings korrespondieren korrekte Ausrichtungen mit definierten Formen und falsche Ausrichtungen werden erzeugt, indem die Ground-Truth-Formen zufällig durcheinander gebracht werden. Das Erscheinungsmodell wird mit Hilfe der aus einer formfreien Textur extrahierten Merkmale trainiert und das Modell-Training verwendet ein Boosting-Framework, bei dem ein Bestand von charakteristischen Merkmalen als schwache Klassifikatoren ausgewählt wird. Die Bewertungsfunktion der kombinierten, starken Klassifikatoren wird als Kostenfunktion für die Gesichtsausrichtung genutzt, wobei eine Methode zur Optimierung des Verlaufsanstiegs zum Maximieren der Zielfunktion verwendet wird.

Das klassifizierungsbasierte Modelllernen leidet unter dem Problem unausgewogener Trainingsdaten. Da negative Proben ausserdem in der Klassifizierungsverlustfunktion nicht unterschieden werden, ist die erlernte Kostenfunktion unter Umständen nicht glatt genug, wenn die Ausrichtung zu sehr von der tatsächlichen Form abweicht. Dies macht die Verwendung des Anpassungsalgorithmus schwieriger, wenn die Anpassung zum globalen Optimum geführt werden soll. In diesem Sinne wird anstatt des Feststellens der Korrektheit der Ausrichtung ein rankingbasiertes Modell gebaut, welches die Halbordnung der Ausrichtung feststellt. Das erlernte Erscheinungsmodell trifft eine Vorhersage über bevorzugte Ausrichtungspaare. Da das Training auf Paarausrichtungsdaten basiert und falsche Ausrichtungen auch durch erhöhte Perturbation der Formen in die selbe Richtung gepaart werden können, wird das Problem der unausgewogenen Daten gelöst und die erlernte Funktion wird aufgrund der zusätzlichen Beschränkungen glätter.

Das Regressionsmodell ist eine direkte Methode um sich der gewünschten Kostenfunktion für die Ausrichtung zu nähern. Insbesondere wird vorgeschlagen, ein Ensemble von Regressionsbäumen zu verwenden, wobei jeder Baum trainiert wird, den Regressionsverlust zu reduzieren. Diese Methode ist als Gradientenverstärkende Regressionsbäume (Gradient Boosting Regression Trees) bekannt und wird im Allgemeinen zur Annäherung von Ranking-Funktionen verwendet. Um die Abschätzungsvarianz des Regressionsmodells weiter zu reduzieren, wird Random Forests eingesetzt um das gradientenverstärkende Lernen zu initialisieren. Wegen der Baumstruktur im Modell ist die korrespondierende Kostenfunktion nicht ableitbar. Für die Modellanpassung wird eine effiziente, direkte Suchmethode verwendet.

Im Folgenden werden die Beiträge dieser Arbeit aufgezählt:

Vorgeschlagen werden Merkmalsrepräsentationen zum Bau von Erscheinungsmodellen, die robust in Bezug auf Beleuchtungsänderungen sind. Ausserdem sind, bedingt durch die Lokalität der vorgeschlagenen Merkmale, die gelernten Modelle in der Lage, bis zu einem gewissen Grad mit Bildrauschen und partieller Verdeckung umzugehen. Der Rechenaufwand zur Extraktion der vorgeschlagenen Merkmale ist darüber hinaus gering, was den effizienten Ausrichtungsalgorithmus für Echtzeitsysteme geeignet macht. Umfangreiche Experimente mit verschiedenen Datenbeständen zeigen, dass die vorgeschlagenen Merkmalsrepräsentation robuster in Bezug auf die Ausrichtung ist, als bei lokalen, regionsbasierten Merkmalen wie z.B. Haar-Wavelets. Das PCT-Merkmal verbessert die Fähigkeit zur Generalisierung ungesichteter Testdaten um ca. 13%. Eine weitere vorteilhafte Eigenschaft des PCT-Merkmals ist der niedrige Konfigurationsraum, der eine schnelle Trainingsprozedur verglichen mit einem Modelltraining mittels Haar-Merkmalen ermöglicht.

Es wurden intensive Untersuchungen über den Aufbau diskriminativer Erscheinungsmodelle für Gesichtsausrichtung aus drei verschiedenen Blickwinkeln von Maschinen-Lernproblemen durchgeführt. Das erste Modell betrachtet Ausrichtung als Klassifizierung, wobei korrekte Ausrichtungen als Positiv- und falsche Ausrichtungen als Negativproben gewertet werden. Das zweite Modell erlernt die partielle Anordnung von Ausrichtungen, basierend auf einem Lernproblem von Ranking während das dritte Modell die beschränkte Anordnung von Ausrichtungen, basierend auf Regression, erlernt. Die Optimierung der erlernten Bewertungsfunktionen basiert auf einem Gradientenverfahren oder direkten Suchmethoden. Experimentelle Ergebnisse zeigen, dass die Ausrichtungsleistung von diskriminativen Erscheinungsmodellen gegenüber dem generativen Erscheinungsmodell signifikant erhöht ist. Die Erscheinungsmodelle, die auf Ranking und Regression basieren, sind den klassifikationsbasierten Modellen wegen ihrer gesteigerten Glätte in den Bewertungsfunktionen überlegen. Des Weiteren erreicht das regressionsbasierte Erscheinungsmodell, das mit einem Ensemble von Regressionsbäumen arbeitet, die beste Leistung in dieser Untersuchung.

Um die Eigenschaften des vorgeschlagenen diskriminativen Erscheinungsmodells weiter zu analysieren, wurde die Ausrichtungsrobustheit unter verschiedenen Bildbedingungen wie Bildrauschen, teilweise Verdeckung und Beleuchtung evaluiert. Durch diese Experimente stellte sich heraus, dass die vorgeschlagene Merkmals- und Erscheinungsmodellierung robust in Bezug auf diese Störfaktoren ist. Insbesondere hat die Anpassung mittels regressionsbasierter Erscheinungsmodelle auch dann noch eine respektable Konvergenzrate, wenn die Bilder teilweise verdeckt oder von Bildrauschen betroffen sind. Weiterhin stellte sich heraus, dass das RAPID-Merkmal, obwohl es die besten Ergebnisse im Benchmark-Datenbestand mit mässigen Beleuchtungsunterschieden erzielte, weniger robust war als das MCT-Merkmal, wenn mit extremen Lichtverhältnissen, wie in der erweiterten YaleB-Datenbank [LHK05a] präsentiert, getestet wurde.

Als Anwendung für die vorgeschlagenen diskriminativen Erscheinungsmodelle wird die Ausrichtung für Normalisierung von Kopfdrehungen in der Cross-Pose-Gesichtserkennung verwendet. Die experimentellen Ergebnisse zeigen, dass die verbesserten Ausrichtungsergebnisse die Erkennungsleistung verbessern. Darüber hinaus wird die Cross-Pose-Gesichtserkennung durch Verwendung Partial Least Squares (PLS) zum Erlernen eines latenten Raums erweitert, der die Korrelation zwischen verschiedenen Ansichten maximiert. Eine ansichtsbasierte Strategie der Normalisierung von Kopfdrehungen wird vorgeschlagen, was die Schwäche der Arbeit von [FES12] abmildert, bei der eine separate und präzise Poseschätzung erforderlich ist. Dadurch erhält das PLS-basierte Framework eine nutzbare Lösung in realen Anwendungen.

Contents

List of Figures

List of Tables

List of Abbreviations

AAM Active appearance model

ASM Active shape model

BAM Boosted appearance model

DAM Discriminative appearance model

GBRT Gradient boosting regression trees

LBP Local binary patterns

LDA Linear discriminant analysis

MAP Maximum a posteriori

MCT Modified census transform

PCA Principal component analysis

PCT Pseudo census transform

PDM Point distribution model

PLS Partial least squares

RAM Ranking appearance model

RAPID RAndom pixel intensity differences

REAM REgression appearance model

RF Random forests

SIFT Scale-invariant feature transform

SVM Support vector machine

1 Introduction

Analyzing facial images is an important task in the research domains of computer vision and pattern recognition for its increasing demand in application fields such as security, entertainment, human-machine interaction, and multimedia indexing. Face alignment (a.k.a. facial image registration) is an early yet essential building block in facial image analysis, which has a crucial impact on the robustness and quality of the later processes. The objective of face alignment is to localize a set of salient feature points, such as eye corners and nose tips, in facial images. The localized feature points can be either used for extracting meaningful local features around or transferring facial images to a canonical coordinate system for comparison purposes.

The invention of the ASM [CT92] and the AAM [CET98a] starts the era of aligning face images using statistical deformable models. The models interpret face images with a shape model and a texture model, which both allow certain deformation. The geometric shape information obtained after model fitting can be employed in many applications such as expression recognition [LCK$^+$10] or human-computer interaction (HCI) [AZC$^+$08]. As facial muscles deform, when posing facial expressions, such as smiling, relative deformations of face shape provide straightforward but distinctive hints for inferring the existence and type of the facial expression. Texture features around eyes or nose can also provide additional information, such as skin wrinkles, which further improve the performance of an expression recognition system. The recovered face shape can also be used for estimating the 3D head pose orientation from 2D images with the POSIT algorithm [DD95], where a generic 3D face model is needed for iterative calculation of the pose parameters. Pose information combined with specific facial actions can be an effective interaction interface for manipulating viewport and characters in computer games.

Face image registration is known for its cruciality in face recognition [ES09]. Early studies rely on a linear transform using the coordinates of eye centers, which ensures that the eyes are positioned at the canonical locations in the aligned frame. This registration method is commonly used in the frontal or view-based face recognition. However, for cross-pose face recognition, Gao *et al.* [GES09] have shown that a linear transform based registration achieves poor recognition results. A nonlinear warping-based registration is proposed for normalizing the face pose into a canonical one, where the warping is defined by

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

Figure 1.1: Sample facial images of different variations, images are selected from the Labeled Face Parts in the Wild (LFPW) database [BJKK11]. (a) Facial occlusion due to sunglasses or hair; (b) Pose (self-occlusion); (c) Facial expression; (d) Low quality, noisy image.

the localized feature points using AAM fitting. The cross-pose face recognition rates are significantly improved using this pose normalization.

1.1 Motivation

Despite the broad applications, face alignment is still a very challenging task. In addition to common factors in computer vision such as lighting conditions, occlusion, and image quality, the expression deformation and geometric structures of different identities are also key factors that make the task difficult. Sample images collected from the internet are shown in Figure 1.1, which demonstrate the possible appearance variations in the real-world facial images.

To be able to cope with enough appearance variations that are presented in real-world facial images, generative appearance models, *e.g.* AAM, require sufficient training data for covering the distribution of appearance variations. This results in too many variation modes, which corresponds to a model with a large number of parameters. Searching for the optimal parameters in a large parameter space is not a trivial task, which is computationally inefficient and prone to local optima. In addition, as the objective for model fitting is defined in a least-squares sense, the global optimum of the objective function is not guaranteed to be located exactly at the true shape parameters. On the other hand, the appearance variations caused by pose changes and local deformations are

nonlinear, therefore, modeling the variations with a linear model is not sufficient. It is often the case that the model cannot explain certain photometric signals and results in a high reconstruction residue. Solutions using nonlinear generative models are proposed in [RPG00], yet the complexity of the model is significantly increased, thus the efficiency of the model fitting is sacrificed.

The study in this thesis aims at developing robust and effective model representations for covering the non-linearity of the possible appearance variations. A discriminative appearance modeling is proposed and intensively studied in this thesis. The discriminative modeling of facial appearance is superior to generative models due to following factors:

- Unlike the fitting by the synthesis strategy in the AAM, where the high dimensional appearance parameters are searched for optimal synthesis, the discriminative appearance model does not involve parameter recovering of the texture for generating texture instances. Instead, the learned appearance model is considered as a cost function for aligning face images, in which only the low dimensional shape parameters are searched. This results in an efficient model fitting.
- The non-linearity in appearance variations can be modeled in the cost functions, in the discriminative models. Although this may increase the computational cost, the model complexity remains the same.
- The key issue in the discriminative appearance modeling is to learn a cost function for scoring the alignments. Ideally, the learning process ensures that the global optimum should be located at the desired place, *i.e.*, the true shape parameters. This scoring function is learned by minimizing a proper loss function defined over the training data. Optionally, the scoring function can be constrained to enforce convexity, leading to a smoother response map with few local extrema. A local optimizer-based model fitting algorithm can eventually benefit from those favorable properties of the learned cost function.
- In general, discriminative models have better generalization ability on unseen data than generative models as they do not rely on any assumptions about the data distribution, which is unpredictable on unseen data.

The feature used for appearance modeling also plays an important role. Using raw pixels for texture modeling is suboptimal, as it has difficulties in handling nonlinear illumination effects and occlusion. We propose to use local gradient features for the appearance modeling as those feature representations are less sensitive to illumination changes, and have been applied in many other systems for tackling the illumination problem. The locality property of the features also enables them in handling local occlusions to some extent. In addition, as the local gradient features are simple to compute, the computational cost is low.

Benchmarking the algorithms developed for face alignment is important for understanding the abilities of handling specific aspects of the problem. It is also interesting to measure how a model and its corresponding fitting algorithm generalize on unseen data. Unfortunately, there is no common benchmark data set so far, which contains sufficient variations for evaluating both robustness and generalization ability of alignment algorithms. Collecting data sets for this purpose is an expensive task as labeling facial landmarks on face images is a tedious and time-consuming work. There are some databases available so far, such as the IMM database [SEL03] and the XM2VTS [MMK+99] database, which are annotated with 58 and 68 landmarks, respectively. However, the number of subjects in the IMM database is rather limited, while the variation in the XM2VTS database is not sufficient.

To this end, we provide a common data set and proper metrics for evaluating different face alignment systems in this thesis. We focus on aligning faces in 2D still images, which are captured with monocular cameras. A data set with the images selected from four different public face databases is proposed. The collection includes different variations in pose, illumination, expression, occlusion, etc. These variations enable us to analyze generalization capabilities of the proposed approach on different levels of variation.

1.2 Approach

We briefly explain the proposed discriminative appearance models for robust face alignment in the following subsections.

1.2.1 Robust Feature Representation

We propose to use local gradient-based features for robust appearance modeling. The features compare pixel values either in a neighbourhood or at a distance. In case of comparing in a local neighbourhood, local structural information is captured. The local structural information presents the existence of features like corner, edge, or structures, which may distinctively depict certain facial regions, *e.g.* a valley structure for pupils or nostrils. In order to derive analytical algorithms for model fitting, an unbinarized census transform is proposed for extracting such local structural information. A feature vector for one pixel location is extracted by subtracting the intensity values in a neighbourhood by their average. The local intensity comparison results in an illumination robust feature representation as the local texture structure is preserved under various lighting conditions with a local zero mean normalization. We name this

transform as Pseudo Census Transform (PCT), in order to distinguish it from the original census transform.

For efficiency reason, the binary pattern feature, Modified Census Transform (MCT), is also used. An analytical fitting algorithm for MCT-based appearance model is not derivable. MCT is also known as an illumination invariant feature, yet it is less informative due to the information loss in binarization.

The local structural feature is efficient yet lacks semantic meaning for assessing the quality of alignments. We present a novel RAndom Pixel Intensity Difference (RAPID) feature for appearance modeling, which also provides semantic meaning for assessing correct alignments. The feature computes the pixel intensity differences at a distance, and the useful features are selected based on correlation. The semantic meaning of these selected features can be interpreted as "the intensity difference between the eye centers is low" or "the pixel intensity on the eyebrow is lower than on the cheek". To reduce the effects of image noise, a quantization process for the intensity difference features is applied.

1.2.2 Discriminative Modeling of Appearance

We investigated several machine learning techniques for learning discriminative appearance models. They are:

- Classification Model. The classification-based model considers face alignment as a binary classification problem. It distinguishes between correct alignments and incorrect alignments. During training, correct alignments correspond to annotated shapes, while incorrect alignments are generated by randomly perturbing the ground truth shapes. The appearance model is trained using the features extracted from a shape-free texture (cf. Figure 3.7(c)), and the model training adopts a boosting framework, where a set of distinctive features are selected as weak classifiers. The scoring function of the combined strong classifier is used as the cost function for face alignment, in which a gradient ascent optimization method is used for maximizing the objective function.
- Ranking Model. The classification based model learning suffers from the problem of imbalanced training data. Moreover, as the negative samples are not distinguished in the classification loss function, the learned cost function may not be smooth enough, when the alignment is far from the true shape. This makes the fitting algorithm difficult in guiding the fitting towards the global optimum. In this sense, instead of learning the correctness of alignments, we build a ranking-based model, which learns the partial orders of alignments. The learned appearance model predicts the preference in alignment pairs. As the training is based on pairs of

alignment data and incorrect alignments can also be paired by increasing shape perturbation in the same direction, the imbalanced data problem is resolved, and the learned cost function will be smoother due to the additional constraints.

- Regression Model. The regression model is a straightforward method for approximating the desired cost function for alignment. In particular, we propose to use an ensemble of regression trees, where each tree is trained aiming at decreasing the regression loss in the gradient direction of the regression loss. The method is known as the gradient boosting regression trees, which is commonly applied for approximating ranking functions. In order to further reduce the estimation variance of the regression model, we adopt Random Forests to initialize the gradient boosting learning. Due to the tree structure in the model, the corresponding cost function is not derivable. We use an efficient direct search method for model fitting.

1.3 Contributions

The contributions of this thesis are listed as follows:

- We propose robust feature representations for building appearance models that are robust against illumination changes. In addition, due to the locality of the proposed features, the learned models are able to handle image noise and partial occlusion to some degree. The computational cost for extracting the proposed features are also low, resulting in efficient alignment algorithms that are applicable for real-time systems. Extensive experiments are conducted on different data sets. The results show that the proposed feature representation is more robust for alignment than other local region based features such as Haar-wavelets. The PCT feature improves the generalization ability on unseen testing data by about 13%. An additional favorable property of the PCT feature is the low configuration space, which results in a fast training procedure compared to the model training with Haar-features.
- We conducted extensive studies on building discriminative appearance models for face alignment in three different perspectives of machine learning problems. The first model considers alignment as classification, in which correct alignments are regarded as positive samples and incorrect alignments as negative samples. The second model learns the partial ordering of alignments, based on a learning to rank problem; while the third model learns the total constrained ordering of alignments, based on regression. The optimization of the learned score functions is based on

gradient ascent or direct search method. Experimental results show that the alignment performance of discriminative appearance models is significantly improved compared to the generative appearance model, *e.g.* AAM. The appearance models based on ranking and regression are superior to the classification-based models due to the increased smoothness in the score functions. Furthermore, the regression-based appearance model, which uses an ensemble of regression trees, achieves the best performance in this study. We also show in the experiments that the regression-based appearance models outperform two state-of-the-art discriminative face alignment models.

- To further analyze the properties of the proposed discriminative appearance models, we thoroughly evaluate the alignment robustness under various imaging conditions, such as image noise, partial occlusion, and lighting. Through the experiments, we find out that the proposed feature and appearance modeling are robust against these confounding factors. In particular, fitting with regression based appearance models still has a decent convergence rate, when the images are partially occluded or corrupted by image noise. In addition, we find out that although the RAPID feature achieves best results on the benchmarking data set with moderate illumination changes, it is less robust than the MCT feature, when tested under extreme lighting conditions as presented in the extended YaleB database [LHK05a].
- As an application for the proposed discriminative appearance models, we apply the alignment for pose normalization in cross-pose face recognition. The experimental results show that the improved alignment results enhance recognition performance. In addition, we extend the cross-pose face recognition by using partial least squares (PLS) for learning a latent space which maximizes correlation between different views. A view-based pose normalization strategy is proposed, which mitigates the weakness in the work by Fischer *et al.* [FES12], where a discrete and precise pose estimation is required. This provides the PLS-based framework an applicable solution in real-world applications.

1.4 Outline

This thesis is organized as follows:

In Chapter 2, an overview of the related work is given. Well-known generic face alignment algorithms are reviewed. The important statistical deformable model, active appearance model, and many of its extended variants are surveyed

as closely related works to the topic of this thesis. Some interesting application systems, which are based on the AAMs, are briefly presented.

In Chapter 3, the classification-based appearance model is introduced. The 2D shape model, which is used throughout this thesis, is first explained. Then, the training of a classification-based appearance model is presented. Finally, the experimental results and the comparison to AAM and other feature representations are given.

In Chapter 4, the ranking-based appearance model is described. A motivation of formulating the appearance model learning as a learning to rank problem is first given at the beginning of this chapter. The details of the ranking model learning and training data generation are given in the second part of this chapter. We present the experimental results for assessing the effectiveness of the ranking-based model in the third part of this chapter.

In Chapter 5, the approach based on the ensemble of regression trees are presented. We first introduce the gradient boosting regression trees for learning the regression-based appearance model. Afterwards, an initialization strategy based on Random Forests is proposed to reduce the estimation variance and speed up the convergence of boosting. The chapter ends with an experimental comparison between the proposed models.

In Chapter 6, an effective appearance model based on random pixel intensity differences is described. In this chapter, the proposed feature extraction and selection is explained first. The details of parameter selection and impact factors in building robust appearance models are discussed. A comparative study on the models presented in the previous chapters is conducted.

In Chapter 7, robustness analysis on different aspects are systematically discussed and an application of the proposed models for face alignment is given. The alignment robustness study includes the impacts of image noise, occlusion, and lighting. We explain the experimental configuration and results in detail. In order to further assess the effectiveness of the proposed discriminative appearance models, we present the details of applying the models in cross-pose face recognition.

In Chapter 8, we discuss the outcomes of the thesis and give final concluding remarks afterwards.

2 Related Work

An overview of related works, which have been conducted on face alignment and their applications, is given in this chapter. The chapter consists of three sections. In Section 2.1, we review generic face alignment algorithms. We focus on deformable model-based methods in Section 2.2, in which we present a conventional model and its extensions for improving alignment efficiency and robustness. We explicitly compare the generative appearance models and discriminative appearance models in this section. Finally, some application systems based on deformable model fitting are described in Section 2.3.

2.1 Generic Facial Image Alignment Methods

As stated in Chapter 1, the goal of facial image alignment is to find a set of corresponding anchor points, with which an image is registered by applying a linear or nonlinear transform. The simplest alignment relies on localizing eye centers, as they are less deformed due to variations in identity or expression. The alignment fixes the coordinates of eye centers in the transformed image. However, using more anchor points provides more accurate recognition performance [GES09], yet localizing other points such as mouth corners or face boundaries is more challenging. In this section, we review some generic approaches for localizing facial feature points.

There are some related surveys, yet most of them are dedicated for eye localization [HJ10, STCZ12]. However, most of the reviewed approaches can be applied for localizing other facial features as well. According to [STCZ12], we briefly classify the approaches into three categories based on the features used for modeling.

Characteristics of facial features The inherent geometries and contrast of individual facial components are exploited in these approaches. Some contextual information around or between facial regions might also be useful for localization. These approaches are simple and straightforward. However, the reliability of localization is very sensitive to imaging conditions.

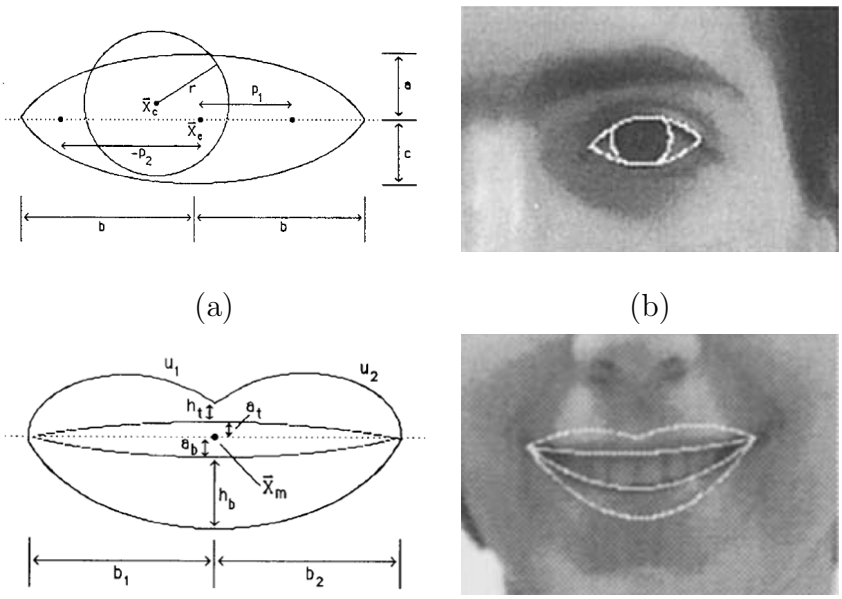
Statistical feature-based model In this kind of approach, useful features are extracted from the appearance of facial components, from which statistical models are learned using a large set of training data. The learned models are able to cope with large variability of facial appearances and imaging conditions.

Structural model This approach explores the spatial structure of individual facial components and (or) the geometrical regulation between each other. Usually, the structural information is combined with the statistical feature-based models to improve the stableness against various uncontrolled conditions.

2.1.1 Characteristics of Facial Features

These approaches explore the distinct inherent characteristics of individual facial components by themselves. Specific information such as shape and intensity contrast along or inside shape contour is used. A representative work of this approach is the deformable template [YHC92]. A parametrized deformable model is proposed which describes the shape of facial components using continuous mathematical formulations. Figure 2.1(a) shows a typical shape model of eyes containing eyelids and iris. The eyelids are represented by two arcs and the iris is represented by a circular shape. Similarly, a mouth shape model is shown in Figure 2.1(b). In addition, the authors also consider the relevant features such as peaks and valleys of intensity values defined inside the regions bounded by shapes. A set of energy functions are designed based on the relevant features. They fit the models to a testing image by searching the continuous parameter space and minimizing the overall energy functions. In practice, the shape models are carefully designed so that they are distinctive and also flexible enough to explain large shape variations. A good initialization is required as the energy functions are minimized using a local optimizer. A multi-stage searching scheme is introduced to handle the initialization problem, in which the weights for different energy functions change accordingly in different stages. In the early stages, the energy functions based on distinctive features (such as a valley feature for the pupil and a peak feature for the eye white) are more favorable for fast convergence. In the later stages, the energy functions based on edge features are more important for fine-tuning. Despite this improvement, the deformable templates still suffer from dependence on image quality and good initialization.

Instead of designing complicated shape models of facial components, some approaches directly use the intensity distribution patterns of *e.g.* eyes for localization. For open eyes, the intensity contrast between different eye components is strong, while the gray intensity at the eye center is usually much lower than the eye white and the other regions in the neighbourhood. Intensity patterns



(a) (b)

(c) (d)

Figure 2.1: Deformable template models for the eyes and mouth from Yuille *et al.* [YHC92].

like these are commonly used as the heuristic evidence for localization. Typical examples are projection function-based approaches [Kan73, ZG04] and iterative thresholding-based approaches [SYW96].

2.1.2 Statistical Feature-based Models

The aforementioned methods rely on intensity contrast of facial components, which are very sensitive to illumination changes. Alternatively, one could extract more reliable feature representation from the image patches of individual facial components. In addition, machine learning techniques are applied to select more discriminative features for building a statistical appearance model. The appearance model not only explains the contrast information, but also approximates the same information to some degree, which leads to a more robust technique.

Popular features are frequency based descriptors such as Haar wavelets, Gabor wavelets, and gradient-based descriptors such as Local binary patterns (LBP) [OPH96], Scale-invariant feature transform (SIFT) [Low04]. In fact,

those feature descriptors are known to be illumination invariant in the context of computer vision.

Usually the dimensionality of the feature descriptors is high and not all dimensions are discriminative enough for classification. Statistical methods are applied on top of the extracted features, in order to learn effective models from relatively few training samples. These methods reduce feature dimensions by learning low dimensional manifolds or selecting distinctive feature sets. The former variant corresponds to generative modeling in statistical learning, where the class conditional probability distribution of a facial component is estimated. Localization is based on checking the likelihood of a testing patch. A modular Principal component analysis (PCA)-based generative method is proposed in [MP97], in which a low dimensional modular subspace for each facial component is learned using PCA. A Gaussian model is estimated for approximating the class conditional distribution of positive patches. In [EZ06], Gaussian models of negative patches are also included and the prediction of a testing patch is based on computing the log-likelihood ratio.

The discriminative methods directly find a discriminant function for separating target facial patches and non-targets. The localization turns out to be a binary classification problem. Common classifiers such as support vector machines [Vap98] and Adaboost [FS97] are applied. The SVM-based models find weights for different feature dimensions. They provide good generalization capability with carefully selected kernel functions and corresponding parameters, which increase the cost on computation time and memory usage. The methods based on Adaboost select discriminative features increasingly and use them to build a more powerful classifier. In [VJ04], a coarse-to-fine strategy is proposed to learn cascades of boosted classifiers. The testing step is very efficient since each level of the cascaded classifiers consists of a linear combination of a few simple weak classifiers and only those candidates with a high likelihood will be passed to succeeding stages. This strategy is originally proposed for detecting human faces (also generic objects) in real-time [VJ04, FE04], an extension to facial feature detection / localization in the context of face region is straightforward.

2.1.3 Structural Models

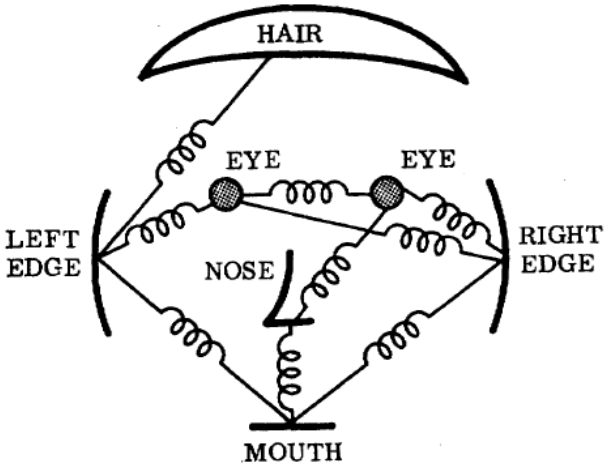
The appearance model of individual facial components ignores the spatial topologies between each other. The topological features are complementary to patch appearance and less affected by environment conditions. This information provides prior knowledge about the constellation of the facial components, which constrains the searching procedure for localization. Here, we discuss three representative methods for modeling the topological structure of facial features.

Pictorial structure model [FE73] uses a graph structure to model topological relationships between components of an object, *e.g.* face. Figure 2.2(a) shows an example of graph structure of a face model, with edges linked to each facial component. This model is used to localize facial components by simultaneously measuring the fitness of local parts and structural deformation in-between. The energy function for optimal matching of the model to a test image is defined as:

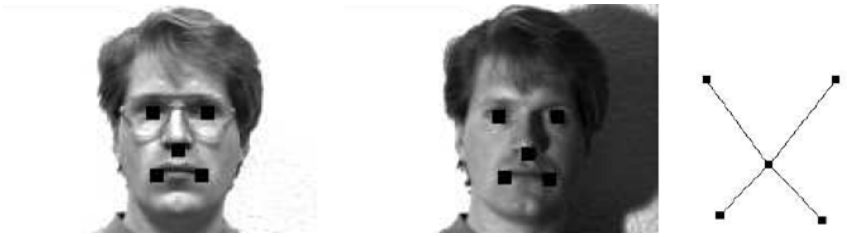
$$L^* = \arg\min_{L} \left(\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j)\in E} d_{ij}(l_i, l_j) \right), \tag{2.1}$$

where $m_i(l_i)$ defines the energy function for localizing component i at position l_i according to appearance model and $d_{ij}(l_i, l_j)$ defines the pairwise energy function, which measures the spatial structure confidence for component i and component j located at position l_i and l_j, respectively. The optimal configuration of component locations is found by minimizing the global energy function. An efficient solution for this optimization problem is proposed by Felzenswalb *et al.* in [FH05], in which the graph structure is simplified to a tree structure. The simplification converts the form of connections between components as a linear one rather than quadratic in the number of possible locations for each component, which eventually makes the optimization more efficient. Further graph-based or tree-based structure models are presented in [VMBP10, ZR12].

Instead of defining structure information based on human expertise, another representative method is ASM [CTCG95], which learns structure knowledge of an object from training data with statistical models. ASM describes the shape of an object using a set of landmark points. The coordinates of these landmark points are stacked in a fixed order, which forms a shape vector. The landmark points are manually annotated, usually on the contour of the facial components. A statistical shape model is built using the annotated training data. The model represents major shape variations and their ranges in the training data. ASM defines a generative profile model based on distribution of gradient information of an underlying landmark point, which measures the similarity of a landmark point at the current location. Searching a shape vector on a test image is implemented in an iterative manner. The similarity of the landmarks points is maximized first according to the profile models. The shape model is then applied to constrain the search so that the shape vector only deforms in the same way as presented in the training data. A coarse-to-fine strategy is presented, which improves the robustness and efficiency of the shape search. A probabilistic extension of ASM is presented in [ZGZ03], in which searching is formulated as a MAP estimation with Bayesian inference. Another important extension on ASM is AAM [CET98a], where the holistic texture of the facial region is modeled with PCA. A more detailed review of AAM and its variants is presented in Section 2.2, since the model presented in this thesis is also an extension to AAM.



(a)



(b)

Figure 2.2: Pictorial structure model for detecting a face object based on parts.

Apart from building structure models of face components explicitly, there are also some noticeable works, which apply structure information implicitly. Implicit shape model (ISM) [LLS06, LLS08] is a typical voting-based structure model, in which the shape model is not explicitly constructed but represented loosely in terms of a bag of patches. The key idea of ISM is to maintain a spatial occurrence distribution for each visual codeword such that it can not only be used for the representation of local appearances but cast votes for possible positions of the object center as well. The voting mechanism is less sensitive to partial occlusion and large appearance variations. Recently, Gall *et al.* extend the ISM framework to Hough forest [GL09], in which local codewords are generated randomly with the use of random forest. Dantone *et al.* [DGFG12] use Hough forest to build pose conditioned voting forests for robust head pose estimation and facial feature detection. Cootes *et al.* [CILS12] additionally apply a statistical shape model to find globally optimized structure configurations. Other approaches, which implicitly use shape models, can be found in [CWWS12]. These methods recover the shape of a test image by using a regression model

explicitly learned from the training data. The structural constraint of the shape variations is implicitly learned in the regression model.

2.2 Statistical Deformable Appearance Models

The models proposed in this thesis are based on AAM, which contains a statistical shape model and a holistic texture model. This section reviews AAM and its variants.

2.2.1 Active Appearance Models

AAM interprets face images with shape and texture. The shape is defined in the same way as in ASM, in which a set of landmark points located at the boundaries of facial components is used to describe the structure information. The texture depicts the pixel information inside the convex hull of a shape, which is represented by intensities or colors (cf. Figure 2.3(b)). Training an AAM requires a set of training images together with their corresponding manually labeled landmark points (cf. Figure 2.3(a)). A shape is represented by a shape vector, which contains n concatenated point vectors, (x_i, y_i). After pose normalization, a shape $\mathbf{s}$ can be represented with a linear shape model, in which the model bases are created by applying PCA,

$$\mathbf{s} = \mathbf{s}_0 + \mathbf{P}_s \mathbf{b}_s, \tag{2.2}$$

where $\mathbf{s}_0$ denotes the mean shape, $\mathbf{P}_s$ stores the eigenvectors from PCA, with the row vectors describing the modes of shape variation learned from training data. $\mathbf{b}_s$ denotes the corresponding shape parameter vector in the shape space. As with the shape vector, a texture vector $\mathbf{g}$ is also projected onto a texture PCA subspace,

$$\mathbf{g} = \mathbf{g}_0 + \mathbf{P}_g \mathbf{b}_g, \tag{2.3}$$

where $\mathbf{g}_0$ denotes the mean texture, $\mathbf{P}_g$ stores the eigenvectors from PCA, with the row vectors describing the modes of texture variation learned from training data. $\mathbf{b}_g$ corresponds to the texture parameters in the texture space. The correlation between shape and texture is decoupled by projecting the concatenated shape parameters and texture parameters onto an appearance PCA subspace

$\mathbf{Q}$. Eventually, the shape and texture of an input image can be represented with the joint appearance parameters,

$$\mathbf{s} = \mathbf{s}_0 + \mathbf{Q}_s\mathbf{c}, \quad \mathbf{g} = \mathbf{g}_0 + \mathbf{Q}_g\mathbf{c}, \tag{2.4}$$

where $\mathbf{c}$ is the appearance parameter vector, $\mathbf{Q}_s$ and $\mathbf{Q}_g$ are the corresponding parts of the base matrix for the joint subspace.

Fitting the model to a new image is defined as a least squares problem, in which difference between the synthesized model appearance and the warped image is minimized with respect to the model parameter $\mathbf{c}$,

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} \|\mathbf{g}_c - \mathbf{g}_m\|^2, \tag{2.5}$$

where $\mathbf{g}_c$ is the warped texture image, and $\mathbf{g}_m$ is the synthesized model image (cf. Figure 2.3). As the image warping is a nonlinear function of the shape parameters, the objective function to be minimized is nonlinear, which is difficult to solve. An iterative method based on gradient descent is applied for this optimization problem. As the image difference is defined in the model frame, Cootes *et al.* [CET98a] assume a linear relationship between the parameter updates $\delta\mathbf{c}$ and the image residues $\mathbf{r}(\mathbf{c}) = \mathbf{g}_c - \mathbf{g}_m$,

$$\delta\mathbf{c} = \mathbf{R}\mathbf{r}(\mathbf{c}). \tag{2.6}$$

The assumption leads to an efficient fitting algorithm, which avoids the time-consuming updating of Jacobian matrix $\mathbf{J} = \frac{\partial \mathbf{r}(\mathbf{c})}{\partial \mathbf{c}}$ in each iteration. The linear regression matrix $\mathbf{R}$ is precomputed by a multivariate linear regression. Figure 2.3(c) shows an example of a fitted model appearance superimposed on a facial image, where the original image is displayed to its right side.

2.2.2 AAM Extensions

Numerous extension studies on AAM have been carried out, aiming to improve the efficiency and robustness for modeling and fitting. We review the AAM variants regarding three aspects, *i.e.* appearance representation, fitting algorithms, and robustness handling.

2.2.2.1 Appearance Representation

The dimension of the texture vector is much higher than the dimension of the texture subspace, this indicates that the texture representation with pixel values contains spatial redundancy. Cootes *et al.* [CET98b] attempt to reduce the

(a) (b) (c)

Figure 2.3: AAM from Cootes *et al.* [CET98a]. (a) An annotated image; (b) A warped AAM texture; and (c) A synthesized model image and its original image.

redundancy by applying sub-sampling, in which only large values are preserved in the regression matrix. The assumption is, however, not always true, which sacrifices the fitting accuracy. In [WT99], the wavelet-based image compression technique is applied for building compact texture representation. Similarly, a wedgelet-based regression tree is applied in [DLSE04] for compressing the texture vector with a high compression rate.

The sub-sampling or compression techniques only remove redundant information but do not provide additional information to increase the representation power. The intensity based texture representation can be easily influenced by illumination changes. As the lighting changes have nonlinear effects on facial images, a single PCA-based linear texture subspace can not fully model the variations under complex conditions. A possible solution is to decompose the texture space into two independent subspaces [KGDL07]. One explains the variations in illumination and the other one contains the variation modes for identity. Gonzalez *et al.* [GlTFM$^+$07] decouple the appearance model into multiple subspaces, each corresponding to pose, expression, and identity variations. The decomposition enhances the representation power of the texture/appearance model. Another solution to the nonlinear lighting problem is based on explicit nonlinear modeling [CD05]. The Gaussian mixture model (GMM) and the nearest neighbour model (NNM) is used for approximating the nonlinear manifolds of the faces.

On the other hand, one can also enrich the texture representation with the use of additional image information such as color or edge features to obtain better discrimination. For example, Edwards *et al.* [ECT99] show that color texture representation in RGB space provides more discrimination than a single gray-channel model. Moreover, other local structure information, such as edge and gradient, is also helpful for improving the model fitting [SCT03, KnC06,

CT01b]. The combination of local structure and color channel achieves better performance as is shown in [KG05, SL03]. However, the multi-channel texture modeling requires more computational cost and storage for fitting, which is not ideal for real-time applications.

Facial expressions or lighting changes usually deform the local image regions. Cristinacce *et al.* [CC06] simplify the texture model with local patch models, which are centered at the landmark points. The local patches are concatenated together and jointly modeled with a shape model in a similar way as AAM [CET98a]. The objective function for fitting is based on maximum likelihood estimation with the shape model constraint. Wang *et al.* [WLC07] train local patch models independently in a discriminative way. However, the fitting is implemented with the Lucas-Kanade gradient-descent algorithm, where the local patch responses are approximated by an isotropic Gaussian, which eventually leads to an ASM-like fitting algorithm. Other extension works model the local patch responses with a full covariance Gaussian [WLC08], Gaussian Mixture Models (GMM) [SLC09a], and Kernel Density Estimation (KDE) [SLC09b], which further improve the fitting performance.

2.2.2.2 Fitting Algorithms

2.2.2.2.1 Generative Fitting AAM fitting in [CET98a] assumes a linear relationship between the parameter updates and the image residues. The assumption is not always correct and computational cost is still high. To improve the fitting efficiency, it is observed in [CET98b] that updating both shape and texture parameters in each iteration is unnecessary. A simpler updating strategy is proposed, where the shape parameters are directly estimated from the texture residuals. The texture parameters are not updated and the texture residual is defined as the difference between the warped texture and the model mean. The computational costs are thus reduced. Hou *et al.* propose a direct appearance model [HLZ01], which projects the texture residual onto a low-dimensional subspace. The resulting parameter vector is used for estimating the regression matrix. Since the dimension of a parameter vector is much smaller than the texture residual, the computational cost in evaluating the regression matrix R is significantly reduced. Another modification for efficient model fitting focuses on increasing convergence speed. In [DRL+06], the canonical correlation analysis (CCA) is used to capture the correlations between the shape and texture updates, which eventually results in an accurate and efficient model fitting.

The analytical derivations of AAM fitting are investigated in [MB04, GMB05, BM01] within the framework of Lucas-Kanade algorithm [BM04]. Due to the

difficulties in analytical derivation for the combined AAM, the independent AAM is considered. The objective function for model fitting is defined as:

$$\sum_{\mathbf{x}\in\mathbf{s}_0}\left[\mathbf{g}_0(\mathbf{x})+\sum_{j=1}^{m}\mathbf{g}_j(\mathbf{x}){b_g}^j-\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{b}_s;\mathbf{t}))\right]^2, \tag{2.7}$$

where $\mathbf{W}$ is the piecewise affine warping function with $\mathbf{b}_s$ and $\mathbf{t}$ as parameters. This problem can be solved efficiently by the inverse compositional (IC) [BGM03] algorithm. The project out (PO) version of the IC algorithm avoids updating the texture parameters, which results in an extremely efficient AAM fitting algorithm. However, the authors suggest using a less efficient simultaneous IC (SIC) algorithm for fitting a generic AAM to facial images of unseen subjects [GMB05].

A statistical formulation of AAM fitting is presented in [CT01a]. The fitting objective function is optimized with maximum a posteriori (MAP) estimation. The formulation enables additional constraints for mitigating the local extrema problem. In addition to shape prior, one could also impose prior information of landmark points if available. The objective function with shape prior and landmark prior is given as follows:

$$E(\mathbf{p})={\boldsymbol{\sigma}_r}^{-2}\mathbf{r}^\top\mathbf{r}+\mathbf{p}^\top({\mathbf{S}_\mathbf{p}}^{-1})+\mathbf{d}^\top{\mathbf{S}_\mathbf{X}}^{-1}\mathbf{d}, \tag{2.8}$$

where the residual $\mathbf{r}$ is considered as isotropic Gaussian with variance ${\boldsymbol{\sigma}_r}^2$. $\mathbf{S}_\mathbf{p}$ denotes the diagonal covariance matrix for the model parameter $\mathbf{p}$. $\mathbf{d}(\mathbf{p})=\mathbf{X}-\mathbf{X}_0$ is the distance between model point positions $\mathbf{X}$ and their true positions $\mathbf{X}_0$ given as priors, such as the eye positions. The diagonal covariance matrix $\mathbf{S}_\mathbf{X}$ denotes the uncertainty of the priors, where the corresponding items without priors are set to zero. With the direct guide of the priors, the fitting is less likely to get stuck into local extrema.

2.2.2.2.2 Discriminative Fitting Although the linear regression and analytic updating methods are greatly successful, the updating functions are intrinsically the approximated estimation to the nonlinear fitting procedure. The problem becomes serious, when the parameters move far away from the true place. In this case, the linear assumption of the relationship between the texture residual and the model parameters displacement does not hold anymore. Therefore, the warping function becomes incorrect. To this end, Saragih and Goecke [SG07] learn this relationship by a nonlinear boosting procedure. The method trains a strong regressor for each model parameter:

$$F^k(\mathbf{g}_c)=\sum_{i=1}^{n_k}\alpha_i^k f_i^k(\mathbf{g}_c), \tag{2.9}$$

where F^k is the strong regressor composed of a number of n_k weak learners f_i^k with corresponding weights α_i^k. As a result, a more accurate estimation of the updating function is obtained.

The PCA-based modeling provides a compact and independent parameter model. However, it is not optimized in the aspect of discriminant for the model fitting, which makes the fitting prone to local minima and in the worst case, cannot ensure that the global minimum corresponds to the correct parameters. To this end, Nguyen *et al.* [NlTF08] learn a cost function by explicitly optimizing such that the local minima occur only at the places corresponding to the correct fitting parameters. The cost function to be learned is defined as:

$$E(\mathbf{d}, \mathbf{p}) = \mathbf{d}(\mathbf{f}(\mathbf{x}, \mathbf{p}))^\top \mathbf{A}\mathbf{d}(\mathbf{f}(\mathbf{x}, \mathbf{p})) + 2\mathbf{b}^\top \mathbf{d}(\mathbf{f}(\mathbf{x}, \mathbf{p})). \tag{2.10}$$

Here the symmetric matrix $\mathbf{A}$ and $\mathbf{b}$ are the parameters of the function, which have to be learned from training data. This function is the general form of the AAM fitting cost function. Nguyen *et al.* [NlTF08] manage to find the suitable $\mathbf{A}$ and $\mathbf{b}$ by imposing constraints on global minima and direction of gradients. The proposed cost function learning implicitly learns a new subspace, which has better generalization and discrimination capabilities compared to the PCA-based modeling.

Liu [Liu07] regards the fitting procedure as a classification problem instead of least-squares optimization as commonly used in image registration problem. The classification function is learned based on the nonlinear boosting algorithm, in which a set of weak classifiers $f_i(\mathbf{p})$ are selected and combined to form a strong classifier $F(\mathbf{p}) = \sum_i f_i(\mathbf{p})$. The modeled fitting procedure turns out to find the optimal parameters $\mathbf{p}$, which maximize the score of the strong classifier. Hao *et al.* [WLD08] propose a pairwise ranking based learning to mitigate the imbalance data problem in binary classification model. The modification also results in a smoother score function, which is favorable for the local optimizer-based fitting.

2.2.2.3 Robustness Handling

The robustness of AAM fitting can be easily affected by occlusions due to the nature of modeling. Occlusion occurs, when the faces are occluded by other objects such as sunglasses or the face itself in case of 3D-pose variation. In [GMB06], Gross *et al.* address the occlusion problem in AAM fitting by introducing a robust error function $\rho(t; \sigma)$, where σ is a vector of scale parameters. This method enables AAM to fit and track the object in case of occlusions.

The majority of the works on AAM fitting assumes near frontal and upright poses. However, for applications, in which large pose rotation occurs, the linear appearance model is not sufficient to recover the nonlinearity due to facial

rotation. A solution to this problem is to introduce pose information using nonlinear models or a 3D shape model. Cootes *et al.* [CWT00] propose a view-based method to approximate the nonlinear rotations. Their method creates five models in different views, *i.e.* frontal, left (right) semi-profile, and left (right) full profile. Each model fits the face in a range of view angles. The combination essentially covers all poses. The authors assume that there is a linear relationship between the model parameters and the corresponding view angle θ. They learn this linear relationship using regression for estimating view angles. Instead of using piecewise linear models for approximating the pose manifold, Romdhani *et al.* [RPG00] use a unified nonlinear model. The self-occluded points are masked with zeros. Kernel PCA is applied to model the nonlinear variations caused by pose rotation.

Another straightforward research direction for tackling the pose problem is the investigation of a 3D shape model, with which the inherent nonlinearity caused by out-of-plane rotation is diminished. Xiao *et al.* [XBMK04] introduce a 2D+3D AAM, in which a 3D shape model is built and used for constraining the 2D shape search:

$$\sum_{\mathbf{x}\in s_0}\left[\mathbf{g}_0(\mathbf{x})+\sum_j \mathbf{g}_j(\mathbf{x}){b_g}^j-\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{b}_s;\mathbf{t}))\right]^2+K\|\mathbf{N}(\mathbf{s}_0+\mathbf{P}_s\mathbf{b}_s;\mathbf{t})-\mathbf{P}(\bar{\mathbf{s}}_0+\bar{\mathbf{P}}_s\bar{\mathbf{b}}_s)\|^2. \tag{2.11}$$

The first term corresponds to the objective for conventional 2D AAM fitting, the second term ensures that the fitted 2D shape should deform similarly as the projected 3D shape. Here, $\mathbf{N}(\mathbf{s};\mathbf{t})$ denotes the similarity transform with parameter $\mathbf{t}$, $\mathbf{P}$ denotes the camera projection matrices, which project shapes from 3D to 2D. The constraint avoids the ambiguous cases, when a fitted 2D shape is not a plausible shape due to the nonlinearity of out-of-plane rotation. Additional studies extend the 3D AAM to fit on image sources of non-monocular cameras, such as multi-view cameras [HXM+04, KBM+05] or stereo cameras [LJJ06, SK06, SK06].

2.3 Applications

AAM has been widely applied in different computer vision research domains including recognition, tracking, synthesis, and segmentation. This section gives a short review of applications using AAM.

AAM can be used to interpret face identities as it provides discriminant information of facial appearance (shape and texture). Edwards *et al.* [ECT98] use model parameters directly and Linear discriminant analysis (LDA) for face identification. AAM is used to normalize face pose to enhance face recognition

performance under pose variations [GKSC06, GES09]. As the shape is an important clue for estimating facial expression, many researchers use AAM to extract shape information for expression analysis [DR07, MBV06, SK06, ADD04, LK08, SK08], pain expression analysis [ALC+07], and action unit detection [LCK+10]. In [GZSM07, LRBS09], the authors use AAM for robust age estimation.

By manipulating the model parameters, AAM is able to synthesize novel instances of face images by projecting back from parameter space to image space. Cootes *et al.* [CWT00] use view-based AAMs for synthesizing novel views of face images by applying linear regression between view-based identity subspaces. Chen *et al.* [CLR+04] present a system for automatic portrait generation using AAM and hair model. AAM is also used to synthesize various facial expressions in [ADD04]. Studies on expression transfer are presented in [TMCB07, MBV06], in which expression of one person is transferred to another with AAM. The model is also applied for building a magic mirror, in which user faces are replaced with preferred faces of celebrities [AGB+12].

AAM is also widely applied for medical image segmentation [CET98a, RCA03, BGS+02, CBET99] as well as tracking of faces [HD05, SK06, DKB07] or even generic objects [Ste01].

3 Classification Appearance Models

In this chapter, we present the first discriminative appearance model in this thesis, which is based on a boosted classification model. A brief introduction and motivation of this approach are first given in Section 3.1. We explain the proposed model in detail in Section 3.2, where a 2D shape model, which is used throughout this thesis, is first explained. Afterwards, the training of a classification-based appearance model is presented. In Section 3.3, we describe the steps for aligning face images with the proposed model. The experimental setup and evaluation results are discussed in Section 3.4.

3.1 Introduction

As described in Section 2.2, AAM [CET98a, MB04] considers both shape and texture information of a face object. It combines constraints on both shape and texture by learning statistical generative models, which approximate the distribution of both information in a training data set. A shape is represented by landmark positions (cf. Figure 3.7(b)), whereas the appearance is represented by pixel intensities in a shape-free face image (cf. Figure 3.7(c)). The fitting of an AAM is defined by solving a least mean square error (LMSE) problem, where the difference between the warped image and the model appearance is minimized. Efficient optimization algorithms, such as the Inverse Compositional (IC) and Simultaneous Inverse Compositional (SIC) methods have been proposed by Baker and Matthews [BM04], which enable fast face alignment for real-time applications. However, the alignment performance degrades quickly, when generic AAMs are trained instead of person specific AAMs [GMB05]. The generalization issue is caused by generative appearance modeling and the LMSE optimization schema as claimed in [Liu07].

In order to tackle this generalization problem, Liu proposed the Boosted Appearance Model (BAM) [Liu07], in which a shape representation similar to AAM is used, whereas the appearance is represented by a set of discriminative Haar-features, trained to form a boosted classifier. The discriminative appearance

models are able to distinguish between correct and incorrect alignments. The optimal BAM fitting for a given test image is searched iteratively by optimizing the corresponding classification score function. It has been shown that the BAM improves the generalization capability compared to AAM.

However, as we know that the number of Haar features to be boosted is extremely large, since the dimension of the parameter space is high. Training a BAM using Haar features requires to boost more than one hundred thousand rectangular features within the mean shape, which results in an inefficient training procedure. To avoid this, we propose to use a local structural feature with less configurable parameters for boosting. This enables the training procedure to be extremely efficient. The local structural feature is inspired by the work of Fröba *et al.* [FE04], in which the modified census transform (MCT) is applied for face detection. The face detector based on the MCT feature yields better detection performance despite its fast training and detection speed compared to the state-of-the-art approach [VJ04]. The MCT feature, however, is a binarized pattern, which is not suitable for deriving an analytical optimization algorithm. In this work, we use the unbinarized census transform feature, which we call pseudo census transform (PCT). The PCT feature is projected discriminatively to a scalar indicating the correctness of face alignments. We boost the scalar values using GentleBoost [FHT00]. Multi-scale PCT features are also investigated. We evaluate our PCT-based BAM fitting on four different data sets. Our proposed approach achieves slightly better performance on seen data compared to the Haar-based BAM. However, results on the unseen data show that the PCT-based BAM outperforms the Haar-based BAM significantly in terms of the average convergence rate, which indicates that our approach generalizes better on unseen data.

3.2 Face Model

The classification-based appearance model presented in this chapter contains a shape model and an appearance model. For the shape model, we use the point distribution model (PDM), which is widely used in other statistical deformable models. The shape model defines a set of parameters, which need to be recovered for a given test image. The appearance model defines a discriminant function over the shape parameters, which is represented with a set of boosted discriminative features. The following subsections describe the details for building the shape and appearance models.

3.2.1 Shape Model

The word "shape" describes the external boundary of an object. It conveys all the geometrical information that remains, when location, scale and rotational effects are filtered out from an object [Ken84]. From this definition, the term shape is invariant to similarity transformations. A simple shape can be described by basic geometric primitives such as a set of points, lines, curves, etc. The most common primitive is the point, which is widely studied in the research domain of statistical shape analysis for its simplicity in representation and strong theoretical support. In case of face objects, the landmark points can be located at the salient facial regions, such as the eye corners and the mouth corners, or at the boundary of the facial organs or the faces. A v-point shape in k ($k = 2, 3$) dimensional space can be represented by a kv dimensional vector. In this thesis, we focus on using 2D landmarks for representing shape information of an image. Hence, mathematically, a 2D shape vector containing v landmarks can be defined as:

$$\mathbf{s} = [x_1, y_1, x_2, y_2, \ldots, x_v, y_v]^\top . \tag{3.1}$$

Figure 3.1 shows an example of a face shape represented with 58 landmarks. The acquisition of these landmarks is usually done by manually placing several points along the contours of the salient facial features and the outline of the face. Annotating these landmarks on hundreds of images is a tedious work. Often, noise may be introduced by vague definition of landmarks and inconsistent labeling, which eventually leads to an imprecise modeling of shape. Figure 3.2(a) displays a closer view of an example shape with the index numbers plotted on the right side of each landmark. A 2D mesh is obtained by applying Delaunay triangulation as plotted in Figure 3.2(b). As defined before, location, scale and rotational effects need to be filtered out to obtain a true shape representation. This is carried out by aligning all shapes to a common coordinate framework. The well-known Procrustes analysis [Goo91] is applied to align all shapes iteratively. Figure 3.3 explains the alignment procedure with the Procrustes analysis. Figure 3.3(a) plots the point cloud of all shapes before alignment. After alignment, the translation, scaling and rotation factors of all shapes are normalized (cf. Figure 3.3(b)). The average of the N aligned shapes can be estimated as:

$$\mathbf{s}_0 = \frac{1}{N} \sum_{i=1}^{N} \tilde{\mathbf{s}}_i, \tag{3.2}$$

where $\tilde{\mathbf{s}}_i$ denotes the i-th aligned shape. After alignment, the only difference of these set of shapes is the shape variation. Figure 3.4(a) shows the spatial variations of each individual landmark. The direction of the major axis of each ellipse plotted in Figure 3.4(a) stands for the principal direction of the point distribution. Note large variations are observed on the landmarks located along the face contours, which explains the large portion energy of nonrigid deformations

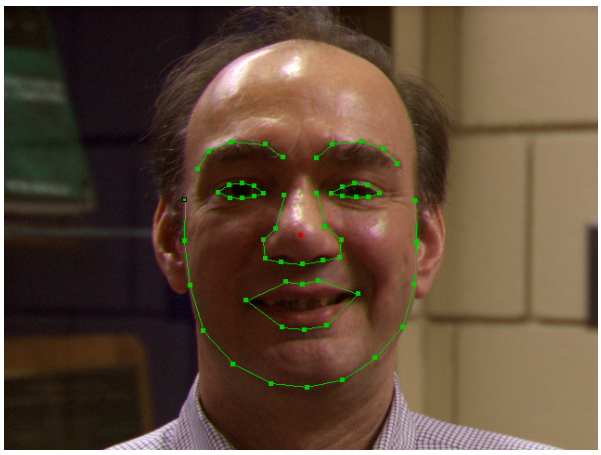
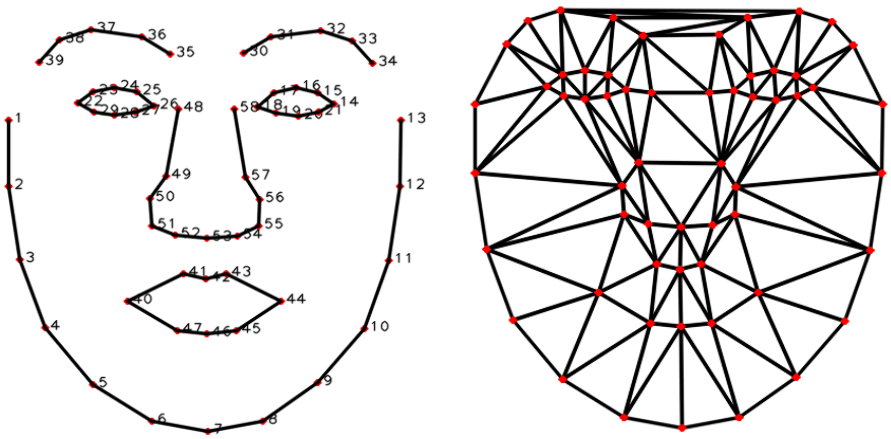


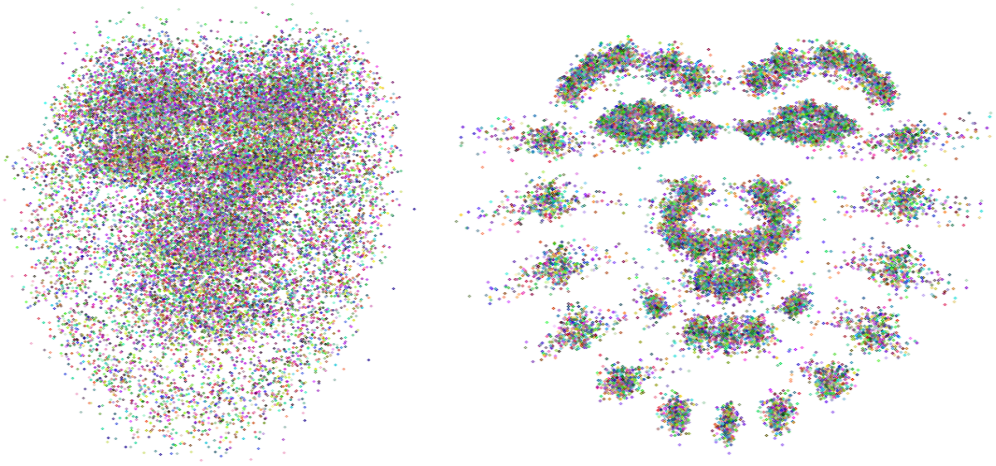
Figure 3.1: Example annotation of a face using 58 landmarks.



(a) (b)

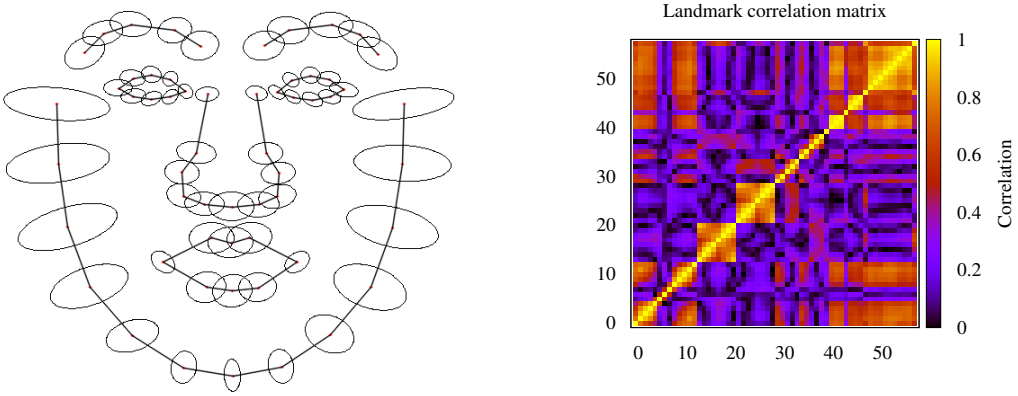
Figure 3.2: (a) Indexed facial landmarks, and (b) Corresponding 2D mesh via Delaunay triangulation.

due to the pan rotation of faces. The inter-point correlation matrix is plotted in Figure 3.4(b). The horizontal and vertical axes in this plot correspond to the landmark indexes (cf. Figure 3.2(a)). Note that the landmarks located on the same facial component are highly correlated. In addition, landmarks on the face contour, nose and mouth regions are also correlated to some degree, as they are all dependent on nonrigid pose changes.



Unaligned shapes Aligned shapes

Figure 3.3: Shape alignment via Procrustes analysis.



(a) (b)

Figure 3.4: (a) Principal point direction and (b) Correlation matrix of landmarks, where the horizontal and vertical axes correspond to the landmark indexes (cf. Figure 3.2(a)).

To decorrelate and reduce the redundancy in the multivariate shape data, PCA (also known as the Karhunen-Loève transform) is used as a dimensionality reduction method, which delivers new axes ordered according to their projection variances. The projected shape vector lies in a low dimensional subspace, in which the variations of the original data is maximally preserved.

PCA finds principle data variation modes by applying eigen-decomposition of the data covariance matrix. The covariance matrix of the shape data is given as:

$$\Sigma_s = \frac{1}{N}\sum_{i=1}^{N}(\tilde{\mathbf{s}}_i - \mathbf{s}_0)(\tilde{\mathbf{s}}_i - \mathbf{s}_0)^\top. \tag{3.3}$$

The principal axes of the $2v$ dimensional shape vectors are given as the n eigenvectors, $\mathbf{P}_s^i$, ordered decreasingly according to their corresponding eigenvalues. A shape $\tilde{\mathbf{s}}$ can be generated with a mean shape $\mathbf{s}_0$ plus a linear combination of the n eigenvectors (eigenshapes):

$$\tilde{\mathbf{s}} = \mathbf{s}_0 + \sum_{i=1}^{n}\mathbf{P}_s^i b_s^i, \tag{3.4}$$

where b_s^i is the parameter for the i-th shape component. Usually, the components with small eigenvalues are discarded, as they explain the deformation energy of annotation noise. To compromise between accuracy and compactness of the model, we retain 95% of the shape variation, which results in a 15 dimensional eigenshape space. This is a rather substantial reduction since the original shape vector has a dimensionality of $2 \times 58 = 116$. Figure 3.5 illustrates the first three modes in the shape model. The first row represents the variation of the first shape component, where the shape in the middle is the mean shape while the one on the left (right) side is generated by setting the first shape parameter to -3σ (3σ) and the remaining parameters to 0. The second row represents the second shape component, and the third row represents the third component. Figure 3.6 enumerates the top ten eigenvalues in the shape model in a descending order. Note the first variation mode, which corresponds to pan rotation, takes up 59% of the overall variations in the training data.

We now use the shape model to interpret the shape information of a given facial image. Given an image $\mathbf{I}$ and its corresponding annotated shape $\mathbf{s}$, we first apply a similarity transform to align the shape to a reference shape [MB04], *e.g.* the mean shape $\mathbf{s}_0$:

$$\mathbf{t} = \arg\min_{\mathbf{t}} \|\mathbf{N}(\tilde{\mathbf{s}}, \mathbf{t}) - \mathbf{s}\|. \tag{3.5}$$

The similarity transform parameter vector $\mathbf{t}$ contains four items, which correspond to the scaling, rotation, and translation factors. The model parameter vector $\mathbf{b}_s$ is estimated as:

$$\mathbf{b}_s = \mathbf{P}_s(\tilde{\mathbf{s}} - \mathbf{s}_0), \tag{3.6}$$

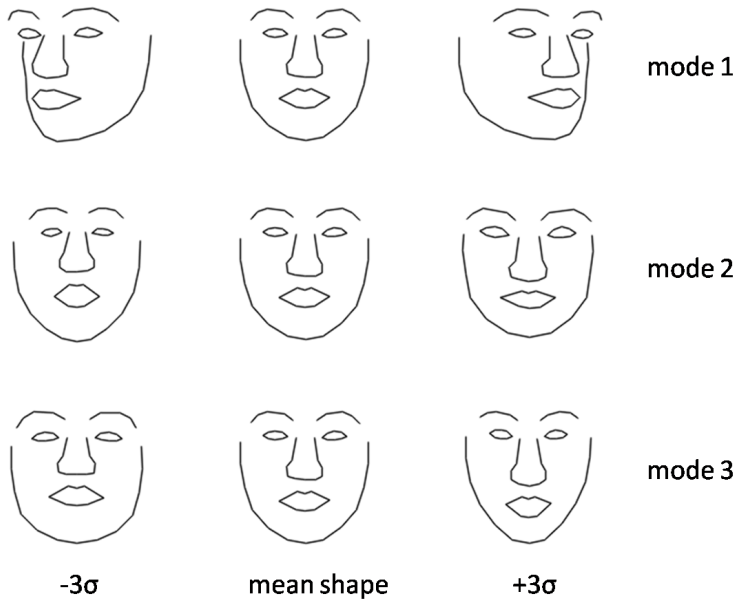


Figure 3.5: Shape deformation of the first three principal modes.

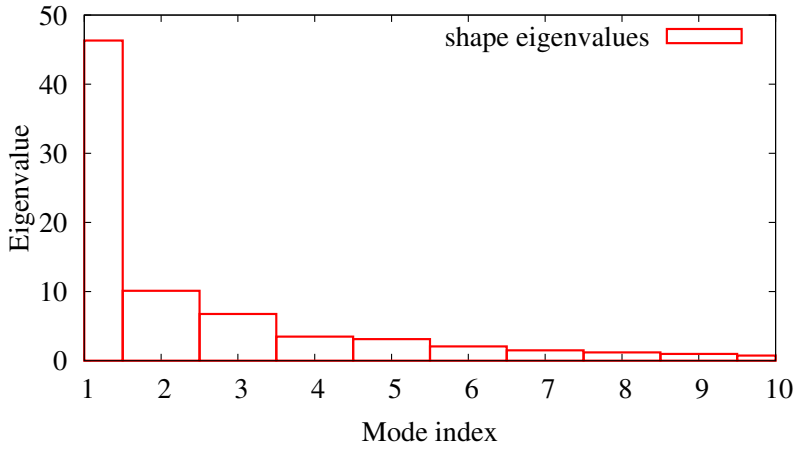
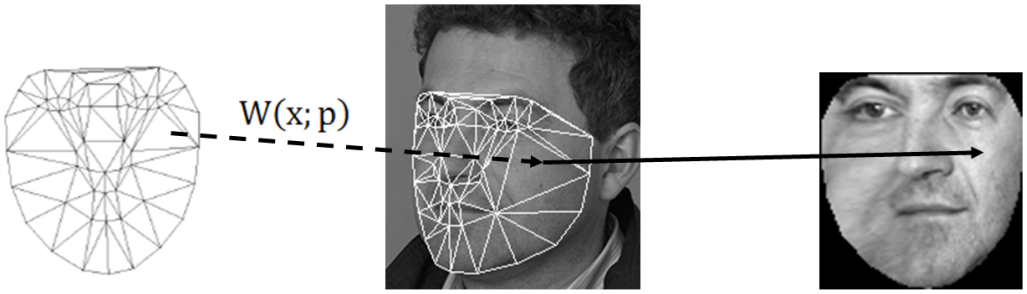


Figure 3.6: Ten largest eigenvalues in a shape model.

where $\tilde{\mathbf{s}}$ is the aligned shape. Hereafter, we use a combined parameter vector $\mathbf{p} = [\mathbf{t} \mid \mathbf{b}_s]$ for parametrizing a shape instance $\mathbf{s}$.

With Delaunay triangulation, the mean shape $\mathbf{s}_0$ (cf. Figure 3.7(a)) and the shape $\mathbf{s}$ (cf. Figure 3.7(b)) are triangulated to a base mesh and an instance face mesh. A non-linear mapping function $\mathbf{W}(\mathbf{x}; \mathbf{p})$ is defined with a piece-wise affine



(a) (b) (c)

Figure 3.7: Shape model and warping function. (a) The mean shape $\mathbf{s}_0$; (b) A face image superimposed with a shape $\mathbf{s}(\mathbf{p})$; (c) A face image warped to the mean shape $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$.

warping, which maps pixel $\mathbf{x}$ defined in the mean shape to the instance shape. A shape-free image $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ (cf. Figure 3.7(c)) is obtained by warping a face image $\mathbf{I}$ to the coordinate of the mean shape. The pixel values at $\mathbf{W}(\mathbf{x};\mathbf{p})$ is approximated with bi-linear interpolation.

3.2.2 Appearance Model

The appearance model is defined by a collection of features extracted on the shape-free face images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$. In [Liu07], the rectangular Haar features are adopted. Haar features are known as efficient local region-based features for general object detection [VJ04]. One drawback of the Haar features is that the configuration space is extremely large, which makes the selection procedure very slow. In [FE04], Fröba *et al.* found out that the feature extracted by the modified census transformation (MCT) outperforms the Haar features in face detection. Especially, due to the low dimensional configuration space of the MCT feature, the detector can be trained very efficiently. Inspired by their work, we propose to select the unbinarized census transform, which we call pseudo census transform (PCT) feature for our appearance model [1]. The PCT feature $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_K)^\top$ is a K dimensional vector, which contains the pixel values in a $\sqrt{K} \times \sqrt{K}$ neighborhood centered at $\mathbf{x} = (r, c)$ and subtracted by local mean. For simplicity, we used a fixed K ($K = 9$) in this work. The PCT

[1] The MCT features cannot be applied for deriving the fitting algorithm, since it is a binary pattern.

feature $\boldsymbol{\varphi}$ is then obtained by ordering the K filter responses of a filter bank plotted in Figure 3.8(b) at a certain position (r, c). The mask of the first filter is defined as follows:

$$\mathbf{T}_0 = \begin{pmatrix} 8/9 & -1/9 & -1/9 \\ -1/9 & -1/9 & -1/9 \\ -1/9 & -1/9 & -1/9 \end{pmatrix}. \tag{3.7}$$

The rest of the filter masks are defined accordingly by shifting the position of the value 8/9 in the matrix (cf. Figure 3.8(b), white cell corresponds to the positive element and gray cell corresponds to the negative elements). Notice that the responses of the filters are equivalent to the PCT feature values. This enables us to define K image templates $\mathbf{A}_{i=1,\ldots,K}$ with the filter mask placed at position $\mathbf{x} = (r, c)$ for one PCT feature. The inner product between the template and the warped image is equivalent to computing the filter responses:

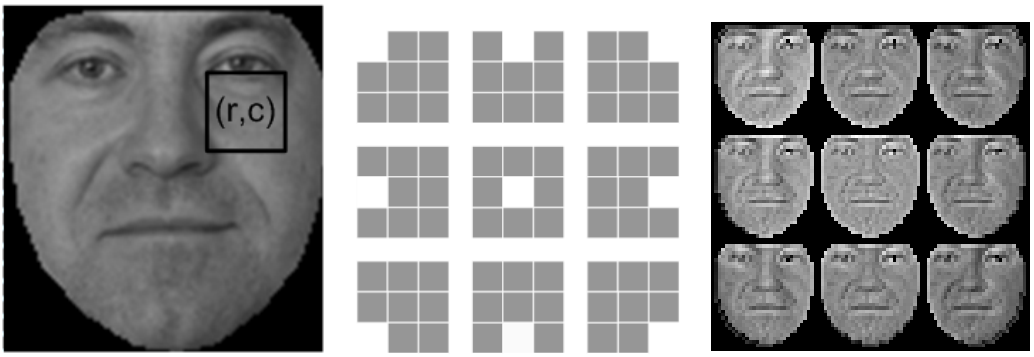
$$\varphi_i = \mathbf{A}_i^\top \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})) = \mathbf{T}_i * \mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p})), i = 1, \ldots, K. \tag{3.8}$$

If we consider $K = 9$, the filter mask at the center of Figure 3.8(b) corresponds to a discrete approximation to a 3×3 Laplacian filter, which results in an illumination invariant filter response. The other filter masks capture contours of different orientations in an image. We apply PCT on two face images under different illumination conditions as shown in Figure 3.9. The filter responses below show that the PCT feature representation is robust against illumination changes.

3.2.2.1 Weak Classifiers

The PCT-feature only captures structural features in a small local area of a face image. Decision rules based on a single feature location are incompetent to form a good discriminative model that separates correct and incorrect alignments. The fitting process will be less efficient, if all feature locations are used. Furthermore, the features located in homogeneous regions, such as the cheek areas, are not very informative. Including those features may even decrease the robustness. We hence employ a boosting procedure to select a set of feature locations, in which the weak classifiers are trained based on each of those features. The selected weak classifiers are aggregated together to form a strong classifier, which we consider as our boosted classification appearance model.

To make the problem simple, we focus on linear models for training our weak classifiers. The simple linear models are less likely overtrained, thus have better generalization ability compared to the nonlinear models.



(a) (b) (c)

Figure 3.8: (a) The parametrization of a weak classifier, *i.e.* center of the PCT filter positioned at (r, c); (b) K PCT filter masks ($K = 9$), the top left filter mask corresponds to the filter kernel defined in Equation 3.7; (c) PCT-filter responses of a shape-free image.

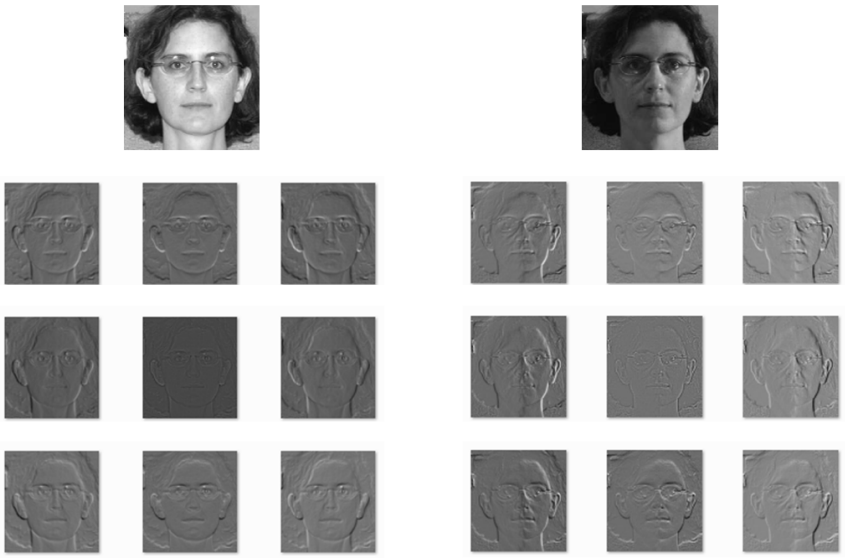


Figure 3.9: PCT filters applied on images with different illumination conditions.

3.2.2.2 Linear Classification Models

The goal of binary classification is to take an input vector $\mathbf{x} \in \mathbb{R}^D$ and assign it to one of the binary classes $\mathcal{C}_k$, where $k = 1, 2$. The input space is divided

into two decision regions with a decision boundary. When considering linear models for classification, the decision boundary is depicted as a linear function of the input vector $\mathbf{x}$. In other words, the linear decision surface is defined by $D-1$ dimensional hyperplanes within the D-dimensional input space. The linear function is called linear discriminant function, which is defined as follows:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b, \tag{3.9}$$

where $\mathbf{w}$ indicates a weighting vector and b is a bias term. An input vector $\mathbf{x}$ is assigned to class $\mathcal{C}_1$ if $f(\mathbf{x}) \geq 0$, and to class $\mathcal{C}_2$ otherwise. The corresponding decision boundary is defined by the equation $f(\mathbf{x}) = 0$. The weighting vector $\mathbf{w}$ is orthogonal to the decision surface.

There are several approaches for determining linear discriminant functions. In this work, we investigate three different approaches, which try to minimize classification loss in various points of view. The first model is based on linear discriminant analysis, which we denote as a probabilistic generative model. The second model is logistic regression, which is actually a generalized linear model based on a probabilistic discriminative model. The third model is based on the maximal margin theory, which is usually called linear support vector machines (SVM).

3.2.2.2.1 Linear Discriminate Analysis (Probabilistic Generative Models) Assuming that the data of the underlying classes are Gaussian distributed, and the parameters of the distribution are known, $p(\mathbf{x}|\mathcal{C}_k) \sim \mathcal{N}(\mu_k, \Sigma_k)$. The decision is determined by comparing the posterior probabilities:

$$p(\mathcal{C}_k|\mathbf{x}) \propto p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k). \tag{3.10}$$

A sample $\mathbf{x}$ is assigned to class $\mathcal{C}_i$, if $p(\mathcal{C}_i|\mathbf{x}) > p(\mathcal{C}_j|\mathbf{x})$. In general, the discriminant function is defined with the log odds ratio $\ln\left[p(\mathcal{C}_1|\mathbf{x})/p(\mathcal{C}_2|\mathbf{x})\right]$ for the two classes. Assuming that the covariance matrices for each class are equal, we define the discriminant function as follows:

$$g(\mathbf{x}) = \ln\left[p(\mathcal{C}_1|\mathbf{x})/p(\mathcal{C}_2|\mathbf{x})\right] \tag{3.11}$$

$$= -\frac{1}{2}(\mathbf{x}-\mu_1)^\top \Sigma^{-1}(\mathbf{x}-\mu_1) + \frac{1}{2}(\mathbf{x}-\mu_2)^\top \Sigma^{-1}(\mathbf{x}-\mu_2) + \ln\frac{P(\mathcal{C}_1)}{P(\mathcal{C}_2)}, \tag{3.12}$$

where Σ is the shared covariance matrix. The quadratic term will be canceled, which results in a linear discriminant function as in Equation 3.9, where

$$\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2), \tag{3.13}$$

and

$$b = \ln \frac{P(\mathcal{C}_1)}{P(\mathcal{C}_2)} - \frac{1}{2}(\mu 1 + \mu 2)^\top \Sigma^{-1}(\mu_1 - \mu_2). \tag{3.14}$$

In practice, μ_k, $\mathbf{\Sigma}_k$ and $P(\mathcal{C}_k)$ are estimated from the training data. We use the relative frequencies of the examples in each class $\hat{P}(\mathcal{C}_k)$ as the estimation of the prior probabilities $P(\mathcal{C}_k)$:

$$\hat{P}(\mathcal{C}_k) = \frac{n_k}{\sum_k n_k}, \quad k = 1, 2. \tag{3.15}$$

μ_k and Σ_k are estimated with the sample mean $\hat{\mu}_k$ and sample covariance matrix $\hat{\Sigma}_k$ in a maximum likelihood sense:

$$\mu_k = \hat{\mu}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{x}_{k_i}, \quad k = 1, 2, \tag{3.16}$$

$$\Sigma_k = \hat{\Sigma}_k = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} (\mathbf{x}_{k_i} - \hat{\mu}_k)(\mathbf{x}_{k_i} - \hat{\mu}_k)^\top, \quad k = 1, 2. \tag{3.17}$$

Considering the previous assumption that different classes share one single covariance matrix, *i.e.* $\Sigma_1 = \Sigma_2 = \Sigma$, we adopt an unbiased estimator as proposed in [XQ07], when n_1 and n_2 are large enough:

$$\hat{\Sigma} = \frac{(n_1 - 1)\hat{\Sigma}_1 + (n_2 - 1)\hat{\Sigma}_2}{n_1 + n_2 - 2} \approx P(\mathcal{C}_1)\hat{\Sigma}_1 + P(\mathcal{C}_2)\hat{\Sigma}_2. \tag{3.18}$$

Note that the Gaussian-based LDA is closely related to the Fisher's linear discriminant (FLD). They only differ in a scale factor.

3.2.2.2.2 Logistic Regression (Probabilistic Discriminative Models) In contrast to generative modeling, we investigate another important learning approach, logistic regression, which explicitly uses the functional form of the generalized linear model and determines the model parameters directly using maximum likelihood. The resulting model represents a discriminative classification model.

Despite its name, logistic regression is a model for classification rather than regression. It uses a sigmoid function of linearly projected feature vector $\mathbf{x}$ to represent the posterior probability of class $\mathcal{C}_1$:

$$p(\mathcal{C}_1|\mathbf{x}) = h(\mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x}). \tag{3.19}$$

For the two-class case, $p(\mathcal{C}_2|\mathbf{x}) = 1 - p(\mathcal{C}_1|\mathbf{x})$. Here $\sigma(.)$ is the logistic sigmoid function. For simplicity of notation, the definition of the parameter vector $\mathbf{w}$ is slightly different from Equation 3.9. The dimension of $\mathbf{w}$ is $D+1$, where D is the dimension of input data. The first element in $\mathbf{w}$ corresponds to the bias term b in Equation 3.9. For this, the original input data $\tilde{\mathbf{x}}$ is augmented with an additional 1 as the first element of $\mathbf{x}$, *i.e.*, $\mathbf{x} = \left[1, \tilde{\mathbf{x}}^\top\right]^\top$.

Considering we have data $\mathbf{x}_i$ and their class labels y_i, where $y_i = 1$ for class $\mathcal{C}_1$, $y_i = 0$ for class $\mathcal{C}_2$. The log-likelihood for the N given data can be written as:

$$\ell(\mathbf{w}) = \sum_{i=1}^{N} \log p(\mathcal{C}_1|\mathbf{x}_i)^{y_i} p(\mathcal{C}_2|\mathbf{x}_i)^{1-y_i} = \sum_{i=1}^{N} \left\{ y_i \mathbf{w}^\top \mathbf{x}_i - \log\left(1 + e^{\mathbf{w}^\top \mathbf{x}_i}\right) \right\}. \tag{3.20}$$

The parameters of the logistic regression model are estimated with maximum likelihood. The Newton-Raphson algorithm is used for solving the nonlinear optimization in an iterative manner:

$$\mathbf{w} = \mathbf{w} - \frac{\nabla \ell(\mathbf{w})}{H(\ell)}, \tag{3.21}$$

where $\nabla \ell(\mathbf{w})$ is the first-order derivative:

$$\nabla \ell(\mathbf{w}) = \frac{\partial \ell(\mathbf{w})}{\partial \mathbf{w}} = \sum_{i=1}^{N} \mathbf{x}_i (y_i - h(\mathbf{x}_i)), \tag{3.22}$$

and $H(\ell)$ is the second-order derivative:

$$H(\ell) = \frac{\partial^2 \ell(\mathbf{w})}{\partial \mathbf{w} \partial \mathbf{w}^\top} = -\sum_{i=1}^{N} \mathbf{x}_i \mathbf{x}_i^\top h(\mathbf{x}_i)(1 - h(\mathbf{x}_i)). \tag{3.23}$$

It is convenient to write the derivatives in matrix notation. Let $\mathbf{y}$ denote the vector of y_i values, X the $N \times (D+1)$ matrix of $\mathbf{x}_i$ values, $\mathbf{h}$ the vector of fitted probabilities with i-th element $h_{\mathbf{w}}(\mathbf{x}_i)$, and $\mathbf{H}$ a diagonal matrix of $(D+1)$ weights with element $h_{\mathbf{w}}(\mathbf{x}_i)(1 - h_{\mathbf{w}}(\mathbf{x}_i))$. Then, we have $\nabla \ell(\mathbf{w}) = \mathbf{X}^\top (\mathbf{y} - \mathbf{h})$ and $H(\ell) = -\mathbf{X}^\top \mathbf{H} \mathbf{X}$. The Newton step thus becomes

$$\mathbf{w} = \mathbf{w} + (\mathbf{X}^\top \mathbf{H} \mathbf{X})^{-1} \mathbf{X}^\top (\mathbf{y} - \mathbf{h}). \tag{3.24}$$

We use a regularization term to regularize the maximization of the log-likelihood function to avoid over fitting:

$$\ell(\mathbf{w}) = \sum_{i=1}^{N} \left\{ y_i \mathbf{w}^\top \mathbf{x}_i - \log\left(1 + e^{\mathbf{w}^\top \mathbf{x}_i}\right) \right\} + \lambda \|\mathbf{w}\|^2. \tag{3.25}$$

We denote this as the L2 regularized logistic regression. The Newton step in the regularized optimization becomes:

$$\mathbf{w} = \mathbf{w} + (\mathbf{X}^\top \mathbf{H}\mathbf{X} + \Lambda)^{-1} \left(\mathbf{X}^\top (\mathbf{y} - \mathbf{h}) - \Lambda \mathbf{w}\right), \tag{3.26}$$

where $\Lambda = \lambda \mathbf{I}_{D+1}$, and $\mathbf{I}_{D+1}$ is an identity matrix of size $D + 1$.

3.2.2.2.3 Linear Support Vector Machines (Maximum Margin) Support vector machine (SVM) becomes the most popular approach for finding decision boundaries, which might give a low generalization error. The decision boundary in SVM is chosen to be the one for which the margin is maximized. The margin is defined as the smallest distance between the decision boundary and any of the data samples.

Given N training data $\mathbf{x}_i$ and corresponding labels y_i, where $y_i \in \{1, -1\}$, we want to find the maximum-margin hyperplane, which best separates the training data with respect to the labels. Mathematically, a hyperplane is defined as $\mathbf{w}^\top \mathbf{x} + b = 0$. If the training data are linearly separable, we select two hyperplanes for separating the data such that there are no data points between them. The distance between the hyperplanes will be maximized. The region bounded by them is called "the margin". These hyperplanes can be described by the equations $\mathbf{w}^\top \mathbf{x} + b = 1$, and $\mathbf{w}^\top \mathbf{x} + b = -1$. The distance between these two hyperplanes is $\frac{2}{\|\mathbf{w}\|}$. Thus maximizing the margin is equivalent to minimizing $\|\mathbf{w}\|$. In order to prevent data points from falling into the margin, the following constraint is enforced:

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \ldots, N. \tag{3.27}$$

The optimization problem is formulated as:

$$\arg\min_{\mathbf{w},b} \|\mathbf{w}\|, \tag{3.28}$$

$$\text{subject to} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \qquad i = 1, \ldots, N. \tag{3.29}$$

In practice, the classes cannot be cleanly separated by a defined hyperplane due to overlap in feature space. For this, the soft margin is introduced to allow some points to be on the wrong side of the margin. A set of slack variables $\xi = \{\xi_1, \xi_2, \ldots, \xi_N\}$ is defined, which measures the degree of misclassification of the data $\mathbf{x}_i$:

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \ldots, N. \tag{3.30}$$

The constrained objective function is then appended with a set of non-zero penalty terms. The optimization becomes a trade off between a large margin and small error penalty:

$$\min_{\mathbf{w},\xi,b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i \right\}, \tag{3.31}$$

$$\text{subject to} \quad y_i(\mathbf{w}^\top \mathbf{x}_i - b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad i = 1, \ldots, N. \tag{3.32}$$

The constrained problem is convex and can be solved with quadratic programming techniques by introducing Lagrange multipliers.

3.2.2.3 Boosting-based Appearance Learning

Having the linear classification models introduced, we apply those models to the PCT features to obtain weak classification scores, which indicates the correctness of alignments. The prediction has high uncertainty due to the weak representation power of a single PCT feature vector. For this, we combine the individual PCT features in a boosting framework, which eventually results in a more reliable strong classifier. The basic idea of boosting is to create a highly accurate classifier by combining many relatively weak and inaccurate classifiers. This approach is also considered as an effective tool for selecting relevant features iteratively from a huge pool of features. Within each boosting iteration, the hardest examples, which are misclassified in the previous iteration, contribute more to finding the current decision boundary that makes a weak classifier. The combination can be a vote of the predictions from the weak classifiers. The confidence for a prediction can be a convex combination of the weak classification score functions.

The boosting algorithm has many variants [MR03]. The Adaboost [FS97] algorithm is one of the most popular variant, which was successfully applied for face detection [VJ04]. However, the discrete Adaboost algorithm provides hard decision functions as weak classifiers, which leads to a piecewise-constant strong classifier. The resulting score function is difficult to be optimized with a local optimizer. The Adaboost algorithm has a soft version called "Real Adaboost", which returns real-valued decisions using half log-odds. A gentler version of Adaboost (thus named as Gentleboost) is proposed in [FHT00], which makes the decision stable, when class probabilities are close to 0 or 1. In this work, we use the Gentleboost algorithm for boosting our strong classifier, as the resulting score function is smoother and thus favorable for a local optimizer. In addition, the Gentleboost algorithm has shown its superior performance in object detection tasks, when compared to the other variants of Adaboost, due to its robustness to noise and outliers [LKP03].

The Gentleboost algorithm is illustrated in Algorithm 1. Given a set of facial images with manual labels, positive and negative training samples are generated according to Section 3.2.3.1. Once the PCT features for a set of training samples are computed and the corresponding linear classification models are trained, an optimal weak classifier f_m is found, such that the weighted least square error is minimal. The aforementioned weak classifier computation is conducted for each feature in the hypothesis space, the optimal weak classifier with minimal error $\epsilon(f)$ is exhaustively searched. The exhaustive search procedure is a time demanding step in the Gentleboost algorithm, Line 3 in Algorithm 1, which is normally fairly slow if the hypothesis space is large. However, in contrast to the Haar feature in [Liu07], the hypothesis space of the PCT feature is much smaller, which is linear to the number of pixels defined inside the masked shape-free image. After updating the strong classifier, the sample weights are also updated, such that the later iterations focus on the difficult samples (Line 5).

Algorithm 1: The GentleBoost Algorithm

Data: Training data $\{\mathbf{x}_i; i = 1, 2, \ldots, N\}$ and their corresponding class labels $\{y_i; i = 1, 2, \ldots, N\}$, where $y_i \in \{-1, +1\}$.

Result: A strong classifier $F(\mathbf{x})$.

1 Initialize weights $w_i = 1/N$, and $F(\mathbf{x}) = 0$;

2 **foreach** *m=1,2,...,M* **do**

3 Fit the regression function $f_m(\mathbf{x})$ by weighted least-squares of y_i to $\mathbf{x}_i$ with weights w_i:

$$f_m(\mathbf{x}) = \arg\min_{f \in \mathcal{F}} \epsilon(f) = \sum_{i=1}^{N} w_i (y_i - f(\mathbf{x}_i))^2; \tag{3.33}$$

4 Update $F(\mathbf{x}) = F(\mathbf{x}) + f_m(\mathbf{x})$;

5 Update the weights by $w_i = w_i e^{-y_i f_m(\mathbf{x}_i)}$;

6 Normalize the weights such that $\sum_{i=1}^{N} w_i = 1$;

7 Output the classifier $F(\mathbf{x}) = \sum_{m=1}^{M} f_m(\mathbf{x})$.

The outputs of the boosting algorithm are a number of weak classifiers, each of which is parametrized with $\mathbf{c}_m = \{r, c\}$. We consider the set of weak classifiers $\{\mathbf{c}_m; m = 1, 2, \ldots, M\}$ as the appearance model of the BAMs.

3.2.3 Learning Alignment

We now formulate the problem of learning a scoring function for assessing the correctness of fitting a face model in this section. More precisely, for a given image, let us suppose that $\mathbf{p}$ is the shape parameter that represents the current alignment of the shape model. We are interested in learning a scoring function F, such that, when maximized with respect to $\mathbf{p}$, it returns the shape parameter corresponding to the correct alignment. Mathematically, if $\mathbf{p}^*$ is the shape parameter representing the correct alignment, F has to be such that

$$\mathbf{p}^* = \arg\max_{\mathbf{p}} F(\mathbf{p}). \tag{3.34}$$

With this formulation, the appearance model is actually a two-class classifier. In particular, we use a linear combination of several PCT features to define the appearance model:

$$F(\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))) = \Sigma_{m=1}^{M} f_m(\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))), \tag{3.35}$$

where $f_m(\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p})))$ is a weak classifier based on one single PCT feature of $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$. For ease of notation, we denote the weak classifier and the strong classifier as $f_m(\mathbf{p})$ and $F(\mathbf{p})$, respectively.

Our weak classifier using the PCT features is defined as follows:

$$f_m(\mathbf{p}) = \frac{\pi}{2} atan(\Sigma_{i=1}^{K} w_i^m S(\mathbf{A}_i^{m\top} \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))) + b^m), \tag{3.36}$$

where $\mathbf{A}_i^m$ is the i-th template defined at the m-th position (r_m, c_m). Since the classifier response $f_m(\mathbf{p})$ is continuous within -1 and 1, the $atan()$ function is used to ensure both discriminability and derivability. $S(.)$ is a sigmoid function defined as $S(t) = \frac{1}{1+e^{-\alpha t}}$, where α is a scale parameter. The sigmoid function normalizes the raw PCT feature values into a range of $(0, 1)$ before a linear projection. The projection vector $\mathbf{w}^m$ and bias b^m are learned on the training data with a linear classification model as described in Section 3.2.2.2. Additional model parameters, such as the cost parameter C in each linear SVM training is searched with cross validation. The regularization parameter for logistic regression is set empirically.

3.2.3.1 Training Samples

As explained before, our appearance model, which is a combination of a set of weak classifiers $f_m(\mathbf{p})$, is defined on the warped shape-free images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$. We need to collect a set of such warped images as our training data.

Given a face image $\mathbf{I}$ with manually labeled landmark $\mathbf{s}$, the corresponding shape parameter vector $\mathbf{p}$ is computed based on Equation 3.6. The warped

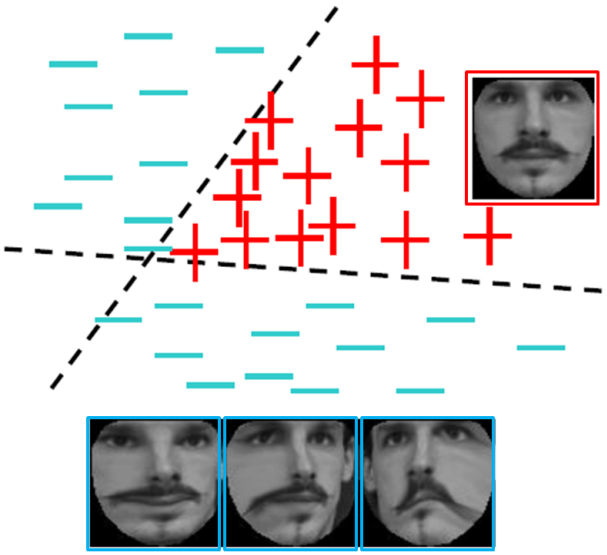


Figure 3.10: Positive and negative samples for training.

shape-free images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ are treated as positive samples ($y_i = 1$) for the boosting. For each image, a number of negative shape vectors $\mathbf{p}'$ are synthesized by random perturbation. Equation 3.37 describes the perturbation, where $\boldsymbol{\nu}$ is a random vector with each element uniformly distributed within $[-1, 1]$, $\boldsymbol{u}$ stores the standard deviations of all shape variation components, and σ is a constant scale that controls the level of perturbation:

$$\mathbf{p}' = \mathbf{p} + \sigma\boldsymbol{\nu} \cdot \boldsymbol{u}. \tag{3.37}$$

Together with the original face image, a perturbed negative shape vector can produce a negative training sample $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}'))$ ($y_i = -1$). This is an imbalanced learning problem, as a number of negative training samples can be generated with one positive training sample. Figure 3.10 illustrates a positive training sample (marked in red) and its corresponding negative samples (marked in blue).

3.2.3.2 Imbalanced Data for Classification

In theory, the data space for negative samples is unlimited and the data space for positive samples is limited to the available annotations. In practice, we generate a reasonable number of negative samples with random sampling within a certain range constrained by a face detection output and prior shape model. Due to the variations such as facial deformation, illumination, and background clutter,

we still need to sample enough negative data to approximate the distribution. We need to find a trade-off between imbalanced data and discriminativeness to avoid biased prediction towards the majority class. We investigate two strategies to handle the imbalanced learning problem according to [HG09], where the sampling and cost-sensitive methods are considered.

3.2.3.2.1 Sampling Methods for Imbalanced Learning One straightforward solution to imbalanced learning problem is to balance the data distribution by applying some sampling strategies. Previous studies show that classifiers trained on balanced data set provides superior performance compared to imbalanced data [CBHK02].

We augment the minority class with additional data by applying oversampling. In our case, the positive samples (minority class) can be either oversampled with tiny random perturbations of ground truth shapes or with synthetic sampling. The range of the perturbation can be limited according to the convergence criterion of a model fitting defined later in the experiment section (cf. Section 3.4), with which we allow a small deviation from the ground truth shape due to the annotation noise. The synthetic samples can be generated from a certain distribution or neighbouring samples. In this work, we generate samples in image space for the distribution-based synthesis, with the assumption of Gaussian distribution of the shape-free texture in the positive class. We use PCA to find the axes of a multi-variate Gaussian model and generate samples randomly in the PCA space.

3.2.3.2.2 Cost-Sensitive Methods for Imbalanced Learning Instead of balancing data distribution by applying different sampling strategies, the cost-sensitive methods assign different costs to misclassified instances [Elk01]. The essential part of the cost-sensitive learning methods is the concept of cost matrix, which is a numerical representation of the penalty of misclassification. In our case, we define a cost C_p as a penalty of misclassifying a positive instance as a negative instance. The cost C_n is defined for the contrary case. Typically, $C_p > C_n$ as we consider misclassifying minority class samples resulting in higher costs. The goal of cost-sensitive learning is to minimize the overall cost on the training set.

There are various ways of implementing cost-sensitive learning. In this thesis, we focus on the most simple variant, which is based on data space weighting. The following paragraphs describe the cost-sensitive version of the aforementioned linear classification models as well as boosting models.

Cost-Sensitive LDA The LDA model described in 3.2.2.2.1 is already derived in a cost sensitive manner. The shared covariance matrix is presented in a weighted form of individual covariance matrices as stated in Equation 3.17. The first term in Equation 3.14 also shifts the classification decision plane towards the minority class.

Cost-Sensitive Logistic Regression For cost-sensitive logistic regression, we assign different weights v_i to positive and negative class instances $\mathbf{x}_i$:

$$v_i = \begin{cases} n_2/N & \mathbf{x}_i \in \mathcal{C}_1 \\ n_1/N & \mathbf{x}_i \in \mathcal{C}_2 \end{cases} \tag{3.38}$$

Where n_1 and n_2 are the number of sample instances in class $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively, and $N = n_1 + n_2$. With this definition, we define the weighted log likelihood function to replace Equation 3.20:

$$\ell(\mathbf{w}) = \sum_{i=1}^{N} v_i \left\{ y_i \mathbf{w}^\top \mathbf{x}_i - \log\left(1 + e^{\mathbf{w}^\top \mathbf{x}_i}\right) \right\}. \tag{3.39}$$

Again we apply the Newton-Raphson method to maximize the weighted log likelihood in Equation 3.39. The Newton step in Equation 3.24 can be modified accordingly as follows,

$$\mathbf{w} = \mathbf{w} + (\mathbf{X}^\top \mathbf{H} \mathbf{V} \mathbf{X})^{-1} \mathbf{X}^\top (\mathbf{y} - \mathbf{p}) \mathbf{V}. \tag{3.40}$$

Here $\mathbf{V}$ is a diagonal matrix with its diagonal elements filled with v_i, *i.e.*, $\mathbf{V} = diag(\mathbf{v})$, where $\mathbf{v} = [v_1, v_2, \ldots, v_N]^\top$.

Cost-Sensitive Linear SVMs The cost sensitive linear SVM is implemented in a similar way to the cost sensitive logistic regression, where we assign individual weights v_i to the training instances with respect to their class labels.

Cost-Sensitive Boosting Similarly, we also apply a cost-sensitive boosting algorithm for boosting and combining weak classifiers. A simple modification is applied in Algorithm 1. Instead of assigning even weights to all data instances in the initialization step, we assign different weights for positive samples and negative samples, respectively. In particular, we assign

$$w_i = \begin{cases} 1/2n_1 & \mathbf{x}_i \in \mathcal{C}_1 \\ 1/2n_2 & \mathbf{x}_i \in \mathcal{C}_2 \end{cases} \tag{3.41}$$

Where n_1 and n_2 are the number of sample instances in class $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively.

(a) (b) (c)

Figure 3.11: Boosted PCT feature locations. (a) Boosted PCT feature locations in PCT-BAM; (b) Boosted PCT feature locations in the original scale in MSPCT-BAM; (c) Boosted PCT feature locations in the half scale in MSPCT-BAM.

3.2.3.3 Learned Appearance Models

We train an appearance model using linear SVM as weak classifier. The training set contains 400 annotated images. For each annotated image, we generate ten negative training samples. The cost sensitive method is applied for handling the imbalanced data problem. Figure 3.11(a) plots the top 40 locations of the boosted features in the learned appearance model. The feature locations are indexed in a 30×30 mask image. The gray pixels inside the mask indicate the locations of the boosted features. Note that the boosted PCT features are mainly located around the natural facial features, *i.e.* the eyes, nose, and mouth region. The features extracted at those locations contribute the most to the face alignment.

The PCT features extracted on the images at different scales 2^{-j} might contribute additional discriminative information for face alignment, where j is the level index in a multi-scale image pyramid. We also boost PCT features on different scales ($j = 0, 1, 2, 3$) of the shape-free images. The location of the boosted features in the original scale ($j = 0$) and the half scaled image ($j = 1$) are displayed in Figure 3.11(b) and (c), respectively. These feature locations are boosted together with all scales. We found that actually there are no features boosted at the scale level 2 and 3, because the images are too small to obtain useful features. Hereafter, we refer to the single scale face model as PCT-BAM (PCT-based boosted appearance model) and the multiple scale face model as MSPCT-BAM.

3.3 Face Alignment

In order to align a PCT-BAM to the face in a given image $\mathbf{I}$, we maximize the classification score function (cf. Equation 3.35) with respect to the shape parameter $\mathbf{p}$. In other words, we need to find the optimal shape parameter, which maximizes the score function. As the score function involves nonlinear image warping, optimizing the objective function is a nonlinear problem. We applied the gradient ascent method to solve this problem in an iterative manner.

The gradient ascent method is a local optimizer, in which a reasonable initialization of the shape parameter $\mathbf{p}_0$ is required. Assuming the current shape parameter $\mathbf{p}$ at the i-th iteration of an alignment procedure, we update $\mathbf{p}$ using the gradient information as follows:

$$\mathbf{p} = \mathbf{p} + \nu \frac{dF}{d\mathbf{p}}, \tag{3.42}$$

where ν is a suitable constant. From Equation 3.8, 3.35, and 3.36, one can calculate the derivative of F with respect to $\mathbf{p}$:

$$\frac{dF}{d\mathbf{p}} = \frac{2}{\pi} \Sigma_{m=1}^{M} \frac{\alpha \Sigma_{i=1}^{K} w_i^m S(\varphi_i^m)(1 - S(\varphi_i^m))[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}]^\top \mathbf{A}_i^m}{1 + [\Sigma_{i=1}^{K} w_i S(\varphi_i^m) + b^m]^2}, \tag{3.43}$$

where $\nabla \mathbf{I}$ is the gradient of the image evaluated at $\mathbf{W}(\mathbf{x}; \mathbf{p})$, and $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is the Jacobian of the warp. The update is repeated several times until the stopping criterion is satisfied. The stopping criterion will be discussed in Section 3.4.

The detailed fitting steps are summarized in Algorithm 2. In the first step (line 3), the shape-free image $\mathbf{I}(\mathbf{W}(\mathbf{x}; \mathbf{p}))$ is calculated with the piece-wise affine warping $\mathbf{W}(\mathbf{x}; \mathbf{p})$. The second step (line 4) is to compute the PCT features for each weak classifier. The third step (line 5) interpolates the gradient of $\mathbf{I}$ at the known warped coordinates $\mathbf{W}(\mathbf{x}; \mathbf{p})$. The fourth step (line 6) is to multiply $\nabla \mathbf{I}$ and the pre-computed $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$. The result $SD = \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is called the steepest descent image, which is an $N \times n$ matrix, where n is the number of shape bases. The fifth step (line 7) is to apply the PCT, which is selected for training a weak classifier, on the SD and project the output with $\mathbf{w}^m$. Basically, the output d_m can be considered as the gradient direction derived from each weak classifier. Its contribution to the final gradient $\frac{dF}{d\mathbf{p}}$ is weighted by $\frac{1}{1+e_m^2}$ in the sixth step (line 8), which combines the weighted gradient directions from each weak classifier. Finally, the shape parameter vector $\mathbf{p}$ is updated (line 9).

Algorithm 2: The face alignment algorithm of PCT-BAM

Data: Input image $\mathbf{I}$, initial shape parameters $\mathbf{p}$, pre-computed Jacobian $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$, the shape model $\mathbf{P}_s$ and the appearance model.

Result: Shape parameters $\mathbf{p}$.

Compute the 2D gradient of the image $\mathbf{I}$.

repeat

- Compute $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ by warping image $\mathbf{I}$ with $\mathbf{W}(\mathbf{x};\mathbf{p})$;
- For each weak classifier compute the feature: $e_m = \Sigma_{i=1}^{K} w_i S(\varphi_i^m) + b^m; m = 1, 2, \ldots, M$;
- Interpolate the gradient of image $\mathbf{I}$ at $\mathbf{W}(\mathbf{x};\mathbf{p})$ with bi-linear interpolation ;
- Compute the steepest descent $SD = \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$;
- Compute the PCT feature from each column of SD and project with $\mathbf{w}^m$: $\mathbf{d}_m = \alpha \Sigma_{i=1}^{K} w_i^m S(\varphi_i^m)(1 - S(\varphi_i^m))[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}}]^\top \mathbf{A}_i^m; m = 1, 2, \ldots, M$;
- Compute $\Delta \mathbf{p}$ using $\Delta \mathbf{p} = \nu \frac{2}{\pi} \Sigma_{m=1}^{M} \frac{\mathbf{d}_m}{1+e_m^2}$;
- Update $\mathbf{p} = \mathbf{p} + \Delta \mathbf{p}$;

until $\|\mathbf{s}(\Delta \mathbf{p})\| \leq \tau$;

3.4 Experiments

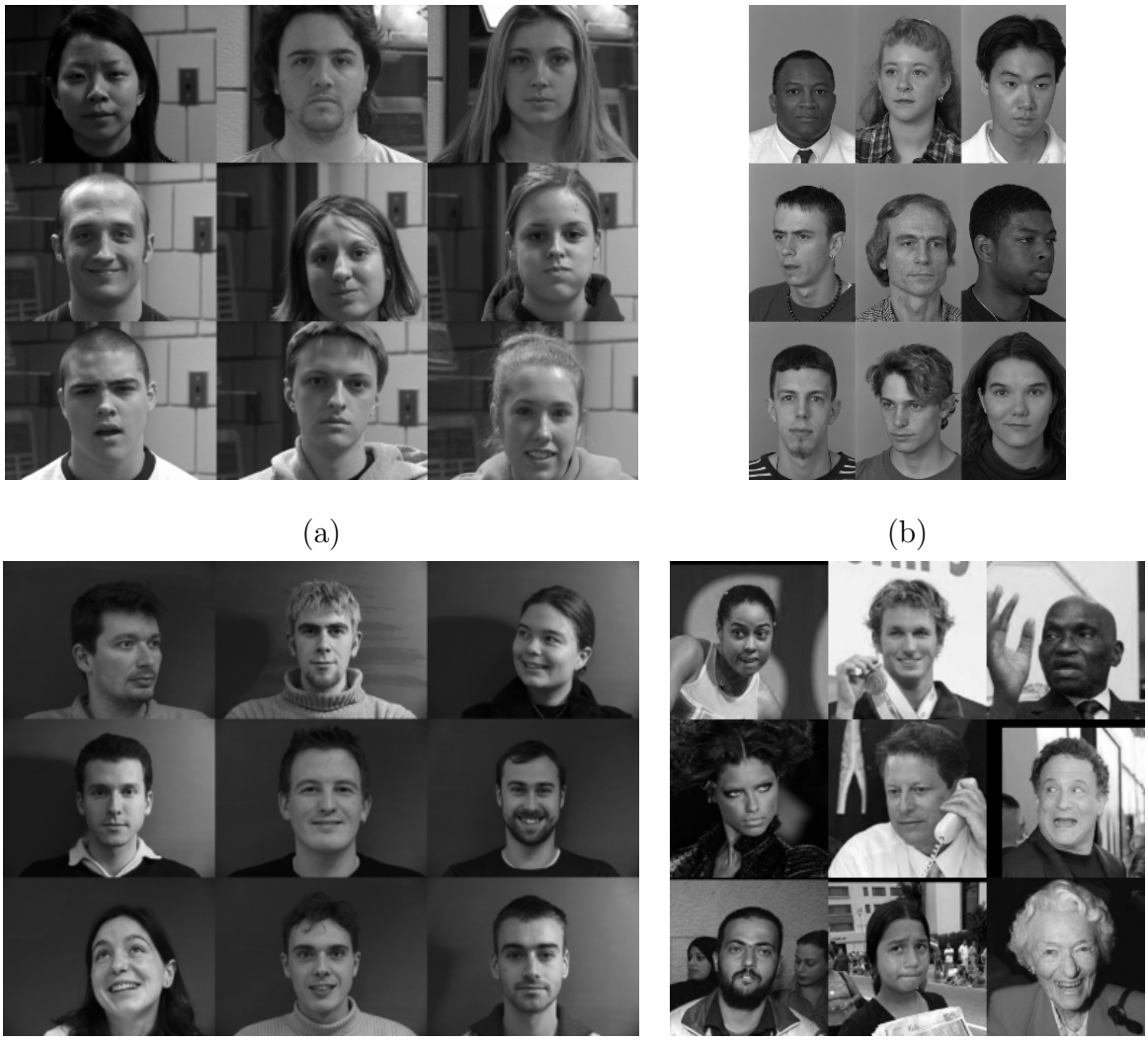
To assess the effectiveness of the proposed model, we evaluate face alignment on a set of annotated images. Labeling facial landmarks on face images is a tedious and time-consuming work. There are some annotated databases available so far, such as the IMM database [SEL03] and the XM2VTS database [MMK+99], which are annotated with 58 and 68 landmarks respectively. However, the number of subjects in the IMM database is rather limited, while the variation in the XM2VTS database is not sufficient. Recently, there are a few annotated databases released, which contain face images downloaded from the web using simple text queries on the sites such as google.com, flickr.com, and yahoo.com [BJKK11, KWRB11]. However, the images in these databases are labeled with sparse facial landmarks, the corresponding shape-free images may contain severe artifacts due to the nature of piece-wise affine warping. Due

to this reason, we collect a set of facial images with sufficient variations and annotate each image with relatively dense facial landmarks.

3.4.1 Evaluation Data Set and Procedure

The data set for evaluation in this work contains 1529 images. These images are collected from multiple publicly available databases, including the FRGC v2.0 database [PFS+05], the FERET database [PWHR98], the IMM database [SEL03], and the Labeled Faces in the Wild (LFW) database [HRBLM07]. Figure 3.12 shows sample images from these four databases. The collected images are partitioned distinctively into four subsets. Table 3.1 lists the properties of each database and partition. Set 1 includes 400 images (one image per subject), where 200 images are from the FRGC database and the other 200 images are from the FERET database. Set 1 is used as the training set. Set 2 includes 389 images from the same subjects but different images than the FRGC database in Set 1. Set 3 includes 240 images from 40 subjects in the IMM database that were never used in the training. Set 4 includes randomly selected 500 images of 500 subjects from the LFW database. This partition ensures that we have two levels of generalization to be tested, *i.e.*, Set 2 is tested as the unseen data of seen subjects; Set 3 and 4 are tested as the unseen data of unseen subjects. Set 4 is a particularly challenging data set, since it is collected from the Internet. The images were captured under cluttered background and various real-world illumination environments using different types of cameras. There are 58 manually labeled landmarks for each of the 1529 images. An example of annotated image is shown in Figure 3.1. The images are down-sampled such that the facial width is roughly 40 pixels across the set in order to speed up the training process.

We compare our proposed PCT-BAM and MSPCT-BAM to the Haar feature-based BAM (Haar-BAM). AAM is also compared, although it has already been shown in [Liu07] that the Haar-BAM outperforms AAM. We train the appearance models with Set 1 by taking the shape-free images extracted with ground truth landmarks as the positive samples and the negative samples are generated by perturbing the shape parameters of the ground truth shapes uniformly in a range of the corresponding deviations. We generate 10 negative samples for each image and in total 4000 negative samples are obtained. The shape model has 15 shape bases, which preserve 95% of shape variations. We use the same mean shape size as in [Liu07]; that means the size of shape-free images is 30×30 pixels. The resulting appearance models contain 50 weak classifiers, where in the PCT-based models, the linear SVM is used for learning weak classifiers. Note that without further specification, PCT-BAM always refer to a BAM in which the linear SVM is applied for training weak classifiers based on the PCT-features.



(a) (b)

(c) (d)

Figure 3.12: Sample images from the face data sets: (a) FRGC v2.0 database; (b) FERET database; (c) IMM database; and (d) LFW database.

	FRGC	FERET	IMM	LFW
Images	589	200	240	500
Subjects	200	200	40	500
Variation	Expression, lighting	Pose	Pose, expression, lighting	All
Set 1	200	200		
Set 2	389			
Set 3			240	
Set 4				500

Table 3.1: Summary of the data set.

The texture model in the AAM has 75 bases, which also preserve 95% of texture variations. We use the Simultaneous Inverse Compositional method [GMB05] for the AAM fitting.

The false alarm rate (FAR) of the strong classifiers of the three models are plotted in Figure 3.13. The FAR is plotted as a function of the number of weak classifiers, when the miss-detection rate on the training set is set to 0%. The plot shows that both PCT-based models converge faster than the Haar-BAM. In particular, for 50 weak classifiers the FAR's of MSPCT-BAM and Haar-BAM are 1.17% and 7.15%, respectively.

A faster convergence means that it is less likely to have local maxima on a classification score surface. Figure 3.14(a) shows that for a given image, a concave surface of classification scores can be observed, while perturbing the shape parameters along two shape bases. The x-axis and y-axis correspond to the perturbation indexes for the 4th and 5th shape bases, respectively. The z-axis corresponds to the classification score. The concavity property of the score surface ensures that the gradient ascent algorithm can perform well.

The perturbation range is set to be 1.6 times the deviation of these two bases. When the perturbation is at the maximal amount for two bases, the corresponding four perturbed landmarks are plotted at Figure 3.14(b). To see the properties of score surfaces, more surfaces are plotted as images in Figure 3.14(c), where the intensity corresponds to the classification score. Each sub-image is generated in the same way as in Figure 3.14(a). In most cases, we see the intensity changes from high to low, when the pixel deviates from the center, *i.e.*, the alignment gets less accurate. This monotonic surface is important for a successful face alignment algorithm.

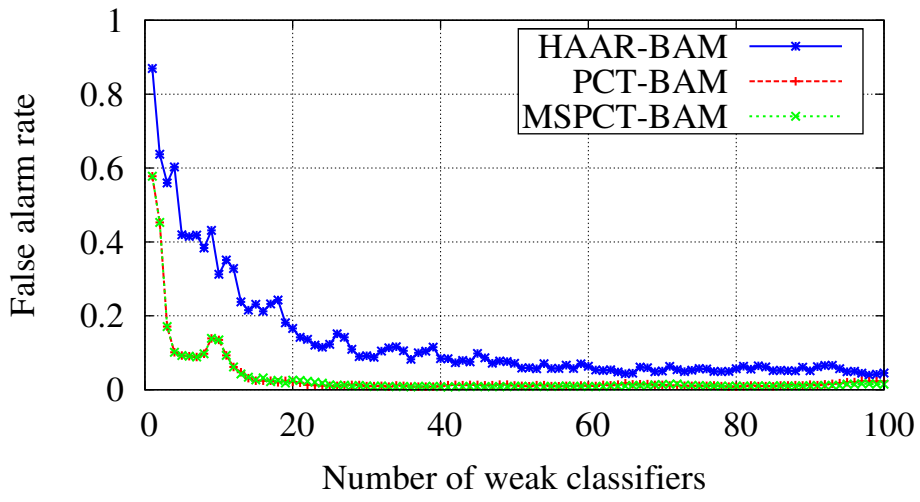


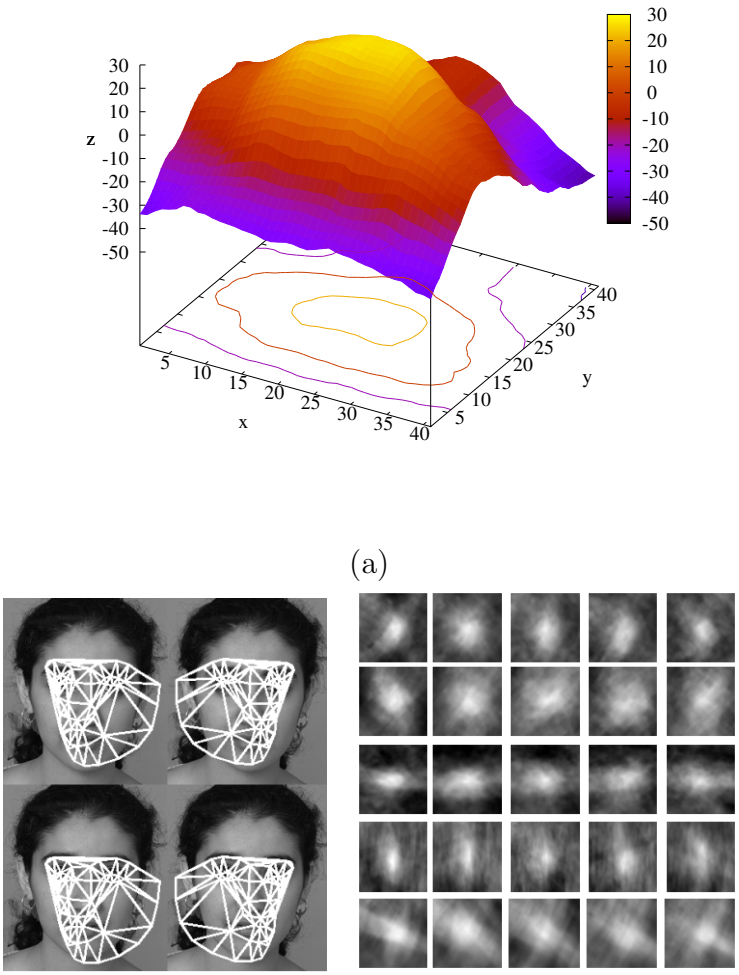
Figure 3.13: False alarm rate of the strong classifiers, when the miss-detection rate on the training set is set to 0%.

3.4.2 Experimental Results

3.4.2.1 Evaluation Metrics

In the evaluation, we use the randomly perturbed ground truth landmarks to initialize each alignment. In order to perform a statistical evaluation of the results, we repeat the random perturbation multiple times on each test image. The initial position of the landmarks is generated by perturbing the shape parameter with independent Gaussian noise with variances multiple of the corresponding deviations. An alignment is considered as converged if the Root Mean Square Error (RMSE) between the aligned landmarks and the ground truth is less than one pixel. For the converged trails, we use two metrics to measure the robustness and accuracy of the alignment. The Average Frequency of Convergence (AFC), which assesses the robustness of the alignment is calculated as the number of converged trials divided by the total number of trials. The second metric is the histogram of the RMSE (HRMSE) of the converged trials, which measures how close the aligned landmarks are to the ground truth.

The evaluation in face alignment using different models is conducted under the same conditions. All algorithms are initialized with the same set of randomly perturbed landmarks. The same step-size ν in Equation 3.42 is used for all BAMs. A common termination condition is used. That is, if the number of iterations is larger than 55 or the RMSE between consecutive iterations is less



(a)

(b) (c)

Figure 3.14: (a) The classification score surface, while perturbing the shape parameters in the neighborhood of the ground truth along the 4th and 5th shape bases; (b) The four perturbed facial landmarks, when the perturbations are at the four corners of the surface above; (c) The classification score surface of five facial images (one by each column), while perturbing the shape parameters along pairs of shape bases (from top to bottom (p_1, p_2), (p_2, p_3), (p_3, p_4), (p_4, p_5),(p_5, p_6)).

than 0.025 pixels. Figure 3.15 plots the AFC of the PCT-BAM, MSPCT-BAM, Haar-BAM and AAM-SIC against the level of the initial landmarks perturbation, computed over Set 1, 2, 3, and 4, respectively. For each perturbation level, we randomly perturb each image of each set five times.

3.4.2.2 Comparison

The AFC plots in Figure 3.15 show that all the discriminative BAMs achieve better alignment results than AAM over the four data sets, which again demonstrates the effectiveness of the discriminative models in face alignment. The results also show that the MSPCT-BAM-based alignment achieves comparable results on the seen data (Set 1 and 2). The robustness of the MSPCT-BAM-based alignment is slightly better than Haar-BAM based alignment with increasing perturbing variance. However, in the experiments on unseen data (Set 3 and 4), the PCT-based (both PCT-BAM and MSPCT-BAM) alignment outperforms the Haar-BAM-based alignment significantly as plotted on the third and fourth row in Figure 3.15. On Set 3, the convergence rate in the MSPCT-BAM fitting is slightly better than Haar-BAM at 0.2σ perturbation level. However, when the perturbation range increases to 1.6σ, the AFC value of MSPCT-BAM is 13% higher than Haar-BAM. On the most challenging testing set (Set 4), in which the imaging conditions are totally different from each other, the performance of both algorithms degrades a lot. However, the performance drop of PCT-BAM (11%) is less than that of Haar-BAM (22%) at the first perturbation index. Additional PCT features selected on other scales also improve the robustness of alignment with large perturbation as can be observed consistently through all the experiments. The accuracies of the three methods are comparable as we can see from the HRMSE plots. The PCT-BAM is slightly superior to Haar-BAM again on the unseen data as displayed in Figure 3.15. Overall, our PCT-based alignment has a better generalization capability than the Haar-BAM-based alignment.

The reason for the performance gains on unseen data is probably that the responses of the PCT filter are somewhat similar to the Laplacian filter, which is a high-pass filter. The 3×3 filter mask in the center of Figure 2(b) is indeed a discretized Laplacian filter. Thus, the corresponding filter responses are less sensitive to illumination changes, which make the PCT-based approach generalize better on unseen data with mismatched illumination conditions.

3.4.2.3 Model Parameters

In this section, we analyze the effects of different parameters in building the boosted classification appearance models.

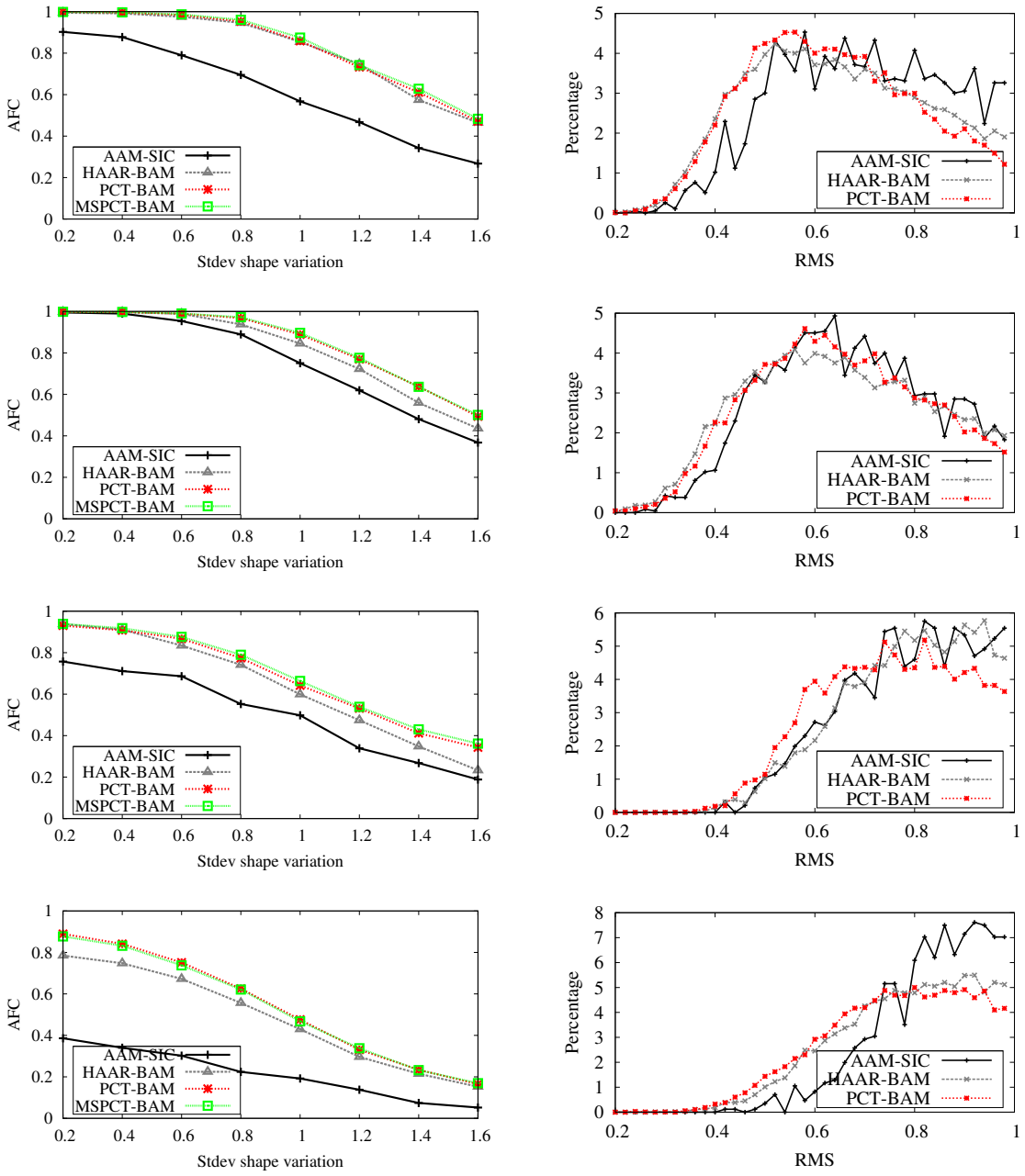


Figure 3.15: Alignment results of different algorithms on Set 1, 2, 3, and 4. From top to bottom, each row corresponds to the results on one set. The left column plots the AFC curves and the right column plots the HRMSE.

	LDA	LR	SVM	LDA	LR	SVM
Perturbation	0.8σ			1.6σ		
Set 1	0.941	**0.957**	0.953	0.446	**0.489**	0.471
Set 2	0.965	**0.971**	0.968	0.490	**0.497**	0.493
Set 3	0.769	**0.778**	0.774	0.330	0.336	**0.343**
Set 4	0.616	0.621	**0.625**	0.166	**0.171**	0.165

Table 3.2: Comparison of alignment performance in AFC rates applied with different types of linear weak classifiers. The AFC rates are obtained at 0.8σ and 1.6σ perturbation levels for initialization.

3.4.2.3.1 Type of Weak Classifier Three different linear model-based weak classifiers are reviewed in Section 3.2.2.2. The cost sensitive method is applied in the training phase of these weak models. We compare the alignment results achieved from the appearance models based on these weak models. Table 3.2 lists the alignment convergence rates on the four evaluation data sets at two levels of shape perturbation, namely, at 0.8σ level and 1.6σ level. From this table, we observe that the logistic regression (LR)-based weak model outperforms the other two models in most cases. Yet on Set 3, SVM achieves a slightly better AFC rate at 1.6σ perturbation level than LR. The results of LDA-based weak model are the worst over all sets and perturbation levels, due to its generative modeling and additional parameters estimation for Gaussian models. However, the differences between the results are not very high. The largest absolute performance difference in AFC rate is 4.3%, which is between LDA and LR on Set 1 at the highest perturbation level.

3.4.2.3.2 Number of Weak Classifiers The number of boosted weak classifiers is an important parameter for learning a good appearance model. On one hand, we want to select more distinct features to improve the representation power of the model. On the other hand, we also want to avoid overtraining. Also, the increased number of weak classifiers increases the computation load for model fitting. We plot the alignment results in Figure 3.16 with an increasing number of weak classifiers selected. The appearance model uses linear SVM as weak classifier. The number of weak classifiers (or selected features) ranges from 10 to 100. Average convergence rates are reported at 0.8σ and 1.6σ perturbation levels. From Figure 3.16(a) and (b), we can see that the alignment performance increases drastically, when less than 50 features are selected. Af-

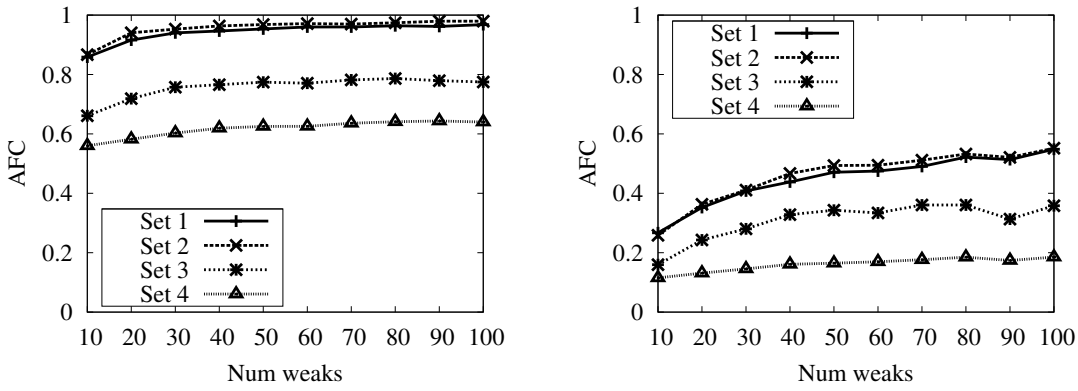


Figure 3.16: Effects of different number of weak classifiers in PCT-BAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

terwards, the convergence rates are still increasing, but not very remarkably. On Set 3 and Set 4, the alignment performance stops to increase as the number of selected features is approaching 100. This shows the evidence that the learned appearance model tends to be overtrained, when more weak classifiers are added.

3.4.2.3.3 Size of Reference Shape Until now, the appearance models are trained using a reference shape, which has a width of 30 pixels. As the hypothesis space of the PCT feature is low, it is tractable to train models with a larger reference shape. We analyze the effects of reference shape size for alignment by changing the size ranging from 20 to 60 pixels. Note the size of a reference shape corresponds to the scale of the masked shape-free image. We train the appearance models with linear SVM as weak classifiers and in total 100 weak classifiers are boosted. The alignment results are compared in Figure 3.17(a) and (b), at 0.8σ and 1.6σ perturbation levels, respectively. We observe that the alignment performance is enhanced with enlarged mask size. The alignment AFC rate with 1.6σ perturbation on Set 3 approaches to its peek of 42%, when the mask image has a width of 35 pixels. When the mask size increases further, the performance starts to decrease again. This observation suggests that a larger reference shape provides more detailed and distinctive feature for learning a good appearance model. However, an overlarge mask may introduce noise in the appearance modeling, due to the locality property of the PCT feature.

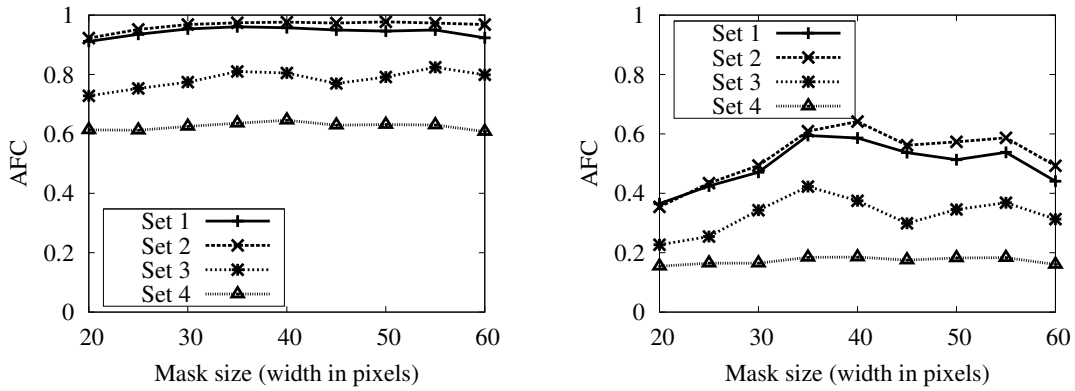


Figure 3.17: Effects of different size of masks (width in pixels) in PCT-BAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

Furthermore, the hypothesis space is also extended with increasing mask size, therefore, boosting 100 features might be suboptimal. We leave this for future study.

3.4.2.3.4 Effects of Sampling We finally discuss the effects of the training samples for learning classification based appearance models. In particular, we address the imbalanced data problem in model training. First, we show the necessity of generating more negative training samples for covering the variations of image background. Figure 3.18 demonstrates the alignment results for the appearance models trained with an increasing number of negative samples. The horizontal axis indicates the ratio of negative samples to positive samples. From the plots, we see that generating one negative sample from one labeled image is not enough. Setting the ratio to 10 is already a good number, yet the performance gain is limited, if a higher ratio is used. Note that the results plotted in Figure 3.18 are achieved with the cost sensitive method applied.

In addition, we also show the effectiveness of oversampling for augmenting the positive training samples. Figure 3.19 compares two oversampling methods. AUG+I denotes the first method as described in Section 3.2.3.2.1, where the positive training samples are augmented with slightly perturbed ground truth data. AUG+II denotes the second method, where synthetic positive samples are generated. Note for both methods, 10 positive and negative samples are

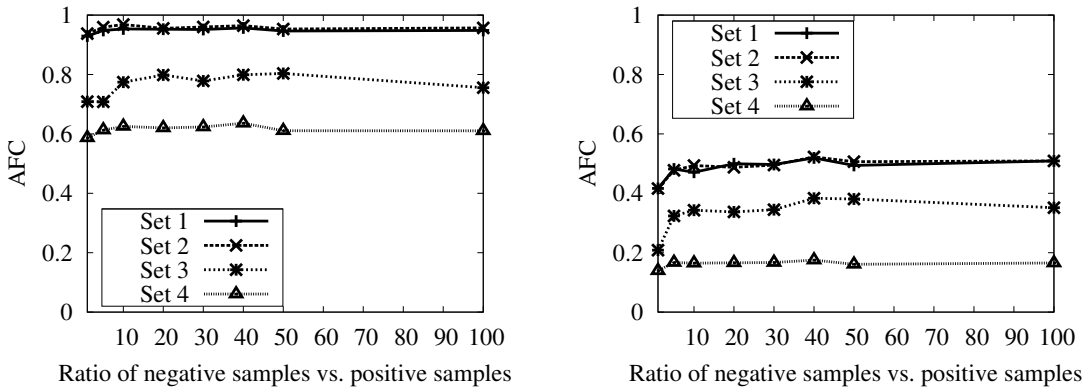
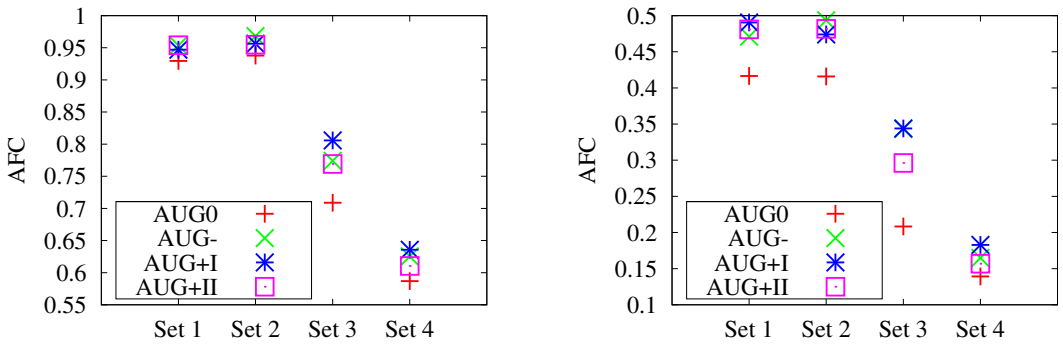


Figure 3.18: Effects of training sample ratio in PCT-BAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

generated with each labeled image. The method AUG0 means no sample augmentation is applied, *i.e.*, one labeled image generates a pair of positive and negative samples. The method AUG- corresponds to the cost sensitive approach, in which a sample ratio of 10 is used. From the results, we learn that the first oversampling method is a good choice for handling the imbalanced data problem. The cost sensitive method is suboptimal, but the differences are minor.

3.5 Conclusions

We introduce the PCT-based boosted appearance model (PCT-BAM), a new discriminative appearance model, which is found to be suitable for robust face alignment. The adopted PCT feature has a much smaller parameter configuration space, which enables efficient model training compared to the training procedure of the Haar-BAM. We compared the proposed PCT-based alignment to the Haar-BAM on seen data and unseen data. Our experimental results on seen data are slightly better. However, our PCT-BAM model shows significant performance improvement on unseen data, which means that the proposed model has a better generalization capability on unseen data. Additional PCT features selected on other scales also improve the robustness of alignment with large perturbation as can be observed consistently through all the experiments.



(a) (b)

Figure 3.19: Effects of training sample augmentation in PCT-BAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

4 Ranking Appearance Models

In this chapter, a ranking based discriminative appearance model is presented. We start this chapter with a short motivation of formulating the appearance modeling as a learning to rank problem in Section 4.1. Section 4.2 and 4.3 describe the details of learning a ranking face model and training data generation. The experimental results for assessing the effectiveness of the ranking-based model are given in Section 4.5. We give concluding remarks for this chapter in Section 4.6.

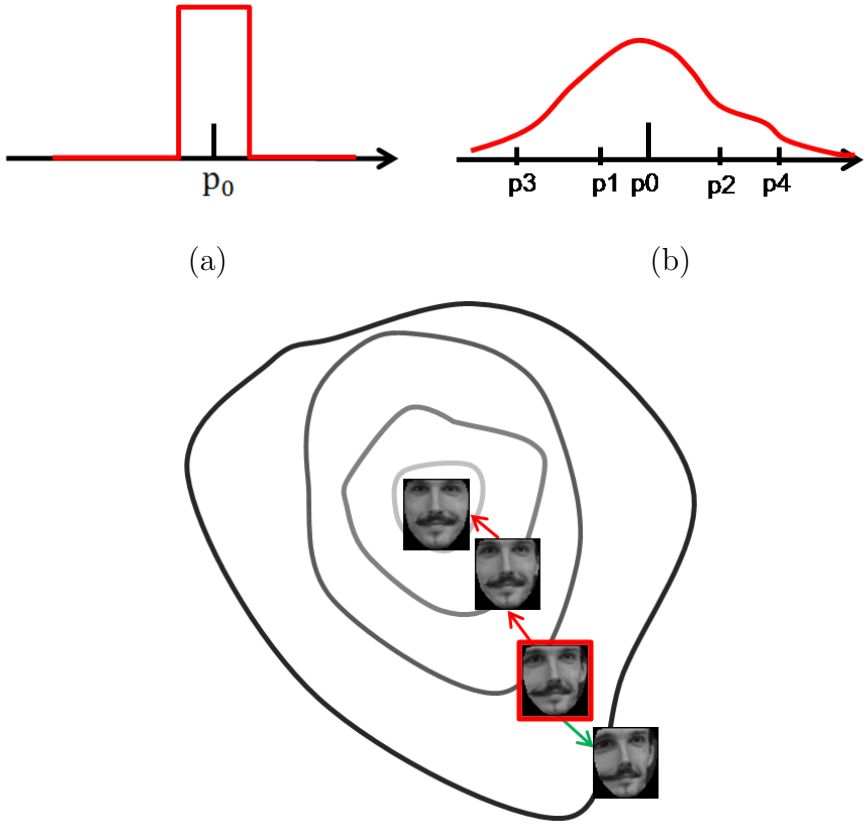
4.1 Introduction

The proposed classification-based appearance model has shown its improved robustness and generalization ability compared to AAM and the Haar-based model. However, it suffers from the imbalanced training data problem as has been shown in Section 3.2.3.2. We propose a few solutions to mitigate the problem by applying the sampling or cost sensitive method. However, avoiding the imbalanced problem by reformulating the model learning is a more elegant solution. On the other hand, as shown in Figure 4.1(a), the classification based score function might not be smooth enough due to the nature of the classification loss functions. As the classification loss functions do not distinguish different negative samples, the resulting score function might be flat or contains many local maxima at the locations that are far from the true shape. This leads to a slow convergence, if the local minimizer uses a fixed step size and often the fitting can easily get stuck in local extrema.

In this chapter, we present another discriminative appearance model, which is based on learning ranking models. Instead of distinguishing the correctness of alignments, the ranking models infer the order of two paired alignments. The ranking-based appearance model (RAM) is trained by boosting a score function in a pairwise ordinal classification way. This model ensures that the score function returns a higher value, if the current alignment is closer to the ground truth than the others in the shape parameter space (cf. Figure 4.1(b)). Figure 4.1(b) illustrates the iso-contours of a ranking score function in the shape parameter space. The superimposed shape-free images reflect the preference of alignments.

A local optimizer benefits from such a model as the gradient of the learned score function is constrained to the same direction towards the ground truth. The proposed ranking appearance model shares a similar idea as presented in [WLD08, ZZCM08], however, we apply the pairwise RankSVM [HGO00] over the PCT features to build weak rankers and the final strong ranking function is obtained by selecting and combining weak rankers in a boosting framework.

We demonstrate in the experiments that the ranking based appearance model achieves better alignment convergence rates than the classification based model.



(a) (b)

(c)

Figure 4.1: (a) Classification cost function; (b) Ranking cost function; (c) Learning preference (partial ordering).

4.2 Face Model

The presented face model in this chapter includes a shape model and an appearance model. The shape model is a generative model, which is also applied in Chapter 3, as well as in many other statistical deformable models [CT92, CET98a]. The appearance model in this chapter is constructed with the application of a ranking model.

4.2.1 Appearance Model

As with the classification-based appearance model, we define the ranking-based appearance model on the shape-free images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$. Here, $\mathbf{W}(\mathbf{x};\mathbf{p})$ is a nonlinear warping function defined by the shape parameters $\mathbf{p}$. In other words, a shape-free image is a shape normalized image, which is also parametrized by $\mathbf{p}$. We extract a set of local features on the shape-free images and learn a ranking function $F(\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p})))$, which predicts the preference of alignments.

The PCT features are adopted as our feature representation, which show superior alignment performance in Chapter 3 compared to the Haar features. For the detailed description of the proposed PCT feature, please refer to Section 3.2.2. For simplicity, we again fix the parameter $K = 9$, *i.e.*, we extract a PCT feature in a 3×3 local neighbourhood.

4.3 Learning Ranking Appearance Models

Learning ranking models is a supervised or semi-supervised machine learning problem, which is becoming a popular research topic due to the increasing demand on internet services, such as information retrieval and recommender systems. The problem differs from other supervised machine learning problems, such as classification and regression in the characteristic of the output space and the loss function. The classification output space is a finite unordered set and usually the $0-1$ loss is used. The output space for regression is a metric space, *i.e.* a set of real numbers. The $L2$ metric is often used for defining a loss function. The output space for ranking is also a finite set, however, there exists a partial ordering among the elements. As there is no metric defined on this space, we cannot use metric loss function. The simple $0-1$ loss for classification problem cannot reflect the partial ordering in the ranking output space.

According to [Liu09], the learning to rank problems can be categorized into three groups by their input representation and loss function: (a) pointwise approach,

(b) pairwise approach, and (c) listwise approach. The pointwise approach associates each training sample with a numerical or ordinal score. This approach is approximated with conventional machine learning methods, such as ordinal classification [LBW07b] or regression [ZCSZ07]. In Chapter 5, we discuss an appearance model based on ordinal regression. We categorize it as a regression model due to the defined loss function. The pairwise approach learns a binary classifier, which determines the preference between a given pair of data samples. The goal is to minimize the average number of swaps in ranking. There are many successful works, which are based on the pairwise approach, such as RankSVM [HGO00] and BoostRank [FISS03]. The listwise approach tries to optimize directly some evaluation metrics, such as mean average precision. Examples of such approach are ListNet [CQL$^+$07] and many of its extensions.

As we are not interested in optimizing any measures for information retrieval, we focus on minimizing the error of pair swapping. Hence we adopt a pairwise approach for learning our ranking based appearance model. In particular, we use a pairwise ordinal classification based method.

4.3.1 Pairwise Ordinal Classification-based RAM

The pairwise approach does not focus on accurately predicting the preference degree of each data sample (in this case alignment). It cares about the relative order between two alignments, which reflects relative preferences. In this sense, it is closer to the concept of "ranking" than the pointwise approach.

As mentioned before, the pairwise approach is usually approximated with a classification problem on pairs of alignments, *i.e.*, to determine which alignment in a pair is preferred. In other words, the goal of learning is to minimize the number of miss-ordered alignment pairs. Note that this pairwise classification differs from the classification in the pointwise approach, since it operates on every two alignments under investigation.

The input space of the pairwise classification approach contains pairs of alignments; both are represented by the feature vectors extracted on their corresponding shape-free images. The output space contains the pairwise preferences $\mathcal{Y} \in \{+1, -1\}$ between each pair of alignments. The hypothesis space contains bi-variate functions h that take a pair of alignments as input and output the relative order between them. In this work, we use a scoring function (ranking function) f to define the hypothesis, *i.e.* $h(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = 2 \cdot I_{f(\mathbf{x}^{(1)}) > f(\mathbf{x}^{(2)})} - 1$, or to be more precise:

$$h(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = \begin{cases} +1 & f(\mathbf{x}^{(1)}) - f(\mathbf{x}^{(2)}) > 0 \\ -1 & f(\mathbf{x}^{(1)}) - f(\mathbf{x}^{(2)}) \leq 0 \end{cases} \tag{4.1}$$

The loss function measures the inconsistency between $h(\mathbf{x}^{(1)}, \mathbf{x}^{(2)})$ and the ground truth label $y_{(1),(2)}$. The classification loss is usually expressed with the differences $(f(\mathbf{x}^{(1)}) - f(\mathbf{x}^{(2)}))$.

Similarly as in Chapter 3, we learn a score function in a boosting framework as our appearance model. The score function is a combination of a set of selected weak ranking functions.

4.3.2 Weak Ranking Function

The weak ranking function is learned with pairwise ordinal classification with the pairs of extracted feature vectors (in this case the PCT features). We use RankSVM [HGO00] for learning our weak ranking functions.

RankSVM applies the maximum margin principle to perform pairwise classification. Given a set of paired feature vectors $(\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)})$ extracted from paired alignments, where $i = 1, \ldots, N$, and the corresponding ground truth label y_i, the mathematical formulation of RankSVM is shown below, where a linear scoring function is used, *i.e.*, $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$,

$$\min \frac{1}{2}\|\mathbf{w}\|^2 + \lambda \sum_{i=1}^{N} \sum_{y_i} \xi_i \tag{4.2}$$

$$s.t. \quad \mathbf{w}^\top (\mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)}) \geq 1 - \xi_i, \quad \text{if} \quad y_i = 1, \tag{4.3}$$

$$\xi_i \geq 0, \quad i = 1, \ldots, N. \tag{4.4}$$

As we can see, the objective function in RankSVM is very similar to that of in SVM (cf. Section 3.2.2.2.3), where the term $\frac{1}{2}\|\mathbf{w}\|^2$ controls the complexity of the model $\mathbf{w}$. Minimizing this term also corresponds to maximizing the margins between different rank levels. The difference between RankSVM and SVM lies in the constraints, which are constructed from alignment pairs. The loss function in RankSVM is a hinge loss defined on data pairs. For example, for a training pair, if alignment $\mathbf{x}^{(1)}$ is labeled as being better than alignment $\mathbf{x}^{(2)}$, *i.e.*, $y = 1$. Then, if $\mathbf{w}^\top \mathbf{x}^{(1)}$ is larger than $\mathbf{w}^\top \mathbf{x}^{(2)}$ by a margin of 1, there is no loss. Otherwise, the loss will be ξ.

Since RankSVM is well rooted in the framework of SVM, it inherits desirable properties of SVM. For example, with the help of margin maximization, RankSVM can have a good generalization ability. Kernel tricks can also be applied to RankSVM, so as to handle complex non-linear problems. In this study, however, we focus on the linear kernel for easing the derivation of the alignment algorithm.

4.3.3 Boosting A Strong Ranking Function

The RankSVM described above provides a linear ranking function $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$, which predicts the absolute preference degree of an alignment with its corresponding shape parameter $\mathbf{p}$. To learn a strong ranking function, we select and aggregate the most discriminative weak ranking functions in a boosting framework. We define a weak ranking function as follows:

$$f_m(\mathbf{p}) = \frac{1}{\pi} atan(\mathbf{w}^{m\top} S(\boldsymbol{\varphi}^m) - t^m). \tag{4.5}$$

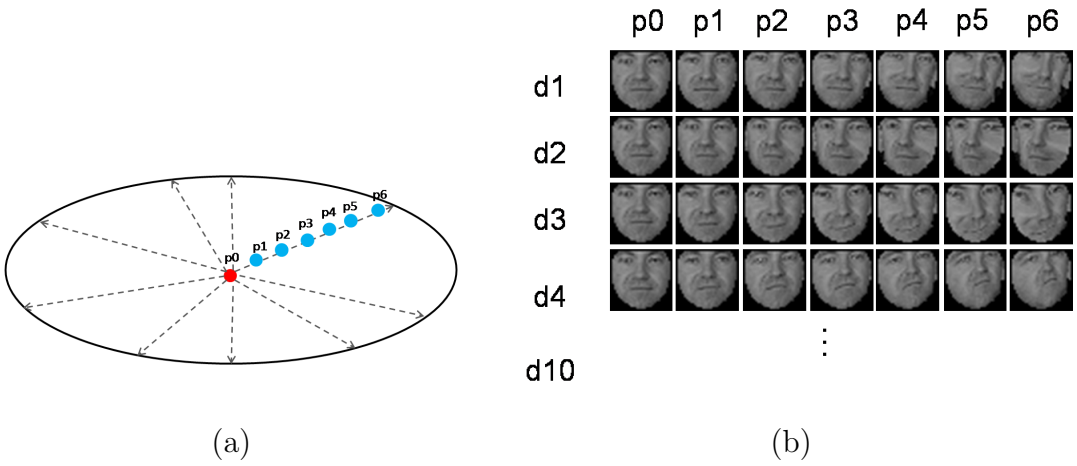
Note that $f_m(\mathbf{p})$ is continuous within $(-0.5, 0.5)$, the $atan()$ function is used to ensure both discriminability and derivability. $\boldsymbol{\varphi}^m$ is a PCT vector as defined in Section 3.2.2. The $S(\cdot)$ is a sigmoid function, which normalizes the raw PCT feature values into a range of $(0, 1)$ before the linear projection defined by a projection vector $\mathbf{w}^m$ learned with RankSVM. The threshold t^m needs to be determined during boosting. The strong ranking function is assumed to be an additive model:

$$F(\mathbf{p}) \doteq \sum\nolimits_{m=1}^{M} f_m(\mathbf{p}). \tag{4.6}$$

With the ranking function given above, we define the strong hypothesis function for pairwise classification: $H(\mathbf{p}_1, \mathbf{p}_2) = \text{sign}[F(\mathbf{p}_1) - F(\mathbf{p}_2)]$, *i.e.* $H(\mathbf{p}_1, \mathbf{p}_2) = +1$ if $\mathbf{p}_1 \succ \mathbf{p}_2$, else $H(\mathbf{p}_1, \mathbf{p}_2) = -1$. We assume H to be an additive model: $H = \sum_{m=1}^{M} h(\mathbf{p}_1, \mathbf{p}_2)$, where $h_m(\mathbf{p}_1, \mathbf{p}_2) = f_m(\mathbf{p}_1) - f_m(\mathbf{p}_2)$. The Gentleboost algorithm is applied for boosting the strong pairwise classifier, and eventually the strong ranking function.

4.3.4 Training Data for Learning

To learn the strong ranking function F, we sample ordering pairs from a training data set containing D facial images with annotated landmarks. For each of the training images, we randomly perturb the ground truth $\mathbf{p}_i$ in U different directions $\{\Delta\mathbf{p}_{iu}\}_{u=1,\ldots,U}$. In each direction we evenly sample V shape parameters $\{\mathbf{p}_i + v \times \Delta\mathbf{p}_{iu}\}_{v=1,\ldots,V}$. Note the samples generated in one perturbation direction result in a list of fully ordered data, *i.e.* $\mathbf{p}_i \succ \mathbf{p}_{iu1} \succ \ldots \mathbf{p}_{iuV}$, which corresponds to a correctly ranked query in the context of information retrieval. Figure 4.2(a) illustrates the applied scheme for data sampling. Note that $\mathbf{p}_0$ stands for the ground truth shape parameter. The ellipse limits the range for parameter perturbation and the rays denote the random directions of perturbations. Figure 4.2(b) shows examples of shape-free-images extracted using the perturbed ground truth shapes. Each row shows different random perturbation directions and the columns show increasing perturbation levels in each direction. From each direction, we can generate V ordinal adjacent pairs using the samples



(a) (b)

Figure 4.2: (a) Sampling scheme; (b) Generated training samples.

including the ground truth. In total, $N = D \times U \times V$ ordinal pairs are generated. We denote each of the pairs as $\{\mathbf{x}_\ell = (x_\ell^{(1)}, x_\ell^{(2)})\}_{\ell=1,\ldots,N}$, where $x_\ell^{(1)} \succ x_\ell^{(2)}$ and their corresponding label as $z_\ell = +1$. The negative samples can be generated by reversing the order of each pair and assigning corresponding label with -1. However, due to the definition of the hypothesis function for the pairwise ordinal classification (cf. Equation 4.1), the negative pairs have the same impact as the positive pairs. Hence we do not include the negative pairs to reduce the redundancy and speed up training. The boosting procedure is summarized in Algorithm 3. Equation 4.7 denotes that in each boosting iteration, a weak ranking function f_m is found by fitting weighted least squares.

4.4 Face Alignment with Rank Appearance Model

We use gradient ascent method for model fitting in a similar way to Chapter 3. Fitting the learned model to a novel image is done by maximizing Equation 4.6 with respect to the shape parameter $\mathbf{p}$. A fixed step size is used in the iterative optimization.

Algorithm 3: PCT-RAM Learning

Data: Training samples, with labels $\{z_\ell = +1\}$

Result: The alignment score function F

Initialize the weights $w_\ell = \frac{1}{N}$ and the score function $F = 0$

foreach *m=1,. . . ,* M **do**

  Fit f_m with weighted least squares, such that

$$f_m = \arg\min_f \sum\nolimits_\ell w_\ell (z_\ell - h(\mathbf{x}_\ell))^2 \tag{4.7}$$

  where $h(\mathbf{x}_\ell) = f(x_\ell^{(1)}) - f(x_\ell^{(2)})$

  $F \leftarrow F + f_m$

  $w_\ell \leftarrow w_\ell \exp(-z_\ell h_m(\mathbf{x}_\ell))$

  Normalize the weights such that $\sum_\ell w_\ell = 1$

return $F = \sum_{m=1}^{M} f_m$

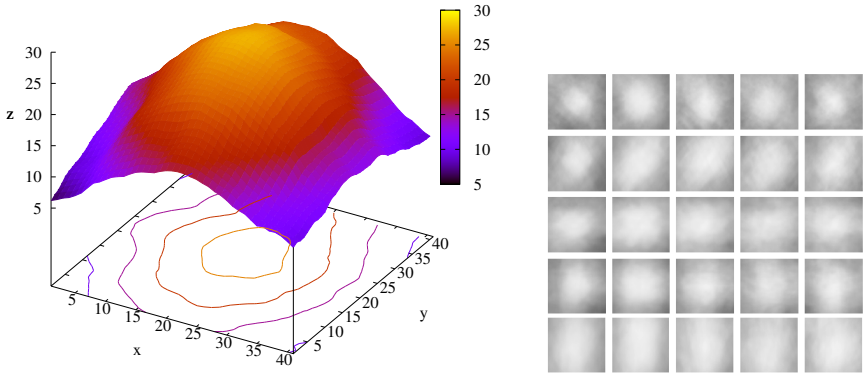
4.5 Experiments

4.5.1 Data and Setup

For evaluating face alignment using the proposed appearance model, we use the data sets presented in Section 3.4.1. Set 1 is used for training the shape and appearance models, and testing is conducted on all the four data sets for analyzing generalization capability at different levels.

We denote the proposed appearance model as PCT-SVM-RAM, as it uses RankSVM as weak classifier and PCT as feature representation. We train a shape model with 15 components preserving 95% of shape variations. The size of the shape-free images is 30×30 pixels. For each annotated training image, we select $U = 10$ random directions and in each direction $V = 6$ positions are evenly sampled. Including the position at ground truth, in total 6 adjacent ordinal pairs can be generated. The overall training set includes $N = 24000$ $(400 \times 10 \times 6)$ ordinal pairs. The resulting ranking appearance model includes 100 weak ranking functions.

In testing, we randomly perturb ground truth landmarks at different noise levels for initializing each alignment. We repeat the random perturbation for each noise level multiple times on each test image in order to perform a statistical evaluation of the results. A fitting is considered as converged if the Root Mean Square Error (RMSE) between the aligned landmarks and the ground truth is less than one pixel. The Average Frequency of Convergence (AFC) is used as the evaluation metric, which assesses the robustness of the alignment. The same termination condition is applied for the fitting procedure as in Section 3.4.2.1.



(a) (b)

Figure 4.3: (a) The ranking score surface while perturbing the shape parameters in the neighborhood of the ground truth along the 4th and 5th shape bases (in the same way as in Figure 3.14(a)); (b) The ranking score surface of 5 facial images (one by each column) while perturbing the shape parameters along pairs of shape bases (from top to bottom (p_1, p_2), (p_2, p_3), (p_3, p_4), (p_4, p_5),(p_5, p_6)).

4.5.2 Comparison

We analyze the response surface of the learned score function as in Section 3.4.1. The response surface of the trained PCT-SVM-RAM is plotted in Figure 4.3(a),

where the same test image (cf. Figure 3.14(b)) is used for generating the response surface. The x-axis and y-axis correspond to the perturbation indexes for the 4th and 5th shape bases, respectively. The z-axis corresponds to the ranking score. The response surface is smooth and concave, which is favorable for a local optimizer to find the global maximum. Note that the iso-contours of the response surface are smoother than those in Figure 4.3(a), which indicates that the ranking score function is smoother than the classification score function, due to the ordering constraints on the negative samples. Further response maps are plotted in Figure 4.3(b) in contrast to Figure 3.14(c). They are generated on five different test images by perturbing parameters along two adjacent shape bases.

We compare the alignment performance of the ranking appearance model with the classification appearance model. The results are plotted in Figure 4.4 in AFC rates at increasing perturbation levels. The methods PCT-BAM-W50 and PCT-BAM-W100 correspond to the PCT-based classification appearance models with 50 and 100 weak classifiers, respectively. Linear SVM is used for training weak classifiers. The methods PCT-RAM-W50 and PCT-RAM-W100 denote the ranking appearance models with 50 and 100 weak ranking functions, respectively. MSPCT-RAM-W100 stands for the method with multi-scale mask images. The AFC curves show that PCT-RAM improves the robustness of face alignment compared to the PCT-BAM. When we observe the AFC rates at the highest noise level, PCT-RAM-W100 outperforms PCT-BAM-W100 by about $6.9\% - 13.7\%$ on different data sets. The most noticeable performance gain is achieved on Set 3. Improving alignment on Set 4 is difficult, probably due to the limitation of the shape model learned on Set 1. Interestingly, the performance gains achieved by selecting increasing numbers of PCT features in PCT-RAM are significant. For example, on Set 3, the AFC rate for PCT-RAM-W100 at 1.6σ noise level outperforms PCT-RAM-W50 by 15.3%. This indicates that the PCT-RAM selects more distinctive features than the PCT-BAM and is less likely to be overtrained. Thus, the PCT-RAM is able to generalize better on unseen data. The multi-scale appearance model, MSPCT-RAM-W100, demonstrates its effectiveness on Set 1 and Set 2. However, the performance gains on Set 3 and 4 are minor.

4.5.3 Effects of Reference Shape Size

We study the influence of mask size in the ranking appearance model learning. In Section 3.4.2.3, we show that a proper mask size used in the classification appearance modeling is essential. In this study, we train PCT-RAM-W100 with different mask image widths, ranging from 20 to 60 pixels. The AFC curves are compared in Figure 4.5(a) and (b), at two different perturbation

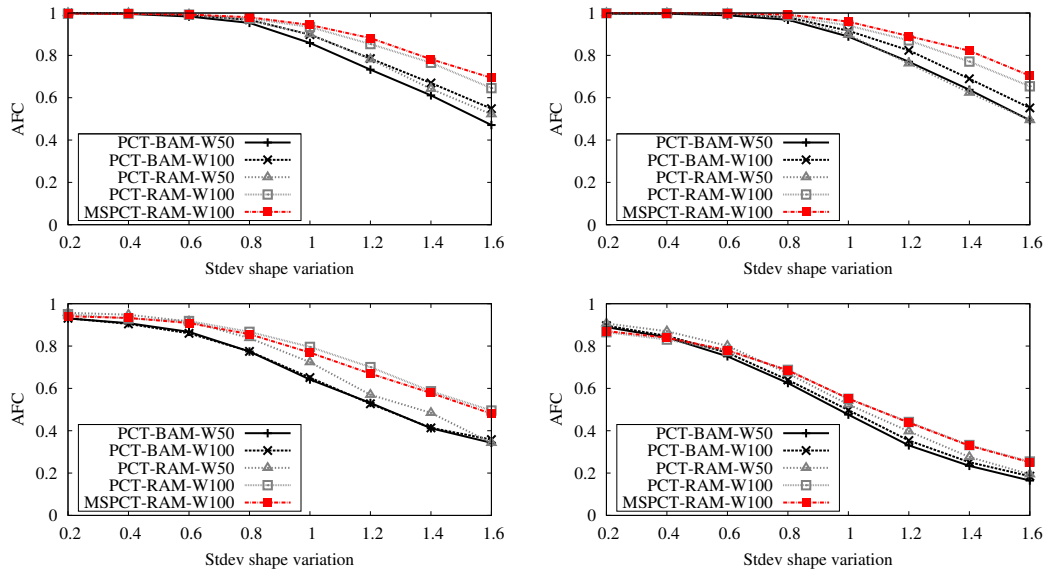


Figure 4.4: Alignment results of ranking-based appearance models on Set 1, Set 2 (first row) and Set 3, Set 4 (second row).

levels for alignment initialization. The results show that increasing mask size in the PCT-RAM training also improves the alignment performance over all data sets. Especially, on Set 1 and Set 2, the AFC rates keep increasing until the mask image is enlarged to 45 pixels width. However, on Set 3, the optimal size for the mask image is 35 pixels width at 1.6σ perturbation level. The performance gains on Set 3 and Set 4 with enlarged mask size are less notable than on Set 1 and Set 2.

4.5.4 Effects of Training Pair Sampling

Instead of generating training pairs using the adjacent pairs sampled in each perturbation direction, we also investigate random permutation of ordinal pairs. In the experiment, we train PCT-RAM-W100 with R random ordinal pairs per direction, where $R = \{1, 5, 6, 10, 15, 20, 21\}$. We repeat the training process 20 times to avoid randomness. The alignment AFC results at 1.6σ noise level are shown in Figure 4.6, in which the average AFC rates and their variances are plotted. The fitting performance of the models trained with only one ordinal pair per direction does not degenerate much. In fact, the mean AFC rates vary slightly with an increasing number of ordinal training pairs used. However, the variances of the AFC rates decrease a lot when R increases.

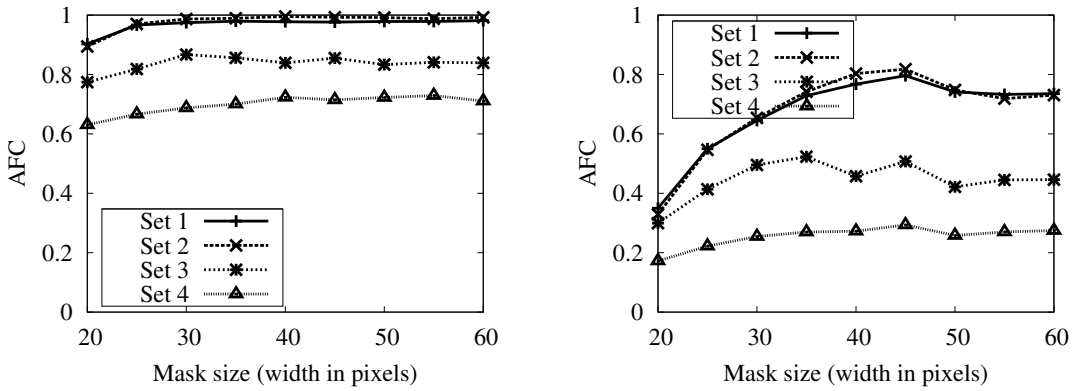


Figure 4.5: Effects of different size of masks (width in pixels) in PCT-BAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

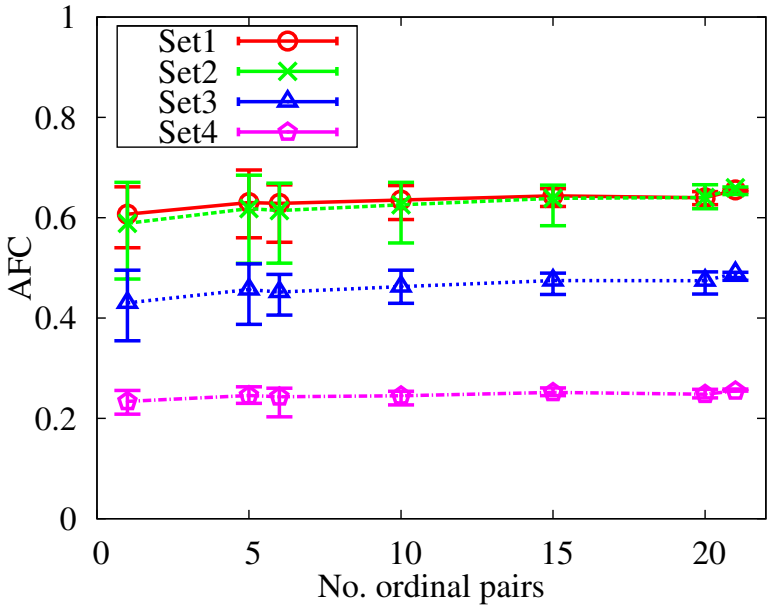


Figure 4.6: AFC results on different data sets with increasing numbers of randomly sampled ordinal pairs.

4.5.5 Effects of Number of Perturbation Directions

We finally discuss the effects of the number of perturbation directions for model learning, *i.e.*, the parameter U in Section 4.3.4. We vary this parameter from 1

to 30 and train PCT-RAM-W100 with adjacent ordinal pairs sampled in each perturbation direction. The alignment results at two noise levels are plotted in Figure 4.7. Surprisingly, we observe that the alignment convergence rates do not always increase, when the parameter U increases. Sometimes, the alignment performance even degrades on Set 2 and Set 3. Based on this observation, we argue that the number of perturbation directions is not a critical parameter in training ranking appearance models, at least when trained on Set 1.

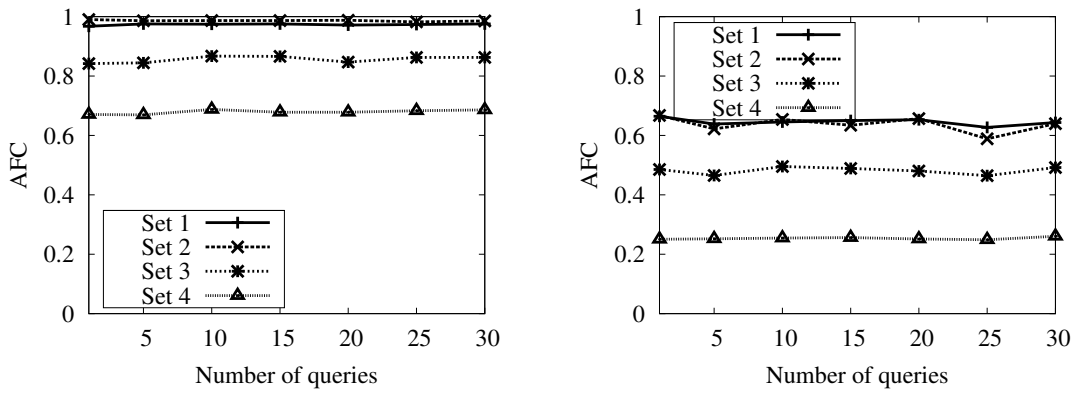


Figure 4.7: Effects of number of queries in PCT-RAM. (a) AFC at perturbation level 0.8σ; (b) AFC at perturbation level 1.6σ.

4.6 Conclusions

We investigate deformable appearance models for face alignment based on learning a ranking function. A pairwise ordinal classification-based method is adopted for learning the ranking model. The PCT feature and RankSVM are used to build weak ranking functions and a strong ranking function is learned via boosting regression stumps. We compare the alignment performance of the proposed ranking appearance model with the classification appearance model. Experiments show that the alignment robustness and generalization ability is significantly improved due to the smoothed cost function.

5 Regression Appearance Models

This chapter presents a regression based appearance model. We explain the motivation of this method in Section 5.1. The face appearance model based on an ensemble of regression trees is described in Section 5.2. We describe the simplex-based optimization method for model fitting in Section 5.4. The experimental setup and results are discussed in Section 5.5, and we conclude this chapter in Section 5.6.

5.1 Introduction

Pointwise ranking learning is another effective approach for solving the learning to rank problem, as is stated in Section 4.3. Popular methods for this type of ranking learning can be pointwise classification [LBW07b] or regression [ZCSZ07]. The pointwise classification method actually corresponds to a multi-class classification problem, in which the relationship of partial ordering is ignored. Although there is no metric defined in the ranking output space, the partial ordering relationship is preserved in the regression method in a more constrained way. Recent works in information retrieval demonstrate that the pointwise ordinal regression methods achieve appealing performance in solving the learning to rank problem.

In this chapter, we focus on learning a ranking appearance model based on a pointwise regression method. Due to the fact that the model training is based on minimizing regression loss, we categorize this approach as the regression-based appearance model (REAM). In particular, we propose to use the gradient boosting regression trees (GBRT) [Fri00] for learning a REAM. As the GBRT is well-known for function approximation, we use it for approximating a desired cost function for face alignment. In addition, it has been shown that the GBRT achieves top results in the domain of web-search ranking [MCW11]. To overcome the drawbacks of the GBRT, *e.g.* prone to overfitting and slow convergence rate, we train Random Forests (RF) and use the outputs as the initial estimation for the GBRT learning. The PCT features and the MCT features are used

for appearance representation. A simplex-based direct search method is applied for optimizing the learned alignment cost function. Experimental results show that the regression trees-based REAM achieves superior results than the pairwise ordinal classification model. The initialization step for the GBRT learning results in a very robust face alignment, which improves performance by about $23.4\% - 26.1\%$ on different data sets compared to the model based on pairwise ordinal classification.

5.2 Face Model

The presented regression-based appearance model includes a statistical shape model based on the 2D PDM (cf. Section 3.2.1) and an appearance model, which is constructed with the application of a regression model.

5.2.1 Feature Representation for the Appearance Model

The regression-based appearance model is again defined on the masked shape-free images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$, which represents the texture information inside the convex hull of a shape, controlled by the parameter vector $\mathbf{p}$. The nonlinear warping function $\mathbf{W}(\mathbf{x};\mathbf{p})$ defines the texture mapping from a shape instance to a reference shape.

We extract local structural features from the masked shape-free images and use them as our appearance model representation. In this work, the Modified Census Transform (MCT) and the Pseudo Census Transform (PCT) are applied. The MCT is originally proposed in [FE04] for developing an efficient and robust face detection algorithm. It is a non-parametric transform inspired by the Census Transform, which is first introduced by Zabih and Woodfill [ZW94] for texture analysis. The transform is defined as a set of 3×3 kernels which captures the local spatial structure of an image. It compares the pixel intensities between all the pixels of the 3×3 neighborhood and the average intensity in the neighborhood. More formally, we define $\bar{I}(\mathbf{x})$ as the average of the pixel intensities in a 3×3 local spatial neighborhood $N(\mathbf{x})$ of the pixel $\mathbf{x}$. The MCT generates an ordered bit string indicating which pixels in $N(\mathbf{x})$ have an intensity higher than $\bar{I}(\mathbf{x})$. Let $\zeta(\bar{I}(\mathbf{x}), I(\mathbf{y})) = 1$ if $\bar{I}(\mathbf{x}) < I(\mathbf{y})$ be the comparison function and $\otimes$ be the concatenation operator, then the transform is defined as: $\Gamma(\mathbf{x}) = \otimes_{\mathbf{y}\in N}\zeta(\bar{I}(\mathbf{x}), I(\mathbf{y}))$. An example MCT feature is illustrated in Figure 5.1(a), in which the resulting MCT has a binary value of 101111000 (=376 in

decimal). Figure 5.1(c) shows a sample output of the MCT applied on a shape-free image displayed in Figure 5.1(b). It has been proven that the transform is fast and robust to illumination changes. The PCT feature is an unbinarized version to the MCT and is introduced in [GEFS11] for deriving analytical alignment algorithm. The detailed description of the PCT feature can be found in Section 3.2.2.

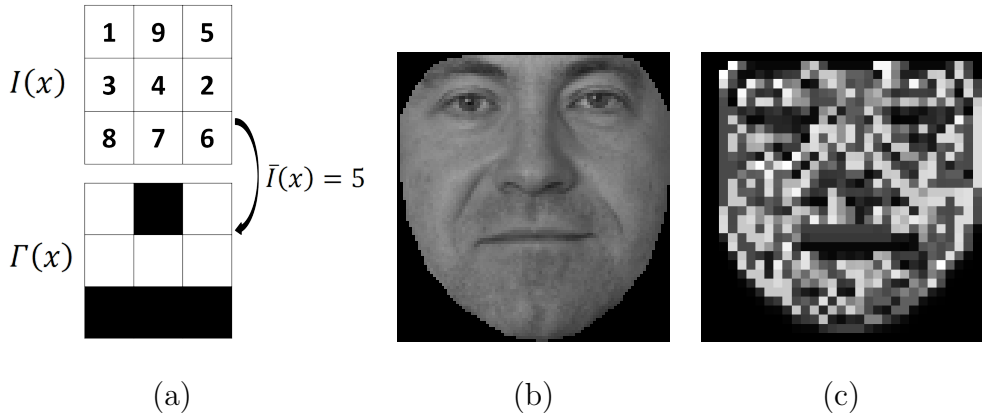


Figure 5.1: (a)MCT for extracting local structure feature; (b) A shape-free face image; (c) MCT output of a shape-free image.

5.3 Learning Regression Appearance Models

The pointwise ranking function learning is a popular trend and achieves remarkable results in the information retrieval domain [MCW11, CC11]. The approaches based on the gradient boosting regression trees enjoy their impressive success for learning ranking function with pointwise training [LBW07a, MCW11], despite its simplicity. This inspires us to employ the GBRT for learning our regression-based appearance model.

5.3.1 Gradient Boosted Regression Trees

The gradient boosted regression trees (GBRT) [Fri00] is a machine learning technique for function approximation, which is based on tree averaging. It iteratively adds shallow trees with biased estimation. Each iteration focuses on the data that are responsible for the current regression residue.

We denote $T(\mathbf{x}_i)$ as the current prediction of sample $\mathbf{x}_i$, and y_i as the corresponding ground truth response. The square loss: $L = \frac{1}{2}\sum_{i=1}^{N}(T(\mathbf{x}_i) - y_i)^2$ is adopted as the loss function as it is widely used in solving regression problems. The GBRT applies the gradient descent method to minimize the loss function in the data space $\mathbf{x}_1, \ldots, \mathbf{x}_N$. During each iteration, the current prediction $T(\mathbf{x}_i)$ is updated with a gradient descent step:

$$T(\mathbf{x}_i) \leftarrow T(\mathbf{x}_i) - \alpha \frac{\partial L}{\partial T(\mathbf{x}_i)}, \tag{5.1}$$

where $\alpha > 0$ denotes the learning rate. Thus, a new tree $h_t(\cdot)$ is chosen with its responses most highly correlated with the negative gradient $-\frac{\partial L}{\partial T(\mathbf{x}_i)}$ over the data distribution:

$$h_t \approx \arg\min_{h \in \mathcal{T}_d} \sum_{i=1}^{N} (h_t(\mathbf{x}_i) - r_i)^2, \text{where} : r_i = \frac{\partial L}{\partial T(\mathbf{x}_i)}. \tag{5.2}$$

As L is the squared loss, the gradient for a sample $\mathbf{x}_i$ becomes the residual from the previous iteration, *i.e.* $r_i = y_i - T(\mathbf{x}_i)$. The standard CART (Classification and Regression Trees) [Bre84] is applied to find a solution to Equation 5.2. $\mathcal{T}_d$ denotes the hypothesis space of regression trees with a depth of d.

The GBRT has a weakness, which lies in the inherent trade-off between the step-size and early stopping. To obtain the true global minimum, the step-size needs to be very small and the number of iterations becomes very large. This results in a large number of regression trees, which essentially decreases the efficiency of the model fitting. To tackle this problem, we try to initialize the GBRT learning with a reasonable start point, which is close enough to the global minimum. We borrow the idea in [MCW11], in which the Random Forests (RF) [Bre01] method is applied for initialization. Basically, Random Forests apply bagging and random feature selection to CART. The bagging (**B**ootstrap **agg**regat**ing**) [Bre96] technique combines models that are trained on randomly generated training sets. It improves the accuracy and robustness of predictions by reducing estimation variance and avoiding overfitting. The Random Forest is considered to be a good choice as it is insensitive to parameter choices and offers low bias estimation as each of the trees are fully grown. One difference between the RF and the GBRT is that, in the RF, only K uniformly chosen features are evaluated to find the best point for each split. Furthermore, unlike the sequential tree construction in the GBRT, the construction of a single tree in the RF is independent of earlier trees; thus the algorithm is easily parallelizable. Only two parameters need to be tuned. M_{RF} specifies the number of trees in the forest and K determines the number of features that each node considers for finding the best split. As suggested in the original paper, we set $K = \sqrt{f}$, where f is the number of features.

Algorithm 4: Random Forests initialized Gradient Boosted Regression Trees

Data: $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, Parameters: $\alpha, M_B, d, K_{RF}, M_{RF}$

Result: The initialized Gradient Boosted Regression Trees T

$F \leftarrow RandomForest(D, K_{RF}, M_{RF})$

Initialization: $r_i = y_i - F(\mathbf{x}_i), i = 1, \ldots, N$

for $t = 1$ **to** M_B **do**

    Find the h_t with the CART according to Equation 5.2;

    Update residue $r_i \leftarrow r_i - \alpha h_t(\mathbf{x}_i)$, $i = 1, \ldots, N$;

$T(\cdot) = F(\cdot) + \alpha \sum_{t=1}^{M_B} h_t(\cdot)$

return $T(\cdot)$

5.3.2 Initialized GBRT-based Regression Model

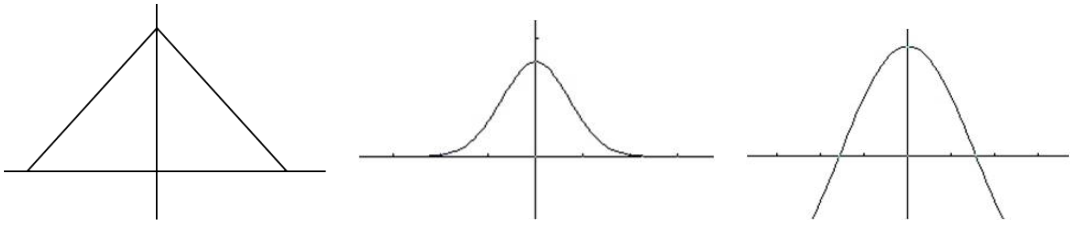
The original GBRT is initialized with the average of the ground truth responses, *i.e.* $T_0(\mathbf{x}_i) = \bar{y}$, where $\bar{y} = \frac{1}{n}\sum_{i=1}^{N} y_i$. Consequently, the initial residual is $r_i = y_i - \bar{y}$. To initialize the GBRT with a better estimation, which is closer to the global minimum, the responses of the RF are used as the initial point for the GBRT. We denote this initialized GBRT as iGBRT. Algorithm 4 details the steps in iGBRT. The output of the final boosted regression model is actually the response of the RF combined with the boosted regression trees.

5.3.3 Training Data for Learning

We apply iGBRT for learning a discriminative score function for face alignment. Basically, an ideal score function should return higher values, if the shape parameter is closer to ground truth than the others. We adopt the data perturbation approach introduced in Section 4.3.4 to generate samples from a training data set containing D facial images with annotated landmarks. For each of the training images, we randomly perturb the ground truth parameter $\mathbf{p}_i$ in U different directions $\{\Delta\mathbf{p}_{iu}\}_{u=1,\ldots,U}$. In each direction we evenly sample V shape parameters $\{\mathbf{p}_i + v \times \Delta\mathbf{p}_{iu}\}_{v=0,\ldots,V-1}$. In total, $N = D \times U \times V$ samples are generated. Instead of assigning ordinal class labels to the ordered pairs as in Section 4.3.4, we assign ranking labels to each of the generated data point. That

means we assign $y_i \in \{1, \ldots, V\}$, where the data generated using the ground truth is assigned with the highest value, *i.e.* V. The other samples in the same direction are assigned with $V - \nu$, where $\nu = 1, \ldots, V - 1$.

The assignment with ranking values corresponds to a triangle target function as displayed in Figure 5.2(a). In addition, we also investigate other nonlinear functions as our regression target function. The Gaussian function (cf. Figure 5.2(b)) and cosine function (cf. Figure 5.2(c)) are used, expecting to learn a smoother regression function. For the Gaussian function, we evenly assign function values within three standard deviations. For the cosine function, the function values are evenly assigned within the range $[-\pi, \pi]$.



(a) (b) (c)

Figure 5.2: Functions for regression target assignment: (a) Triangle function; (b) Gaussian function; (c) Cosine function.

5.4 Face Alignment with Regression Appearance Models

The regression model with combined regression trees is considered as the cost function for face model alignment. Our goal is to find the optimal shape parameters, which maximize the regression function given a testing image. The optimization is a constrained optimization problem, in which the shape parameters searching is limited with a shape prior. We convert the alignment objective function to an unconstrained optimization as follows:

$$O(\mathbf{p}) = -T(\mathbf{p}) + \beta \sum_{i=0}^{n} \frac{p_i^2}{\lambda_i}, \tag{5.3}$$

where β is the parameter, which we estimate from the training data. λ_i is the eigenvalue corresponding to the shape parameter p_i. The first term in Equation 5.3 corresponds to the regression function to be maximized. The second

term corresponds to the negative log-likelihood of the shape parameters, which needs to be minimized.

We add an additional term to the alignment objective function based on the idea of the Active contours [KWT88]. Here, an edge energy term is added to the objective function (cf. Equation 5.3), meaning that a good alignment should also have large edge responses along the contours of a face or other sub-regions such as the nose and mouth. Example edge responses on different images are shown in Figure 5.3. The edge responses are obtained by applying the Canny edge detector on the Gaussian smoothed images. A post processing step with morphological dilation is applied to connect the detected contours. With the additional constraint on the edge responses, the alignment objective function is then defined as:

$$O(\mathbf{p}) = -T(\mathbf{p}) + \beta \sum_{i=0}^{n} \frac{p_i^2}{\lambda_i} - \gamma \sum_{j=0}^{v} E(\mathbf{x}_j). \tag{5.4}$$

Where γ denotes the weight for the edge term, which is also estimated from the training data. $E(\mathbf{x}_j)$ denotes the edge response value at location $\mathbf{x}_j$ for the j-th vertex in a model shape.

Figure 5.3: Edge responses (second row) superimposed on the original images. The first row shows the corresponding original images.

As it is difficult to derive the analytical gradient for the learned objective function using regression trees, we apply the Nelder-Mead simplex method [NM65] to minimize Equation 5.3. The Nelder-Mead algorithm is designed to solve the classical unconstrained optimization problem of minimizing a given nonlinear

function $f : \mathbb{R}^n \to \mathbb{R}$. The method uses only function values at some points in $\mathbb{R}^n$ and does not try to form an approximate gradient at any of these points. Hence it belongs to the general class of direct search methods [KLT03].

The Nelder-Mead method is simplex-based. A simplex S in $\mathbb{R}^n$ is defined as the convex hull of $n+1$ vertices $\mathbf{x}_0, \ldots, \mathbf{x}_n \in \mathbb{R}^n$. For example, a simplex in $\mathbb{R}^2$ is a triangle and simplex in $\mathbb{R}^3$ is a tetrahedron.

The simplex-based direct search method begins with a set of $n+1$ points $\mathbf{x}_0, \ldots, \mathbf{x}_n \in \mathbb{R}^n$ that are considered as the vertices of a working simplex, S, and the corresponding set of function values at the vertices $f_j := f(\mathbf{x}_j)$, for $j = 0, \ldots, n$. The initialized simplex is applied with several transformations including reflection, expansion, contraction, and multi-contraction, which are illustrated in Figure 5.4. The transformations are repeated until the termination criteria are met.

5.5 Experiments

5.5.1 Data and Setup

For evaluating face alignment using the proposed appearance model, we use the data set presented in Section 3.4.1. Set 1 is used for training the shape and appearance models and testing is conducted on all the four data sets for analyzing generalization ability at different levels.

Using Set 1, we train a shape model with 15 components preserving 95% of shape variations. The size of shape-free images is 30×30 pixels. For each image, we select $U = 10$ random directions and in each direction $V = 6$ positions are evenly sampled. Including the position at the ground truth, in total 7 ordered data samples can be generated. The overall training set includes $N = 24400$ $(400 \times 10 \times 6 + 400)$ ordered data samples.

In testing, we randomly perturb ground truth landmarks at different noise levels for initializing each alignment. We repeat the random perturbation for each noise level multiple times on each test image in order to perform a statistical evaluation of the result. A fitting is considered as converged if the Root Mean Square Error (RMSE) between the aligned landmarks and the ground truth is less than one pixel. The Average Frequency of Convergence (AFC) is used as the evaluation metric, which assesses the robustness of the alignment. The metric AFC is calculated as the number of converged trials divided by the total number of trials.

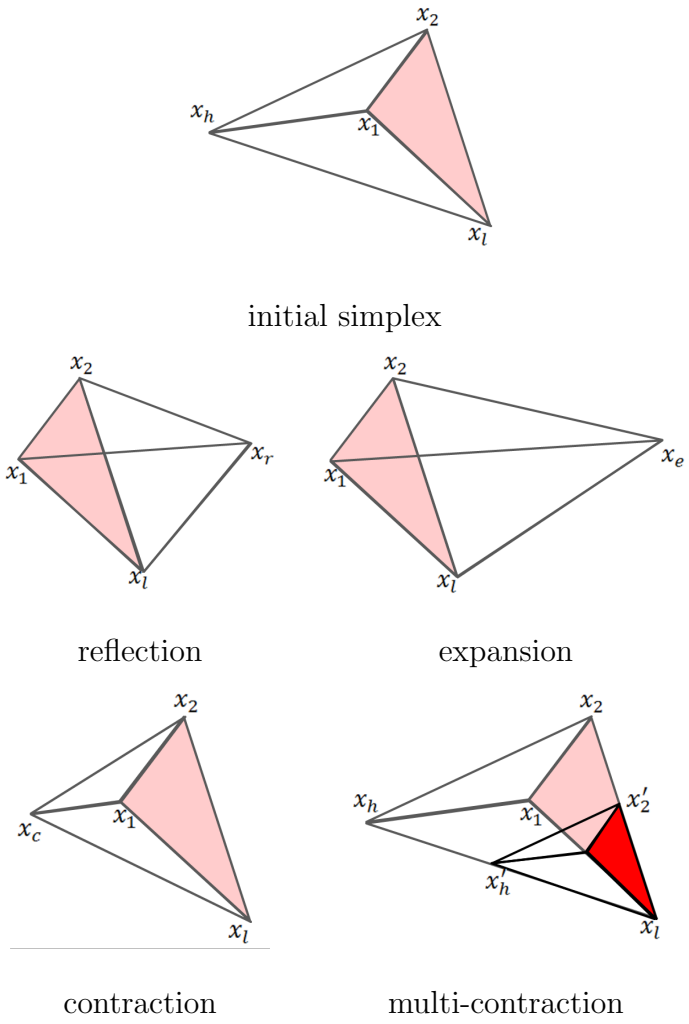
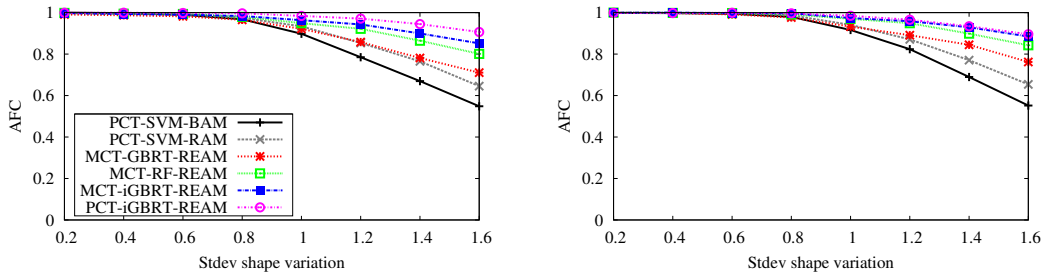


Figure 5.4: Simplex transformations in the Nelder-Mead algorithm.

5.5.2 Comparison

We first evaluate the REAM with regression trees and the MCT is used as feature. To assess the effectiveness of the RF initialized GBRT, we first compare the ranking performance for different models based on regression trees, *i.e.* RF, GBRT, and iGBRT. The training data for building the regression trees are prepared according to Section 5.3.3. We again use Set 1 as training set to extract the training samples. We set $M_{RF} = 100$ and $M_B = 100$. For GBRT and iGBRT, we set the tree-depth $d = 4$ and the learning rate $\alpha = 0.05$. The testing data is extracted in the same scheme on all four data sets. The ranking results are plotted in Figure 5.6(a), in which the percentage of the swapped pairs is considered as the ranking error rate. From the plot, we can observe that, for all data sets, the ranking error rates of RF are always lower than GBRT. The



(a) Set 1

(b) Set 2

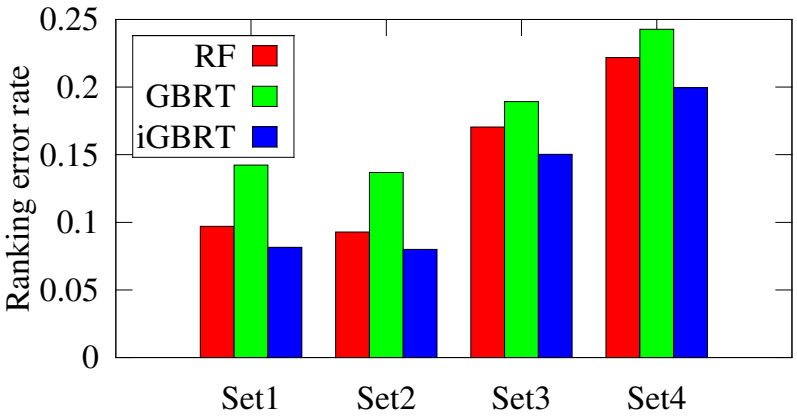
(c) Set 3

(d) Set 4

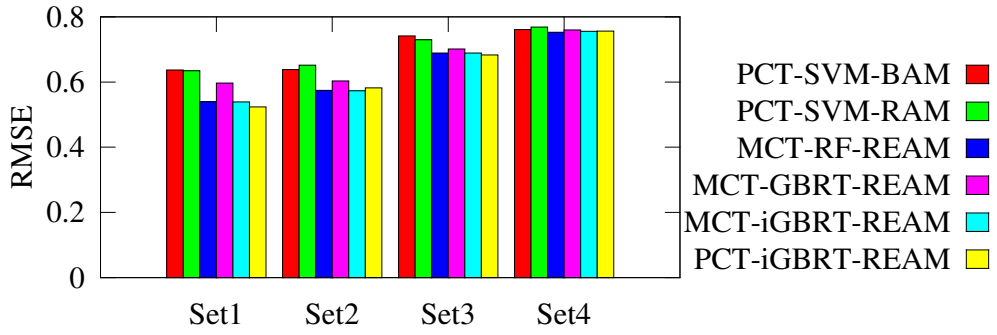
Figure 5.5: Regression trees-based alignment results on Set 1, Set 2 (first row) and Set 3, Set 4 (second row).

bagging technique and low bias regression make RF resist to overfitting. The iGBRT outperforms RF and GBRT consistently on all data sets.

The superior performance of iGBRT is proven again in the face alignment experiments. The REAM based on Random Forests (MCT-RF-REAM) shows large improvement compared to the pairwise RAM (PCT-SVM-RAM). The most significant improvement is observed on Set 2, where MCT-RF-REAM obtains around 22.3% performance gain over PCT-SVM-RAM at the highest perturbation level. It is found that MCT-RF-REAM already boosts the fitting performance to a large extent. The introduction of iGBRT-based REAM increases the robustness further. In order to show that iGBRT also works for the PCT feature based representation, we train regression trees on top of PCT features after RankSVM scores are obtained. The alignment results (PCT-iGBRT-REAM) are plotted in Figure 5.5, which show further improvement over MCT-iGBRT-REAM. This proves that the unbinarized census transform provides additional discriminative information for training the regression trees-based REAM. Finally, in Figure 5.6(b), we show the face alignment accuracy (pixels in average) of the converged trials with different models. The iGBRT-based REAM almost always outperforms the other models.



(a)



(b)

Figure 5.6: (a) Ranking error rates of different regression trees-based REAM; (b) Average face alignment accuracy (in pixels) of the converged trials.

Models	MCT-GBRT	MCT-RF	MCT-iGBRT	PCT-iGBRT
Fitting cost	15.67ms	29.88ms	34.72ms	256.95ms
AFC @ 1.6σ	57.90%	61.9%	65.7%	72.89%

Table 5.1: Computational cost (ms in average) and fitting performance on Set 3

	Triangle	Gaussian	Cosine	Triangle	Gaussian	Cosine
Perturbation	0.8σ			1.6σ		
Set 1	**0.983**	0.983	0.979	**0.851**	0.808	0.826
Set 2	**0.996**	0.996	0.987	**0.885**	0.852	0.855
Set 3	**0.915**	0.908	0.913	**0.657**	0.609	0.617
Set 4	**0.732**	0.729	0.710	**0.406**	0.390	0.381

Table 5.2: Comparison of alignment performance in AFC rates applied with different target functions for learning regression models. The AFC rates are obtained at 0.8σ and 1.6σ perturbation levels for initialization.

We analyze the computational cost for fitting different models in Table 5.1. We run the fitting experiments on a machine with Intel Xeon CPU (2.93GHZ) in an unparallelized C++ implementation. The second row in Table 5.1 lists the average fitting time (in millisecond) on the images in Set 3. The third row shows their AFC rates at 1.6σ noise level. We observe that although PCT-iGBRT-REAM achieves better results than MCT-iGBRT-REAM, the computational cost for each fitting is much higher due to the projection step using RankSVM.

5.5.3 Effects of Regression Target Function

In addition to assigning rank values as regression targets, we also employ nonlinear functions as our regression target function. The Gaussian function and cosine function are used. We train regression appearance models with MCT-iGBRT using the evenly sampled function values as regression targets. Table 5.2 lists the alignment AFC rates achieved on all four testing data sets at two initial perturbation levels. We notice that using rank values (triangle function) as regression target always outperforms the other two nonlinear functions. However, the differences are rather small. The only notable performance gain is achieved on Set 3 at 1.6σ perturbation level, where the linear target function performs 4.8% better than the Gaussian function and 4.0% better than the cosine function.

	iGBRT	iGBRT+Edge	iGBRT	iGBRT+Edge
Perturbation	0.8σ		1.6σ	
Set 1	0.983	**0.987**	0.851	**0.858**
Set 2	**0.996**	0.992	0.885	**0.886**
Set 3	0.915	**0.922**	0.657	**0.674**
Set 4	0.732	**0.734**	0.406	**0.42**

Table 5.3: Comparison of alignment performance in AFC rates applied with and without edge constraint. The AFC rates are obtained at 0.8σ and 1.6σ perturbation levels for initialization.

5.5.4 Effects of Edge Constraint

We assess the effectiveness of the additional edge constraint in the alignment cost function. Table 5.3 compares the alignment results obtained with edge constraint (iGBRT+Edge) to the results obtained without edge constraint (iGBRT). We notice that at 1.6σ noise level, the edge constraint always improves the alignment performance. The improvements are more notable on Set 3 and Set 4, which indicates that the edge information can be helpful for avoiding some local minima. Note that the cluttered background in the images in Set 4 may also result in strong edge responses (cf. Figure 5.3). These edge responses may confuse the alignment cost function and eventually have a negative impact on the alignment performance. However, the improved alignment convergence rate on Set 4 proves that the negative impact is surpassed by the positive impact.

5.6 Conclusions

We present a regression appearance model based on the gradient boosted regression trees. The Random Forests technique is used to initialize the GBRT training iterations. The initialization provides the GBRT with an initial estimation with low bias and requires fewer iterations to converge to the global optimum. We conduct experiments on four different data sets. The results show that the regression trees-based REAM significantly improves the robustness and accuracy in terms of face alignment. Our best proposed model (PCT-iGBRT-REAM) boosts the alignment performance about $23.4\% - 26.1\%$ on different

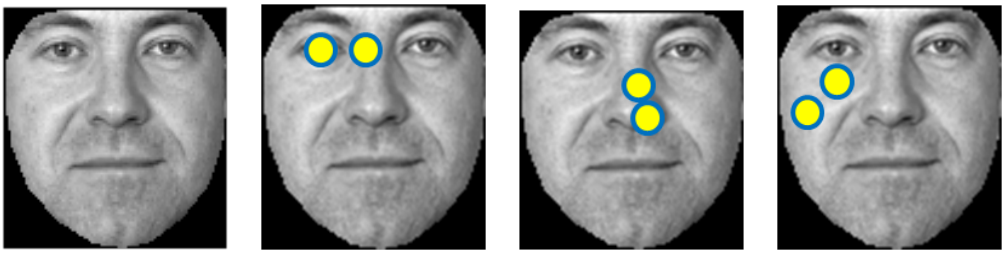
data sets compared to the model based on pairwise ordinal classification (PCT-SVM-RAM).

6 Regression Appearance Model based on Random Pixel Intensity Differences

This chapter presents a novel feature representation for building regression based appearance models. The feature employed in this study is simple but effective. Moreover, it provides semantic meanings for assessing the correctness of alignments. In Section 6.1, we give a short introduction and motivation of the proposed approach. We address the details of the feature representation in Section 6.2 and its learning procedure for training a regression appearance model in Section 6.3. The experimental results are discussed in Section 6.5. And finally, we draw some conclusions for this chapter in Section 6.6.

6.1 Introduction

The local structural feature-based appearance models are designed to be robust against illumination changes. However, this type of feature only compares pixel intensities in a local neighborhood. In this work, we propose a novel appearance model based on RAndom Pixel Intensity Differences (RAPID) features. The intensity differences are extracted on pixel locations within a certain distance range. The motivation behind this feature is that meaningful contextual difference features can be selected for learning a smooth alignment cost function. A quantization technique is applied on the difference features for mitigating the inherent image noise. The quantized pixel intensity differences are selected and randomly sampled for learning a robust alignment cost function. The cost function is learned in an ordinal regression manner using Random Forests. To ensure that more informative features are selected, we propose a correlation-based approach to filter out features, which are less correlated to the regression labels. We also utilize the label distribution information stored on the leaf nodes of each regression tree in the RF to discard highly uncertain estimates. We evaluate the proposed appearance model on four different data sets. Experiments show that the quantization technique makes alignment more robust against image noise. The uncertainty-based estimation filtering



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

Figure 6.1: (a) A shape-free face image; Pixel intensity differences extracted around texture-rich face regions such as between eye center and eye corner (b); and between nostril and nose bridge (c); Pixel differences extracted around homogeneous regions (d) are less likely to be selected.

increases the average convergence rate slightly. We also show that our model boosts the alignment performance about $8.7\% - 17.2\%$ compared to the MCT feature and about $5.5\% - 10.32\%$ compared to the PCT feature. In addition, we compare the proposed alignment algorithms with two state-of-the-art discriminative face alignment models. Experiments demonstrate that the appearance model proposed in this chapter achieves superior alignment performance as well as generalization capabilities.

6.2 Face Model

The presented appearance model consists of a shape model and an appearance model. A statistical shape model is adopted as in Section 3.2.1. The appearance model is constructed independently in a sense of discriminative learning. The following subsections describe the feature representation of the appearance models, as well as the learning problem for face alignment.

6.2.1 Feature Representation for the Appearance Model

The proposed appearance model is again defined on the masked shape-free images $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$. The local structure features such as the MCT [FE04] and the PCT [GEFS11, GES12] compare pixel intensities in a local neighborhood, which results in an illumination invariant feature representation. However, relationships between pixels that are far apart might convey more discriminative information for representing the appearance model. Differences of pixel intensities provide a simple representation and effective computing, yet also yields impressive performance given sufficient training data as reported in [GL09, DWP10, CWWS12]. In this work, in addition to the pixel intensity differences, we also investigate the absolute intensity differences. The quantization of the differences is employed to make the representation more robust against the inherent image noise.

6.2.1.1 Pixel Intensity Difference

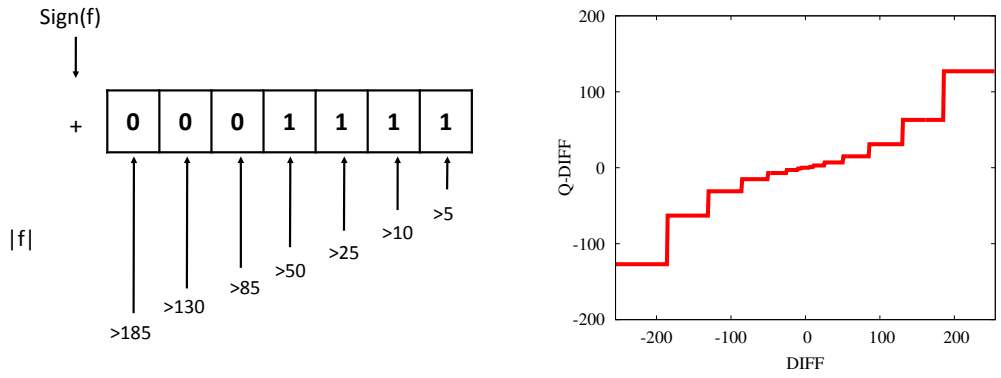
The intensity relationship reveals reasonable confidence of whether a face image is well aligned. Good features are usually located at texture rich areas, *i.e.* around the facial features. For example, a good pixel difference feature could be "the eye center is darker than the nose bridge" (cf. Figure 6.1(b)), or "the nostril is darker than the nose tip" (cf. Figure 6.1(b)). The features located in homogeneous regions suppose to be less informative (cf. Figure 6.1(c)). We denote each of the intensity difference feature as $f_{\mathbf{x}_1,\mathbf{x}_2}(\mathbf{I}) = \mathbf{I}(\mathbf{x}_1) - \mathbf{I}(\mathbf{x}_2)$, where $\mathbf{x}_1$ and $\mathbf{x}_2$ are the pixel locations defined in the image coordinate of a shape-free image $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$ (For simplicity we denote $\mathbf{I}$ as the shape-free image $\mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))$). We also consider absolute differences $|f_{\mathbf{x}_1,\mathbf{x}_2}(\mathbf{I})|$ by omitting the sign of differences.

6.2.1.2 Intensity Difference Quantization

We propose a quantization method to encode the difference values into bit strings. The quantization function is defined as

$$\Gamma(f) = \text{sign}[f] \bigotimes_{i=1..7} \zeta(b_i, |f|), \tag{6.1}$$

where $\mathbf{b} = [5, 10, 25, 50, 85, 130, 185]^\top$ is a quantization reference vector. The operator $\bigotimes$ denotes the operation for concatenating bits. Figure 6.2(a) displays



(a) (b)

Figure 6.2: (a) Coding of the pixel intensity difference with quantization method "Q-DIFF1" ; (b) The quantization function corresponds to "Q-DIFF1".

how a pixel difference feature $f = 65$ is quantized. The quantized feature results in $\Gamma(f) = 15$. Figure 6.2(b) plots the quantization function Γ, with the axis "DIFF" as input and axis "Q-DIFF" as output. We notice that the quantization function approximates the discretized logit function. The idea behind this coding is to smooth more noise with increasing absolute intensity difference. We also tried another quantization method (Q-DIFF2), in which an additional high order bit is added: $\Gamma(f) = \bigotimes_{i=1..7} \zeta(b_i, |f|) \bigotimes \mathbf{1}_{f>0}$. Here $\mathbf{1}_{f>0}$ is the indicator function defined as

$$\mathbf{1}_{f>0} = \begin{cases} 1 & f > 0 \\ 0 & f <= 0. \end{cases} \tag{6.2}$$

6.3 Learning Appearance Models

We apply ordinal regression to learn a ranking-based appearance model. It has been demonstrated in Chapter 5 that learning ranking models with ordinal regression yields better performance than a pairwise ordinal classification model. We employ Random Forests for ordinal regression due to its robustness and simplicity.

There are N^2 possible pixel difference features, where N is the number of pixels inside the shape-free image mask. The number of possible features is huge and moreover, a large percentage of the features are not very informative for

learning a good model, including those features may increase the variance of the final estimation. For this reason, we only consider features extracted within a certain maximum distance $d_{\max}$, under the assumption that meaningful features are located at the positions around facial features but not regions between the forehead and chin, where the pixel distances are large. In addition, we propose a correlation based feature selection to remove less informative features. We consider those features, which are highly correlated to the labels assigned to training samples, as good candidates for training.

6.3.1 Preparation of Training Data for Learning

The training samples are generated as suggested in Section 5.3.3. We generate data samples from a training data set containing D facial images with annotated landmarks. For each of the training images, we randomly perturb the ground truth $\mathbf{p}_i$ in U different directions $\{\Delta\mathbf{p}_{iu}\}_{u=1,\ldots,U}$. In each direction we evenly sample V shape parameters $\{\mathbf{p}_i + v \times \Delta\mathbf{p}_{iu}\}_{v=0,\ldots,V-1}$. For each direction, we can generate V ordered samples including the ground truth. The ideal cost function, which we want to learn, is supposed to return higher values, if the shape parameter is closer to ground truth than the others. We assign ranking labels to each of the data points. That means we assign $y_i \in \{1, \ldots, V\}$, where the samples generated using the ground truth are assigned with the highest value, *i.e.* V. The other samples in the same direction are assigned with $V - \nu$, where $\nu = 1, \ldots, V - 1$.

6.3.2 Feature Selection

We use a correlation-based method to select distinctive features on the generated samples using the assigned labels $\mathbf{y}$. In particular, for a single pixel intensity difference feature extracted on all samples $\mathbf{f}$, we calculate a correlation coefficient $Corr(\mathbf{f}, \mathbf{y})$ for measuring its relationship to the assigned labels. We sort the features according to the absolute correlation coefficients $|Corr(\mathbf{f}, \mathbf{y})|$ in a descending order. The informative features are selected by preserving a fixed number of features in the sorted list resulting a feature set $\mathcal{S}$. Alternatively, one can also select features with $|Corr(\mathbf{f}, \mathbf{y})|$ above a threshold τ_{corr}. The selected features are, however, redundant. To remove the redundancy, we randomly select a subset $\mathcal{R}$ from $\mathcal{S}$.

Following correlation coefficients are considered for the feature selection, namely, the Pearson correlation, Spearman rank correlation, and Kendall rank correlation.

6.3.2.1 Pearson Correlation

The Pearson correlation coefficient is the most common measure of correlation in statistics, which shows the linear relationship between two variables. The Pearson correlation between variable $\mathbf{x}$ and $\mathbf{y}$ is calculated as:

$$Corr(\mathbf{x}, \mathbf{y}) = \frac{Cov(\mathbf{x}, \mathbf{y})}{\sigma_{\mathbf{x}} \sigma_{\mathbf{y}}}, \tag{6.3}$$

where $Cov(\mathbf{x}, \mathbf{y})$ denotes the covariance between $\mathbf{x}$ and $\mathbf{y}$. $\sigma_{\mathbf{x}}$ represents the standard deviation of the elements in $\mathbf{x}$, while $\sigma_{\mathbf{y}}$ represents the standard deviation of the elements in $\mathbf{y}$. The results of Equation 6.3 are between -1 and 1. A result of -1 means that there is a perfect negative correlation between the two variables, while a result of 1 means that there is a perfect positive correlation between the two variables. A result of 0 means that there is no linear relationship between the two variables.

6.3.2.2 Spearman Rank Correlation

The Spearman rank correlation is a nonparametric measure of statistical dependence between two variables. It assesses how well the relationship between two variables can be described using a monotonic function. A Spearman correlation of 1 (or -1) results, when two variables being compared are monotonically related, even if their relationship is not linear. In contrast, this does not give a perfect Pearson correlation.

The Spearman correlation is calculated by applying the Pearson correlation to the rank values of the data rather than the actual data values. For two n dimensional random variables $\mathbf{x}$ and $\mathbf{y}$, the n raw values $x_i,\ y_i$ are converted to ranks $\hat{x}_i,\ \hat{y}_i$, and the correlation ρ is computed using the Pearson correlation between the ranked variables. Identical values are assigned a rank equal to the average of their positions in the ascending order of the values. In applications, where ties are known to be absent, a simpler procedure can be used to calculate ρ. The difference $d_i = \hat{x}_i - \hat{y}_i$ between the ranks of each observation on the two variables are calculated, and ρ is given by:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}. \tag{6.4}$$

6.3.2.3 Kendall Rank Correlation

The Kendall rank correlation is another measure of rank correlation, which is usually called the Kendall τ. The calculation of this correlation is based

on counting the number of concordant pairs and discordant pairs in the n-dimensional joint random variables $\mathbf{x}$ and $\mathbf{y}$. Any pair of observations (x_i, y_i) and (x_j, y_j) are said to be concordant if their ranks for both elements agree. Otherwise they are considered as a discordant pair.

The Kendall τ coefficient is defined as:

$$\tau = \frac{\#\text{concordant pairs} - \#\text{discordant pairs}}{\frac{1}{2}n(n-1)}. \tag{6.5}$$

The denominator is the total number of pair combinations, so the coefficient must be in the range $[-1, 1]$. If the agreement between the two rankings is perfect (*i.e.*, the two rankings are the same), the coefficient has the value 1. If the disagreement between the two rankings is perfect (*i.e.*, one ranking is the reverse of the other), the coefficient has the value -1. If $\mathbf{x}$ and $\mathbf{y}$ are independent, then we would expect the coefficient to be approximately zero.

6.3.3 Random Forests

Random Forests are proposed by Breiman [Bre01], which basically apply bagging and random feature selection to CART. It builds an ensemble of fully grown trees, $T_i(\mathbf{f})$, with a random subset of the training data. For each node split, only K random features are considered. The bagging technique reduces the estimation variance of the individual regression tree with slightly increased bias. The random feature selection makes the model robust to noise and outliers. On each leaf node of a regression tree, we store the distribution of the samples attached to this node with normal distribution $\mathcal{N}(\mu, \sigma^2)$, where μ is the mean and σ^2 is the variance of the labels assigned to the samples. The parameter setting in RF training is simple. As reported in the original paper [Bre01], the estimation performance saturates as the number of trees M_{RF} increases. Hence setting this parameter is not essential. The parameter K, which determines the number of features used for node splitting, is set to be $K = \sqrt{|\mathbf{f}|}$, as suggested in [Bre01]. After RF training, we obtain a discriminant appearance model with a fitting cost function defined as:

$$T(\mathbf{f}) = \frac{1}{M_{RF}} \sum\nolimits_{i=1}^{M_{RF}} T_i(\mathbf{f}), \tag{6.6}$$

where $T_i(\mathbf{f})$ returns the μ value stored on the leaf node it reaches and the final estimation is averaged over all trees. We also use the variance σ^2 to determine the uncertainty of estimation. If the uncertainty is high ($\sigma > \tau_\sigma$), the prediction from this tree will be discarded. This filtering results in M'_{RF} activated trees and the fitting cost function is defined by averaging the predicts over the M'_{RF} activated trees. We illustrate an overview of the training and testing procedure in Figure 6.3.

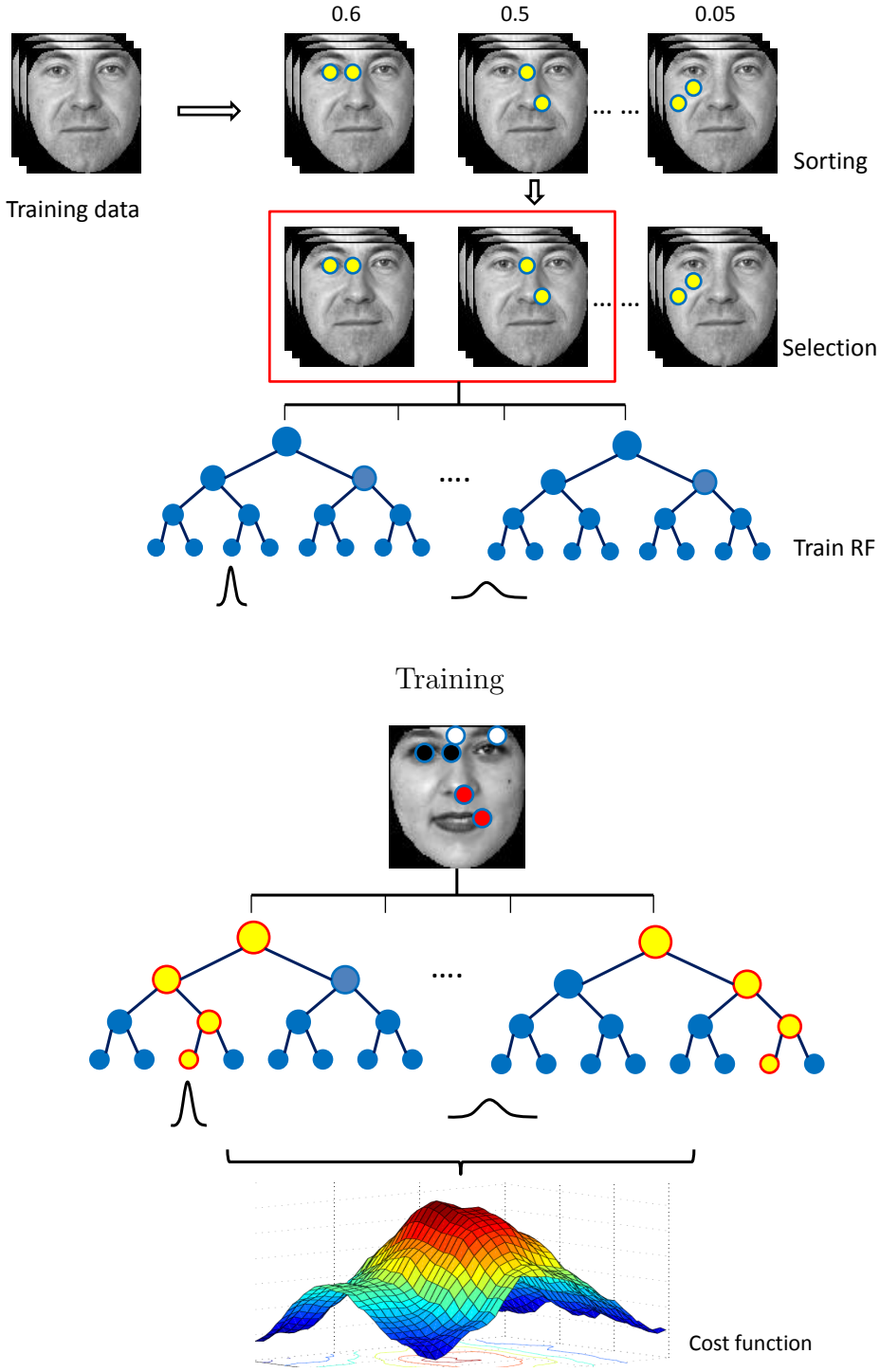


Figure 6.3: Overview of model training and testing.

6.4 Fitting the Appearance Model

Face alignment is equivalent to maximizing the cost function defined in Equation 6.6, subject to the additional constraint on shape prior. The shape prior ensures that the optimization is constrained within a pre-trained face shape space as defined in Section 3.2.1. As the shape parameters are modeled with a multivariate Gaussian with diagonal covariance matrix, we ensure the constraint by maximizing the likelihood of a shape parameter $\mathbf{p}$ defined by

$$\mathcal{L}(\Lambda|\mathbf{p}) \propto \prod_{i=1}^{n} \exp(\frac{-p_i^2}{\lambda_i}), \tag{6.7}$$

where λ_i is the eigenvalue corresponding to shape parameter p_i. Maximizing $\mathcal{L}(\Lambda|\mathbf{p})$ is equivalent to minimize the negative log-likelihood, and the constraint objective function can be then defined as follows:

$$O(\mathbf{p}) = -T(\mathbf{p}) + \beta \sum_{i=1}^{n} \frac{p_i^2}{\lambda_i}, \tag{6.8}$$

where β is the parameter that we estimated from the training data. As with in Section 5.4, the Nelder-Mead simplex method [NM65] is applied for minimizing the alignment objective function.

6.5 Experiments

6.5.1 Data Sets

For evaluating face alignment using the proposed appearance model, we use the data set presented in Section 3.4.1. Set 1 is used for training the shape and appearance models and testing is conducted on all the four data sets for analyzing the generalization ability at different levels.

6.5.2 Evaluation

Throughout the experiments in this chapter, we use Set 1 for training shape model and appearance models. We train a shape model with 15 components preserving 95% of shape variations. The size of shape-free images is 30×30 pixels. For each image, we select $U = 10$ random directions and in each direction

$V = 6$ positions are evenly sampled. Including the position at ground truth, in total 7 ordered data samples can be generated. The overall training set includes $N = 24400$ ($400 \times 10 \times 6 + 400$) ordered data samples. We generate 100 regression trees in the RF training. The maximum depth of each tree is 20.

In testing, we randomly perturb ground truth landmarks at different noise levels for initializing each alignment. We repeat the random perturbation for each noise level 5 times on each test image in order to perform a statistical evaluation of the result. We claim a fitting is converged if the Root Mean Square Error (RMSE) between the aligned landmarks and the ground truth is less than one pixel. The Average Frequency of Convergence (AFC) is used as the evaluation metric, which assesses the robustness of the alignment. The metric AFC is calculated as the number of converged trials divided by the total number of trials.

6.5.3 Results and Analysis

6.5.3.1 Model Parameters

Effect of Quantization The first experiment compares (absolute) intensity differences and their quantized versions. We compare the alignment metric AFC on the difficult data sets (Set 3 and Set 4) to see clear differences between the various feature representations. Figure 6.4 plots the average convergence rates at the highest perturbation level with 1.6σ. Results show that quantizing the difference features increases the robustness of face alignment on both data sets. Preserving the sign of the differences (Q-DIFF1) helps and setting the sign indicator on the high order bit (Q-DIFF2) is suboptimal. In this experiment, we fix the parameter $d_{max} = 16$ and $|\mathcal{S}| = 80000$. We randomly select $|\mathcal{R}| = 2000$ features in total.

Range of Pixel Distance We explore the range of distance between two pixels for extracting the features. We vary the parameter $d_{\max}$ from 2 to 31. The achieved results on Set 3 and Set 4 are plotted in Figure 6.5. We observe that selecting features in a small distance range already produces decent performance. When $d_{\max}$ increases, the average convergence rate increases due to more informative features with larger distances being selected. However, the performance decreases, when the distance range increases further. This indicates that informative features are mainly found in mid-range distances such as pixels between the eye corners and the eye centers or between the nostril and the nose tip, etc. The pixel difference in large distance ranges introduces noise as they are mainly extracted in homogeneous regions such as the forehead and chin.

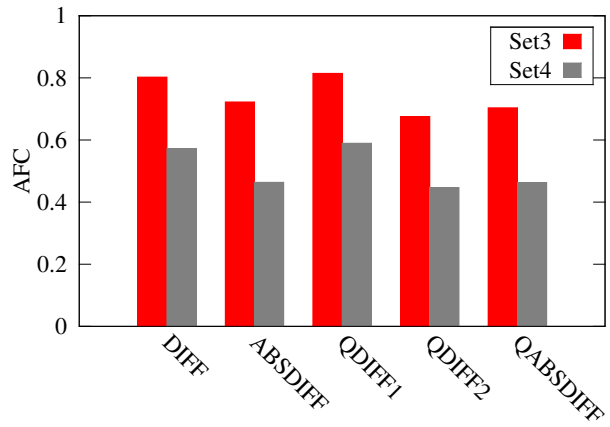


Figure 6.4: Alignment performance in AFC metric obtained with various difference features and quantization techniques. The results are reported on Set 3 and Set 4.

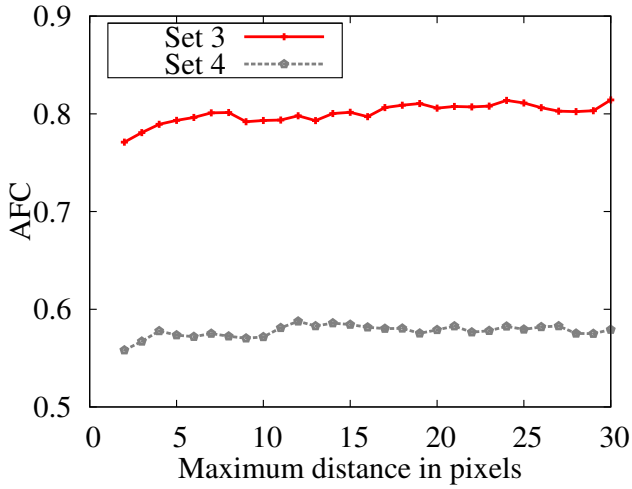


Figure 6.5: Effects of alignment performance by varying distance range d_{max} for selecting random pixel intensity differences.

Uncertainty Filtering The goal of this experiment is to check whether discarding individual trees in RF with large uncertainty can improve the robustness of the RF estimation. Figure 6.6 plots the average convergence rates on Set 3 and Set 4 with increasing τ_σ. When the threshold is low, the estimation focuses only on trees with high certainty. However, this will discard too many trees and increase the estimation bias. When τ_σ increases, the AFC metric increases until

a peak, when $\tau_\sigma = 1.2$. We observe that with a proper threshold, the robustness of alignment is slightly improved compared to without uncertainty filtering.

Feature Dimension In this experiment we analyze how the number of features used in the RF training affects the cost function estimation. We conduct experiments on Set 3 and Set 4 with an increasing number of features randomly selected with distance range threshold $d_{max} = 16$. Results presented in Figure 6.7 show that selecting 2000 features yields optimal performance. When fewer features are used, the estimation is biased due to limited informative features. Using more features does not improve the results. Thus, we set the parameter $|\mathcal{R}| = 2000$ in all other experiments, which results in decent AFC and optimal computational cost.

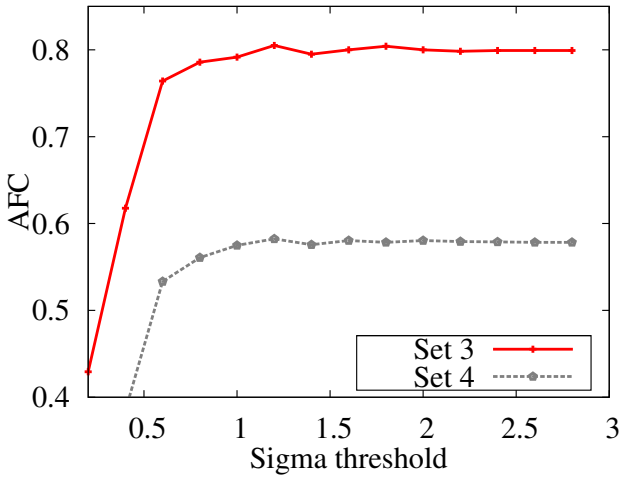


Figure 6.6: Selecting threshold τ_σ.

Correlation Methods for Feature Selection Different correlation methods for RAPID feature selection are studied in this experiment. We compare the simple Pearson correlation coefficient to the other rank correlation coefficients, namely, the Spearman rank correlation and the Kendall rank correlation. The other model parameters are set as follows, $|\mathcal{R}| = 2000$, $d_{max} = 16$ and $\tau_\sigma = 1.2$. We train an appearance model using the Q-DIFF1-based feature. The alignment results are given in Table 6.1, where PC denotes for Pearson correlation, SRC denotes for Spearman rank correlation and K-τ stands for Kendall rank correlation. The results suggest that the Pearson correlation-based method selects more distinctive features for training a good appearance model. Although SRC and K-τ are designed for measuring rank correlations, the alignment results based on these measures are not as reliable as the PC. We conjecture that feature selection based on a linear correlation has a better generalization ability as the AFC result using PC at 1.6σ perturbation level is 2% better than the

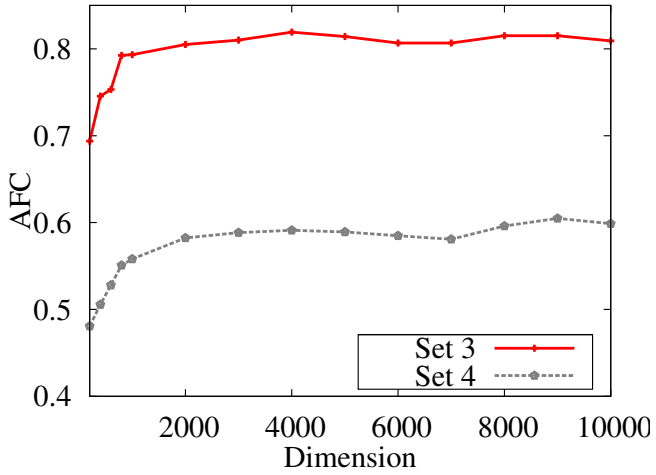


Figure 6.7: Alignment performance vs. feature dimension.

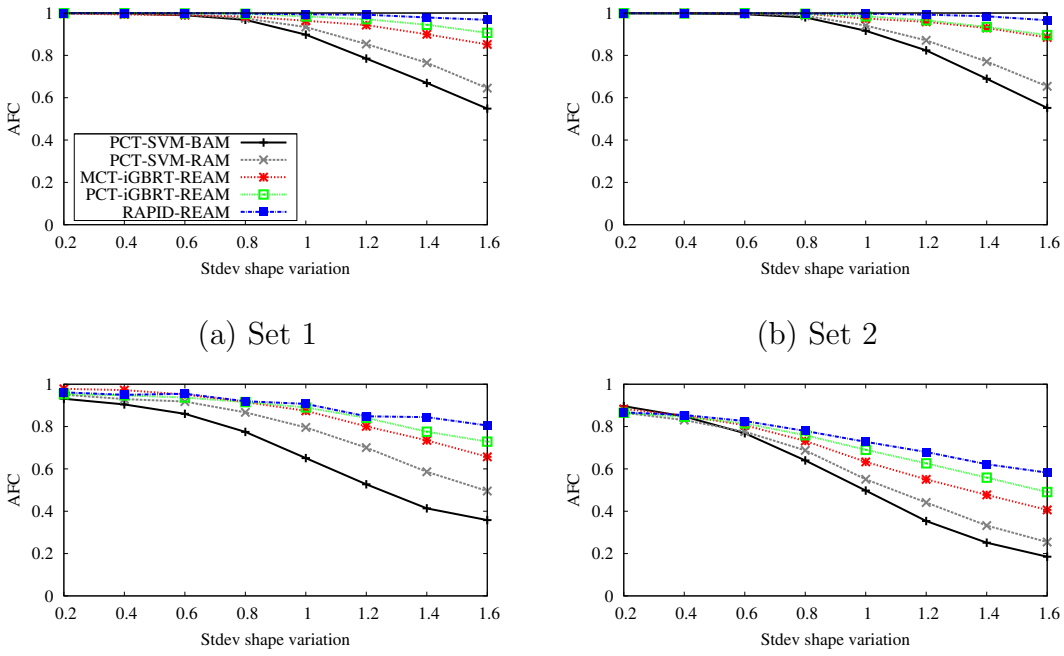
other two rank-based correlations, when tested on Set 4. While on Set 1 and Set 2, the AFC rates are more or less similar.

	PC	SRC	K-τ	PC	SRC	K-τ
Perturbation	0.8σ			1.6σ		
Set 1	**0.997**	0.998	0.997	0.968	**0.974**	0.968
Set 2	0.998	**0.999**	0.996	**0.966**	0.959	0.962
Set 3	**0.920**	0.898	0.888	**0.805**	0.772	0.749
Set 4	**0.779**	0.767	0.773	**0.582**	0.559	0.556

Table 6.1: Comparison of alignment performance in AFC rates resulted from using different correlation methods for selecting feature candidates.

6.5.3.2 Comparison

Comparison of the Proposed Models We finally compare our proposed appearance model to the models based on local structure features such as the MCT and the PCT [GES12]. The average convergence rates at different perturbation levels obtained on all four data sets are plotted in Figure 6.8. The best model proposed in this chapter is denoted as RAPID-REAM, where the Q-DIFF1-based feature is applied. The results are reported with the following



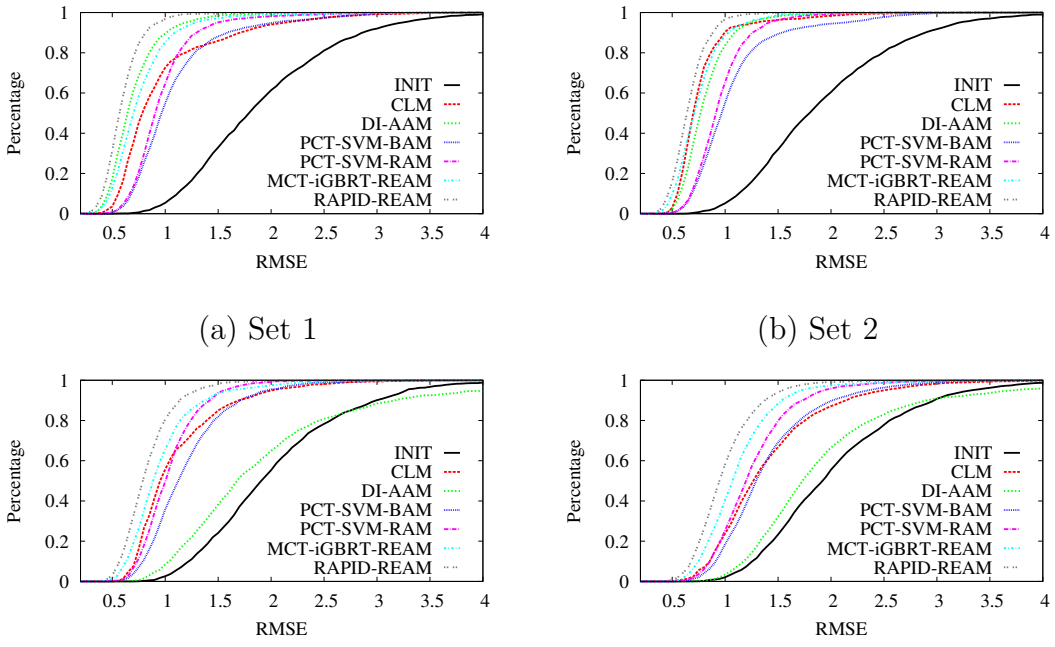
(a) Set 1 (b) Set 2

(c) Set 3 (d) Set 4

Figure 6.8: RAPID-based alignment results on Set 1, Set 2 (first row) and Set 3, Set 4 (second row).

parameter setting: $|\mathcal{R}| = 2000$, $d_{max} = 16$ and $\tau_\sigma = 1.2$. MCT-iGBRT-REAM corresponds to the regression-based appearance model proposed in Chapter 5 using the MCT feature, and PCT-iGBRT-REAM corresponds the model based on the PCT feature. PCT-SVM-BAM denotes the classification-based appearance model proposed in Chapter 3, and PCT-SVM-RAM corresponds to the ranking-based appearance model presented in Chapter 4. From the plots, we notice that the RAPID-based appearance model clearly outperforms all the models based on local structure features. This gives evidence that comparing pixels at a certain distance learns the pixel contextual relationships in the shape-free image coordinates, which is more informative for learning alignment cost functions than comparing pixels in a neighborhood.

Comparison with Other Discriminative Appearance Models In addition, we compare all the models proposed so far in this thesis with two other representative discriminative appearance models. The first model [SG07] is based on a holistic appearance model, which we denote as DI-AAM. The second model [SLC09b] is based on a local patch-based model, which is known as the Constrained Local Model (CLM). We use Set 1 for training DI-AAM and CLM. The same parameter settings are applied for training the models as



(a) Set 1 (b) Set 2

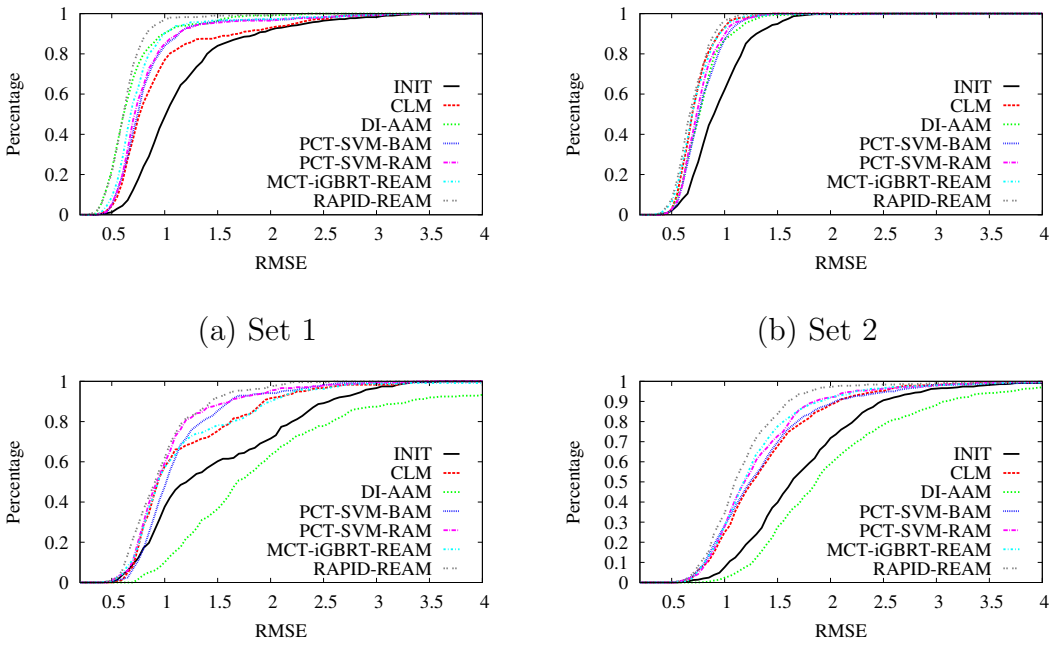
(c) Set 3 (d) Set 4

Figure 6.9: Comparison with other discriminative appearance models, namely, DI-AAM [SG07], and CLM [SLC09b]. The alignments are initialized with 1.6σ perturbation noise.

suggested in both papers[1]. Figure 6.9 plots the cumulative distribution of the landmark errors of alignments on four test sets. The alignments are initialized with 1.6σ perturbation noise. The curve INIT corresponds to the initial cumulative distribution of landmark errors. If we consider RMSE < 1 pixel as converged alignments, our best model, *i.e.*, RAPID-REAM, always outperforms DI-AAM and CLM. We notice that the convergence rates of DI-AAM are close to the rates of RAPID-REAM on Set 1 and Set 2. However, on Set 3 and Set 4, the convergence rates of DI-AAM drop drastically. It is very likely that the model is overfitted on seen subjects trained with suggested parameter settings. CLM has different problems than DI-AAM. When it is tested on Set 2, in which all testing images contain frontal faces, the convergence rate of CLM is quite close to RAPID-REAM. However, the alignment performance of CLM degenerates clearly on Set 1, 3, and 4, in which non-frontal facial images are included. The local patch-based model has difficulties in aligning non-frontal faces, due

[1]We use J. Saragih's implementation for both models, and the source code is available on http://jsaragih.org/

to the simple similarity warping of local patches. In addition to using random perturbation for initializing alignments, we also apply a mean shape for initialization (cf. Figure 6.11). The mean shape is translated and scaled accordingly to cover the face area of a test facial image. Figure 6.10 plots the cumulative distribution of the landmark errors obtained by different models. The curve INIT corresponds to the initial cumulative distribution of landmark errors. Our best model, RAPID-REAM, again outperforms DI-AAM and CLM, as well as other discriminative appearance models presented in the previous chapters.



(a) Set 1 (b) Set 2

(c) Set 3 (d) Set 4

Figure 6.10: Comparison with other discriminative appearance models, namely, DI-AAM [SG07], and CLM [SLC09b]. The alignments are initialized with a mean shape.

6.6 Conclusions

In this chapter, we propose a novel discriminative appearance model based on pixel intensity differences. To ensure that the representation is robust against image noise, we apply a quantization technique on the difference features. The quantized pixel intensity differences are selected and randomly sampled for

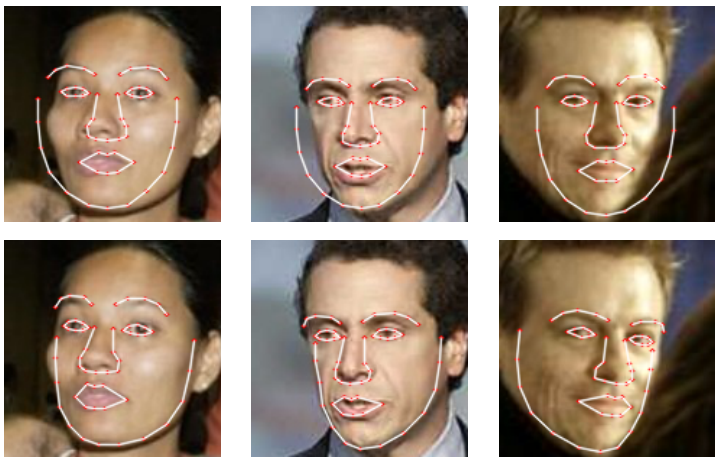


Figure 6.11: Initialized with a mean shape and aligned shapes.

learning a robust alignment cost function. We learn the cost function in an ordinal regression manner using Random Forests. The evaluation of the proposed appearance model is conducted on four different data sets. Experiments show that the quantization technique makes alignment more robust. The appearance model based on the proposed RAPID-feature boosts the alignment performance by about $8.7\% - 17.2\%$ compared with the MCT feature, and about $5.5\% - 10.32\%$ compared with the PCT feature. A comparison with two state-of-the-art discriminative face alignment approaches further demonstrates the superiority of the proposed appearance models in this work.

7 Robustness Analysis and Applications

This chapter is split into two parts, the first part systematically analyzes the influences of different imaging conditions for face alignment, while the second part presents an application for the proposed alignment algorithms. We first compare the alignment performance of different models with an increasing amount of noise added to the original images. Alignment robustness under the occurrence of partial occlusion is studied, in which the occlusion is simulated by rendering random 2D boxes of increasing size. The influence of lighting conditions is also investigated, where additional data are used for evaluation. In the second part, we apply the proposed face alignment algorithms in cross-pose face recognition. We show improved recognition performance with the extension to the PLS-based framework.

7.1 Robustness Analysis

In this section, we analyze face alignment performance of the proposed models under various confounding factors, such as image noise, occlusion, and illumination variations.

7.1.1 Image Noise

By image noise, we mean the random variations in intensity or color values in images. Usually, noise is introduced by the sensor or circuitry of photo scanners or digital cameras. It degenerates the quality of an image by adding additional unwanted signals. In other words, the original image signals are corrupted by the noise signals. Figure 1.1(d) displays a scanned old image, in which noticeable image noise is produced by the scanner. For modern digital cameras, the signal gain is increased for correct exposure under low lighting condition, which significantly increases the salt-and-pepper noise due to photodiode leakage currents. There are also other types of image noise such as Gaussian noise or

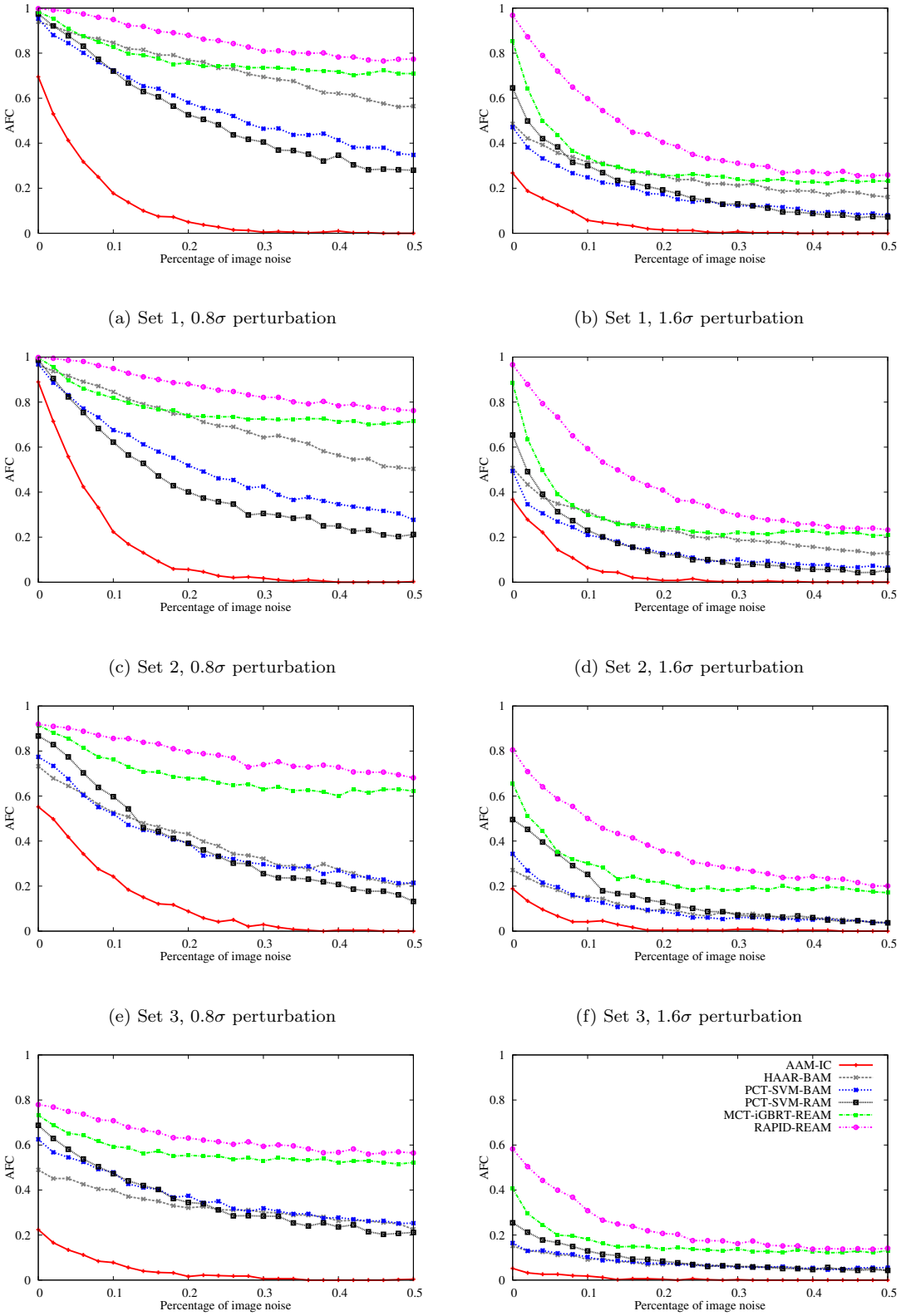
film noise. Because the salt-and-pepper noise often presents in digital images, we study its influence on the robustness of face alignment.

The salt-and-pepper noises are uniformly distributed random white and black pixels in an image. Figure 7.1(b) shows an example of a facial image with such noise added. The noise can be reduced by applying a median filter. In this study, we create salt-and-pepper noise at different levels on the testing images. The noise level is defined by the percentage of noise pixels. Figure 7.1(b), (c), and (d) show noise level at 0.02, 0.2, and 0.4, respectively, where Figure 7.1(a) is the corresponding original image at noise level 0. We notice that, at noise level 0.4, the details of the facial components are rather vague while one can still see the overall structure of a human face.

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

Figure 7.1: Artificial image noise at different levels. (a) original image, (b) 2% noise, (c) 20% noise, (d) 40% noise.

We evaluate face alignment using different models with up to 0.5 noise level. The evaluated models include the generative AAM-SIC, the discriminative HAAR-BAM, PCT-SVM-BAM, PCT-SVM-RAM, MCT-iGBRT-REAM, and RAPID-REAM. At first glance, the pixel comparison based approaches suppose to be sensitive to the salt-and-pepper noise, as each of the features is extracted only based on a few pixel intensity values, which may be easily affected by the increasing noise. However, from Figure 7.2, the appearance models based on the proposed feature representations are actually more robust against noise than the holistic intensity based appearance model (*i.e.* AAM). The rows in Figure 7.2 plot the alignment results (in AFC) on Set 1, Set 2, Set 3, and Set 4, respectively. The plots on the left column display the AFC rates with the shapes initialized at 0.8σ noise level, while the plots on the right column show the AFC rates with the shapes initialized at 1.6σ noise level. The alignment convergence rates of the compared models all decrease as more image noise is added to the testing images, which is expected due to the loss of information. Over all



(a) Set 1, 0.8σ perturbation

(b) Set 1, 1.6σ perturbation

(c) Set 2, 0.8σ perturbation

(d) Set 2, 1.6σ perturbation

(e) Set 3, 0.8σ perturbation

(f) Set 3, 1.6σ perturbation

(g) Set 4, 0.8σ perturbation

(h) Set 4, 1.6σ perturbation

Figure 7.2: Alignment results with noise effects on Set 1, 2, 3, and 4.

data sets and initialization conditions, AAM-SIC fails to converge at all when more than 30% image noise is added. However, the alignments based on the proposed models can still achieve moderate convergence rates. At the highest noise level in this study, where 50% of the pixels are corrupted with noise, the proposed appearance model based on the RAPID feature obtains 56.4% AFC rate on Set 4 at 0.8σ perturbation level, and 14.2% AFC rate at 1.6σ perturbation level. Although significant improvements can be observed when comparing the HAAR-BAM and the other proposed discriminative appearance models, we have to admit that the curves for the MCT-iGBRT-REAM decrease faster than HAAR-BAM at the small image noise levels, especially when the shapes are initialized at 1.6σ perturbation level. We conjecture that a small percentage of image noise does have an impact on the local structure feature-based appearance models. However, when the noise percentage is small, one can easily apply median filters (or bilateral filters) to eliminate the noise.

7.1.2 Occlusion

In real-world applications, the face of a recorded subject can often be partially occluded by accessories such as sunglasses, scarves, or other objects such as a cup. Analyzing the occluded facial images is difficult, as part of the image signals are missing, or in the worst case, replaced with random textures from the occluding objects. Apparently, aligning partially occluded facial images is also not a trivial task. Ekenel *et al.* [ES09] argue that often the problem in recognizing occluded faces is not only the occlusion itself, but also the alignment. Even with a manual alignment, the reference points for alignment are difficult to locate for human annotators, when the facial components are occluded. The experiments in [ES09] show that with proper alignment, face recognition performance under occlusion can be significantly improved.

We study the robustness of our proposed appearance models with partial occlusion. As there is no data set available, which contains occlusion in face images in various regions and scales, we simulate occlusion by placing white boxes of increasing sizes at random locations in the face area. Figure 7.3 shows example images at different occlusion levels. Figure 7.3(b) displays an image with 20% amount of face pixels are occluded. Figure 7.3(c) and Figure 7.3(d) show 40% and 60% of occlusion. Note that in reality, occluded pixels can be arbitrary signals depending on the occluding objects. One could also render virtual objects on the images for more realistic simulation. In this study, we use homogeneous texture only to investigate the behavior of face alignment under missing image signals.

In a similar vein as in Section 7.1.1, we evaluate face alignment with up to 0.6 occlusion level. Figure 7.4 shows the alignment convergence rates on the four

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

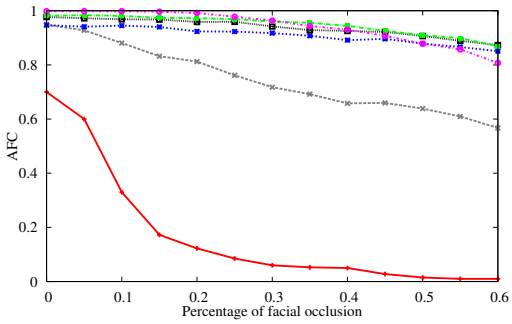
Figure 7.3: Synthetic image occlusion at different levels. (a) 10% occlusion, (b) 20% occlusion, (c) 40% occlusion, (d) 60% occlusion.

benchmarking data sets. Each row corresponds to the AFC results achieved on Set 1, Set 2, Set3, and Set 4, respectively. The left plots show the results with shape initialization at 0.8σ perturbation level, while the right plots corresponds to 1.6σ perturbation level. The convergence rates for AAM-SIC degrade very quickly as the occlusion level increases. When the alignments are initialized further apart from the ground truth (1.6σ), they are hardly converged. On the other hand, we observe relative mild degeneration of alignment performance in the local feature based appearance models. Especially for the MCT-iGBRT-REAM, the AFC rate only decreases about 11.8%, when the occlusion level increases to 0.6 on Set 4 at 1.6σ perturbation level. When the initial shapes are less perturbed, the convergence rates of the MCT-iGBRT-REAM become better than the RAPID with the increasing occlusion level. In other words, the MCT-based regression appearance model is less sensitive to partial occlusion than the RAPID-based model.

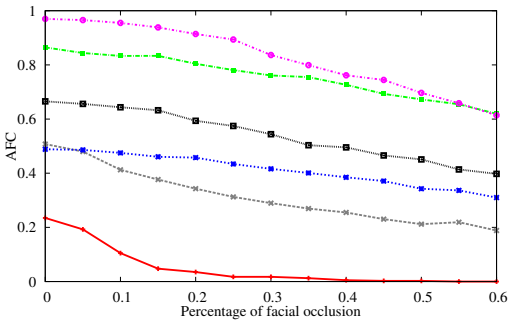
7.1.3 Illumination

Illumination is another crucial factor that affects the robustness of face alignment. Especially, the side illumination produces local shadows, which deform the facial appearance in local regions. It is thus difficult to handle the effects of side illumination for holistic intensity based models due to the non-linearity of the deformations. In this section, we evaluate the proposed models for aligning faces in poorly lit images at different levels.

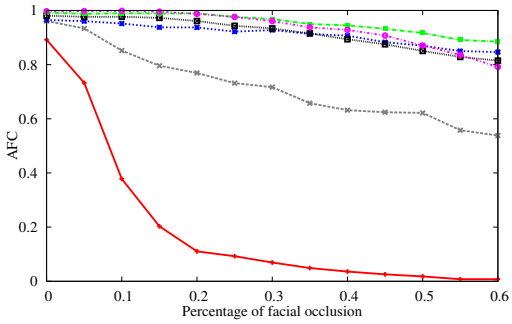
Although the benchmarking data sets used in the previous experiments contain variations of lighting conditions, the evaluation results do not clearly show the



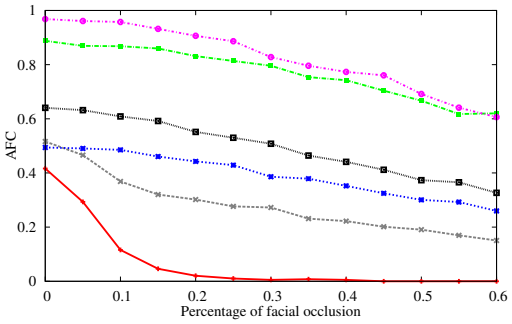
(a) Set 1, 0.8σ perturbation



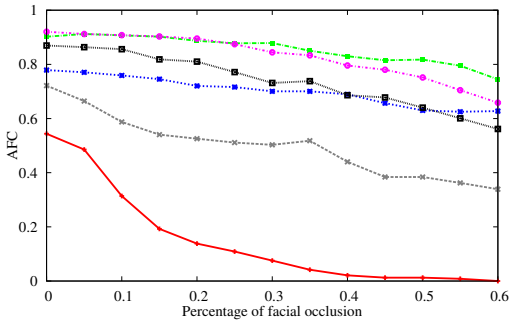
(b) Set 1, 1.6σ perturbation



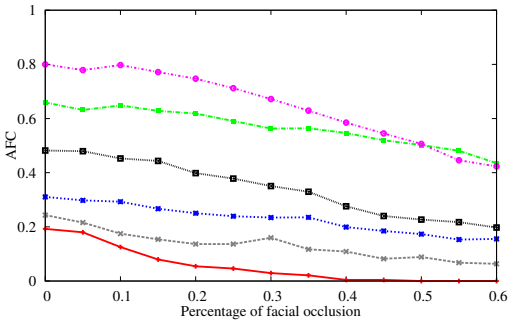
(c) Set 2, 0.8σ perturbation



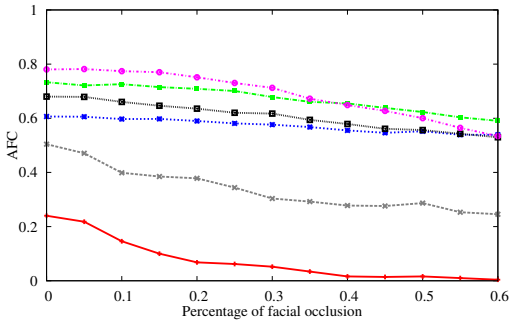
(d) Set 2, 1.6σ perturbation



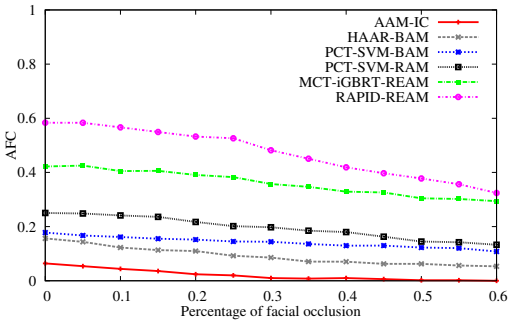
(e) Set 3, 0.8σ perturbation



(f) Set 3, 1.6σ perturbation



(g) Set 4, 0.8σ perturbation



(h) Set 4, 1.6σ perturbation

Figure 7.4: Alignment results with occlusion effects on Set 1, 2, 3 and 4.

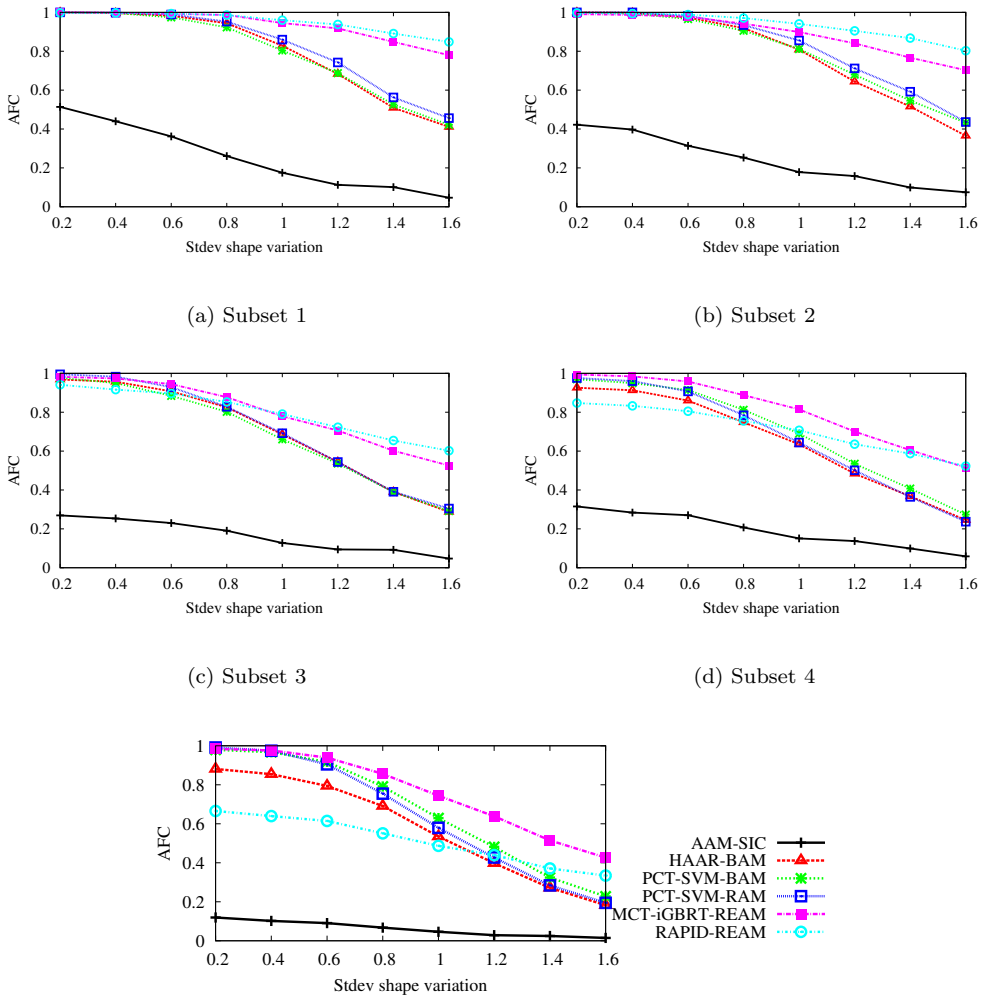
Figure 7.5: Sample images from the extended YaleB database.

alignment behavior of the proposed models under different illumination conditions. On one hand, the data sets are not split specifically for analyzing illumination problems with different levels of lighting effects. In addition, the lighting effects under extreme cases are not considered. We use a subset of the extended Yale face database B [LHK05b] as our evaluation database for this purpose, as it covers illumination variations at different levels.

The extended Yale face database B was collected at the Yale University. The database contains pose and illumination variations. There are 38 subjects in the extended Yale face database B. Illumination variations are obtained by using a geodesic lighting rig with 64 computer controlled strobes. This way, for each person, in each pose, 64 images with different illumination conditions have been captured.

These 64 images are divided into five subsets according to the angle between light source direction and camera's optical axis. Subset 1, with the angles less than 12 degrees, contains seven images. Subset 2, with the angles between 20 and 25 degrees, contains 12 images. Subset 3, with the angles between 35 and 50 degrees, contains 14 images. Subset 4, with the angles between 60 and 77 degrees, contains 12 images and finally subset 5, with the angles larger than 77 degrees, contains 19 images. From the database, frontal face images under all illumination variations were selected. With an increasing subset number,

illumination variations become stronger as can be observed from the sample images in Figure 7.5.



(a) Subset 1

(b) Subset 2

(c) Subset 3

(d) Subset 4

(e) Subset 5

Figure 7.6: Alignment results on the illumination subsets from the extended YaleB database.

The alignment convergence curves achieved on the five illumination subsets are plotted in Figure 7.6. AAM fails to deal with the strong illumination variations, and results in very low AFC rates even at low perturbation levels. The PCT-based appearance models show improved robustness compared to the Haar-based model, especially on subset 3, 4 and 5. This proves the PCT feature representation is less sensitive to illumination changes than the Haar-based representation. However, the PCT feature has also its limitation in dealing with the illumination problem. As the illumination level increases, the performance

gaps between the PCT-based models and MCT and RAPID based models get larger. The performance difference between the PCT-SVM-BAM and PCT-SVM-RAM is getting smaller. This is because the distribution of the PCT features is changed, if a large portion of the facial region is poorly illuminated, which eventually brings a negative impact on the alignment cost function even with a more sophisticated learning algorithm. An interesting observation is that the alignment performance of the RAPID-based appearance model degrades, when the lighting conditions become very poor. The appearance model based on the MCT feature is able to handle tremendous illumination mismatch remarkably well. We conclude from this study that the local structural binary patterns are more robust against large illumination changes than the unbinarized local structure feature. The structure information is still preserved in the binary patterns under the extreme lighting conditions.

7.2 Application: Face Alignment in Cross-pose Face Recognition

Face recognition in frontal face images has achieved considerable success in the past two decades [ZCPR03]. Nowadays, researchers focus more on face recognition under pose variations for real-world applications. Recognizing faces across different poses is still a difficult task due to the nonlinear appearance deformation and self-occlusion in profile face images [ZG09]. It has been shown in [GES09] that aligning face images in different poses using a simple affine transformation results in significant performance drops, when the pose mismatch between the gallery set and the probe set is high. Several approaches have been proposed to solve the pose mismatch problem. The basic idea is to apply transforms between face images in different poses, either in the image space or in the feature space. The typical transforms in the image space are based on nonlinear image warping [GES09, GKSC06] or image synthesis [ECT98]. The typical feature-based transforms are based on regression [FES12, SH08]. In this study, we investigate both type of transforms for cross-pose face recognition using the proposed face alignment algorithm.

7.2.1 Canonical Pose Normalization

The canonical pose normalization method normalizes face images in different pose angles to a single canonical pose, *e.g.* the frontal pose [GES09]. The transform is defined with a nonlinear warping based on a set of localized facial

landmarks. The most straightforward warping method is the piecewise affine warping, which can also be used in the shape fitting for sampling the texture inside a face mesh (cf. Figure 3.7). The warping is realized by mapping the pixels in the fitted and triangulated shape $\mathbf{s}$ to a reference shape $\mathbf{s}_0$. For each pixel $\mathbf{x} = (x, y)^\top$ in a triangle in the reference shape $\mathbf{s}_0$, it finds a unique pixel $W(\mathbf{x}; \mathbf{p}) = \mathbf{x}' = (x', y')^\top$ in the corresponding triangle in the triangulated shape $\mathbf{s}$. The implementation for the piecewise affine warping is detailed in [MB04]. The piecewise affine warping is simple, yet the deformation field is not smooth. Straight lines may be bended across triangle boundaries, which results in unexpected artifacts.

The facial landmarks were localized with AAM fitting in [GES09] with a progressive method for improving the alignment robustness against pose angle. In this work, we show that improved alignment performance leads to superior pose normalization quality, hence increases face recognition accuracy. We align facial images with view-based models, *i.e.*, a frontal-view model and a side-view model, to mitigate the large pose problem. The RAPID-based appearance model is trained due to its superior performance. Annotated images in a subset of the Multi-PIE database [GMC+10] are used for training the frontal and side-view models. Three manually labeled anchor points are used for initializing the alignments, namely, the eye centers and mouth center.

After applying pose normalization, the masked shape-free facial images are obtained, which have a resolution of 120×120 pixels size. The conventional face recognition techniques can be applied for face identification. In a similar vein as in [GES09], we crop out the chin area in the pose normalized facial images, as it does not contribute too much discriminative information compared to other facial regions. Following the approach in [ES06], we scale the cropped images to 64×64 pixels size and then divide them into 64 non-overlapped blocks of 8×8 pixels size. On each local block, the discrete cosine transform (DCT) is performed. The obtained DCT coefficients are ordered using a zig-zag scanning. The first component is skipped, because it represents the average pixel intensity of the entire block. The following ten low frequency coefficients are retained, which yields a ten dimensional local feature vector. Finally, the 64 local feature vectors are concatenated to construct the feature vector of a whole face image. The DCT preserves the total image energy of the processed input block; therefore blocks with different brightness levels lead to DCT coefficients with different magnitudes. In order to balance each local block's contribution to the classification, the local feature vector is normalized to unit norm. To balance the contribution of each DCT frequency, each DCT coefficient is divided by its corresponding standard deviation [ES06] before the unit normalization. The nearest neighbor classifier with the L1 distance metric is used for classification.

The cross-pose face recognition experiments are conducted on a subset of the FERET database [PWHR98]. This subset contains 200 subjects, where each

Figure 7.7: Inconsistency in pose angle annotation in FERET database. Left image (id: 00700) and right image (id: 01042) are selected from subset *bi*, in which the pose angles of all images are annotated with $-60°$. Yet apparently, the annotations are inconsistent and inaccurate.

subject has nine image sessions corresponding to nine different pose angles. As is illustrated in Figure 7.7, due to the way of recording of this database, in which the subjects were asked to facing to different orientations in discrete angles, the variation of actual pose angle in each subset is large. Sample images of a single subject are shown in Figure 7.8. The label *ba* indicates the frontal session, which is used as the gallery set. The remaining eight sessions contain non-frontal face images with different pose angles ($\pm 60\,°$, $\pm 40\,°$, $\pm 25\,°$, and $\pm 15\,°$). Each of these eight sessions is used as probe set. Sample pose normalized images are depicted in Figure 7.9.

In general, fitting shape in near-frontal face images is more robust than in semi-profile face images. In case of alignment in semi-profile face images, even a small misalignment in the chin area may cause a large error in the warped face image. The reason is that the partially self-occluded face part is over-sampled during the warping, which exaggerates the misalignment error. Another fact, which we have noticed is that even if the fitting on a semi-profile face image is perfect, the warped frontal-view face still looks different from the real frontal face due to artifacts. Take the image session *bi* in Figure 7.9 for example: the left half-face is over-sampled and the right half-face is down-sampled after texture warping. This effect makes the left-eye wider and the right-eye narrower, which results in a different local appearance compared to the gallery image *ba* in Figure 7.9.

The recognition results are listed in Table 7.1. The method AAM denotes the system presented in [GES09]. The method DAM denotes the proposed discriminative appearance model with the RAPID feature representation. The method GT stands for the pose normalization with perfect alignments, *i.e.*, the

Figure 7.8: Sample images from the FERET database in nine different pose angles.

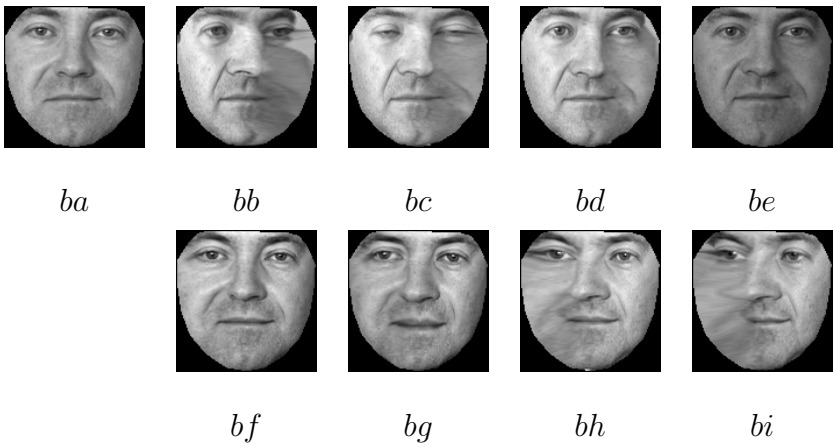


Figure 7.9: Canonical pose normalization.

ground truth shapes. The experiment shows improved recognition rates over all pose angles with the proposed DAM-based alignment compared with the AAM. In particular, the performance gains on large pose angles are remarkable. On the probe set *bb* ($-60\,^{\circ}$), the absolute improvement of the correct recognition rate is 11%, while on the probe set *bi* ($60\,^{\circ}$), the recognition rate has an absolute improvement of 3%. The performance improvements on the other probe sets are less significant; probably the alignment performance is already good and

space for improvement is limited on those sessions. As demonstrated by the GT method, we notice that the canonical pose normalization method has its limit in cross-pose face recognition. Except for the large pose angles, the recognition rates achieved by DAM are already very close to those achieved by using the perfect alignments. The limitation of this approach could be caused by the artifacts introduced in the piecewise affine warping due to the sparse vertices in the adopted shape model. A dense shape model might solve the problem to some degree [BV03], yet the computational cost is high.

Pose session	*bb*	*bc*	*bd*	*be*	*bf*	*bg*	*bh*	*bi*
GT	0.605	0.835	0.985	0.995	0.995	0.965	0.87	0.595
AAM	0.44	0.815	0.93	0.97	0.985	0.915	0.785	0.525
DAM	0.55	0.82	0.945	0.995	0.99	0.945	0.795	0.555

Table 7.1: Cross-pose face recognition with canonical pose normalization on the FERET pose data sets. The columns in this table correspond to the correct recognition rates of different alignment methods tested on different probe sets.

7.2.2 View-based Pose Normalization

To overcome the limitation in the canonical pose normalization approach, we present a view-based pose normalization method. This method combines the image transform and feature transform. Instead of normalizing a face image to a single canonical pose, it splits the continuous pose space into several discrete pose angles. For a given face image, the method warps it to the closest discrete pose angle, and apply feature transform with the warped image. The benefits of this method are three-fold: (1) The view-based image warping mitigates the impacts of artifacts that are introduced in the large pose mismatch. (2) It normalizes the pose variation in a probe set due to inaccurate pose labeling or estimation. (3) The method enables the feature transform method in handling continuous pose, as face images are normalized to shape templates in discrete view angles. A numerable set of view dependent regression models can be applied on the view normalized facial images.

We apply face alignment using multiple view-based discriminative appearance models on images in different poses (pan angles). Three view-based DAMs (in

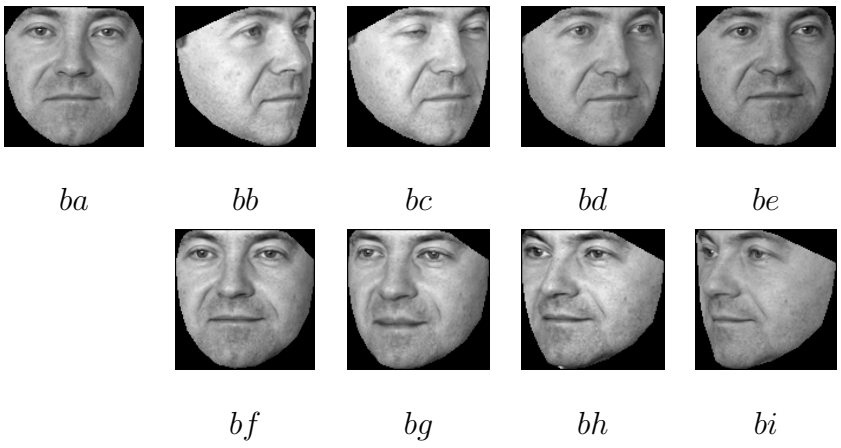


Figure 7.10: View-based pose normalization.

0°, 30°, and 60° pan angles) are trained again using annotated face images in the Multi-PIE database [GMC+10], as this database contains facial images with pose angles (in pan) ranging from −90° to 90° with a step of 15°. After localizing the facial landmarks, we normalize the facial images with their closest view dependent reference shapes. Figure 7.10 lists the view-based pose normalization outputs corresponding to the original images in Figure 7.8. Note as the images are warped using view-dependent shape models, artifacts in canonical pose normalization due to large pose mismatch is avoided. View dependent regression models are applied on the features extracted on the view normalized facial images.

In this study, we adopt partial least squares (PLS) as the regression models for feature transform. PLS is a statistical technique originally proposed as an alternative to ordinary least squares regression in the field of chemometrics [RK05]. The method finds a common vector space for input vectors $\mathbf{x}_i$ and corresponding output vectors (responses) $\mathbf{y}_i$, in a way that the covariance between the projected input vectors and projected output vectors is maximized:

$$[\mathbf{w}, \mathbf{c}] = \underset{\|\mathbf{w}\|=\|\mathbf{c}\|=1}{\arg\max} \left[cov(\mathbf{X}\mathbf{w}; \mathbf{Y}\mathbf{c})\right]^2, \tag{7.1}$$

where $\mathbf{X}$ is a given input matrix and $\mathbf{Y}$ is the corresponding output matrix. $\mathbf{w}$ and $\mathbf{c}$ are the corresponding projection basis vectors, with which the latent scores $\mathbf{t}$ and $\mathbf{u}$ are obtained:

$$\mathbf{t} = \mathbf{X}\mathbf{w} \quad \text{and} \quad \mathbf{u} = \mathbf{Y}\mathbf{c}. \tag{7.2}$$

An iterative algorithm [RK05] is applied for computing the basis vectors of an N dimensional latent space. After N iterations, it computes projection matrices $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)$ and $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_N)$ containing the iteratively computed basis vectors.

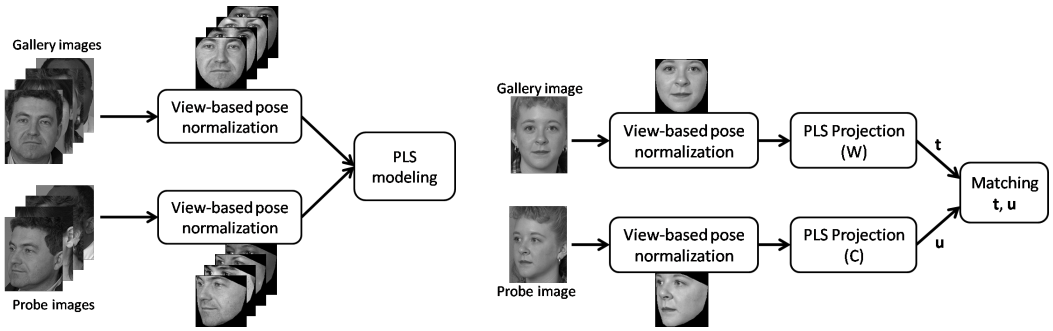


Figure 7.11: Overview of the view-based pose normalization for cross pose face recognition.

One can use PLS as a regression method to predict output vectors $\mathbf{y}$ from input vectors $\mathbf{x}$. However, one can also match features directly in the latent space. In the particular case of pose invariant face recognition, we consider vectors from pose p_0 as $\mathbf{X}$ and vectors of the same faces from a different pose p_1 as $\mathbf{Y}$. We compute PLS projection matrices $\mathbf{W}$ and $\mathbf{C}$ over the data. As matrix $\mathbf{X}$ covariates with matrix $\mathbf{Y}$ in the latent space, it enables us to exploit recognizing face images in different poses by matching the pose-independent latent identity vectors $\tilde{\mathbf{x}} = \mathbf{W}^\top \mathbf{x}$ and $\tilde{\mathbf{y}} = \mathbf{C}^\top \mathbf{y}$, instead of directly on the pose-dependent input vectors $\mathbf{x}$ and $\mathbf{y}$. The PLS based cross-pose face recognition is explored in [FES12], in which superior recognition results are achieved compared to other regression models such as linear regression.

We apply necessary cropping depending on the view angles of the normalized facial images, based on the belief that some part of the region, e.g. one fourth of the image area in the left part of the normalized image from subset *bb* in Figure 7.10(c), might not contribute biometric discriminative information but resulting noise due to facial hair. The raw pixel values are stacked to build feature vectors for learning PLS models as well as testing. Figure 7.11 illustrates the overview idea of the view-based pose normalization for cross-pose face recognition. In the training phase, PLS regression models are learned for the coupled views in gallery and probe sets. In the testing phase, the view normalized gallery images and probe images are projected into the learned latent identity space, in which face matching is applied.

To evaluate the recognition performance of the proposed method, we conduct experiments on the pose subset of the FERET database as in Section 7.2.1. As with in [FES12], for each probe session, we use 100 image pairs (coupled image from gallery and probe set) for training the pose dependent PLS models, and the remaining 100 image pairs for testing.

Pose session	*bb*	*bc*	*bd*	*be*	*bf*	*bg*	*bh*	*bi*
DAM	0.54	0.82	0.95	1.0	1.0	0.97	0.79	0.55
PLS	0.67	0.78	0.80	0.81	0.79	0.8	0.66	0.51
DAM-PLS	0.8	0.83	0.86	0.95	0.94	0.92	0.84	0.82
GT-PLS	0.82	0.92	0.96	0.99	0.97	0.96	0.95	0.83

Table 7.2: Cross-pose face recognition with view-based pose normalization on the FERET pose data sets. The columns in the table correspond to the correct recognition rates of different alignment methods tested on different probe sets.

The average of the aligned shapes in the training set in each pose session is used as the view dependent reference shape. The view-based pose normalization is applied on the images in the training set for training pose dependent PLS models. In testing, the view-based pose normalization is again applied on the testing images, and afterwards the pose dependent PLS models are applied for classification. The intensity values in the cropped holistic images are used as features. We use the nearest neighbour classifier with the L2 distance metric in latent identity vector space.

The recognition results are presented in Table 7.2. The PLS (with a bit abuse of notation) denotes the holistic intensity based approach presented in [FES12], in which face is aligned with affine transformation. The DAM-PLS method denotes the view-based pose normalization using RAPID-based DAM, while the GT-PLS approach uses ground truth for the view-based pose normalization. The results show that the PLS method is sensitive to the pose variation in each probe session. The recognition performance on the small pose angle sessions, such as *be* and *bf*, drops by around 20%, compared with the canonical pose normalization with DAM. On the other hand, however, this method achieved better results on the probe set *bb* than any of the canonical pose normalization methods. This implies that the PLS regression model is effective in solving the cross-pose face recognition problem, when the pose variation in a probe set is small. The proposed DAM-PLS method outperforms the PLS method on all eight probe sets. Especially, the recognition performance on set *be* increases 13% and 31% on set *bi*. The absolute improvement is 14.25% in average. However, the recognition rates on the sessions with smaller pose angles are not as good as expected, comparing to the methods based on the canonical pose normalization. The reason for this observation is that the PLS is a subspace based regression

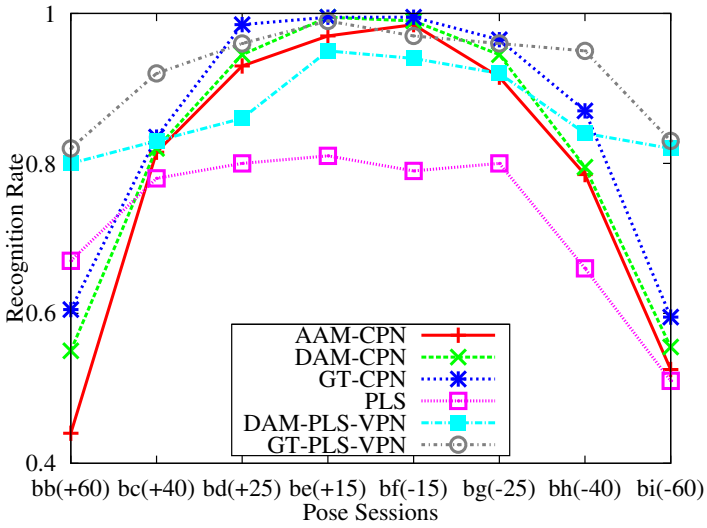


Figure 7.12: Cross-pose face recognition with canonical and view-based pose normalization on the FERET pose data sets. The horizontal axis corresponds to different pose sessions as probe sets. The vertical axis denotes the corresponding correct recognition rates.

model, which is also sensitive to alignment errors. The results with the GT-PLS method show that perfect alignments provides as good recognition rates on small pose angles as the canonical normalization approach. However, for semi-profile faces, the performance gains of using the proposed view-based pose normalization are significant. For the sake of clear comparison, the recognition rates listed in Table 7.1 and Table 7.2 are plotted together in Figure 7.12. Note that the results in Table 7.1 and Table 7.2 are not directly comparable, due to different number of test images. However, as recognition performance based on DAM are more or less consistent in these two tables, an indirect comparison is still valid. Based on the above observations, we recommend a hybrid solution for cross-pose face recognition. When the facial pose is small (within a range of $\pm 25\,^{\circ}$), the canonical pose normalization is favorable; when the pose angle is large, the view-based normalization method is favorable. Moreover, we expect the recognition performance to be improved further with advanced features in stead of raw intensity values.

7.3 Conclusions

We conduct robustness analysis for the proposed discriminative appearance models. The influential factors such as image noise, occlusion, and lighting conditions are considered. The synthetic image noise and occlusion is generated at different levels in the experiments. Results show that the proposed local gradient feature based appearance representation is more robust against noise and occlusion compared to the local region or holistic representation. In terms of illumination, we observe the MCT-based appearance model is superior to PCT-based and RAPID-based models. In the second part of this chapter, we apply the proposed discriminative appearance model face alignment in cross-pose face recognition. We show improved recognition performance with canonical pose normalization method using the DAM-based face alignment. A view-based pose normalization approach is also presented combined with the PLS-regression model. Experiments show that this approach significantly improves recognition performance, when the testing face pose angles are large.

8 Conclusions

In this thesis, we present discriminative appearance models for aligning facial images robustly under different imaging conditions. We address the robustness and generalization problems of face alignment. The proposed appearance models use feature representations based on local gradient features such as the PCT, MCT, and RAPID. The gradient features are based on pixel value comparisons, which provide robustness against changes in illumination. Due to the locality, the local gradient features are less sensitive to appearance variations in local regions caused by partial face occlusion or facial expression. Another important contribution in this thesis is the discriminative modeling of the features for learning deformable appearance models with enhanced generalization capacity. Unlike the generative modeling, the use of discriminative models reduces the dimension of the search space on one hand, on the other hand, it enables a framework for learning an alignment cost function with the desired properties. The discriminative appearance modeling is explored in three different perspectives of machine learning problems, *i.e.* classification, ranking, and regression.

To evaluate the proposed discriminative appearance models in face alignment, we prepare and propose a benchmarking data set, which includes images collected from four publicly available face databases, namely, the FERET [PWHR98], FRGC [PFS+05], IMM [SEL03], and LFW database [HRBLM07]. Extensive experiments have been carried out to analyze the effects of the training parameters for different models on the alignment performance. The parameters include the size of the reference shape, number of features selected in the model, number and distribution of training samples, etc. As we are using the local gradient based features, a proper size for the reference shape should be used as suggested by the experimental results. A small size may lose the details of facial appearance, while a large size may introduce noise in the model. For the PCT-based classification appearance model, a reference shape with 35 pixels width achieved the best alignment performance. The feasibility of applying a larger reference shape is enabled by the use of local structural features, which have a low dimensional configuration space. This makes the appearance learning tractable in the boosting framework, while the Haar based feature might be intractable due to the high configuration dimension. The experimental results also show that the PCT based appearance model outperforms the Haar based model.

To mitigate the imbalanced data problem in the classification based appearance models, we investigate ranking based models by learning partial ordering of alignments. The ranking appearance model is realized with pairwise classification, which classifies correctness of ordering paired alignments. The ranking based modeling also enforces constraints on the incorrectly aligned data in learning the alignment cost function, which results in a smoother score function than in the classification based models. Experiments show superior alignment performance of the ranking appearance models. We also show that extracting more alignment pairs for training using random permutation improves alignment performance slightly.

The regression-based appearance model enforces even more constraints on the alignment cost function learning, where the response target of the regression model is defined with *e.g.* a triangle function. The gradient boosted regression trees (GBRT) is adopted for learning our regression model due to its success in solving learning to rank problems. The random forests technique is used to initialize the GBRT training iterations. The initialization provides the GBRT with an initial estimation with low bias and requires less iterations to converge to the global optimum. The experimental results show that the regression trees-based appearance models significantly improve the robustness and accuracy in terms of face alignment. Our best proposed model (PCT-iGBRT-REAM) boosts the alignment performance by about $23.4\% - 26.1\%$ on different data sets compared to the model based on pairwise ordinal classification (PCT-SVM-RAM). To learn appearance models with features, which have semantic meanings, we propose another regression based model, where the model representation is based on the RAPID features. Instead of using a gradient feature in a local adjacent region, the RAPID features are extracted between pixel locations at a certain distance. The experiments demonstrate that the RAPID based appearance model is more robust in face alignment than local structural features such as the PCT and MCT on the benchmarking data sets. Furthermore, we show in the experiments that the RAPID-based appearance model outperforms two state-of-the-art discriminative face alignment models.

To analyze the properties of the proposed discriminative appearance models further, we thoroughly evaluate the alignment robustness under various imaging conditions, such as image noise, partial occlusion, and lighting. Through the experiments, we found out that the proposed feature and appearance modeling is robust against these confounding factors. In particular, fitting with regression based appearance models still has a decent convergence rate under large occlusion and a large amount of image noise. In addition, we found out that although random pixel intensity features achieved best results on the benchmarking data set with moderate illumination changes, it is less robust than the local structural features, when tested on extreme lighting variation presented in the extended YaleB database.

As an application for the proposed discriminative appearance model, we applied the alignment for pose normalization in cross-pose face recognition. The experimental results show that the improved alignment results enhance the recognition performance. In addition, we extend the cross-pose face recognition by using partial least squares (PLS) for learning a latent space, which maximizes correlation between different view-points. The view-based normalization mitigates the weakness in the original PLS-based approach, in which a discrete and precise pose estimating is required. This makes the PLS-based framework more applicable in real-world applications.

8.1 Future Work

There are several possibilities to further improve or extend the proposed discriminative appearance models. The Nelder-Mead simplex-based method is sensitive to initialization. Applying a stochastic Nelder-Mead method might help the optimization avoiding some local extrema. The current regression-based appearance model learns a single regression model, which is suboptimal for approximating a local extrema free cost function. A cascaded or stage-wise regression model might be a better solution. For the RAPID-based appearance models, more sophisticated feature selection methods can be applied. Good features should not only be correlated to the regression targets, but also be less dependent from the selected candidate feature set. The artifacts from the piecewise affine warping on semi-profile facial images should also be considered. Features extracted on the self-occluded facial part can be less reliable for contributing to learning a smooth cost function. Finally, a 3D shape model can be applied to mitigate pose ambiguity to some degree.

Bibliography

[ADD04] B. Abboud, F. Davoine, and M. Dang, "Facial expression recognition and synthesis based on an appearance model," *Signal Processing: Image Communication*, vol. 19, no. 8, pp. 723–740, 2004.

[AGB+12] N. M. Arar, F. Güney, N. K. Bekmezci, H. Gao, and H. K. Ekenel, "Real-time face swapping in video sequences - Magic mirror," in *Proc. of Int. Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, 2012, pp. 48–53.

[ALC+07] A. Ashraf, S. Lucey, J. Cohn, T. Chen, Z. Ambadar, K. Prkachin, P. Solomon, and B.-J. Theobald, "The painful face: pain expression recognition using active appearance models," in *Proc. of Int. Conference on Multimodal interfaces*, 2007, pp. 9–14.

[AZC+08] U. Agris, J. Zieren, U. Canzler, B. Bauer, and K.-F. Kraiss, "Recent developments in visual sign language recognition," *Universal Access in the Information Society*, vol. 6, no. 4, pp. 323–362, 2008.

[BGM03] S. Baker, R. Gross, and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework: Part 3," Carnegie Mellon University Robotics Institute, Tech. Rep. CMU-RI-TR-03-35, 2003.

[BGS+02] R. Beichel, G. Gotschuli, E. Sorantin, F. Leberl, and M. Sonka, "Diaphragm dome surface segmentation in CT data sets: A 3-D active appearance model approach," in *Proc. of SPIE Medical Imaging 2002: Image Processing*, vol. 4684, 2002, pp. 475–484.

[BJKK11] P. N. Belhumeur, D. W. .Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consesus of examplars," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 545–552.

[BM01] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 1090–1097.

[BM04] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," *Int. Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[Bre84] L. Breiman, *Classification and regression trees.* Chapman & HallCRC, 1984.

[Bre96] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996.

[Bre01] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[BV03] V. Blanz and T. Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, Sep. 2003.

[CBET99] T. F. Cootes, C. Beeston, G. J. Edwards, and C. J. Taylor, "A unified framework for atlas matching using active appearance models," in *Proc. of Int. Conference on Information Processing in Medical Imaging*, 1999, pp. 322–333.

[CBHK02] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[CC06] D. Cristinacce and T. F. Cootes, "Feature detection and tracking with constrained local models," in *Proc. of British Machine Vision Conference*, 2006, p. 929938.

[CC11] O. Chapelle and Y. Chang, "Yahoo! learning to rank challenge overview," *JMLR Workshop and Conference Proceedings: Proceedings of the Yahoo! Learning to Rank Challenge*, vol. 14, pp. 1–24, Jun. 2011.

[CD05] C. M. Christoudias and T. Darrell, "On modelling nonlinear shape-and-texture appearance manifolds," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2005, pp. 1067–1074.

[CET98a] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *Proc. of* 5^{th} *European Conference on Computer Vision*, vol. 2, 1998, pp. 484–498.

[CET98b] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "A comparative evaluation of active appearance model algorithms," *Proc. of British Machine Visin Conference*, vol. 2, pp. 680–689, 1998.

[CILS12] T. F. Cootes, M. Ionita, C. Lindner, and P. Sauer, "Robust and accurate shape model fitting using random forest regression voting," in *Proc. of 12th European Conference on Computer Vision*, vol. 7, 2012, pp. 278–291.

[CLR+04] H. Chen, Z. Liu, C. Rose, Y. Xu, H. Y. Shum, and D. Salesin, "Example-based composite sketching of human portraits," in *Proc. of Int. Symposium on Non-photorealistic Animation and Rendering*, 2004, pp. 95–153.

[CQL+07] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proc. of the 24th Int. Conference on Machine learning*, ser. ICML '07, 2007, pp. 129–136.

[CT92] T. F. Cootes and C. J. Taylor, "Active shape models," in *Proc. of 3rd British Machine Vision Conference*, 1992, pp. 266–275.

[CT01a] T. F. Cootes and C. J. Taylor, "Constrained active appearance models," in *Proc. of IEEE Int. Conference on Computer Vision*, vol. 1, 2001, pp. 748–754.

[CT01b] T. F. Cootes and C. J. Taylors, "On representing edge structure for model matching," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 1114–1119.

[CTCG95] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[CWT00] T. F. Cootes, K. Walker, and C. J. Taylor, "View-based active appearance models," in *Proc. of 4^{th} Int. Conference on Automatic Face and Gesture Recognition*, Grenoble, France, 2000, pp. 227–232.

[CWWS12] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2887–2894.

[DD95] D. F. Dementhon and L. S. Davis, "Model-based object pose in 25 lines of code," *Int. Journal of Computer Vision*, vol. 15, no. 1-2, pp. 123–141, 1995.

[DGFG12] M. Dantone, J. Gall, G. Fanelli, and L. J. V. Gool, "Real-time facial feature detection using conditional regression forests," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2578–2585.

[DKB07] G. Dedeoglu, T. Kanade, and S. Baker, "The asymmetry of image registration and its application to face tracking," *EEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 807–823, 2007.

[DLSE04] S. Darkner, R. Larsen, M. B. Stegmann, and B. K. Ersboll, "Wedgelet enhanced appearance models," in *Proc. IEEE Int. Conference on Computer Vision and Pattern Recognition Workshop*, 2004, pp. 177–177.

[DR07] D. Datcu and L. Rothkrantz, "Facial expression recognition in still pictures and videos using active appearance models: A comparison approach," in *Proc. of Int. Conference on Computer Systems and Technologies*, 2007, pp. 112–112.

[DRL+06] R. Donner, M. Reiter, G. Langs, P. Peloschek, and H. Bischof, "Fast active appearance model search using canonical correlation analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1690–1694, 2006.

[DWP10] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1078–1085.

[ECT98] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Face recognition using active appearance models," in *Proc. of* 5^{th} *European Conference on Computer Vision*, vol. LNCS-Series 1406-1607, 1998, pp. 581–595.

[ECT99] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Advances in active appearance models," in *Proc. of Int. Conference on Computer Vision*, vol. 1, 1999, pp. 137–142.

[Elk01] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. of Int. Joint Conference on Artificial Intelligence*, 2001, pp. 973–978.

[ES06] H. K. Ekenel and R. Stiefelhagen, "Analysis of local appearance-based face recognition: Effects of feature selection and feature normalization," in *Proc. of CVPR Biometrics Workshop*, 2006.

[ES09] H. K. Ekenel and R. Stiefelhagen, "Face alignment by minimizing the closest classification distance," in *Proc. of Int. Conference on Biometrics: Theory, Applications and Systems (BTAS09)*, 2009.

[EZ06] M. R. Everingham and A. Zisserman, "Regression and classification approaches to eye localization in face images," in *Proc. of Int. Conference on Automatic Face and Gesture Recognition (AFGR)*, 2006, pp. 441–448.

[FE73] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Trans. on Computers*, vol. 22, no. 1, pp. 67–92, 1973.

[FE04] B. Fröba and A. Ernst, "Face detection with the modified census transform," in *Proc. of* 6^{th} *Int. Conference on Automatic Face and Gesture Recognition*, 2004, pp. 91–96.

[FES12] M. Fischer, H. K. Ekenel, and R. Stiefelhagen, "Analysis of partial least squares for pose-invariant face recognition," in *Proc. of IEEE Int. Conference on Biometrics: Theory, Applications and Systems*, 2012.

[FH05] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *Int. Journal of Computer Vision (IJCV)*, vol. 61, no. 1, pp. 55–79, 2005.

[FHT00] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[FISS03] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," *Journal of Machine Learning Research*, vol. 4, pp. 933–969, Dec. 2003.

[Fri00] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, pp. 1189–1232, 2000.

[FS97] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[GEFS11] H. Gao, H. K. Ekenel, M. Fischer, and R. Stiefelhagen, "Boosting pseudo census transform features for face alignment," in *Proc. of British Machine Vision Conference*, Dundee, UK, 2011.

[GES09] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Pose normalization for local appearance-based face recognition," in *Proc. of Int. Conference on Advances in Biometrics*, 2009, pp. 32–41.

[GES12] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Face alignment using a ranking model based on regression trees," in *Proc. of British Machine Vision Conference*, Guildford, UK, 2012.

[GKSC06] J. Guillemaut, J. Kittler, M. T. Sadeghi, and W. J. Christmas, "General pose face recognition using frontal face model," in *Proc. of* 11^{th} *Iberoamerican Congress in Pattern Recognition*, vol. 4225/2006, 2006, pp. 79–88.

[GL09] J. Gall and V. S. Lempitsky, "Class-specific hough forests for object detection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1022–1029.

[GlTFM+07] J. Gonzalez, F. D. la Torre Frade, R. Murthi, N. GuilMata, and E. Zapata, "Bilinear active appearance model," in *Proc. IEEE Workshop on Non-rigid Registration Tracking Learning*, 2007.

[GMB05] R. Gross, I. Matthews, and S. Baker, "Generic vs. person specific active appearance models," *Image and Vision Computing*, vol. 23, no. 12, pp. 1080–1093, 2005.

[GMB06] R. Gross, I. Matthews, and S. Baker, "Active appearance models with occlusion," *Image and Vision Computing*, vol. 24, no. 6, pp. 593–604, 2006.

[GMC+10] R. Gross, I. Matthews, J. F. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.

[Goo91] C. Goodall, "Procrustes methods in the statistical analysis of shape," *Journal of the Royal Statistical Society, Siries B*, vol. 53, no. 2, pp. 285–339, 1991.

[GZSM07] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, pp. 2234–2240, 2007.

[HD05] S. Hamlaoui and F. Davoinen, "Facial action tracking using an AAM-based condensation approach," in *Proc. of IEEE Int. Conference on Acoustics, Speech, and Signal Processing*, vol. 2, 2005, pp. 701–704.

[HG09] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[HGO00] R. Herbrich, T. Graepel, and K. Obermayer, "Large margin rank boundaries for ordinal regression," *Advances in Large Margin Classifiers*, pp. 115–132, 2000.

[HJ10] D. W. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 478–500, 2010.

[HLZ01] X. Hou, S. Z. Li, and H. Zhang, "Direct appearance models," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 828–833.

[HRBLM07] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, Oct. 2007.

[HXM+04] C. Hu, J. Xiao, I. Matthews, S. Baker, J. F. Cohn, and T. Kanade, "Fitting a single active appearance model simultaneously to multiple images," in *Proc. of British Machine Vision Conference*, 2004.

[Kan73] T. Kanade, "Picture processing by computer complex and recognition of human faces," 1973, phD. Thesis.

[KBM+05] S. C. Koterba, S. Baker, I. Matthews, C. Hu, J. Xiao, J. Cohn, and T. Kanade, "Multi-view AAM fitting and camera calibration," in *Proc. of Int. Conference on Computer Vision*, 2005, pp. 511–518.

[Ken84] D. G. Kendall, "Shape manifolds, procrustean metrics, and complex projective spaces," *Bulletin of the London Mathematical Society*, vol. 16, no. 2, pp. 81–121, 1984.

[KG05] F. Kahraman and M. Gokmen, "Illumination invariant face alignment using multi-band active appearance model," in *Proc. of Int. Conference on Pattern Recognition and Machine Learning*, 2005, p. 118127.

[KGDL07] F. Kahraman, M. Gokmen, S. Darkner, and R. Larsen, "An active illumination and appearance (AIA) model for face alignment," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–7.

[KLT03] T. G. Kolda, R. M. Lewis, and V. Torczon, "Optimization by direct search: New perspectives on some classical and modern methods," *SIAM Review*, vol. 45, pp. 385–482, 2003.

[KnC06] P. Kittipanya-ngam and T. F. Cootes, "The effect of texture representations on AAM performance," in *Proc. Int. Conference on Pattern Recognition*, vol. 2, 2006, pp. 328–331.

[KWRB11] M. Köestinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *Proc. of First IEEE Int. Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[KWT88] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. Journal of Computer Vision*, vol. 8, no. 2, pp. 321–331, 1988.

[LBW07a] P. Li, C. Burges, and Q. Wu, "Learning to rank using classification and gradient boosting," in *Proc. of the Int. Conference on Advances in Neural Information Processing Systems (NIPS)*, 2007.

[LBW07b] P. Li, C. J. C. Burges, and Q. Wu, "Mcrank: Learning to rank using multiple classification and gradient boosting," in *Proc. of the Int. Conference on Advances in Neural Information Processing Systems*, 2007.

[LCK+10] P. Lucey, J. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010, pp. 94–101.

[LHK05a] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 27, no. 5, pp. 684–698, 2005.

[LHK05b] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 684–698, 2005.

[Liu07] X. Liu, "Generic face alignment using boosted appearance model," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07)*, 2007, pp. 1–8.

[Liu09] T.-Y. Liu, "Learning to rank to information retrieval," *Foundations and Trends in Information Retrieval*, vol. 3, no. 3, pp. 225–331, 2009.

[LJJ06] J. Liebelt, X. Jing, and Y. Jie, "Robust AAM fitting by fusion of images and disparity data," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2483–2490.

[LK08] H. Lee and D. Kim, "Expression-invariant face recognition by facial expression transformations," *Pattern Recognition Letters*, vol. 29, no. 13, pp. 1797–1805, 2008.

[LKP03] R. Lienhart, A. Kuranov, and V. Pisarevsky, "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," in *Proc. of* 25^{th} *Pattern Recognition Symp.*, 2003, pp. 297–304.

[LLS06] B. Leibe, A. Leonardis, and B. Schiele, "An implicit shape model for combined object categorization and segmentation," *Towards Category-Level Object Recognition*, pp. 496–510, 2006.

[LLS08] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *Int. Journal of Computer Vision (IJCV)*, vol. 77, no. 1-3, pp. 259–289, 2008.

[Low04] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[LRBS09] K. Luu, K. Ricanek, T.-D. Bui, and C.-Y. Suen, "Age estimation using active appearance models and support vector machine regression," in *Proc. of Int. Conference on Biometrics: Theory, applications and systems (BTAS)*, 2009, pp. 1–5.

[MB04] I. Matthews and S. Baker, "Active appearance models revisited," *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 135–164, 2004.

[MBV06] I. Macedo, E. V. Brazil, and L. Velho, "Expression transfer between photographs through multilinear AAMs," in *Proc. of Computer Graphics and Image Processing*, 2006, pp. 239–246.

[MCW11] A. Mohan, Z. Chen, and K. Q. Weinberger, "Web-search ranking with initialized gradient boosted regression trees," *JMLR Workshop and Conference Proceedings: Proceedings of the Yahoo! Learning to Rank Challenge*, vol. 14, pp. 77–89, Jun. 2011.

[MMK+99] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTS: The extended M2VTS database," in *Int. Conference on Audio and Video-based Biometric Person Authentication (AVPBA)*, 1999.

[MP97] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 19, no. 7, pp. 696–710, 1997.

[MR03] R. Meir and G. Rätsch, "An introduction to boosting and leveraging," in *Advanced lectures on machine learning*, S. Mendelson and A. Smola, Eds. New York, NY, USA: Springer-Verlag, 2003, pp. 118–183.

[NlTF08] M. H. Nguyen and D. la Torre Fernando, "Local minima free parameterized appearance models," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[NM65] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, pp. 308–313, 1965.

[OPH96] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[PFS+05] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 947–954.

[PWHR98] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *Image and Vision Computing*, vol. 16, no. 5, pp. 295–306, 1998.

[RCA03] M. G. Roberts, T. F. Cootes, and J. E. Adams, "Linking sequences of active appearance sub-models via constraints: An application in automated vertebral morphometry," in *Proc. of British Machine Vision Conference*, vol. 1, 2003, pp. 349–358.

[RK05] R. Rosipal and N. Krämer, "Overview and recent advances in partial least squares," in *Proc. Subspace, Latent Structure and Feature Selection: Statistical and Optimization Perspectives Workshop*, 2005, pp. 1–6.

[RPG00] S. Romdhani, A. Psarrou, and S. Gong, "On utilising template and feature-based correspondence in multi-view appearance models," in *Proc. of European Conference on Computer Vision*, vol. 1, 2000, pp. 799–813.

[SCT03] I. M. Scott, T. F. Cootes, and C. J. Taylor, "Improving appearance model matching using local image structure," in *Proc. of* 18*th Int. Conference on Information Processing in Medical Imaging*, vol. 2732, 2003, pp. 258–269.

[SEL03] M. B. Stegmann, B. K. Ersboll, and R. Larsen, "FAME - A flexible appearance modeling enviroment," *IEEE Trans. on Medical Imaging*, vol. 22, no. 10, pp. 1319–1331, 2003.

[SG07] J. Saragih and R. Goecke, "A nonlinear discriminative approach to AAM fitting," in *Proc. of Int. Conference on Computer Vision*, 2007, pp. 1–8.

[SH08] M. S. Sarfraz and O. Hellwich, "Statistical appearance models for automatic pose invariant face recognition," in *Proc. of IEEE Int. Conference on Automatic Face and Gesture*, 2008, pp. 1–6.

[SK06] J. Sung and D. Kim, "Large motion object tracking using active contour combined active appearance model," in *Proc. of IEEE Int. Conference on Computer Vision Systems*, 2006, pp. 31–37.

[SK08] J. Sung and D. Kim, "Pose-robust facial expression recognition using view-based 2D + 3D AAM," *IEEE Trans. on System, Man, and Cybernetics - Part A: System and Humans*, vol. 38, no. 4, pp. 852–866, 2008.

[SL03] M. B. Stegmann and R. Larsen, "Multi-band modelling of appearance," *Image and Vision Computing*, vol. 21, no. 1, pp. 61–67, 2003.

[SLC09a] J. M. Saragih, S. Lucey, and J. F. Cohn, "Enforcing convexity for improved alignment with constrained local models," in *Proc. of Int. Conference on Computer Vision*, 2009, pp. 2248–2255.

[SLC09b] J. M. Saragih, S. Lucey, and J. F. Cohn, "Face alignment through subspace constrained mean-shifts," in *Proc. of Int. Conference on Computer Vision*, 2009, pp. 1034–1041.

[STCZ12] F. Song, X. Tan, S. Chen, and Z.-H. Zhou, "A literature survey on robust and efficient eye localization in real-life scenarios," NUAA, Tech. Rep., 2012.

[Ste01] M. B. Stegmann, "Object tracking using active appearance models," in *Proc. of Danish Conference on Pattern Recognition and Image Analysis*, vol. 1, 2001, pp. 54–60.

[SYW96] R. Stiefelhagen, J. Yang, and A. Waibel, "A modelbased gaze tracking system," in *Proc. of IEEE Int. Joint Symposia on Intelligence and Systems*, 1996, pp. 304–310.

[TMCB07] B. Theobald, I. A. Matthews, J. F. Cohn, and S. M. Boker, "Real-time expression cloning using appearance models," in *Proc. of Int. Conference on Multimodal Interfaces*, 2007, pp. 134–139.

[Vap98] V. N. Vapnik, *Statistical learning theory.* New York: John Wiley & Sons, 1998.

[VJ04] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[VMBP10] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2729–2736.

[WLC07] Y. Wang, S. Lucey, and J. F. Cohn, "Non-rigid object alignment with a mismatch template based on exhaustive local search," in *Proc. of Int. Conference on Computer Vision*, 2007, p. 18.

[WLC08] Y. Wang, S. Lucey, and J. F. Cohn, "Enforcing convexity for improved alignment with constrained local models," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[WLD08] H. Wu, X. Liu, and G. Doretto, "Face alignment via boosted ranking model," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[WT99] C. B. H. Wolstenholme and C. J. Taylor, "Wavelet compression of active appearance models," in *Proc. 2nd Int. Conference Med. Image Comput. Comput.-Assisted Intervention*, 1999, pp. 544–554.

[XBMK04] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2D+3D active appearance models," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 535–542.

[XQ07] J. Xie and Z. Qiu, "The effect of imbalanced data sets on lda: A theoretical and empirical analysis," *Pattern Recognition*, vol. 40, no. 2, pp. 557–562, Feb. 2007.

[YHC92] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable template," *Int. Journal of Computer Vision (IJCV)*, vol. 8, no. 2, pp. 99–111, 1992.

[ZCPR03] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, Dec. 2003.

[ZCSZ07] Z. Zheng, K. Chen, G. Sun, and H. Zha, "A regression framework for learning ranking functions using relative relevance judgments," in *Proc. of the 30th Int. ACM SIGIR Conference on Research and development in information retrieval*, ser. SIGIR '07, 2007, pp. 287–294.

[ZG04] Z.-H. Zhou and X. Geng, "Projection functions for eye detection," *Pattern Recognition*, vol. 37, no. 1, pp. 1049–1056, 2004.

[ZG09] X. Zhang and Y. Gao, "Face recognition across pose: A review," *Pattern Recognition*, vol. 42, no. 11, pp. 2876–2896, Nov. 2009.

[ZGZ03] Y. Zhou, L. Gu, and H. Zhang, "Bayesian tangent shape model: Estimating shape and pose parameters via bayesian inference," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2003, pp. 109–116.

[ZR12] X. Zhu and D. Ramanana, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2879–2886.

[ZW94] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. of the European Conference on Computer Vision*. Springer-Verlag, 1994, pp. 151–158.

[ZZCM08] J. Zhang, S. K. Zhou, D. Comaniciu, and L. McMillan, "Discriminative learning for deformable shape segmentation: A comparative study," in *Proc. of European Conference on Computer Vision*, 2008.

Publications

[AGB+12] N. M. Arar, F. Güney, N. K. Bekmezci, H. Gao, and H. K. Ekenel, "Real-time face swapping in video sequences - Magic mirror," in *Proc. of Int. Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, 2012, pp. 48–53.

[AGEA10] I. Ari, H. Gao, H. K. Ekenel, and L. Akarun, "Facial expression and head gesture recognition using temporal self-similarity and bag of words of facial landmarks," in *Proc. of IEEE 18th Signal Processing and Communications Applications Conference (SIU)*, 2010, pp. 836–839.

[AGEA12a] N. M. Arar, H. Gao, H. K. Ekenel, and L. Akarun, "Face recognition using curvature Gabor features," in *Proc. of Signal Processing and Communications Applications Conference (SIU)*, 2012, pp. 1–4.

[AGEA12b] N. M. Arar, H. Gao, H. K. Ekenel, and L. Akarun, "Selection and combination of local gabor classifiers for robust face verification," in *Proc. of IEEE Fifth Intl. Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2012, pp. 297–302.

[BPT+12] H. Bredin, J. Poignant, M. Tapaswi, G. Fortier, V. B. Le, T. Napoleon, H. Gao, C. Barras, S. Rosset, L. Besacier *et al.*, "Fusion of speech, faces and text for person identification in TV broadcast," in *Proc. of ECCV Workshops on Information Fusion in Computer Vision for Concept Recognition*, 2012, pp. 385–394.

[EFG+07] H. K. Ekenel, M. Fischer, H. Gao, K. Kilgour, J. Marcos, and R. Stiefelhagen, "Universität Karlsruhe (TH) at TRECVID 2007," in *NIST TRECVID Workshop*, Gaithersburg, USA, 2007.

[EGS07] H. K. Ekenel, H. Gao, and R. Stiefelhagen, "3-D face recognition using local appearance-based models," *IEEE Transaction on Information Forensics and Security*, vol. 2, no. 3, pp. 630–636, 2007.

[EGS08] H. K. Ekenel, H. Gao, and R. Stiefelhagen, "Karlsruhe Institute of Technology (KIT) at TRECVID 2008," in *NIST TRECVID Workshop*, Gaithersburg, USA, 2008.

[ESG+07] H. K. Ekenel, J. Stallkamp, H. Gao, M. Fischer, and R. Stiefelhagen, "Face recognition for smart interactions," in *Proc. of IEEE Intl. Conference on Multimedia and Expo (ICME)*, 2007, pp. 1007–1010.

[ESGS09] H. K. Ekenel, A. Schumann, H. Gao, and R. Stiefelhagen, "Karlsruhe Institute of Technology (KIT) at TRECVID 2009," in *NIST TRECVID Workshop*, Gaithersburg, USA, 2009.

[GEFS10] H. Gao, H. K. Ekenel, M. Fischer, and R. Stiefelhagen, "Multi-resolution local appearance-based face verification," in *Proc. of 20th Intl. Conference on Pattern Recognition (ICPR)*, 2010, pp. 1501–1504.

[GEFS11] H. Gao, H. K. Ekenel, M. Fischer, and R. Stiefelhagen, "Boosting pseudo census transform features for face alignment," in *Proc. of British Machine Vision Conference (BMVC)*, Dundee, UK, 2011.

[GES09] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Pose normalization for local appearance-based face recognition," in *Proc. of Intl. Conference on Advances in Biometrics (ICB)*, 2009, pp. 32–41.

[GES10] H. Gao, H. Ekenel, and R. Stiefelhagen, "Robust open-set face recognition for small-scale convenience applications," in *Proc. of 32nd Annual Symposium of the German Association for Pattern Recognition (DAGM)*, 2010, pp. 393–402.

[GES12a] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Face alignment using a ranking model based on regression trees," in *Proc. of British Machine Vision Conference (BMVC)*, Guildford, UK, 2012.

[GES12b] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Identifying important people in broadcast news videos," in *Proc. of IAPR Conference on Machine Vision Applications (MVA)*, 2012.

[GES12c] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "A ranking model for face alignment with pseudo census transform," in *Proc. of 21st Intl. Conference on Pattern Recognition (ICPR)*, 2012, pp. 1116–1119.

[HGGE12] N. Hesse, T. Gehrig, H. Gao, and H. K. Ekenel, "Multi-view facial expression recognition using local appearance features," in *Proc. of 21st International Conference on Pattern Recognition (ICPR)*, 2012, pp. 3533–3536.

[KEG+08] K. Kumatani, H. K. Ekenel, H. Gao, R. Stiefelhagen, and A. Ercil, "Multi-stream gaussian mixture model based facial feature localization," in *Proc. of 16th IEEE Conference on Signal Processing and Communication and Applications (SIU)*, 2008, pp. 1–4.

[QGE12] C. Qu, H. Gao, and H. K. Ekenel, "Rotation update on manifold for robust non-rigid structure from motion," in *Proc. of IEEE Intl. Conference on Image Processing (ICIP)*, 2012.