DISS. ETH NO. 15676

Object Detection using Scale-specific Boosted Parts and a Bayesian Combiner

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Technical Sciences

presented by

HANNES KRUPPA

Dipl. Informatik.-Ing. ETH

born 4^{th} of October, 1974
citizen of
Germany

accepted on the recommendation of

Prof. Dr. Bernt Schiele, examiner
Prof. Dr. Luc Van Gool, co-examiner

2004

Abstract

This thesis develops new algorithms for object detection in still images. As a starting point the particular case of face detection is investigated seeking improvements which can be generalized to other object types. For improving face detection, one possible way is to search for a better representation that is used within the traditional detection paradigms. However, since face detection research has seemingly reached a plateau, a more radical approach is pursued here.

The approach of this thesis is to identify, develop and evaluate complementary cues which can then be combined with traditional face detection methods. Of interest are cues which help improve detection accuracy and have so far been overlooked or not evaluated thoroughly. Inspired by the failure analysis of a state-of-the art face detector we explore two different candidate cues.

The first cue is human skin. The proposed skin detection algorithm combines a comprehensive skin color model with shape models using mutual information matching. An important result is that the combination of color and shape information outperforms purely color-based approaches. Experiments show that the skin cue is in fact complementary to traditional appearance-based face detectors. As a result combinations of the two can significantly improve the detection rate or precision.

The second cue proposed in this thesis is a face's local context which typically contains a person's upper-body silhouette. The most remarkable property of the developed local context detector is its robustness to resolution degradations. The detector is shown to yield additional correct face detections which are complementary to those of traditional face and skin cues.

While the skin cue is rather specific for humans, it turns out that the local context idea allows for a fruitful generalization to other object classes: just as the local context contributes to the detection at small scales, we can form a set of scale-specific object parts to accommodate a set of different scales. Such scale-dedicated parts, in short scaleparts, cover specific parts of the target object at various positions, extents and resolutions. The proposed scaleparts framework builds on boosted classifier cascades for implementing fast and highly discriminant part detectors and uses Bayesian Networks to combine their outputs. Experiments in face and car detection demonstrate that the scaleparts approach is robust with respect to a wide range of resolution situations. A side product is the first local-to-global (entire body) detector for pedestrians.

From an abstract point of view, the scaleparts framework aims to increase robustness to resolution changes in object detection by combining several scale-dedicated part detectors. Its successful application to both faces and cars suggests a potential for detecting additional object types within the same framework, which may serve as future research.

Seite Leer /
Blank leaf

Zusammenfassung

In dieser Dissertation werden neue Algorithmen zur Objektdetektion in Einzelbildern entwickelt. Objektdetektion ist laengst zu einer Standarddisziplin innerhalb des wissenschaftlichen Gebiets des maschinellen Sehens geworden. Dies sowohl wegen der vielen Anwendungsmöglichkeiten als auch wegen der grossen Bedeutung für die Künstliche Intelligenz Forschung.

Die Verfahren werden hierbei ausgehend vom speziellen Teilgebiet der Gesichtsdetektion entwickelt und dann auf andere Objektklassen übertragen (z.B. Personen und Fahrzeuge).

Ein möglicher und oft beschrittener Weg besteht darin, nach Verbesserungen in der Objektrepräsentation zu suchen. Es scheint jedoch, dass die Forschung zur Gesichtsdetekion auf diesem Weg an einem Sattelpunkt angelangt ist, was einen etwas radikaleren Ansatz rechtfertigen würde.

Der prinzipielle Ansatz dieser Dissertation besteht darin, komplementäre Informationsquellen für die Gesichtsdetektion zu identifizieren, zu realisieren und diese auszutesten. Von Interesse sind Informationsquellen, die die Detektionsgüte verbessern und die bisher nicht oder unzureichend evaluiert wurden. Diese können dann mit den herkömmlichen Verfahren zur Gesichtsdetektion kombiniert werden.

Nach einer Fehleranalyse eines der besten zur Zeit verfügbaren Gesichtsdetektoren untersuchen wir zwei Informationsquellen en detail: die eine basiert auf Hautdetektion, die andere berücksichtigt den lokalen Kontext eines Gesichts.

Der in dieser Dissertation entwickelte Algorithmus zur Hautdetektion kombiniert ein umfassendes statistisches Hautfarbmodell mit einem Formmodell mittels "mutual information" Maximierung. Ein wichtiges Ergebnis ist die Überlegenheit des kombinierten Ansatzes im Vergleich mit einem rein farbbasierten Ansatz. Dieses Ergebnis beruht auf umfangreichen Versuchsreihen.

Es wird auch gezeigt, dass die Gesichtsdetektionsrate mittels Hautfarbdetektion stark verbessert werden kann, dank der Invarianz gegenüber Kopfneigung und -drehung. Alternativ kann die Präzision erhöht werden, indem detektierte Gesichter zusätzlich via Hautdetektor geprüft werden.

Der lokale Kontext eines Gesichts meint dessen nähere Umgebung, die insbesondere Teile des Oberkörpers enthält. Ähnlich zu den psychophysiologischen Ergebnissen von Torralba und Sinha zeigt sich, dass der lokale Kontext auch für das maschinelle Sehen effektiv nutzbar ist. Lokaler Kontext ist besonders bei schwierigen Bildverhältnissen von Bedeutung, zum Beispiel bei kleiner Bildauflösung. Aus diesem Grunde widmet unsere Analyse dem Faktor Bildauflösung besondere Aufmerksamkeit.

Während der Hautdetektor spezifisch für Gesichts- oder Meschendetektion nutzbar ist, erlaubt der lokale Kontextdetektor eine ergiebige Verallgemeinerung auf andere

Objektklassen. Analog zum lokalen Kontextkonzept für geringe Bildauflösungen lässt sich ein ganzes Set von skalenspezifischen Objektteilen ("scaleparts") bilden, um verschiedensten Auflösungssituationen gerecht zu werden. Besonders wichtig dabei ist die Diskriminanz der Objektteildetektoren, die durch den AdaBoost Algorithmus erreicht wird.

Die Kombination einzelner Objektteile erfolgt mit Hilfe von Bayes'schen Netzen. Diese erlauben es, die möglichen Abhängigkeiten und Interaktionen zwischen Objektteilen automatisch aus Daten zu erlernen. Experimente im Bereich Gesichtsdetektion und Fahrzeugdetektion belegen die Robustheit des Ansatzes über weite Auflösungsbereiche. Als Nebenprodukt entsteht dabei der wohl erste Personendetektor, der lokale und globale (ganzer Körper) Objektteile integriert.

Die Anwendung dieses "Scalepart"-Ansatzes auf andere Objektklassen ist Gegenstand zukünftiger Forschung.

Acknowledgments

My sincerest thanks go to Prof. Bernt Schiele who has continuously contributed to this thesis with many original ideas, sound and helpful criticism and the necessary impetus to get things going. I would like to thank Prof. Luc Van Gool for agreeing to be the co-examiner of this work.

Special thanks go to my close scientific collaborators Prof. Modesto Castrillon-Santana and Martin Bauer, not to forget all undergraduate students who have contributed to this thesis through their project work: Lukas, Daniel, Thomas, Kliment and Michael.

For being great colleagues I would like thank my fellow "PCCV" PhD students Bastian, Edgar, Florian, Julia, Martin, Nicky and Stavros.

At ETH I feel greatly indebted to the Xibalba computer cluster coordinators Mark (Schmitt), Nurhan, Brian, Bioern and Christoph for providing mission-critical support for many of my computations.

At CMU I would like to thank Prof. Sebastian Thrun for being my host at the Center for Automated Learning and Discovery and Prof. Martial Hebert for introducing me to his seminar.

For important and stimulating discussions I would also like to thank Chuck Rosenberg, Henry Schneiderman, Tom Minka, Shyjan Mahamud, Sanjiv Kumar, and Michael Schroeder. I would like to thank Michael Jones, Henry Schneiderman and Martin Bauer again for providing skin and face data sets.

I would also like to take this opportunity to say hi to research colleagues Dimitris, Mike, Nick, Greg, Dieter, Steffen, Wolfram, Rudi, Dirk, Maren, Cyrill, Antonio, Kevin, and to Michel Vidal-Naquet and Krystian Mikolajczyk who I hope will find this thesis interesting to read.

Seite Leer /
Blank leaf

Contents

1

Introduction and Motivation

1.1 Detecting Objects in Still Images

This thesis presents new object detection algorithms and their evaluation in face detection and car detection tasks. Let us first consider face detection to illustrate the more general problem of object detection. [Yang *et al.*] define the face *detection* task as follows: "Given an arbitrary still image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return their locations and extents". This is in contrast to the problem of *recognition*, where the goal is to identify specific instances within a certain object class: A face detection system knows how to differentiate faces *from everything else*, while a face recognition system knows the difference between A's face and B's face.

Research in computational object detection has a history of more than 30 years – both as a topic of basic research as well as for specific applications. For example, it is widely considered a critical component for enabling an "intelligent" machine understand what it sees and is therefore a traditional topic of basic artificial intelligence research. Another example is psychophysics where computational object detection approaches can be used to assess hypotheses about the functioning of the human visual system. But equally important, object detection by computers has many commercial, military and civil applications. Here, some of the most important application domains are:

Human Computer Interaction: Face detectors are extremely popular components in conversational interfaces, for example in robots. Figure 1.1 suggests future applications in commodity household-robots, which roboticists have been dreaming about for more than 60 years [Corn 1996]. In fact face detectors have been used in museum-tourguide robots to enable meaningful interaction with people [Thrun *et al.* 2000] as well as in robots that assist the elderly [Montemerlo *et al.* 2002]. Similar conversational interfaces appear in Smart Kiosks, a freestanding computer system capable of social interaction

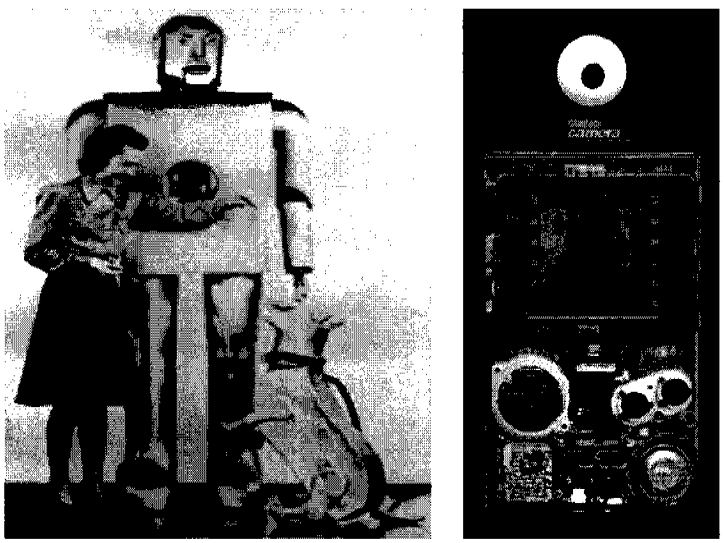

***Figure 1.1**: Left: object detection and in particular face detection capabilities are critical components for human computer interaction, for example, with a robot. Right: the Nintendo GameBoy Camera allows the player's face to be inserted into a simple game – which involves face detection.*

with multiple users [Rehg *et al.* 1999]. Another example is video conferencing software, where exact knowledge of face positions and sizes helps to optimize bandwidth allocation. Finally, recent computer games make use of face detectors, too (which is illustrated on the right side of figure 1.1).

Intelligent Driver Assistance: Several computer systems that assist in driving require object detection capabilities [Ninomiya *et al.* 1995, Remagnino *et al.* 1997, Franke *et al.* 1998]. In this respect the urban traffic environment is particularly challenging: The leading vehicle must be detected and its distance, speed and acceleration must be estimated in longitudinal and lateral direction. The course of the lane must be extracted even if it is not given by well painted markings. Small traffic signs and traffic lights have to be detected and recognized in a highly cluttered environment. Different additional traffic participants like bicyclists or pedestrians must be detected and classified. Finally, stationary obstacles that limit the available free space e.g. parking cars must be detected.

Content-based Image Enhancement: Large foto printers of imaging companies like Kodak, Fuji and Agfa analyze digitized fotos and do certain automatic corrections – part of the traditional job of a human foto laboratory worker. Better understanding of the image's content allow for better corrections resulting in better pictures. As a very specific example, automatically detected faces can be used as a cue for detecting and correcting global or local color

casts. This simply builds on the fact that facial skin color is confined to a specific chromatographic subspace of the spectrum. Detected faces can also be used for automatic image derotation. Likewise, automatic object detection is soon to be integrated in digital cameras. As camera technology changes from film to digital capture, cameras are becoming part optics and part computer. Such a camera can automatically focus, color balance, and zoom on a specified object of interest, in particular a human face.

Image Retrieval: The availability of large digital image collections has grown dramatically in recent years. The Associated Press collects and archives an estimated 1,000 photographs a day (www.ap.org). The number of images on the World Wide Web is at least in the hundreds of millions. However, the usability of these collections is limited by lack of effective retrieval methods. Currently, to find a specific image in such a collection, we have to search using text-based captions and low-level image features such as color and texture. Automatic object detection and recognition could be used to extract more information from these images and help automatically label and categorize them. By making these databases easier to search, they will become accessible to wider groups of users, such as television broadcasters, law enforcement agencies, medical practitioners, graphic and multimedia designers, book and magazine publishers, journalists, historians, artists, and hobbyists.

Surveillance: Mounting video cameras is cheap, but finding available human resources to observe the output is expensive. Here object detection technology plays an important role for constructing a software-based artificial observer that human security officers can use as a tool. In addition to the obvious security applications, video surveillance technology has been proposed to measure traffic flow, detect accidents on highways, monitor pedestrian congestion in public spaces, compile consumer demographics in shopping malls and amusement parks, log routine maintenance tasks at nuclear facilities, and count endangered animal species. Object detection is a key component for such applications, in particular for initializing tracking and to recover from tracking failure.

Target Acquisition and Tracking: The numerous military applications of computational object detection are often integrated in visual surveillance tasks. This includes patrolling national borders, measuring the flow of refugees in troubled areas, monitoring peace treaties, and providing secure perimeters around bases and embassies, as in the current DARPA Human ID project (as a sidenote to stress its importance: this project has been allocated USD 50 million).

1.2 Challenges in Object Detection

The detection of real-world objects of interest, such as faces, people, and cars, poses many challenging problems. An ideal detector must be able to differentiate any instantiation of the target class from everything else in the world. More specifically, it has to accommodate all possible variations of the target's appearance, e.g. with respect to color, texture, pose, lighting, and size and at the same time be highly specific to avoid confusion with complex background clutter. A more general difficulty is the geometric ambiguity which arises from projecting three dimensions of the world onto two dimensions in the image.

Hence – despite its many applications and the large interest of many researchers – general object detection by computer is still far from being solved. This is equally true for the specific case of face detection which is probably the best-studied object class within object detection research. In the following case study we illustrate some typical failures of a state-of-the-art face detector. This case study also serves as a motivation for the specific approach adopted by this thesis.

1.3 Case Study: Real-world Performance of the Schneiderman-Kanade Detector

According to two recent survey articles on face detection research ([Yang *et al.* , Hjelmas and Low 2001]) the most successful face detectors are all appearance-based (i.e. they start from a collection of 2D-views). There is a set of canonical challenges associated with appearance-based detection. Following is a description of typical challenges with specific examples for faces – but all these issues are relevant for detecting other object classes as well.

Pose: Face images vary due to the relative camera-face pose (in-plane and out-of-plane rotations, frontal, 45 degree, profile, upside down). As a result some facial features such as an eye or the nose may become partially or wholly occluded.

Individual Features and Expression: Facial features such as beards, mustaches, and glasses may or may not be present and there is a great deal of variability among these components including shape, color, and size. Likewise the appearance of faces are directly affected by a person's facial expression (laughing, smiling, crying etc.).

Occlusion: Faces may be partially occluded by other "objects". For example in a group picture some faces or bodies may partially occlude other faces. Also,

in pictures people regularly occlude parts of their own face with their hands. This is often a result of a particular gesture or activity such as leaning on a table and supporting the chin with one hand.

Imaging conditions: When the image is formed, factors such as lighting (spectra, source distribution and intensity) and camera characteristics (sensor response, lenses, resolution) affect the appearance of a face.

According to the above mentioned survey articles the face detector due to Schneiderman and Kanade [Schneiderman and Kanade 2000] achieves the highest accuracy levels on a standardized test set of 125 grayscale images. A recurring criticism of this finding is that this test database has been partially created by the same research group which can result in an unfair bias. However, unlike other competing approaches (e.g. the more recent work of [Viola and Jones]) Schneiderman also provides a web-interface to his implementation of the detector where one can upload one's own test images[1].

We have used this facility to upload a set of 250 vacation snapshots to analyze errors and their causes. Errors can be either missed detections or false alarms. Since these are real-world photographs we have no access to the underlying imaging parameters. This makes inferring the exact cause of an error difficult. However, we can still try to make an educated guess about what caused a specific error. Since in practice several error causes coincide (e.g. an occlusion combined with a rare pose) we try to identify the most important cause.

We have tried to manually group the encountered error cases which resulted in seven categories: "face-like texture", "group pictures", "pose/individual", "too small", "image orientation", "occlusion" and "other" each of which is illustrated by example images in figure 1.3. Following is a detailed explanation of these categories and the associated figures 1.2 and 1.3.

Face-like Texture: These are false alarms that frequently occur in complex, highly structured images (25% of all errors). This is often the case in outdoor images, for example on trees and bushes. We can further subdivide false alarms according to their "faceness". This ranges from random patterns that humans also might classify as faces and other patterns which humans would definitely not classify as faces but which are caused by an intrinsic representational mistake of the underlying detection algorithm (often these mistakes come from a simplifying assumption in the model). One other type of false alarms are inaccurate detections. These detections cover part of a face but their position and scale is so inaccurate that they are not counted as successful detections.

[1] http://vasc.ri.cmu.edu/cgi-bin/demos/findface.cgi

Group Pictures: This is a recurring theme in photography and it usually implies that several people look directly into the camera). Given a fixed image resolution and several faces this means that individual faces are necessarily represented with less detail than say in a portrait situation. Unfortunately, face detection test databases such as the CMU+MIT data set ([Schneiderman 2000]) avoid this problem: they represent group pictures at high resolution and portraits as smaller resolution so that (a) the minimum resolution requirements for the detector are always met and (b) there are no "unnecessary" distracting background patterns caused by too much image detail. Contrastingly, we have assumed the same pixel resolution in all images which is more realistic because it corresponds to the original camera output. However, as a result faces are often too small to detect (17% of all errors).

Too small: Errors in this category are also caused by too small faces although the actual scene is not a group picture (16% of all errors). Instead these are often scenes with greater emphasis on the background (e.g. buildings or landscapes) and with only few faces in it. For example, the Schneiderman-Kanade detector requires faces to be at least 24×28 pixels in size. Processing at low resolutions is generally desirable because high resolution images cause at least two problems: the search becomes slower and the number of false positives is likely to increase. This is simply because there are more possibilities to make an error in an exhaustive search. Note that categories "face-like texture", "group pictures" and "too small" already make up for 57% percent, i.e. more than half, of all errors.

Pose/individual: In snapshot photography people often do not look directly into the camera which can result in missed detections (16% of all errors). Their actual pose can vary greatly. Also people's faces vary individually with respect to skin complexion, beards or glasses for example which poses great challenges to the detector.

Image Orientation: It is natural for photographers to turn the camera once in a while to make an upright image. Upright images of people will show faces 90 degrees in-plane rotated. Since the Schneiderman-Kanade detector deals with out-of-plane rotation but not with in-plane rotation this results in missed detections (15% of all errors). Usuallly upright images have to be derotated manually by the photographer. Unlike the standardized test sets such as the MIT+CMU data set we have not derotated images before testing which we think is more realistic.

Occlusion: Total or partial occlusion of faces is likely to result in missed detections (7% of all errors). Clothes are a common source for the partial occlusion of faces. Also, people often occlude parts of their own face with their hands, for

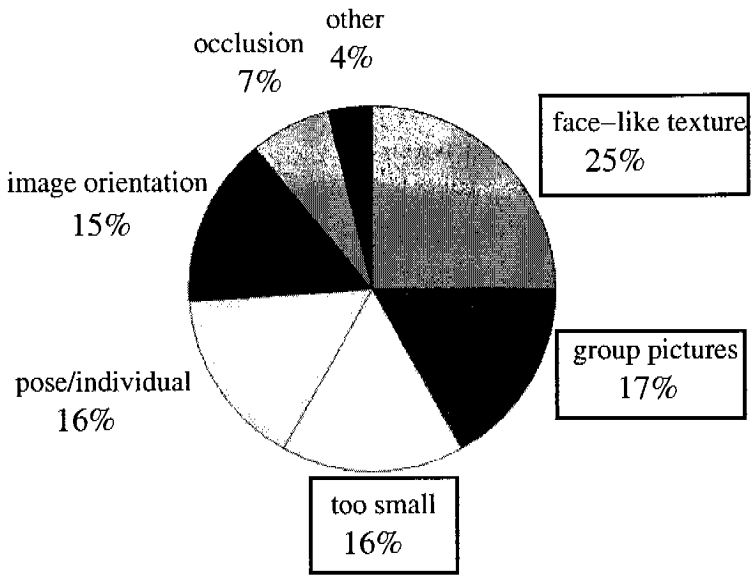

***Figure 1.2**: Breakup chart of encountered error types in a representative set of 250 vacation snapshots using the Schneiderman-Kanade face detector. The top three error sources "face-like texture", "group pictures" and "too small" account for more than 50% of all errors. The underlying error cause for the latter two is lack of image resolution. As a result faces are smaller than the detector's search window size and are therefore not found by the detector.*

example when leaning on one arm at a table with the hand supporting the chin.

Other: This final category subsumes various error sources (4% of all errors) such as illumination problems (where illumination is clearly the predominant problem) or ambiguous data labeling. Figure 1.3 shows the example of a face drawing in the lower right corner of the figure. The drawing is not a real human face – i.e. this is a detection error – but the detector has correctly identified the facial pattern.

Categories "face-like texture", "group pictures" and "too small" make up for 57% percent, i.e. more than half, of all encountered errors. The latter two have the same cause, that is, insufficient image resolution. The novel face detection cues which are going to be developed in this thesis are directly inspired by face-like texture and low resolution errors and the promise is that this will remedy a large fraction of encountered errors. Ultimately, we not only need to know what types of errors exist but also how often each of them occurs. This allows us to set priorities and to define our approach. A breakup quantifying different error sources is given in figure 1.2. Keeping in mind the above mentioned difficulties with this type of analysis the listed percentages should be seen as a rough approximation.

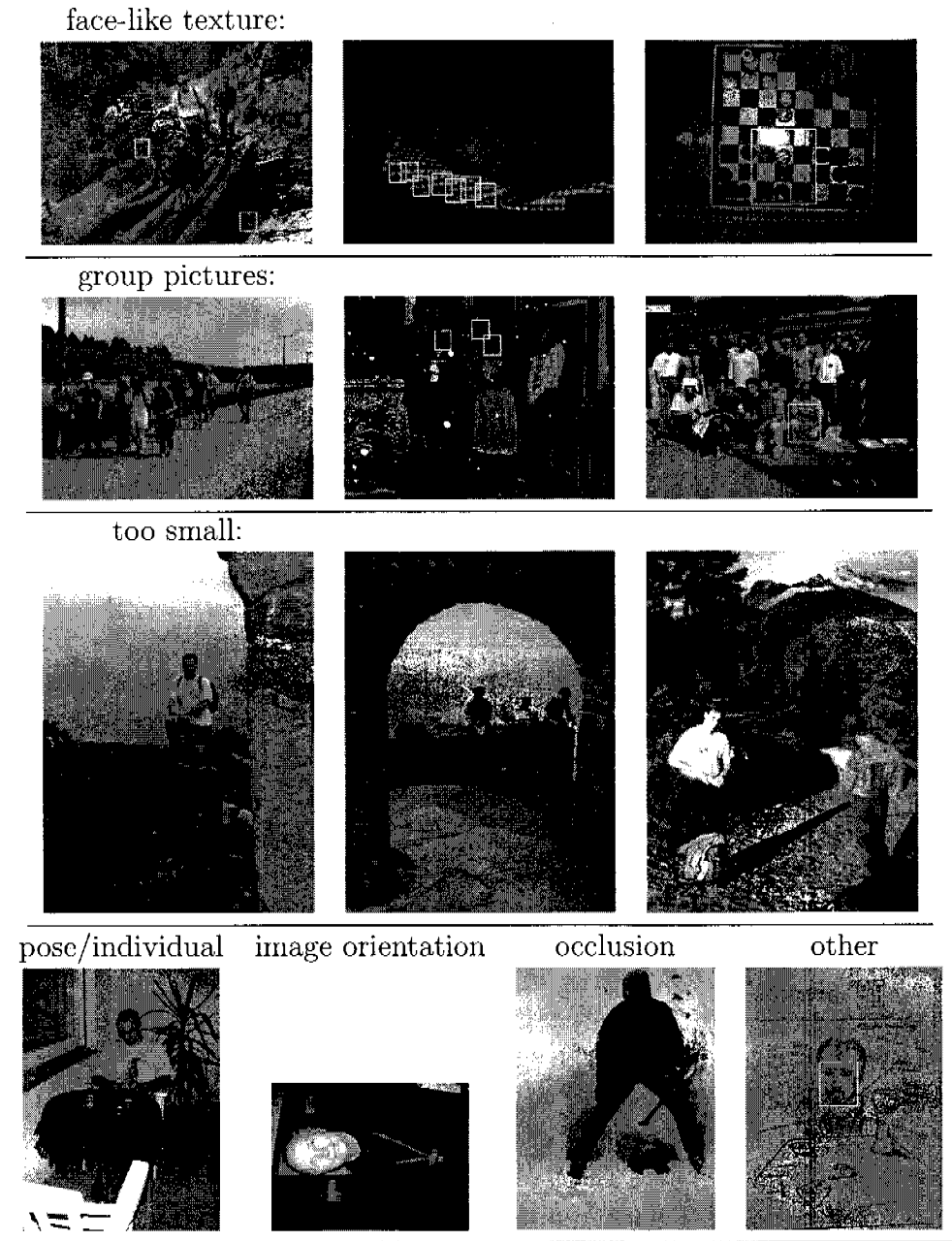

Figure 1.3: *"Face-like texture" (top row) causes false detections which vary with respect to their actual "faceness": they can be face-like such as the foilage in the left picture or they can be less face-like and must therefore be seen as artefacts of a representational mistake in the detector (such as on the checker board in the right image). Another common error source is the lack of image detail which results in missed detections (categories "group pictures" and "too small"). This often occurs in group scenes (second row) or in scenes with few people and greater emphasis on the background (e.g. buildings or landscapes as in row three). Details for the less common error categories "pose/individual", "image orientation", "occlusion" and "other" can be found in section 1.3.*

Two cues are going to be investigated: The first cue employs a combination of color and shape information to help discriminate against face-like background texture. Color is not used by the Schneiderman-Kanade detector. Second, for dealing with the problem of low image resolution we will apply contextual cues which indirectly hint at the presence or absence of the target object. Since low resolution is the common error cause for categories "group pictures" and "too small" the promise is that context will be beneficial for both of these cases.

Though not further investigated in this thesis the following list briefly mentiones alternative cues or approaches inspired from the remaining error categories "pose/ individual", "image orientation", "occlusion" and "other".

Errors caused by *pose/individual* face variation can be attributed to the appearance based approach. As a result, rarely occuring poses or individual expressions are often not adequately represented in the training data. This can be remedied by resorting to a model with a stronger geometric flavor where pose changes can be adressed by a geometric transformation in the model.

In-plane rotated faces in rotated images (category *image rotation*) can be detected by Rowley's face detector [Rowley *et al.* 1998]. However, Rowley's detector does not deal with out-of-plane rotations. Within digital photography research automatic image derotation has turned into a discipline of its own. Current approaches use "meta-information" provided by certain camera models (e.g. coming from a tilt sensor) or they try to infer the true image orientation directly from the image based on – for example – sky detection or a holistic low-level feature analysis [Vailaya *et al.* 1999].

The common way to deal with partial *occlusion* is to apply a parts-based approach – which the Schneiderman-Kanade detector already does. As a result only 7% of all errors have been attributed to occlusion problems.

Errors of type *other* are diverse but as an example we can consider the problem of differentiating a drawing from an actual face. This is a subtle and difficult problem but one potential cue here might be texture.

1.4 Thesis Outline

In this thesis we want to improve the state of the art in object detection. To this end we specifically investigate the detection of faces as a test case and guiding example. The promise is that further improvements in face detection are useful for the detection of other object classes as well – which in this thesis is tested specifically for car detection.

Improving face detection is challenging: the field has reached a performance plateau in terms of accuracy since the work of [Schneiderman and Kanade 2000] despite the large interest by many computer vision researchers. Significant improvements have been made in speed and in making the learning more automatic [Viola and Jones], but not in accuracy. This viewpoint is supported by recent comprehensive surveys on the topic [Yang *et al.* , Hjelmas and Low 2001].

Given this situation, we expect that more radical approaches have to be explored. Thus, rather than developing a new representation embedded in the traditional detection paradigm our approach is to identify and apply *complementary cues* that can be *combined* with traditional face detectors to leverage their accuracy. We seek complementary cues that help in the detection task and that so far have been overlooked or not evaluated in combinations of several cues. Given the above case study on error types we are especially interested in a cue that helps discriminate face-like texture from faces as well as a cue that allows to detect low resolution faces. The first cue we propose is based on skin patch detection, the second on using a face's local context. The local context concept will then be generalized for use in parts-based car detection.

In the following the content of each chapter is summarized.

Chapter 2 reviews related work with a focus on publications that had strong influence on this thesis.

Chapter 3 develops a skin patch detector that can be used as a cue for detecting human body parts – and human faces in particular. It is shown that a combination of color and simple shape models outperforms a state-of-the art skin detector which is purely based on color. The proposed algorithm uses the concept of mutual information for measuring the "agreement" between color and shape distributions. The approach is thoroughly evaluated on large image sets. A particular focus is the use of skin patch detection as an auxiliary cue for face detection and combinations of both cues are analyzed here as well.

Chapter 4 introduces a completely orthogonal cue which is purely based on intensities and does not require color information. Motivated by psychophysical findings the key idea is to indirectly detect faces by taking into account their *local context*. By "context" we mean indirect evidence, that is, features that do not lie directly on the target object but for example in its close vicinity and thereby indirectly hint at the target object's presence. The local context of a face, for example, includes a person's full head and upper body silhouette. "Contextual" detection contrasts the predominant "object-centered" paradigm, which only considers features that occur directly on the target object. The approach is evaluated on several large image sets using the Schneiderman-Kanade as well as the Viola-Jones detector architectures as a basis.

Chapter 5 generalizes the local context idea by introducing "scaleparts". The key idea behind scaleparts is to seek for highly discriminant object parts which accommodate different levels of image detail, or scales. Thus scaleparts serve as a basis for an object detector that can successfully deal with a wide range of image resolutions. The approach is implemented using AdaBoost learning.

Chapter 6 develops a scaleparts-based object detector by combining scaleparts with the help of a Bayesian Network. The Bayesian Network formalism allows for encoding and learning interactions between different part detectors as well as for inferring a combined answer from all (potentially contradicting) evidence. The result is a detector that successfully deals with a wide range of image resolutions. The detector's ability is demonstrated on a difficult face detection task.

Chapter 7 applies the scalepart concept to car detection. This demonstrates that the approach works with other object categories as well.

Chapter 8 draws conclusions and delineates future research.

Seite Leer /
Blank leaf

2

The State Of the Art

This chapter summarizes briefly some references which have been sources of inspiration for this thesis. Here, we do not aim to give a comprehensive overview of the numerous object detection algorithms. Instead, we are particularly interested in state-of-the-art face detection methods as a starting point for our own developments. Therefore, section 2.1 reviews three state-of-the-art face detectors which are among the most successful approaches according to the comprehensive survey articles of [Yang *et al.*] as well as [Hjelmas and Low 2001]). Since our work also touches aspects of pedestrian detection (in chapters 4 and 5) and is also applied to car detection (chapter 7) the section includes these topics as well.

As mentioned in the introduction, our analysis of face detection errors suggests a potential benefit in using color-based and contextual information. By "context" we mean indirect evidence, that is, features that do not lie directly on the target object but for example in its close vicinity and thereby indirectly hint at its presence. This is in contrast to the predominant "object-centered" paradigm in detection, which only considers features that occur on the object. Section 2.2 therefore reviews work on color-based detection of human skin as well as context-supported detection.

Ultimately, we are interested in combining all cues into one detector. Among the many established cue fusion techniques we have been particularly inspired by the elegance and versatility of Bayesian Networks. Section 2.2 therefore rewiews Bayesian Networks as a cue fusion technique which is going to be adopted in chapters 6 and 7.

2.1 Detection of Faces, Pedestrians and Cars

It is important to contrast detection with the problem of recognition, where the goal is to identify specific instances of a class. A face detection system knows how to differentiate faces from everything else, while a face recognition system (see for example [Zhao *et al.* 2000]) knows the difference between person A's face and person

B's face. A typical detection-style algorithm scans the input image at all positions and scales by classifying each possible subwindow independently. It then reports the number, positions and sizes of found targets.

The next three sections describe the face detectors of Schneiderman-Kanade (section 2.1.1), Rowley-Kanade (section 2.1.2) and Viola-Jones (section 2.1.3). While the Schneiderman-Kanade detector is known for its high accuracy, and the Rowley-Kanade method for its speed the more recent approach of Viola-Jones performs comparably under both aspects [1] – but builds on quite different algorithms. The Schneiderman-Kanade approach is, however, the only scheme among these three which has been shown to work for cars, too.

Section 2.1.4 describes Papageorgiou's pedestrian detection work along with follow-up work of other authors. Certain aspects of parts-based pedestrian detection re-occur in chapters 4 and 5 which develop contextual face detection cues. These contextual cues are then generalized to a parts-based object detection approach in chapters 5 and 6 where individual parts cover different extents at various resolutions. This idea is in a way similar to Vidal-Naquet's and Ullman's notion of intermediate complexity fragments which is summarized in section 2.1.5. Finally, section 2.1.6 describes the car detection work of Agarwal and Roth whose car test data base is going to be used in chapter 7.

2.1.1 Schneiderman-Kanade

Schneiderman and Kanade [Schneiderman and Kanade 2000] propose a parts-based object detector. They not only report impressive detection results for face and car detection but also offer a web-based demo (images are processed over night). Their system is one of the few appearance-based detectors that successfully deals with out-of-plane rotation – both for face detection and for car detection.

At the core of the detector is the evaluation of a likelihood ratio term:

$$\prod_{k=1}^{n} \prod_{x,y \in region} \frac{p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid object\right)}{p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid nonobject\right)} > \theta \qquad (2.1)$$

Here, the two likelihoods for the positive and the negative class $p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid object\right)$ and $p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid nonobject\right)$ are decomposed over n different feature types and the locations x, y within the detection window. This decomposition is based on the assumption of independence, i.e. the Naive Bayes assumption. From the above equation follows that the likelihood functions themselves depend on $i(x)$ and $j(y)$ which are coarse quantizations

[1] Competitive performance on a benchmark database was shown for frontal face detection only. This excludes the detection of rotated faces/profiles.

of feature positions within the detection window. This spatial dependency allows to capture the global geometric layout: within the detection windows certain features might be likely to occur at one place but are unlikely to occur at another place. There are $n = 17$ predefined feature types involved. An input instance is first decomposed by a two-dimensional 3-level wavelet transform using a biorthogonal filter. Features are then extracted by examining local arrangements of a small number of wavelet coefficients within one subband or from different subbands. This allows to capture local dependencies across space (arrangements with different extents), frequency (different wavelet levels) and orientation (subbands with different filtering directions).

Instead of estimating the above individual likelihoods the decision boundary is directly approximated using bootstrapping [Sung and Poggio 1998] and a variant of AdaBoost learning [Schneiderman 2000].

Despite its high classification performance there is not much follow-up work by other authors. This is probably because the original approach has so many components with many hand-tuned parameters and relatively few automatic learning steps. Also, the authors invest a lot of manual work into tightly controlling the variation in the positive set (manual lighting correction and landmark-based geometric alignment). Only recently Schneiderman developed a more learning-oriented formulation [Schneiderman 2003].

2.1.2 Rowley and Kanade

Before the seminal work of Viola and Jones, one of the fastest appearance-based detectors was Rowley and Kanade's Neural Network based face detector [Rowley *et al.* 1998] which also deals with in-plane rotations.

Rowley's approach is designed to detect upright faces at any degree of rotation in the image plane. It is not optimized to deal with out of plane rotations though. The system consists of a cascade of multiple neural networks. Its first stage tries to determine the orientation of the current input window and then forwards this input to a specialized face classifier. The classifier is a multilayer perceptron that expects a 20x20 region as input. Rowley performed experiments with both single neural networks and modular systems consisting of several neural networks. In all the algorithms, each 20x20 input region is pre-processed to correct for differences in lighting conditions: a linear function of intensity is fitted to the region and subtracted out, then histogram normalization is performed. The neural network topology used in all experiments consists of one layer of hidden units where each hidden unit has a receptive field of either 5x5, 10x10, or 20x5 inputs. This particular network topology emphasizes local features over global ones, since the hidden units have only local support.

Rowley's and Schneiderman's detectors are compared on the MIT+CMU test set in [Schneiderman 2000]. On this test data Rowley's detector has a detection rate of 86% with 31 false alarms while Schneiderman's detector has a detection rate of 94% with 65 false alarms. Unfortunately, this comparison is not entirely conclusive because of the different false alarm levels, however, a more comprehensive analysis (ROC curves) has not been published.

2.1.3 Viola-Jones and related

Recently Viola and Jones introduced a method for frontal face detection [Viola and Jones]. They report a detection accuracy that is comparable to the Schneiderman-Kanade detector on a standardized test set – but with a much faster algorithm running at 15Hz. For this speed previous approaches had to rely on color or motion information and were thus restricted to certain types of applications.

The Viola-Jones classifier combines the following three strategies for speed:

fast to compute features

classification based on simple linear thresholds, reminiscent of decision stumps in decision trees

a cascaded detector whos cascade structure is learned during training

All features are computed as differences of pixel sums (i.e. integrals) over rectangular regions. These features are simple in the sense that they only capture horizontal and vertical bars and edges. On the other hand they can be computed at constant time independent of size and position with the help of summed area tables ("integral images"). As a result computing these features at all scales and positions is faster than computing an image pyramid which is required by Schneiderman-Kanade and Rowley-Kanade.

AdaBoost [Schapire and Singer 1999] is used both to select features and to train the actual classifier. AdaBoost is a greedy iterative fitting procedure that in each round selects the feature which best classifies the training data. It then reweighs the training set assigning high weights to misclassified instances. AdaBoost provably drives the training error to zero exponentially in the number of rounds and at the same time achieves large margins rapidly [Schapire *et al.* 1998]. However, with each round it also slows down the classifier by increasing its complexity.

Hence, instead of training one monolithic slow-to-evaluate classifier Viola et al. propose an algorithm for learning a classifier cascade [Fleuret and Geman 2001] where each cascade stage is trained using AdaBoost. A cascade allows to shortcut the

computation for almost all negative test instances and only compute all features for the most promising test candidates. An exhaustive search over a single image typically means examining several tens of thousands of candidate windows where only very few ones correspond to targets ("rare event detection").

The Viola-Jones detector has spawned a flurry of follow-up papers from other authors many of which use other feature types [Garg *et al.* 2002] or extend the original feature set [Lienhart *et al.* 2002b]. Others employ variants of the boosting procedure, like FloatBoost [Z.Q. Zhang and Zhang 2002], LogitBoost or Gentle AdaBoost to mitigate its greediness. Applications include profile detection of faces, lip tracking and banner (commercials) detection [Lienhart *et al.* 2002b].

2.1.4 Papageorgiou et al and related

Papageorgiou et al [Papageorgiou *et al.* 1998, Papageorgiou and Poggio 2000] develop a classifier based on a Support Vector Machine (SVM) and Haar wavelet features and apply it to face, car and pedestrian detection. The features – obtained from a stationary discrete wavelet transform (DWT) – form an overcomplete dictionary of localized intensity differences at several scales. Coefficients from two hand chosen scales are then used for SVM training.

The authors also develop a faster version by making the following simplifications

> instead of three wavelet transforms for each color channel they compute one transform on intensity values
>
> they hand select 29 of the 1326 initial features by considering each coefficient's magnitude and its position on the person.
>
> they compute a reduced set of synthetic support vectors
>
> they apply a fast global prefilter ("focus of attention") to identify promising image regions and to reduce the number of classifier applications

In a similar vein but more learning-oriented [Depoortere *et al.* 2002] introduce further speed-ups to the original scheme while maintaining high accuracy levels.

A drawback of Papageorgiou's original scheme is that because of its global approach it cannot handle partial occlusion. [Mohan *et al.* 2001] therefore develop a parts-based version of Papageorgiou's pedestrian detector. Instead of using a single classifier for the whole body, Mohan uses separate SVMs to detect hand-specified body parts (left arm, right arm, upper body, and legs) and then combine the results by training a SVM on top of the part classifiers (stacking). The resulting system outperforms Papageorgiou's original scheme and achieves a 92% detection rate at

roughly 1 false positive per image. The authors attribute this improvement to the fact that the component-based approach uses more prior knowledge, encoded as explicit knowledge about the geometric properties of the human body and explicitly allowing for variations in the human form.

2.1.5 Vidal-Naquet and Ullman

Ullman and Vidal-Naquet develop a part-learning approach based on Mutual Information [Ullman *et al.* 2002] for parts-based object detection. They call their parts "fragments". Their approach optimizes the position, size and resolution of object fragments by maximizing their mutual information with the target class. Mutual Information typically peaked for intermediate complexity fragments: fragments of intermediate size at high resolution or fragments of larger size at intermediate resolution. The superiority of intermediate complexity fragments can be explained as the interplay of two factors: specificity and relative frequency. For example, a large face fragment can provide reliable indication of the presence of a face in an image, although the likelihood of encountering such a fragment in a novel face image is low. Consequently, the information carried by such a part with respect to the class is limited. A smaller part has a higher likelihood of appearing in different face images, but the likelihood of its presence in non-face images is also higher.

Although they use different terminology the approach is essentially discriminative. A major drawback is its relying on slow normalized cross-correlations coupled with the combinatorial explosion of possible fragment candidates. To make this computationally feasible they resort to extremely small training patch resolutions. As a result extracted fragments (parts) are sometimes as small as 4×4 pixels [Vidal-Naquet and Ullman 2003].

2.1.6 Agarwal-Roth and related

[Agarwal and Roth 2002] present another parts-based object detection approach which builds on an interest point detector. First, small patches around interest points are grouped using an agglomerative clustering scheme. Images are then represented by using parts from these clusters, along with their spatial relations. The information which parts and which binary spatial relations were observed in the images are combined as sparsely coded entries in a very high-dimensional feature vector. Based on this representation, a SNoW classifier is trained to discriminate image windows containing the object from negative examples. The authors apply their method to the problem of detecting side views of cars and make their test database available to other authors.

Garg et al [Garg *et al.* 2002] present an interesting extension which adds a global, holistic detector based on ICA features to the local part-based approach described above. Both local and global measurements are fused by supervised learning of a second order polynomial classifier which uses the outputs of ICA-based classification and part-based classification as training inputs.

2.2 Color and Context as Cues in Object Detection

This section is concerned with specific aspects of color and context as cues in object detection. With respect to color we focus on face detection which boils down to the detection of human skin using skin color models. Contrastingly, the work on contextual cues summarized here has been shown to apply to various object classes. Altogether, the section briefly describes three classes of color-based skin detection algorithms as well as two contextual detection approaches. Section 2.2.1 describes comprehensive skin color models which try to capture the true variation of skin appearance and which are therefore widely applicable. Such a model is going to be used in the next chapter (chapter 3). Then, section 2.2.2 describes specialized skin color models which are effective as part of a larger system (such as a face tracker) – but which are not as versatile as comprehensive skin models. For completeness, physical models derived from spectrographic analysis are briefly summarized in section 2.2.3.

Compared to skin color modelling the literature on context supported detection is sparse. Section 2.2.4 summarizes Strat's seminal work [Strat and Fischler 1991] which is based on an object-based representation of context. This notion of context will be adopted by chapter 4. The formal derivation and the term *local context* introduced in chapter 4 is, however, more closely related to [Torralba 2001]. Torralba uses a scene-centered representation of context which is explained in the final section 2.2.5.

2.2.1 Comprehensive Skin Color Models

The most comprehensive skin color model we are aware of is due to Jones and Rehg [Jones and Rehg 1999]. In their case, a skin color model is learned from a huge collection of manually segmented web images. A Bayesian classifier for skin color is then constructed which also incorporates a model of the non-skin class. They use 4483 training photos which form the non-skin color model and 2339 training photos which form the skin color model.

Fleck and Forsyth [Fleck *et al.* 1996, Forsyth and Fleck 1997] as well as Wang et al. [Wang *et al.* 1997] propose systems for filtering adult images by finding naked

people. In the approach by Fleck et al a combination of low-level image filters is used to extract skin regions. This combination builds on skin color and texture features. By geometrical analysis, images with at least 30% skin are then searched for body parts.

Interestingly, Jones and Rehg's color model by itself leads to comparable performance to the system of Forsyth et al which uses the combination of features. This underlines the importance of appropriate color models for skin detection.

2.2.2 Specialized Skin Color Models

Many systems for tracking or detecting people in user-interface or video-conferencing applications have employed skin color models. Histogram models are employed by [Schiele and Waibel 1995] and [Kjeldsen and Kender 1996]. [Yang *et al.* 1998] model skin color as a single Gaussian, while [Jebara and Pentland] employ a mixture density. [Terrillon and Akamatsu 2000] examine Gaussian skin models and compare nine different color (i.e. feature) spaces. Mottaleb et al. add facial feature detection and manually defined geometric constraints on top of this [Rein-Lien Hsu 2002].

In all of these systems, the color model is trained on a relatively small number of example images taken under a set of illumination conditions representative for the specific task at hand. Most, with the exception of [Kjeldsen and Kender 1996, Jebara and Pentland], do not use non-skin models. These color models are effective in the context of a larger system, but they do not address the question of building a global skin model which can be applied to a large set of images. Instead they are specialized to a certain application.

2.2.3 Physical Skin Color Models

[Angelopoulou 2001] and, independently, [Stoerring *et al.* 1999] use spectrographic analysis to derive a physical reflectance model of skin. The characteristic human skin tones are due to light reflectance in a thin surface layer, the epidermis, and a thicker layer underneath, the dermis. The light absorption in the dermis is mainly due to the ingredients in the blood such as haemoglobin, bilirubin and beta-carotene which are basically the same for all skin types. However, skin color is mainly determined by the epidermis transmittance which depends on the *dopa-melanin* concentration and hence varies among human races. Apart from these characteristics the physical skin reflectance model incorporates both camera and light source parameters. In uncontrolled scenes, however, these parameters are not known.

2.2.4 Object-based Representation of Context

[Strat and Fischler 1991] built the CONDOR system which uses contextual information for object recognition. The system is based on a large number of hand-written rules that constitute the knowledge database of the system. A collection of rules ("context sets") defines the conditions under which it is appropriate to use an operator to identify a candidate region or object. The candidates are then the inputs for other rules that will activate other vision routines. The ideal output of the system is a labeled 3D model of the scene. From a more abstract viewpoint context is defined here as a collection of objects or regions (already recognized or at least given candidate object labels). Predefined rules about the world in which the system is expected to operate produce reliable inferences using the candidates as input. As we will see in chapter 4 the contextual notion in this thesis is also object-based: we seek to discern specific object instances within the scene to indirectly infer information about the target object.

2.2.5 Scene-centered Representation of Context

In the real world, there exists a strong relationship between the environment and the objects that can be found within it. Experiments in scene perception and visual search [Biederman *et al.* 1982] have shown that the human visual system makes extensive use of these relationships for facilitating object detection and recognition suggesting that the visual system first processes context information in order to index object properties. In particular, scene recognition experiments suggest that information about scene identity may be available before performing a more detailed analysis of the individual objects [Biederman 1987].

[Torralba 2001] applies this finding to computational vision. Accordingly, the context is considered a single entity ("the scene") that is to be recognized by means of a scene-centered representation bypassing the identification of the constituent objects. They take a probabilistic approach and model the joint probability of scene features and object-centered features (i.e. features on the target object).

Starting from the traditional object-centered viewpoint this can be formalized using Bayes' Rule as follows:

$$P(O \mid \mathbf{v}) \simeq P(O \mid \mathbf{v}_L) = \frac{P(\mathbf{v}_L \mid O)}{P(\mathbf{v}_L)} P(O) \tag{2.2}$$

where the image measurements $\mathbf{v}$ are local measurements on the target object, that is $\mathbf{v} = \mathbf{v_L}$. The object-centered object likelihood is denoted by $P(\mathbf{v}_L \mid O)$ and $P(O)$ is the object specific prior. However, in order to capture dependencies between an

object and its context the measurement vector can be extended to include features outside the target object (scene features):

$$\mathbf{v} = \{\mathbf{v}_L, \mathbf{v}_C\} \tag{2.3}$$

where $\mathbf{v}_C$ are measurements of the object's context/scene. Applying Bayes Rule now leads to an expression where all probabilities are conditioned on contextual information

$$P(O \mid \mathbf{v}) = \frac{P(O, \mathbf{v})}{P(\mathbf{v})} = \frac{P(\mathbf{v}_L \mid O, \mathbf{v}_C)}{P(\mathbf{v}_L \mid \mathbf{v}_C)} P(O, \mathbf{v}_C) \tag{2.4}$$

Based on this, Torralba implements a system which offers a procedure for object priming, context driven focus of attention and automatic scale-selection on real-world scenes.

2.3 Cue Fusion using Bayesian Networks

Cue fusion is a recurring theme in pattern recognition (see for example [Kittler *et al.* 1998]) and in particular in computer vision. Since we are going to use classifiers based on various visual cues we will use the terms "classifier combination/fusion" and "cue combination" interchangeably. In this thesis we will employ Bayesian networks, a special type of a graphical model, for cue combination (chapters 6 and 7). This is the common theme of the three articles summarized in this section which describe systems of increasing complexity. The following section 2.3.1 describes a speaker detection scheme combining four visual cues. Here we also list a number of arguments why Bayesian Networks are an attractive tool for cue fusion in general. Then, section 2.3.2 describes follow-up work using Dynamic Bayesian Networks which also adresses discriminative parameter and structure learning. Finally, section 2.3.3 gives a summary of a different system that uses Bayesian Networks to combine speech and vision in a human-computer interaction task with a robot.

2.3.1 Rehg, Murphy and Fieguth

[Rehg *et al.* 1999] use Bayesian networks for visual cue fusion. Four "off-the-shelf" vision algorithms are combined to enable a smart kiosk detect the presence of a speaker: skin color, texture, Rowley's face detector and a mouth motion detector. The actual cues are kept simple and mainly serve as a means to demonstrate their use within a Bayesian net. The structure of the network is manually defined. It encodes the context of the sensing task and knowledge about the operation of the sensors. Different structures with different capabilities are compared in this work.

The conditional probabilities along the arcs of the proposed network relate the sensor outputs to the task variables. These probabilities are learned automatically from training data. The authors admit that their actual test case is rather simple as illumination is tightly controlled and the background is mostly uniform and static. Nevertheless the Bayesian Network technique is interesting in several ways:

domain-knowledge about the individual cues and the task can be translated into the network's structure

Bayesian networks provide an intuitive graphical framework for expressing contextual knowledge, coupled with efficient algorithms for learning and inference (fusing inputs)

they can represent complex probability models, but their learning rules are simple closed-form expressions given a fully-labeled data set

contradicting evidence is dealt with in a principled way (the so-called "explaining-away" mechanism)

2.3.2 Choudhury, Rehg, Pavlovic, Garg, Huang and Pentland

Several authors propose interesting extensions to the speaker detector network of [Rehg *et al.* 1999]. Here we summarize three important extensions proposed by [Pavlovic *et al.* 2001] and by [Choudhury *et al.* 2002]. First, [Pavlovic *et al.* 2001] use a Dynamic Bayesian Network – an extension of the original static Bayesian network which can handle time-evolving random variables and their dependencies. Typical examples of Dynamic Bayesian Networks include Kalman filters and Hidden Markov Models. The Dynamic Bayesian Network is used to smooth the outputs of query variables over time resulting in higher classification stability. Within their own testing scenario the authors report an improvement of 15% over an averaged 65% detection rate of the original static Bayesian network. Second, [Pavlovic *et al.* 2001] propose a discriminative training procedure for parameter learning in Bayesian Network classifiers. This addresses an important finding of [Friedman *et al.* 1997] that the standard parameter learning through density estimation can result in suboptimal classifiers. The discriminative training is based on the AdaBoost procedure which reweighs training instances iteratively to generate a classifier ensemble. The authors report an additional 5-10% improvement in detection rate when using discriminative training. Third, [Choudhury *et al.* 2002] address the problem of structure learning. In the previous approaches the network topology was manually predefined and fixed. [Choudhury *et al.* 2002] introduce a new boosted structure learning algorithm based on AdaBoost. Given labeled data, the algorithm modifies both the network

structure and parameters so as to improve classification accuracy. Their approach uses a standard structure learning algorithm as a component (in their case they use a variant of the K2 algorithm of [Cooper and Herskovitz 1992]). Quantitative comparisons to both standard structure learning techniques and boosted parameter learning on a fixed structure demonstrate modest performance improvements. However, the analysis of the learned network structure can give important insights about the true relationship of variables and helps to uncover wrong assumptions in a manually defined topology.

2.3.3 Wachsmuth, Socher, Brandt-Pook, Kummert and Sagerer

[Wachsmuth *et al.* 2000] develop a system where a human verbally instructs a robot in a construction task (with "Baufix" toy construction elements). The human speaker has to specify building blocks in the scene by describing properties of these objects without knowing the exact terms. Therefore, the system has to interpret mostly vague descriptions which are then interpreted by a Bayesian network to find the most plausible interpretation. The system is quite complex and can also generate new Bayesian networks dynamically from the spoken utterance and the visual scene representation.

3

Color-supported Skin and Face Detection

This chapter investigates skin detection as a complementary cue for finding faces. The promise is to leverage color and shape information that is currently neglected by appearance-based, intensity-driven face detectors.

Section 3.1 develops a novel skin detection algorithm which combines color and shape cues to extract skin patches. The proposed combination scheme uses mutual information to register evidence from both cues. The hypothesis is that the combination of color and shape information outperforms traditional approaches that solely rely on color. This is tested and confirmed by experimental evaluations in section 3.2.

Then in section 3.3 we specifically investigate the usefulness of the developed skin detector for face detection. We do this by comparing and combining detections from two appearance-based face detectors (the Schneiderman-Kanade face detector and the Rowley-Kanade face detector) with detected skin regions. Finally, section 3.4 summarizes our findings and draws conclusions.

3.1 Using Color and Shape Models for Skin Detection

Skin detection by computer traditionally relies on color information. The characteristic human skin tones are due to light reflectance in a thin surface layer, the epidermis, and a thicker layer underneath, the dermis. The light absorption in the dermis is mainly due to the ingredients in the blood such as haemoglobin, bilirubin and beta-carotene which are basically the same for all skin types. However, skin color is mainly determined by the epidermis transmittance which depends on the *dopa-melanin* concentration and hence varies among human races [Stoerring *et al.* 1999].

Light sources, that is, their intensity, color and location also affect skin appearance. Other objects within the scene may cast shadows or reflect additional light and so forth. Unfortunately, there are many other objects in the world which are easily confused with skin because of their skin-like color such as certain types of wood, copper, sand as well as clothes. Imaging noise and smoothing artefacts can appear as speckles of skin-like color, too. As a result, finding skin based on color alone has many pitfalls.

To further illustrate this, figure 3.1 shows some examples of color based skin detection. Input images are shown in the top row and the bottom row shows the corresponding skin likelihood map. These likelihood maps are generated by evaluating each pixel's likelihood of being skin and are visualized such that high probabilities appear as white pixels. The leftmost example has been taken with the camera's

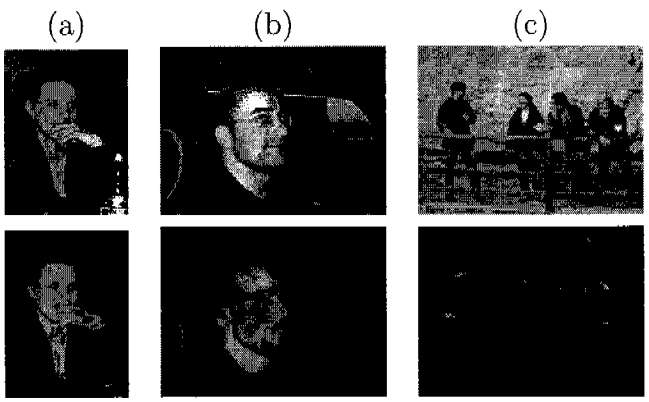

Figure 3.1: *Skin segmentation based on skin color. White pixels in the likelihood maps (bottom row) indicate high probability. Figure (a) shows specular reflections on the lady's hand. JPEG artefacts can cause false negatives and false positives as in example (b). Example (c) shows that background objects (in this case the wooden fence) can have skin-like color tones.*

flash. This causes discontinuities in the skin likelihood map due to specular reflections which visually appear as "holes" on the lady's hand. In the example in the middle the car's interior has skin tones and is therefore hard to discern from the actual skin region, i.e. certain portions of the driver's face. This image also illustrates JPEG compression and smoothing artefacts (block structure in the upper image regions). Example (c) shows a wooden fence and a crowd of people to emphasize the fact that wood (which frequently appears outdoors as well as indoors) is a common distractor in color-based skin detection.

To overcome color-specific limitations such as those illustrated above the following sections formulate a probabilistic algorithm which incorporates shape information in the detection process. The underlying assumption is that skin patches appear

as approximately contiguous skin-colored regions of certain shapes such as ellipses. The following section (3.1.1) shows how to combine color and shape information by maximizing their mutual information. The key idea is to use mutual information as an objective function for fitting a shape prior to subregions of a color-based skin likelihood map as those shown in figure 3.1. Local contiguous image regions – rather than individual pixels – can then be classified as skin or non-skin based on the computed mutual information score.

Sections 3.1.2 and 3.1.3 describe the employed probabilistic models of skin color and shape and section 3.1.4 introduces a gradient ascent method for efficient mutual information maximization.

3.1.1 Model Combination By Mutual Information Maximization

Skin patches can be described as contiguous subregions of certain colors and shapes. Our approach is therefore to combine color with shape information.

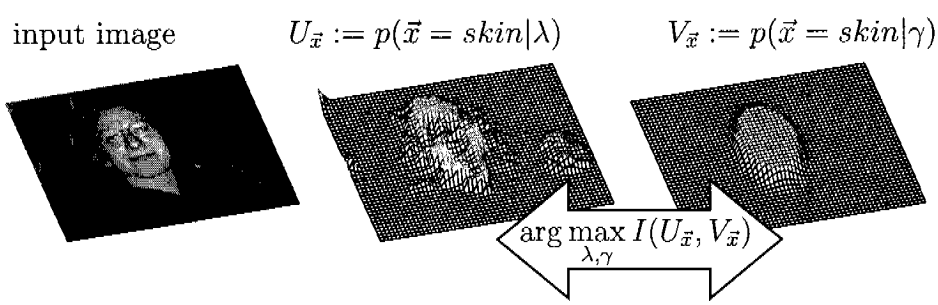

Figure 3.2: *Maximization of Mutual Information: The observed distribution of skin pixels is compared to an expected distribution which effectively imposes shape constraints. The mutual information between the two distributions is computed. The algorithm maximizes mutual information by gradient ascent on the parameters γ of the expected distribution.*

Figure 3.2 is a graphical sketch of the algorithm. For the sake of argument we assume a parametric color model with parameters λ which is applied to each colored pixel $\vec{x}$ of the input image. This results in an *observed* distribution of skin color:

$$p(\vec{x} = skin|\lambda) \tag{3.1}$$

which is then combined with the skin-specific *expected* shape distribution. We assume an elliptical shape distribution parameterized by γ which determines its actual form, location and size:

$$p(\vec{x} = skin|\gamma) \tag{3.2}$$

The combination is adaptive, that is, the parameter space of γ is searched to maximize *mutual agreement* among the two distributions in an information-theoretic sense. The parameters γ which maximize mutual agreement thus determine the location of a subregion of $p(\vec{x} = skin|\lambda)$ that best fulfills the constraints embodied by the expected distribution $p(\vec{x} = skin|\gamma)$. For maximizing mutual agreement between the two distributions we maximize their *mutual information* which is a measure of statistical dependence.

To further undermine the relevance of mutual information in this context, we briefly refer to the well-known Kullback-Leibler divergence. The KL-divergence between a probability mass function $p(u, v)$ and a distinct probability mass function $q(u, v)$ is defined as:

$$D(p(u,v)||q(u,v)) = \sum_{i,j} p(u_i, v_j) \cdot \log \frac{p(u_i, v_j)}{q(u_i, v_j)} \tag{3.3}$$

The KL-divergence (also called relative entropy or information divergence) is often used as a distance measure between two distributions[1]. By defining

$$q(u, v) = p(u) \cdot (v) \tag{3.4}$$

the mutual information can be written as the KL-divergence between $p(u, v)$ and $p(u) \cdot p(v)$:

$$I(U;V) = D(p(u,v)||p(u) \cdot p(v)) \tag{3.5}$$

Mutual information therefore measures the distance between the joint probability $p(u, v)$ and the probability $q(u, v) = p(u) \cdot p(v)$, which is the joint probability under the assumption of independence. Conversely, it measures mutual dependency or the amount of information one distribution contains about another. As a result mutual information can be used to measure *mutual agreement* between distributions.

The concept of maximizing mutual information has been previously used in several vision and machine learning tasks e.g. for feature selection [Lew and Huijsmans 1996, Colmenzrez and Huang 1997], for determining the most discriminant viewpoints of objects [Schiele and Crowley 1998] as well as for image registration [Wells III *et al.* 1996]. An important result of [Viola 1995] is that mutual information outperforms cross correlation measures when one of the signals is partially occluded. Traditional correlation is often significantly disturbed by occlusions, since they lead to substantial penalties for disagreement of intensities. The Mutual information measure on the other hand degrades gracefully when subject to partially occluded imagery. As a result color and shape distributions can be aligned robustly despite spurious "holes"

[1]However, in contrast to a "distance" measure the triangle inequation does not hold for a "divergence".

in the skin color distribution caused by specularities, shadows or the eyes, mouth and beard of a face.

The remainder of this section describes concrete implementations of the skin color model $p(\vec{x} = skin|\lambda)$, the shape model $p(\vec{x} = skin|\gamma)$ as well as an efficient search technique for mutual information maximization.

3.1.2 Skin Color Model

Evaluating a pixel's skin probability $p(\vec{x} = skin|\lambda)$ requires a statistical skin model – in this notation represented by parameters λ. Here we introduce two different variants. The first one is the comprehensive histogram model of Jones and Rehg [Jones and Rehg 1999], the second one is a compact Gaussian model in HSI color space.

Non-parametric Models of Skin and non-Skin Color

Jones and Rehg have constructed histogram-based skin color and non-skin color models from a vast collection of web images (see also chapter 2, section 2.2.1). Using their annotated data we have reconstructed both proposed histograms with (32^3) bins each (24-bit, i.e. 3 Bytes color depth, RGB). This bin configuration provided the best empirical results in their evaluations [Jones and Rehg 1999] and also outperformed a Gaussian mixture model. The histograms model the *joint* statistics of RGB values for skin and non-skin, respectively, and are trained from nearly 1 billion labeled pixels. The underlying training images contain examples of different skin complexions such as Asian, African and Caucasian. Using this model the likelihood that a given pixel is skin is approximated by the relative frequency in the corresponding histogram bin. Then following Bayesian Discriminant Analysis the skin posterior probability for a given pixel rgb can be computed as

$$p(skin|rgb) = \frac{p(rgb|skin)}{p(rgb|skin) + p(rgb|nonskin)} \tag{3.6}$$

Parametric Likelihood Model

A more compact statistical model can be obtained by using Gaussians instead of high-dimensional histograms. The resulting parametric model requires less training data to learn but sacrifices some of the accuracy and versatility of the comprehensive histogram model described above, e.g. with respect to different skin complexions.

In our case the model has been trained from a small image collection of Caucasian facial skin using the Maximum Likelihood Estimator. We transform the RGB values

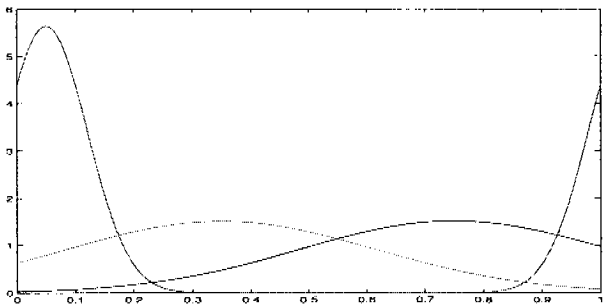

***Figure 3.3**: A statistical model representing Caucasian skin color in HSI space. The Gaussian on the left represents the hue, the one in the middle is saturation and the right-most Gaussian models the statistics of intensity values.*

to HSI space, where the distribution of skin can be well approximated by three independent Gaussians, that is the sufficient statistics are $\lambda = (\mu_h, \sigma_h, \mu_s, \sigma_s, \mu_i, \sigma_i)$ which represent the means and standard deviations along the hue (μ_h, σ_h), saturation (μ_s, σ_s) and intensity (μ_i, σ_i) axes. This is because in HSI color space Caucasian skin pixels approximately fall into a three dimensional upright hyperellipse which justifies the independence assumption. Figure 3.3 shows a plot of the color model. The likelihood of skin for a given pixel is then defined by multiplying the probabilities of the three independent Gaussians.

3.1.3 Skin Shape Model

To model the characteristic shape of skin regions $p(\vec{x} = skin|\gamma)$ our implementation uses ellipses as shape primitives. Ellipses are frequently used for modeling the human body, in particular the head, arms and limbs (see for example [Wren *et al.* 1997]). To recall, the shape distribution is parameterized by γ and represents the expected shape of an image region that corresponds to a skin patch. It is represented as a probability distribution with a compact elliptic *plateau* of high probability which smoothly decrease at the boundary (c.f. right side of figure 3.2). Mathematically, the shape distribution is defined as

$$f_{shape} = \begin{cases} logistic(a) & : \quad \frac{(x-x_c)^2}{w^2} \pm \frac{(y-y_c)^2}{h^2} \leq 1 \\ 0 & : \quad else \end{cases}$$

This means $\gamma = (x_c, y_c, w, h)$, where (x_c, y_c) denotes the plateau's center within the distribution, and (w, h) are the plateau's dimensions, assuming the shape of an upright ellipse. The actual probabilities come from the *logistic function*

$$logistic(a) = \frac{1}{1 + exp^{-a}} \tag{3.7}$$

We choose values of a such that probabilities within the plateau become relatively high and adapt the parameter towards the plateau's boundary so that probabilities decrease smoothly and eventually reach zero. While our current implementation defines only upright elliptical regions, the extension to tilted regions is straight-forward, but incurs the cost of increased model complexity of γ. A possible configuration of the shape density is visualized on the right side of figure 3.2.

The shape model embodies two important properties about the distribution of skin color. First, skin is distinguished as a contiguous region and second, skin often appears in elliptical shapes. The statistical formulation of the algorithm expresses that these requirements need to be met just in an approximate, statistical sense, which makes the scheme robust. Thus, only compact regions are regarded valid while local noise is ignored.

3.1.4 Gradient Ascent For Maximizing Mutual Information

For efficient mutual information maximization we use a form of gradient ascent on the parameter vector $\gamma = (x_c, y_c, w, h)$ traversing an adaptive local search grid. The grid center (x_c, y_c) is succinctly placed over maxima of the color distribution $p(\vec{x} = skin|\lambda)$ starting at the global maximum of this distribution and proceeding at other maxima in descending order. In the current implementation the horizontal and vertical step width (x_w, x_h) is initially set to 1/5 of the image's width and height (i.e. for a 150×100 image $w = 20$ and $h = 30$). At each iteration the algorithm follows the steepest mutual information gradient by modifying (x_c, y_c, w, h) accordingly. The search follows a coarse-to-fine strategy, i.e. the step size that affects center points x_c, y_c and dimensions w, h is succinctly decreased (e.g. halved after each iteration step).

Typically, convergence is reached in about 5 to 10 iterations. Once the algorithm has converged γ represents a single computed skin region hypothesis associated with a mutual information value. The associated mutual information is used as a confidence value. A high value expresses an agreement between color-based and shape-based skin models and thus a higher confidence in the skin hypothesis.

For generating multiple hypotheses the skin distribution $p(\vec{x} = skin|\lambda)$ is then reshaped. More specifically, after a hypothesis has been formed the associated region is first excluded from the original distribution and thereafter, the scheme is repeated. The value of mutual information is used to decide if a hypothesis is valid and if the search is to be continued. After a predefined number of examined hypotheses with a low mutual information value, the algorithm stops.

3.2 Color + Shape vs. Color alone

This section evaluates the proposed skin detection algorithm and compares it to purely color-based skin detection. Here we employ the comprehensive histogram model of Jones and Rehg and the elliptical shape model as described in section 3.1.2. In the comparison we use the original image set of Jones and Rehg and adopt their training and testing methodology, which is described in section 3.2.1. Apart from a qualitative analysis of a set of representative examples, we also provide a comprehensive quantitative comparison of the proposed algorithm in section 3.2.2.

3.2.1 Test Set of Web Images

The data set of Jones and Rehg consists of 13,640 web images 6822 of which are used for training and the remaining 6818 for testing. We reconstructed the original skin and non-skin models to allow a direct comparison using their testing methodology: within the training set 4483 photos are used to fill the non-skin color histogram and 2339 photos form the skin color histogram. Similarly, in the test set there are 4482 non-skin and 2336 skin photos. For all skin photos Jones and Rehg provide manually created skin segmentation masks which allows us to label individual pixels as either skin or non-skin.

3.2.2 Qualitative and Quantitative Analysis

The Jones and Rehg skin classifier (see section 3.1.2) classifies a particular RGB value (pixel) as skin by thresholding the posterior probability

$$p(x = skin|rgb) > \theta \tag{3.8}$$

where $\theta \in [0; 1]$ is the threshold. Classifier performance is then quantified by computing the ROC curve (Receiver Operator Characteristic curve) which measures the threshold-dependent trade-off between recall and precision. "Recall" gives the fraction of pixels labeled as skin that were classified correctly, while "precision" gives the fraction of pixels classified as skin which had also been labeled as skin:

$$\text{recall} = \frac{\text{number of correctly classified skin pixels}}{\text{number of pixels labeled as skin}} \tag{3.9}$$

$$\text{precision} = \frac{\text{number of correctly classified skin pixels}}{\text{number of pixels classified as skin}} \tag{3.10}$$

Figure3.4 shows a few example images and their ROC curves of the Jones and Rehg classifier (in short: "color", visualized as a dashed curve) versus the proposed

approach (in short: "color + shape", visualized as a solid curve). Precision is shown on the x-axis, and recall is shown on the y-axis. In these examples the proposed approach improves the equal error rate[2] between 15% as in image (d) and 35% (3.4 as in image (a). As can be seen the proposed approach is especially effective in scenes with background clutter (images (a), (c) and (f)) and difficult lighting situations (note the shadow on the left face half in image (d)). These images also demonstrate the viability of the underlying skin color model with respect to different ethnic groups (African, Caucasian, Asian).

The remainder of this section provides a more detailed quantitative and qualitative account on performance for the entire data set of 6818 test images. To summarize the performance on an individual image we define two ROC-based measures Δ_{abs} and ρ. The first measure, Δ_{abs}, integrates over the absolute differences between ROC_{color} and $ROC_{color+shape}$. This is equivalent to the total area lying *between* the two curves. Note that by definition $\Delta_{abs} \in [0;1]$.

$$\Delta_{abs} = \int_0^1 |ROC_{color+shape} - ROC_{color}|$$

The second measure, ρ, is defined as the ratio between the areas enclosed by either ROC curve. That is, it is based on the area under the ROC curve (AUC) which is a commonly used measure to summarize a classifier's performance over consecutive thresholds

$$\rho = \begin{cases} \frac{\int_0^1 ROC_{color+shape}}{\int_0^1 ROC_{color}} - 1 & : \int_0^1 ROC_{color} > \int_0^1 ROC_{color+shape} \\ & : \\ 1 - \frac{\int_0^1 ROC_{color}}{\int_0^1 ROC_{color+shape}} & : else \end{cases}$$

This definition ensures that $\rho \in [-1;1]$. Positive values of ρ indicate improvements through the proposed approach, a value of $\rho = 0$ corresponds to about equal performance and if $\rho < 0$ then the purely color-based approach performs better. The two measures Δ_{abs} and ρ are then used to index the entire set of test images.

The plot shown in figure 3.5 is a visualization of the relative performance with Δ_{abs} on the vertical and ρ on the horizontal axis. In this plot, each of the 3784 test image is represented by a cross. In 2456 of 3784 test cases (corresponding to 65%) the proposed skin finder compares favorably, increasing both precision and recall. This means the combination with shape information turned out to be beneficial in the majority of cases. The nine images shown on the top and bottom of the plot are shown at their respective coordinates (Δ_{abs}, ρ) and serve to illustrate different outcomes in relative performance.

[2]The equal error rate is defined as the curve point where precision and recall are equal. In figure 3.4 this corresponds to the intersection point between ROC curve and the first diagonal (bisecting line).

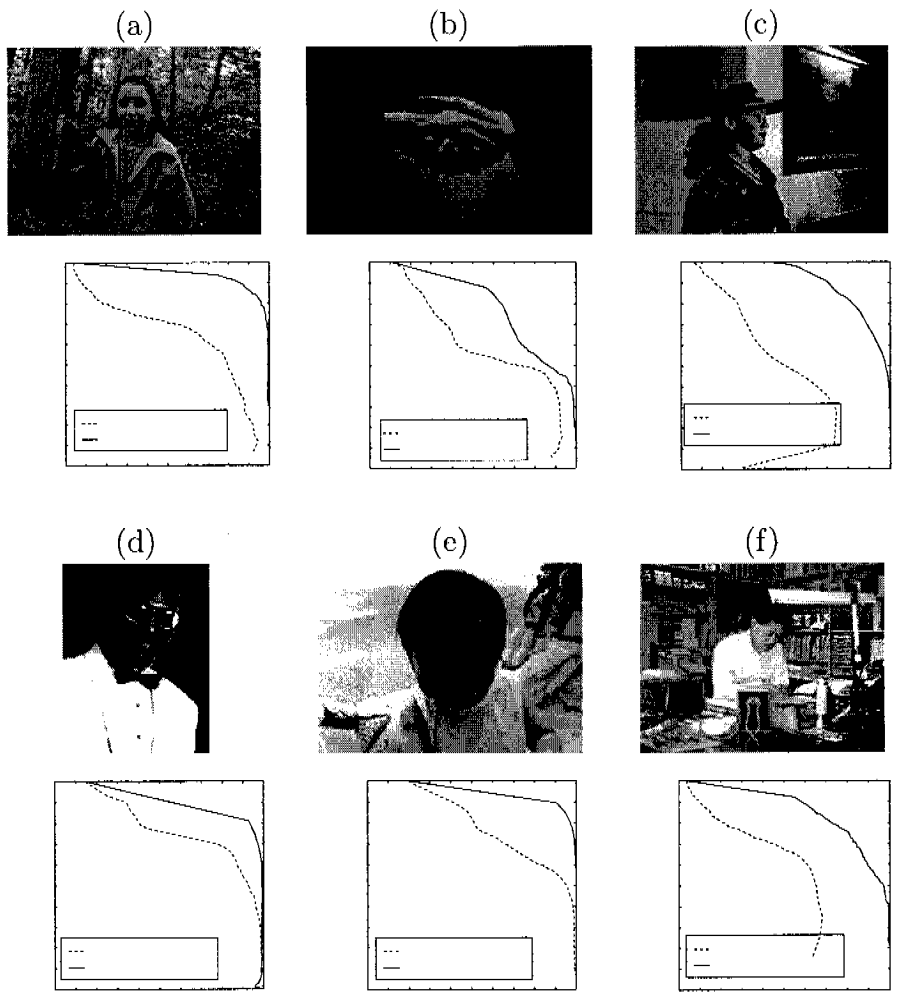

Figure 3.4: *Example web images and ROC curves for the proposed skin finder "color & shape" versus the skin classifier of Jones and Rehg "color". Using the proposed algorithm the equal error rate is improved by at least 15% (example d) and up to 35% (example a). The examples demonstrate the versatility of the skin detector with regard to different skin tones.*

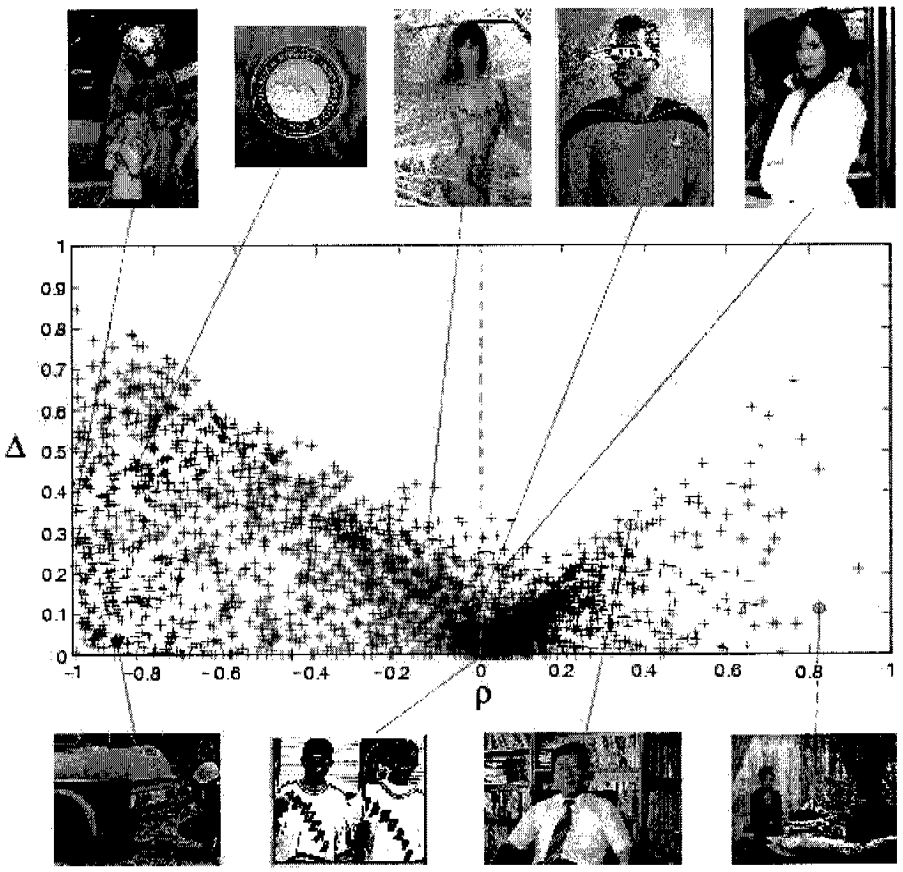

Figure 3.5: *Performance comparison on the entire test set of 6818 images. The performance measures $\Delta_a bs$ and ρ and ρ are based on the areas under the ROC curves of the two competing classifiers. In 2456 of these 3784 test cases the proposed skin finder compares favorably, increasing both precision and recall. The nine images shown on the top and bottom of this plot illustrate different cases of relative performance. They are more closely examined in section 3.2.2.*

The same images are shown enlarged together with their skin posterior maps in figure 3.6. The skin posterior maps also show the elliptical skin regions which have been extracted by the proposed skin detector. They have been grouped according to the observed relative performance: in images (a)-(c) the purely color-based approach performs better, approximately equal performance is achieved in images (d)-(f) and in images (g)-(i) the proposed approach incorporating shape information performs better.

In cases (a)-(c) the proposed skin detector erroneously hypothesizes that there is no skin contained in these images and hence there are no ellipses in the associated skin posterior maps. This is because the examined skin hypotheses only yield low information values since they do not conform to the elliptical region constraint. However, as can be seen from the skin posterior maps, a large proportion of background pixels

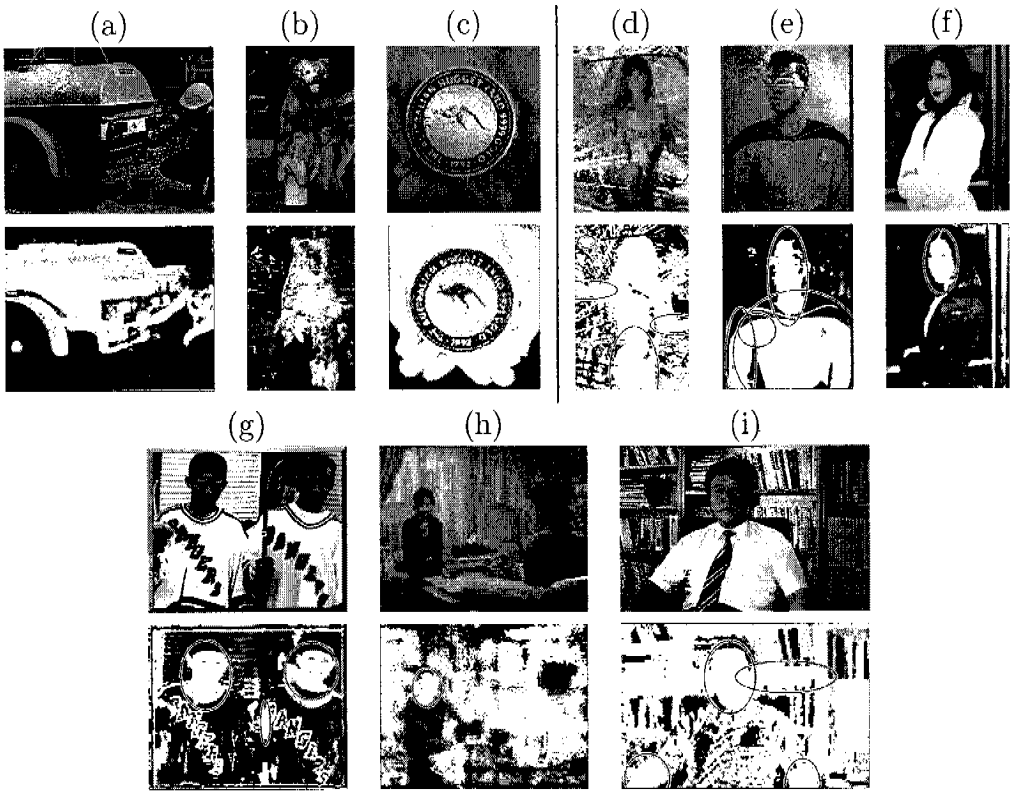

Figure 3.6: *These images characterize three different cases when comparing the purely color based approach ("color") color is better than color+shape in images a)-c), both approaches perform about equal in (images d-f) and color+shape outperforms pure color in images g)-i). Especially in images g)-i) skin patches can be well approximated by upright ellipses, their support is large enough and the background is cluttered with skin-like colors: this is the typical situation where the proposed scheme provides the largest benefit. Also note that a connected-components approach would have difficulties with such a situation.*

has skin-like colors but the actual skin regions are relatively small. From this point of view, the proposed algorithm succeeds in avoiding the many false positives in these images by concluding that there is no skin. Images (d)-(f) are three examples where ρ is approximately 1, that is, both approaches perform about equal. In image (d) not all skin regions are detected (the face is missed) but the proposed algorithm successfully curbs on the number of potential false positives – the palm tree leaves in the background have yellow tones and wooden color which are skin-like. Contrastingly, in example (e) all skin regions (here: the face) are successfully located. However, background distractors (the shirt) appear as a contiguous region of skin-like color that can be approximated with the elliptical shape model. As a result, both approaches produce similar amounts of true and false positives. Case (f) is

similar to (d). However, the amount of avoided false positives is slightly larger than the amount of missed skin pixels resulting in a value of ρ larger than 1. Finally in cases (g)-(i) the proposed method clearly improves on purely color-based segmentation: in image (g) the proposed skin finder extracts both faces which substantially curbs on the number of false positives from the background. Only a small portion of the wooden hockey stick in the middle of the skin posterior map is misclassified as skin. Examples (h) and (i) are particularly hard cases: as can be seen from their associated skin posterior maps, the majority of pixels is of skin-like color but the actual skin regions are quite small. This means that any purely color-based approach has to pay a very high penalty for correctly classifying the contained skin pixels. The proposed skin detector, however, is able to accurately locate the person's face and neck in case (h) and likewise the person's face and arms in case (i). The skin patches in these images can be well approximated by upright ellipses, their support is large enough and the background is cluttered with skin-like colors: this is the typical situation where the proposed scheme provides the largest benefit over color segmentation. Also note that a connected-components approach would have difficulties with such a situation.

3.3 Skin Detection vs. Face Detection

This section investigates the usefulness of skin detection for finding faces. Following the basic motivation of this thesis the key issue is whether skin detection is a suitable complementary cue for finding faces or not. The promise of incorporating skin concepts is to leverage color and shape information that is currently neglected by appearance-based, intensity-driven face detectors. As has been demonstrated in the previous sections the proposed skin detector is capable of localizing facial skin through the assumption of an upright elliptical shape model. In fact, this particular shape assumption can be seen as a bias for facial skin regions. Hence, we seek answers to the following two questions: Can we find faces via skin detection that are missed by current face detectors? And: Can we reduce the number of false alarms using skin concepts?

To shed light onto these questions, detected skin regions are compared to the detections of the Schneiderman-Kanade face detector [Schneiderman and Kanade 2000] and to the detections of the Rowley-Kanade face detector [Rowley *et al.* 1998]. To recall, the Schneiderman-Kanade detector is able to detect both frontal and out-of-plane rotated faces (profiles). Contrastingly, the Rowley-Kanade detector detects frontal and in-plane rotated faces. By considering detections of both face detectors we can thus better evaluate the true potential and possible complementarity of the skin cue. It must be emphasized again that both face detectors are intensity-driven and appearance-based, they do not use color nor do they use an explicit geometric

shape model (see chapter 2, sections 2.1.1 and 2.1.2 for their technical summary). The following section 3.3.1 describes the employed test set with face annotations. Then, sections 3.3.2 and 3.3.3 present and analyze the obtained qualitative and quantitative results.

3.3.1 Test Set of Color Consumer Digital Photographs

The following experiments are based on a set of color photos where all human faces have been manually annotated [3]. These annotations will serve as ground truth for computing the number of hits and false alarms for a given detector. The test database contains 535 color JPEG images which have been downsampled from a 3.3 Megapixel resolution to 150 × 100 pixels. A subset of these images has been also used in the error analysis of chapter 1. Most pictures of the test data are snapshots of one or multiple persons. By this we mean that people are not necessarily looking directly into the camera like on a portrait. Instead the database contains faces in many different orientations, at various scales, looking in arbitrary directions. The total number of faces contained in these photographs is 692. These photos cover a wide range of real-world situations both indoor and outdoor. For example, indoor scenes include a number of flash-light photos from meetings and parties. Outdoor scenes include photos from ski-trips and beach scenes.

3.3.2 Qualitative Analysis

Figures 3.7 and 3.8 show example images with the outputs of the two face detectors as well as the proposed skin detector's output. For evaluating Schneiderman's approach we accessed his implementation through the provided on-line demo[4]. Rowley provided a copy of his system. The 6 images of figure 3.7 illustrate the effects of scale and rotation variation, as well as the effects of occlusion, face-like distractors and complex scenes with many people. Following is a qualitative analysis of these cases comparing the three detectors. Example (a) illustrates the effect of scale variation. There are 4 frontal faces in this image (the fourth's person body is covered by the left person, but all four faces are at least partially visible). Both face detectors miss the two small faces in the background since the support regions are too small. Only the skin detector finds all four faces (the left hypothesis is wider because it actually contains two faces).

[3]Current standardized Face Detection Test Databases such as the CMU+MIT test set unfortunately do not provide color information. For this reason we chose to create this new test set.

[4]http://vasc.ri.cmu.edu/cgi-bin/demos/findface.cgi

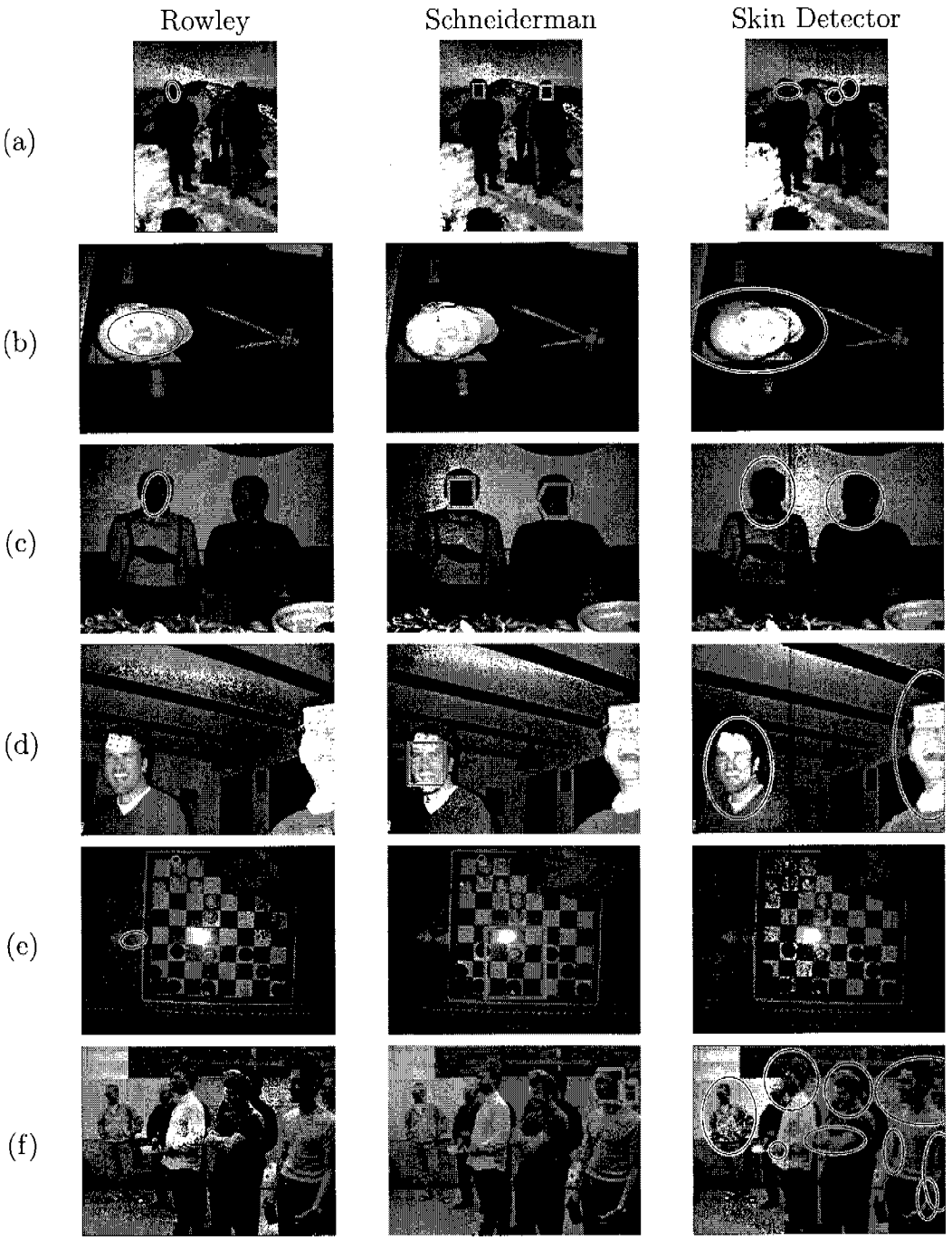

Figure 3.7: *Characterization of appearance-based face detection vs. skin detection. The first two columns show the face detection results of Rowley and Schneiderman, row three shows the skin finder's output. These examples illustrate characteristic effects of (a) scale, (b) in-plane rotation, (c) out-of-plane rotation, (d) occlusion, (d) face-like distractors, (f) crowds*

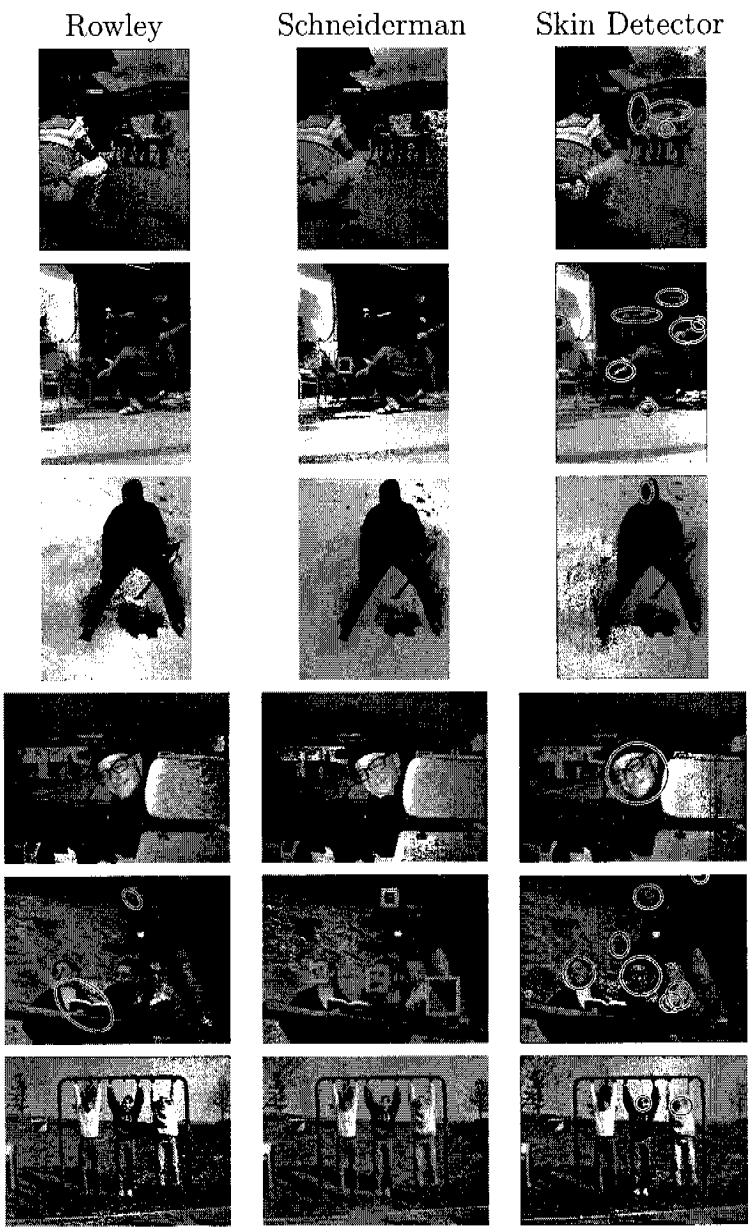

***Figure 3.8**: Image gallery showing additional results of appearance-based face detectors (Rowley, Schneiderman) and the proposed skin detector.*

The next two rows illustrate the effect of rotation. Rowley is able to deal with a 90 degree in-plane rotation in example (b), where Schneiderman fails. The situation is reversed when presenting an out-of-plane rotated face as shown in example(c). Again only the skin detector can handle both cases correctly because it is rotation insensitive. Also note how in image(b) the proposed approach robustly deals with the specularities on the person's forehead and cheeks.

Image (d) illustrates the effect of occlusion. The face to the right is only half visible. Both face detectors fail here even though Schneiderman's detector is clearly "parts-based" (in the sense that the modeling is not holistic, but instead local facial subregions are treated independently). Rowley's approach misses the other face, too, because of in-plane rotation. Both faces are detected by the skin-based approach. In addition, the skin detector is robust to variations in facial expression (in this case laughing).

Face-like distractors pose additional challenges. Checkered surfaces are likely to cause false positive detections, especially with Schneiderman's detector. The pattern is likely to be confused with facial features like mouth, nose and eyes. It often appears on shirts and other clothes, in case (e) it appears on a chess board. Both face detectors falsely detect faces. The skin detector is not misled by this situation. Note that in this example, the skin detector misses the chess player's hand. This is a limitation of the current scheme which is limited to skin regions of upright elliptical shape.

Finally, case (f) shows a more complex scene with a crowd of people. Faces are in profile view, some of them are only partially visible. Since the skin detector allows for discontinuities in the skin color distribution, it also works when people wear beards, piercings or glasses (as can be observed for the middle person in this example). Appearance based face detectors are more likely to encounter difficulties in these cases. Rowley has no detections, Schneiderman can deal with profiles and returns two faces. The face detector retrieves all five faces and some additional skin regions. Additional result images can be found in figure 3.8. The next section aims at quantifying the usefulness of skin concepts in face detection.

3.3.3 Quantitative Analysis

To further analyze the potential of a combined scheme building on both skin and face concepts we empirically evaluated the performance of the individual detectors on all 535 images.

Detection rates and false alarm levels for each detector are quoted in table 3.1. As can be seen, the skin finder returned 74.4% of all faces, whereas Schneiderman's face detector has a recall rate of 55.9% and Rowley's scheme 27.7%. That is, the skin

detector's recall rate in detecting faces is almost 20% higher than Schneiderman's algorithm and 47% higher than Rowley's approach. As can be expected we found complementary results for precision. Since the proposed skin detector is designed to return skin region in general, not just faces, its precision is only 28.3%. Rowley's scheme reaches 60.5% and Schneiderman 70.1% on this data set. As precision and recall rates of skin and face detectors turned out to be *complementary* we examined their combinations. When counting all faces found by either Rowley's scheme OR

	tp	fn	fp	precision	recall
Schneiderman	387	305	165	70.1%	55.9%
skin patch OR Schneiderman	639	53	1458	30.5%	92.3%
skin patch AND Schneiderman	263	429	12	95.6%	38.0%
Rowley	150	542	94	60.5%	27.7%
skin patch OR Rowley	562	130	1397	40.2%	81.2%
skin patch AND Rowley	103	589	2	98.1%	14.9%

***Table 3.1**: A quantitative account of appearance-based face detection (Schneiderman, Rowley) and its combination with the proposed skin detector. Results are from a test set of 411 real-world consumer photographs. Here, Schneiderman's scheme compares favorably to Rowley's. When combining Schneiderman's face detector with the proposed skin finder, recall (OR-combination) or precision (AND-combination) is leveraged to above 90% in both cases.*

the skin detector the recall rate is boosted to 81.2% (versus an initial rate of 27.7%). Precision is raised to 98.1% (versus 60.5%) when counting only those faces found by BOTH detectors. Results from combining the skin detector with Schneiderman's approach reveal the following: precision reaches 95.6% (versus 70.1%) which is comparable to the combined performance using Rowley's approach. Recall is raised to 92.3% using a logical OR combination. This is even higher than the combination with Rowley's scheme. Depending on the type of combination Schneiderman's face detector in combination with the proposed skin finder reaches precision or recall rates above 90%.

3.4 Conclusion

This chapter has proposed a novel method for finding skin. There are two main results:

First, there is a clear benefit in modelling skin as approximately contiguous regions of certain colors *and* shapes rather than relying on color alone.

Second, appearance-based face detectors can benefit from skin detection since they do have certain complementary strengths and weaknesses.

The skin detection algorithm builds on statistical skin color and shape models which are combined to extract skin region hypotheses. A novel combination scheme is presented which draws upon concepts from information theory. The combination is adaptive in that model parameters are automatically tuned to maximize mutual agreement among color and shape distributions. Its evaluation is based on a comparison with the most comprehensive statistical skin color model we are currently aware of [Jones and Rehg 1999]. It has also been shown that a face detector's false alarms can be greatly reduced based on skin detection. Alternatively, more faces can be found by including found skin patches. Skin detection thus becomes useful in these limits of very high recall or very high precision.

However, skin detection as discussed in this thesis ultimately requires color information which might not always be available. For instance, many surveillance applications are based on cameras which often do not provide color. Also, color information can sometimes be too corrupted (e.g. by illumination conditions or sensor noise) to be useful. Therefore, a completely orthogonal cue solely based on intensity information is explored in the next chapter.

Seite Leer /
Blank leaf

4

Context-supported Face Detection

This chapter investigates the use of a face's local context as indirect evidence for face detection. Section 4.1 briefly reviews the concept of local context in its relation to human vision. It is well known that the human visual system makes heavy use of local context and contextual cues in general – while computational vision traditionally does not do so. The section then describes a straight-forward approach to test the concept of local context within computational face detection. The promise is to go beyond the detection capabilities of traditional face detectors, especially as concerns resolution robustness,. This is particularly motivated by the large number of missed low resolution targets in the error analysis of chapter 1. A proof of concept is then delivered in section 4.2 which implements a local context detector within the Schneiderman-Kanade framework. This first version of the local context detector is evaluated on the standard MIT+CMU image set and a set of vacation photographs. Then, section 4.3 develops a much faster version based on Viola-Jones' cascade learning algorithm and evaluates the detector on two PETS[1] video sequences. Finally, section 4.4 summarizes this chapter's most important findings.

4.1 The Role of Local Context

Sinha and Torralba [Sinha 2002, Torralba and Sinha 2001] conducted experiments in which humans were asked to discriminate faces from face-like random patterns. Classification performance was observed as the resolution of the test images was successively decreased. Torralba's experiments show that as the level of image detail decreases humans make increasingly use of the *local context*, i.e. a local area surrounding the face to help in their decision. As a result the human visual system can robustly discriminate real faces from face-like patterns at very low resolutions. This phenomenon is illustrated in figure 4.1 showing a face and its vicinity at decreasing

[1]Performance Evaluation of Tracking and Surveillance (PETS), International Workshop

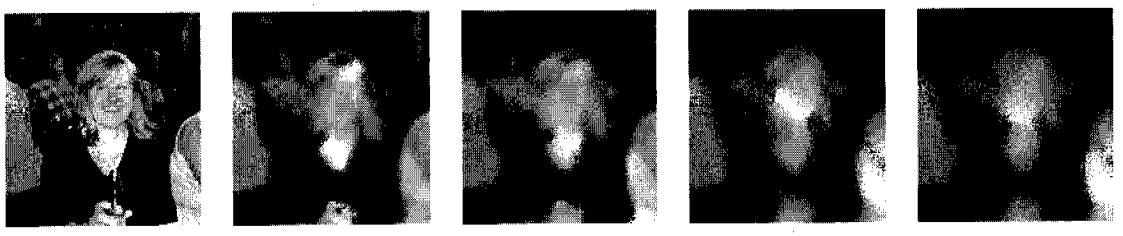

***Figure 4.1**: A face and its local context. As the image resolution is decreased (left to right), details such as facial features vanish. However, even at the lowest resolution humans are still able to infer the presence of a face because its context can be recognized.*

resolutions. A human observer can still infer the presence and location of a face at the lowest resolution shown here because the context can still be recognized.

Computational systems on the other hand traditionally assume that the only image features that are relevant for the detection of an object at one spatial location are the features that potentially belong to the object – and not to the background or the surrounding area. This computational approach is therefore called *object-centered*. As a result the robustness of computational detectors with respect to degrading resolution is much more limited. When comparing object-centered face detectors with human observers, [Sinha 2002] finds that current computational detectors not only require a much larger amount of facial detail for detecting faces in real scenes, but also yield false alarms that are correctly rejected by human observers.

This raises the question whether local context can be effectively employed for computational vision, too. In computational face detection the active use of local context implies including the target's close vicinity into the computation as a form of "indirect evidence". Figure 4.2 illustrates some of the variation contained within the close surrounding of faces. This may include portions of various types of backgrounds or parts of other people standing nearby. However, below the target face it is also likely to contain some parts of the person's upper body, shoulders, the neck and head contours. Since this is a recurring pattern the hope is to automatically recognize the local context based on these cues. To test the feasibility of the local context idea we therefore train an object detector with instances that contain a person's entire head, neck and part of the upper body. Examples of the training data are shown in figure 4.3 along with instances of the traditional object-centered paradigm for comparison (bottom row – these examples are taken from the original Schneiderman-Kanade training set).

During detection the actual face location is inferred by assuming a fixed position within the detection window. This is illustrated in figure 4.4. The size and location of the face (w, h) are set proportionally to the width W and height H of the detected local context: $w = W/2$, $h = H/2$. The offset relative to the upper left corner is

***Figure 4.2**: Examples of different faces (inner rectangle) and their local context (outer rectangle). These images illustrate the great variation contained within the close surrounding of faces but also demonstrate that a person's upper body, shoulders, the neck and head contours are strong cues that hint at the presence of a face. For illustration purposes only one face per image is examined here.*

computed as $(W/4, H/10)$. These constants have been estimated from a small set of example instances of faces within their local context.

4.2 Proof of Concept

This section describes the implementation and evaluation of a face detector which is based on the concept of local context detection. The actual implementation is based on the Schneiderman-Kanade face classifier. Section 4.2.1 gives a brief summary about this classifier. Implementation details have been confined to a separate section 4.2.2. Then, two sections are devoted to empirical evaluations: section 4.2.3 compares the quantitative performance of the local context detector to the original Schneiderman-Kanade detector and to the skin-based face detector developed in chapter 3. Finally, section 4.2.4 further examines images where faces have been detected through local context, but were missed by both competing detectors.

4.2.1 Implementation based on Schneiderman-Kanade

The employed appearance-based detector is a modified version of the Schneiderman-Kanade approach. The detector runs a detection window over the input image in an exhaustive search over all positions and scales. Each input instance is classified

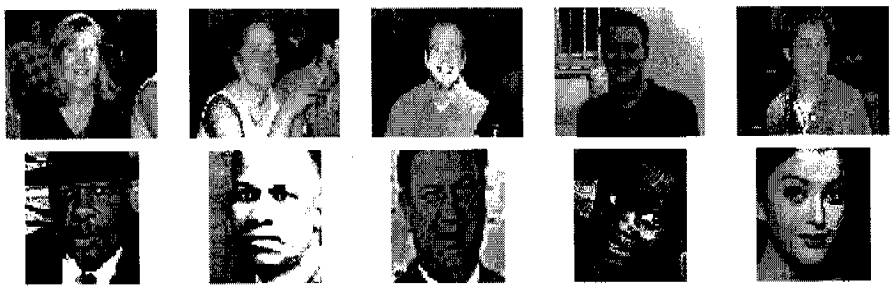

***Figure 4.3**: Examples of training instances used in the proposed approach based on local context (top row) versus the traditional object-centered approach (bottom row). The resolution is 56x48 and 48x56 pixels, respectively.*

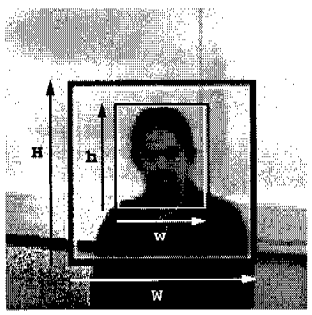

***Figure 4.4**: Whenever the local context of a face is detected the actual face location is inferred by assuming a fixed position within the detection frame. Here $w = W/2$, $h = H/2$ and the offset relative to the upper left corner is set to $(W/4, H/10)$.*

independently based on a likelihood ratio which is computed from face and nonface probabilities. These probabilities are represented as a function of the observed features. The features of this detector capture local arrangements of quantized wavelet coefficients. To illustrate the differences of what is being modeled in the local context case and in the object-centered case an example training instance is depicted in figure 4.5 together with a visualization of the corresponding wavelet decompositions. In the case of local context (right side in the figure), the wavelet decomposition shows most parts of the upper body's contours, as well as the collar of the shirt and the boundary between forehead and hair. The shoulders, for example, appear in the HL subband (upper right subband) while the upper arms and the head contour are clearly visible in the LH subband (lower left subband). Also note that facial parts such as eyes and mouth are hardly discernible in the right wavelet transform and therefore *do not* contribute to the modeling of local context. Hence, this is quite different from data presented to traditional face detectors, where facial parts are typically the most informative cues. Facial features are clearly visible in the left wavelet transform: the upper right subband shows eyes and mouth, the

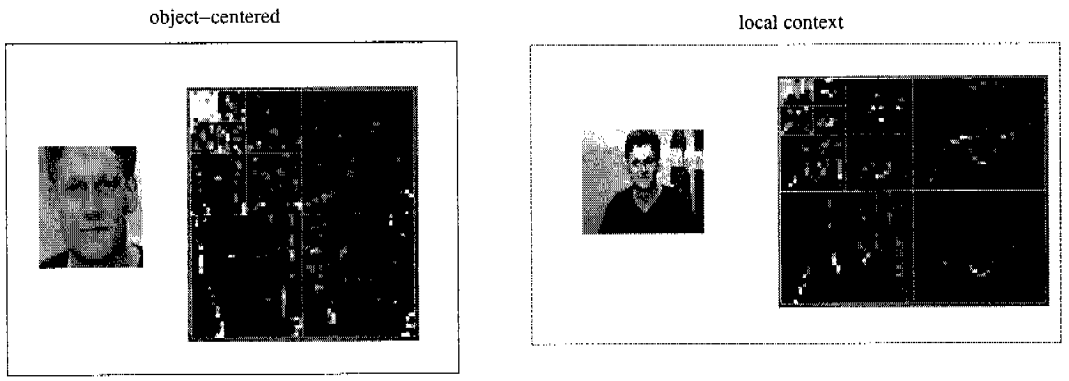

***Figure 4.5**: This figure illustrates the differences in the features (wavelet coefficients) that are being modeled in the object-centered case (left box) and in the local context case (right). In the left wavelet decomposition the facial features such as eyes and mouth are clearly visible. These can hardly be discerned in the wavelet decomposition on the right side. However, other features such as the collar of the shirt, shoulders, head and body contours become visible here which do not exist in the object-centered case.*

lower left subband also shows the eyes as well as the nose.

4.2.2 Implementation Details

The classifier computes a likelihood ratio from the face likelihood and nonface-likelihood of n feature types $pattern_k$ evaluated over all positions x, y within the detection window

$$\prod_{k=1}^{n} \prod_{x,y \in region} \frac{p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid object\right)}{p_k\left(pattern_k\left(x,y\right), i(x), j(y) \mid nonobject\right)} > \theta \tag{4.1}$$

The decomposition (products) is based on the assumption of independence, i.e. the Naive Bayes assumption. From the above equation follows that the likelihood functions themselves depend on $i(x)$ and $j(y)$ which are coarse quantizations of the feature position x, y within the detection window. This spatial dependency allows to capture the global geometric layout: within the detection windows certain features might be likely to occur at one position but unlikely to occur at another. There are $n = 17$ different types of features involved. Figure 4.6 describes their basic concept and shows one particular feature type $pattern_{\tilde{k}}$ for illustration. An input instance is first decomposed by a two-dimensional 3-level wavelet transform using a biorthogonal 5/3 filter bank. Features are then extracted by examining local arrangements of a small number of wavelet coefficients (in this example eight coefficients). An arrangement can combine coefficients from one single subband or

span over different subbands. In the example, four coefficients from the HL subband and four from the LH subband (both from the first level of the transform) are combined to form one feature value. This way of combining information from different subbands allows to capture dependencies across space (arrangements with different extents), frequency (different wavelet levels) and orientation (subbands with different filtering directions, i.e. LH and HL). The individual class-conditional likelihoods

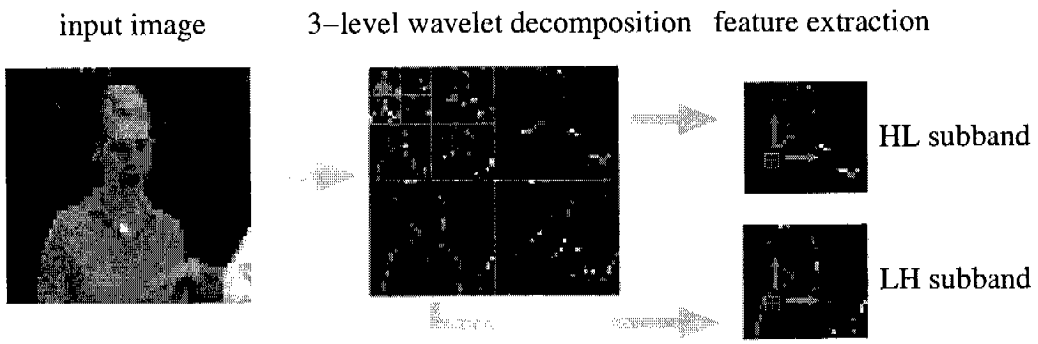

***Figure 4.6**: Feature extraction: local arrangements of quantized wavelet coefficients are combined to form one single feature value. In this example four coefficients from the HL subband (upper right subband in the wavelet decomposition) and four from the LH subband (lower left subband) capture the dependency between horizontal and vertical orientations. Such arrangements are examined over all locations within the involved subbands (oversampling).*

$p_k\left(pattern_k\left(x,y\right),i(x),j(y) \mid object\right)$ and $p_k\left(pattern_k\left(x,y\right),i(x),j(y) \mid nonobject\right)$ are represented by high dimensional histograms. Depending on the feature type these can have up to 1.1 million bins. 2000 local context examples were collected for training the positive class. These instances were gathered from the world wide web and from private foto collections. In order to limit the amount of variation and hence increase discriminance only frontal views have been used. Further, taking the shoulders as landmark points the instances have been roughly aligned and vertically mirrored leading to an effective training set size of 4000 instances. It is very important to collect good examples for the *nonobject* class. Training with random samples will most likely result in an inaccurately modeled decision boundary. Ideally, the likelihood should accurately model nonobjects which are very similar to objects, i.e. they can accurately represent the decision boundary. For collecting nonobject instances we apply an algorithm that is sometimes referred to as "bootstrapping": an initial version of the detector is run over images not containing the target object. Any false alarms are collected as negative training examples. These are then used to retrain $p_k\left(pattern_k\left(x,y\right),i(x),j(y) \mid nonobject\right)$ and the algorithms is iterated. We use a validation set to determine when to stop this iteration. For detecting faces larger than the detection window the input image is successively downscaled. The

current implementation uses a scaling factor of $2^{\frac{1}{4}}$, i.e downscaling four times corresponds to reducing the original resolution by a factor of 2. Since the local wavelet transform is not shift-invariant an overcomplete transform of the entire input image is computed. This allows to apply the classifier at the original image resolution (except the right and bottom image border which is induced by the size of the detection window). For example, given a single 180 by 240 image, the detector searches seven scales, applying the classifier 74144 times altogether. After all positions and scales have been classified independently, the final detection results are extracted by an arbitration scheme: the most confident detection is determined and nearby confidence values (both spatially and in scale) are tagged to belong to this same detection. This continues as long as the confidence exceeds the current acceptance threshold θ.

4.2.3 Comparison of Detection Rates of face, skin and local context detectors

In order to understand the relevance of local context several experiments have been carried out on two data sets, the MIT+CMU frontal face data set as well as a set of color vacation fotos. Both data sets are disjoint from the data used for training. The MIT+CMU data set consists of 125 grayscale images with 448 frontal faces[2]. The vacation foto set consists of 535 color images with 901 faces, 388 of which are profiles. Images in both data sets are upright. Note that the image resolution of the MIT+CMU data set is correlated with the image content: portraits have lower resolution than group pictures which effectively guarantees a certain minimum resolution for the contained faces. There is no such correlation for the vacation data set where all images are 180 by 240 pixels.

The left plot in Figure 4.7 shows the face detection performance on the MIT+CMU data set in terms of the ROC curve. The percentage of retrieved faces is shown on the vertical axis and the absolute number of false detections on the horizontal axis with a logarithmic scale. Both the performance of the object-centered and the local context detector are shown. For the object-centered detector the authors' original implementation was used which can be accessed through an on-line web interface [3]. Both frontal faces and profile faces can be detected. The threshold range in this implementation has a lower limit because the detector is cascaded: only for candidates that make it to the final classifier stage a full confidence score is computed. This means the ROC curve can be plotted only up to a certain number of false positives. Note that the local context detector has to infer the actual location

[2]As in [Schneiderman and Kanade 2000] five images containing drawings of faces have been removed from the original set of 130 images

[3]http://www.vasc.ri.cmu.edu/facedemo/

of a face indirectly. We do this by assuming a fixed head position within the detected local context. As can be seen in the left of figure 4.7 the object-centered approach

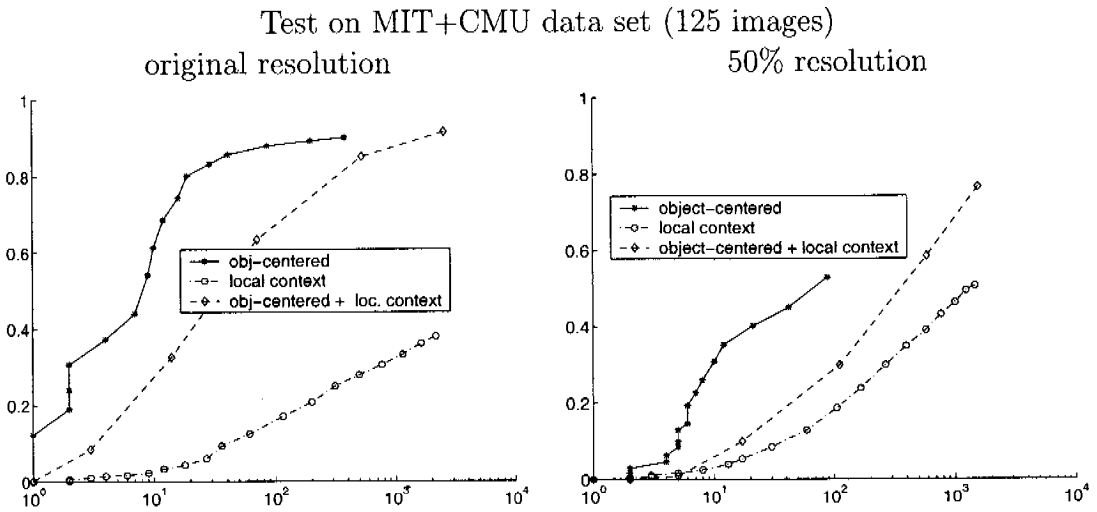

***Figure 4.7**: ROC curve showing the percentage of detected faces (vertical) vs. the absolute number of false positives (horizontal, logarithmic scale). At the original image resolution (left plot) the object-centered approach is sufficient to detect 90% of all faces. When the image resolution is decreased by 50% (right plot) the object-centered approach by itself retrieves only 55% of the faces. For the object-centered detector detection rates at higher false alarm levels cannot be computed because of technical limitations of the web-based implementation. Interestingly, the combination with detections from the local context cue immediately yields an additional 25% of correct detections, which suggests the local context contributes truly novel correct detections.*

detects 90% of the faces with very few false positive detections. The curve flattens from thereon not yielding any additional faces. On the other hand the local context detector produces many false alarms and at 10^3 reaches a detection rate of 40%. However, the true potential of local context becomes apparent as the resolution of the test images is decreased. At half of the original resolution both detectors individually retrieve about 55% of the faces (right plot). At this reduced resolution, a large part of labeled faces become smaller than the detection window size of the object-centered detector – which means that its detection rate cannot further increase (unless one admits additional random detections). By merging detections of both object-centered and local context detector, their joint ROC curve can be computed. Interestingly, the joint detection rate reaches more than 80%. This means that the joint detection rate surpasses the rates of each individual classifier by more than 25%. The gain in joint detection rate indicates that object-centered cues and local context cues are actually *complementary*. A look at the images where a face

was found by one cue but not by the other revealed two cases: in portrait images the local context is often not contained. The image is cropped to contain the head and the neck only but not the full silhouette of the upper body. The object-centered approach is obviously well suited for this case while the local context cue is not. On the other hand the decrease in resolution deteriorates some of the facial features which may lead to failure of the object-centered detector. This case is typically well suited for the local context cue.

The remainder of this section is concerned with a similar analysis using color images. The availability of color information allows for an additional comparison to the skin detector developed in chapter 3. A particularly interesting question is if the local context cue yields novel detections neither found by the skin detector nor by the object-centered detector (the local context detector and the object-centered approach still use only grayscale information). The image set used here contains 535 consumer digital photographs from a single camera covering a wide range of real-world situations both indoor and outdoor (parties, ski-trips, beach scenes etc.). The left plot in figure 4.8 shows the individual ROC curves of the object-centered detector, the skin detector and the local context detector when applied to this data set. The object-centered detector has the steepest slope and reaches a detection rate of 30% which proves the difficulty of the dataset. The skin detector reaches 40% and the local context detector 50%. On the right side in the figure the joint performance of all possible pairs as well as all three cues taken together is evaluated. The joint detection rate of object-centered and skin detector reaches 60%. Either combination with the local context detector, however, retrieves even 70% of the faces in the database. Combining all three cues does not further increase the detection rate (the curve lies on top of the local context + skin curve and is therefore hard to distinguish). This indicates that the 10% improvement between the object-centered+skin pair and the other two pairs is due to detections from the local context cue. For this database this means about 90 additionally retrieved faces that otherwise would have been overlooked.

4.2.4 Analysis of Novel Face Detections

As suggested by the above quantitative results the local context cue yields novel correct detections. This section gives a qualitative analysis of the specific images where novel detections occurred. A representative subset of these images is shown in figure 4.9. Novel detections are visualized as rectangles. The images in the top row show successful detections of very small faces the smallest being 15 by 20 pixels in size. The object-centered approach requires a higher resolution that is not available here. This supports the hypothesis that local context is indeed useful in low resolution cases. A typical example are group pictures where faces naturally occur at lower resolution than say in portraits. On the other hand group pictures

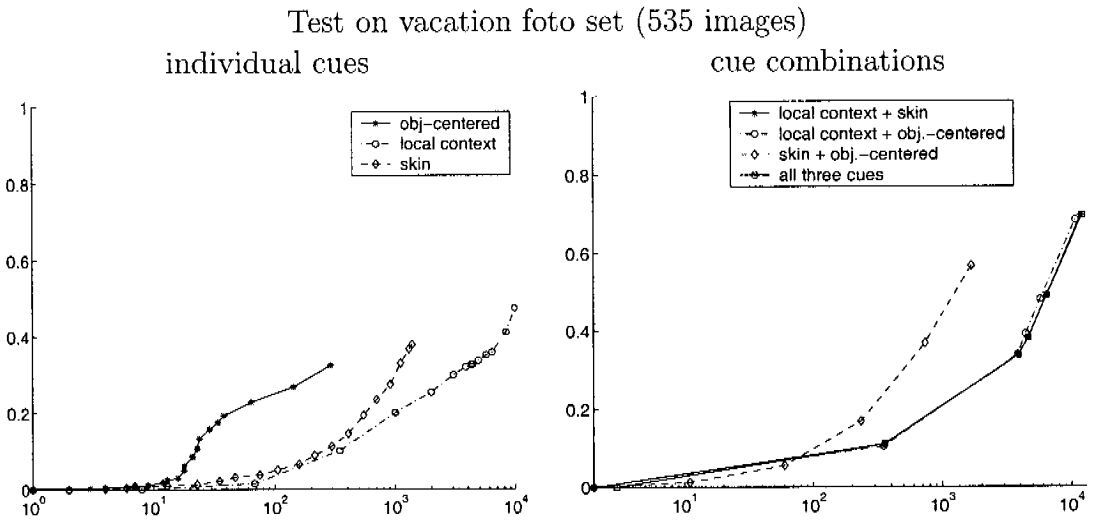

***Figure 4.8**: Performance of the individual cues and their combinations. The 10% improvement between the object-centered+skin pair and the other two pairs is due to detections from the local context cue (right plot). For this database this means about 90 additionally retrieved faces that otherwise would have been overlooked.*

provide rich contextual information such as the heads or shoulders of other people standing nearby. The local context for an individual face is thus affected.

While in these previous examples all the facial details are still visible they are partially concealed in the next few examples shown in row 2. This often happens in snapshot fotos because of a special facial pose, occlusion or low light. As for facial pose, an appearance-based object-centered approach could theoretically be trained accordingly, but in practice acquiring enough of this type of training data is difficult. Occlusion occurs, for example, because people often lean onto their hands, or wear sunglasses or beards. The object-centered approach can sometimes overcome this problem at higher resolutions but fails as resolution decreases. A particularly interesting low light situation is shown in the first image of row 5. Due to lack of contrast this face could not be detected by the object-centered approach even though the resolution would be well sufficient. Likewise skin color could not be detected given such difficult illumination. The local context could still be detected. The inferred face position, however, is not very accurate because a fixed position within the detected local context is assumed. Alternatively one could issue a local search for head contours, e.g. within a the upper half of the detected local context.

Figure 4.9: *Novel face detections indirectly inferred from the local context cue. The object-centered approach fails on these instances mainly because faces are too small or because of their specific pose. The skin detector fails on the same instances mainly because of illumination problems or because the extracted skin region's size and location is too inaccurate.*

4.3 A Real-time Local Context Detector

The previous section has demonstrated the feasibility of using local context for computational face detection. In this section we develop a much faster implementation of the local context detector by learning boosted classifier cascades [Viola and Jones]. The following section 4.3.1 summarizes the most important facts about integral image features, AdaBoost and cascade learning. Implementation details are reported in section 4.3.2. Then, section 4.3.3 evaluates the detector on a standard video sequence used in the International Workshop on Performance Evaluation of Tracking and Surveillance (PETS). This is done without tracking, that is, the detector is applied independently to each video frame. Finally, section 4.3.4 discusses results on a second PETS video sequence: an outdoor surveillance scenario.

4.3.1 Implementation based on Viola-Jones

The employed framework is Lienhart's extended version of the Viola-Jones detector [Lienhart *et al.* 2003a]. Details about the underlying algorithms, the learning approach and the parameters are given in a separate section (section 4.3.2). Basically,

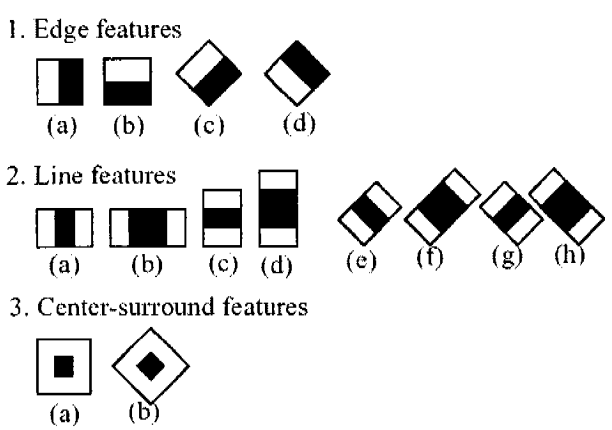

***Figure 4.10**: Lienhart's extended integral feature set including rotated features and center-surround features. These features are reminiscent of Haar wavelets and early features of the human visual pathway such as center-surround and directional responses.*

the features of this detector are weighted differences of integrals over rectangular subregions. Figure 4.10 visualizes Lienhart's extended set of available feature types (figure taken from [Lienhart *et al.* 2002b]) where black and white rectangles correspond to positive and negative weights, respectively. The feature types consist of four different edge features, eight line features and two center-surround features.

The learning algorithm (which is in a way reminiscent of decision-tree learning) automatically selects the most discriminant features considering all possible feature types, sizes and locations. Again, we can compare the selected features in case of the object-centered approach versus the local context approach. Figure 4.11 is a visualization of the selected rectangle features. Two different training instances (individuals) are shown in the local context case for illustration purposes. The features displayed here are evaluated in the first stage of the detector cascade and can therefore be regarded as the most important features. There are 9 features in the object-centered case and 14 features in the local context case (the learning algorithm determines the number of required features per stage automatically).

In the object-centered case features f1, f2 and f3 capture inner-face regions, in particular around the eyes and around the mouth and nose by using horizontal and vertical line features. Additional features are mostly edge features to capture the contours of the head and chin (f5, f8 and f9 in the figure). Contrastingly, for the local context case the very first feature f1 extends over the entire patch, i.e. it actually makes use of the local context. The following features capture head and body contours (features f2-f5). Other features capture the left and right shoulder (features f9 and f10 in the example 1, feature f14 in example 2). Hence, the information used in this local context detector is again quite different from traditional face detectors, which rely on facial parts alone.

4.3.2 Implementation Details

The employed detector framework is based on the idea of a boosted classifier cascade (see [Viola and Jones]) but extends the original feature set and offers different boosting variants for learning [Lienhart *et al.* 2002b]. This section summarizes the most essential implementation details regarding features, learning algorithm and training parameters. The feature types as depicted in figure 4.10 are reminiscent of Haar wavelets and early features of the human visual pathway such as center-surround and directional responses. Their main advantage is that they can be computed in constant time at any scale. Each feature is computed by summing up pixels within smaller rectangles

$$\text{feature}_I = \sum_{i \in I = \{1,\ldots,N\}} \omega_i * RecSum(r_i) \tag{4.2}$$

with weights $\omega_i \in \mathbb{R}$, rectangles r_i and their number N. Only weighted combinations of pixel sums of two rectangles are considered, that is, $N = 2$. The weights have opposite signs (indicated as black and white in the figure), and are used to compensate between differences in area. Efficient computation is achieved by using *summed area tables*. Rotated features and center-surround features have been added to the original feature set of Viola-Jones by Lienhart et al [Lienhart *et al.* 2002b] using

rotated summed area tables. Within figure 4.10 the original set consists only of features (1a), (1b), (2a) and (2c) as well as one diagonal feature which is subsumed by the rotated features. Lienhart's augmented feature set has been shown to extend the expressiveness and versatility of the original features leading to more accurate detectors. Note again that this feature representation does not require to compute an image pyramid to search over different scales.

The cascade learning algorithm is similar to decision-tree learning. Essentially, a classifier cascade can be seen as a degenerated decision tree. For each stage in the cascade a separate subclassifier is trained to detect almost all target objects while rejecting a certain fraction of the non-object patterns. That is, an individual stage is characterized by its false alarm rate f_i and its detection rate d_i. Taking into account the entire cascade, the overall false positive rate F and detection rate D is then given by:

$$F = \prod_{i=1}^{K} f_i \qquad D = \prod_{i=1}^{K} d_i \tag{4.3}$$

For example, if a 20 stage detector is trained such that at each stage 50% of the non-object patterns are eliminated (target false positive rate) while falsely eliminating only 0.1% of the object patterns (target detection rate) then the expected overall detection rate is $0.999^{20} \approx 0.98$ with a false positive rate of $0.5^{20} \approx 0.9 * 10^{-6}$. Ultimately, the desired number of stages, the target false positive rate and the target detection rate per stage allow to trade-off accuracy and speed of the resulting classifier. This also explains the different numbers of features in the first stage for the object-centered detector and for the local context detector as shown in figure 4.11.

Individual stages are trained using boosting which combines a set of "weak learners" into a powerful committee ("strong classifier"). In this case a weak learner is equivalent to one specific feature and a (automatically learned) threshold on its value. Each round of boosting selects the weak learner (i.e. feature type and threshold) that best classifies the weighted training set. The first boosting round assumes a uniform weighting of the training data while successive stages assign higher weights to misclassified training instances. This lets the algorithm focus on the "hard" cases in successive rounds. The different boosting variants Discrete, Real and Gentle AdaBoost offered by Lienhart's framework mainly differ in how they reweigh the training data in each round. For details the reader is referred to [Freund and Schapire 1996]. It has been empirically shown in [Lienhart *et al.* 2003a] that the Gentle Adaboost variant outperforms Discrete and Real Adaboost for face detection tasks both in accuracy and speed. Thus Gentle Adaboost (GAB) has been adopted throughout this thesis.

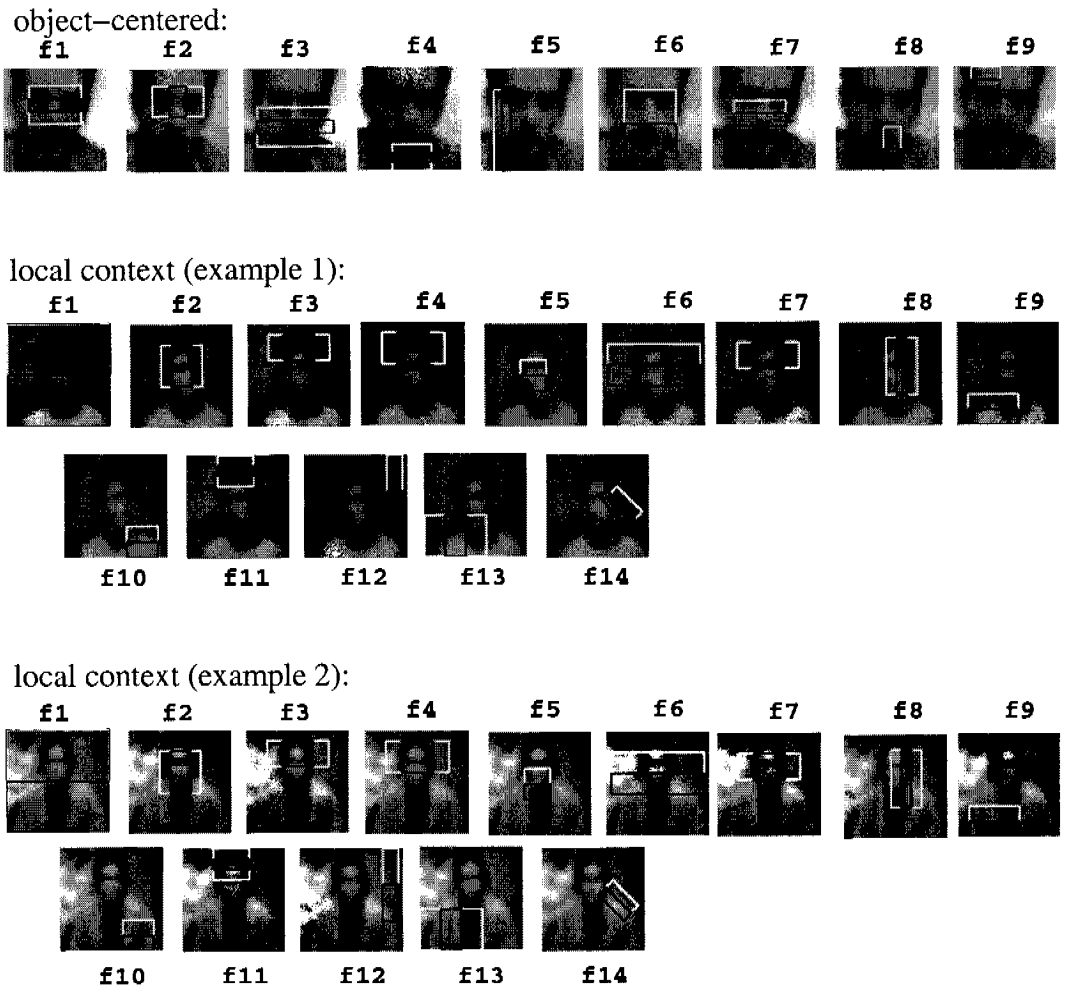

***Figure 4.11**: Automatically selected features (first classifier stage) for the object-centered face detector (9 features overlaid on a random training instance, top row) and the local context detector (14 features overlayed on two different training instances), bottom two rows – the number of features is automatically determined by the learning algorithm). In the local context case the first feature (f1) extends over the entire patch, thus making use of information that is not available in the object-centered case. In addition, features f9 and f10 as well as feature f14 capture the left and right shoulder to help in the face detection task.*

Parameter	Loc.Context	Obj-centered
Positive examples	960	5000
Negative examples	1232	3000
Stages	21	25
Min hit rate	0.995000	N/A
Max false alarm rate	0.500000	N/A
Width	20	24
Height	20	24

***Table 4.1**: Comparison of training parameters for the local context detector and the object-centered detector. An optimized and pre-built detector of Lienhart et al was used here for which not all parameters have been reported (N/A in the table).*

Finally, we summarize the most important training parameters such as the type and number of training instances, the target rates and the detection window size. About 1000 local context examples were gathered from the world wide web and from private photo collections. Only frontal views have been used for training and instances have been roughly aligned. Each positive example is scaled to 20×20 pixels. For gathering negative examples a subset of the WuArchive[4] image database has been used that did not contain any people. Similar to the training of the Schneiderman-Kanade detector, these images are repeatedly scanned by the learning algorithm to search for object-like patterns ("bootstrapping"). Training the local context detector was carried out on a 1GHz Pentium III machine with one gigabyte RAM. The training was stopped after 21 stages had been computed which took about 48 hours (the time for computing additional stages increases rapidly). Table 4.3.2 compares the training parameters of the local context detector to Lienhart's object-centered face detector -- which is also used in the following experiments.

4.3.3 Results on the FGNET video conference data

To understand the relevance of local context, several experiments have been carried out on PETS video sequences, namely the FGNET video conference data from PETS 2003 and the parking lot sequence from PETS 2000. Both data sets are disjoint from the data used for training. From the FGNET video conference data every 100th frame from sequences A, B, and D (cameras 1 and 2) is used in the following experiments[5]. This results in a total of 502 frames containing 1160 faces, 160 of which are profiles (about 14%). Each frame has a resolution of 720×576 pixels and

[4] http://wuarchive.wustl.edu/

[5] a similar subset was used in [Cristinacce and Cootes 2003]. However, in this paper we have included frames 21900-22300 as well

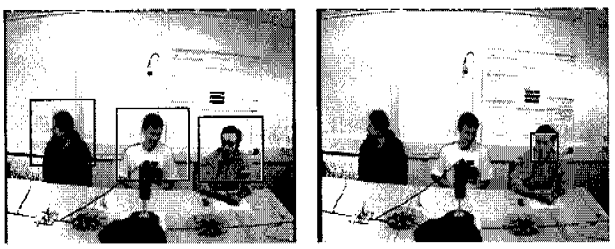

***Figure 4.12**: FGNET sequence, face pose changes: The left plot shows detections of the local context of faces while the right plot shows the output of the object-centered detector. The latter misses two faces because it is restricted to frontal view detection. While an additional specialized side-view detector could be trained (assuming the object-centered paradigm) this would at least require to gather a large amount of additional training data of profile views. Contrastingly, the local context approach does not require such specialized training and is robust to face pose changes.*

faces are about 48×64 pixels large. The sequences show a conference room across either side of a desk with people occasionally entering and leaving the room. The left plot in figure 4.13 shows the face detection performance on the FGNET data set in terms of the ROC curve. The percentage of retrieved faces is given on the vertical axis and the number of false positives per frame is shown on the horizontal axis. Points on the curve are sampled by varying the detection threshold b of the final stage in the detector cascade:

$$H = sign\left[\sum_{i=1}^{K} h_i + b\right] \tag{4.4}$$

where H is the output of the final classifier, and h_i denotes the individual stages. The cascade is successively cropped in order to yield additional detections. Both the performance of the object-centered and the local context detector are shown. For the object-centered version the face detector by Lienhart et al. has been used[6]. The detector has been shown to yield excellent performance comparable to the Schneiderman-Kanade detector [Lienhart *et al.* 2003a]. As can be seen in the left plot of figure 4.13 at 5 false alarms the object-centered detector retrieves 80% of the faces and the curve flattens from thereon. Contrastingly, the local context detector yields 95% of the faces at the same level of false alarms. This can be explained by profile views of faces contained in the data which are not detected by the object-centered detector used here. The local context, however, is not affected by face pose changes and can thus detect both frontal and side-views. An example frame containing side-views is shown in figure 4.12 also showing the outputs of both

[6]this face detector is now part of the Open Computer Vision Library, www.sourceforge.net

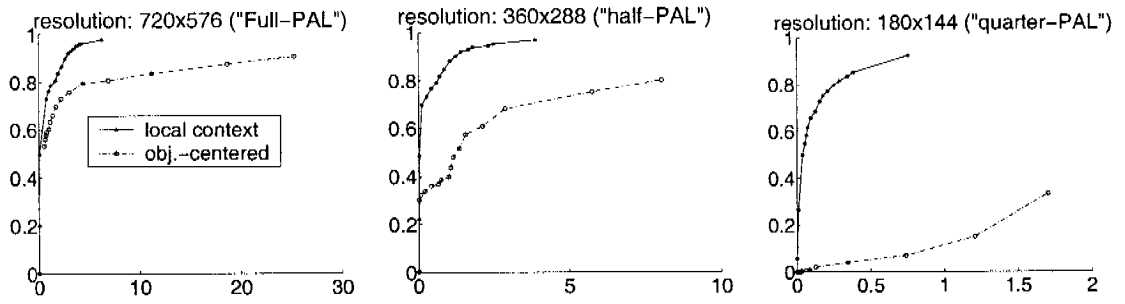

***Figure 4.13**: Detection accuracy on the FGNET data set. Each plot shows the percentage of detected faces (vertical) vs. the number of false positives per frame (horizontal). The ROC curves show the performance of the object-centered detector and the local context detector. Note that the frame resolution is decreased from the left plot to the right plot. At the original resolution shown on the right side, the local context detector dominates already because it is more robust to face pose changes. At lower frame resolutions (middle plot and right side plot) facial details deteriorate and the object-centered approach becomes unreliable. The local context on the other hand is not affected. Since the local context detector can operate robustly at very low frame resolutions it actually runs 15 times faster than the traditional object-centered approach at the same level of accuracy.*

detectors. In the object-centered approach one would have to separately collect profile instances and train a specialized detector in order to achieve a comparable performance level. However, it has also been found in [Schneiderman and Kanade 2000, Z.Q. Zhang and Zhang 2002] that profile detection generally tends to be more error-prone because important discriminant cues (such as the eye-to-eye symmetry in frontals) are missing. The local context detector overcomes these difficulties successfully.

Another concern is the available image resolution which has already been shown to be an important parameter when comparing object-centered and contextual cues (section 4.2.3). To examine effects of resolution degradation for this video sequence, each frame is downsampled from the original resolution (720×576 which approximately corresponds to PAL resolution) to 360×288 ("half-PAL") and to 180×144 pixels ("quarter-PAL"). This means, the available face resolution decreases from 48×64 to 24×32 and to 12×16 pixels approximately. Figure 4.14 shows an example frame where each row corresponds to a different resolution (all frames have then been resized for visualization purposes). The left column shows detections based on local context, the right column is from the object-centered approach. At half-PAL resolution (middle row) the object-centered detector apparently becomes less tolerant to variations in facial appearance, in this case caused by slight pose changes of the middle and right person. As a result it fails to detect these faces at lower

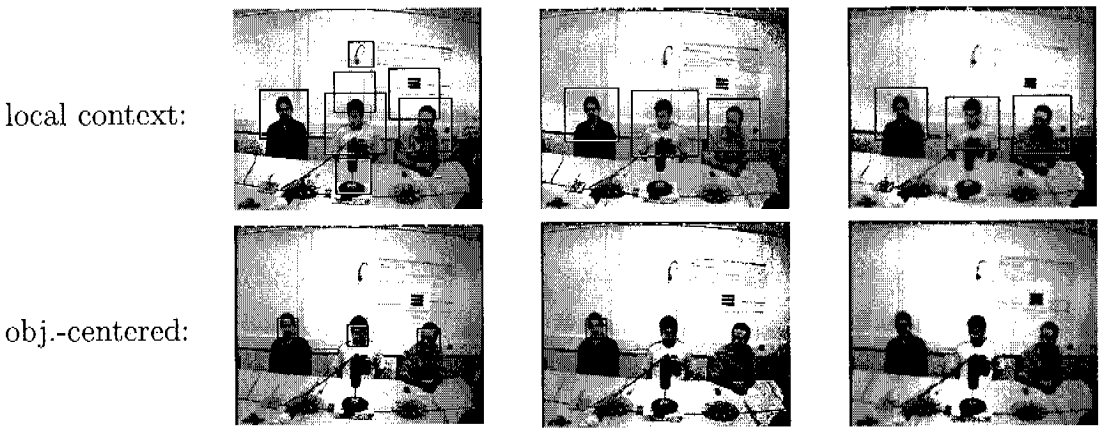

***Figure 4.14**: FGNET sequence, resolution changes: Each row of this figure corresponds to a different frame resolution (frame resolution decreases from the top to the bottom row). The images have been rescaled to the same size only for illustration purposes. The left column shows detections based on local context, the right column is from the object-centered approach. This example illustrates that as facial details degrade the object-centered approach misses actual faces. The local context cue is much less affected by resolution changes. It consistently retrieves all three faces at all tested resolutions, while the object-centered approach does so only for the highest frame resolution.*

resolutions. The local context detector on the other hand does not rely on facial details alone and can still locate all faces successfully. The situation aggravates for the object-centered approach as resolution is further decreased. At quarter-PAL resolution it does not return any detections in this example while the local context approach again detects all faces.

A quantitative account of this experiment is given by the plots in the middle and to the right of figure 4.13 showing the ROC curves of both detectors when applied to the downsampled data sets. At half-PAL resolution (middle plot) the object-centered detector yields about 70% of the faces at 5 false alarms which is a 15% drop compared to the full resolution. It is directly affected by the decrease in available facial detail. Contrastingly, the local context detector's performance remains stable at 95% given the same number of false alarms. This effect becomes even stronger at quarter-PAL resolution. The corresponding ROC is shown to the right in the same figure. In the quarter-PAL case the object centered approach detects only 10% at 1 false alarm per frame while the local context detector succeeds for more than 90% of the faces. Overall the local context detector provides improvements in detection rates by 15%, 25% and 80% at corresponding levels of false alarms.

The possibility to robustly operate at low resolutions can provide a significant speed-up in face search. For illustration, consider the case where we want to obtain 80%

of the faces with less than 5 false positives per frame. Using the face detector this is only possible at the highest frame resolution (full-PAL), where processing of a single frame takes 1.2 seconds. Contrastingly, the local context detector achieves this accuracy already at the quarter-PAL resolution within 0.08 seconds per frame. This corresponds to a 15-fold speedup. It must be emphasized here that the local context detector was not systematically optimized (e.g. by testing different training parameters) and thus the reported results could probably be further improved.

4.3.4 Results on Outdoor Surveillance Data

For investigating their suitability to surveillance both detectors were applied to the parking lot test sequence of PETS 2000. This video sequence shows a parking lot from a single static camera looking from above. The video shows cars and people entering and exiting the scene. Every 10th frame was used which results in 146 frames containing 210 faces. Each frame has a resolution of 756×576 pixels. Faces are as small as 12×14 pixels. Figure 4.15 shows an example frame of the sequence

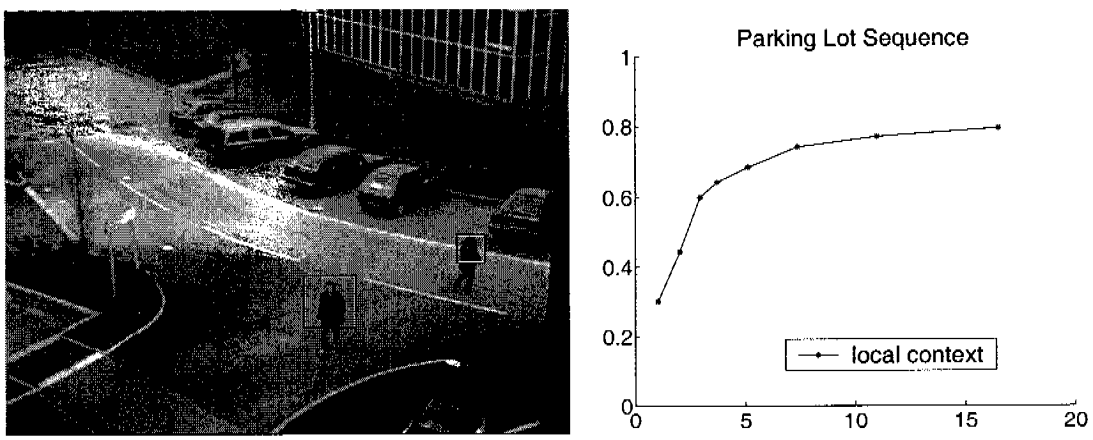

Figure 4.15: *In the parking lot sequence at least 1'000'000 candidate subregions per frame (example frame shown on the left) had to be correctly rejected. The situation is much more demanding than the FGNET scenarios because of the perspective (downward looking camera) and the background clutter. The ROC curve (right) shows the performance of the local context detector. The object-centered approach fails because faces are too small in this sequence, so its ROC is not shown. The local context detector retrieves about 75% of the actual faces at 7 false alarms per frame. This clearly indicates that the local context detector goes beyond the capabilities of object-centered face detectors.*

with detections of the local context detector (left side in the figure). The situation is much more difficult than in the video conference setting: the background is more

complex which inevitably leads to more false alarms. Moreover, the perspective is difficult as the camera is looking downward and people occasionally turn their upper bodies to the side. The maximum total area covered by faces is only about 0.1% per frame compared to about 2% in the indoor sequence. This means that at least 1'000'000 candidate subregions per frame have to be correctly rejected to yield an acceptable false alarm rate. Note that the legs of people in this sequence are sometimes occluded, for instance when they walk in between parking cars. This makes the scenario difficult for pedestrian detection approaches such as [Papageorgiou *et al.*]. However, their upper bodies are permanently visible which effectively allows local context detection.

The ROC curve for the local context detector is shown on the right side of figure 4.15. The object-centered approach fails completely on this sequence: it would require about twice the available image resolution to detect any faces. Hence only the ROC of the local context detector is visible here. As can be seen, it retrieves about 75% of the actual faces at 7 false alarms per frame. This already shows that the local context detector goes beyond the capabilities of object-centered face detectors. Moreover, in a surveillance application one could further reduce the number of false alarms for example by using background subtraction ([Cristinacce and Cootes 2003]).

4.4 Conclusion

It has been shown by quantitative and qualitative analyses that the detection of the local context of faces in grayscale images is feasible. This is in contrast to the traditional object-centered approach to face detection where the role of local context has so far been neglected. However, experimental results indicate that using local context yields correct detections that are beyond the scope of the classical object-centered approach and even beyond those of less specific cues such as skin detectors. This holds not only for low resolution cases but also for difficult poses, occlusion and difficult lighting conditions. Unlike the skin detector the local context detector does not require color which makes it widely applicable.

The real-time implementation based on the Viola-Jones framework was evaluated in a video conferencing and in a surveillance setting. It has been shown by ROC analysis that the local context cue consistently outperforms the competing object-centered face detector. This is mainly due to its greater robustness to pose changes, to variation in individual appearance as well as to low image resolutions. Robust operation at low resolutions not only speeds up the search process but is also of particular interest for surveillance where close up shots of people are often not available. The analysis of the outdoor sequence shows that the local context detector

goes beyond the capabilities of object-centered face detectors and correctly infers the locations of human faces as small as 9×12 pixels.

5

From Local Context to Scaleparts

As has been shown in the previous chapter, faces can be detected indirectly by detecting their local context. The particular aspect of local context that we chose to model was a face's close vicinity which contains the upper body. In particular it has been shown (section 4.2.3 of chapter 4) that local context is useful for face detection in low resolution images.

In this chapter, we generalize the local context concept to what we shall call *scaleparts*. Scaleparts are object parts suiting different levels of available image detail ("scales"). We will develop and evaluate this idea for face detection first, and show its applicability to other object classes – in this case cars – in chapter 7.

Section 5.1 introduces the general idea behind scaleparts and scalepart-based object detection. Then in section 5.2 we train two new detectors using AdaBoost: for a person's lower body and for the full body silhouette. Analogous to the previous chapter, the promise is that these body parts will allow for indirectly detecting faces at even lower resolutions.

Ultimately, the promise is that by combining detectors for face, upper, lower and full body one can reliably detect faces over a wide range of scales. After analyzing the individual performance of these four detectors in section 5.3 and a brief discussion in section 5.4 their actual combination is covered in chapter 6.

5.1 Scalepart-based Object Detection

The concept of "scaleparts" emerges from a straight-forward generalization of the local context idea of chapter 4. In the following section 5.1.1 we describe the scalepart concept by considering face detection as a case study. Then, in section 5.1.2 the scaleparts approach is described more generally and with respect to other types of target objects.

5.1.1 Body Scaleparts for Contextual Face Detection

In chapter 4 the particular aspect of local context that we chose to model was a face's close vicinity which contains the upper body. The key insight is that upper body detection succeeds at lower image resolutions than traditional face detection. This is because of the difference in features used: face detection mostly relies on the inner facial features such as eyes, mouth and nose which become indiscernible at lower resolutions. Upper body detection on the other hand mostly uses silhouette characteristics. As a natural extension we can build additional detectors for other body parts such as the lower body or the full body silhouette and use them in conjunction with the face and upper body detectors as cues for faces. If we look at an image of a human body over consecutive scales, e.g. through an image pyramid, then the face is best distinguishable at a higher pyramid level (large scale). Contrastingly, the full body silhouette remains distinguishable at lower pyramid levels (small scale). This is the original motivation behind the term "scaleparts" because these parts can accommodate different resolution situations or "scales". Figure 5.1 illustrates the

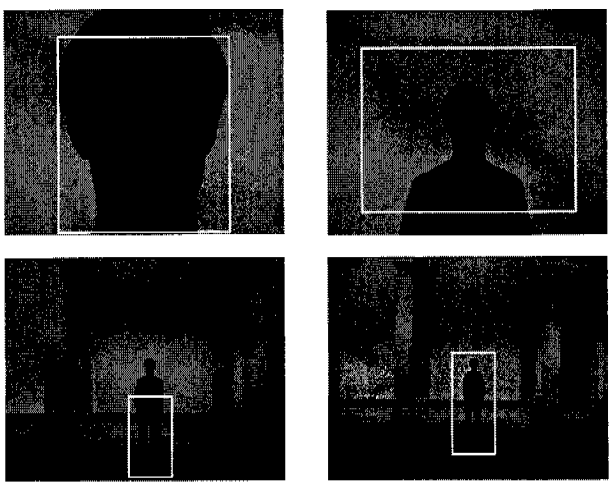

***Figure 5.1**: Set of body "scaleparts": object parts suited to different levels of image detail. For example, scalepart a) captures more facial detail than scalepart b) which in turn includes shoulder features. Likewise, scalepart c) captures leg details more accurately than scalepart d) which in turn encompasses the entire body silhouette.*

four different body scaleparts that we are going to use for face detection: face, upper body, lower body and full body. This specific choice of scaleparts, their positions, resolutions and extents is not uniquely defined but clearly motivated by the need to accomodate different levels of available image detail: the face detector captures more facial detail than the upper body detector which in turn includes shoulder features. Likewise, the lower body detector captures leg details more accurately than the full body detector which in turn encompasses the entire body silhouette. A particular

promise is that by combining all four scaleparts one obtains a detector which is robust to resolution changes. Intuitively, the face detector will be most effective in portrait situations, whereas the full body detector can contribute at full distance shots. The actual implementation of lower and full body detectors is described in section 5.3. The following section discusses the scaleparts approach within a more general setting.

5.1.2 General Scaleparts-based Object Detection

As stated previously, the specific choice of scaleparts, their positions, resolutions and extents is driven by the need to accomodate different levels of available image detail. For illustration figure 5.2 shows scalepart candidates for sideviews of motorbikes, jumbos and cows which we have chosen by hand. Parts are sorted from left to right according to their extent and resolution. In general we model parts that cover large extents of the object (e.g. the entire cow) at a relatively low resolution and smaller parts (e.g. the cow's head) at a relatively high resolution. Interestingly, such "intermediate complexity" parts were found to be most discriminant ("informative") for a given object class with respect to background clutter by [Ullman *et al.* 2002]. The superiority of intermediate complexity parts can be explained as the interplay of two factors: specificity and relative frequency. Considering faces as an example, a large face part can provide reliable indication of the presence of a face in an image, although the likelihood of encountering such a part in a novel face image is low (because of appearance variation within the face class). Consequently, the information carried by such a part with respect to the class is limited. A smaller part has a higher likelihood of appearing in different face images, but the likelihood of its presence in non-face images is also higher.

From a machine learning point of view "intermediate complexity" parts result in an efficient use of a fixed and limited number of training data: a higher resolution model might be theoretically better but will require more training data. Conversely, a lower resolution model will potentially omit discriminant and therefore important part details. Ideally, scaleparts themselves and their number would emerge directly from the training data using an appropriate learning algorithm. However, the space of possible parts is vast: it contains (for all training instances) every possible part position, extent and resolution. This space can be reduced by only considering intermediate complexity parts as mentioned above. However, given the relatively large number of training instances and target resolutions that we seek to achieve, a greedy search within this reduced space still remains impractical. Therefore, we currently choose scaleparts manually as rectangular regions.

Following is a list of important terms related to the scalepart framework:

Scaleparts are object parts suiting different levels of image detail ("scales").

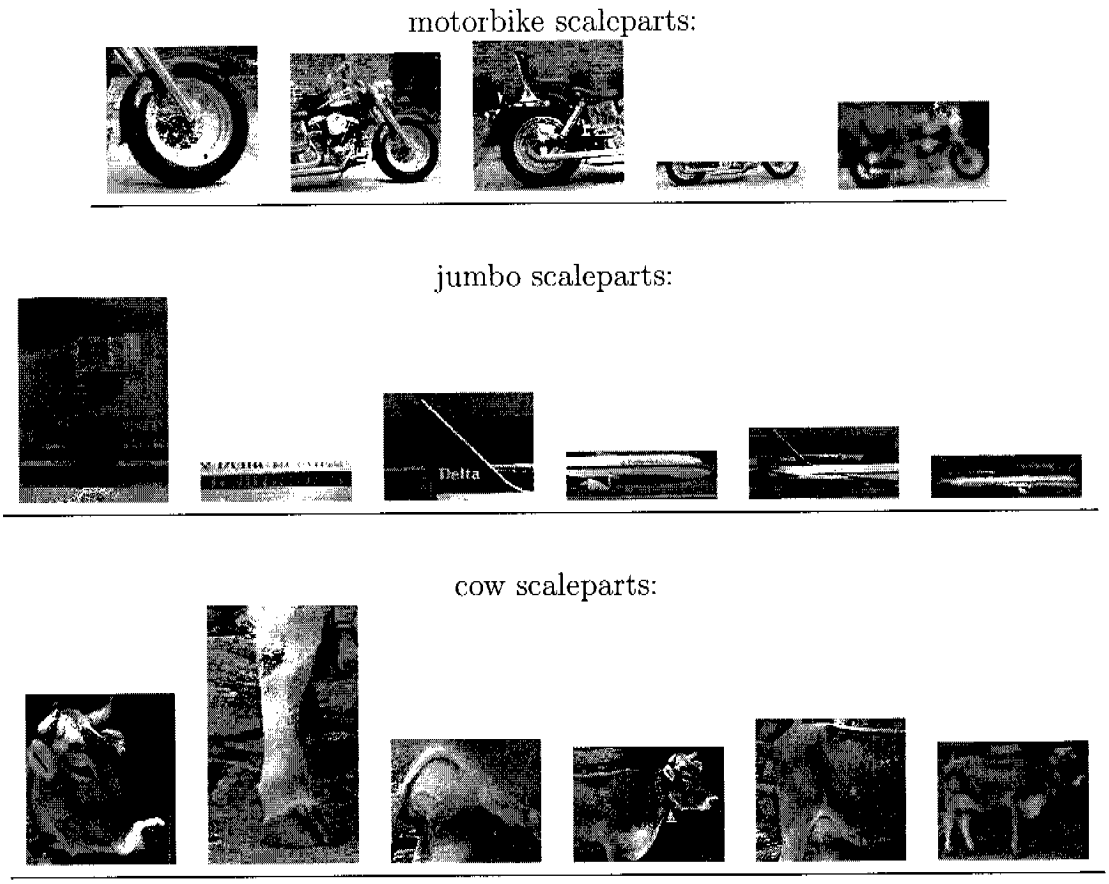

***Figure 5.2**: Scalepart candidates for sideviews of motorbikes, jumbos and cows. Parts are sorted from left to right according to their extent and resolution. The specific choice of parts is not uniquely defined but clearly motivated by the need to accomodate different levels of available image detail. In general, parts that cover small extents of the object are modelled at a relatively high resolution and larger parts at a relatively low resolution.*

Scalepart detectors are used to locate scaleparts within complex images. In the concrete implementation of this thesis this means running a detection window over all positions and scales of the input image, classifying each window independently as containing or not containing the specific scalepart. The two terms "scalepart" and "scalepart detector" are going to be used interchangeably.

Scalepart-based object detectors combine a set of scaleparts into a single object detector that can operate over a wide range of image resolutions. The combination mechanism is explained in chapter 6.

As a summary, figure 5.3 shows a three-step procedure for learning a scalepart-based object detector. In step one individual scaleparts are defined and according detectors

1. for all scaleparts:
   (a) initialize scalepart position, extent and resolution
   (b) learn scalepart detector using AdaBoost
2. construct Bayesian Net to capture scalepart dependencies
3. learn node conditional probability distributions

***Figure 5.3**: Learning a scalepart-based object detector. This chapter deals with step 1. Steps 2 and 3 are covered in the next chapter.*

are learned from data which is the topic of the following section. The second and third step in the procedure combine a set of scaleparts into one detector (chapter 6). The remainder of this chapter deals with the learning of two new body scalepart detectors and their evaluation on independent test data.

5.2 Learning Individual Scaleparts: Lower and Full Body

This section describes the training of two new body scaleparts: one is a detector for the lower body, the other for the full body – both seen from the front. Section 5.2.1 summarizes their learning as boosted classifier cascades. This classifier architecture has been successfully applied already in the previous chapter demonstrating both high accuracy in detection and fast execution. Then, section 5.2.2 presents an analysis of the features which have been automatically selected by the AdaBoost training procedure. The fact that the two detectors rely on different features is a first sign for their complementarity and motivates their combination.

5.2.1 Scalepart training

Both detectors are trained using the CBCL data set [1] which consists of 924 pedestrians. For training, these have been downsampled from their original resolution of 64 × 128 pixels: lower body patches are downsampled to a resolution of 19 × 23 and full body patches are resized to 14 × 28 pixels which maintains their original aspect ratio. Sample instances of the original set are depicted in figure 5.4 whereas

[1] http://www.ai.mit.edu/projects/cbcl/software-datasets/

Parameter	Full Body	Lower Body
Width	14	19
Height	28	23
Positive examples	924	924
Negative examples	2000	2000
Stages	30	27
Min hit rate	0.995000	0.995000
Max false alarm rate	0.500000	0.500000
Training Time	1 week	1 week

***Table 5.1**: AdaBoost learning parameters for the full and lower body detectors.*

figure 5.5 shows cropped lower body instances slightly enlarged. These images contain frontal and back views of people, but no side views. Also, these instances have been aligned with respect to their height and width. They have not been tightly segmented to ensure that the entire body silhouette is fully visible in all training instances. Especially at small image resolutions resolutions we expect the body silhouette to be an important and discriminant cue. The same holds for leg contours in the case of the lower body detector.

***Figure 5.4**: Examples of full body training instances from the original CBCL data set. These images contain frontal and back views of people, but no side views.*

***Figure 5.5**: Examples of lower body training instances cropped from the CBCL data set.*

These sample training instances also show the variation in appearance as concerns people's clothing, the form of the body, the actual position of legs, where they look and what they carry and so forth. As a final note, faces within this data are only 15 $\times$ 15 pixels which is too small for training an effective face detector. For training the

two scalepart detectors we have to strike a compromise between a limited amount of training data, the required level of detail that capture a specific part's characteristics and the expected AdaBoost training time. The latter is directly dependent on the training patch size which determines the number of integral image features that are examined within AdaBoost. During training, the cascade learning algorithm continuously monitors the discriminance level on the training set in terms of hit rate and false alarms. One way to optimize a scalepart's extent and resolution is to train separate detector candidates and choose the one which best performs on the training set (positive and negative training instances). If the training set is large then such a detector is likely to yield the best generalization performance as well, which can be explained by AdaBoost's relationship to margin maximization [Schapire *et al.* 1998]. If the training set is rather small then a high training set accuracy is more likely to indicate overfitting. With roughly 1000 training instances available, the best results were obtained with a training patch size of about 400 pixels square. Interestingly, this corresponds to the finding of [Lienhart *et al.* 2002a] in their empirical comparison of input sizes ranging from 18×18 to 32×32 pixels, despite the fact that this analysis used more training data (5000 instances) and was specific to face detectors (no other object types or parts). Table 5.1 summarizes the most important parameters for the AdaBoost training procedure analogous to table 4.3.2 of chapter 4. As can be seen, lower body patches are downsampled to a resolution of 19×23 and full body patches are resized to 14×28 pixels. As in the previous chapter, negative instances are drawn from the WuArchive image database. After a total training time of about 1 week on a 1GHz dual processor Pentium III machine (1 GB RAM – important for storing feature precomputations) 30 stages were completed for the full body detector and 27 stages for the lower body detector (trained on two separate machines). The features that AdaBoost has selected to differentiate full and lower body patterns from background patterns are examined in the following section.

5.2.2 Analysis of Features Selected by AdaBoost

Both full and lower body detectors are compared in this section by visualizing the first few learned features. Due to the detector's cascade structure these can be seen as the most important features. Figure 5.6 is a visualization of the full body detector's first two cascade stages (out of 30 stages) which hold 9 and 15 features, respectively. The feature visualization (black and white boxes) is analogous to chapter 4, figure 4.10. Feature "1a" refers to stage "1", first feature ("a"). When examining the occuring feature types, their positions and extents, it turns out that first stage features in the full body detector capture the entire person's silhouette (features 1a, 1c) and torso (features 1d, 1f, 1i). The second cascade stage also has a number of torso-related features (2i, 2l, 2m, 2o) but also seems to look at a more detailed

level such as arms (features 2b, 2c) and legs (2a, 2f, 2h). Interestingly, among these most important features there are none which are exclusively dedicated to the upper body or to the head. This justifies having a separate upper body detector which represents shoulder features at a higher resolution. In contrast to the full body

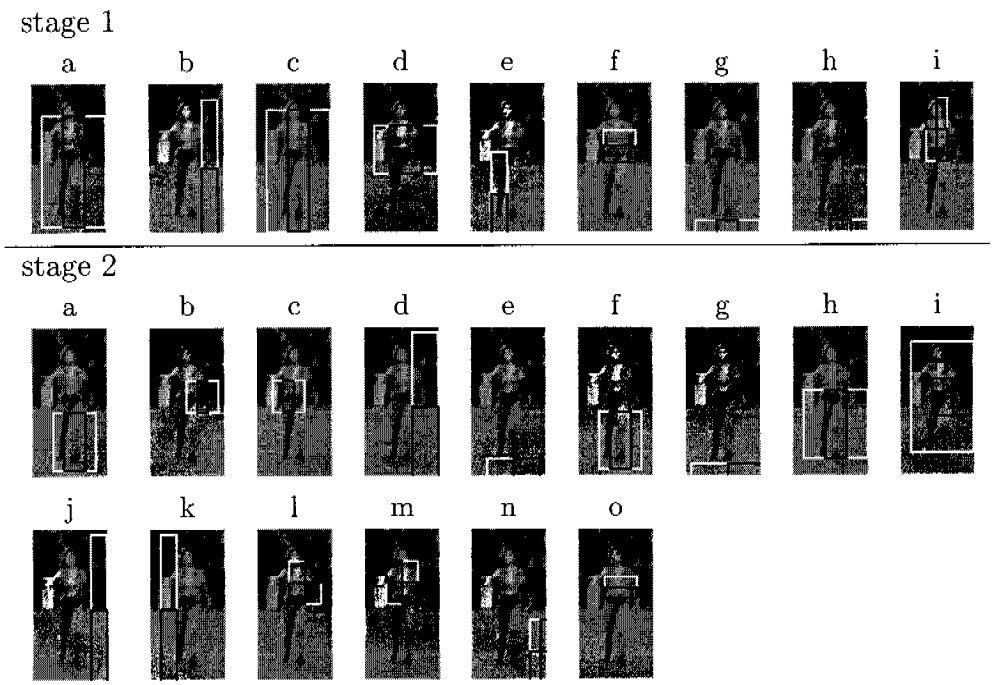

Figure 5.6: *Full body features selected by AdaBoost. The figure shows the first two out of 30 stages of the learned cascade. Many features sit on the silhouette and torso, but also on arms and legs. Interestingly, among these most important features there are none which are exclusively dedicated to the upper body or to the head. This justifies having a separate upper body detector which represents shoulder features at a higher resolution.*

detector, the lower body detector (figure 5.7) has learned to distinguish legs, that is, both legs seen together (1a, 1m) as well as individual legs (1b, 1c, 1k).

There is also a number of features in both detectors that sit on the background. They seem to capture the horizon line (1b, 2e, 2j, 2k in the full body detector) and a global illumination gradient (1g, 1h, 2e, 2g, 2m as well as 1o 1p 1q in the lower body detector, figure 5.7) which can be seen as contextual cues.

5.3 Scaleparts Detection Performance

In the following tests, a person has to be detected in indoor and outdoor video sequences. The test data is illustrated and described in the next section 5.3.1. This

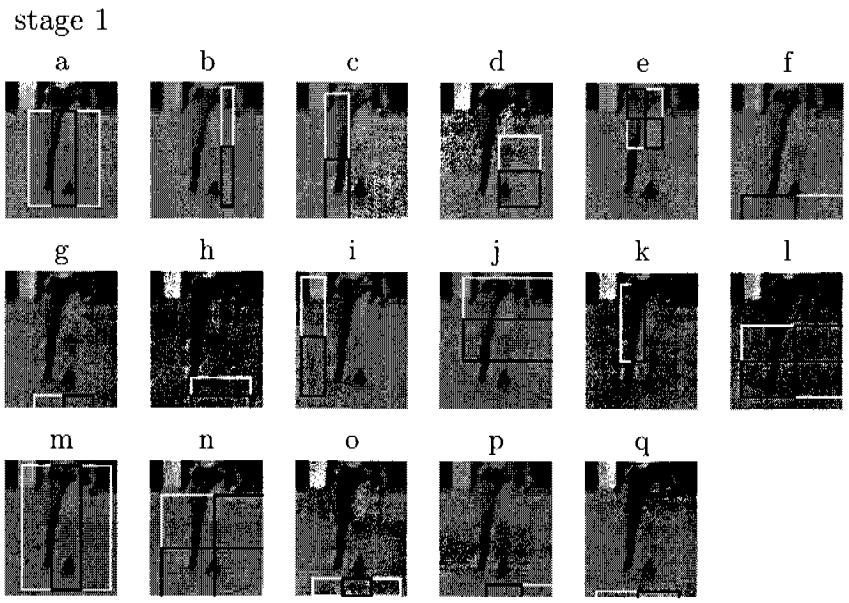

***Figure 5.7**: Lower body features selected by AdaBoost. The figure shows the first stage of the learned cascade. This detector allocates many more features to individual legs than the full body detector.*

is followed by a detection performance analysis for each individual body scalepart detector: full body, lower body, upper body and face. The analysis has two parts: In section 5.3.2 each detector is tested only on "compatible" video frames where the respective target is visible at sufficient resolution. For example, the face detector is evaluated on those frames where the face size is at least 20×20 pixels – which is the image patch size this detector has been trained with. This is only a lower bound: larger instances are detected by successive downscaling of the input image. Testing on compatible frames is suited to highlight the particular challenges of the two test sequences. To directly compare the different scaleparts, they are evaluated in a common task in section 5.3.3. The task is face detection which for the upper, lower and full body means that their detections are converted to face coordinates.

5.3.1 Indoor and Outdoor Test Sequence

Two independent test scenarios are considered in which a camera zooms in and out on a person's face. This specifically allows to analyze effects of resolution changes. In reality a similar situation occurs, for example, when a person is to be automatically spotted at a larger distance and then identified via face recognition.

Figures 5.8 and 5.9 illustrate the indoor and the outdoor scenario with a few representative frames and a visualization of the actual face height over the entire sequence.

Each frame has a resolution of 360 by 280 pixels (indoor) and 320 by 240 pixels (outdoor), respectively. Since each frame is scanned over all positions and scales, each detector examines about 25'000 subwindows per frame (at a downscaling factor of 1/1.2) – and ideally rejects all but a single one since there is exactly one face per frame. In the indoor sequence face heights range from 246 pixels in close-up shots (frame 42) to 16 pixels in distant shots (frame 111), likewise in the outdoor sequence the range is from 100 pixels (frame 57) to 8 pixels (frame 30).

The outdoor sequence is more difficult in several regards: first, the person's head is moving as can be seen in frames 30, 65 and 72. Second, the background is much more complex and cluttered, it contains a house front with many windows and a tree branch (upper left of frame 30), for example. The sequence is shorter with 76 frames versus 616 frames because it has been more coarsely subsampled from the original video stream.

5.3.2 Body Scalepart Detection on Compatible Frames

This section compares the performance on the two sequences for each individual scalepart detector. Figures 5.10, 5.11, 5.12 and 5.13 show ROC curves with the absolute number of false alarms on the x-axis and the detection rate on the y-axis. Each figure is dedicated to one of the four body scalepart detectors and shows two ROC plots (for indoor and outdoor test frames). The ROC curves evaluate the performance on compatible frames where the respective target is visible at sufficient resolution. This means each scalepart detector has to fulfill its own specific task on a specific subset of video frames. For example, the face detector is evaluated on those frames where the face size is at least 20×20 pixels – which is the image patch size this detector has been trained with. Likewise the full body detector is evaluated only on frames that actually contain the entire body at a resolution of at least 14×28 pixels. Thus, ground truth is specific for each scalepart and defined analogous to figure 5.1. The number of compatible frames is printed in the lower right corner of each ROC plot. Within one figure we can now compare a detector's performance on the indoor sequence (left) versus the outdoor sequence (right). As a common reference point each plot also shows a dashed vertical line which correponds to a false alarm level of 1 false alarm every 10th frame. For example, in the left plot of figure 5.10 there are 367 compatible indoor frames on which the face detector has been tested. With 1 false alarm every 10th frame this amounts to 37 false alarms (rounded from 36.7) which is the reference point indicated by the dashed vertical line.

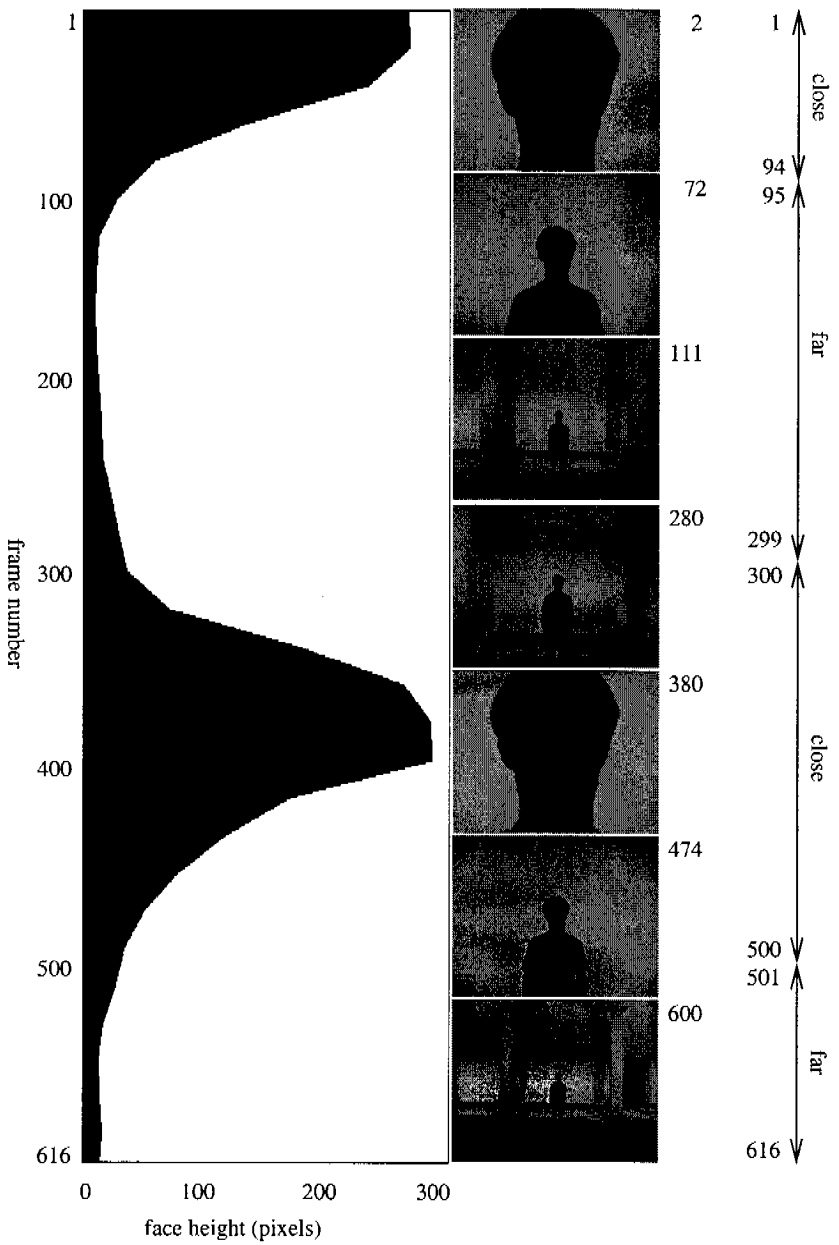

***Figure 5.8**: Indoor test sequence. Each of the 616 frames is 360 × 280 pixels and contains one face.*

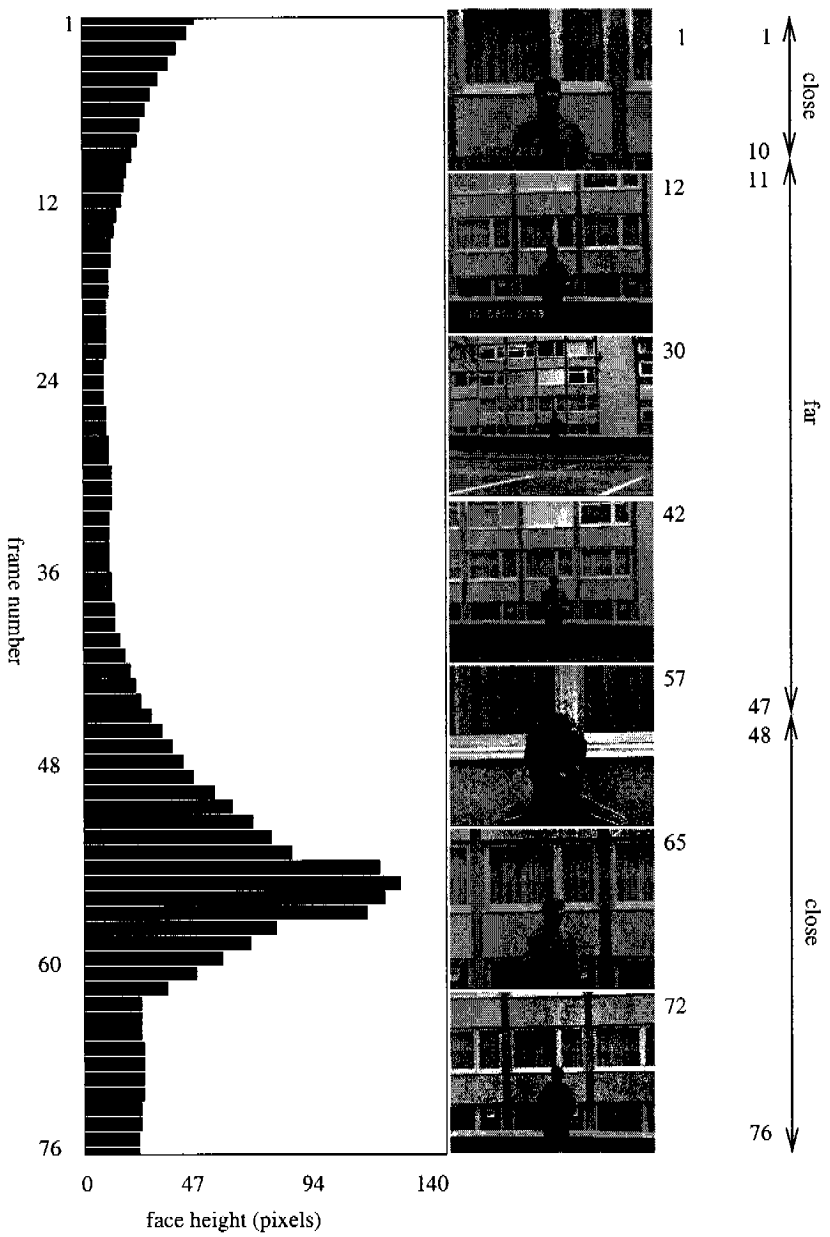

***Figure 5.9**: Outdoor test sequence. Each of the 76 frames is 320 × 240 pixels and contains one face. The video sequence has been subsampled more coarsely than the indoor sequence and also shows much more clutter.*

At this reference point the face detector[2] (figure 5.10) reaches a 90% detection rate indoors whereas it returns only about 40% in the outdoor sequence. This difference is because of head movement in the outdoor sequence and the face detector's restriction to frontal faces. Note again that the remaining cues are invariant to head movement. Overall the ROC curves are quite flat and stay at their 90% and 40% levels continuously almost from their beginning: at zero false positives the face detector still finds 87% of all faces indoors and 38% outdoors.

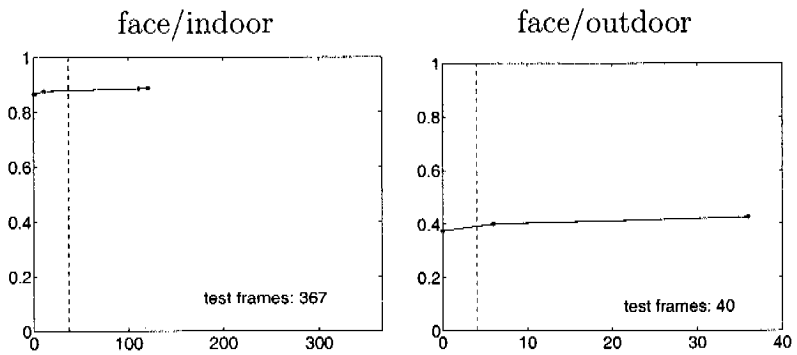

***Figure 5.10**: ROC performance of the face detector on frames of the indoor and outdoor sequence in which the face is sufficiently large ("compatible frames"). The number of test frames is shown in the bottom right corner of the plot. The dashed vertical line serves as a reference point for comparisons and corresponds to 1 false alarm every 10th frame. The curve shows a much better performance indoors because in these frames the face is always in a frontal pose. In the outdoor sequence, however, the face turns occasionally (head movement) which leads to missed detections.*

Figure 5.11 shows ROC curves for the upper body detector. At the reference point (dashed line) the upper body detector reaches 90% indoors but only 80% outdoors. The indoor curve raises steeply from 85% at zero false positives to almost 100% at 214 false positives which corresponds to one false positive every other frame. In the outdoor case the upper body detector starts from about 72% and raises slowly to a detection rate of 90% at 40 false alarms (which in this case corresponds to about 6 false alarms per 10 frames). Even at 1 false alarm per frame the outdoor curve does not increase beyond 90%. This corresponds to the intuitive effect of the more complex background which makes additional correct detections more costly resulting in a flatter ROC curve.

For the same reason the outdoor ROC curves for the lower and full body detectors in figures 5.12 and 5.13 are flatter than the indoor curves. This highlights the difficulty of the outdoor dataset. In case of the lower body detector the performance at the

[2]This is Lienhart's best-performing 20×20 face detector [Lienhart *et al.* 2003a].

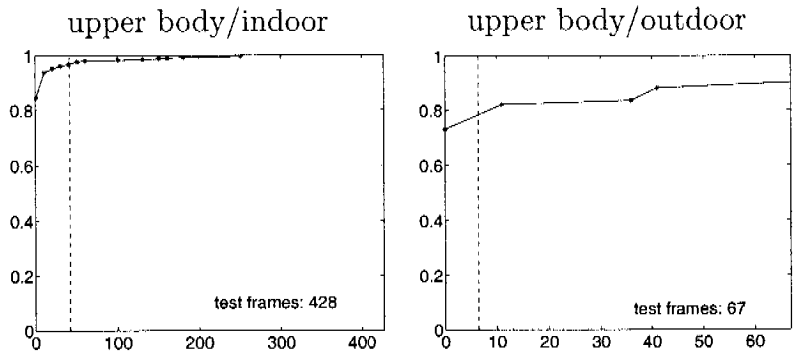

Figure 5.11: *ROC performance of the upper body detector indoors (left) and outdoors (right). Unlike the face detector the upper body detector is not affected by head movements which results in high detection rates for both sequences. However, the indoor ROC curve is much steeper than the outdoor curve. This corresponds to the intuitive effect of the more complex background in the outdoor sequence where higher detection rates come at a higher price (false alarms).*

reference point is similar for both detectors and around 92%. The curves, however, are rather different: in the indoor sequence the detection rate rises steeply from 81% at zero false alarms and attains a 100% detection rate at about 1 false alarm every 4th frame. In case of the outdoor sequence the detection rate is almost constantly at the 92% level from the beginning and stays there even when admitting 1 false alarm in every frame.

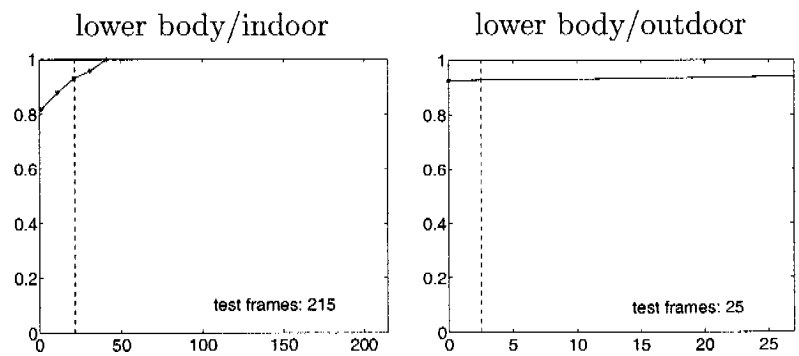

Figure 5.12: *The lower body detection rate on compatible frames is at 92% at 1 false alarm every 10th frame (dashed vertical line in each plot). This detection rate is roughly constant in the outdoor case. In the indoor case the detection rate increases to 100% at about 1 false alarm every 4th frame.*

For the full body detector as displayed in figure 5.13 the detection rate at the reference point (dashed line, corresponding to 1 false alarm every 10th frame) is better outdoors (72%) than indoors (40%). However, the indoor curve again rises

much faster than the outdoor curve: it starts at 16% (zero false alarms) and reaches 90% at 110 false alarms – which corresponds to 1 false alarm every other frame. The outdoor ROC curve on the other hand starts at a higher detection rate of 68% but reaches 90% only if allowing almost one false alarm in every frame.

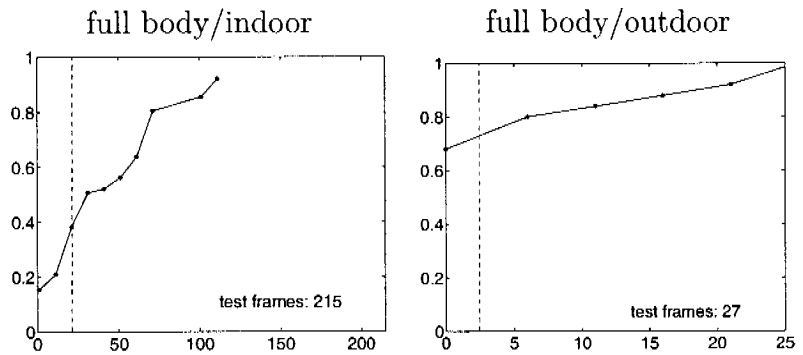

***Figure 5.13**: Compared to the other detectors the full body detector reaches high detection rates (90%) only at a relatively high number of false alarms – about 1 false alarm every other frame. The flatter curve to the right is again due to the more complex background of the outdoor sequence which attracts more false positive detections.*

In summary face, upper and lower detectors reach about 80% to 90% in detection rates at a single false alarm every 10th frame. Only the full body detector achieves a lower level of discriminance than the other scaleparts which for the indoor sequence means 1 false alarm every other frame for a 90% detection rate. The analysis also highlight the increased difficulty of the outdoor sequence caused by head movement and the more complex background. The latter also accounts for the flatter ROC curves. It is also noteworthy that subsets for lower body and full body are similar but not identical: both are visible at the same time in these sequences but the resolution requirement for the lower body is obviously tighter, that is, the full body detector also sees test frames that are too small for the lower body detector.

5.3.3 Body Scaleparts for Face Detection

This section directly compares the individual body scaleparts by applying them to face detection. More precisely, scalepart detectors are now applied over all positions and scales to locate faces of the indoor and outdoor test sequence. Unlike the previous section this section applies these detectors to *all* frames regardless of their resolution "compatibility". Also, any detections of the upper, lower or full body detectors are immediately converted into face coordinates. Figure 5.14 illustrates this conversion for full body and lower body detections which assumes a fixed relative

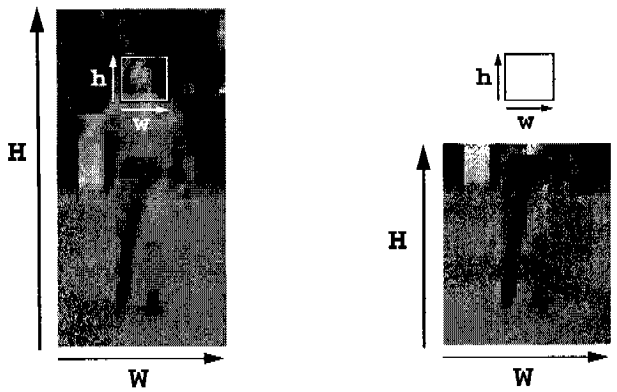

Figure 5.14: *Inferring the face position from full (left) and lower (right) body detections. Similar to chapter 4 we assume a fixed position of the person's face relative to the detection frame. For lower body detections the estimated face position actually lies* outside *the detection frame.*

position within the original detection frame. The procedure is analogous for upper body detections and has been already introduced in the previous chapter 4, section 4.1.

The detection performance for both sequences is shown in figure 5.15. The ROC plot on the left side shows the performance on the indoor sequence, the right plot shows the outdoor performance. Each plot shows four curves corresponding to the four detectors. The x-axis shows the absolute number of false alarms limited to 60 (indoor, left plot) and 8 (outdoor, right plot) which in both cases corresponds to about 10% of the sequences' lengths for easier comparison.

In case of the indoor sequence the best-performing face detection cue is the upper body which is then followed by the face cue, while lower and full body cues have similar but lower ROC curves. The upper body cue is so successful because it is more often visible at sufficient resolution than any other scalepart: the face detector is ideal for close-ups and the full and lower body detectors are well suited after fully zooming out – for everything in between the upper body works best. Its ROC curve is flat, stays around 65% in detection rate from the beginning and is shorter than the other curves[3]. The face detector curve is also flat and stays around 52% from the beginning. Lower and full body detectors on the other hand have slowly rising ROC curves: they rise from 2% (zero false positives) to about 30% (right border of the plot corresponding to about 1 false alarm every 10th frames). The lower body curve is consistently above the full body curve with a difference of 2% to 10%, but

[3]The upper body curve (left plot) would actually decrease at higher false alarm rates but due to the way the curve is computed only the monotonically increasing part is plotted. As a result, certain curves can be shorter than others. Curves decrease when correct detections become too inaccurate (see also [Cristinacce and Cootes 2003]).

this gap is gradually closing at higher false alarm levels.

In the outdoor sequence (right plot) the upper body cue also performs best, but the face cue is now only as good as lower and full body cues. This is mainly because the face detector cannot cope with the head movement within this sequence. The upper body's ROC curve slowly rises from 65% to 70% which is slightly better than indoors. All other three curves reach detection rates of about 20% to 22% – which is almost 10% less then indoors. This was to be expected given the more difficult sequence. Also, the full body's ROC curve is consistently above the lower body's curve whereas the opposite is true in the indoor sequence. The reason behind this is that the full body is detectable in more frames than the lower body because of its weaker resolution requirements. In other words, there are more "compatible" frames for full body than for lower body detection. As a final result of this quantitative analysis even the best single detector misses almost one third of the faces (detection rates are limited to some 65% or 70%) – this is even though a human observer will not have any problems to locate all the target faces at even the smallest resolution occuring here. However, the promise is that by combining all four cues one can do better.

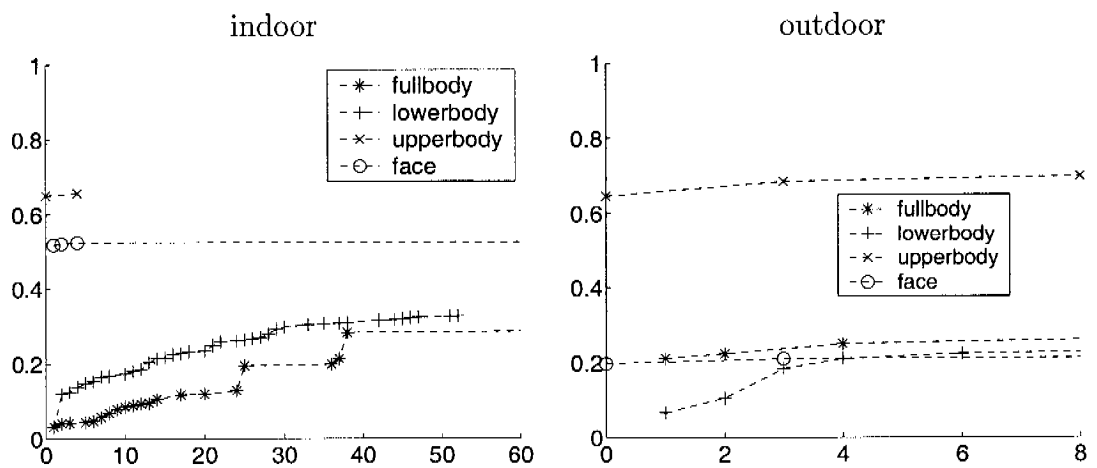

***Figure 5.15**: Individual Scalepart Performance for face detection on the indoor sequence (left plot) and outdoor sequence (right plot). Detection rates are shown on the vertical axis, the absolute number of false alarms is shown on the horizontal axis. No single cue can detect more than 70% of the faces at a precision level of 1 false alarm every 10th frame - which motivates their combination.*

5.4 Discussion

This chapter has introduced the concept of *scaleparts*. The scaleparts approach is motivated by the need to accomodate different levels of available image detail and can be seen as a generalization of the local context concept of the previous chapter. Accordingly, scaleparts are defined as object parts suiting different levels of available image detail – or "scales".

In addition to the upper body detector of the previous chapter a lower body and a full body detector are developed and evaluated. In this, one has to take into account the available limited amount of training data, the required level of detail that capture a specific part's characteristics and the expected AdaBoost training time. The latter is directly dependent on the training patch size which determines the number of integral image features that are examined within AdaBoost

Test results on indoor and outdoor situations demonstrate the complementarity of the developed body scaleparts with respect to different resolutions. The promise is that their combination will allow for "contextual" face detection robust to resolution changes. This is covered in the next chapter.

6

Scaleparts-based Face Detection

This chapter is concerned with the development of a scaleparts-based face detector. The detector combines the four body scaleparts of chapter 5 with the aim to overcome their individual limitations. For scalepart combination, this chapter employs Bayesian Networks which provide an elegant and efficient means for probabilistic cue fusion based on learning and inference.

Section 6.1 develops a Bayesian Network topology that implements the scaleparts-based face detector. Then, section 6.2 focuses on how to learn the conditional probability distributions in the observed network nodes. These distributions model the interactions between scaleparts taking into account their individual reliabilities. The combined face detector is tested in section 6.3 and compared to the performance of each individual cue. Conclusions are drawn in section 6.4.

6.1 Cue Combination with a Bayesian Network

Bayesian Networks graphically encode the joint probability distribution of a set of random variables. The graph theoretic side of Bayesian Networks provides both an intuitively appealing interface by which humans can model highly-interacting sets of variables as well as a data structure that lends itself naturally to the design of efficient learning and inference algorithms. A particular application of Bayesian Networks is cue fusion as used, for example, by [Rehg *et al.* 1999] whos work has been an important source of inspiration (see chapter 2, section 2.3 for additional related work on Bayesian Networks for cue fusion). Note that, despite the name, Bayesian networks do not necessarily imply a commitment to Bayesian methods; rather, they are so called because they use Bayes' rule for inference. The particular focus of this section is the development of a suitable network topology. Figure 6.1 shows an example topology with parent node V ("visible") and child nodes f ("face"), u ("upper body"), l ("lower body") and b ("full body") corresponding to the body scaleparts of the previous chapter. In this case, every node is a binary random

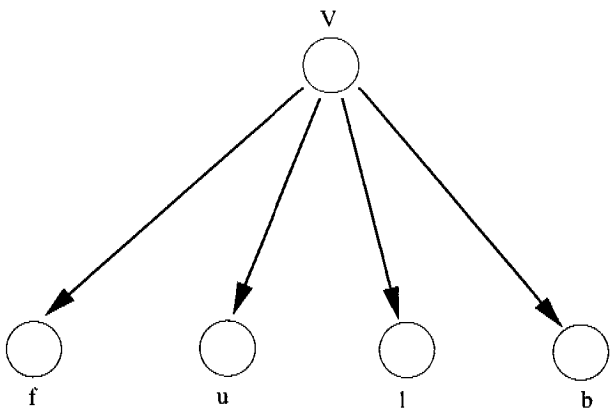

***Figure 6.1**: "Naive Bayes" topology with parent node V ("visible") and children f ("face"), u ("upper body"), l ("lower body") and b ("full body") which refer to the body scaleparts of chapter 5. In this special case, the observed nodes (shaded) are conditionally independent given the parent node.*

variable X_i (with $i = 1..5$) which can take upon the values 0 or 1. The values of nodes f, u, l and b are directly observed by applying the respective scalepart detector. To recall, each body scalepart is implemented as a boosted classifier cascade:

$$H = sign\left[\sum_{j=1}^{K} h_j + b\right] \tag{6.1}$$

where h_j denote the individual cascade stages and H is the classifier's binary output which in the original formulation corresponds to the sign of the expression and which is converted here to values 0 and 1:

$$X_i = \begin{cases} 1 & \text{iff} \quad \text{part } i \text{ detected} \\ \\ 0 & \text{else} \end{cases} \tag{6.2}$$

Ultimately, we are interested in the value of the parent node V which indicates the presence of a face given all evidence. The distribution over the possible values of V is inferred from the observed evidence and since we are querying V's value, V is also called a *query node*. This is in contrast to the *observable nodes* (f, u, l, b – shaded in figure 6.1) whos values are directly observed by evaluating the underlying body scaleparts.

The directed arcs from V to all other nodes imply that V *causes* observations f, u, l, and b. The causal interpretation of Bayesian nets is an important advantage because it lets one encode prior knowledge in a straight-forward manner. The graph structure also expresses conditional independence of certain variables. Specifically, the graph in figure 6.1 implies that all observed nodes are conditionally independent

given the parent node (and is therefore referred to as the Naive Bayes case). In a more general notation this corresponds to the following factorization:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|parents(X_i)) \tag{6.3}$$

If we take the logarithm of this equation then the product on the right side turns into a sum and we obtain the standard vote accumulation scheme with weighted votes:

$$\begin{aligned} \log P(X_1, \ldots, X_n) &= \log \textstyle\prod_{i=1}^{n} P(X_i|parents(X_i)) \\ &= \textstyle\sum_{i=1}^{n} \log P(X_i|parents(X_i)) \end{aligned} \tag{6.4}$$

The problem is, however, that this does not take into account dependencies among the observed evidence. For example, for face detection in small images there is a dependency between face and full body observations: a small image can either show a close-up shot of a face with all facial details visible or a more distant shot where the entire body is visible – but not both at the same time. In conclusion, a different network topology is required which is capable of modelling such dependencies among observed variables.

Such a network is depicted in figure 6.2. The network splits up the parent node into two separate nodes C ("close") and F ("far") which indicate the presence of a face in a close-up shot ($C = 1, F = 0$) or at a distance ($C = 0, F = 1$). This network is capable of accurately representing the mentioned cue dependencies: intuitively, a close-up view of a face may cause face and upper body observations which is expressed by the directed arc from C to F and from C to u. But neither lower nor full body are visible then. Conversely, a far-away face causes observations of a full and a lower body but the resolution situation does not allow to observe a face directly. The upper body, however, can possibly be observed in both situations. This causal dependency is reflected by the parent-child relationship, i.e. the network's arcs and their directions.

6.2 Learning and Inference

In order to infer the posterior distributions $P(F|f, u, l, b)$ and $P(C|f, u, l, b)$ which let us decide over the most probable values of C and F, we need to know the probability densities of each observed node conditioned on these two variables. These as well as the prior probabilities $P(C)$ and $P(F)$ are learned from data which is discussed in section 6.2.1. Then, section 6.2.2 explains how the classifier accumulates evidence from the individual scaleparts in order to infer a single, consistent classification ouput value.

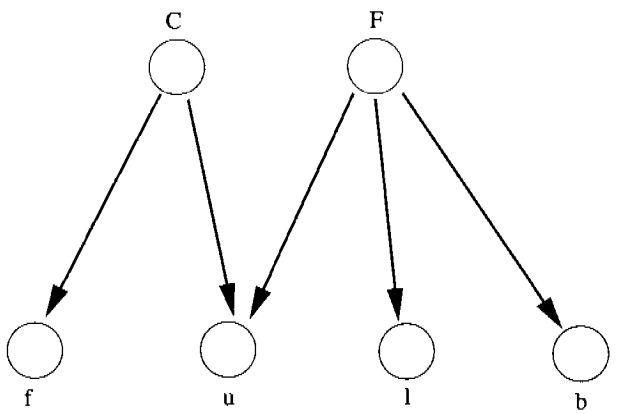

Figure 6.2: *Bayesian Net Topology for face detection with query nodes C and F to distinguish close-up and far shots of faces. According to this topology, a close-up face causes an observation of face and possibly upper body but neither lower nor full body are visible then. Conversely, a far-away face causes observations of a full and a lower body but the resolution situation does not allow to observe a face directly. The upper body, however, might be visible in both situations.*

6.2.1 Learning Conditional Probability Densities

In a discrete-node Bayesian Network the probability densities associated with each node can be efficiently represented using Conditional Probability Tables (CPTs) (equivalent to histograms, i.e. non-parametric density estimates). Since in our case all variables are binary we only have to estimate entries of small CPTs. As training data we use every second frame of the indoor sequence introduced in chapter 5, figure 5.8 which amounts to 308 training frames. In frames where the camera has zoomed out, the person's knees or feet become visible as in frame 280 of figure 5.8. These frames are labeled "far" $(C = 1, F = 0)$, all other frames are labeled "close" $(F = 0, C = 1)$. The entries of the CPTs (tables 6.1 to 6.4) are then learned from

C	$P(f = 0\|C)$	$P(f = 1\|C)$
0	0.8520	0.1480
1	0	1

Table 6.1: *The learned CPT for the face node in network 6.2*

the labeled data using maximum likelihood estimation:

$$\hat{\Theta}_{X_i|Pa(X_i)} = \hat{p}(X_i|Pa(X_i)) = \frac{N_{X_i,Pa(X_i)}}{N_{Pa(X_i)}} \tag{6.5}$$

Here, $\hat{\Theta}_{X_i|Pa(X_i)}$ denotes the table entry for the observed node X_i conditioned on its parents $Pa(X_i)$. The corresponding probability estimate $\hat{p}(X_i|Pa(X_i))$ is ob-

C	F	$P(u=0\|C,F)$	$P(u=1\|C,F)$
0	0	0.9951	0.0049
1	0	0.6653	0.3347
0	1	0.0187	0.9812
1	1	0.5000	0.5000

***Table 6.2**: The learned CPT for the upper body node in network 6.2*

F	$P(l=0\|F)$	$P(l=1\|F)$
0	0.7256	0.2744
1	0.3563	0.6438

***Table 6.3**: The learned CPT for the lower body node in network 6.2*

tained through frequency counting: $N_{X_i,Pa(X_i)}$ is the fraction of detections (or non-detections, respectively) of scalepart X_i for the possible values of its parents $Pa(X_i)$ In network 6.2 both query nodes C and F are root nodes. They are not associated with a CPT but instead with a prior probability which is also estimated from data:

$$\hat{\Theta} = \hat{p}(Q_i) = \frac{N_{Q_i}}{\sum_i N_{Q_i}} \tag{6.6}$$

with N_{Q_i} being the number of labels ("close" or "far") where query node $Q_i = 1$.

We can now examine the CPT entries more closely to see what has been learned. The probabilities $P(f=1|C=1)$ and $P(f=1|C=0)$ in table 6.1 correspond to estimates of the true positive and false positive rates of the face detector in a close-up situation. In a similar way, the other two probabilities are estimates of the true negative $P(f=0|C=0)$ and false negative rates $P(f=0|C=1)$. Basically, for close-ups, the learned numbers express a very high confidence in the face detector's output with a true positive rate of 1 and a false positive rate of 0.15.

The upper body CPT (table 6.2) is the only one with a fan-in (i.e., the number of incoming arcs/parents) greater than one. The CPT shows a much higher confidence in upper body detections at distant shots with $P(u=1|F=1,C=0)=0.98$ than in close-ups: $P(u=1|F=0,C=1)=0.34$. This is because in a relatively large number of training frames only the face itself is visible. The value pair $C=1$ and $F=1$ never occurs in the data because it is illegal: the semantics of both query nodes independently decide about a face's presence and the same face cannot be "close" ($C=1$) and "far" ($F=1$) at the same time. However, both query nodes can agree that there is no face at the examined location ($C=0$ and $F=0$). Lower and full body detectors (tables 6.3 and 6.4) attain lower true positive rates than the other two cues (0.62 and 0.46). It turns out that many detections overlap with the person but are not accurate enough to be counted as true positive.

F	$P(b=0\|F)$	$P(b=1\|F)$
0	0.8786	0.1214
1	0.5437	0.4562

***Table 6.4**: The learned CPT for the full body node in network 6.2*

Generally, for counting true positives and false alarms we adopt the matching rules of [Lienhart *et al.* 2003a]. A true positive is declared if and only if:

> the Euclidean distance between the center of a detected and actual target's center is less than $\omega \times$ targetwidth (the latter is referring to the true width of the target) and
>
> the width (i.e. size) of the detected target is within $\pm\delta \times$ targetwidth.

The parameters were left at $\omega = 0.6$ and $\delta = 0.6$ for all evaluations.

Relatively speaking, the face detector achieves the highest true positive rate followed by upper, lower and full body detectors. The lowest false positive rate is achieved by the upper body detector followed by face, full and lower body detectors.

6.2.2 Evidence Accumulation, Inference and Classification

In the implementation all scalepart detectors are applied independently over all positions and scales. The next step is to find sets of overlapping detections. Each such set forms a face hypothesis that contains one to four detections from the four employed body scaleparts (evidence accumulation).

Then, for each of these face hypotheses the posterior probabilities of both query nodes C and F are computed via Bayesian inference. We use the standard junction-tree inference algorithm [Cowell *et al.* 1999]. A combined face hypothesis is then assigned the resulting maximum posterior probability:

$$P(\text{FACE}) = max\{P(C), P(F)\} \tag{6.7}$$

For classifying a given face hypothesis as either face or non-face, the probability P(FACE) is thresholded. The analysis in the following section uses ROC curves to examine the detection accuracy at all possible thresholds.

6.3 Face Detection with the Combined Detector

This section empirically analyzes the performance of the learned Bayesian Network-based classifier. To this end, the classifier is applied to image frames not seen during

training. The first test sequence consists of the remaining 308 frames from the indoor sequence of chapter 4 (figure 5.8), the second test sequence consists of all 76 frames from the outdoor sequence (figure 5.9).

Section 6.3.1 is an ROC analysis of both test sets, comparing the Bayesian Network classifier of figure 6.2 to the naive Bayes network of figure 6.1 as well as to the four individual scalepart detectors. The next three sections investigate the components and mechanisms of the classifier at work: section 6.3.2 looks at the evidence nodes, examining which scaleparts contribute to correct detections. Section 6.3.3 provides a detailed analysis of the inferred query node values. At the core of this section is is a case-based analysis relating the inferred query node posteriors to different detection constellations. Certain constellations exhibit contradicting evidence which Bayesian Networks can "explain away" – a mechanism that is further illustrated in section 6.3.4.

6.3.1 Performance of the Scaleparts-based Face Detector

This section compares the Bayesian Network classifier of figure 6.2 to the Naive Bayes classifier of figure 6.1 and to the individual scalepart detectors. ROC curves for the indoor sequence are displayed in figure 6.3. The discrimination task has been made more difficult by including simulated false alarms, because the scalepart detectors hardly cause any false alarms in these images [1]. The "Bayesian Net" and "Naive Bayes" curves are computed by thresholding $P(\text{FACE})$ or $P(V = 1|f, u, l, b)$, respectively, for all possible thresholds. Since both classifiers operate on four binary observed nodes their value range is theoretically limited to 16 different posterior values. This makes the ROC curves look edgy. Smoother curves could be obtained by adopting continuous nodes or by smoothing posteriors over time using Dynamic Bayesian Networks (as explained in chapter 2, section 2.3.2). As a first result, scalepart combination in indoor and outdoor test frames (figure 6.4) is able to yield detection rates beyond 95% – compared to a maximum of 70% of the best individual scalepart detector (upper body). Generally, the Bayesian Network yields high detection rates with less false alarms than the Naive Bayes classifier. This is an effect of the difference in assumed (in)dependencies. For example, in the Naive Bayes case, an overlap of all four scaleparts results in the highest possible confidence score. The same constellation implies a case of contradicting evidence in the Bayesian network: the face scalepart is associated with "close" situations, and the full body scalepart with "far" situations resulting in a lower confidence score and better separability with respect to correct detections.

In the indoor case (figure 6.3) the best-performing scalepart detector (upper body)

[1] The false alarm situation is much more challenging in chapter 7 were an analogous ROC analysis is presented for car detection without simulated false positives.

reaches its maximum detection rate of 65% from the beginning at 0 false alarms. The Naive Bayes classifier reaches the same detection rate at 10 false alarms and at 45 false alarms its detection rate jumps beyond 95%. The Bayesian Network attains the same detection rates at fewer false alarms: 65% at 1 false alarm and beyond 98% at 33 false alarms.

In the outdoor sequence the best-performing scalepart detector's curve rises from 65% at 0 false alarms to 70% at 9 false alarms. The Naive Bayes classifier performs worse if only 9 false alarms or fewer are admitted, retrieving not more than 5% of the faces. At 12 false alarms it reaches a detection rate of 88%. Again, the Bayesian Network is able to reach high detection rates at low false alarm levels: at 0 false alarms 64% and at 3 false alarms 80% which already outperforms the upper body scalepart by more than 10% in detection rate.

It is worth noting that in about half of the test frames the face height is less than 20 pixels. This means that a traditional face detector that does not incorporate contextual information is extremely unlikely to achieve a detection rate on these test sets which is greater than 50% (the smallest sized face detector we are aware of is [Sung and Poggio 1994] with 19×19 pixels).

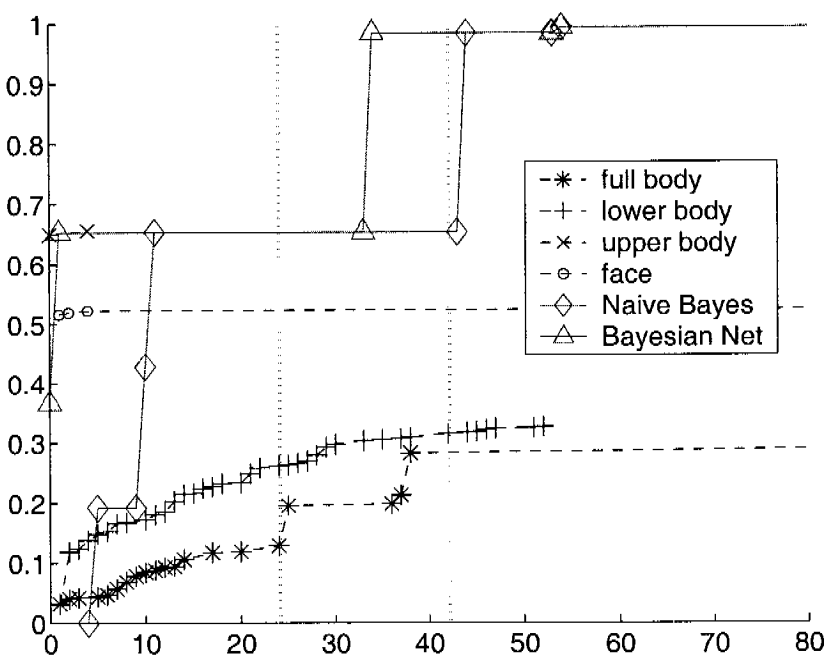

Figure 6.3: *Indoor test: ROC curves comparing the Bayesian Network of figure 6.2 to the Naive Bayes classifier of figure 6.1 and to the individual scalepart detectors. The Bayesian Network retrieves 98% of the faces with 33 false alarms which exceeds the detection rate of the best-performing individual scalepart (upper body) by more than 20%.*

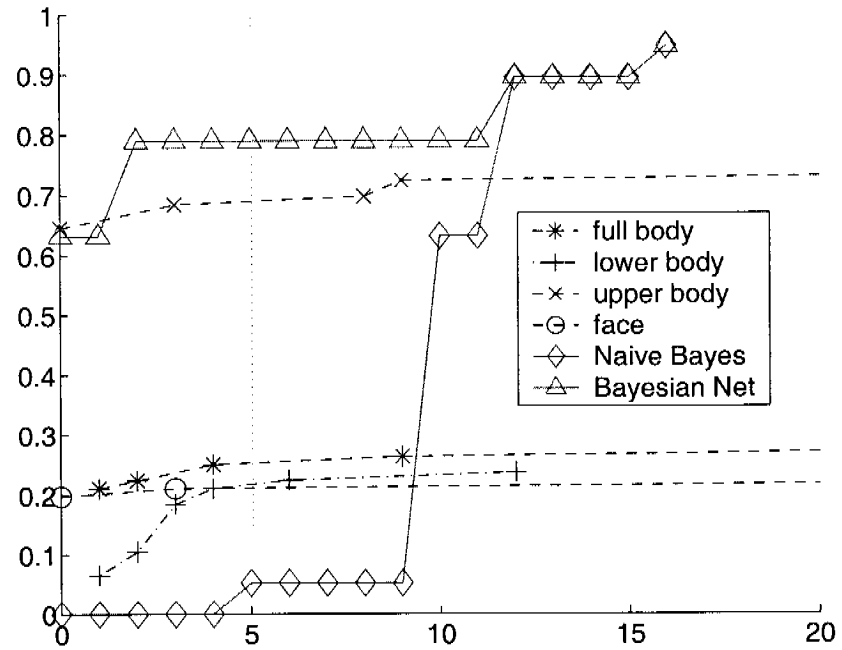

Figure 6.4: *Outdoor test: ROC curves comparing the Bayesian Network of figure 6.2 to the Naive Bayes classifier of figure 6.1 and to the individual scalepart detectors. Again, the Bayesian Network outperforms the Naive Bayes classifier which reaches comparable detection rates only at higher false alarm levels.*

6.3.2 Analysis of Evidence Node Values

In our discrete-node model we can easily monitor the activitiy in the evidence nodes to find out which scaleparts contribute to correct detections. For the indoor test sequence figure 6.5 visualizes whether or not a cue has successfully detected the face per test frame ("cue activation chart"). This chart shows the frame numbers on the horizontal axis and the names of the individual scaleparts on the vertical axis. A black dot in the chart implies that the corresponding scalepart detector contributes to a correct detection (these dots lie so closely that they form contiguous line segments in the chart). Figure 6.6 shows the corresponding chart for the outdoor test sequence.

In the indoor chart in figure 6.5 the complementarity of face versus lower and full body cues is clearly visible. One can also see that the upper body detector is active in many "far" frames (for example frames 100-200) supported by lower and full body detections.

When comparing figures 6.5 and 6.6 two differences can be observed between the indoor and the outdoor test cases: First, in the indoor case, there are always two or three cues which overlap, i.e. they are simultaneously active. In the outdoor

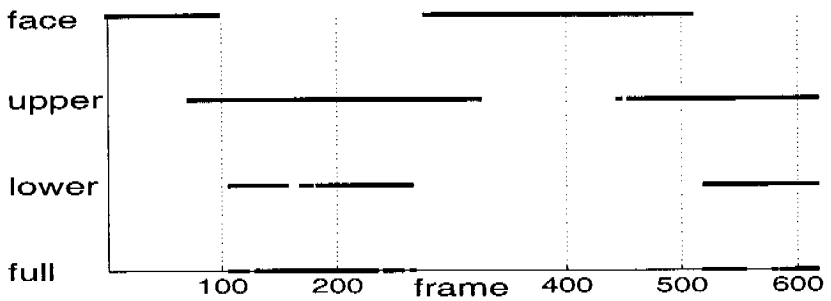

Figure 6.5: *Indoor cue activation chart showing frame numbers on the horizontal axis and individual scaleparts on the vertical axis. A black dot in the chart implies that the corresponding scalepart detector contributes to a correct detection (these dots lie so closely that they form contiguous line segments in the chart). The complementarity of face versus lower and full body cues is clearly visible here.*

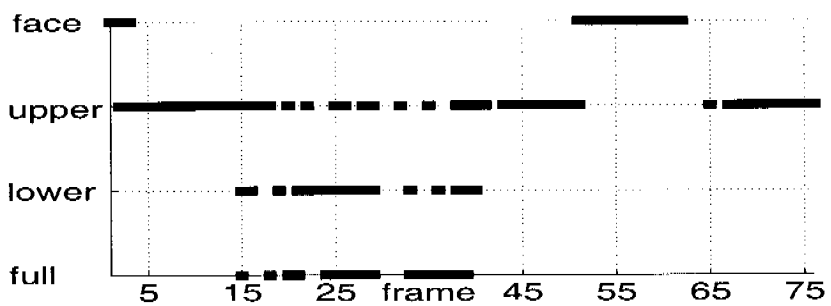

Figure 6.6: *Outdoor cue activation chart. The detection task is generally harder in this sequence which can be seen from the "holes" in the line segments.*

sequence the overlap is between 0 and 3 cues, meaning that in certain frames the face is missed. Second, in the indoor case the cue activations are quite continuous (resulting in continuous line segments within the chart) while in the outdoor case some cues exhibit "jittering" (as in frames 15-40). The jitter is caused by the coarse subsampling of video frames: this means fewer frames and in particular greater differences between two consecutive frames – such that the conditions for scalepart detection are subject to greater changes, too. Also the detection task is generally harder in this sequence.

6.3.3 Analysis of Query Node Values

This section provides a detailed analysis of the inferred values in query nodes C and F for both test sequences.

Indoor Sequence

On the left side of figure 6.7 are three plots which for all frames (frame numbers are given on the x-axis) visualize the inferred posterior probabilities $P(C = 1|f, u, l, b)$, $P(F = 1|f, u, l)$ and $P(\text{FACE} = 1)$ on the y-axis. The plot in the bottom left corner $P(\text{FACE} = 1)$ also shows the annotation of frames as either "close" or "far" on the x-axis (for example, frames 95–299 are "far" and frames 300–500 are "close"). As can been seen in this plot we generally have (with very few exceptions):

$$P(\text{FACE} = 1) > 0.5$$

which is equivalent to

$$P(\text{FACE} = 1) > P(\text{FACE} = 0)$$

In other words, the detector classifies the face correctly in almost all frames: overall this value has been correctly estimated for about 95% of all test frames (both test sequences show a face in every frame, so ideally, we would have $P(\text{FACE} = 1) > 0.5$ for all frames). Similarly, one can compare the estimated values of C and F with the sequence annotation. For the indoor sequence, C and F have been correctly estimated in 99% and 96% of the cases.

Following is a a case-based analysis relating the inferred query node posteriors to the underlying detection constellations. In figure 6.7 the two posterior plots $P(C = 1|f, u, l, b)$ and $P(F = 1|f, u, l)$ contain reference numbers from 1 to 10 which index different situations (test frames) shown on the right half of the figure. For example, plot $P(C = 1|f, u, l, b)$ has a reference number "1" at probability 0.84. This reference number refers to the image frame labeled "situation 1" (frame 3) depicted in the upper right half of the figure. All images also show the frame number as well as the names of active body scalepart detectors: in situation 1 only the face detector has detected the target face.

Following is a discussion of 10 situations which illustrate the mechanisms in the Bayesian Network classifier:

Situation 1: (frame 3) is a portrait situation which can be handled only by the face scalepart detector. The high posterior value of 0.84 for $C = 1$ can be directly attributed to the high confidence of $P(f = 1|C = 1) = 0.85$ learned for this scalepart detector. The upper body scalepart can not be detected since for example the shoulders are not visible. The posterior for $F = 1$ is as low as 0.17, but not zero, because there is a non-zero probability that the other scaleparts have false negatives.

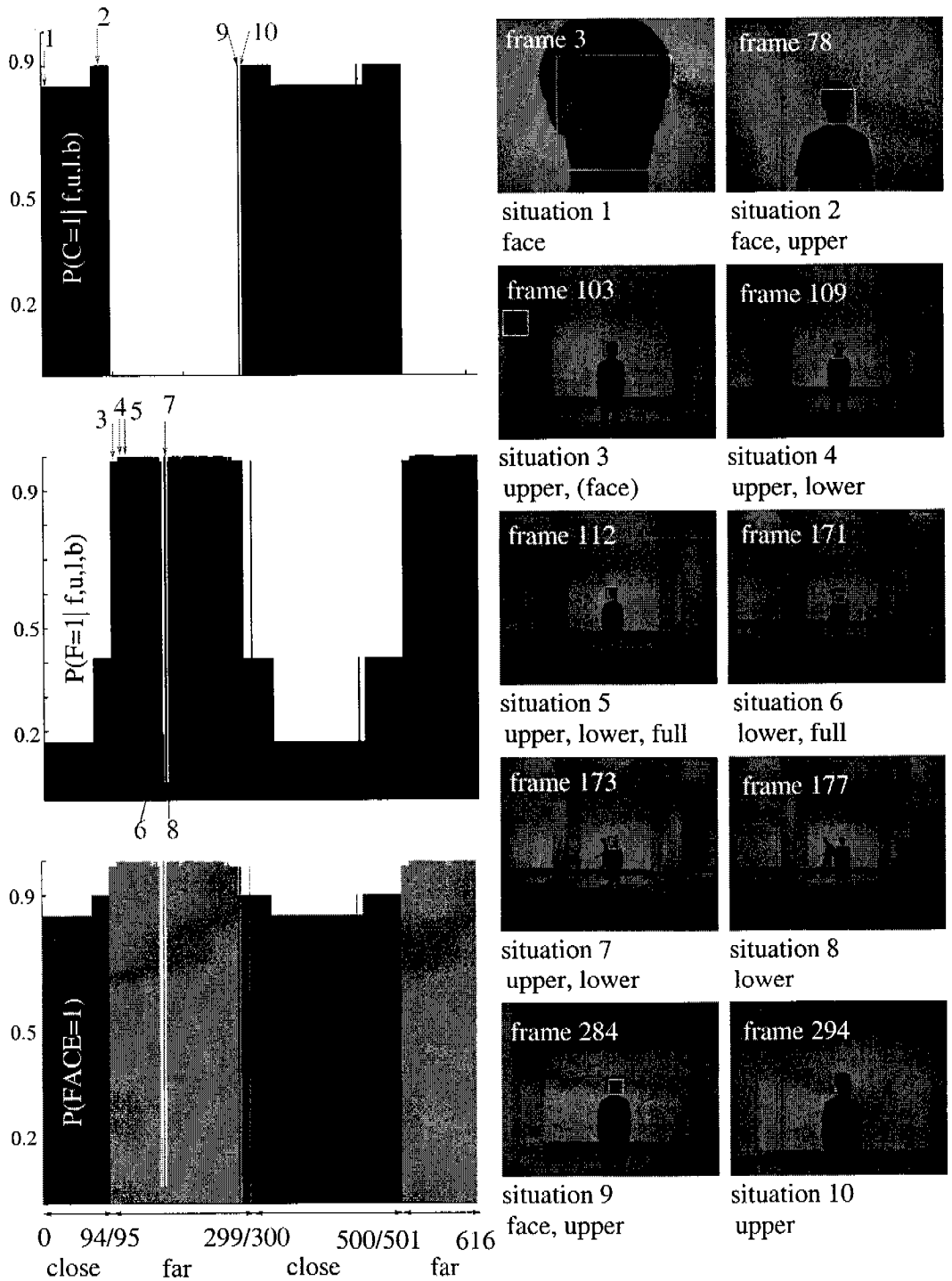

***Figure 6.7**: Case-based analysis of the indoor test sequence relating the inferred query node posteriors (plots on the left side) to the underlying detection constellations (images in the right half).*

Situation 2: In frame 78 the shoulders become visible and the upper body scalepart fires correctly. This increases the posterior probability for "close" to 0.9. However, it also raises the posterior for a "far" situation since the upper body contributes to both cases. The resulting probability of 0.41 for $F = 1$, however, is much lower than 0.9 for $C = 1$.

Situation 3: In frame 103 facial details are no longer distinguishable for the face scalepart and the only detecting cue here is the upper body. This makes $F = 1$ more likely than $C = 1$ with a posterior of 0.98 vs. 0. The high posterior for "far" directly results from the upper body's CPT (table 6.2) where $P(u|C = 0, F = 1) = 0.97$. The non-detection of the face cue further supports the "far" hypothesis. The posterior for $C = 1$ collapses which can be attributed to the learned zero false negative rate for "close" situations, that is, $P(f = 0|C = 1, F = 0) = 0$ according to table 6.1.

The information flow between nodes F and C is due to the "explaining away" phenomenon in Bayesian networks which is discussed separately in section 6.3.4.

Situation 4: The posterior for $F = 1$ is raised from 0.98 to 0.99 in situation 4 (frame 109) through the detection of the lower body: lower and upper body both support this hypothesis. Because of "explaining away" this would actually lower the posterior for $C = 1$. However, its posterior has already reached zero.

Situation 5: With an additional detection by the full body scalepart, the posterior for $F = 1$ reaches a maximum of 0.9967 in situation 5 (frame 112). For the next few frames this posterior slightly oscillates as individual scalepart detectors hit or miss the target. However, these deviations are extremely small (1% or less) as long as the upper body cue is detecting correctly.

Situation 6: In frame 171 (situation 6) only lower and full body scaleparts help detect the face but not the upper body cue. Since their reliabilities are relatively low (0.63 and 0.46 according to tables 6.3 and 6.4) the posterior for $F = 1$ drops to 0.19 (versus zero for $C = 1$). The low posterior can also be attributed to the high trust placed in the upper body scalepart which has hardly missed any "far" face during training. From this point of view, the isolated detections of lower and full body are "suspicious".

Situation 7: In situation 7 (frame 173) the upper and lower body scaleparts detect the face which raises the posterior for $F = 1$ again. There is an additional person in the background walking from right to left whos face has not been annotated because of the sideview perspective. The person causes a small amount of false alarms (the upper body detector seems to be mislead here sometimes,

because the backpack which the person is wearing creates a silhouette similar to a frontal upper body).

Situation 8: Only the lower body detects the face in situation 8 (frame 177). The confidence in this detection is, however, quite low: the posterior probability is 0.04 for $F = 1$ (versus zero for $C = 1$) because the lower body scalepart has been learned not to be very reliable according to table 6.3.

Situation 9: is analogous to situation 2: the camera has zoomed in again onto the face, resulting in detections of face and upper body scaleparts. The other two scaleparts cannot contribute because the lower and full body are not visible. This constellation correctly favors $C = 1$ with a posterior of 0.89 over $F = 1$ whos posterior probability is 0.41.

Situation 10: The face scalepart fails on frame 294 but the upper body scalepart still works correctly. Situation 10 is thus analogous to situation 3.

Outdoor Sequence

Analogous to the above analysis of the indoor sequence, figure 6.8 shows the corresponding plots and images for the outdoor sequence. Here, the value of FACE is correctly estimated in about 83% of the cases. Variables C and F have been correctly estimated for 50% and 68% of all test frames. The estimation of C is not very accurate because from the indoor training subset we have learned to trust the face detector – ignoring possible head movements (which occur only in the outdoor sequence). Likewise, no detections or few detections of "unreliable" cues (such as lower or full body) will result in zero or low posterior values. This causes the same "jitter" in the probability plot which has also been observed in the cue activation charts of section 6.3.2.

Following is a case-based analysis of 10 situations indexed in the two plots $P(C = 1|f, u, l)$ and $P(F = 1|f, u, l)$:

Situation 1: The face scalepart fires in frame 1 (situation 1) which is a rare event in the outdoor sequence because of small face sizes and head movement. This means that for the majority of frames the Bayesian network has to overcome missing detections of the face scalepart which has been learned to be highly reliable.

Situation 2: Analogous to the indoor sequence the joint detection of face and upper body scaleparts in frame 3 increases the posterior for $C = 1$.

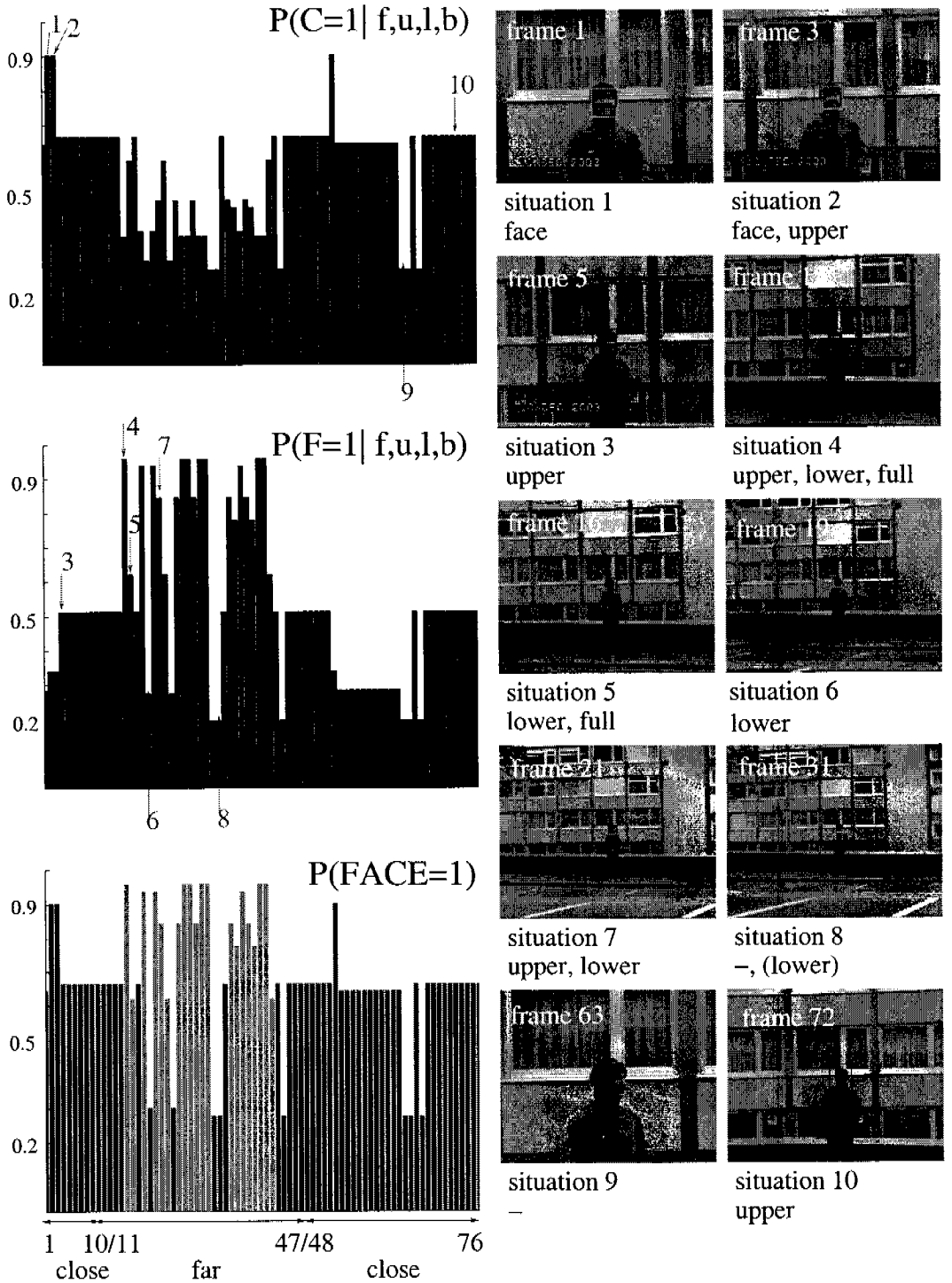

Figure 6.8: *Case-based analysis of the outdoor test sequence analogous to figure 6.7.*

Situation 3: In frame 5 the target face is already too small for the face scalepart. The upper body, however, detects the face despite its non-frontal pose. This situation is another example of "explaining away" in the Bayesian Net: the decrease of $P(C = 1|f, u, l, b)$ caused by the now missing face detection causes an increase in the posterior probability for $F = 1$.

Situation 4: Upper, lower and full body scaleparts deliver correct detections in frame 15 and cause a sharp increase in the posterior of $F = 1$. This also causes the posterior of $C = 1$ to decrease, again because of "explaining away".

Situations 5,6,7,8: These four situations illustrate the difficulty of the sequence resulting in missed detections. In principle, the resolution and visibility in these frames allow for detections of upper, lower and full body. However, in frame 31 for example, the upper body is turned away from the camera in a "far" situation (pose change). The only detection in this frame is due to the lower body which, however, is too inaccurate and counted as a false alarm.

Situations 9, 10: These are two additional hard cases: in situation 9 (frame 63) both upper body and face are turned away from the camera. In situation 10 (frame 72) the upper body scalepart correctly infers the presence of a face despite the difficult pose.

6.3.4 Explaining Away

A particular property of Bayesian network models is their ability to *explain away* competing evidence. In the proposed Bayesian Network classifier of figure 6.2, the two query nodes C and F can compete in explaining the observed evidence, which has been discovered in situations 3 and 4 of both test sequences in the case-based analysis (section 6.3.3). This section enlarges upon situation 3 of the indoor sequence, illustrating the explaining away mechanism in more detail.

In figure 6.9 all observed nodes have been instantiated with the current evidence, that is the output of the scalepart detectors. In this situation only the upper body detector hints at the presence of a face. The inferred query node posteriors take upon values $P(C = 1|f, u, l, b) = 0$ and $P(F = 1|f, u, l, b) = 0.9855$, respectively.

Now the situation changes in figure 6.10 where the face detector indicates the presence of a face, too. The observed node changes its value from 0 to 1 and the posterior of $C = 1$ jumps from 0 to 0.89 – this is because in a "close" situation, a detected face is the strongest possible evidence.

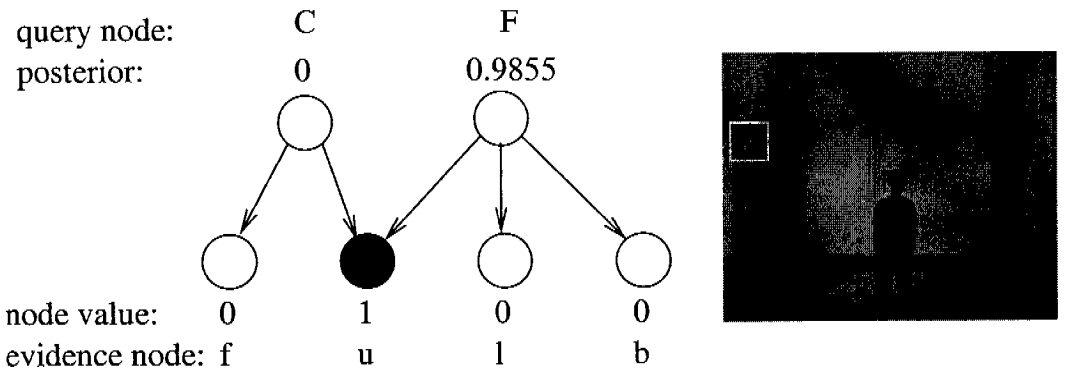

Figure 6.9: *Conditional dependence: Query nodes are shaded, observed nodes are black if their value is zero or white otherwise. Since the observed nodes have been instantiated (their values have been observed and are fixed) the two query nodes become conditionally dependent. Here, the node values reflect the accumulated evidence on the actual face ($f = 0$, $u = 1$, $l = 0$, $b = 0$).*

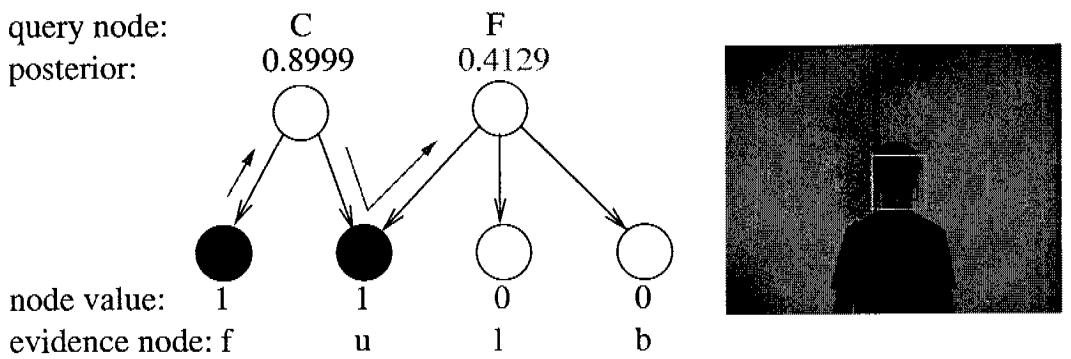

Figure 6.10: *Explaining Away: The posterior of query node C changes from 0 to 0.89 because of new evidence from the face detector ($f = 1$, $u = 1$, $l = 0$, $b = 0$). This initiates an information flow (red/light gray arrows) via the upper body node, and results in a decrease of node F's posterior to 0.41.*

The two parents of the upper body node now compete in explaining the observation of the upper body. Node C "explains away" this observation because it is consistent with the observed face scalepart. This results in a decrease of the posterior for $F = 1$ from 0.9855 to 0.4129.

6.4 Conclusion

In this chapter we have applied a discrete-node Bayesian Network for fusing scalepart detectors. Bayesian Networks allow for encoding scalepart interactions in terms of causality (parent-child relation) and conditional independence. The individual node distributions have been learned from data via maximum likelihood estimation. Bayesian Networks are a particular family of graphical models for which exist many efficient learning and inference algorithms. The power of this formalism makes our approach scalable to include many more evidence nodes (additional scaleparts, global context cues, other modalities like range information etc.) with continuous and/or discrete evidence. The experimental evaluations have shown that the combination – in this case resulting in a scaleparts-based face detector – clearly goes beyond the capabilities of individual scaleparts – with improvements of up to 20% in detection rate at a similar level of false alarms. This is despite the fact that the face height is less than 20 pixels in about half of the test frames.

As a final remark, the proposed implementation of the scaleparts-based face detector unifies discriminative (AdaBoost) and generative (Bayesian Nets) techniques. Seen as a hierarchy, AdaBoost selects low-level visual features such as edges and bars to discriminate certain object parts from background clutter. These parts are then combined in a Bayesian Network where we can use *prior knowledge* about part interactions in designing the network topology.

7

Scaleparts-based Car Detection

The previous chapter has successfully applied a scalepart-based object detector to face detection. Scaleparts corresponded to a person's face, upper, lower and full body. They were used to indirectly infer the presence of a face, especially at low image resolution. This can be seen as an example of context-supported detection because it considers information (the upper, lower and full body) which is located *outside* the actual target (the face).

In this chapter we switch back to the traditional object-centered detection paradigm where all considered features sit on the target. The purpose of this chapter is to show that the scaleparts idea is applicable to other types of objects as well. Here, we are specifically looking at car detection in cluttered images.

More specifically, a scalepart-based detector for side views of cars is developed and evaluated. Section 7.1 discusses the learning of car scaleparts. The detection performance for the individual car scaleparts is then investigated in section 7.2 and a Bayesian combiner is introduced and evaluated in section 7.3. Section 7.4 discusses results and delineates possible extensions.

7.1 Learning Individual Scaleparts

This section is concerned with the learning of car scaleparts. The set of scaleparts proposed here consists of a full car, a "half car" and wheels. First, section 7.1.1 describes the employed training data of sideviews of cars as well as the training parameters for AdaBoost learning. Then, section 7.1.2 analyzes the features which were automatically selected during training and compares them among the three car scaleparts.

7.1.1 Training Data

We choose three scaleparts for detecting sideviews of cars: the full car, a "half car" and a single wheel. Following the rationale of the scalepart framework (see chapter 5, section 5.1) this choice of parts is motivated by the need to accomodate different levels of available image detail.

As a starting point we use the 550 training images of side views of sedan type cars of [Agarwal and Roth 2002]. Since the integral image features considered by AdaBoost are highly localized the training data has been further refined in several ways. First, the variation in appearance has been reduced by flipping all cars so that they all face the same direction. Second they have been aligned using manually defined landmark points. Third, the training data has been annotated to allow for extracting half cars and individual wheels. As a final preprocessing step, training instances with partial occlusions have been removed.

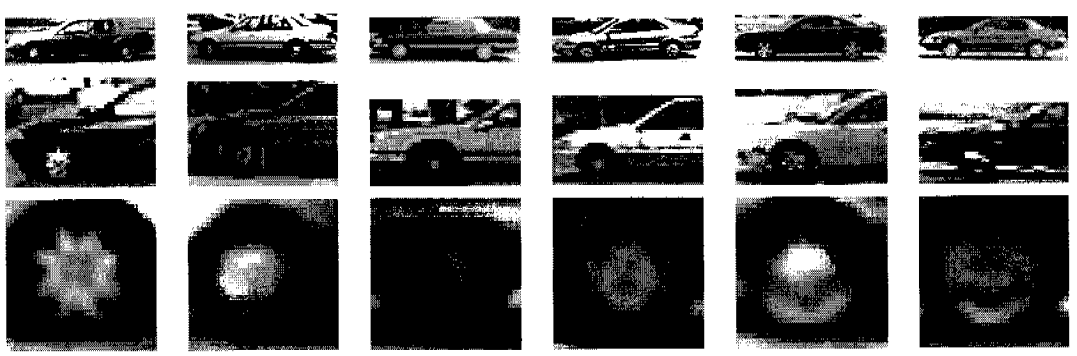

***Figure 7.1**: Preprocessed instances for scaleparts training (rows): full car, half car and wheels. These have been extracted from the training set of [Agarwal and Roth 2002]. Training instances with partial occlusions have been removed.*

Figure 7.1 shows example training instances for full cars (in total 429 instances), for half cars (in total 429 instances) and for wheels (in total 805 instances). For half cars we have tested both frontal (left-facing in our case) and rear halves and have achieved slightly better results with the former. However, when downscaled to the actual training resolutions, the sedan car type appears quite symmetric anyway. As a result, both the half and full car scaleparts are not limited to the orientation they were trained with. They do show a certain bias towards the trained orientation but frequently detect the opposite orientation as well, as we will see in section 7.2.1.

The training resolutions for full, half and wheel are shown in table 7.1 which summarizes the most important learning parameters. For each detector, multiplying the training instance's width and height amounts to about 400 pixels square – the same rule of thumb, born out of experiments, was used in chapter 5. Note that compared

Parameter	Full Car	Half Car	Wheel
Width	14	19	20
Height	28	23	20
Positive examples	429	429	805
Negative examples	2000	2000	2000
Stages	30	27	14
Min hit rate	0.995000	0.995000	0.995000
Max false alarm rate	0.500000	0.500000	0.500000
Training Time	1 week	1 week	1 week

***Table 7.1**: Car scaleparts learning parameters. As a rule of thumb, born out of experiment the product of width and height is about 400 pixels square. Compared to the training of full and lower body there is only half the amount of training data available here for full car and half car training instances. The WuArchive image database was used as negative image set.*

to the training of full and lower body there is only half the amount of training data available here for full car and half car training instances. The WuArchive image database was used as negative image set as in chapter 5.

7.1.2 Selected Features

It is interesting to examine the specific features that have been found to be discriminant for wheels, half cars, and full cars, respectively.

The first 5 stages of the learned full car detector are visualized in figure 7.2. Again, the feature visualization (black and white boxes) is analogous to chapter 4, figure 4.10. Feature "1a" refers to stage "1", first feature ("a"). According to the figure the most important features in full car detection seem to be the car's front mudguard and the ground shadow from the chassis. They already appear in the first cascade stage (1a and 1b) and occur again in later stages. For example, additional mudguard features are (2a, 3a) and additional ground shadow features are (3b, 3e, 4b, 5d). Other features are dedicated to car doors (2b,2c,3d) and wheels (4a). It is interesting that among the first 20 features shown here, there are only two which contain parts of the windows (5b, 5e). All other features are strictly confined to the lower chassis and ground.

As shown in figure 7.3, the half car detector's first feature (1a) is also covering parts of the mudguard, but sits slightly higher: unlike the full car detector it excludes the wheel axle – which appears as the brightest spot within the wheel. Compared to the full car cascade the half car detector seems to pay more attention to the wheels manifested in a higher number of according features (1b, 2a, 2e, 3a, 3c). Other

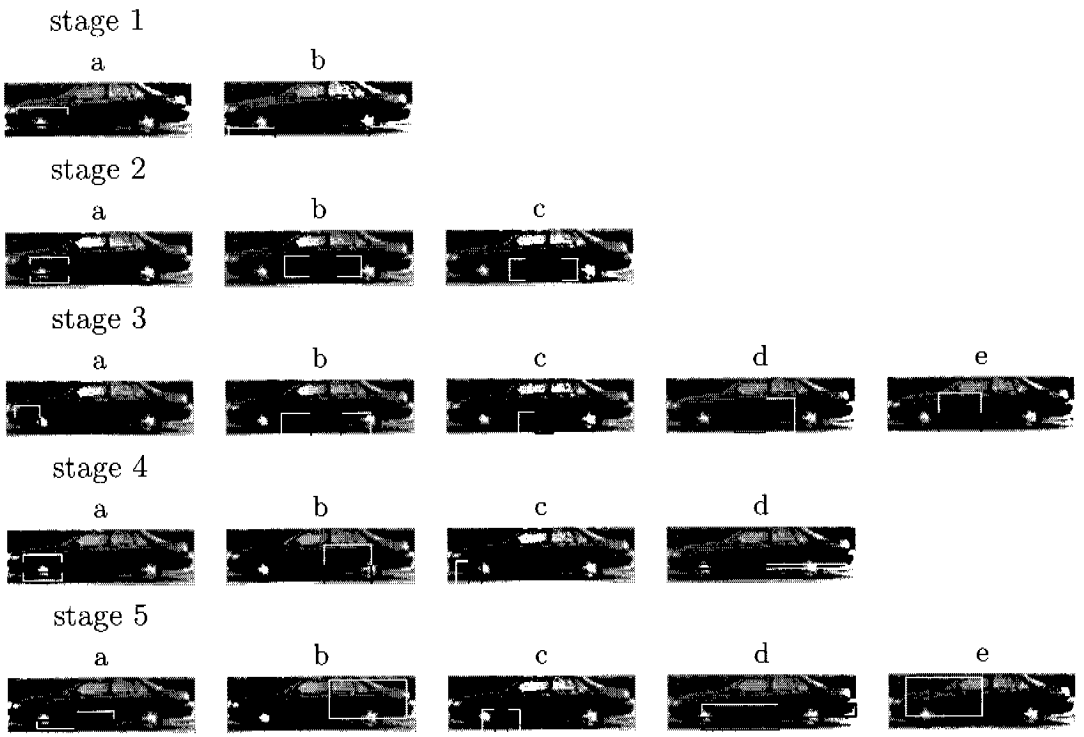

***Figure 7.2**: Full car features selected by AdaBoost. The figure shows the first few stages of the learned cascade. A full car is distinguished by the car's ground shadow from the chassis and the frontal mudguard also including part of the wheel.*

features cover the lower chassis (2d, 3d, 3e) – but unlike the full car detector only few touch the ground shadow.

For the wheel detector (figure 7.4) certain features cover almost the entire rim and tyre (1a, 1b, 3a). Other, smaller features pick up on the tyre boundary alone (2b, 2d, 2e and most features of stage 4). This is also an example of the usefulness of the center-surround feature (3a) and rotated features (4a) which help approximate the circular wheel shape.

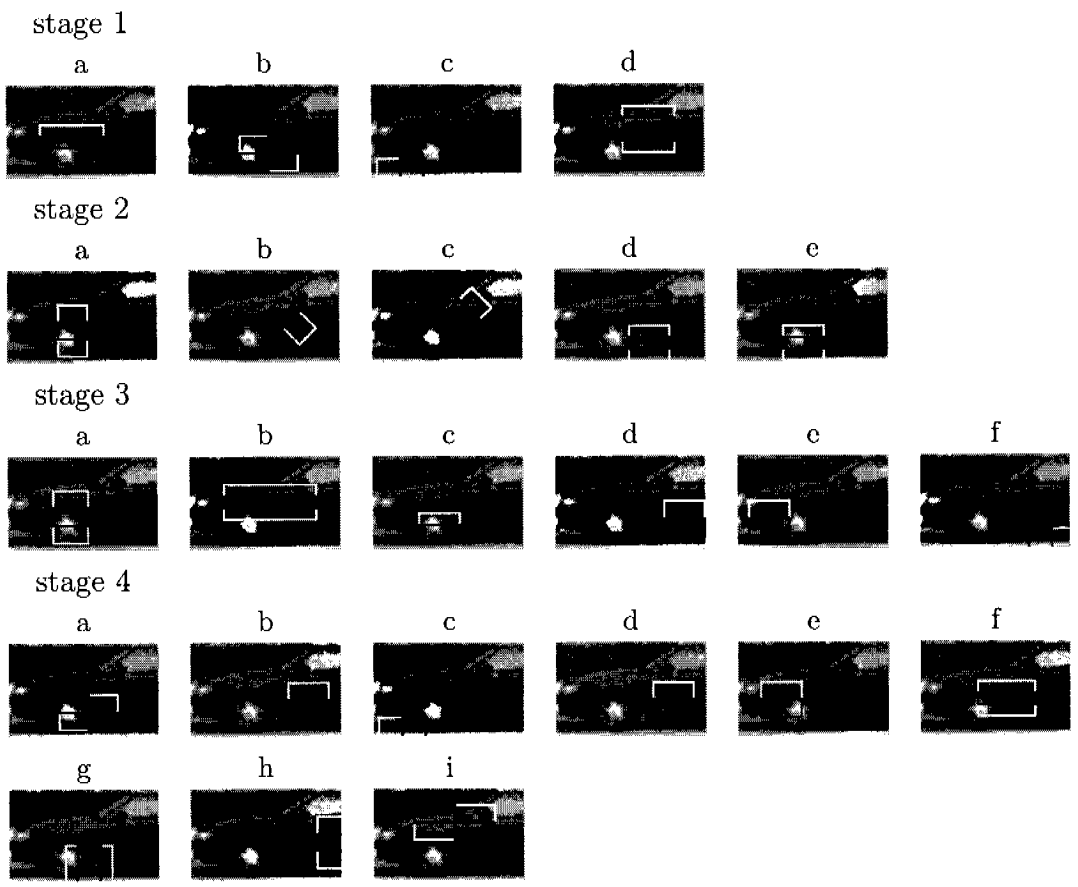

Figure 7.3: *Half car features selected by AdaBoost. The figure shows the first few stages of the learned cascade. A half car is recognized by the region around mudguard and wheel. Generally this detector pays more attention to the wheel than the full car detector.*

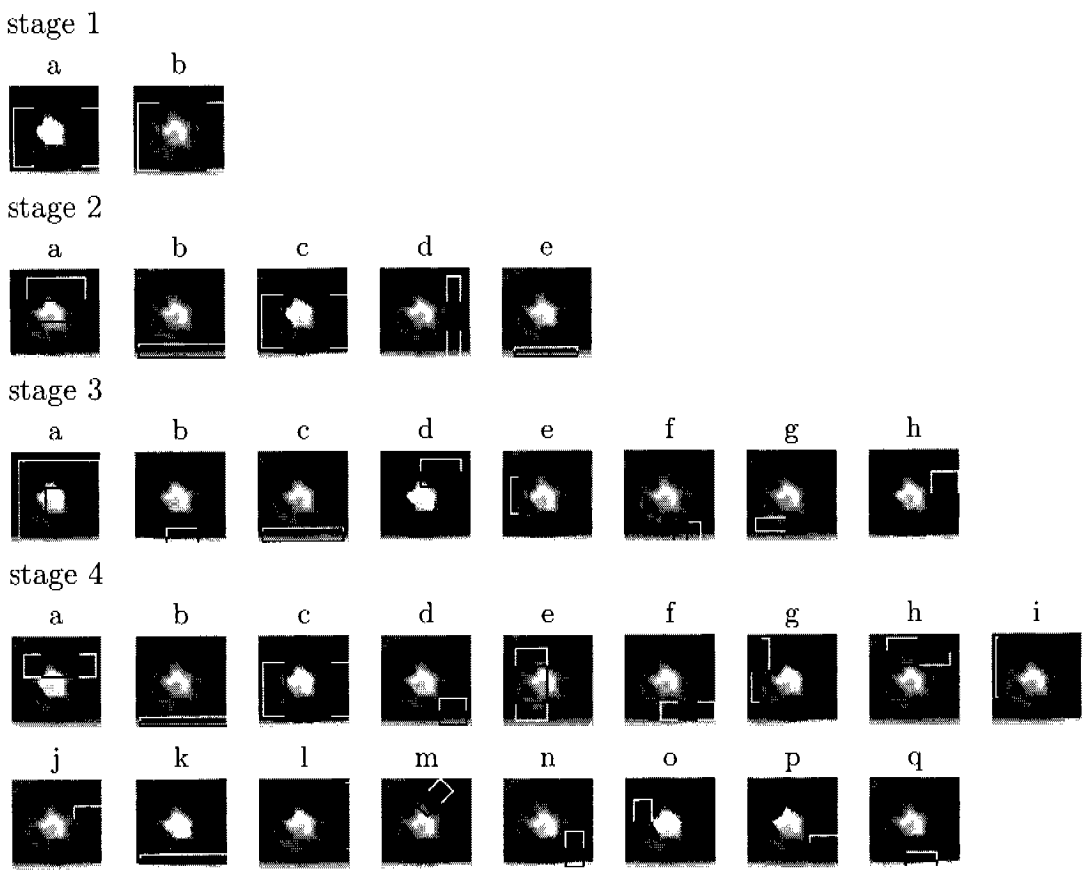

Figure 7.4: *Wheel features selected by AdaBoost. The figure shows the first few stages of the learned cascade. The wheel detector uses rotated and center-surround features to capture parts of the rim, and the tyre as well as the specific contrast between axle and tyre.*

7.2 Scalepart Detection Performance

Each learned detector is evaluated on Agarwal and Roth's test database. This database consists of 170 grayscale images containing 200 sideviews of cars. Most of these cars are of the sedan type, they come in different "colors" (i.e. gray levels) and can face left or right. In the original test images all cars are of the size of 100 × 40 pixels. We will refer to this set of images at their original resolution as "original".

Agarwal and Roth only consider detection at a single scale. Instead, we are especially interested in performance over a wide range of scales. Therefore, we additionally create a lower and a higher resolution version of the data using bicubic interpolation. This results in 200 cars at 60 × 24 pixels ("lores"), 200 at the original resolution 100 × 40 pixels ("original") and 200 at 250 × 100 pixels ("hires"). This is equivalent to scaling factors of 0.6, 1.0 and 2.5.

Each scalepart detector is applied independently over all positions and scales to these test images. The following section 7.2.1 discusses a set of representative detections. ROC curves are then examined in section 7.2.2.

7.2.1 Qualitative Analysis of Detected Car Scaleparts

Figure 7.5 shows the detected car scaleparts in 7 test scenarios at the three different resolutions "lores", "original" and "hires". As in chapter 5 each car scalepart detector is applied independently over all positions and scales with a downscaling factor of 1/1.2. Input images vary in size, but each detector generally has to examine at least several tens of thousands of subwindows for "lores" and "original" resolution images or even several hundred thousand subwindows for "hires" images. For example, "hires" image (a) in the top row of figure 7.5 is 755 × 365 pixels which means that 779'289 subwindows have to be classified. The detection arbitration procedure is identical to the one used in previous chapters: overlapping detections are merged and their positions and sizes are averaged resulting in the detections shown in 7.5.

Full car detections are shown with a continuous line bounding box, half cars have dashed lines, and wheel detections have lines of dashs and dots. The different line styles also help to identify the responsible cue in case of false alarms.

Each column in the figure corresponds to a specific image resolution and car size. To recall, within each column the car size is actually constant, but the images have been rescaled for better visualization. Each row shows a particular test image to allow for observing resolution-dependent changes. Generally, images have been rescaled for visualization only, in fact their true sizes vary by a factor of $4.16 = \frac{2.5}{0.6}$ between "hires and "lores".

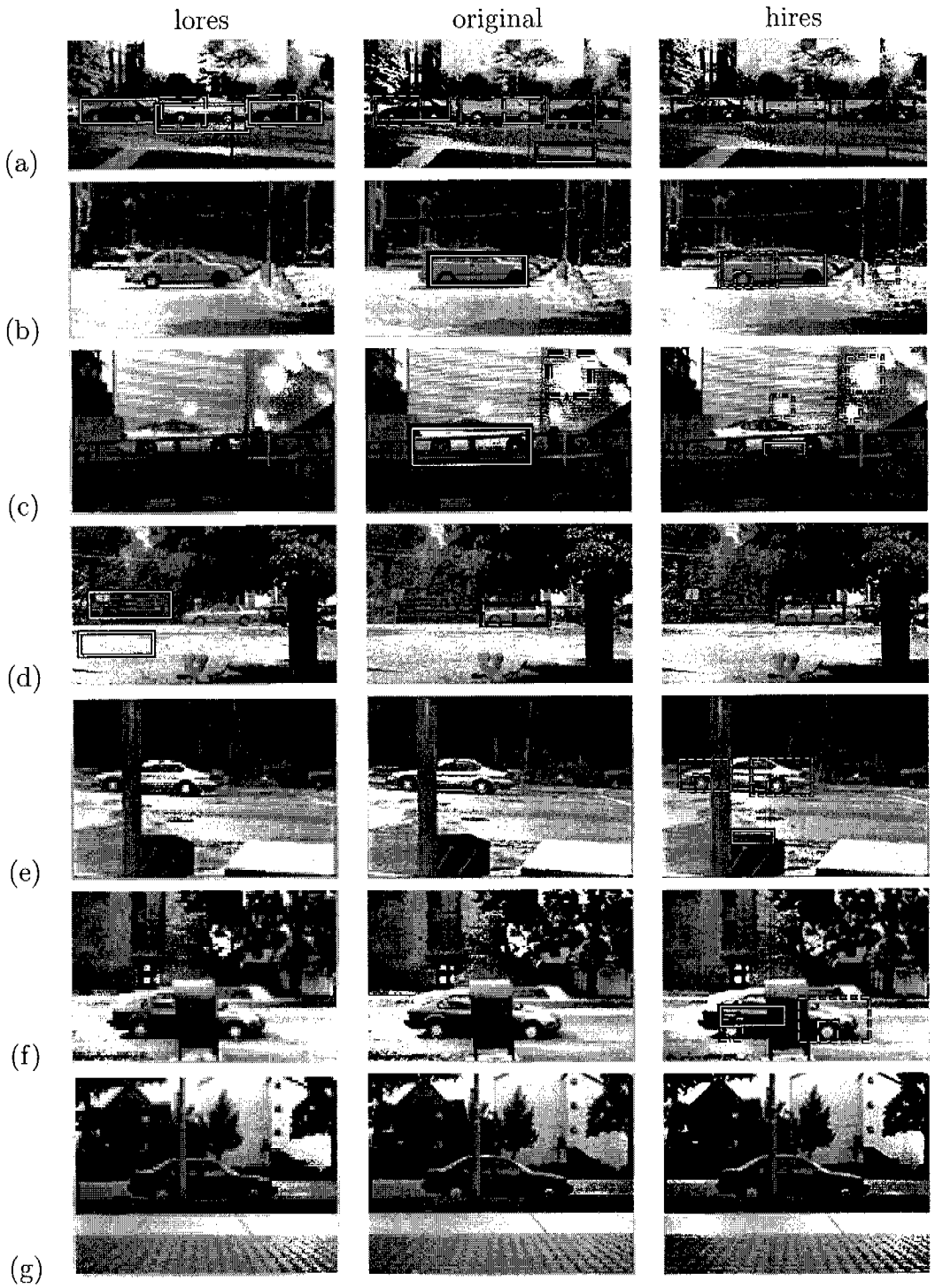

Figure 7.5: *Detection of car scaleparts in difficult test instances. Each scalepart detector searches over all positions and scales. Columns represent the different test image resolutions, each row shows a particular test image. The figure illustrates correct detections of cars that vary in appearance, size and orientation, as well as occlusion handling, missed detections and background distractors.*

In the following we discuss the examples of figure 7.5 in more detail. Example (a) shows the detection of three parking cars within one image. The full car detector finds all cars irrespective of the actual resolution. However, the detection of the car in the middle gets more accurate at resolutions "original" and "high" – which is also generally the case. The half car detector detects but one car irrespective of the actual resolution and all wheels are found in the "hires" version. Wheels at the lower two resolutions are too small to be detected: the wheel detector size is 20×20 pixels, "lores" wheels are 7×7 pixels). A false positive full car detection appears at resolution "hires".

In example (b) a right-facing car is detected by the full car detector at resolutions "original" and "hires". At "hires" the rear of the car is detected by the half car detector. This can be explained by the symmetry of the sedan car type and the low training resolution. Note how the left wheel is detected but the right wheel is not – probably for its unusually dark color. A rather difficult test instance is (c) because the car is occluded by the fence and slightly in-plane rotated since it is going uphill. The car is detected at the original resolution via the full car scalepart. At the next higher resolution this detection is suppressed by an overlapping smaller and unfortunately much more inaccurate detection. This is an effect of the detection arbitration scheme which fuses overlapping detections by averaging their positions and sizes. Several spurious "wheels" appear in these images, too. However, given that detectors operate entirely locally these types of false alarms seem quite reasonable.

Example (d) of figure 7.5 is a "hidden" car with weak contrast which is successfully detected by half and full car detectors at resolutions "original" and "hires". At the lowest resolution the car could not be detected.

Rows (e)-(g) are all examples for partial occlusion which can be resolved in cases (e) and (f) at the highest resolution but not at lower resolutions. The wheel detector obviously plays an important role in these cases. In (g) wheels are also partially occluded.

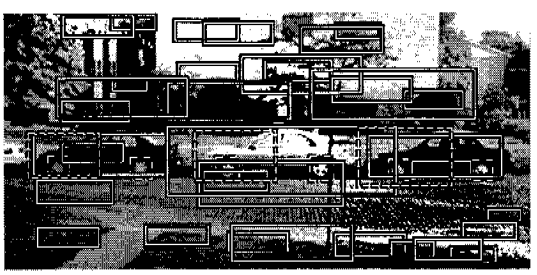

Figure 7.6: *In case of the full car detector admitting low confidence detections results in many false alarms.*

For this visualization a threshold has been set and fixed for each detector. For wheel

and half car detectors this threshold is always the same. For the full car detector the threshold is the same for all "lores" detections. However, a more conservative threshold has been used for displaying "original" and "hires" detections, because at higher resolutions the full car detector produces many false alarms. This is illustrated in figure 7.6 which shows image (a) "hires" with the full car detector set to the "lores" threshold.

7.2.2 ROC Analysis

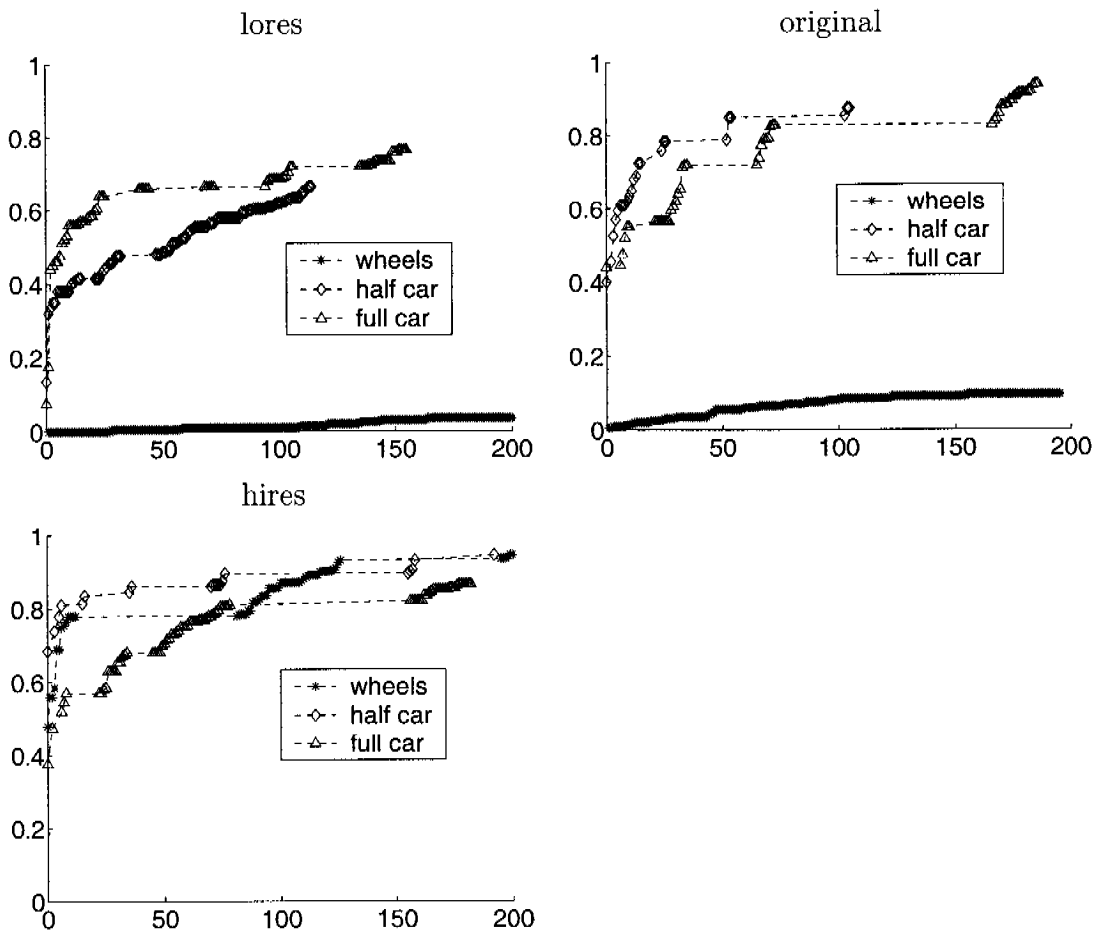

***Figure 7.7**: Individual car scalepart performance (wheel, half car and full car)*

This section compares the contributions of full car, half car and wheel scaleparts for car detection at different resolutions. ROC curves for test sets "lores", "original" and "hires" are shown in figure 7.7 in order of increasing image resolution.

In the "lores" case the full car is the best-performing scalepart: its curve is consistently above the other two and rises from a detection rate of about 50% at 10 false alarms to 65% at 20 false alarms and eventually reaches about 80% at 170 false alarms. The half car detector retrieves up to 65% of the cars at 110 false alarms. At 20 false positives it finds about 40%. Actual wheels at his resolution are less than half the size of the wheel detector's search window and can therefore only be detected "accidentally" at extremely low thresholds. The detection rate therefore stays below 10% even at 200 false alarms.

At the "original" resolution both half and full car scaleparts exhibit high detection rates. At 10 false alarms both retrieve about 50% of the cars, then at 20 false alarms the half car detector reaches 70% versus 55% of the full car. At higher false alarm rates the half car curve rises faster and is consistently above the full car curve but does not exceed a detection rate of 85% at 110 false alarms. The full car curve, however, reaches up to 90% at about 190 false positives. Wheels at this resolution are still beyond the wheel scalepart's reach: their size is about 12×12 pixels.

In the "hires" case wheel and half car scaleparts both have steep ROC curves: at 20 false alarms each of them finds close to 80% of the cars. From then on both curves rise slowly and reach a detection rate of up to 95% at 190 false alarms. The full car scalepart has a flatter curve reaching about 55% at 20 false alarms and 85% at 190 false alarms.

Interestingly, the half car detector slightly outperforms the wheel detector at the highest resolution and not vice versa. This is surprising because wheels are the most detailed scalepart among the three and therefore particularly suited to the high resolution case. However, there seem to be more wheel-like background distractors in these images which result in a slower rising curve for "wheels" than for "half car". Also, half car and wheel detectors continuously improve their detection rates as resolution increases. The full car detector, however, evokes an increasing number of false alarms and reaches high detection rates only at increasing false alarm levels.

7.3 A Scaleparts-based Car Detector

This section develops and analyzes a scaleparts-based detector for sideviews of cars. First, a Bayesian Network for scalepart combination is introduced in section 7.3.1. As a proof of concept, the network parameters are manually instantiated to allow a first evaluation of the combined detector in section 7.3.2. Experimental results merging "lores", "original" and "hires" test images into one test set are discussed in section 7.3.3.

7.3.1 Bayesian Network Topology

We consider the Naive Bayes case as a starting point for a Bayesian Net-based combined part detector. This network is displayed in figure 7.8.

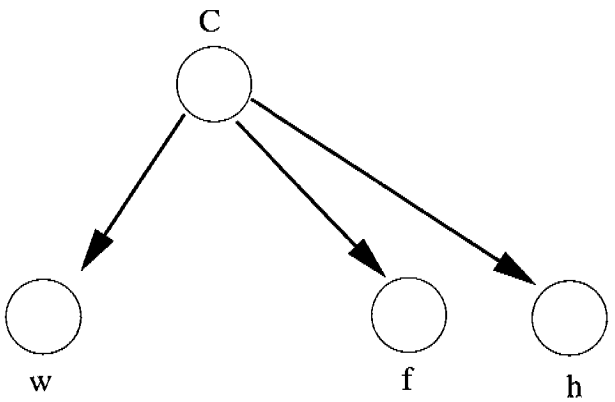

Figure 7.8: *Naive Bayes combination of car scaleparts.*

The parent node *car* causes observations of a full and a half car and of wheels. In this network the absence of wheels will effectively lower the confidence of any car hypothesis. However, this is inappropriate for low image resolutions were the level of image detail is not sufficient for detecting wheels. Also, a topology with "close" and "far" nodes analogous to chapter 6 is not applicable here: this topology for body detection encodes the assumption that certain scaleparts cannot occur together (in particular fullbody and face scaleparts). In the car images at "original" and "hires" resolutions, however, all three car scaleparts frequently coincide.

Ideally, one would estimate the global resolution situation of the test image first to modulate the influence of individual scaleparts. The proposed topology in figure 7.9 therefore conditions the wheel node on a new observed node *lores* which takes upon the value "1" for a low resolution image. Likewise the full car detector is conditioned on that same variable. The wheel and full car node now have the two common parents *car* and *lores*. With the instantiation of *lores* the two observed nodes wheel and full car become conditionally dependent. This triggers a similar information flow and "explaining-away" behavior as in section 6.3.4 of the previous chapter.

Let F, C, L be the binary random variables for the full car, car and lores nodes. Then by conditioning on L the probability $P(F = 1|C = 1, L = 1)$ can be different from $P(F = 1|C = 1, L = 0)$. This condtioning enables adjusting the influence of wheel and full car nodes in the network for different resolution situations.

A pragmatic implementation for computing the value of *lores* is to use any wheel detections as an indicator for $lores == 0$ and set $lores == 1$ otherwise. Note that

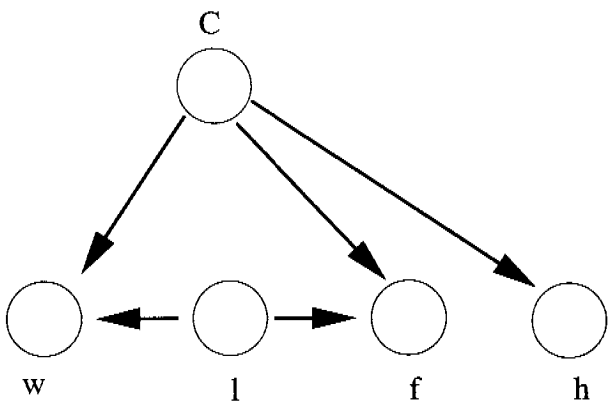

***Figure 7.9**: The additional node* lores *adjusts the influence of* full car *and* wheel *for a given hypothesis with respect to the resolution of the image.*

this means wheel detections *within the entire image* whereas all other observations are strictly local to the current hypothesis. Thus after applying the individual scalepart detectors the value of *lores* has to be computed only once for the entire image.

All conditional probability tables (CPTs) can be learned as described in chapter 6, section 6.2. The learning entails labelling training data with the desired values of C (car) and L (lores), followed by maximum likelihood estimation of the table entries. The following section 7.3.2 describes a simplified approach with manually chosen CPT entries which is sufficient to deliver a proof of concept for the feasibility of scaleparts-based car detection.

7.3.2 Conditional Probability Tables For Car Detection

This section explains the manually estimated entries for the conditional probability tables (CPTs) of network 7.3. Automatic learning of CPTs using Maximum Likelihood estimation is covered in chapter 6, section 6.2. Table 7.2 shows the CPT associated with the full car node F. Here, probabilities are conditioned on variables CAR and $LORES$. Since rows must sum up to 1 we only need to estimate 4 table entries:

$P(F = 1|CAR = 0, LORES = 0) = 0.8$ is the estimated false alarm rate for high resolution images. It is higher for the full body scalepart then for any other detector.

$P(F = 1|CAR = 1, LORES = 0) = 0.5$ is the estimated true positive rate in high resolution images. Since wheel and half car scaleparts are estimated to be more reliable at higher resolutions this value is also set conservatively.

$P(F = 1|CAR = 0, LORES = 1) = 0.1$ is the estimated false alarm rate for low resolution images. This low number as well as the following

$P(F = 1|CAR = 1, LORES = 1) = 0.8$ expresses the high confidence put in the full car scalepart. It often is the only working cue in low resolution situations.

CAR	LORES	$P(F = 0\|CAR, LORES)$	$P(F = 1\|CAR, LORES)$
0	0	0.2	0.8
1	0	0.5	0.5
0	1	0.9	0.1
1	1	0.2	0.8

***Table 7.2**: The CPT for the full car node in network 7.9*

The half car node (CPT shown in table 7.3) has only one parent, variable "CAR". $P(H = 1|CAR = 0) = 0.1$ expresses its low expected false alarm rate and $P(H = 1|CAR = 0) = 0.7$ its relatively high true positive rate.

Finally, the wheel node is again conditioned on two variables analogous to the full car node. However the table entries are almost complementary to those of the full car CPT.

$P(W = 1|CAR = 0, LORES = 0) = 0.1$ is the estimated false alarm rate for high resolution images which is low.

$P(W = 1|CAR = 1, LORES = 0) = 0.7$ is the estimated true positive rate in high resolution images. This relatively high number expresses that wheel and half car scaleparts are most important for high resolution detection.

$P(W = 1|CAR = 0, LORES = 1) = 0.01$ is the estimated false alarm rate for low resolution images. The wheel detector is very specific and indeed produces extremely few false alarms in low resolution images. This is also why it was chosen for estimating the value of variable "lores" which indicates the global resolution situation of the image.

$P(W = 1|CAR = 1, LORES = 1) = 0$ basically says that the wheel detector is to be switched off completely in low resolution situations.

7.3.3 Results Over All Resolutions

Applying the Naive Bayes classifier of figure 7.8 and the Bayesian Network of figure 7.9 to all $510 = 3 \times 170$ test images ("lores", "original" and "hires" taken together)

CAR	$P(H=0\|CAR)$	$P(H=1\|CAR)$
0	0.9	0.1
1	0.3	0.7

Table 7.3: *The learned CPT for the half car node in network 7.9*

CAR	LORES	$P(W=0\|CAR, LORES)$	$P(W=1\|CAR, LORES)$
0	0	0.9	0.1
1	0	0.3	0.7
0	1	0.99	0.01
1	1	1	0

Table 7.4: *The learned CPT for the wheel node in network 7.9*

results in the ROC curves shown in figure 7.10. These curves are "edgy" because the classification function is computed from four binary input nodes which limits the number of possible posteriors to 16.

In this plot, the Naive Bayes classifier's detection rate does not exceed 60% but reaches this detection rate at only 20 false alarms. At this low false alarm level it improves over the Bayesian Network which reaches 27%. However, at higher false alarm levels (390 false positives which corresponds to less than 1 false alarm every other image), the Bayesian Network is able to retrieve 492 out of 600 cars, or 82% which clearly improves over the Naive Bayes classifier's performance. It is important to note that for exceeding a detection rate of 67%, some of the 200 smallest cars ("lores", 60×24 pixels) must be correctly detected. Since for finding these small cars the combined detectors have to heavily rely on the "full car" scalepart, the number of false alarms is relatively high.

We also tried to compare these results directly to those of other authors, for example to [Agarwal and Roth 2002]. However, there is no evaluation in the literature which mixes different resolutions of the input images to obtain a larger test set with different car sizes. We therefore restricted the car search to a single scale (because [Agarwal and Roth 2002] do not search over multiple scales). When applying the Bayesian Network to the original resolution images, the resulting detection rates were only slightly lower but incurred more false alarms. However, one would have to subject the competing approaches to the same test set containing different car sizes for a meaningful comparison. We conjecture that especially in the limit of very low image resolutions the scaleparts-based approach will prove most effective because unlike other approaches it also incorporates holistic information (the full car scalepart was shown to be important for low resolution detection).

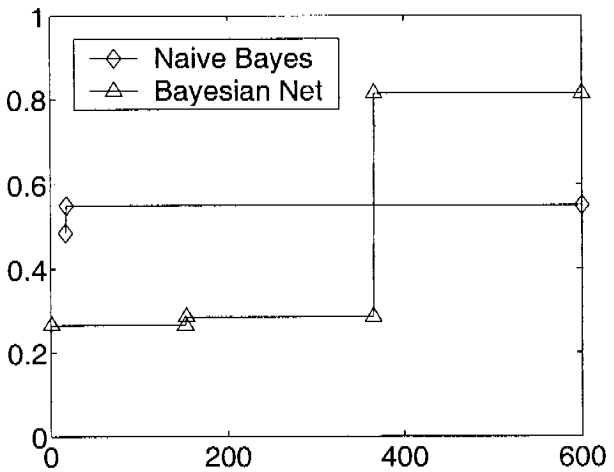

Figure 7.10: *Car detection at all resolutions ("lores", "original" and "hires" images combined, containing 600 cars). The ROC curves are based on the Naive Bayes classifier of figure 7.8 and the Bayesian Network of figure 7.9. Using the Bayesian Network, 82% of all cars could be found at a cost of about 390 false positives. This corresponds to less then 1 false alarm every other image. Finding the 200 smallest cars (60×24 pixels) in this data set is especially hard, but necessary to exceed a detection rate of 2/3, i.e. 67%. The Naive Bayes classifier's detection rate stays continuously below 60%.*

7.4 Discussion

This chapter has introduced a scaleparts-based car detector for the detection of sideviews of cars based on the scalepart framework of chapters 5 and 6. Individual scaleparts have been learned as wheel, half car and full car detectors using AdaBoost. In case of the full car, the most important selected features seem to be the car's ground shadow from the chassis and the frontal mudguard also including part of the wheel. The half car is mainly detected again through features on mudguard and wheel but the detector generally pays more attention to wheels. The wheel detector uses rotated and center-surround features to capture parts of the rim, and the tyre as well as the specific contrast between axle and tyre. Interestingly, all detectors even at the smallest resolution incorporate wheel parts in early cascade stages but they do not include regions of the upper car half, such as windows or roof.

The detection capabilities of these car scaleparts show complementary strengths with regard to different image resolutions. However, they also exhibit differences in discriminance levels. Half car and wheel detectors cause significantly fewer false alarms than the full car detector. However, the latter is the most important cue at low image resolutions. The combined detector – again based on a Bayesian Network

– therefore adjusts the influence of individual scaleparts depending on the estimated global resolution situation of the current image.

It is likely that a more discriminant full car detector can be trained especially when using a higher training resolution. However, initial experiments indicate that this also requires a much large number of preprocessed high quality training data. Also, it might be helpful to add additional complementary cues that can support the full car detector at low resolutions. These could be scene-triggered contextual priors [Torralba 2001], or other contextual cues, such as "streets". To smooth the classification function represented by the Bayesian Network, one would have to introduce discrete value ranges (instead of binary nodes) or resort to continuous node representations. One could imagine a hybrid Bayesian Network where all observed nodes are continuous and the query node is binary to indicate the presence or absence of the target object.

Seite Leer /
Blank leaf

8

Conclusion and Perspective

This thesis has developed new approaches and algorithms for object detection. Object detection is a standard discipline in computer vision both for its many applications and for its important role in artificial intelligence research. Nevertheless object detection capabilities in machines are far from being comparable to human or mammal vision.

This is equally true for the specific task of face detection – even though within computational object detection research it is probably the most advanced subfield. The general guideline for this thesis was to look for new ways to improve face detection first and then try to generalize to other object classes. For improving face detection one possible way is to search for a better representation that can be used within the traditional detection paradigm. However, since research on face detection has seemingly reached a plateau the quest for improvements seems to call for a more radical approach.

The approach taken in this thesis was to first identify, develop and evaluate complementary cues for face detection. Of interest are cues which successfully contribute to the detection task and have so far been overlooked or not evaluated thoroughly. They can then be combined with traditional successful face detection methods. Two cues are closely investigated, one is based on finding skin patches the other is based on finding a face's local context.

While the skin patch detector is rather specific to human and face detection it turns out that the local context detector allows for a fruitful generalization to other object classes. As has been shown empirically in this thesis local context becomes particularly useful as the level of available image detail decreases. Thus, just as local context contributes to detection at a smaller scale we can form a whole set of scale-dedicated object parts to accommodate different situations of available image detail.

Such scale-dedicated object parts, in short scaleparts, can be efficiently detected using boosted classifier cascades trained from data using AdaBoost. A scalepart-based

object detector can then be learned which combines individually detected scaleparts. This thesis proposes to use Bayesian Networks for the mentioned combination and evaluates the approach in face and car detection tasks.

8.1 Contributions

This thesis contributes to the field of object detection in three distinct ways.

Skin Patch Detection In skin patch detection the combination with shape outperforms purely color-based approaches.

Chapter 3 has developed a novel skin patch detector based on a combination of color and shape models. Empirical evaluations on a large body of data demonstrate the detector's effectiveness. In particular, these evaluations show that the combination with shape outperforms purely color-based approaches. These results are based on a comprehensive statistical skin color model which captures the variation of different skin complexions. The additional shape constraints help increase the discriminance with respect to background of skin-like color.

Another important result concerns the relation between skin patch detection and appearance-based face detection. It turns out that the skin cue can increase the face detection rate significantly due to its strong invariance to pose changes. Alternatively, it can greatly improve precision when used to post-filter face hypotheses that have been extracted based on appearance.

Local Context Inspired by Torralba and Sinha's experiments with human individuals chapter 4 has demonstrated the effectiveness of using local context for computational vision.

The importance of local context increases as imaging conditions deteriorate. The presented analysis has focused on available image resolution where local context increases detection robustness at small image scales. Robust operation at low image resolutions (i.e. small images) is also attractive because it greatly reduces object search time. And unlike skin detection, the approach does not require color information which makes it widely applicable.

Scalepart-based Object Detection Robust detection over wide resolution ranges can be achieved using scale-specific parts ("scaleparts") that cover various extents of the target object.

The concept of scaleparts (chapter 5) is directly motivated by the local context concept in connection with smaller image scales. The specific choice of parts is not uniquely defined but motivated by the need to accomodate different levels

of available image detail. The approach emphasizes that individual scaleparts be discriminant with respect to the background which is implemented using AdaBoost. Part combination is implemented using Bayesian Networks which allow for learning part interactions from data. A side product is the first local to global (entire body) detector for pedestrians (for cars, such a global to local detector has been proposed by [Garg *et al.* 2002]).

8.2 Additional Remarks

A number of additional insights have been gained from this work which could be useful for other researchers in the field.

First, our experiments in face and car detection heavily rely on fast-to-compute integral image features. In particular the same feature types are used for *all* part detectors developed in chapters 5 and 7. Despite their simplicity we found that these features coupled with AdaBoost learning yield remarkably discriminant classifiers for parts as diverse as faces, car wheels, lower bodies or car halves. We believe that this feature set is quite rich and will be directly applicable to many other object classes.

Second, we noticed that the local context detector (chapter 4) which detects upper bodies of people is of immediate use for many popular vision applications. This ranges from conversational interfaces, to head-tracking in video-conferences to people detection in surveillance videos. The local context detector is also distinguished by its high level of discriminance. The achieved discriminance level seems to outperform Mohan's upper body detector [Mohan *et al.* 2001]. In their work the upper body was the least discriminant cue of all detected body parts.

Third, the implementation of the scaleparts idea in chapter 5 brings together two successful Machine Learning techniques: AdaBoost learning and Bayesian Networks. In this marriage of a discriminative and generative method we aim at combining the best of both worlds. The promise of discriminative learning is to make the best use of a limited amount of training data by directly focusing on the decision boundary. For cue combination Bayesian Networks provide an intuitive, interpretable and modular way both for encoding expert domain knowledge and for learning cue interactions from data.

8.3 Perspective

There are several natural extensions of this work:

Other Object Classes It would be interesting to test the generality of scalepart-based object detection by applying it to other rigid objects such as motorbikes, boats, airplanes, etc.

Cue Fusion The current combiner is based on a discrete-node Bayesian Network. This is motivated by the original AdaBoost formulation which returns discrete class labels. However, AdaBoost variants with confidence-rated predictions and a Bayesian Network with continuous nodes might lead to more accurate detection.

Learning At least three additional learning problems naturally arise from this work:

First, our scalepart learning currently relies on the user for initializing the number of parts and their coarse positions and extents. We can as well imagine an unsupervised algorithm where scaleparts emerge directly from the training data. Clustering would be a natural way but the difficulty is in the complexity of the data which is governed by the large number of training instances and the possible resolutions, positions and sizes of part candidates.

Second, in the current scheme each scalepart detector cascade is individually learned and applied. A more efficient way would be to learn a classification tree which shares features among scaleparts in the first cascade stages and differentiates between different scaleparts at later stages.

Third, instead of separately learning part detectors and part integration one could try and learn everything at the same time. A successful example of such simultaneous learning is [Viola *et al.* 2003]. In this work, they integrate motion features with appearance in one learning loop.

Features Integral image features so far have been successfully applied for the detection of cars (side views), pedestrians (frontal), faces (frontals and profiles), facial features (for mouth tracking) [Lienhart *et al.* 2003b] and brand logos [Lienhart *et al.* 2003a]. It would be interesting to empirically investigate the limitations of these features. One could try to detect wiry objects [Carmichael and Hebert 2003], desk items [Mahamud and Hebert 2003] or animals, for example and try to push the limits by adding new fast-to-compute feature types.

Context In this work, we have focused on *local* context. Contrastingly, [Torralba 2001] have proposed global contextual cues based on a holistic image analysis. These concepts could be easily integrated for scaleparts as new observable nodes within the Bayesian Network. There might also exist intermediate contextual levels between global and local which are useful for object detection.

List of Figures

List of Tables

Seite Leer /
Blank leaf

Bibliography

[Agarwal and Roth 2002] S. Agarwal and D. Roth. Learning a sparse representation for object detection. pages 113–130, 2002.

[Angelopoulou 2001] Elli Angelopoulou. Understanding the color of human skin. In *SPIE Conference on Human Vision and Electronic Imaging VI*, pages 243–251, 2001.

[Biederman *et al.* 1982] I. Biederman, R.J. Mezzanotte, and J.C. Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. In *Cognitive Psychology*, volume 14, pages 143–177, 1982.

[Biederman 1987] I. Biederman. Recognition-by-components:a theory of human image interpretation. In *Psychological Review*, volume 94, pages 115–147, 1987.

[Carmichael and Hebert 2003] O. Carmichael and M. Hebert. Shape-based recognition of wiry objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03)*, pages 401–408, 2003.

[Choudhury *et al.* 2002] T. Choudhury, J. Rehg, V. Pavlovic, and A. Pentland. Boosting and structure learning in dynamic bayesian networks for audio-visual speaker detection. In *International Conference on Pattern Recognition (ICPR'02)*, 2002.

[Colmenzrez and Huang 1997] A. Colmenzrez and T. Huang. Face detection with informationbased maximum discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 782–787, 1997.

[Cooper and Herskovitz 1992] G. Cooper and E. Herskovitz. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 1992.

[Corn 1996] J. J. Corn. Elektro and sparko, westinghouse historical collection, 1939 new york world's fair. In *Yesterday's Tomorrows*. Johns Hopkins University Press, 1996.

[Cowell *et al.* 1999] Cowell, Dawid, Lauritzen, and Spiegelhalter, editors. *Probabilistic networks and expert systems*. Springer Verlag, 1999.

[Cristinacce and Cootes 2003] David Cristinacce and Tim Cootes. A comparison of two real-time face detection systems. In *Fourth IEEE International Workshop*

on Performance Evaluation of Tracking and Surveillance (PETS-ICVS). IEEE, March 31 2003.

[Depoortere *et al.* 2002] Vincent Depoortere, Jeroen Cant, Bram Van den Bosch, Jan De Prins, Fik Fransens, and Luc Van Gool. Efficient pedestrian detection: a test case for svm based categorization. In *Cogvis Workshop 2002, Zurich*, 2002.

[Fleck *et al.* 1996] Margaret M. Fleck, David A. Forsyth, and Chris Bregler. Finding naked people. In *ECCV (2)*, pages 593–602, 1996.

[Fleuret and Geman 2001] Francois Fleuret and Donald Geman. Coarse-to-fine face detection. *Int. Journal of Computer Vision*, 41(1):85–107, 2001.

[Forsyth and Fleck 1997] D. Forsyth and M. Fleck. Body plans. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 678–683, 1997.

[Franke *et al.* 1998] U. Franke, D. Gavrila, S. Görzig, F. Lindner, F. Paetzold, and C. Wöhler. Autonomous driving goes downtown. *Intelligent Systems*, 13(6):40–48, 1998.

[Freund and Schapire 1996] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning, Procs. of the 13th Int. Conf.*, pages 148–156, 1996.

[Friedman *et al.* 1997] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 1997.

[Garg *et al.* 2002] A. Garg, S. Agarwal, and T. Huang. Fusion of global and local information for object detection. In *International Conference on Pattern Recognition (ICPR'02)*, volume 3, pages 723–726, 2002.

[Hjelmas and Low 2001] Erik Hjelmas and Boon Kee Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.

[Jebara and Pentland] Tony Jebara and Alex Pentland. Parametrized structure from motion for 3d adaptive feedback tracking of faces. *IEEE Conference on Computer Vision and Pattern Recognition.*

[Jones and Rehg 1999] Michael J. Jones and James M. Rehg. Statistical color models with application to skin detection. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 274–280, 1999.

[Kittler *et al.* 1998] J. Kittler, M. Hatef, R. Duin, and J. Matas. On combining classifiers. *PAMI*, 20(3):226–239, March 1998.

[Kjeldsen and Kender 1996] Rick Kjeldsen and John Kender. Finding skin in color images. In *2nd IEEE International Conference on Automatic Face and Gesture Recognition (FG'96)*, pages 312–317, 1996.

[Lew and Huijsmans 1996] M. Lew and N. Huijsmans. Information theory and face detection. In *International Conference on Pattern Recogntion*, pages 601–610, 1996.

[Lienhart *et al.* 2002a] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. Mrl technical report, Intel Labs, 2002.

[Lienhart *et al.* 2002b] Rainer Lienhart, Luhong Liang, and Alexander Kuranov. An extended set of haar-like features for rapid object detection. Technical Report MRL, Intel Research, June 2002.

[Lienhart *et al.* 2003a] Rainer Lienhart, Alexander Kuranov, and Vadim Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *DAGM'03*, pages 297–304, Magdeburg, Germany, September 2003.

[Lienhart *et al.* 2003b] Rainer Lienhart, Luhong Liang, and Alexander Kuranov. A detector tree of boosted classifiers for real-time object detection and tracking. In *IEEE ICME2003*, July 2003.

[Mahamud and Hebert 2003] Shyjan Mahamud and Martial Hebert. Minimum risk distance measure for object recognition. In *International Conference on Computer Vision*, 2003.

[Mohan *et al.* 2001] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):349–361, 2001.

[Montemerlo *et al.* 2002] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, and V. Verma. Experiences with a mobile robotic guide for the elderly. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 587–592, 2002.

[Ninomiya *et al.* 1995] Y. Ninomiya, S. Matsuda, M. Ohta, Y. Harata, and T. Suzuki. A real-time vision for intelligent vehicles. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 315–320. IEEE, 1995.

[Papageorgiou and Poggio 2000] C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, June 2000.

[Papageorgiou *et al.*] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *International Conference on Computer Vision*, pages 555–562.

[Papageorgiou *et al.* 1998] C. Papageorgiou, T. Evgeniou, and T. Poggio. A trainable pedestrian detection system. In *Proceedings of the IEEE International Conference on Intelligent Vehicles*, pages 241–246, Stuttgart, 1998.

[Pavlovic *et al.* 2001] V. Pavlovic, A. Garg, J. Rehg, and T. Huang. Multimodal speaker detection using error feedback dynamic bayesian networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'01)*, pages 34–43, 2001.

[Rehg *et al.* 1999] J. Rehg, K. Murphy, and P. Fieguth. Vision-based speaker detection using bayesian networks. In *Proceedings of the International Conference on Computer Vision*, pages 110–116, 1999.

[Rein-Lien Hsu 2002] Anil K. Jain Rein-Lien Hsu, Mohamed Abdel-Mottaleb. Face detection in color images. *PAMI*, pages 696–706, January 2002.

[Remagnino *et al.* 1997] P. Remagnino, A. Baumberg, T. Grove, D. Hogg, T. Tan, A. Worrall, and K. Baker. In integrated traffic and pedestrian model-based vision system. In *Proceedings of the British Machine Vision Conference*, 1997.

[Rowley *et al.* 1998] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

[Schapire and Singer 1999] Robert E. Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37(3):297–336, 1999.

[Schapire *et al.* 1998] Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. In *Annals of Statistics*, volume 26, pages 1651–1686, 1998.

[Schiele and Crowley 1998] B. Schiele and J.L. Crowley. Transinformation for active object recognition. In *Sixth International Conference on Computer Vision (ICCV'98)*, pages 249–254, January 1998.

[Schiele and Waibel 1995] B. Schiele and A. Waibel. Gaze-tracking based on face–color. In *IWAFGR 95, International Workshop on Automatic Face–and Gesture–Recognition*, pages 344–349, June 1995.

[Schneiderman and Kanade 2000] Henry Schneiderman and Takeo Kanade. A statistical method for 3d object detection applied to faces and cars. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1751, 2000.

[Schneiderman 2000] Henry Schneiderman. A statistical approach to 3d object detection applied to faces and cars. Technical Report CMU-RI-TR-00-06, Carnegie-Mellon University, 2000.

[Schneiderman 2003] H. Schneiderman. Learning statistical structure for object detection. In *International Conference on Analysis of Images and Patterns*, pages 434–441, 2003.

[Sinha 2002] Pawan Sinha. Qualitative representations for recognition. In H.H. Buelthoff et al., editor, *Biologically Motivated Computer Vision (BMCV)*, pages 249–262, 2002.

[Stoerring *et al.* 1999] Moritz Stoerring, Hans J. Andersen, and Erik Granum. Skin colour detection under changing lighting conditions. In *7th International Symposium on Intelligent Robotic Systems'99*, pages 187–195, July 1999.

[Strat and Fischler 1991] T. M. Strat and M. A. Fischler. Context-based vision: recognizing objects using information from both 2-d and 3-d imagery. *PAMI*, 13(10):1050–1065, 1991.

[Sung and Poggio 1994] Kah-Kay Sung and Tomaso Poggio. Example-based learning for view-based human face detection. Technical Report AI Memo no. 1521, MIT, December 1994.

[Sung and Poggio 1998] Kah-Kay Sung and Tomaso Poggio. Example-based learning for view-based human face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(1), January 1998.

[Terrillon and Akamatsu 2000] Jean-Christophe Terrillon and Shigeru Akamatsu. Comparative performance of difference chrominance spaces for color segmentation and detection of human faces in complex scene images. In *Proc. of the 4th International Conference on Automatic Face and Gesture Recognition*, pages 54–61, 2000.

[Thrun *et al.* 2000] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Probabilistic algorithms and the interactive museum tour-guide robot minerva. *International Journal of Robotics Research*, 19(11):972–999, 2000.

[Torralba and Sinha 2001] Antonio Torralba and Pawan Sinha. Detecting faces in impoverished images. In *AI Memo 2001-028, CBCL Memo 208*, 2001.

[Torralba 2001] Antonio Torralba. Contextual modulation of target saliency. In *Advances in Neural Information Processing Systems*, pages 1303–1310, 2001.

[Ullman *et al.* 2002] S. Ullman, M. Vidal-Naquet, and E. Sali. Visual features of intermediate complexity and their use in classification. *Nature Neuroscience*, 5(7):682–687, 2002.

[Vailaya *et al.* 1999] Aditya Vailaya, HongJiang Zhang, and Anil Jain. Automatic image orientation detection. In *IEEE ICIP 1999*, pages 600-604, October 1999.

[Vidal-Naquet and Ullman 2003] M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *International Conference on Computer Vision*, 2003.

[Viola and Jones] Paul Viola and Michael J. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[Viola *et al.* 2003] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *International Conference on Computer Vision*, pages 734–741, 2003.

[Viola 1995] Paul A. Viola. Alignment by maximization of mutual information. Technical Report AITR-1548, Massachusetts Institute of Technology, mar 1995.

[Wachsmuth *et al.* 2000] S. Wachsmuth, G. Socher, H. Brandt-Pook, F. Kummert, and G. Sagerer. Integration of vision and speech understanding using bayesian networks. *Videre: A Journal of Computer Vision Research*, 1(4):62–83, 2000.

[Wang *et al.* 1997] James Ze Wang, Jia Li, Gio Wiederhold, and Oscar Firschein. System for screening objectionable images using daubechies' wavelets. In *International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services*, pages 20–30, 1997.

[Wells III *et al.* 1996] W.M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi–modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1(1):35–52, march 1996.

[Wren *et al.* 1997] Christopher Richard Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *PAMI*, 19(7):780–785, 1997.

[Yang *et al.*] Ming-Hsuan Yang, David Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

[Yang *et al.* 1998] Jie Yang, Weier Lu, and Alex Waibel. Skin-color modeling and adaptation. In *ACCV (2)*, pages 687–694, 1998.

[Zhao *et al.* 2000] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips. Face recognition: A literature survey. Technical Report CfAR Technical Report CAR-TR-948, University of Maryland, 2000.

[Z.Q. Zhang and Zhang 2002] S.Z. Li Z.Q. Zhang, L. Zhu and H.J. Zhang. Real-time multi-view face detection. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, volume 4, pages 67–81, May 2002.

Curriculum Vitae

Hannes Kruppa

Date of birth: 4th of October, 1974

Place of birth: Tübingen, Germany

Education:	1994–1996	Computer Science and Medicine, University of Tübingen, Germany
	1996–1998	Computer Science and Management, Swiss Federal Institute of Technology (ETH), Zürich
	1998–1999	Computer Science and Robotics, Carnegie-Mellon University (CMU), Pittsburgh USA
		Graduation with the degree *Dipl. Informatik-Ing. ETH.*

Occupation:	1999–2004	Research Assistant at ETH Zürich, Perceptual Computing and Computer Vision Group

---