Informatieselectie en -fusie in visuele systemen

Information Selection and Fusion in Vision Systems

Linda Tessens

Promotoren: prof. dr. ir. W. Philips, prof. dr. ir. H. Aghajan
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen

Vakgroep Telecommunicatie en Informatieverwerking
Voorzitter: prof. dr. ir. H. Bruneel
Faculteit Ingenieurswetenschappen
Academiejaar 2010 - 2011

ISBN 978-90-8578-385-5
NUR 958, 959
Wettelijk depot: D/2010/10.500/61

Universiteit Gent
Faculteit Ingenieurswetenschappen
Vakgroep Telecommunicatie en Informatieverwerking

Promotoren: Prof. Dr. Ir. Wilfried Philips
Prof. Dr. Ir. Hamid Aghajan

Universiteit Gent
Faculteit Ingenieurswetenschappen

Vakgroep Telecommunicatie en Informatieverwerking
Sint-Pietersnieuwstraat 41, B-9000 Gent, België

Tel.: +32-9-264.34.12
Fax.: +32-9-264.42.95

Voorzitter: Prof. Dr. Ir. Herwig Bruneel

Dit werk kwam tot stand in het kader van een mandaat Aspirant van het FWO (Fonds Wetenschappelijk Onderzoek - Vlaanderen).

Proefschrift tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen
Academiejaar 2010-2011

Members of the jury

Prof. dr. ir. Hamid Aghajan (Stanford University, advisor)
Prof. dr. ir. Bjorn De Sutter (Ghent University)
Prof. dr. ir. Richard Kleihorst (Vlaams Instituut voor Technologisch Onderzoek)
Prof. dr. ir. Adrian Munteanu (Vrije Universiteit Brussel)
Prof. dr. ir. Wilfried Philips (Ghent University, advisor)
Prof. dr. ir. Rik Van de Walle (Ghent University, chairman)
Prof. dr. ir. Peter Veelaert (Ghent University College, secretary)
Prof. dr. ir. Sabine Wittevrongel (Ghent University)

Affiliations

Research Group for Image Processing and Interpretation (IPI)
Interdisciplinary Institute for Broadband Technology (IBBT)
Department of Telecommunications and Information Processing (TELIN)
Faculty of Engineering
Ghent University

Sint-Pietersnieuwstraat 41
B-9000 Ghent
Belgium

Acknowledgements

This work would not have been possible without the support and help of many people.
I would like to thank my advisor Prof. Philips for his guidance, support and trust during the past years. The research climate he creates allowed me to pursue my interests freely, and his critical mind helped me to conduct my research thoroughly. Special thanks to my co-advisor Prof. Aghajan for the opportunity he offered me to stay at Stanford University during several months. This was an extremely instructive and inspiring time. Also many thanks for the fruitful collaboration that followed. Prof. Aghajan's suggestions and visions shaped this work. Furthermore I am thankful to the Flemish Fund for Scientific Research (FWO) for largely funding this research.
I would like to thank the members of my jury, Prof. Hamid Aghajan, Prof. Bjorn De Sutter, Prof. Richard Kleihorst, Prof. Adrian Munteanu, Prof. Wilfried Philips, Prof. Rik Van de Walle, Prof. Peter Veelaert, and Prof. Sabine Wittevrongel, for being in my thesis committee, and for their many comments and suggestions to improve this thesis.
My gratitude goes out to the ATP personnel at TELIN, and to all my colleagues at the Image Processing and Interpretation group at Ghent University, at the Wireless Sensor Networks Lab at Stanford University, at Vision Systems at Hogeschool Gent, and all those that I met at conferences. Thank you for the productive interaction and the helpful suggestions. I am especially thankful to Prof. Aleksandra Pižurica, who guided me through the first years of my PhD, and to Marleen Morbee, with whom I have successfully collaborated on a daily basis. Of course I also enjoyed the pleasant atmosphere at the TELIN department.
Finally a word of gratitude to my family and friends, who are behind me every step of the way.

Ghent, October 2010
Linda Tessens

Samenvatting

In onze maatschappij is het gebruik van beeld- en videodata zeer wijdverbreid geworden, en de geproduceerde hoeveelheid beeld- en videodata is navenant gegroeid. Omgaan met deze enorme datahoeveelheden is duur en technisch uitdagend. Bovendien is niet alle opgenomen data nuttig. Een groot deel is irrelevant (heeft geen belang) of redundant (bevat geen nieuwe informatie). Om de gigantische hoeveelheid data geproduceerd door visuele systemen tot handelbare proporties te reduceren, zijn technieken die irrelevantie en redundantie in de data verminderen van groot belang. Het filteren van relevante data en de samenvatting ervan omvat twee uitdagingen: irrelevante data identificeren zodat die kan verwijderd worden, en redundante data samenvatten zodat dezelfde informatie slechts één keer weerhouden wordt.
Vermindering van irrelevantie en redundantie kan in verschillende gradaties gerealiseerd worden:

- **Compressie:** een basisstap is de opgenomen beeld- of videodata - verliesloos of met verlies - te encoderen zodat de voorstellingsgrootte gecomprimeerd wordt.
- **Fusie:** de volgende stap is informatiefusie, waarbij gegevens van verschillende bronnen gecombineerd worden tot één enkel resulterend product. Dit bevat in het ideale geval alle informatie die interessant is voor de taak die men wenst uit te voeren.
- **Selectie:** bovendien kan men enkel informatie die interessant is voor de taak selecteren en de rest verwijderen.

Deze thesis behandelt fusie en selectie van informatie in visuele systemen. De ontwikkelde algoritmen evolueren van technieken voor de selectie en fusie van visuele data op pixelniveau tot methoden die het mogelijk maken op een hoger niveau van abstractie te redeneren over het belang van observaties en manieren om ze te combineren tot een bruikbaar resultaatproduct. We ontwikkelen technieken voor fusie en selectie van informatie in twee types beeldsystemen: lichtmicroscopen en netwerken van slimme camera's.
Een lichtmicroscoop heeft een beperkte scherptediepte. Daarom is het vaak onmogelijk om een beeld van een 3D object op te nemen waarin alle delen scherp zijn afgebeeld. Een standaardtechniek om de scherptediepte van de microscoop virtueel uit te breiden is een 'stapel' beelden op te nemen van het 3D object. De afstand tussen de beeldsensor en het object is in elk beeld anders. Zo verkrijgt men een aantal beelden die snedes genoemd worden, en waarin telkens een

ander deel van het object scherp is afgebeeld. Deze techniek leidt duidelijk tot een beeldstapel die zeer nuttige informatie bevat (scherpe afbeeldingen van alle delen van het object), maar helaas ook heel wat irrelevante informatie (wazige beeldgebieden) of redundante informatie (beeldgebieden die meermaals scherp afgebeeld zijn in de beeldstapel).

Wij stellen een techniek voor om alle interessante informatie in een beeldstapel te selecteren en te fuseren tot een enkel resulterend beeld dat alle delen van het object scherp weergeeft. Meer bepaald buiten we de richtingsgevoeligheid van de curvelettransformatie uit om fusieresultaten van hoge kwaliteit te verkrijgen, zowel voor echte microscoopdata als voor kunstmatig gegenereerde beeldstapels. We tonen dat het toevoegen van consistentie- en ruimtelijke gladheidscontroles over het algemeen leidt tot betere fusieresultaten. Voor echte testdata leidt het opleggen van deze voorwaarden tot een verminderd aantal artefacten in het gefuseerde beeld.

Ruis, aanwezig in alle beeldvormingssystemen, heeft een verstorend effect op de voorgestelde beeldfusietechniek. We stellen meerdere oplossingen voor om de invloed van ruis op het fusieproces te beperken. We tonen dat het opleggen van de veronderstellingen van ruimtelijke gladheid in en consistentie tussen de curveletdecompositiebanden een regularizerend effect heeft en de fusiekwaliteit bevordert. We wijzen ook op de alternatieve oplossing van het ontruizen van de curveletcoëfficiënten alvorens te fuseren.

Om een op curvelets gebaseerde ruisonderdrukkingstechniek te ontwikkelen, onderzoeken we de verschillen in statistisch gedrag tussen curveletcoëfficiënten die een significant ruisvrij signaal bevatten en die waarin geen interessant signaal aanwezig is. We ontwikkelen een ruisonderdrukkingsmethode voor curvelets die we *ProbShrinkCurv* noemen en die een aanpassing is van de op wavelets gebaseerde *ProbShrink* ruisonderdrukkingsmethode [Pižurica and Philips, 2006]. Daartoe maken we gebruik van de kennis opgedaan in onze statistische studie om een geschikte lokale activiteitsmaat te ontwerpen voor de nieuwe methode.

Ontruizen van de curveletcoëfficiënten van de ruizige snedes met *ProbShrinkCurv* alvorens ze te fuseren, verbetert het fusieresultaat aanzienlijk. De beste fusieresultaten worden verkregen wanneer ontruizing voor fusie gecombineerd wordt met een fusieproces waarin ruimtelijke gladheid en subband-consistentie opgelegd worden.

Cameranetwerken met overlappende gezichtsvelden vormen het tweede type visuele systemen dat we in deze thesis bestudeerd hebben. Omdat zulke netwerken verschillende zichten op dezelfde scène weergeven, hebben ze aanzienlijke voordelen tegenover een enkele camera met een vast gezichtspunt. Zo kunnen cameranetwerken bijvoorbeeld occlusieproblemen verhelpen; in gebaarherkenning kunnen aanwijzingen van verschillende gezichtspunten tot een robuustere beslissing leiden.

Recente hardwareontwikkelingen hebben de invoering van 'slimme' camera's mogelijk gemaakt. Dit zijn camera's waarin communicatie- en verwerkingshardware is geïntegreerd. Ze laten toe flexibelere en schaalbaardere netwerken te bouwen omdat de vereiste beeldverwerking verdeeld kan worden over de ca-

mera's. Het gezamenlijk verwerken van de uitvoerdata van de slimme camera's kan gebeuren in een basisstation of op een van de camera's.

Gegevensverwerking in een netwerk van slimme camera's brengt enkele specifieke uitdagingen met zich mee. De hardware die geïntegreerd is met de beeldsensor, wordt doorgaans speciaal ontworpen voor beeldverwerking (hoge graad van parallellisatie), wat een voordeel is, maar hij heeft ook enkele beperkingen wat betreft geheugen en rekenkracht. Als de hoeveelheid uitvoergegevens van de slimme camera's laag wordt gehouden, wordt draadloze operatie mogelijk. Dit is een voordeel voor de flexibiliteit van het systeem. Werking op batterijen met lange autonomie is in dit geval ook wenselijk.

De algoritmen ontwikkeld voor cameranetwerken in deze thesis zijn alle ontworpen rekening houdend met hun mogelijke implementatie in netwerken van slimme camera's, ofwel in hun huidige vorm, ofwel in een aangepaste, afgeslankte vorm. Daartoe is aandacht besteed aan aspecten zoals gegevenssnelheid en processorbelasting.

Wanneer de camera's in een netwerk dezelfde gebeurtenis of hetzelfde object vanuit verschillende gezichtspunten observeren, verhoogt dit niet enkel de hoeveelheid nuttige informatie. Een groot deel van de gegevens geproduceerd door het netwerk is redundant of zelfs irrelevant. We hebben twee hoofdbenaderingswijzen gevolgd om dit probleem aan te pakken: informatiefusie, waarbij relevante data van verschillende bronnen in een enkel resulterend product gecombineerd wordt, en informatieselectie, waarbij de gegevens die het waardevolst zijn voor een specifieke taak geïdentificeerd worden.

We stellen een nieuwe methode voor om bezettingsinformatie van verschillende camera's te fuseren om een 2D overzicht van de bezetting van een scène te verkrijgen. Dit 2D overzicht wordt een bezettingskaart genoemd. Deze methode is gebaseerd op fusie van vloerbezettingskaarten van individuele camera's m.b.v. de Dempster-Shafer-theorie van bewijsvoering. Experimenten en een vergelijking met de state of the art tonen duidelijke verbeteringen aan van de gefuseerde vloerbezettingskaarten wat betreft concentratie van het bezettingsbewijs rond daadwerkelijke personenposities. We demonstreren ook de doeltreffendheid van de voorgestelde methode in een cameranetwerk dat uit vier sensoren bestaat en dat in ware tijd opereert.

Om de implementatie van deze methode in een netwerk van slimme camera's te vergemakkelijken, onderzoeken we een alternatieve versie die slechts een lage gegevenssnelheid vereist en een lage processorbelasting met zich mee brengt. Deze versie vereist dat de personen in de scène voldoende groot in de camerabeelden verschijnen. Als dit het geval is kunnen camera's een compacte scanlijn van de gedetecteerde voorgrond doorsturen i.p.v. het ganse voorgrondbeeld.

Verder introduceren we een praktische methode om te bepalen welke deelverzameling van camera's in een netwerk van slimme camera's het beste zicht heeft op de personen in een scène en hun vorm. Het bestaat uit gedistribueerde en centrale processen. Om een geschikte overzichtscamera te kiezen houdt het algoritme rekening met het aantal gezichten gedetecteerd door elk van de camera's, en met de snelheid en de posities van de objecten t.o.v. de kijkrichting en

de kijkhoek van de camera's. Dit hoofdzicht wordt aangevuld met bijkomende zichten die de observatie uitbreiden en die toelaten de 3D vorm van de personen in de scène te reconstrueren. Om deze bijkomende zichten te selecteren gebruiken we de bezettingskaart als een ruwe benadering van de 2D vorm van de mensen in de scène.
Bovendien stellen we een cameraselectie-algoritme voor dat geschikt is voor ware-tijd-operatie. Experimentele resultaten tonen aan dat het voorgestelde algoritme een performantie heeft die dicht bij de optimale resultaten ligt. Ook worden twee verschillende netwerkoperatieprotocollen voorgesteld. Het eerste heeft als doel de observatiefrequentie van de sensoren te verhogen, het tweede vermindert de vertraging tussen observatie en beeldverzending. Experimentele resultaten tonen aan dat de voorgestelde protocollen de observatiefrequentie verhogen en de vertraging verminderen zonder de performantie van 3D vormreconstructie sterk te verminderen.
Een cruciale component in een doeltreffend cameraselectiesysteem is het kwantificeren van de bijdrage van een of meerdere camera's tot het vervullen van een taak. We beschrijven een nieuw, algemeen raamwerk om de kwaliteit te evalueren waarmee een deelverzameling van camera's een netwerktaak vervult. De voorgestelde geschiktheidsmaat is afgeleid van de Dempster-Shafer-theorie van bewijsvoering en kan toegepast worden op een breed gamma van computervisieproblemen.
Als demonstratietoepassing gebruiken we de maat om sensoren te selecteren in een cameranetwerk waarin meerdere objecten worden gevolgd. Deze methode is getest op duizenden beelden in verschillende omgevingen en laat toe personen te volgen met een dynamische selectie van slechts drie camera's met dezelfde nauwkeurigheid als wanneer steeds alle camera's (zeven, acht of tien) worden gebruikt. Wanneer met slechts twee camera's gevolgd wordt, is er slechts een lichte afname in performantie. De voorgestelde methode presteert duidelijk beter dan andere cameraselectiemethodes voor het volgen van personen.
Samengevat zijn de belangrijkste bijdragen van dit proefschrift:

- een nieuwe beeldfusiemethode om de scherptediepte uit te breiden van een optisch systeem zoals een lichtmicroscoop [Tessens et al., 2007a,b];

- een statistische studie van curveletcoëfficiënten, op basis waarvan we een nieuwe ontruizingsmethode hebben voorgesteld [Tessens et al., 2006b,c, 2008c]. Van deze ontruizingsmethode werd aangetoond dat ze de fusieresultaten op beeldstapels besmet met ruis verbetert;

- een nieuwe methode om vloerbezettingskaarten te berekenen in een cameranetwerk door de vloerbezetting van elke camera afzonderlijk te fuseren m.b.v. de Dempster-Shafer-theorie van bewijsvoering [Morbee et al., 2008, 2010a; Tessens et al., 2008b];

- een nieuwe methode om doeltreffend camerazichten te selecteren om mensen in een scene te observeren en hun 3D vorm te reconstruëren in een netwerk van slimme camera's [Lee et al., 2008; Tessens et al., 2008b];

- een nieuw algemeen raamwerk om de kwaliteit te kwantificeren waarmee in een netwerk een deelverzameling van camera's een netwerktaak vervult [Tessens et al., 2010].

In totaal leidde het onderzoek gedurende dit doctoraat tot twee publicaties in internationale tijdschriften met collegiale toetsing [Morbee et al., 2010a; Tessens et al., 2008c], een ingediende [Tessens et al., 2010] en een tijdschriftpublicatie in voorbereiding [Morbee et al., 2010b]. Er is ook een octrooiaanvraag ingediend [Morbee and Tessens, 2010]. Bovendien werden dertien conferentieartikels gepubliceerd in internationale conferenties [Lee et al., 2008; Morbee et al., 2007a,b, 2008, 2009; Soleimani et al., 2010; Tessens et al., 2006a,b,c, 2007a,b, 2008b, 2009].

Summary

In our society the use of image and video data has become very widespread, and the produced amount of image and video data has grown accordingly. Handling these huge amounts of data is costly and technically challenging. Moreover not all of the recorded data is useful. A large portion is irrelevant - it is of no importance - or redundant - it does not provide new knowledge. To reduce the huge amount of data produced by imaging systems to workable proportions, techniques that decrease irrelevance and redundancy in the data are of paramount importance. The process of filtering out the relevant data and summarizing it encompasses two challenges: identifying irrelevant data such that it can be discarded, and summarizing redundant data such that the same information is only retained once.
Irrelevance and redundancy reduction can be realized in different gradations.

1. **Compression:** a basic step is to - losslessly or lossily - encode the recorded image or video data such that its representation size is compressed.

2. **Fusion:** the next step is information fusion, which combines data from different sources into a single output product. This ideally contains all information of interest to the task at hand.

3. **Selection:** additionally, one can select only information of interest to the task at hand and discard the rest.

This thesis deals with fusion and selection of information in visual systems. The developed algorithms evolve from techniques for visual data selection and fusion at the pixel level to methods for reasoning about the importance of observations and ways of combining them into a useful output product at a higher level of abstraction. We develop techniques for effective fusion and selection of information in two types of imaging systems: conventional light microscopes and smart camera networks.
A conventional light microscope has a limited depth of field. For this reason, it is often not possible to acquire an image of a 3D object in which all parts of the object appear in focus. A standard technique to virtually extend the depth of field of a microscope is to record an image 'stack' of the 3D object. The distance between the image sensor and the object is varied in each image, such that a set of images called slices is obtained in which each time a different part of the object is in focus. Clearly this technique results in an image stack, which contains very useful information (sharp images of all object parts), but

unfortunately also a lot of irrelevant information (blurred image regions) or redundant information (image regions that are imaged sharply in several slices of the stack).

We propose a technique to select and fuse all information of interest in an image stack for depth of field extension into a single output image that contains all parts of the object in focus. More precisely we exploit the directional sensitivity of the curvelet transform to produce high quality fusion results, both on real microscopy data and on artificially generated image stacks. We show that adding consistency and spatial smoothness checks to this curvelet-based fusion method generally leads to better fusion results. For real test data, imposing these constraints leads to a reduced number of artifacts in the fused image.

Noise, present in all image capturing systems, has a disturbing effect on the proposed image fusion technique. We propose several solutions to temper its influence on the fusion process. We show that imposing the assumptions of spatial smoothness within and consistency between the curvelet decomposition sub-bands has a regularizing effect and improves the fusion quality. We also point out that denoising the slices in the curvelet domain prior to fusion is an alternative solution.

In order to develop a curvelet-based denoiser, we investigate the differences in statistical behavior between curvelet coefficients containing a significant noise-free component and those in which no signal of interest is present. We develop a denoising method for curvelets called *ProbShrinkCurv*, which is an adaptation of the wavelet-based *ProbShrink* denoising method [Pižurica and Philips, 2006]. To this end, we put the knowledge gained from our statistical study to use in the design of an appropriate local spatial activity indicator (LSAI) for this new method.

Using *ProbShrinkCurv* to denoise the curvelet coefficients of noisy slices prior to fusion improves the fusion result considerably. The best fusion results are obtained when denoising prior to fusion is combined with a fusion process in which spatial smoothness and sub-band consistency constraints are imposed.

Camera networks with overlapping fields of view are the second type of visual systems that we have treated in this PhD dissertation. Because such networks present different views on the same scene, they have substantial advantages over a single fixed viewpoint camera. E.g., in scene monitoring, camera networks can alleviate occlusion problems; in gesture recognition, cues coming from different viewpoints can lead to a more robust decision; in free viewpoint television, the quality of the rendered intermediate views benefits from a larger number of cameras.

Recent hardware developments have made the introduction of 'smart cameras' possible. These are cameras with on-board processing and communication hardware. They allow for the construction of more flexible and scalable camera networks because the required image processing can be distributed over the cameras. The collaborative processing of the output data of the smart cameras can take place either in a base station or on one of the cameras.

Data processing in a smart camera network entails some specific challenges.

The hardware embedded with the image sensor is usually especially designed for image processing (high degree of parallelization), which is an advantage, but it also has some limitations in terms of memory and processing power. If the amount of output data of the smart cameras is kept low, wireless operation becomes possible. This is a huge advantage for the flexibility of the system. Battery operation is in this case also desirable.

The algorithms for camera networks developed in this thesis are all designed taking into account their possible implementation in smart camera networks, either as they are or in a modified, more light-weight form. To this end, attention has been paid to issues such as data rates and computational load.

When the cameras in a network observe the same event or subject from different viewing perspectives, this not only increases the amount of useful information. A large part of the data produced by the network is redundant or even irrelevant. We have followed two main approaches to solve this problem: information fusion, which combines relevant data from different sources into a single output product, and information selection, which identifies which data is most valuable for a specific task.

We propose a new method for fusing occupancy data from different cameras to obtain a 2D overview of the occupancy of a scene, called an occupancy map. This method is based on Dempster-Shafer based fusion of single view ground occupancy maps. Experiments and a comparison with the state-of-the-art show clear improvements in the fused ground occupancy maps in terms of concentration of the occupancy evidence around ground truth person positions. We also demonstrate the effectiveness of the proposed method in a four camera network operating in real time.

To facilitate the implementation of this method in smart camera networks, we modify it into a low data rate and low load version. This version requires that the persons in the scene appear sufficiently large in the camera views. If this is the case, cameras can send only scan-lines of the detected foreground, not the full foreground image.

Furthermore we introduce a practical method to determine which sensor subset in a smart camera network has the best view on the persons in a scene and their shape. It consists of distributed and central processes. To choose an appropriate key camera the algorithm takes into account the number of faces detected by each of the cameras, and the velocity and positions of the objects relative to the viewing direction and viewing angle of the cameras. This principal view is complemented with additional views, which extend the observation and which allow reconstructing the 3D shape of the people in the scene. To select these additional views we use the occupancy map as a crude 2D shape approximation of the people in the scene.

Moreover, we propose a greedy camera selection algorithm for real time network operation. Experimental results show that the proposed algorithm provides a performance very close to the optimal results. Also, two different network operation protocols are proposed. The first scheme aims to improve the sensor observation frequency and the second scheme decreases the delay between

view observation and image transmission. Experimental results demonstrate that the proposed protocols improve observation frequency and latency without much degrading the performance of 3D shape reconstruction.
A crucial component in an effective camera selection system is quantifying the contribution of one or more cameras to the accomplishment of a task. We present a novel, general framework to evaluate the quality with which a subset of cameras accomplishes a network task. The proposed set suitability value is derived from the Dempster-Shafer theory of evidence and can be applied to a wide range of vision problems.
As a proof of concept, we use it for sensor selection in a camera network in which multiple targets are tracked. This method has been tested on thousands of frames in different environments and allows to track persons using a dynamic selection of as little as three cameras with the same accuracy as when using all cameras (seven, eight or ten) all the time. When tracking with two cameras, there is only a minor performance drop. The proposed method clearly outperforms other camera selection schemes for tracking.
To summarize, the main contributions of this thesis are:

- a novel image fusion method to extend the depth of field of optical systems such as conventional light microscopes [Tessens et al., 2007a,b];
- a statistical study of curvelet coefficients, based on which we have presented a novel denoising method [Tessens et al., 2006b,c, 2008c]. This denoising method has been shown to improve fusion results on image stacks that are contaminated with noise;
- a novel method to calculate ground occupancy maps in a camera network by fusing ground occupancies from each view separately according to the Dempster-Shafer theory of evidence [Morbee et al., 2008, 2010a; Tessens et al., 2008b];
- a novel method to effectively select camera views for observing people in a scene and reconstructing their 3D shape in a network of smart cameras [Lee et al., 2008; Tessens et al., 2008b];
- a novel general framework to quantify the quality with which in a network a subset of cameras accomplishes a network task [Tessens et al., 2010].

In total, the research during this PhD resulted in two publication in international peer-reviewed journals [Morbee et al., 2010a; Tessens et al., 2008c]. One article is under review [Tessens et al., 2010] and one in preparation [Morbee et al., 2010b]. A patent application has been submitted [Morbee and Tessens, 2010]. Furthermore thirteen conference papers have been published in the proceedings of international conferences [Lee et al., 2008; Morbee et al., 2007a,b, 2008, 2009; Soleimani et al., 2010; Tessens et al., 2006a,b,c, 2007a,b, 2008b, 2009].

Table of Contents

1

Introduction

1.1 Introduction

In the last decades image and video data have developed into a convenient and widely used aid to assist people in a multitude of application areas: surveillance, video conferencing, medical care, traffic monitoring, etc.
As visual information becomes more and more widespread, the produced amount of image and video data grows accordingly. Handling these huge amounts of data is costly and technically challenging. Moreover not all of the recorded data is useful. A large portion is irrelevant - it is of no importance - or redundant - it does not provide new knowledge.
To reduce the huge amount of data produced by imaging systems to workable proportions, techniques that decrease irrelevance and redundancy in the data are of paramount importance. Indeed, the transmission, storage and processing of irrelevant and/or redundant data only leads to wasted resources (transmission bandwidth, storage capacity, processing power). The process of filtering out the relevant data and summarizing it encompasses two challenges:

- identifying irrelevant data such that it can be discarded or not processed;
- summarizing redundant data such that the same information is only retained once.

Note that irrelevance and redundancy have to be defined with respect to the information we wish to extract from the visual data, i.e., the task we want to accomplish. Consider the example of two cameras with overlapping fields of view. Part of their images display the same region of the scene twice. If a human observer wishes to see what is happening in the scene, the second view of the same region is redundant, and only the image region in which a new part of the scene is visible is of interest. However, if we use the two images to estimate a stereoscopic 3D reconstruction of the scene, it is exactly the overlapping part of the views that interests us. The rest of the images are useless for stereoscopic 3D reconstruction.
Irrelevance and redundancy reduction can be realized in different gradations.

1. **Compression:** a basic step is to - losslessly or lossily - encode the recorded image or video data such that its representation size is compressed. Obviously encoding each video stream will reduce the necessary transmission bandwidth and storage capacity in our two camera example. It will however not reduce the computational burden of processing the data.

2. **Fusion:** a next step is information fusion, which combines data from different sources into a single output product. This ideally contains all information of interest to the task at hand. If in our two camera example we are interested in a 3D reconstruction of the scene, this fusion product can, e.g., be the depth map of the part of the scene which is visible in both views.

3. **Selection:** additionally, one can select only information of interest to the task at hand and discard the rest. In our two camera example this can be achieved by selecting regions of interest in one or both of the camera images.

Data fusion and selection always entail the risk of losing information of interest, either by mistakenly deleting it, or in the fusion case by operations inherent to the combination process. A big research challenge is to develop data fusion and selection techniques that minimize this loss.
In this thesis we develop techniques for effective fusion and selection of information in two types of imaging systems: conventional light microscopes and smart camera networks. The choice for these two imaging systems has grown organically during the course of this PhD. More precisely, the opportunity to perform research on smart camera networks arose from the collaboration with Prof. Aghajan. He kindly offered the possibility of a research stay at the Wireless Sensor Networks Lab at Stanford University.
A conventional light microscope has a very limited depth of field. For this reason, it is not always possible to acquire an image of a 3D object with it in which all parts of the object appear in focus. Instead, a stack of images is captured in which each time a different part of the object is imaged sharply. Such a stack, which in practical applications contains tens of images, is not user-friendly for visual inspection. It is therefore desirable to 'extend' the depth of field of the microscope by fusing all information of interest in the images (here: sharp object parts) into one single output image, which contains a sharp view on every object part. We propose a novel method for the fusion of such image stacks and we also look into depth of field extension for noisy input images.
The second category of imaging systems considered in this work are smart camera networks. Recent years have seen an explosive growth of the number of cameras deployed in our society. From surveillance cameras observing public spaces over IP cameras showing the weather conditions in skiing areas to web cameras for communicating over the Internet, cameras are everywhere. We call a camera *smart* when it has on-board image processing and communication hardware. A number of cameras form a *network* when their video data is

at some point processed jointly - by a person or by a machine - to extract the desired information from it. In this thesis we present a novel method to fuse ground occupancy maps computed from a set of overlapping camera views. Furthermore we develop data selection techniques that automatically determine which camera views are most suited for a particular task. The tasks covered in this thesis are visual scene inspection and people tracking.

1.2 Contributions and Publications

The main novelties and contributions presented in this thesis are:

- a novel image fusion method to extend the depth of field of optical systems such as conventional light microscopes. This method uses the curvelet transform to distinguish between in-focus and blurred image regions. The curvelet transform is a wavelet-like image transformation that decomposes the image in several frequency scales and orientations. Because of its high directional sensitivity, it is well suited to identify high frequency image content. Using this method we have improved image fusion results for depth of field extension in terms of PSNR by several dBs.
  This research has been published at ICASSP [Tessens et al., 2007b] and at the SPS-Darts conference [Tessens et al., 2007a].

- a statistical study of curvelet coefficients, distinguishing between two classes of coefficients: those that contain a significant noise-free component, which we call "signal of interest", and those that do not. By investigating the marginal statistics, we have developed a prior model for curvelet coefficients. The analysis of the joint intra- and inter-band statistics has enabled us to develop an appropriate local spatial activity indicator for curvelets. Finally, based on our findings, we have presented a novel denoising method, inspired by a recent wavelet domain method *ProbShrink*. The new method outperforms its wavelet-based counterpart and produces results that are close to those of state-of-the-art denoisers. This work has led to one journal publication ([Tessens et al., 2008c]) and several conference publications ([Tessens et al., 2006b,c]).
  This denoising method is used to improve fusion of image stacks that are contaminated with noise.

- a novel method to calculate ground occupancy maps with a set of calibrated and synchronized cameras. In particular, we have proposed Dempster-Shafer based fusion of the ground occupancies computed from each view separately. The method yields very accurate occupancy detection results and in terms of concentration of the occupancy evidence around ground truth person positions it outperforms the state-of-the-art probabilistic occupancy map method and fusion by summing.
  This method has been published as a letter in [Morbee et al., 2010a]. Preparatory work has been published in [Morbee et al., 2008; Tessens et al., 2008b].

- a novel method to effectively select camera views for observing people in a scene and reconstructing their 3D shape in a network of smart cameras. Within a network, the contribution of a camera to the observation of a scene depends on its viewpoint and on the scene configuration. This is a dynamic property, as the scene content is subject to change over time. The proposed selection is based on the information from each camera's observations of persons in a scene, and only low data rate information is required to be sent over wireless channels since the image frames are first locally processed by each sensor node before transmission. The selected set of views constitutes a significantly more efficient scene representation than the totality of the available views. This is of great value for the reduction of the amount of image data that needs to be stored or transmitted over the network.
  This work has been presented at the ICDSC conference [Tessens et al., 2008b] and at the ACIVS conference [Lee et al., 2008].

- a novel, general framework to quantify the quality with which a subset of cameras accomplishes a network task. This is a crucial component in effective sensor selection schemes. The proposed set suitability value is derived from the Dempster-Shafer theory of evidence and can be applied to a wide range of vision problems. As a proof of concept, we have used it for sensor selection in camera networks in which multiple people are tracked. With the proposed camera selection method, we dynamically assign as little as three cameras to each tracking target and track it in difficult circumstances of occlusions and limited fields of view with the same accuracy as when using seven, eight or ten cameras. The proposed method clearly outperforms other camera selection schemes for tracking in terms of average position error and number of target losses.
  This work is expected to lead to one journal publication which is currently under review [Tessens et al., 2010].

In total, the research during this PhD resulted in two publication in international peer-reviewed journals [Morbee et al., 2010a; Tessens et al., 2008c]. One article is under review [Tessens et al., 2010] and one in preparation [Morbee et al., 2010b]. A patent application has been submitted [Morbee and Tessens, 2010]. Furthermore thirteen papers have been published in the proceedings of international conferences [Lee et al., 2008; Morbee et al., 2007a,b, 2008, 2009; Soleimani et al., 2010; Tessens et al., 2006a,b,c, 2007a,b, 2008b, 2009].

The work on camera networks has been performed in close collaboration with my colleague Marleen Morbee. Some of the general concepts of the developed methods also appear in her PhD dissertation. However, there is always an essential difference in focus between the two theses. The work on occupancy calculation in this thesis elaborates on the different possibilities for fusing the ground occupancy data. This aspect is not treated in my colleague's thesis, where the emphasis lies on an efficient calculation and usage of scan-lines. This aspect is merely briefly mentioned in this thesis.

View selection for observing people in a scene and reconstructing their shape is not discussed at all in my colleague's thesis.
The method for quantifying the quality with which a subset of cameras accomplishes a network task has been used in this thesis as a tool to select cameras for a single task at a time. The emphasis lies on a thorough study of the different aspects of task quality quantification and its effect on the final quality of the accomplished task. In my colleague's dissertation, the potential of this framework for task assignment is explored. More precisely, a technique to distribute *several* tasks over the network cameras in an optimal way with respect to the achievable frame rate is proposed in her thesis. Solutions to the related optimization problem are also investigated.
Some other colleagues in the Image and Interpretation group at Ghent University, and in Vision Systems at Hogeschool Gent have also worked on multi-camera problems, though no one really focused on smart camera networks. Note for example the work of Teelen [2010] on geometric uncertainty models for correspondence problems between cameras.

1.3 Outline

The outline of this thesis is as follows.
In Chapter 2 we study image fusion for depth of field extension of conventional light microscopes. We discuss the most common methods for such image fusion and introduce our own method based on the curvelet transform (a wavelet-like multi-resolution geometric transform). We also present a performance comparison with other methods.
Chapter 3 focuses on the influence of noise on image fusion for depth of field extension. We demonstrate how noise severely reduces fusion quality and we present denoising in the curvelet transform domain as an effective and efficient solution to this problem. We present an extensive statistical study of curvelet coefficients and use it to develop a novel curvelet-based image denoiser. Finally, we use this denoiser to improve image fusion in the presence of noise.
Chapter 4 covers the fusion of visual information in a camera network with the purpose of reaching a decision about the location of people in a scene. We introduce the Dempster-Shafer theory of evidence as an effective way of fusing evidence of the occupancy of ground positions.
In Chapter 5 we move on to data selection in camera networks. We develop a method to automatically determine in a network of smart cameras with correlated views which views are best suited for visual inspection purposes and shape reconstruction.
Chapter 6 presents a general framework to quantify the quality with which a subset of cameras in a smart camera network accomplishes a network task. We discuss why in camera networks generalized information theory is better suited for this than its classical counterpart. We use the introduced camera set suitability value for camera selection in networks. As a proof of concept,

we apply the developed camera selection method to a multiperson tracking scenario.
Chapter 7 presents the general conclusions of this dissertation.

2

Image Fusion for Depth of Field Extension

All optical imaging systems have a limited depth of field. Parts of 3D objects or scenes that fall outside the focusing range of the imaging system appear blurred in the image. This problem is particularly prevalent in conventional light microscopy. There, the object under investigation is often thicker than the depth of field of the microscope. By moving the object along the optical axis of the microscope, all object parts can be consecutively moved into the in-focus region of the microscope. In this way, a stack of images called *slices* is produced, each containing blurred and in-focus parts of the objects. To reduce the number of images one has to inspect to get a complete picture of an object or a scene, it is desirable to transform this stack into one single image that contains all the in-focus parts of the image stack . This can be achieved through fusion of the slices.

2.1 Related Work

An overview of existing image fusion algorithms can be found in [Valdecasas et al., 2001]. In this work as well as in [Li et al., 1995], it is shown that wavelet-based approaches generally perform better than other methods for extended depth of field processing of images. Pyramid representations of images in general are powerful tools for image fusion because they are better suited to separate high and low frequency image content than methods that operate in the image domain. Compared to other pyramids such as the Laplacian pyramid, the wavelet representation offers some advantages such as directional information about features (horizontal, vertical, diagonal) and information about their scale.

Forster et al. developed a very promising technique based on the complex wavelet transform rather than on the real wavelet transform [Forster et al., 2004]. Using complex wavelets allows to distinguish between the detail information of the images (represented by the phase of the complex wavelet coef-

ficients) and the weighting or strength of this detail information (encoded in the magnitude of the wavelet coefficients). In this work, the importance of the choice of the image transform was illustrated.
In recent years, many novel geometric image transforms have been developed, such as the ridgelet transform [Candès and Donoho, 1999], the wedgelet transform [Donoho, 1999], the contourlet transform [Do and Vetterli, 2005] and the curvelet transform [Candès and Donoho, 1999], just to name a few. These new transforms truly capture the geometric information present in images, and in this sense overcome the limitations of classical wavelets.
These geometric transforms all provide the potential to improve wavelet-based image fusion techniques because they can represent geometric features that are naturally present in images more efficiently. The geometric transforms that have been proposed in literature all possess different properties, the differences between particular transforms being larger on some points than on others. The wedgelet transform for example is an image representation that adapts to the image content, whereas the ridgelet, contourlet and curvelet transforms are fixed transforms. Ridgelets are particularly well suited for line singularities in images, contourlets and curvelets can handle general curvilinear discontinuities. Curvelets are directional basis functions that are highly localized, both in space and frequency. They have been originally designed in the continuous domain but also have a discrete implementation. The contourlet transform is a curvelet-like transform that has been designed directly in the discrete domain. It is implemented as a filter-bank decomposition of the image in scale and orientation. However, depending on the choice of filters, contourlets may be not well localized in the frequency domain. Moreover, the curvelet transform is better founded mathematically, which made it possible to prove that curvelets provide an optimally sparse representation of piecewise smooth images with singularities along smooth edges, with the best M-term non-linear approximation [Candès and Donoho, 2004; Candès et al., 2006]. The approximation error decays as $O\left(\left(\log M\right)^3 M^{-2}\right)$ and hence the observed decay of the absolute values of the curvelet coefficients is known to be very fast. The combination of these properties of the curvelet transform (i.e., a fixed transform, suited to represent general curvilinear image structures and with a strong mathematical basis), motivated us to develop a curvelet-based image fusion technique.
In Section 2.3, we will present some practical background information on the curvelet transform. In Section 2.4, we introduce our curvelet-based image fusion method. Results are summarized in Section 2.5 and we end with some concluding remarks in Section 2.6. The contributions of this chapter are listed in the following section.

2.2 Contributions

We propose an image fusion technique that exploits the excellent ability of the curvelet transform to separate high and low frequency image content. Because

of the high directional sensitivity of the curvelet transform, *all* high frequency information present in an image, regardless of its orientation, is contained in the highest frequency sub-bands. These sub-bands are processed with a maximum absolute value selection rule commonly used in wavelet-based image fusion methods. For the remaining low-frequency sub-band, we propose a novel selection method that is based on inter-sub-band consistency.
We also describe how some general assumptions about spatial smoothness and consistency can further improve the selection of in-focus image areas using the curvelet transform.

2.3 The Curvelet Transform

A first formulation of the curvelet transform was based on the ridgelet transform [Candès et al., 2000]. This first generation curvelet transform has been redesigned into a new mathematical construction that is simpler and more transparent. It is this second generation curvelet transform that we use throughout this dissertation.
In the following, we briefly summarize the main concepts of this transform necessary for understanding the techniques developed in this dissertation. The interested reader is referred to [Candès and Donoho, 2004] and [Candès et al., 2006] for a comprehensive description of curvelets.
Conceptually, the curvelet transform is a multi-scale pyramid with many directions and positions at each scale. To introduce curvelets mathematically, let x denote the spatial coordinate in two dimensions, ω the frequency domain coordinate in two dimensions, and r and θ polar coordinates in the frequency domain. Second generation curvelets are defined at a scale j as rotations and translations of a 'mother' curvelet φ_j. Let θ_l be a sequence of equispaced rotation angles: $\theta_l = 2\pi 2^{-\lfloor j/2 \rfloor} l$, l=0,1,... such that $0 \leq \theta_l \leq 2\pi$, and k a sequence of translation parameters $k = (k_1, k_2) \in \mathbb{Z}$.
A curvelet at orientation l and position $x_k^{(j,l)} = R_{\theta_l}^{-1}(k_1 2^{-j}, k_2 2^{-j/2})$, with $R_\theta(.)$ the rotation over θ radians, is defined as:

$$\varphi_{j,l,k}(x) = \varphi_j \left(R_{\theta_l}(x - x_k^{(j,l)}) \right), \tag{2.1}$$

with φ_j the mother curvelet at scale j. This mother curvelet is defined by means of its Fourier transform $\hat{\varphi}_j(\omega) = U_j(\omega)$, which is known as the frequency window. In the Fourier domain U_j is given by

$$U_j(r, \theta) = 2^{-3j/4} W(2^{-j} r) V(\frac{2^{\lfloor j/2 \rfloor} \theta}{2\pi}), \tag{2.2}$$

with $W(r)$ and $V(\theta)$ smooth, non-negative, real-valued windows such that the support of U_j is a polar 'wedge'. The induced tiling of the frequency plane is depicted in Fig. 2.1.

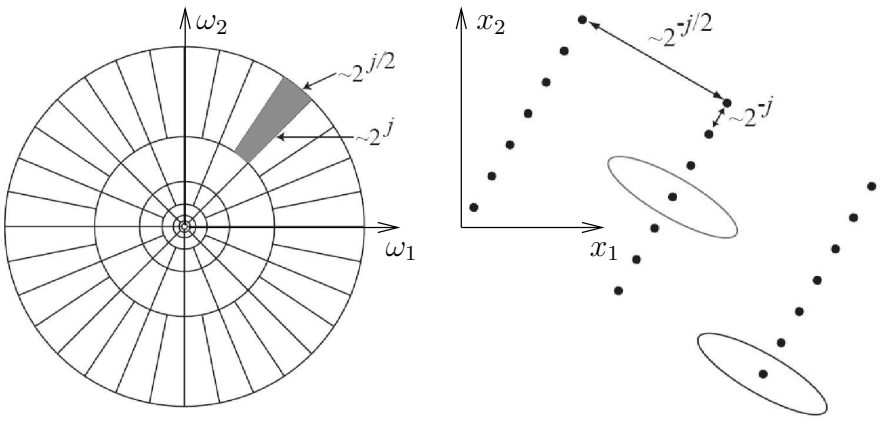


Figure 2.1: Left: induced tiling of the frequency plane. Right: Cartesian grid in the spatial domain for the scale j and orientation l of the shaded wedge. The dots indicate the positions $x_k^{(j,l)}$, $k = (k_1, k_2) \in \mathbb{Z}$, of the curvelets of this scale and orientation. The ellipses roughly delineate the support of some curvelets.

A curvelet coefficient is the inner product between an image f and a curvelet $\varphi_{j,l,k}$:

$$c(j,l,k) := \langle f, \varphi_{j,l,k} \rangle = \int_{\mathbb{R}^2} f(x) \overline{\varphi_{j,l,k}(x)}\, dx, \tag{2.3}$$

or equivalently:

$$c(j,l,k) := \frac{1}{(2\pi)^2} \int \hat{f}(\omega) \overline{\hat{\varphi}_{j,l,k}(\omega)}\, d\omega = \frac{1}{(2\pi)^2} \int \hat{f}(\omega) U_j(R_{\theta_l}\omega) e^{i\langle x_k^{(j,l)}, \omega \rangle}\, d\omega.$$

At the coarsest scale, isotropic wavelets are used as basis functions.
This continuous transform [Candès and Donoho, 2004] has several digital implementations. The two most recent ones were introduced in [Candès et al., 2006]. In these implementations the tiling of the frequency space as shown in Fig. 2.1 is replaced by a Cartesian-friendly digital tiling depicted in Fig. 2.2. Based on this tiling two digital formulations of curvelet coefficients are proposed.
The first proposed Cartesian curvelets are of the form

$$\tilde{\varphi}_{j,l,k}(x) = 2^{3j/4} \tilde{\varphi}_j \Big(S_{\theta_l}^T (x - S_{\theta_l}^{-T} b) \Big), \tag{2.4}$$

where b takes on the discrete values $(k_1 2^{-j}, k_2 2^{-j/2})$ and S_{θ_l} is the shearing matrix

$$S_{\theta_l} = \begin{bmatrix} 1 & 0 \\ -\tan(\theta_l) & 1 \end{bmatrix}. \tag{2.5}$$

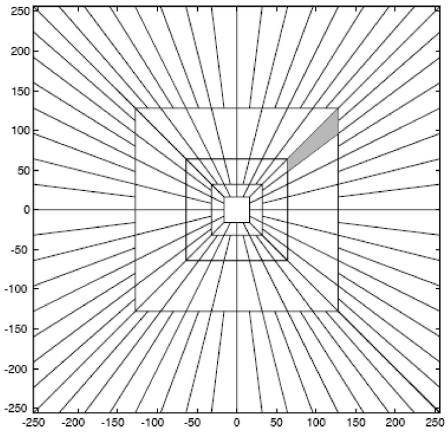


Figure 2.2: Digital coronization of the frequency plane.

The curvelet coefficients are now given by

$$c(j,l,k) = \int \hat{f}(\omega)\tilde{U}_j(S_{\theta_l}^{-1}\omega)e^{i\langle S_{\theta_l}^{-T}b,\omega\rangle}\,d\omega. \tag{2.6}$$

Evaluating this expression requires evaluating the inverse discrete Fourier transform on the sheared grid $S_{\theta_l}^{-T}(k_1 2^{-j}, k_2 2^{-j/2})$, which is not possible with the classical FFT algorithm. A solution is to pass the shearing operation to $\hat{f}$:

$$c(j,l,k) = \int \hat{f}(S_{\theta_l}\omega)\tilde{U}_j(\omega)e^{i\langle b,\omega\rangle}\,d\omega \tag{2.7}$$

and to evaluate this expression by applying the unequally-spaced fast Fourier transform, USFFT. This requires interpolation of $\hat{f}$ for each scale and orientation. The interpolation is the computationally most expensive step. By organizing this step such that related interpolation problems (i.e., for different scales and orientations) are done simultaneously, the complexity of the algorithm is $O(n^2 \log n)$, where n^2 is the number of pixels.
An alternative is to translate curvelets at each scale and orientation on a regular rectangular grid instead of a tilted grid and define Cartesian curvelets as

$$c(j,l,k) = \int \hat{f}(\omega)\tilde{U}_j(S_{\theta_l}^{-1}\omega)e^{i\langle b,\omega\rangle}\,d\omega \tag{2.8}$$

where b takes on values on the rectangular grid $(k_1 2^{-j}, k_2 2^{-j/2})$. Because the frequency window $\tilde{U}_j$ is now sheared, a periodization of the frequency samples (i.e., 'wrapping' them around the origin by re-indexing) is necessary to avoid a dramatic oversampling of the coefficients.
As the latter implementation exhibits a somewhat faster running time, especially for the inverse transform (see [Candès et al., 2006]), we use it throughout

this work. The use of the USFFT-based digital curvelet transform would lead us to similar results and conclusions.
Because of boundary and periodicity issues, the design of digital curvelets at the finest scale (highest frequencies) is not straightforward. For this reason in [Candès et al., 2006] one can choose between a wavelet and a curvelet decomposition at this scale. Using curvelets at all scales leads to a transform that provides approximate rotation invariance (sharp directional selectivity), which is beneficial for many applications. This is why in this work we always choose a curvelet decomposition at the finest scale, in spite of disadvantages such as some possible aliasing and an increased redundancy of the transform. Here, the redundancy of the transform is the proportion of the number of curvelet coefficients needed to represent an image to the number of pixels of the image. E.g., for a 512×512 image, 4 scales in the curvelet decomposition and 16 orientations at the coarsest curvelet level, redundancy increases from 2.74 to 7.16 with curvelets at the finest scale.
Fig. 2.3b shows the curvelet decomposition of the test image in Fig. 2.3a into 4 frequency scales with 8 orientations at the coarsest curvelet scale. The low-pass image is located at the center of the representation. The curvelet coefficients are arranged around it. For representation purposes, we display the *magnitudes* of the coefficients. Those with value zero are marked in white, whereas coefficients with large magnitudes are dark. From the prevalent white color of Fig. 2.3b, it is clear that the curvelet decomposition of this image is extremely sparse.
The curvelet coefficients are grouped according to orientation and scale. The concentric coronae represent the different scales, starting with the lowest scale (low frequencies) in the center. Sub-bands of the same scale are ordered within these coronae so that the orientation suggested by their position matches the spatial frequencies they represent. E.g., the diagonal lines in Fig. 2.3a produce high curvelet coefficients in the sub-bands along the direction perpendicular to them. One can clearly discern four quadrants in each corona, which we will number in a clockwise direction, starting with the upper one. Quadrant 1 contains the magnitudes of the real parts and quadrant 3 of the imaginary parts of the (complex) curvelet coefficients produced by mostly horizontally oriented curvelet functions. Mutatis mutandis, the same holds for quadrant 2 and 4.

2.4 Curvelet-based Image Fusion

To select the in-focus image parts throughout an image stack of a 3D object, we must be able to distinguish between in-focus and out-of-focus regions. Conceptually, edges and details appear to be 'smeared out' in blurred image regions. Mathematically, this means that a blurry image region contains fewer high frequency components than an in-focus one.
The sub-bands in the curvelet decomposition of an image can be considered band-pass filtered versions of this image. This means high and low frequency image content are separated by this transform. As was mentioned before, the

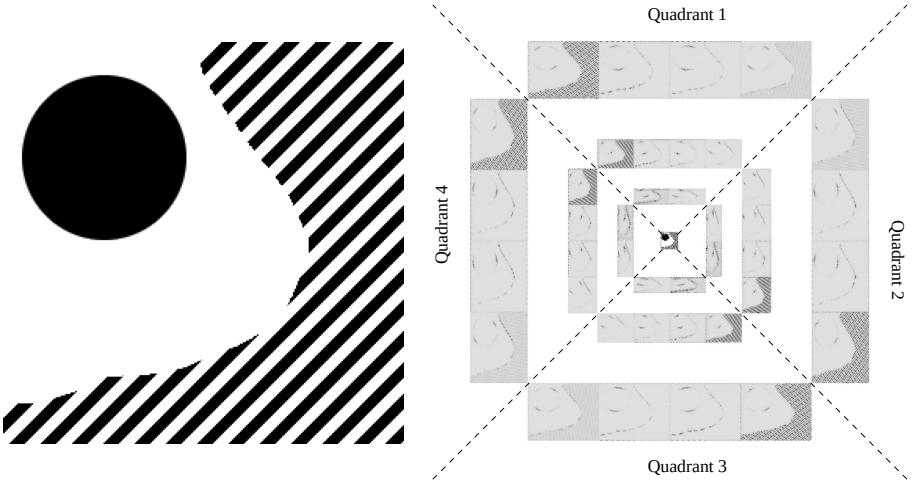


Figure 2.3: (a) A 256 × 256 test image. (b) Its curvelet decomposition into 4 scales with 8 orientations at the coarsest scale. The low-pass image is located at the center of the representation. Curvelet coefficients with value zero are marked in white, whereas coefficients with a large magnitude are dark. The dotted lines mark the border between the four quadrants.

curvelet decomposition of a piece-wise smooth image is extremely sparse (a consequence of the high directional sensitivity of the curvelet transform) and virtually all information about high frequency image features is contained in the high frequency sub-bands of the decomposition. This means that blurring will primarily have an effect on the high frequency sub-bands, and the distinction between in-focus and out-of-focus image regions must thus be made here. We will discuss how to do this in Section 2.4.2.
Because of the extreme sparseness of the curvelet decomposition, a small number of scales suffices to identify the in-focus image regions within the stack. In this work, we have used a decomposition into 3 scales (including the low-pass image). Using more scales increases redundancy in the transform and does not lead to better fusion results.
We will now first describe the assumptions underlying the proposed image fusion algorithm, after which we will discuss its different parts.

2.4.1 Assumptions

We assume the slices in the image stacks satisfy the following constraints.

- **Perfect registration:** the slices need to share a common coordinate system. This guarantees that a pixel with certain coordinates corresponds to the same part of an object for all images in the stack. If the observed specimen is not moving, this constraint is not a problem in microscopy

because the objective lens is only moved along the optical axis to capture the slices. Similarly, in conventional photography or video, perfectly registered images are obtained when the scene is still and the optical axis does not change between capture. If it is not possible to obtain perfectly registered images by controlling the capturing set-up or the scene, rigid or non-rigid registration of the slices is necessary. A plethora of image registration methods have been described in literature (see [Zitova and Flusser, 2003] for an overview). We refer the reader to these techniques and will not treat image registration in this thesis.

- **Absence of noise:** As we have briefly mentioned previously, our distinction between in- and out-of-focus image regions is based on their frequency content. If the image is degraded by noise in the high frequency range, the algorithm will mistakenly classify noise as information of interest at the expense of real image features and this will deteriorate fusion results. Chapter 3 of this thesis is dedicated entirely to fusion of noise-degraded images.

2.4.2 Processing of the High Frequency Sub-Bands

Big curvelet coefficients in the high-frequency sub-bands correspond to image features with high spatial frequency components. We assume these features lie in an in-focus image region. By selecting the coefficients throughout the stack with the highest absolute value at each position, orientation and scale, we assure that the most salient image features throughout the stack are preserved. This maximum absolute value selection rule is the selection rule used in many wavelet-based image fusion schemes (see [Forster et al., 2004; Li et al., 1995; Zhang and Blum, 1999]). It actually consists of two steps:

- quantifying the degree to which a coefficient corresponds to a salient image feature. This process is commonly referred to as *activity level measurement*. In the maximum absolute value selection rule, the absolute value of a coefficient is its activity level;

- combining the coefficients of the slices. In the maximum absolute value selection rule, the coefficient with the maximal activity level is selected as the output coefficient.

For both steps, alternatives exist.
Instead of considering coefficients separately to determine their activity level, many authors have proposed window-based activity measures which consider coefficients in a small (typically 3×3 or 5×5) window centered at the current coefficient. Popular methods include weighted averaging of the coefficients in the window [Valdecasas et al., 2001; Zhang and Blum, 1999], or applying a rank filter which picks the ith largest coefficient in the window [Li et al., 1995; Valdecasas et al., 2001; Zhang and Blum, 1999]. The idea of window-based approaches is to detect the presence of a dominant feature in the local neighborhood. Region-based activity measures rely on segmentation and determine

the activity of an entire region, for example as the average absolute value of all the coefficients associated with this region [Zhang and Blum, 1999]. The rationale underlying these alternative methods of activity measurement is essentially the same as in the basic approach of taking the magnitude of a coefficient as its activity level, namely that salient image features give rise to coefficients with a high magnitude.
These variations on the basic activity level measurement can potentially improve the fusion results. However, they would also introduce extra parameters that influence the results. For example for the window-based activity measure, fusion results will vary with the window size and the rank filter choice. For the region-based approach, high quality image segmentation will be very important to achieve high quality fusion. In this work we focus on the influence of the image transform on the fusion results. Therefore we do not investigate these variations on the basic activity level measurement. Instead, we restrict ourselves to measuring the activity level of a coefficient by its absolute value and we note that more elaborate activity measures hold the potential for improving the results.
A different method of coefficient combination than choosing the coefficient with the highest activity level is calculating the output coefficient as a weighted average of the coefficients of the different slices. The weight assigned to a coefficient typically depends on its activity level [Zhang and Blum, 1999]. In [Zhang and Blum, 1999] this method was found to exhibit the same performance as choosing the coefficient with the highest activity level. Selecting the coefficients corresponding to in-focus image pixels can also be treated as a classification problem. In [Li et al., 2004], support vector machines are used to this effect. A disadvantage of this approach is the training required for the support vector machines. In this work we select the coefficients with maximal activity level as the output coefficients.

2.4.3 Consistency and Smoothness Checks

When selecting curvelet coefficients from different slices, one can naturally assume some spatial smoothness within and consistency between the decomposition bands. To impose these assumptions, consistency and smoothness checks must be performed.
A first possible check is based on the assumption of inter-sub-band consistency: all curvelet coefficients corresponding to a feature at a specific spatial location in the image should in theory be taken from the same slice, regardless of the orientation and the scale of the feature. In practice, this is not always the case after the previous image fusion step described in Section 2.4.2. A feature with a particular orientation and scale gives rise to curvelet coefficients with a high absolute value in a small number of sub-bands. In the other bands, the corresponding coefficients from the sharply imaged slice have a small magnitude. Some minor fluctuations and disturbances in the other slices may give rise to curvelet coefficients that have a slightly bigger magnitude than the correct ones. The maximum absolute value selection rule will select the slightly

bigger coefficients, which leads to inconspicuous false features being incorporated in the fused image. To prevent this, we can impose the constraint that all corresponding curvelet coefficients should be selected from the slice which supplied among these coefficients the one with the largest absolute value after the previous step.
A second check assumes some spatial smoothness of the sharp image regions. Indeed, the slice in which a realistic 3D-sample appears in-focus might change abruptly over the sample, but this will happen in a piece-wise continuous manner. Therefore, if a majority of neighboring coefficients in a 3×3-window are from the same slice, the current coefficient is taken from that slice too.

2.4.4 Processing the Low-Pass Image

By definition, the low-pass image contains the low frequency features in the image. The saliency of these features cannot be determined in the same manner as for the high frequency features that show up in the detail sub-bands of the curvelet decomposition.
Because low frequency features are not affected as strongly by blurring as high frequency image content, in literature, the low-pass image of the fused image is obtained by averaging the low-pass images of all slices. However, in our case where the number of scales in the decomposition is very low, the blurring does influence the low-pass image as well. In this case, averaging is too crude a technique to obtain the low-pass image of the fused image and a correct selection of the low-pass coefficients becomes very important. Therefore we propose a novel strategy to perform this task: we select the low-pass coefficients based on the assumption of inter-sub-band consistency. This means that the curvelet coefficients in the low-pass image should be taken from the same slice as the corresponding curvelet coefficients in the high-frequency sub-bands. If no inter-sub-band consistency check has been performed for the high-frequency sub-bands, not all corresponding curvelet coefficient may have been selected from the same slice. In this case we select the low frequency coefficient from the slice which supplied among all corresponding high frequency coefficients the one with the largest absolute value. In general this assures that at each spatial position in the low-pass image, the curvelet coefficient from the correct, in-focus slice is selected.

2.4.5 Image Fusion Algorithm

Our curvelet-based image fusion technique can be summarized as follows:

1. All images of the image stack are decomposed into their complex curvelet coefficients $C_{i,j,z}(x, y)$, where z denotes the slice index, i the scale and j the orientation within the scale. x and y are spatial coordinates.

2. For each position in every sub-band, the curvelet coefficient with the

highest absolute value over all the slices in the stack is selected:

$$F_{i,j}(x,y) = C_{i,j,\mathrm{argmax}_z(|C_{i,j,z}(x,y)|)}(x,y). \tag{2.9}$$

3. Consistency and smoothness constraints are imposed (see Section 2.4.3).
4. The low-pass image is processed (see Section 2.4.4).
5. The inverse curvelet transform of the curvelet coefficients $F_{i,j}(x,y)$ is calculated and the fused image f is obtained.

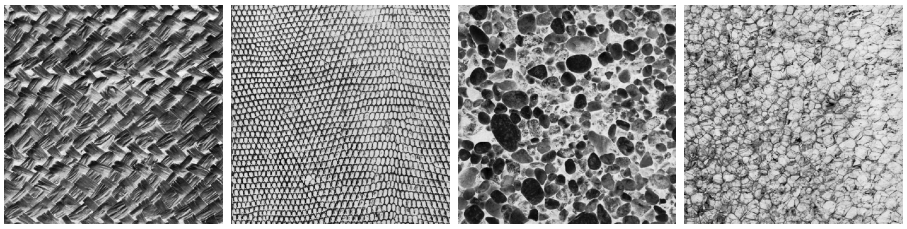
2.4.6 Pre- and Post-Processing

As a pre-processing step, multi-channel slices are first converted into gray-scale images $s_z(x,y)$ by a weighted linear combination of the color channels $s_z^{(k)}(x,y)$: $s_z(x,y) = \sum_k w_k s_z^{(k)}(x,y)$. As was proposed by Forster et al., the weights w_k are obtained from a principal component analysis and are chosen such that the color channels are projected on the direction in the color channel space with maximal variance [Maloney, 1999]. In this way, images with a predominant color lead to gray-scale images with more contrast and saliency than if fixed weights are used [Forster et al., 2004].
Note that these grayscale images are only intermediate products of the algorithm, necessary to perform the image fusion. If the input of the fusion algorithm is a stack of color images, the output is a color image that is the fusion product of the multi-channel slices, as will be explained now.
After the inverse curvelet transform, the fused image may contain false gray-scale values. These are gray-scale values that were not present in any of the slices and thus may introduce artifacts. As Forster et al. suggested, we remove them through 'reassignment'. Multi-channel reassignment for each channel k can be expressed as:

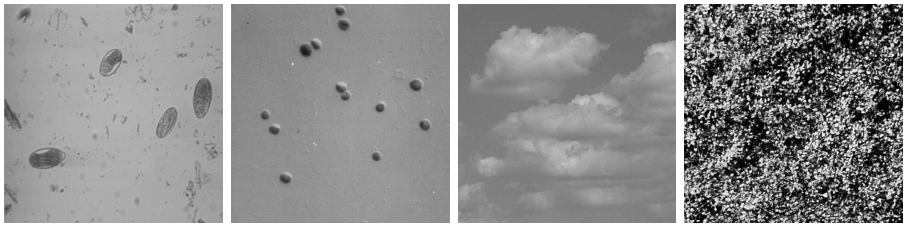
$$q^k(x,y) = s^k_{\mathrm{argmin}_z|f(x,y)-s_z(x,y)|}(x,y). \tag{2.10}$$

2.5 Results

To evaluate our curvelet-based image fusion method, we test it both on artificially generated test data and on real microscopy images. We compare it with the complex wavelet-based method of Forster et al. with and without sub-band and spatial smoothness checks [Forster et al., 2004], and with a pixel domain variance-based one. In this last method, the distinction between in-focus and out-of-focus image regions is made based on the local variance. This local variance is calculated in a 3×3 window around every pixel in every slice. At each spatial position in the fused image, the pixel from the slice with the highest local variance throughout the slice is selected.
For all methods, the images are pre- and post-processed as described in Section 2.4.6.



(a) D18 (b) D22 (c) D23 (d) D112



(e) Eggs (f) Algae (g) Clouds (h) Leaves

Figure 2.4: The test images used for the creation of artificial test data.

2.5.1 Results on Artificial Test Data

To test our method in a quantitative way, we have generated some artificial image stacks. Four stacks were obtained by mapping Brodatz textures (see Figure 2.4a-d) onto a 3D surface and blurring them with a Gaussian blurring kernel with a size that varies with the slice number (1 to 7)[1] [Forster et al., 2004]. Four other stacks containing three slices were obtained based on the images in Figure 2.4e-h. The images *Eggs* and *Algae* are 512×512 gray-scale microscopy images, the other images are 512×512 color images of textures taken from the MIT Vision Texture Database. An example stack can be seen in Figure 2.5. In each of the slices, another part is left unblurred. The blurring is introduced through convolution with a 5×5 Gaussian blurring kernel with standard deviation 1.

First we investigate the influence of the different types of checks (sub-band and spatial). Each stack is processed with the proposed curvelet fusion method without checks, with only a sub-band check, with only a spatial check and with a spatial check after the sub-band check. The result is compared to the original image and the peak signal-to-noise ratio (PSNR) is calculated. The results are shown in Table 2.1.

From Table 2.1 we see that for most stacks the introduction of smoothness and consistency checks is beneficial for the curvelet-based image fusion method. The gain in PSNR ranges from 0.05 dB for the *D22* texture to 4.52 dB for the *Clouds* image. For some stacks, both smoothness and consistency checks deteriorate the performance of the fusion. This is for example the case for the

[1] These stacks were kindly provided by D. Van De Ville, BIG EPFL.

Figure 2.5: Example of an artificial image stack.

Table 2.1: Image fusion results in terms of PSNR for different gray-scale and color image stacks, using the curvelet-based method without checks, with only a sub-band check, with only a spatial check and with a spatial check after the sub-band check. For each image the best result is marked in bold.

	No Checks	Sub-band Check	Spatial Check	All Checks
D18	34.50	34.80	34.64	**34.82**
D22	29.56	**29.61**	29.60	**29.61**
D23	35.64	36.09	35.86	**36.17**
D112	33.23	33.62	33.37	**33.69**
Eggs	**64.35**	62.55	63.74	60.65
Algae	**66.39**	63.14	64.42	61.20
Clouds	52.70	54.77	57.15	**57.22**
Leaves	**40.79**	38.92	39.73	38.32

microscopy images Eggs and Algae. In these images, as well as in Clouds and Leaves, the transitions between the in-focus and blurred image regions are very abrupt.

Around such transitions, the assumption of spatial smoothness is not valid. When the transitions are very abrupt, such as in the last four stacks of our test set, any error causes considerable changes in the output image, leading to a quality drop that is more important than the beneficial effect of the checks in the image regions that are not near transitions. Moreover, a very abrupt transition might itself get detected as a feature. Enhancing such a false feature by spatial and sub-band checks will reduce the quality of the fused image. These effects play less for the first four stacks of our test set, where the transitions between sharp and blurred image regions are more gradual.

To verify that indeed the abruptness of the transitions and not the nature of the images underlies the quality drop caused by spatial and smoothness checks for, e.g., Eggs and Algae, we generate artificial image stacks with smooth transitions from these images. We do this following the same procedure as

Table 2.2: Image fusion results in terms of PSNR for the Eggs, Algae, Clouds and Leaves image stacks, generated to have smooth transitions between the blurred and the in-focus image regions. Fusion is done using the curvelet-based method without checks, with only a sub-band check, with only a spatial check and with a spatial check after the sub-band check. For each image the best result is marked in bold.

	No Checks	Sub-band Check	Spatial Check	All Checks
Eggs (smooth transitions)	37.73	42.52	38.93	**44.60**
Algae (smooth transitions)	40.52	46.00	41.00	**46.40**
Clouds (smooth transitions)	41.09	44.41	42.30	**45.60**
Leaves (smooth transitions)	31.29	31.42	31.44	**31.51**

used for generating artificial stacks from the Brodatz textures, i.e., we map the images Eggs, Algae, Clouds and Leaves onto a 3D surface and blur them with a Gaussian blurring kernel with a size that varies with the slice number (1 to 7). These stacks are then processed with the proposed curvelet fusion method without checks, with only a sub-band check, with only a spatial check and with a spatial check after the sub-band check. Fusion results are listed in Table 2.2. From Table 2.2 it is clear that for these stacks with gradual transitions between blurred and in-focus image regions, performing spatial and smoothness checks improves the fusion results.

Note that the image stacks used for generating the results of Table 2.2 are only used here to demonstrate that abrupt transitions between blurred and in-focus image regions can lead to a deterioration of the fusion results when performing checks. In the remainder of this thesis, these stacks will not be used anymore and the proposed methods will only be tested on the image stacks described at the beginning of this section.

The contribution of the spatial and the sub-band checks to the quality improvement of the fused image is almost equal, and their combined application leads to the best results.

Apart from the fused image, the proposed fusion algorithm also outputs a slice selection decision for each coefficient in the curvelet decomposition of the fused image. As one pixel in the image corresponds to more than one curvelet coefficient, there is more than one selection decision per image pixel. A slice selection decision per pixel can serve directly as a (discrete) estimate of the distance between the object part that is projected onto this pixel and the imaging sensor. To briefly explore the potential of our technique as a 'depth from defocus' method, we propose here to transform the coefficient selection decisions in a selection decision per pixel by majority voting. I.e., for each pixel we check from which slices the curvelet coefficients corresponding to this pixel are selected in the curvelet decomposition of the fused image. A pixel is said to be in focus in the slice from which the majority of coefficients in the curvelet decomposition of the fused image is selected.

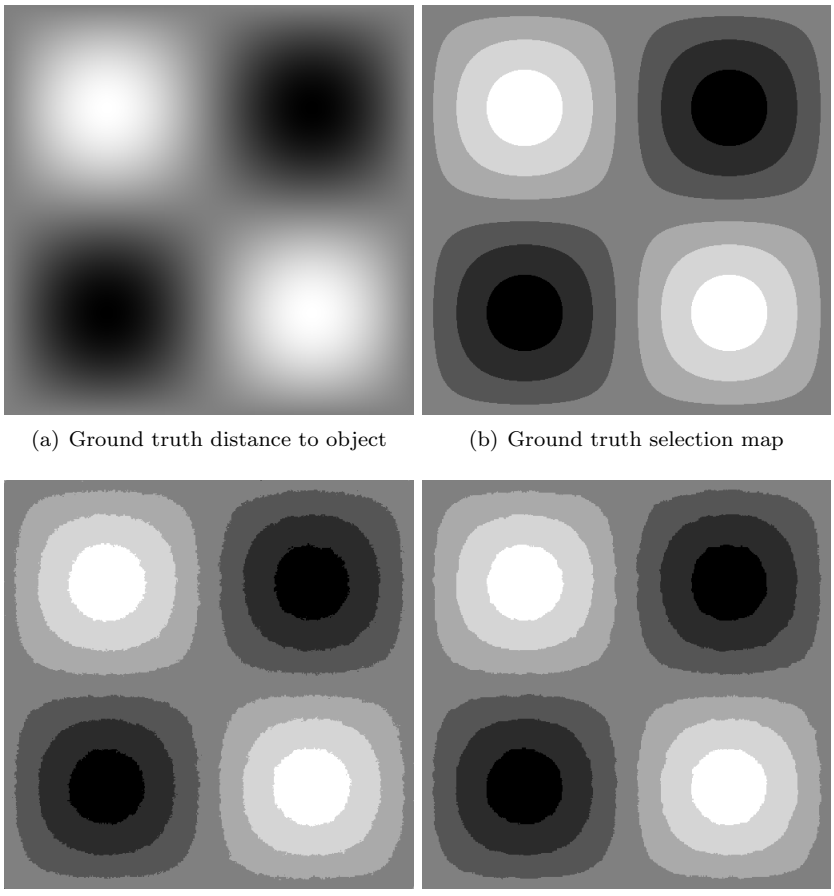
Table 2.3: Percentage of pixels in the fused image selected from the correct slice for different gray-scale and color image stacks, using the curvelet-based method without checks, with only a sub-band check, with only a spatial check and with a spatial check after the sub-band check.

	No Checks	Sub-band Check	Spatial Check	All Checks
D18	97.31	97.72	98.01	98.28
D22	97.33	97.40	98.15	98.06
D23	97.17	97.28	97.98	97.87
D112	97.52	97.63	98.24	98.14
Eggs	99.59	99.58	99.63	99.62
Algae	99.77	99.74	99.79	99.76
Clouds	99.64	99.61	99.66	99.60
Leaves	99.41	99.35	99.48	99.43

In Table 2.3 we list the percentage of image pixels that using this technique has been selected from the correct in-focus slice. For the first four stacks, where the distance z between object and sensor varies smoothly (see, e.g., Fig. 3.1a), the correct slice is the slice which is at a distance of the image sensor closest to z (see, e.g., Fig. 3.1b). As can be observed, the percentage of correctly selected pixels is quite high for all tested stacks. The performance is generally higher for the last four stacks in our test set, where transitions between slices are abrupt and less confusion is possible. In line with expectations, sub-band and spatial checks improve the depth estimation results for nearly all stacks. A visual result is shown for the *D112* texture in Fig. 2.6c-d. We see that apart from irregularities at the transitions between slices, the slice selection decisions are nearly all correct. The irregularities are smoothed out to some extent when checks are performed.

We now compare the performance of the proposed curvelet-based fusion with the variance method described above and with the complex wavelet-based method of Forster et al. with and without sub-band and spatial smoothness checks. The PSNR-values for all methods are grouped in Table 2.4. Let us first discuss fusion without imposing any smoothness or consistency constraints. From Table 2.4, we can see that the curvelet-based method outperforms the other methods. The average gain in PSNR over the variance method is 7.88 dB. The complex wavelet method lags behind the curvelet method by 3.23 dB on average. Spatial and sub-band checks have a similar influence on the complex wavelet-based method as on the proposed curvelet-based one so the relative performance of the two methods is the same with checks as without checks. However, the average difference between the two shrinks by a little more than a dB to 2.21 dB.

It is interesting to note that the curvelet-based method performs particularly well for the *Eggs* and *Algae* images, which have many very clear edges. On the



(a) Ground truth distance to object (b) Ground truth selection map

(c) Selection map without checks (d) Selection map with checks

Figure 2.6: For the *D112* Brodatz texture, (a) ground truth distance to object, (b) ground truth selection map and selection map obtained from the proposed fusion algorithm, (c) without and (d) with checks. The different gray tones correspond to different distances between object and image sensor.

Table 2.4: Image fusion results in terms of PSNR for different gray-scale and color image stacks, using the variance method, the complex wavelet-based method of Forster et al. [Forster et al., 2004] with and without checks and the curvelet-based method with and without checks. For each image the best result is marked in bold.

	Variance	Complex Db6	Complex Db6 with checks	Curvelets	Curvelets with checks
D18	29.29	31.13	34.35	34.50	**34.82**
D22	24.00	27.52	29.08	29.56	**29.61**
D23	27.81	32.98	35.29	35.64	**36.17**
D112	28.38	29.27	32.59	33.23	**33.69**
Eggs	47.76	59.80	59.73	**64.35**	60.65
Algae	53.34	62.17	58.77	**66.39**	61.20
Clouds	54.79	49.26	49.21	52.70	**57.22**
Leaves	28.75	39.20	34.97	**40.79**	38.32

contrary, the variance method produces poor results for these images. Indeed, the curvelet transform is particularly well suited for piece-wise smooth images, whereas the variance method tends to introduce artifacts around abruptly-changing image structures. For the *Clouds* image, roles are reversed and the variance method even outperforms both multi-resolution methods.

Computation time experiments were performed on an AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set. The performed computations were floating-point computations. In Matlab code, with the computation intensive parts implemented in c, averaged over 10 experiments curvelet-based fusion without checks of a stack of 15 512×512 images took 71.48 s to execute. Imposing smoothness and consistency constraints took an additional 77.50 s. In other words fusion with checks takes 71.48 s + 77.50 s = 148.99 s, or about twice the time of fusion without checks. With and without checks the speed limitation of the execution of the algorithm was the computational power of the processor and not the memory bandwidth. In an implementation with a comparable level of optimization, the complex wavelet-based method of Forster et al. without checks required on average over 10 experiments 41.38 s to execute. This is less than the curvelet-based method. The explanation lies in the higher redundancy of the curvelet transform. For 3 decomposition levels and 8 orientations at the coarsest level, this redundancy is 7.25. The redundancy of the complex wavelet-transform of 2D images is fixed at 4. The ratio between the two execution times is thus in line with the ratio between the redundancies of the two transforms. Executing the complex wavelet-based method with checks took on average 56.28 s, or only 14.90 s for the checks. This is a lot less than for the checks in the curvelet-based method. An important factor here is the smaller number of sub-bands in the complex wavelet decomposition compared to the number of sub-bands in the curvelet decomposition of an image.

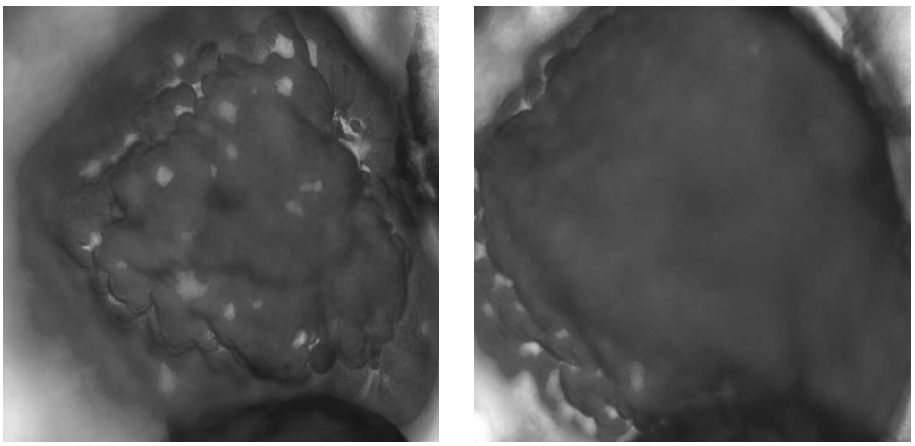


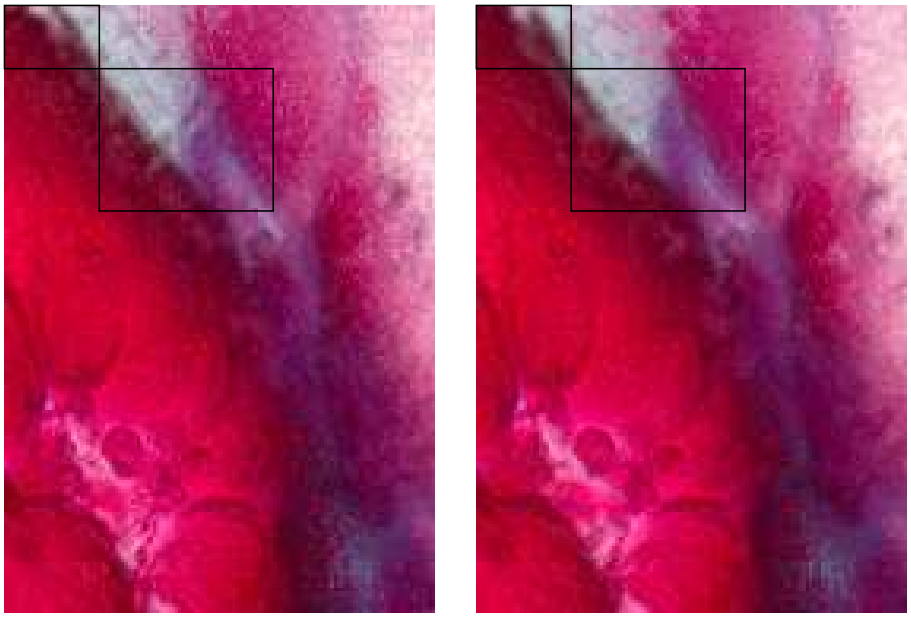
Figure 2.7: Some slices of a stack of 15 microscopic images of Peyer plaques from the intestine of a mouse.

The variance method, which operates in the image domain, is a lot faster than the other two methods. In an implementation with a comparable level of optimization as for the other two methods, averaged over 10 experiments, this algorithm took 3.07 s to execute.

2.5.2 Results on Real Test Data

We now test the methods on a stack of 15 512×512 color microscopic images of Peyer plaques from the intestine of a mouse.[2] The same images are used in [Forster et al., 2004]. Some slices are shown in Figure 2.7.
First, we visually compare the fusion result of the curvelet-based fusion method when imposing smoothness and consistency constraints versus the fusion result when not performing any checks. As no ground-truth image is available, only this visual evaluation of the results is possible. Figure 2.9d-e shows the fused images. Figure 2.8 zooms in on small cut-outs of the upper right corner of the fused images. One can see that small artifacts (see delineated regions) are removed thanks to the constraints.
In Figure 2.9, the image fusion results of the variance method, the complex wavelet-based method without and with spatial and sub-band checks and the curvelet-based method without and with spatial and sub-band checks are compared. We can see that in the image produced by the variance method, sharp edges are surrounded by artifacts. The complex wavelet-based method, both with and without checks, leaves some image regions blurred (see delineated regions). The curvelet-based method leads to a complete in-focus image, without introducing artifacts. This demonstrates that curvelets can be successfully

[2]The images are courtesy of Jelena Mitic, Laboratoire d'Optique Biomédicale at EPF Lausanne, Zeiss and MIM at ISREC Lausanne.



(a) Without checks. (b) With checks.

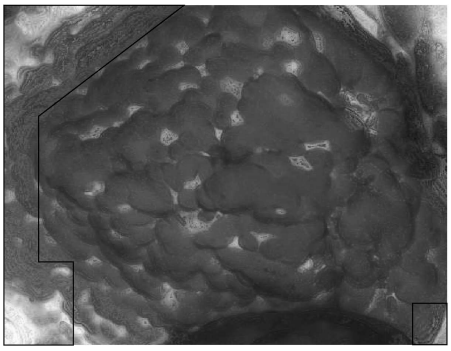
Figure 2.8: Detail of the fused image produced by the curvelet-based fusion method (a) without performing checks and (b) with checks. One can see that small artifacts (see delineated regions) are removed thanks to sub-band and spatial checks.

used for image fusion of real microscopy image stacks.

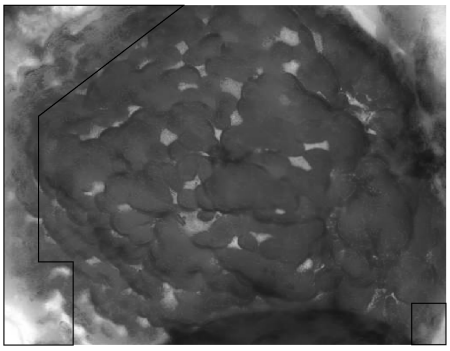
2.6 Conclusion

In this chapter we have demonstrated that the directional sensitivity of the curvelet transform and its excellent ability to separate high and low frequency image content can be turned to good account to extend the depth of field of imaging systems. The proposed method produces high quality fusion results, both on real microscopy data and on artificially generated image stacks. Our method outperforms state-of-the-art fusion algorithms. The average performance gain is 3.23 dB over the complex wavelet-based technique of [Forster et al., 2004] and 7.88 dB over the common pixel domain variance-based method. Moreover, we have shown that adding consistency and spatial smoothness checks to this curvelet-based image fusion method generally leads to better fusion results. For real test data, imposing these constraints leads to a reduced number of artifacts in the fused image. This demonstrates the suitability of our curvelet-based method for the artifact-free extension of the depth of field of imaging systems.
Additionally, we have hinted at the potential this method holds as a depth

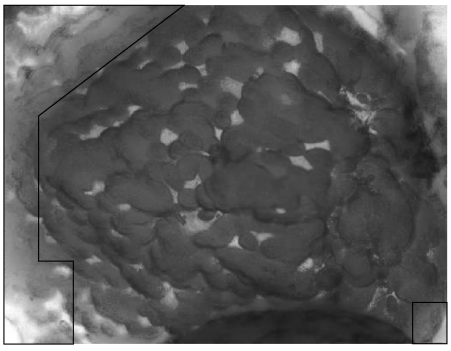
from defocus technique by identifying which slice contains a sharp image of each object part.



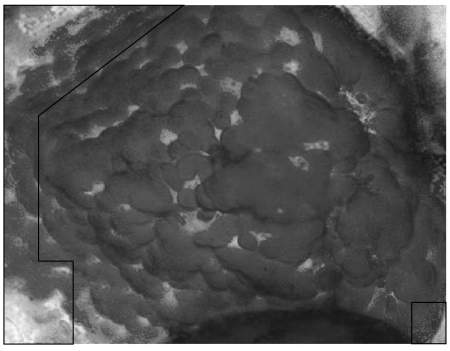
(a) Variance method.



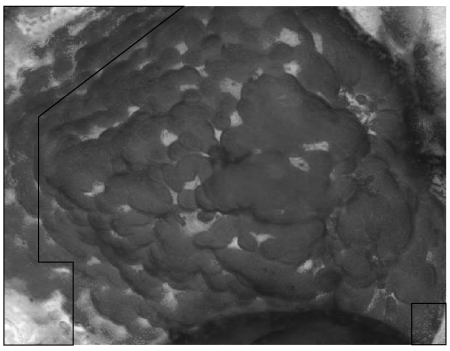
(b) Complex wavelet-based method without checks [Forster et al., 2004].



(c) Complex wavelet-based method with checks [Forster et al., 2004].



(d) New curvelet-based method without checks



(e) New curvelet-based method with checks.

Figure 2.9: Image fusion results of the tested methods. In the image produced by the variance method, sharp edges are surrounded by artifacts. The complex wavelet-based method leaves some image regions blurred (see delineated regions). The curvelet-based method leads to a complete in-focus image, without introducing artifacts.

3

Fusion of Noise-Degraded Images

All image recording systems suffer to some degree from different types of degradation of the recording quality. Part of these degradations are inherent to capturing light (photon shot noise). Other disturbances are introduced by the recording equipment, such as reset noise, thermal noise, transistor dark currents, readout noise and dark-current shot noise. Digitization of the signal causes quantization noise, and further digital processing inside the camera, such as demosaicing and image enhancement, can also give rise to noise. See [Boie and Cox, 1992; Irie et al., 2008] for a detailed discussion of camera noise. Although this is an oversimplification of reality, the joint effect of all these noise sources is very often modeled as additive white Gaussian (AWG) noise. This is the noise model we adopt in this thesis.
As mentioned in Section 2.4.1, the fusion method outlined in Chapter 2 is very sensitive to white noise because the algorithm assumes that high-frequency image content always indicates the presence of salient image features, whereas in a noise-contaminated image it may sometimes be caused by noise. Sensitivity to noise is a common problem among image fusion methods [Petrovic and Xydeas, 2000]. We will now discuss this problem more in depth.

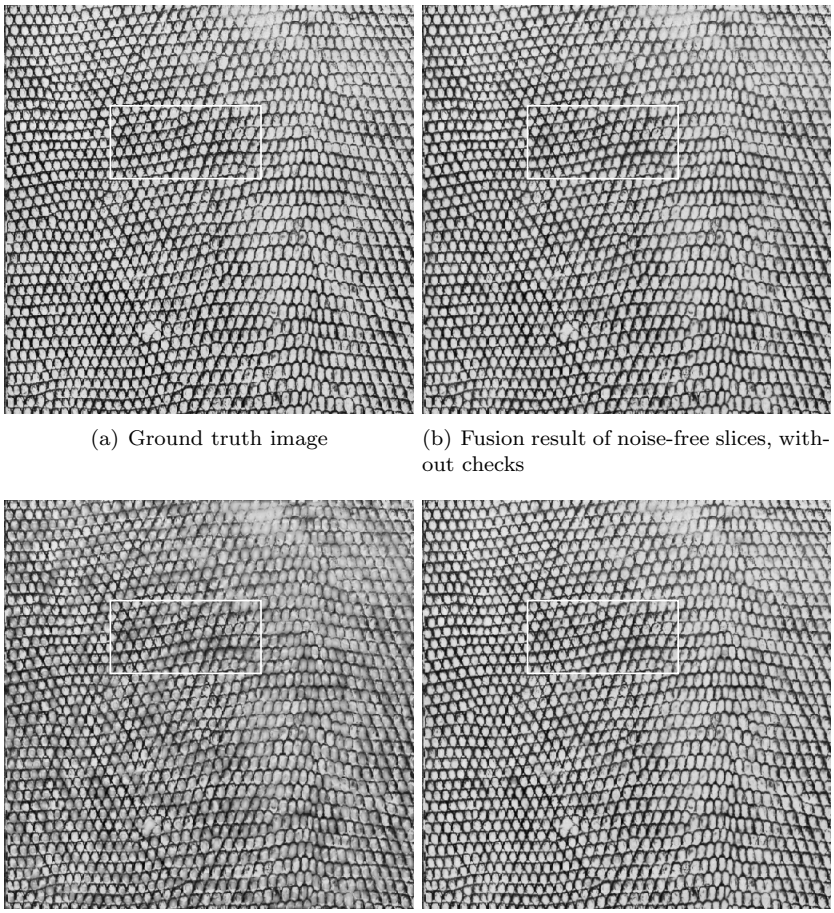
3.1 Influence of Noise on Image Fusion

Noise in the input image stack deteriorates the quality of the fused image in two ways. Firstly, as explained, the noise severely disrupts the fusion process itself. Secondly, the output image is corrupted by the noise present in the input slices. For example, the average PSNR of the slices in the *Eggs* image stack, contaminated with additive white Gaussian noise with standard deviation $\sigma = 10$, is 28.12 dB. Fusing these noisy slices with the method presented in the previous chapter results in an output image with a PSNR of 26.70 dB. This result is lower than the average PSNR of the noisy input slices because the noise has disturbed the fusion process. If each noisy curvelet coefficient is selected

from the same slice as the corresponding coefficient in the noise-free case, the fusion of the noisy stack leads to an image with a PSNR of 28.32 dB, i.e., approximately the average PSNR of the input stack.
In this thesis we focus on the first effect of noise in the input stack, namely the degradation of the fusion process. Improving the fusion process itself in the presence of noise is important because it results in more correct slice selection decisions. As explained in Section 2.5.1, these decisions are a useful basis for depth estimation. Moreover, as illustrated by the above example, more correct slice selection decisions allow to maintain the quality level of the input stack for the fused image.
To assess the influence of the noise on the fusion process independently of the fused image degradation caused by noisy input images, we adopt the following procedure. We determine for each curvelet coefficient from which slice it is to be selected based on the noisy input stack. We use this 'selection map' to actually select the coefficients from the noise-free image slices. Mathematically, this can be formulated as follows. Let $C_{i,j,z}(x,y)$ denote the noise-free curvelet coefficient at scale i, orientation j and spatial coordinates x and y, of the slice with index z. Let $C^n_{i,j,z}(x,y)$ be the corresponding noise-contaminated curvelet coefficient. Coefficients of the fused image are then selected as:

$$F_{i,j}(x,y) = C_{i,j,\mathrm{argmax}_z(|C^n_{i,j,z}(x,y)|)}(x,y). \tag{3.1}$$

In a real situation, the noise-free coefficients would of course not be available and the fused image obtained from these coefficients $F_{i,j}(x,y)$ is an artificial one that is used here only for evaluation purposes. From Equation 3.1, the disrupting effect of noise on the fusion process is clear. A curvelet coefficient amplitude may be large because of a prevalent, sharply imaged feature, but also because of noise. Equation 3.1 cannot distinguish between these two cases. Thus, at a particular position in a sub-band, a specific slice z may be selected because of noise, and not because the slice is in focus.
To highlight the effect of noise on the fusion process, we artificially contaminate the image stack of the *D22* Brodatz texture with additive white Gaussian noise with a standard deviation of $\sigma = 10$ (for other standard deviations of the noise, we would reach similar conclusions). As the noise originates from the light recording equipment, after blur due to limited depth of field is introduced by the optical system of the imaging device, the noise standard deviation is assumed constant across slices.
Fig. 3.1c shows for this noisy image stack the fused image obtained using the selection rule of Eq. 3.1. As a reference, the ground truth image and the fused image obtained from the noise-free stack are shown in Fig. 3.1a-b. Clearly, the noise has disturbed the fusion process and important artificially introduced structures show up in the fused image, caused by coefficients being selected from blurred regions. Performing the smoothness and consistency checks described in Section 2.4.3 has a regularizing effect, as is apparent from Fig. 3.1d. Many of the wrong coefficient selections have been corrected and virtually no blurry patches remain visible in the fused image.



(a) Ground truth image

(b) Fusion result of noise-free slices, without checks

(c) Fusion result of noise-free slices based on noise-degraded slices, without checks

(d) Fusion result of noise-free slices based on noise-degraded slices, with checks

Figure 3.1: For the *D22* Brodatz texture, (a) ground truth image and the fused image obtained by fusing noise-free images (b) based on noise-free slices, without performing smoothness and consistency checks, and (c) based on slices contaminated with additive white Gaussian noise with a standard deviation of $\sigma = 10$, without checks, and (d) with checks. Although quality differences can be noticed all over the images, the white rectangle delineates a region where the differences are particularly clear.

	Noise-free Input Stack		Noisy Input Stack	
	No checks	All checks	No checks	All checks
D18	34.50	34.82	21.99	29.11
D22	29.56	29.61	19.64	28.58
D23	35.64	36.17	22.33	29.08
D112	33.23	33.69	22.04	28.97
Eggs	64.35	60.65	40.68	40.55
Algae	66.39	61.20	38.53	38.53
Clouds	52.70	57.22	39.32	39.28
Leaves	40.79	38.32	36.84	37.03

Table 3.1: Fusion results in terms of PSNR (dB) of noise-free input slices based on a selection map obtained from noise-free slices, without and with consistency and smoothness checks, and from slices contaminated with additive white Gaussian noise of $\sigma = 10$, without and with consistency and smoothness checks.

These observations are confirmed numerically in Table 3.1. Here, the results in terms of PSNR are shown for fusion using the selection rule of Eq. 3.1 with noisy image stacks contaminated with AWG noise with standard deviation $\sigma = 10$. To facilitate comparison with the results of noise-free fusion, the results from the previous chapter are repeated in this table. Clearly the noise severely disrupts the fusion process and the quality of the fused images based on a noisy stack is far below that of the noise-free case. Imposing spatial smoothness and sub-band consistency constraints raises the quality of the fusion result for most image stacks, especially the first four ones in which the transition between blurry and in-focus image regions is not abrupt (see Section 2.5.1 for a discussion of the influence of checks on the fusion result). The improvement is quite large, e.g., 8.94 dB for the *D22* texture.

An alternative possibility to improve the fusion result in the presence of noise is to remove noise from the slices prior to fusion. In Section 3.8 we will illustrate that this is an effective way of improving fusion quality in the presence of AWG noise in the input image stack.

Image denoising is a very well studied problem. The most effective denoising methods developed in recent years include *BiShrink* [Sendur and Selesnick, 2002], *BLS-GSM* [Portilla et al., 2003], the *BM3D* method [Dabov et al., 2007] and the *ProbShrink* method for wavelets [Pižurica and Philips, 2006]. The curvelet transform already used for image fusion in Chapter 2 also holds the potential to improve existing image denoising techniques because of its truly two-dimensional nature and its associated high directional sensitivity. For our purpose, using a curvelet-based denoiser is also advantageous because it reduces the computational requirements of the combined denoising and fusion algorithm. The curvelet transform, already used for image fusion in Chapter 2, also for image denoising holds the potential to improve existing techniques because of its truly two-dimensional nature and its associated high directional sensitivity. For our purpose, using a curvelet-based denoiser is also advan-

tageous because it reduces the computational requirements of the combined denoising and fusion algorithm. More precisely, as the fusion method is performed after each slice has been denoised using a curvelet-based technique, a curvelet transform per slice is saved. On an AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set, in Matlab code, averaged over 10 experiments the curvelet transform of a 512×512 image took 3.36 s to execute. This means that for a stack of 15 512×512 slices, 50.41 s of computation time can be saved.
In the following Sections 3.2 to 3.7 we present the curvelet-based denoising method that we have developed and show that it produces results that are close to those of state-of-the-art denoisers. In Section 3.8 we show how denoising prior to fusion improves the performance of our fusion algorithm in the presence of noise.

3.2 Denoising in the Curvelet Domain

As mentioned in Section 2.3, the curvelet transform is one in a series of new geometric transforms that have been developed in recent years. The potential that these geometric transforms hold for denoising of images has been investigated by many researchers, e.g., in [Candès, 2001; da Cunha et al., 2006; Po and Do, 2006; Starck et al., 2002]. As is the case when denoising images using the classical wavelet transform, noise reduction in the new transform domains results from greatly reducing the magnitude of coefficients that contain primarily noise, while reducing others less. Thresholding as it has been applied in the wavelet domain [Donoho, 1995] has also been successfully used in the curvelet and the contourlet domains [da Cunha et al., 2006; Starck et al., 2002]. Optimizing the choice of the threshold between these two classes of coefficients improves the denoising performance of a method, for wavelets [Abramovich et al., 1998; Chang et al., 2000a] as well as for other transforms [da Cunha et al., 2006].
A very broad class of wavelet-based denoisers is based on estimating the noise-free coefficients by minimizing a Bayesian risk, either by minimum mean squared error or maximum a posteriori estimation [Chipman et al., 1996; Clyde et al., 1998; Moulin and Liu, 1999; Simoncelli and Adelson, 1996]. These methods are optimized with respect to the marginal statistics of the coefficients within each sub-band by imposing a prior distribution on the noise-free transform coefficients. A particular success has been exhibited by denoising methods where the local context is considered in the choice of one or more parameters of the prior model [Chang et al., 2000b; Guerrero-Colon and Portilla, 2005; Mihcak et al., 1999; Pižurica and Philips, 2006; Portilla et al., 2003; Romberg et al., 1999; Sendur and Selesnick, 2002]. Recently, Po et al. have transferred this reasoning to the contourlet domain [Po and Do, 2006]. To enable this transfer, marginal and joint image statistics on oriented multi-scale pyramids, of which curvelets are a special case, have previously been studied by Po et al. for the contourlet case and by Boubchir et al. and Alecu et al. for the curvelet

case [Alecu et al., 2006; Boubchir and Fadili, 2005a,b; Po and Do, 2006]. Additionally, Boubchir et al. have proposed a multivariate prior model for curvelet coefficients [Boubchir and Fadili, 2005a,b].
Over the past years mixture priors have been shown very effective in wavelet processing [Abramovich et al., 1998; Abramovich and Sapatinas, 1999; Abramovich et al., 2002; Chipman et al., 1996; Clyde et al., 1998; Pižurica and Philips, 2006; Vidakovic, 1998; Vidakovic and Ruggeri, 2001]. During this PhD work, we have investigated how to develop a related prior for curvelet coefficients [Tessens et al., 2006a], more in particular similar to the one proposed by the authors in [Pižurica and Philips, 2006]. As part of this endeavor, we have extended the statistical analyses of [Alecu et al., 2006; Boubchir and Fadili, 2005a,b; Po and Do, 2006] by investigating the different behavior of curvelet coefficients containing a significant noise-free component on the one hand, and coefficients in which such a "signal of interest" is absent on the other hand [Tessens et al., 2006a,c]. Based on our findings and inspired by the wavelet domain *ProbShrink* estimator [Pižurica and Philips, 2006], we have also defined and analyzed different intra-band [Tessens et al., 2006a] and inter-band [Tessens et al., 2006c] local spatial activity indicators (LSAIs) in the curvelet domain. In [Tessens et al., 2008c], next to carrying out our previous statistical studies in a more comprehensive way, we have introduced and analyzed a new LSAI that includes both intra- and inter-band dependent curvelet coefficients. Using this new LSAI in the curvelet-based denoising method *ProbShrinkCurv* that we have developed in [Tessens et al., 2006a] has allowed us to improve upon our previous denoising results, reported in [Tessens et al., 2006a] and [Tessens et al., 2006c].
The remainder of this chapter is organized as follows: in Section 3.3, the notations and terminology used in this chapter are introduced. In Section 3.4, a comparative statistical analysis of the two classes of curvelet coefficients mentioned above is presented. Section 3.5 discusses a novel curvelet-based context adaptive denoising method, whereas Section 3.6 studies the parameter that marks the threshold between the coefficient classes. Section 3.7 summarizes the main results and conclusions of this work on curvelet-based denoising. In Section 3.8 we use the developed *ProbShrinkCurv* denoiser to improve the performance of our fusion algorithm in the presence of noise. The conclusions of this chapter are presented in Section 3.9.

3.3 Terminology and Notations

We will use the same notations and terminology as Po et al. [Po and Do, 2006], Boubchir et al. [Boubchir and Fadili, 2005b] and Alecu et al. [Alecu et al., 2006]. Given a curvelet coefficient X, we will denote by

- N_i the neighboring curvelet coefficient in the same sub-band (all neighbors are numbered from 1 to 8, starting with the neighbor located at the upper left and proceeding clockwise).

- C_k the cousin curvelet coefficient located at the same relative position as X in a different sub-band at the same scale. k denotes the sub-band (all sub-bands within each corona are numbered, starting from 1 with the sub-band at the upper left and proceeding clockwise).
  - C_k is an *adjacent* cousin of X if sub-band k lies next to the sub-band in which X is located.
  - C_k is an *opposing* cousin of X if sub-band k lies opposite the sub-band in which X is located, or in other words, if C_k is the real (imaginary) part of the complex curvelet coefficient of which X is the imaginary (real) part.
- P the parent curvelet coefficient, located at the same relative position as X in the same sub-band but at a coarser scale.

3.4 Curvelet Statistics

Image statistics in the curvelet domain have been recently studied by Boubchir et al. [Boubchir and Fadili, 2005a,b] and Alecu et al. [Alecu et al., 2006], with a comparative analysis to wavelet domain statistics. Po et al. [Po and Do, 2006] have done the same for the contourlet transform. We will now take a step further in this direction by analyzing the statistics of two classes of curvelet coefficients: those containing a significant noise-free component (to which we will refer as *significant* coefficients), and the coefficients in which no signal of interest is present (which we will call *insignificant*). Note that for this statistical analysis we will make use of both noise-free and noisy image versions. The use of noise-free image versions will enable the development of contextual models that we employ in the actual denoising procedure presented in Section 3.5, where noise-free image versions are not available.
In our approach, significant coefficients are defined as those that have a *noise-free* component larger, in absolute value, than a threshold T. We call such a component our "signal of interest". In our statistical analysis experiments, we determine the locations of the significant *noisy* curvelet coefficients by thresholding their noise-free counterparts. Hence, a noisy curvelet coefficient is marked as significant if the corresponding curvelet coefficient of the noise-free image version exceeds a threshold T in magnitude.
An important issue at this point is the choice of the parameter T. This choice cannot be considered independently from the goal of this statistical analysis: developing a denoising method for curvelets which is aimed at minimizing the mean squared error between the denoised and the noise-free image (the method will be described in detail in Section 3.5). Therefore, we postpone a discussion of this parameter to Section 3.6. Let it suffice for now to say that the threshold will be related to the standard deviation of the noise through a constant factor.
All the statistics in this section have been obtained from the curvelet decompositions of images contaminated with additive white Gaussian (AWG) noise

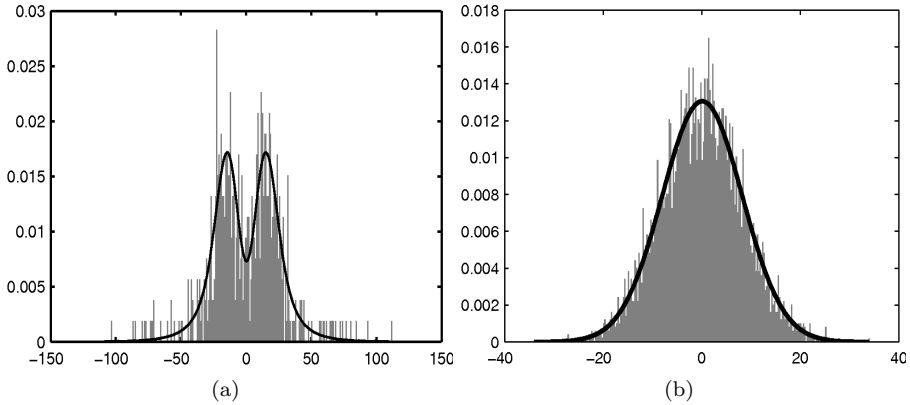


Figure 3.2: Histograms of noisy curvelet coefficients (a) with noise-free component larger than a threshold T and (b) in which no signal of interest is present. Overlayed on the histograms, the estimated pdfs of the significant and insignificant noise-contaminated curvelet coefficients: $f(x|H^{0,1}) * \phi(0, \sigma')$.

with standard deviation $\sigma = 20$, unless explicitly mentioned otherwise. The threshold between significant and insignificant coefficients was set to $1.3\sigma'$, with σ' the standard deviation of the noise in the sub-bands.

3.4.1 Marginal Statistics

Fig. 3.2a shows the 256-bin histogram of the significant curvelet coefficients of a sub-band of the finest scale of the curvelet decomposition of a noisy version of *Peppers*. Fig. 3.2b shows the same but for the insignificant curvelet coefficients of this sub-band.

In [Alecu et al., 2006; Boubchir and Fadili, 2005a,b] the authors showed that the probability density functions (pdfs) of *noise-free* curvelet coefficients x follow well a generalized Laplacian (also called generalized Gaussian) distribution. In the following, we denote the probability density function of x by $f(x)$, so

$$f(x) = \frac{\nu}{2s\Gamma(\frac{1}{\nu})} \exp\Big(-|\frac{x}{s}|^{\nu}\Big), \tag{3.2}$$

where s and ν are parameters of the generalized Laplacian distribution.

We adopt the modeling framework proposed by Pižurica and Philips [Pižurica and Philips, 2006] for the significant and insignificant wavelet coefficients, and we apply it in the curvelet domain as follows. Let H^1 denote the hypothesis that a curvelet coefficient x is significant, and let H^0 denote the opposite. By our definition of significant curvelet coefficients, the pdf of these coefficients can be modeled by the tails of the generalized Laplacian that models the pdf

$f(x)$ of x:

$$f(x|H^1) = Af(x) \text{ for } |x| > T \text{ and } f(x|H^1) = 0 \text{ otherwise,} \tag{3.3}$$

with A a normalizing constant. Similarly, the pdf of noise-free insignificant coefficients has the shape of the central part of $f(x)$, or:

$$f(x|H^0) = Bf(x) \text{ for } |x| \leq T \text{ and } f(x|H^0) = 0 \text{ otherwise,} \tag{3.4}$$

where B is a normalizing constant.

Now we investigate the distributions of the *noisy* curvelet coefficients when the input noise is AWG with standard deviation σ. The curvelet transform, which is a linear transform, transforms AWG noise into additive correlated Gaussian noise in each sub-band. The first order pdf of the noise in the sub-bands is a normal distribution $\phi(0, \sigma')$. Since the second generation curvelet transform we are using corresponds to a tight frame, the standard deviation σ' is $\sigma/\sqrt{\alpha}$, with α the redundancy factor of the transform. The pdfs of the significant and insignificant noisy curvelet coefficients can be modeled by the distribution of their respective noise-free counterparts, convolved with this normal distribution $\phi(0, \sigma')$: $f(x|H^{0,1}) * \phi(0, \sigma')$.We estimate $f(x)$, necessary for the calculation of $f(x|H^1)$ and $f(x|H^0)$, from the noise contaminated curvelet coefficients by using the method of Simoncelli et al. [Simoncelli and Adelson, 1996] for estimating the generalized Laplacian distribution from wavelet coefficients contaminated with additive *white* Gaussian noise. Although, as pointed out before, the noise in the curvelet sub-bands is not white, Fig. 3.2 shows that this approximate model matches well with observations.

Note that the proposed prior model belongs to a broader class of finite mixtures of two distributions, one modeling the statistics of "significant" coefficients and the other one of the "insignificant" coefficients [Abramovich et al., 1998; Abramovich and Sapatinas, 1999; Abramovich et al., 2002; Chipman et al., 1996; Clyde et al., 1998; Johnstone and Silverman, 2005; Vidakovic, 1998; Vidakovic and Ruggeri, 2001]. In earlier models of this class, the mixed distributions are usually two normal distributions (e.g., in [Chipman et al., 1996]), a normal distribution and a point mass at zero (e.g., in [Abramovich et al., 1998; Clyde et al., 1998]) or a Laplacian distribution and a point mass at zero [Johnstone and Silverman, 2005]. For such prior models, the mixing proportion (i.e., $P(H^0)$) as well as the hyperparameters are usually estimated jointly using a maximum likelihood (ML) estimator with an expectation-maximization (EM) algorithm. The proposed prior, on the contrary, has the double advantage that the parameter estimation procedure is simpler (no iterative joint estimation necessary) and that it can cope with a more complicated model of the noise-free coefficients (generalized Laplacian). The only parameter which cannot be directly estimated from the data is the threshold T. The choice of this parameter is discussed in Section 3.6.

3.4.2 Joint Statistics

Previous studies revealed that noise-free curvelet coefficients are strongly correlated in local intra-band neighborhoods and that these local correlations are stronger than their inter-scale and inter-orientation counterparts [Alecu et al., 2006; Boubchir and Fadili, 2005a,b]. We will now investigate whether the joint statistics of significant coefficients differ from the joint statistics of coefficients in which no signal of interest is present. Such a difference would facilitate the denoising of the coefficients. More in particular, we will focus on the *magnitude* of the coefficients.

The correlation coefficients in this section have been calculated as the average of the correlation coefficients obtained from the magnitudes of the next to highest frequency scale curvelet coefficients of a test set of 44 images. Note that if the insignificant coefficients were pure noise, their correlation coefficients could also be obtained by calculating the sample covariance matrix of the curvelet decomposition of a scaled delta function (which has the same power spectrum as AWGN) [Portilla et al., 2003]. However, although the relative influence of the noise is much bigger on the insignificant coefficients than on the significant ones, insignificant coefficients are not exactly the same as noise, and we will therefore not adopt this theoretical method for the computation of their correlation coefficients.

As test images we chose the images from the *Miscellaneous* volume of the USC-SIPI image database (converted to gray-scale) [USC-SIPI, 2009]. We deliberately chose different images than the ones on which we test our fusion algorithm to avoid developing a tailor-made denoiser that only works on that test set. To evaluate the general validity of the results better, we also report the correlation coefficients for 4 specific images: *House*, *Peppers*, *Barbara* and *Baboon*.

3.4.2.1 Intra-Band Correlations

In [Alecu et al., 2006; Boubchir and Fadili, 2005a,b], it was shown that each curvelet coefficient is strongly correlated with its eight direct neighbors. Alecu et al. additionally showed that the correlation is not equally strong for all the neighbors. By construction, curvelet coefficients are more correlated with the neighbors that lie in the direction parallel to the main direction of the curvelet function by which they were produced. In this section, we examine whether the same holds for the two categories of coefficients that are of interest to us : the significant and the insignificant ones.

Recall from Section 2.3 that in the wrapping-based digital curvelet implementation, which we use throughout this dissertation, the spatial grid on which the curvelets are translated at each scale and orientation is a regular rectangular grid, not one adapted to the orientation of the curvelet (as in Fig. 2.1). To recover the neighboring coefficients of a curvelet coefficient along the main direction of the curvelet basis function, interpolation is needed.

Consider for example a curvelet coefficient and its four neighbors, arranged as

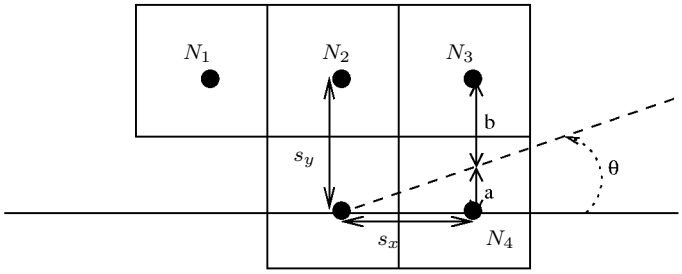


Figure 3.3: A curvelet coefficient and four of its neighbors (marked by N_i). The curvelet basis function is oriented in the direction θ.

in Fig. 3.3. As the direction of the curvelet function, characterized by θ, does not coincide with one of the grid lines, none of the four neighbors lies exactly in the direction of the curvelet. N_3 and N_4 are closest and we therefore linearly interpolate them :

$$N' = \frac{a}{a+b}|N_3| + \frac{b}{a+b}|N_4|, \tag{3.5}$$

with $a = s_x tan\theta$ and $b = s_y - s_x tan\theta$, where s_x is the horizontal and s_y the vertical sampling period. For other values of θ, N_3 and N_4 need to be substituted by the appropriate neighbors and the interpolation weights a and b should be adapted accordingly.

In the following we will always analyze the correlation between a reference coefficient and this coefficient N', calculated from its neighbors. We will refer to N' as the neighbor lying in the direction of highest or maximal correlation. Of course, by symmetry, each coefficient has two such neighbors. A similar reasoning applies to the analysis of correlation in the direction perpendicular to the direction of highest correlation, i.e., the direction of smallest or minimal correlation.

Table 3.2a shows in the second column the mean correlation coefficient between the significant coefficient magnitudes and one of their two neighbors in the direction of maximal correlation, calculated from the next to highest scale sub-bands of our image test set. Table 3.2a column 4 shows the mean correlation coefficient between the insignificant coefficient magnitudes and one of their two neighbors in the direction of maximal correlation. These big mean correlation coefficients match with theoretical expectations of high correlation in the direction of the curvelet basis function. In the perpendicular direction of *minimal* correlation, one observes that this correlation has virtually disappeared for the insignificant coefficients and has become very small for the significant ones (cfr. Table 3.2b). For the significant coefficients, the standard deviation of the correlation coefficients calculated over the image test set is rather high. This means that correlation with this kind of coefficients is highly image-dependent. Indeed, correlation is quite high for *Barbara* and *Baboon* but has almost disappeared for *Peppers* and *House* (see Table 3.3b). Unlike

Coefficients conditioned on	Correlation coefficients			
	Significant coeff.		Insignificant coeff.	
	Mean	Std. Dev.	Mean	Std. Dev.
a) Neighbors max correlation	0.46	0.06	0.26	0.04
b) Neighbors min correlation	0.10	0.11	0.03	0.02
c) Adjacent cousins	0.16	0.08	0.03	0.02
d) Opposing cousins	0.14	0.07	0.02	0.02
e) Parents	0.17	0.07	0.04	0.02

Table 3.2: Mean correlation coefficients between significant curvelet coefficient magnitudes on the one hand and on the other hand (a) neighbors in the direction of maximal correlation, (b) neighbors in the direction of minimal correlation, (c) adjacent cousins, (d) opposing cousins and (e) parents. Similar for insignificant curvelet coefficients.

Barbara and *Baboon*, *Peppers* and *House* both correspond better to the image model for which curvelets are especially suited, namely piece-wise smooth with discontinuities along curvilinear edges. Indeed, *Barbara* and *Baboon* both have a less sparse curvelet representation than *Peppers* and *House*. In a decomposition into 4 scales and with 16 orientations at the coarsest level, 11.81% of the coefficients is classified as significant for *Baboon* and 5.08% for *Barbara* vs. only 2.62% for *Peppers* and 3.49% for *House*. Thus the achieved decorrelation of the transform coefficients is higher for these two images.
As mentioned at the start of Section 3.4, the images from the test set were contaminated with AWGN with $\sigma = 20$ to obtain the correlation coefficients in Table 3.2. Figure 3.4 shows the evolution of the correlation coefficients between (in)significant coefficients and neighbors in the direction of highest and lowest correlation as a function of the standard deviation σ of the AWGN with which the images in the test set were contaminated. In order not to overload the graph, error bars were omitted. The authors have verified that the standard deviations of the correlation coefficients are approximately constant for all values of σ, and for $\sigma = 20$, these numbers can be found in Table 3.2. For significant coefficients, the relative relationship of the two curves is maintained for all values of σ. For insignificant coefficients, correlation with neighbors in the direction of minimal correlation remains negligible, regardless of σ, whereas the correlation with neighbors in the direction of maximal correlation displays a rising trend.

3.4.2.2 Inter-Band Dependencies

In the studies of Boubchir et al. [Boubchir and Fadili, 2005a,b] and Alecu et al.[Alecu et al., 2006], it was found that curvelet coefficients of different subbands are approximately *decorrelated*, but some *dependencies* between sub-

Coefficients conditioned on	Significant coefficients			
	House	Peppers	Barbara	Baboon
a) Neighbors max correlation	0.43	0.50	0.51	0.45
b) Neighbors min correlation	0.07	0.08	0.35	0.19
c) Adjacent cousins	0.11	0.19	0.11	0.18
d) Opposing cousins	0.08	0.19	0.22	0.19
e) Parents	0.22	0.25	0.04	0.17

Table 3.3: Correlation coefficients for some specific images between significant curvelet coefficient magnitudes on the one hand and on the other hand (a) neighbors in the direction of maximal correlation, (b) neighbors in the direction of minimal correlation, (c) adjacent cousins, (d) opposing cousins and (e) parents.

Coefficients conditioned on	Insignificant coefficients			
	House	Peppers	Barbara	Baboon
a) Neighbors max correlation	0.28	0.28	0.27	0.23
b) Neighbors min correlation	0.02	0.02	0.03	0.03
c) Adjacent cousins	0.02	0.03	0.04	0.03
d) Opposing cousins	0.02	0.02	0.02	0.03
e) Parents	0.03	0.04	0.03	0.03

Table 3.4: Correlation coefficients for some specific images between insignificant curvelet coefficient magnitudes on the one hand and on the other hand (a) neighbors in the direction of maximal correlation, (b) neighbors in the direction of minimal correlation, (c) adjacent cousins, (d) opposing cousins and (e) parents.

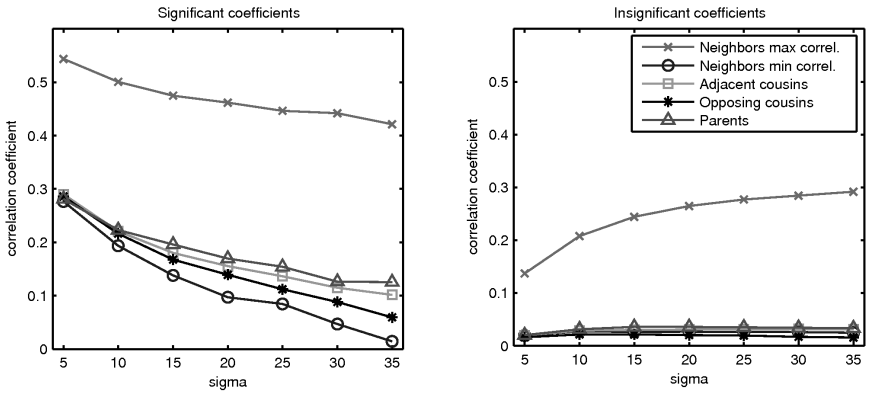


Figure 3.4: Correlation coefficients between (in)significant coefficients and neighbors in the direction of maximal and minimal correlation, adjacent and opposing cousins and parents, as a function of the standard deviation σ of the contaminating additive white Gaussian noise.

bands do exist. Boubchir et al. and Alecu et al. observed dependency between a curvelet coefficient and its parent, as well as between a curvelet coefficient and its cousins. The strength of these inter-orientation dependencies decreases with the increase of the difference in orientation, but one also observes dependency with respect to the opposite orientation [Alecu et al., 2006]. We will now further extend this study of inter-band curvelet statistics to the two categories of coefficients that we consider in this study: the significant and the insignificant curvelet coefficients.

The average correlation coefficients between significant, resp. insignificant coefficient magnitudes and the magnitudes of their adjacent and opposing cousins and their parents are indicated in Table 3.2c-e. The same observations can be made here as in the previous section for the correlation coefficients in the direction of smallest correlation : correlation has virtually disappeared for the insignificant coefficients and is small for the significant ones. Fig. 3.4 confirms that this observation also holds for other values of the standard deviation of the contaminating AWGN. The high standard deviation of the correlation coefficients for the significant coefficients implies that the correlation is highly image-dependent (see Table 3.2c-e). This is confirmed by the values in Table 3.3c-e.

3.4.3 Local Spatial Activity Indicators

Inspired by the *ProbShrink* wavelet domain denoising method of [Pižurica and Philips, 2006], we now define and analyze different local spatial activity indicators (LSAIs) in the curvelet domain. In general, for each curvelet coefficient we define the LSAI as a function of those coefficients that are well correlated

when the coefficient is significant. We have investigated this last property in the previous section about joint curvelet statistics (Section 3.4.2), and this study leads us to propose four LSAIs that will potentially perform well in the curvelet denoiser that will be developed in Section 3.5: two intra-band [Tessens et al., 2006a], one inter-band [Tessens et al., 2006c] and one novel combined intra-inter-band LSAI. We will evaluate these LSAIs, firstly by investigating the correlation between the magnitudes of the curvelet coefficients and their corresponding LSAI, since this is the correlation which is exploited in our newly developed denoising method (see Section 3.5), and secondly by plugging them into this denoising method. The second evaluation will be performed in the next section.
As in the previous section, the correlation coefficients in this section have been calculated as the average correlation coefficients obtained from the next to highest frequency scale sub-bands of the USC-SIPI image test set, and results are also reported for 4 specific images (*House*, *Peppers*, *Barbara* and *Baboon*).

3.4.3.1 Anisotropic Intra-Band LSAIs

Because of the correlation properties of curvelet coefficients, an anisotropic LSAI seems appropriate. We propose two anisotropic LSAI candidates z that are calculated as the mean absolute value of the $n-1$ coefficients N' within a small $1 \times n$ neighborhood δ (that excludes the reference coefficient itself) oriented in the direction of either highest or lowest correlation:

$$z = \frac{1}{n-1} \sum_{i \in \delta_{(un)corr}} |N'_i|. \tag{3.6}$$

The coefficients N'_i within this window δ are interpolated from their neighbors as explained in Section 3.4.2.1, formula 3.5.
In Table 3.5a-b the average correlation coefficients between the curvelet coefficient magnitudes of the next to finest scale sub-bands of the images from the test set and the LSAI oriented in the direction of maximal and minimal correlation are indicated (in both cases, n is set to 5).
The correlation coefficients for these LSAIs follow the trend of those between a coefficient and its neighbor, discussed in Section 3.4.2.1 (see Table 3.2a-b). Indeed, the anisotropic LSAI oriented in the direction of maximal correlation is highly correlated with both the significant and the insignificant coefficients. Again, this correlation is higher for the significant coefficients than for the insignificant ones. For both classes, it is also bigger than the correlation between a coefficient and just one neighbor, indicated in Table 3.2a, as the LSAI summarizes information from more coefficients (here 4 as opposed to only 1) and can also capture correlations over longer distances. For the anisotropic LSAI oriented in the direction of minimal correlation, correlation is low in both cases, although slightly higher for the significant coefficients. Again, the standard deviation of the correlation coefficients is very high in the significant case, which means that this correlation is highly image-dependent (see Tables 3.6 and 3.7

for correlation coefficients of some specific images). We can again observe that the decorrelation of the curvelet coefficients in the direction of minimal correlation is highest for *Peppers* and *House*, both sparsely represented in the curvelet domain.

3.4.3.2 Adjacent, Opposing and Parents (AOP) Inter-Band LSAI

In [Tessens et al., 2006c], we have defined and discussed several inter-band LSAIs. These different LSAIs were calculated for each curvelet coefficient as the average magnitude of the adjacent cousins; the adjacent and opposing cousins; the adjacent cousins and the parent; or the adjacent and opposing cousins and the parent (AOP). The last LSAI proved to be the best performing one in terms of denoising capabilities. Therefore, we will discuss only this inter-band LSAI here.
For a coefficient y in a sub-band k, it is defined as

$$z = \frac{1}{4}\Big(|C_{k-1}| + |C_{(k+1) \bmod \mathrm{K}}| + |C_{(k+K/2) \bmod \mathrm{K}}| + |P|\Big), \qquad (3.7)$$
$$k \in \{1, \ldots, K\} \text{ and } \mathrm{C}_0 = \mathrm{C}_\mathrm{K}$$

where *mod* stands for the modulo operation, K is the number of orientations at the scale to which y belongs and notations introduced in Section 3.3 are used. In Table 3.5c the average correlation coefficients between the curvelet coefficient magnitudes and the AOP LSAI are indicated. One can notice that the insignificant curvelet coefficients are approximately decorrelated with this LSAI whereas between the significant coefficients and this LSAI some correlation exists. From Tables 3.6c and 3.7c we can see that this behavior is present for all our example images.

3.4.3.3 Combined Intra- and Inter-Band LSAI

In order to exploit both the intra- and inter-band correlations between curvelet coefficients, we now define a novel LSAI that combines the best performing intra- and inter-band LSAIs. Specifically, we define an intra-inter-band (IIB) LSAI as the average of the anisotropic intra-band LSAI oriented in the direction of lowest correlation and the AOP inter-band LSAI :

$$z = \frac{1}{2}\Bigg[\frac{1}{4}\Big(|C_{k-1}| + |C_{k+1 \bmod \mathrm{K}}| + |C_{k+K/2 \bmod \mathrm{K}}| + |P|\Big) + \frac{1}{n-1}\sum_{i \in \delta_{uncorr}} |N_i'|\Bigg], k \in \{1, \ldots, K\} \text{ and } \mathrm{C}_0 = \mathrm{C}_\mathrm{K} \qquad (3.8)$$

In Table 3.5d the average correlation coefficients over our image test set between the curvelet coefficient magnitudes and this candidate LSAI are indicated. From Table 3.5d, one can again notice that the insignificant curvelet coefficients are approximately decorrelated with this LSAI. The average correlation coefficient for the significant coefficients is lower than in the AOP LSAI case.

LSAI	Correlation coefficients			
	Significant coeff.		Insignificant coeff.	
	Mean	Std. Dev.	Mean	Std. Dev.
a) An. LSAI max correlation	0.62	0.06	0.33	0.06
b) An. LSAI min correlation	0.12	0.13	0.04	0.03
c) AOP LSAI	0.26	0.09	0.06	0.02
d) Combined IIB LSAI	0.20	0.12	0.05	0.03

Table 3.5: Average correlation coefficients between the magnitudes of significant curvelet coefficients and (a) the anisotropic LSAI oriented in the direction of highest correlation, (b) the anisotropic LSAI oriented in the direction of lowest correlation, (c) the adjacent, opposing and parents LSAI and (d) the combined intra- and inter-band (IIB) LSAI. Similar for insignificant curvelet coefficients.

Coefficients conditioned on	Significant coefficients			
	House	Peppers	Barbara	Baboon
a) An. LSAI max correlation	0.58	0.67	0.65	0.60
b) An. LSAI min correlation	0.11	0.07	0.42	0.27
c) AOP LSAI	0.27	0.35	0.25	0.30
d) Combined IIB LSAI	0.20	0.21	0.44	0.31

Table 3.6: Correlation coefficients for some specific images between the magnitudes of significant curvelet coefficients on the one hand and on the other hand (a) the anisotropic LSAI oriented in the direction of highest correlation, (b) the anisotropic LSAI oriented in the direction of lowest correlation, (c) the adjacent, opposing and parents LSAI and (d) the combined intra- and inter-band (IIB) LSAI.

Coefficients conditioned on	Insignificant coefficients			
	House	Peppers	Barbara	Baboon
a) An. LSAI max correlation	0.35	0.37	0.33	0.27
b) An. LSAI min correlation	0.03	0.04	0.05	0.04
c) AOP LSAI	0.05	0.06	0.07	0.06
d) Combined IIB LSAI	0.04	0.05	0.06	0.05

Table 3.7: Correlation coefficients for some specific images between the magnitudes of insignificant curvelet coefficients on the one hand and on the other hand (a) the anisotropic LSAI oriented in the direction of highest correlation, (b) the anisotropic LSAI oriented in the direction of lowest correlation, (c) the adjacent, opposing and parents LSAI and (d) the combined intra- and inter-band (IIB) LSAI.

3.5 Context Adaptive Image Denoising using Curvelets

Based on our findings of Sections 3.4.1 and 3.4.3, we now develop a curvelet domain version *ProbShrinkCurv* of the *ProbShrink* denoising method [Pižurica and Philips, 2006].

3.5.1 The *ProbShrinkCurv* Denoiser

Consider an input image, contaminated with additive white Gaussian noise. After transforming the image to the curvelet domain, the noise is transformed into additive correlated Gaussian noise in each sub-band. This correlation is not modeled in the following. Let y_l denote, for a given scale and orientation, the curvelet coefficient at position l. Let y_l be composed of an unknown noise-free curvelet coefficient x_l and of a noise component n_l: $y_l = x_l + n_l$, where the variables n_l are identically distributed Gaussian random variables which are statistically independent from y_l. Let H_l^1 denote the hypothesis that x_l represents a significant image feature and H_l^0 the hypothesis that x_l contains no signal of interest. The hypothesis H_l^1 is specified as $|x_l| \geq T$, whereas H_l^0 is equivalent to $|x_l| < T$, with T a chosen threshold (see Section 3.6). Finally, let z_l be any arbitrary indicator of the local spatial activity, defined as in Section 3.4.3.

The minimum mean squared error (MMSE) estimate of x_l is [Ephraim and Malah, 1984; McAulay and Malpass, 1980]

$$\hat{x}_l = E(x_l|y_l, z_l) = E(x_l|y_l, z_l, H_l^1)P(H_l^1|y_l, z_l) + E(x_l|y_l, z_l, H_l^0)P(H_l^0|y_l, z_l).$$

As H_l^0 refers to the absence of a signal of interest, $E(x_l|y_l, z_l, H_l^0) = 0$. We further assume we can approximate $E(x_l|y_l, z_l, H_l^1)$ by y_l. This leads to

$$\hat{x}_l = P(H_l^1|y_l, z_l)y_l. \tag{3.9}$$

Using Bayes rule, we can rewrite this expression as

$$\hat{x}_l = \frac{\Lambda_l}{1 + \Lambda_l} y_l \tag{3.10}$$

where Λ_l is the general likelihood ratio $\Lambda_l = \rho_l \zeta_l \nu_l$ with $\rho_l = P(H_l^1)/P(H_l^0)$, $\zeta_l = p(z_l|H_l^1)/p(z_l|H_l^0)$ and $\nu_l = p(y_l|H_l^1)/p(y_l|H_l^0)$. Applying the inverse curvelet transform to the estimated noise-free curvelet coefficients $\hat{x}_l$ yields the denoised image.

We will now comment in detail on the calculation of each of the factors of Λ_l, namely ρ_l, ν_l and ζ_l.

3.5.2 Calculation of the Generalized Likelihood Ratio

ρ_l can be rewritten as [Pižurica and Philips, 2006]

$$\rho_l = \frac{P(H_l^1)}{P(H_l^0)} = \frac{\int_{-\infty}^{+\infty} f(x_l|H^1)dx_l}{\int_{-\infty}^{+\infty} f(x_l|H^0)dx_l}$$
$$= \frac{\int_{-\infty}^{-T} f(x_l)dx_l + \int_{T}^{\infty} f(x_l)dx_l}{\int_{-T}^{T} f(x_l)dx_l} = \frac{1 - \int_{-T}^{T} f(x_l)dx_l}{\int_{-T}^{T} f(x_l)dx_l}. \quad (3.11)$$

It has been shown in [Alecu et al., 2006; Boubchir and Fadili, 2005a,b] that the noise-free curvelet coefficients x follow well a generalized Laplacian distribution. So with their pdf $f(x_l) = \frac{\nu}{2s\Gamma(\frac{1}{\nu})} \exp\Big(-|\frac{x_l}{s}|^\nu\Big)$ and by substituting $t = (\frac{x_l}{s})^\nu$, we find that

$$\int_{-T}^{T} f(x_l)dx_l = \frac{\nu}{s\Gamma(\frac{1}{\nu})} \int_0^T \exp\Big(-(\frac{x_l}{s})^\nu\Big)dx_l$$
$$= \frac{1}{\Gamma(\frac{1}{\nu})} \int_0^{(\frac{T}{s})^\nu} t^{\frac{1}{\nu}-1} e^{-t} dt = \Gamma_{inc}\Big((\frac{T}{s})^\nu, \frac{1}{\nu}\Big), \quad (3.12)$$

where $\Gamma_{inc}(y,a) = \frac{1}{\Gamma(a)} \int_0^y t^{a-1} e^{-t} dt$ is the incomplete gamma function. ρ_l now becomes

$$\rho_l = \frac{1 - \Gamma_{inc}\Big((\frac{T}{s})^\nu, \frac{1}{\nu}\Big)}{\Gamma_{inc}\Big((\frac{T}{s})^\nu, \frac{1}{\nu}\Big)}. \quad (3.13)$$

For the calculation of $\nu_l = p(y_l|H_l^1)/p(y_l|H_l^0)$, we have shown in Section 3.4.1 that $p(y_l|H_l^{0,1})$ can be modeled as $f(x|H^{0,1}) * \phi(0,\sigma')$, with $f(x|H^{0,1})$ as defined in Eqs. 3.3 and 3.4. As both σ' and $f(x_l) = \frac{\nu}{2s\Gamma(\frac{1}{\nu})} \exp\Big(-|\frac{x_l}{s}|^\nu\Big)$ can be estimated from the noisy coefficients [Donoho and Johnstone, 1994; Simoncelli and Adelson, 1996], this allows us to obtain ν_l.

The calculation of ζ_l depends on which of the LSAIs proposed in Section 3.4.3 is used. The choice of this LSAI will be based on the study of joint curvelet statistics of Section 3.4.3. We will discuss this choice later on in this section. First we focus on the derivation of the pdf of the LSAI z_l, conditioned on either the hypothesis H_l^0 or H_l^1. For compactness of notation, we will suppress the position index l in what follows.

When z is an intra-band LSAI, i.e., $z = \frac{1}{n-1} \sum_{i \in \delta_{(un)corr}} |N_i'|$, it is calculated from coefficients that lie within a small spatial neighborhood δ around the central coefficient. The statistical characterization of the LSAI is greatly simplified by assuming, as in [Mihcak et al., 1999; Pižurica and Philips, 2006], that all the coefficient magnitudes $|y|$ within this small neighborhood, including the ones that are interpolated from their neighbors, are identically distributed and are *conditionally* independent (given H^0 or H^1). Under these assumptions,

$p(z|H^0)$ can be obtained by convolving $p(|y||H^0)$ $n-1$ times with itself, and $p(z|H^1)$ similarly.
To verify experimentally that the assumption of conditional independence holds, we have plotted in Figure 3.5a the joint histogram of the magnitudes of a significant coefficient and its significant neighbors within a 1×5 spatial neighborhood δ, for a sub-band of the next-to-highest scale of *Barbara*, contaminated with AWGN with $\sigma = 10$. In this experiment δ is oriented in the direction of minimal correlation because this neighborhood will prove to be the most useful for denoising in the next section. We have verified that experiments with δ oriented in the direction of maximal correlation lead to similar conclusions. The correlation coefficient between the magnitudes of significant coefficients and their neighbors in the direction of minimal correlation is 0.44 for the sub-band used in this experiment. In Figure 3.5b the approximation by a conditionally independent model $p(|y||H^1)*p(|y||H^1)$ is shown. A comparison of Figures 3.5a and b reveals that the model matches well with the empirical histogram, despite the correlation between the coefficients. Figures 3.5c and d show the same but for *Baboon*, contaminated with AWGN with $\sigma = 20$. For this sub-band, the correlation coefficient between significant coefficient magnitudes and their neighbors in the direction of minimal correlation is 0.07.
For the calculation of the pdf of the AOP inter-band LSAI, we make similar assumptions. More precisely, we assume that the coefficients from other sub-bands that are incorporated in the LSAI are insignificant if y_l is insignificant and similarly for the significant case. We also assume that these coefficients are *conditionally* independent (given H^0 or H^1). Thus, in this case, $p(z|H^0)$ can be approximated by convolving $p(|y||H^0)$ with the pdfs of the coefficient magnitudes of each other sub-band, conditioned on H^0, and similar for H^1.
As the combined intra- and inter-band LSAI is computed as the average of the intra-band LSAI oriented in the direction of minimal correlation and the AOP LSAI, its pdf can be obtained through the convolution of the pdfs of the LSAIs it is calculated from.

3.5.3 Choice of the LSAI

We will now evaluate the denoising potential of the LSAIs that we have proposed and studied in Section 3.4.3 based on the results of Section 3.4.2. We have used the *ProbShrinkCurv* method (4 scales in the curvelet decomposition, 16 orientations at the coarsest level) with the different LSAIs proposed in Section 3.4.3 to denoise some 512x512 gray-scale images contaminated with AWG noise with standard deviations 5, 10, 20, 30 and 50. The PSNR results are shown in Table 3.8. They have been averaged over 10 noisy versions of each image, and in the last column, the standard deviation of these results is indicated for each LSAI.
From these results it can be noticed that for an anisotropic LSAI the orientation is important. Except for the case where $\sigma = 5$, the denoising result is better when the anisotropic LSAI is oriented in the direction of minimal correlation compared to when the LSAI is oriented in the direction of maximal correlation.

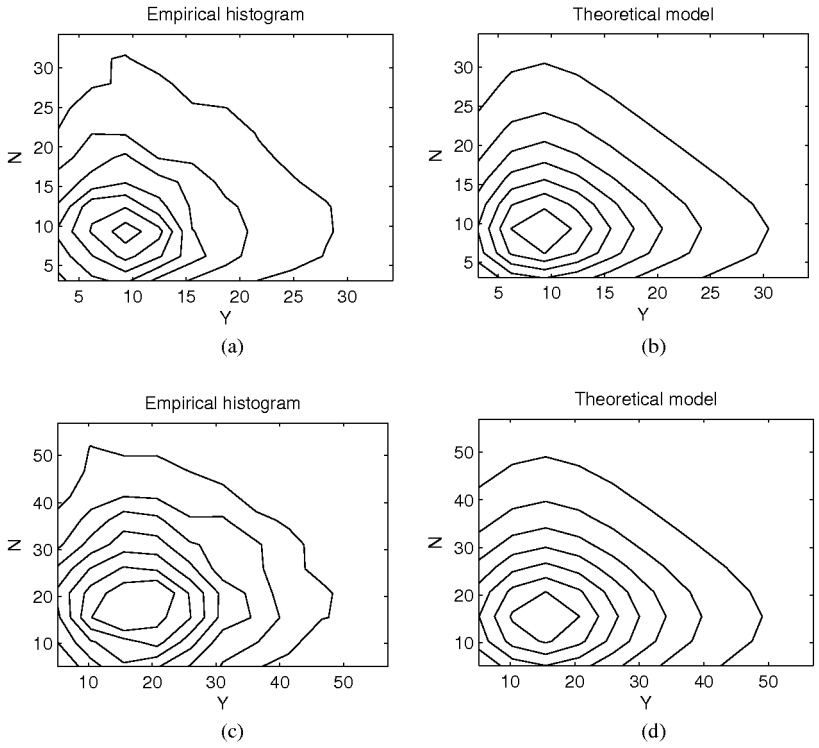


Figure 3.5: Joint histogram of a significant coefficient and its significant neighbors within a 1×5 spatial neighborhood δ, oriented in the direction of minimal correlation, for a sub-band of the next-to-highest scale of (a) *Barbara*, contaminated with AWGN with $\sigma = 10$, (c) *Baboon*, contaminated with AWGN with $\sigma = 20$. (b) and (d) Approximation by a conditionally independent model $p(|y||H^1) * p(|y||H^1)$.

LSAI	σ	PSNR (dB)				σ_{PSNR}
		Lena	Barb	Pepp	Bab	
a) An. LSAI max corr	5	38.61	37.16	36.56	34.69	0.013
b) An. LSAI min corr		38.53	37.14	36.45	34.59	0.015
c) AOP LSAI		**38.80**	**37.27**	**36.70**	**34.92**	0.013
d) IIB LSAI		38.64	37.16	36.37	34.63	0.014
a) An. LSAI max corr	10	35.36	33.62	33.97	30.41	0.023
b) An. LSAI min corr		35.65	33.84	34.12	30.38	0.023
c) AOP LSAI		35.71	33.71	34.20	**30.59**	0.022
d) IIB LSAI		**35.85**	**33.89**	**34.22**	30.48	0.022
a) An. LSAI max corr	20	31.54	29.90	30.94	26.58	0.026
b) An. LSAI min corr		32.24	30.36	31.34	26.60	0.026
c) AOP LSAI		32.34	30.15	31.49	**26.82**	0.025
d) IIB LSAI		**32.57**	**30.45**	**31.63**	26.76	0.018
a) An. LSAI max corr	30	29.08	27.68	28.69	24.61	0.046
b) An. LSAI min corr		30.08	28.29	29.37	24.73	0.044
c) AOP LSAI		30.18	28.02	29.55	**24.92**	0.043
d) IIB LSAI		**30.44**	**28.37**	**29.73**	24.87	0.043
a) An. LSAI max corr	50	26.17	24.79	25.62	22.39	0.059
b) An. LSAI min corr		27.38	**25.49**	26.53	22.66	0.056
c) AOP LSAI		27.45	25.02	26.69	**22.78**	0.053
d) IIB LSAI		**27.70**	25.45	**26.81**	22.75	0.054

Table 3.8: *ProbShrinkCurv* denoising results of some 512x512 gray-scale images, using the different LSAIs of Section 3.4.3. The noisy input images are contaminated with AWG noise with different standard deviations σ. This table shows the denoising results in terms of PSNR using (a) a 1x5 anisotropic LSAI, oriented in the direction of maximal correlation, (b) a 1x5 anisotropic LSAI, oriented in the direction of minimal correlation, (c) an adjacent, opposing and parent LSAI and (d) a combined intra- and inter-band (IIB) LSAI. The last column shows the estimated standard deviation of the results for each LSAI.

The difference in terms of PSNR depends on the image and increases with increasing standard deviation of the added noise. The explanation for this trend is that additive white Gaussian noise is transformed into correlated noise by the curvelet transform [Starck et al., 2002]. When calculating the LSAI of a coefficient in a neighborhood that coincides with the direction of this correlation, it will be contaminated by the same noise that disturbed the coefficient and thus will not be a good indicator of the local spatial activity, even though the significant coefficients are highly correlated along this direction (see Table 3.5a). This is increasingly so for higher noise levels (cfr. the rising trend of the correlation between insignificant coefficients and neighbors in the direction of highest correlation in Figure 3.4). Because significant coefficients are still somewhat correlated along the direction of lowest correlation (see Table 3.5b), calculating the LSAI in a neighborhood oriented in this direction will lead to a better denoising performance. For very small noise standard deviations, e.g., for $\sigma = 5$, the neighboring coefficients in the direction of highest correlation are a better indicator of the local spatial activity than the neighbors in the perpendicular direction because the disturbing influence of the noise is small. For such small noise levels, calculating the LSAI in a neighborhood oriented in this direction leads to better denoising results.

Table 3.8 further shows that the AOP LSAI outperforms the intra-band LSAI in the direction of lowest correlation for all the tested images except for *Barbara*. This result confirms our observations of Section 3.4.3. Indeed, the correlation of the intra-band LSAI in the direction of lowest correlation with significant coefficients is smaller than in the AOP LSAI case (see Table 3.5b and c and Table 3.6b and c). *Barbara* is the exception here, because the correlation of the significant coefficients with the intra-band LSAI in the direction of lowest correlation is extremely high, much higher than the correlation with the AOP LSAI, and much higher than for the other tested images. The importance of this intra-band correlation for *Barbara* explains why the LSAI exploiting this correlation performs better in the denoising method.

For standard deviations bigger than 5, the denoising performance of the combined intra- and inter-band LSAI is superior to that of the AOP LSAI. This is somewhat surprising, as we have not observed a greater average correlation between this LSAI and the significant coefficients for the images of our test set (see Section 3.4.3). A possible explanation is that the intra- and inter-band LSAIs contribute complementary information to the denoiser. The LSAI contributes to the determination of the level of 'significance' of each coefficient to be denoised. When adding intra- to inter-band information, individual coefficients that correlate well with the inter-band LSAI but not with the intra-band one will be judged as 'less significant', but others will behave in the opposite way. In other words, the total fraction of significance over all the coefficients in the sub-bands is not increased, but spread over more coefficients. This leads to a better denoising result for the tested images (except for a small deterioration for *Baboon*). For $\sigma = 5$, the information contributed by the uncorrelated intra-band LSAI to the combined intra- and inter-band LSAI deteriorates the

denoising performance. We had already observed that the uncorrelated intra-band LSAI performs poorly for such low noise levels.
Because of the superior denoising performance of the intra-inter-band LSAI for all noise levels except for very low ones we will choose it in our *ProbShrinkCurv*-method to compare it with state-of-the-art denoising methods (see Section 3.7).
The execution times of the denoising methods using the different LSAIs have also been measured. Experiments were performed on an AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set. The performed computations were floating-point computations. In Matlab code, with the computation intensive parts implemented in c, averaged over 10 experiments, denoising a 512×512 image with the anisotropic LSAI oriented in the direction of maximal and minimal correlation took 3.43 s, respectively 3.57 s to execute. For the AOP inter-band LSAI we measured 3.13 s and for the combined intra- and inter-band LSAI 8.87 s. Note that the denoising methods using LSAIs of similar size exhibit similar execution times. The intra- and inter-band LSAI combines the coefficients of the inter-band and an intra-band LSAI and therefore has a slower running time.

3.6 Choice of the Threshold T

A crucial issue that has not been addressed up to this point is the choice of the threshold T. This threshold determines what our signal of interest actually is. This signal of interest should be chosen such as to minimize the MSE of the denoised image. An analytical derivation seems intractable for the assumed prior. To make this choice nonetheless in a theoretically founded way, we follow the approach of Jansen et al. and Pižurica et al. in [Jansen and Bultheel, 2001; Pižurica and Philips, 2006] and base ourselves on *oracle* thresholding [Mallat, 1998] (see below). Oracle thresholding provides us with the MMSE estimate of transform coefficients corrupted with AWG noise by zeroing the ones with noise-free component below the standard deviation σ' of the noise in the sub-bands. Thus, $T = \sigma'$ marks the boundary between significant and insignificant coefficients.
Direct application of this estimator to our denoising method is unrealistic, and for several reasons. Firstly, this approach requires an *oracle* to inform us of the value of the noise-free coefficient in order to make our classification decision about the noise-contaminated coefficient. As the noise-free coefficients are not known to us during denoising because they are what we wish to estimate from the noise-contaminated coefficients, such an oracle is not available in a realistic scenario. Secondly, this choice minimizes the MSE when denoising is achieved by hard thresholding the noisy coefficients. Our denoiser soft shrinks rather than thresholds the coefficients. Finally, the noisy curvelet coefficients that we consider are not contaminated with white but with colored noise. Considering all these factors, we expect the optimal value of the threshold T not to coincide exactly with the standard deviation of the contaminating noise in the sub-bands but to peak in its vicinity.

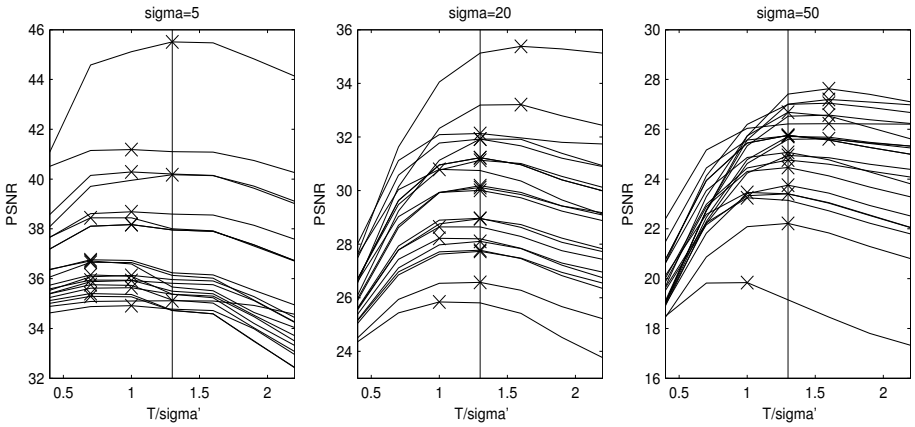


Figure 3.6: Denoising performance of the *ProbShrinkCurv* method as a function of the ratio threshold to sub-band noise standard deviation T/σ' for several images of the USC-SIPI image test set and for several image noise levels (from left to right $\sigma = 5, 20$ and 50). Recall that for the curvelet transform the noise standard deviation in the sub-bands σ' is $\sigma/\sqrt{\alpha}$, with α the redundancy factor of the transform. The optimal choice of the ratio T/σ' for each image is marked by a cross. The solid vertical line marks $T = 1.3\sigma'$.

The standard deviation of the noise is often not known to the denoising technique in practical situations, but can be estimated from the corrupted data, e.g., using the MAD estimator of Donoho et al. [Donoho and Johnstone, 1994]. The influence of an inaccurate estimate of the standard deviation of the noise on the denoising performance will be discussed in Section 3.7.3. In subsequent experiments, we assume that the standard deviation of the noise is known.

To verify the expectation of the optimal value of T being proportional to the standard deviation of the contaminating noise in the sub-bands, we investigate in Figure 3.6 the influence of the threshold T on the denoising performance of *ProbShrinkCurv* for several noise levels and for the images of the *Miscellaneous* volume of the USC-SIPI image test set (images used in Section 3.7 were removed from the test set to avoid overfitting of T). For each image we calculated the results for 7 different values of T, each time averaged over 10 noisy versions, and this for image noise standard deviations $\sigma = 5$, 20 and 50 (recall that for the curvelet transform the corresponding noise standard deviation in the sub-bands σ' is $\sigma/\sqrt{\alpha}$, with α the redundancy factor of the transform). In Figure 3.6 the resulting curves are plotted for half of the images (not for all to avoid overloading the graphs). The optimal choice of the ratio T/σ' for each image is marked by a cross. From these figures it can be noted that the optimal value for T indeed always lies in the vicinity of $T = \sigma$ but also that it is image and noise level dependent. The overall trend is that at lower noise levels the best denoising performance is achieved for lower values of T/σ'.

σ	T/σ'					
	0.4	0.7	1.0	1.3	1.6	1.9
5	0.7591	0.0714	**0.0374**	0.3707	0.4502	1.1447
10	1.9985	0.6963	0.1253	**0.1089**	0.3646	1.0022
20	3.8806	1.4579	0.3252	**0.0754**	0.2777	0.7480
30	4.6897	2.2665	0.4825	**0.0761**	0.2404	0.6448
50	5.4500	2.4833	0.7104	**0.1028**	0.2328	0.5495

Table 3.9: Average denoising quality drop (in dB) over all the images in the USC-SIPI test set when fixing the ratio threshold T to sub-band noise standard deviation σ' to a particular value. Recall that for the curvelet transform the noise standard deviation in the sub-bands σ' is $\sigma/\sqrt{\alpha}$, with α the redundancy factor of the transform.

The average quality drop over all the images (except for the images used in Section 3.7) when fixing the ratio T/σ' to a particular value has been quantified in Table 3.9 for noise levels $\sigma = 5$, 10, 20, 30 and 50. Firstly, we can notice that at moderate noise levels, the average quality drop when choosing a value for T/σ' within 23% of the optimum does not exceed $0.5\,dB$. Secondly, we can observe that the trend of lower noise levels favoring a lower threshold and vice versa is confirmed. From $\sigma = 10$ onwards, however, the smallest overall drop in performance is achieved when keeping T/σ' constant, namely at 1.3. From these experiments, we have chosen $T = 1.3\sigma'$ throughout this chapter (also for the statistical analysis of Section 3.4).
In a specific set-up where a camera is always used in the same circumstances to capture similar images, it is possible to fine-tune the threshold T to the application. In such a controlled environment, a calibration of the camera system with regard to the introduced noise is also possible.

3.7 Results

In this section, we report on the denoising performance of our newly developed method and we provide a comparison with some state-of-the-art denoisers.

3.7.1 *ProbShrinkCurv* Denoising Results

Denoising results of the *ProbShrinkCurv* method on some 512×512 and 256×256 gray-scale images are reported in Table 3.10. When possible, we have used the versions of the images included with the online implementation of [Portilla et al., 2003]. In these experiments, the standard deviation of the AWGN was assumed known. Results have been averaged over 10 noise realizations for each image and for each noise level. The standard deviation of the results for each noise level are reported in the last column of Table 3.10. We used 4 scales in

$\sigma/PSNR$	lena	barbara	boats	baboon	house	σ_{PSNR}
	512×512				256×256	
2 / 42.03	42.38	42.58	42.35	42.00	43.49	0.014
5 / 34.13	37.86	37.16	36.19	34.63	38.26	0.015
10 / 28.13	35.20	33.86	33.12	30.48	34.84	0.033
15 / 24.61	33.38	31.83	31.23	28.15	32.92	0.049
20 / 22.13	32.02	30.38	29.93	26.76	31.50	0.032
25 / 20.23	30.97	29.23	28.88	25.63	30.41	0.042
30 / 18.70	30.01	28.29	28.01	24.87	29.45	0.057
35 / 17.44	29.21	27.43	27.23	24.15	28.59	0.123
50 / 14.61	27.14	25.32	25.35	22.75	26.60	0.102

Table 3.10: Denoising results in terms of PSNR (dB) of some 512x512 and 256x256 gray-scale images. The last column shows the estimated standard deviation of the results for each noise level.

the curvelet decomposition and 16 orientations at the coarsest level. Varying these parameters alters the denoising results. The optimal numbers are image-dependent, but we found that this choice produces satisfying results for a broad class of images.

There are several ways to use *ProbShrinkCurv* to denoise color images. A naive approach would be to denoise each color band separately using the proposed method. A better option is to extend the proposed method to exploit the correlation between color bands. In [Pižurica and Philips, 2006] it has been proposed to incorporate the correlation between color bands in the definition of the LSAI, i.e., to also include correlated coefficients from other color bands in the calculation of the LSAI. The application of the *ProbShrinkCurv* technique to the denoising of color images has not been studied in this thesis because this study is not expected to provide any different insights than the study of denoising gray-scale images.

3.7.2 Comparison With Other Denoisers

In Fig. 3.7 we compare the results of the newly developed method to some state-of-the-art denoisers, namely *BiShrink* using a dual tree complex wavelet decomposition [Sendur and Selesnick, 2002], *BLS-GSM* with the parameters set as in [Portilla et al., 2003] and operating on a full steerable pyramid decomposition of the image, the *BM3D* method as reported in [Dabov et al., 2007] and the *ProbShrink* method for wavelets in its redundant wavelet transform implementation [Pižurica and Philips, 2006]. A comparison to simple curvelet domain hard thresholding is also provided (4 scales in the curvelet decomposition, 16 orientations at the coarsest level and threshold at $k\sigma$ with $k = 4$ at the finest scale and 3 otherwise). Implementations of curvelet hard thresholding and of the methods of [Sendur and Selesnick, 2002], [Portilla et al., 2003], [Dabov et al., 2007] and [Pižurica and Philips, 2006] are publicly available and

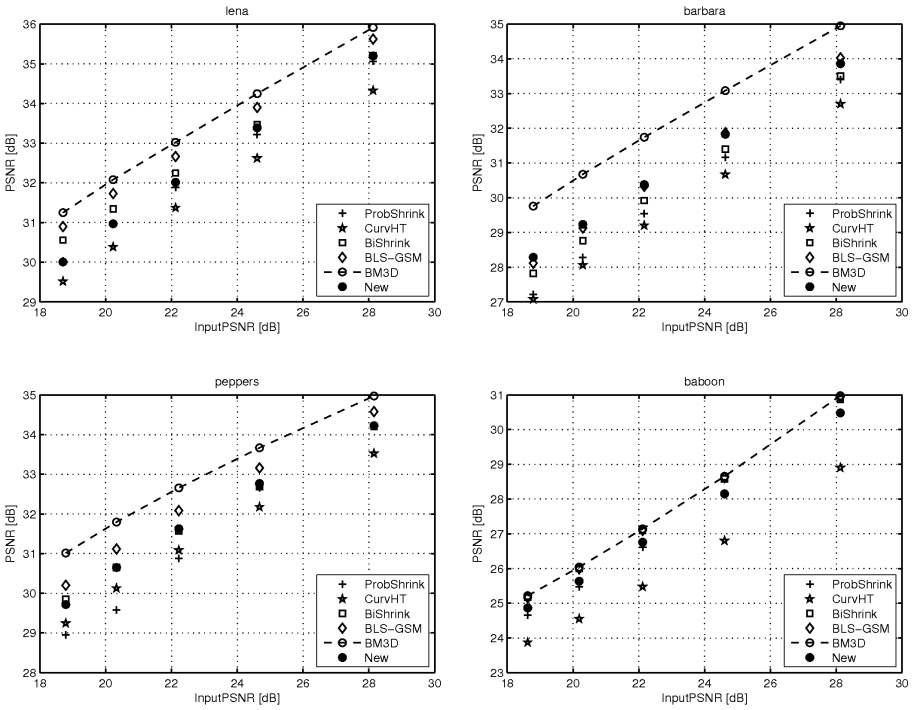


Figure 3.7: Output PSNR as a function of input PSNR for several 512×512 images for the following methods: *BiShrink* using a dual tree complex wavelet decomposition [Sendur and Selesnick, 2002], *BLS-GSM* with the parameters set as in [Portilla et al., 2003], the *BM3D* method as reported in [Dabov et al., 2007], the *ProbShrink* method for wavelets in its redundant wavelet transform implementation [Pižurica and Philips, 2006], curvelet hard thresholding and the proposed *ProbShrinkCurv* method.

were used to produce the results of Fig. 3.7. Results have been averaged over 10 noise realizations for each image and for each noise level. For all methods, we assume that the standard deviation of the noise is known to the denoising technique. Denoising results on images with unknown noise variance will be discussed in Section 3.7.3.

From these results, we can observe that the *ProbShrink* method adapted to curvelets outperforms or matches its wavelet-based counterpart for all images. Improvements are smallest for *Lena* and *Baboon* and biggest for *Barbara* and *Peppers* (up to 1.08 *dB*). Differences are more pronounced for big standard deviations of the AWGN. In fact, for small standard deviations, starting from $\sigma = 5$ and smaller, the *ProbShrink* method for wavelets performs better. These results were not included in Fig. 3.7 in order not to overload it. This trend complies with our observations from Sections 3.5.3 and 3.6, where we found that for small noise levels the AOP LSAI and a smaller threshold T would be

more appropriate than the intra-inter-band LSAI and the $T = 1.3\sigma'$ which we have chosen because of the other noise levels.
The improvements of *ProbShrinkCurv* over simple curvelet domain hard thresholding are considerable for all images at all noise levels, but they are most notable for *Barbara* and *Baboon*, i.e., for images that are not sparsely represented in the curvelet domain. For these images, the more complex Bayesian and neighborhood-adaptive approach of *ProbShrinkCurv* provides a clear advantage over simple hard thresholding.
The performance of *ProbShrinkCurv* in comparison with other state-of-the-art techniques is somewhat image and noise level dependent, but overall we can observe that our new denoiser is competitive with *BiShrink* (based on a transform of similar redundancy as *ProbShrinkCurv*, which is about 7.2 for our choice of parameters) but numerically outperformed by *BM3D* for all images and by *BLS-GSM* for all images except for *Barbara*, for which denoising results are similar.
In Fig. 3.8 and 3.9 we visually compare some cut-outs of the denoising results. This visual comparison reveals that the denoising results of *ProbShrinkCurv* generally give a much sharper impression than the results obtained through all other methods (*ProbShrink*, curvelet hard thresholding, *BiShrink*, *BLS-GSM* and *BM3D*). The good edge preserving qualities of *ProbShrinkCurv* are illustrated on the feathers in Lena's hat, which are sharply preserved in the denoising results of *ProbShrinkCurv* shown in Fig 3.9h and which are over-smoothed in the denoising results of *BLS-GSM* (Fig 3.9f) and *BM3D* (Fig 3.9g).
Compared to *ProbShrink* for wavelets and *BiShrink*, we see that the *ProbShrinkCurv* method is less plagued by impulse-like artifacts and artificial patterns (visible, e.g., in *Barbara*'s face, cfr. Figs. 3.8c, e and h). Some minor stripe-like artifacts are visible, but a lot less than in the curvelet hard thresholding case (compare *Barbara*'s mouth in Figs. 3.8d and h).
A further meaningful evaluation of the quality of the denoised images depends on the purpose of the denoising. If the denoising is performed for aesthetic purposes, extensive psycho-visual experiments would be required. If the denoising is the first step prior to other image processing steps (such as image segmentation or compression), the used quality metric should be chosen as a function of these subsequent processing steps.

3.7.3 Denoising With Unknown Noise Variance

If the standard deviation of the noise is not known to the denoising technique, as is often the case in practical situations, one has to estimate it from the corrupted data, e.g., using the MAD estimator of Donoho et al. [Donoho and Johnstone, 1994]. The inaccuracy of this estimate affects the denoising performance of the methods. In Table 3.11 we show the difference in PSNR performance between denoising with known and estimated noise variance, for several methods and noise levels, averaged over each time 10 noise realizations of the images *Lena*, *Barbara*, *Peppers* and *Baboon*. In the curvelet-based methods the noise variance is estimated from the last orientation sub-band at the finest scale and in the

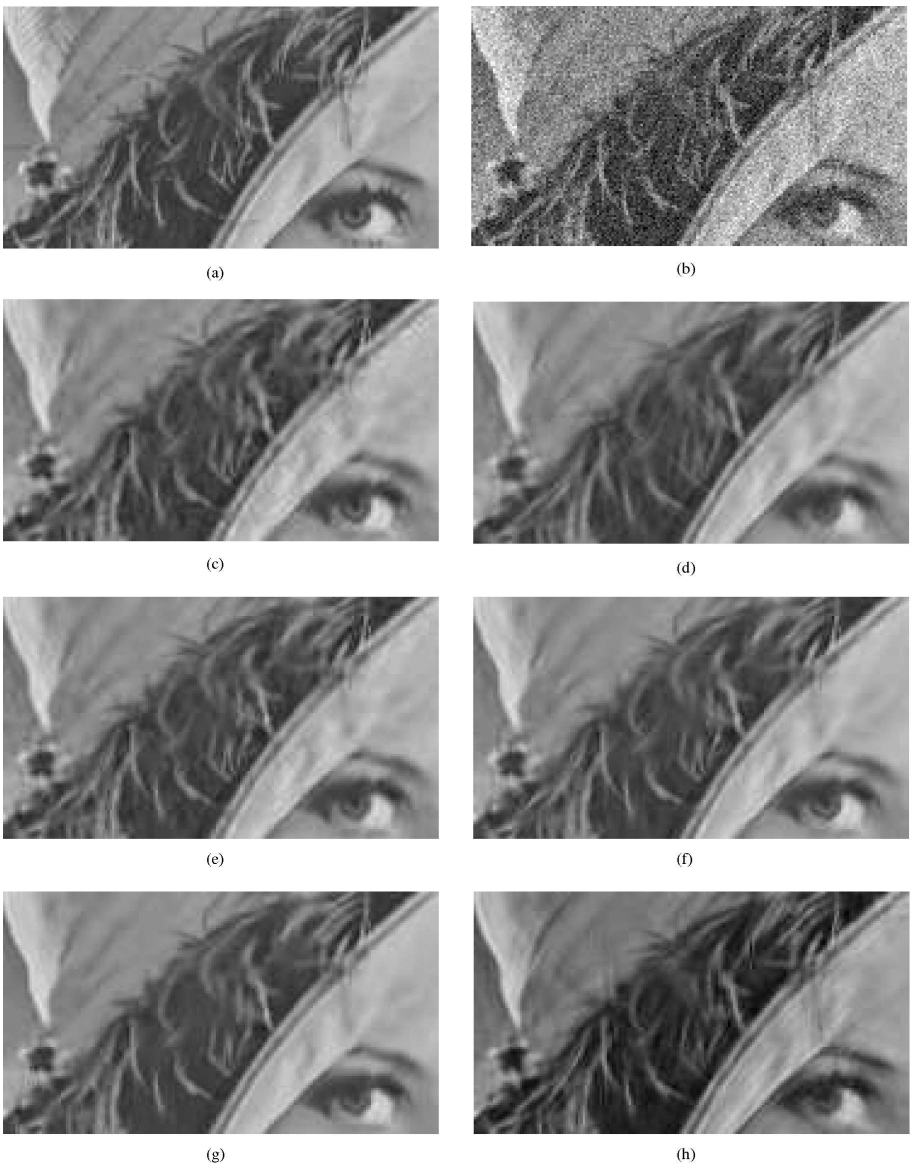


Figure 3.8: Detail of the denoising results of *Lena*. (a) the original image, (b) the noisy image (noise standard deviation 20), denoising result of (c) *ProbShrink*, (d) curvelet hard thresholding, (e) *BiShrink*, (f) *BLS-GSM*, (g) *BM3D* and (h) *ProbShrinkCurv*.

Figure 3.9: Detail of the denoising results of *Barbara*. (a) the original image, (b) the noisy image (noise standard deviation 20), denoising result of (c) *ProbShrink*, (d) curvelet hard thresholding, (e) *BiShrink*, (f) *BLS-GSM*, (g) *BM3D* and (h) *ProbShrinkCurv*.

Denoiser	σ =2	σ =10	σ =20	σ =30	σ =50
ProbShrink	4.28	0.57	0.12	0.05	-0.25
CurvHT	4.36	0.38	0.08	-0.01	-0.08
BiShrink	4.14	0.25	0.09	0.08	0.04
BLSGSM	4.99	0.75	0.20	0.13	-0.39
BM3D	4.43	0.37	0.11	0.06	-0.01
ProbShrinkCurv	4.41	0.24	0.06	0.04	0.11

Table 3.11: PSNR difference (in *dB*) between denoising with known and with estimated noise standard deviation σ, averaged over the images *Lena*, *Barbara*, *Peppers* and *Baboon*.

BiShrink method from the first finest scale sub-band in the dual tree complex wavelet decomposition (as it is implemented in the *BiShrink* code available online). In the other denoisers, no noise estimation is implemented and we therefore estimate the noise variance from the diagonal detail coefficients at the finest scale of an undecimated wavelet decomposition using the Haar wavelet. From this table it is obvious that the denoising performance of the compared methods drops very dramatically for low noise levels but becomes more robust to inaccuracies in noise variance estimation at higher noise levels. We conclude that the performance differences are of the same order of magnitude for all methods compared here.

3.7.4 Execution Times

In Table 3.12 we compare the mean execution times of *BiShrink*, *BLS-GSM*, *BM3D*, *ProbShrink*, curvelet hard thresholding and *ProbShrinkCurv* when denoising a 512 × 512 gray-scale image. Results have been averaged over 10 experiments on a AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set. The performed computations were floating-point computations. All algorithms run in Matlab with the computation intensive parts implemented in C. We can see that the mean execution time of *ProbShrinkCurv* is lower than the mean execution time of the methods with the best denoising performance (*BLS-GSM* and *BM3D*), and that it is also lower than that of its wavelet-based counterpart. The method is about half as fast as curvelet hard thresholding.

3.8 Fusing Denoised Images

We now use the curvelet-based *ProbShrinkCurv* denoiser developed in the previous sections to improve the fusion result of stacks which are contaminated with noise.
To this end, we denoise the noisy slices with the *ProbShrinkCurv* method before deciding for each curvelet coefficient from which slice it should be selected

Denoising method	Mean execution time
BiShrink [Sendur and Selesnick, 2002]	2.14 s
BLS-GSM [Portilla et al., 2003]	75.19 s
BM3D [Dabov et al., 2007]	11.98 s
ProbShrink [Pižurica and Philips, 2006]	10.52 s
Curvelet hard thresholding	4.93 s
ProbShrinkCurv	8.87 s

Table 3.12: Mean execution times when denoising a 512×512 gray-scale image. Results are averaged over 10 experiments on a AMD Athlon(TM) 64 3400+ 2.40 GHz processor.

using the method described in Chapter 2. To separate the effect of the denoising on the fusion process from its effect on the PSNR of the input slices, we use this 'selection map' to fuse the noise-free slices, not the denoised ones. Mathematically, this can be formulated as follows. Let $C_{i,j,z}(x,y)$ denote the noise-free curvelet coefficient at scale i, orientation j and spatial coordinates x and y, of the slice with index z. Let $C^d_{i,j,z}(x,y)$ be the corresponding denoised curvelet coefficient, obtained by denoising the noisy coefficient $C^n_{i,j,z}(x,y)$ with the *ProbShrinkCurv* denoiser. Coefficients of the fused image are then selected as:

$$F_{i,j}(x,y) = C_{i,j,\mathrm{argmax}_z(|C^d_{i,j,z}(x,y)|)}(x,y). \tag{3.14}$$

In a real application these are of course not available, and this method is used here only for evaluation purposes.

Because denoising and fusion take place in the same transform domain, both operations can be easily combined and only one forward and one inverse curvelet transformation are necessary. This is advantageous from a computational point of view. Of course the denoising can also be performed with any other of the many denoisers described in literature.

In subsequent experiments, we start from the noisy input stacks that were used in the experiments of Section 3.1. The slices in these stacks are contaminated with AWG noise with $\sigma = 10$. Different values of σ would lead us to similar conclusions. In Table 3.13 the average PSNR of the noise-contaminated slices in each stack are listed, as well as the average PSNR of the denoised slices. The same settings for the denoising algorithm as in Section 3.7 were used to obtain these denoising results. As the fusion algorithm operates on gray-scale images (see Section 2.4.6), color images were converted to gray scale prior to denoising using the technique described in Section 2.4.6. The standard deviation of the noise was assumed unknown and was estimated from the data using the MAD estimator of Donoho et al. [Donoho and Johnstone, 1994]. As expected, the denoising method works particularly well for the images *Eggs* and *Algae*, which are piece-wise smooth and therefore have a sparse representation

	Average over Slices		Denoised Input Stack	
	Noisy	Denoised	No checks	All checks
D18	28.13	30.55	26.80	34.26
D22	28.13	29.11	28.63	29.52
D23	28.14	31.03	28.10	34.69
D112	28.12	29.57	28.33	33.26
Eggs	28.12	37.82	41.97	41.33
Algae	28.12	38.37	38.54	38.74
Clouds	28.12	38.74	39.22	39.00
Leaves	28.31	28.83	38.48	37.93
Average Gain over Noisy Input Stack:			3.59	2.20

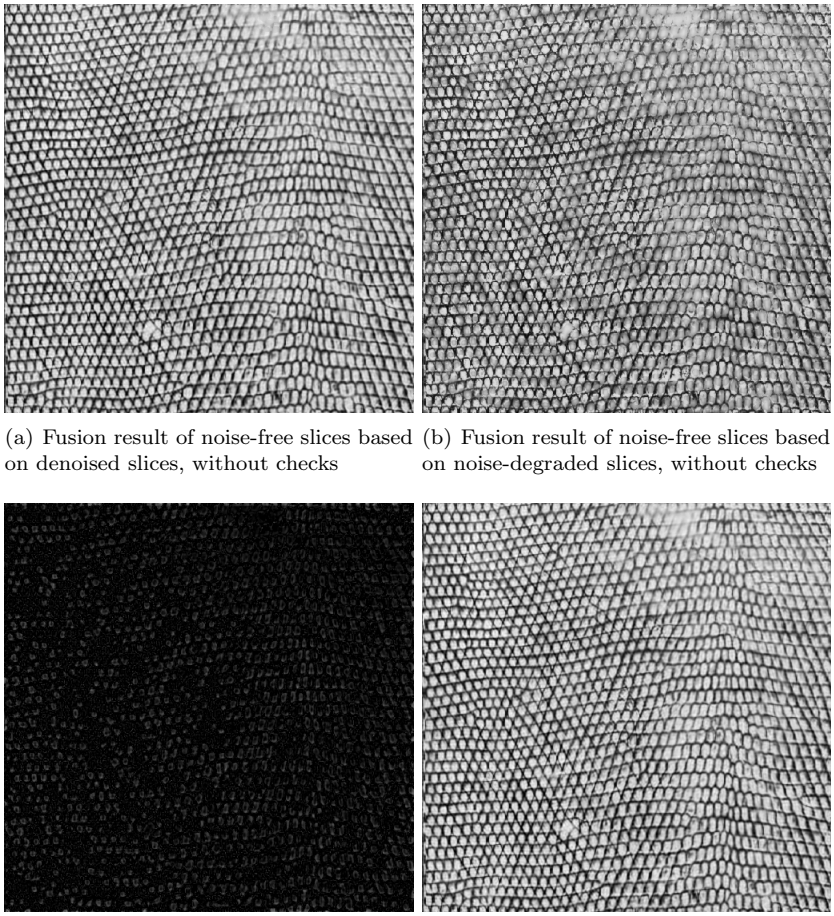
Table 3.13: Average PSNR (dB) of the slices in the stack contaminated with additive white Gaussian noise of $\sigma = 10$ and of the denoising result of these noisy slices. Also the result in terms of PSNR of fusing the noise-free slices based on a selection map obtained from these denoised slices. On the bottom line the average gain in PSNR of fusion based on denoised slices over fusion based on noisy slices (results of Table 3.1) is indicated.

in the curvelet-domain. For the other images, which are all heavily textured, the PSNR increase is smaller but still important.

Fig. 3.10a shows for the *D22* Brodatz texture the visual fusion result using the selection rule of Eq. 3.14. No spatial smoothness or sub-band consistency checks were performed. To facilitate comparison with the fusion result of noise-free slices based on noise-degraded slices, Fig. 3.1c is repeated in Fig. 3.10b and the absolute difference image between Fig. 3.10a and Fig. 3.10b is shown in Fig. 3.10c. A comparison between Fig. 3.10a and Fig. 3.10b reveals that thanks to the denoising the prevalent artificial structures caused by disturbances in the fusion process due to noise do not appear in the result image. Fig. 3.10d shows the fusion result of noise-free slices based on denoised input slices but now with smoothness and consistency constraints imposed. It is hard to visually notice a difference between Figs. 3.10a and d.

The numerical results of this fusion process for the stacks in our test set are listed in the last two columns of Table 3.13, for fusion without checks and with smoothness and spatial checks. Comparing the fusion results without and with checks in the Tables 3.13 and 3.1, we observe an increase in PSNR for nearly all stacks when denoising prior to fusion. This increase amounts to up to $8.99\,dB$ for *D22* for fusion without checks. For *Clouds* a small deterioration can be observed, which can be explained by the abruptness of the transitions between blurred and in-focus image regions in the last four stacks of our test set (see Section 2.5.1). The average gain in PSNR of fusion based on denoised slices over fusion based on noisy slices is listed on the bottom line of Table 3.13. The gain amounts to several dB and is largest when no checks are performed.

The best fusion results are obtained by combining denoising prior to fusion with fusion with all checks.



(a) Fusion result of noise-free slices based on denoised slices, without checks

(b) Fusion result of noise-free slices based on noise-degraded slices, without checks

(c) Absolute difference between fusion result based on denoised and noise-degraded slices

(d) Fusion result of noise-free slices based on denoised slices, with checks

Figure 3.10: For the *D22* texture, the fused image obtained by fusing noise-free images without checks, (a) based on denoised and (b) based on slices contaminated with additive white Gaussian noise with a standard deviation of $\sigma = 10$. (c) The absolute difference image between (a) and (b). (d) The fused image obtained by fusing noise-free slices with checks based on denoised slices.

3.9 Conclusion

In this chapter we have ascertained that noise has a disturbing influence on image fusion for depth of field extension. We have shown that imposing spatial smoothness and sub-band consistency constraints has a regularizing effect and improves the fusion quality. We have presented denoising of the slices in the curvelet domain prior to fusion as an alternative solution.
In order to develop a curvelet-based denoiser, we have investigated the differences in statistical behavior between curvelet coefficients containing a significant noise-free component and those in which no signal of interest is present. We have then discussed the adaptation of the *ProbShrink* denoising method for wavelets [Pižurica and Philips, 2006] to curvelets, resulting in a method which we have called *ProbShrinkCurv*. In particular, we have put the knowledge gained from our statistical study to use in the design of an appropriate local spatial activity indicator (LSAI) for this new method.
When considering intra-band coefficients for the LSAI, we have found that, although curvelet coefficients are more correlated along the principal direction of their generating basis function, neighboring coefficients in the perpendicular direction are a better indicator of the significance of the reference coefficient in terms of denoising results. We have further ascertained that it is beneficial to also incorporate coefficients from adjacent, opposing and parent sub-bands in the LSAI.
The resulting denoising method, *ProbShrinkCurv*, outperforms its wavelet-based counterpart and produces results that are both visually competitive with and numerically close to those of state-of-the-art denoisers.
Using *ProbShrinkCurv* to denoise the curvelet coefficients of the noise-contaminated slices prior to fusion improves the fusion result considerably. The average gain over our test set amounts to 3.59 dB when no checks are performed and 2.20 dB when smoothness and consistency are imposed. The best fusion results are obtained when denoising prior to fusion is combined with fusion with all checks.

4

Data Fusion for Occupancy Reasoning

In many applications, the deployment of a network of cameras with overlapping fields of view provides substantial advantages over a single fixed viewpoint camera. Images from the same event or subject can be gathered from different perspectives. When processed collaboratively, this extra data can provide interesting additional information. E.g., in scene monitoring, camera networks can alleviate occlusion problems; in gesture recognition, cues coming from different viewpoints can lead to a more robust decision; in free viewpoint television, the quality of the rendered intermediate views benefits from a larger number of cameras.

A crucial issue to fully exploit these extra possibilities is how to fuse the information of different cameras opportunistically. Due to unsuited viewpoints and/or the information loss inherent to the projection of a 3D scene on a 2D camera image, the observations of each camera alone can be inconclusive. A useful data fusion scheme for vision networks should be able to exploit agreement among cameras: the uncertainty about an observation should drop as more cameras corroborate each other's output.

In this chapter we present a novel method for calculating occupancy maps with a network of calibrated and synchronized cameras. In particular, we propose Dempster-Shafer based fusion of the ground occupancies computed from each view. The method yields very accurate occupancy detection results and in terms of concentration of the occupancy evidence around ground truth person positions it outperforms the state-of-the-art probabilistic occupancy map method and fusion by summing.

The recent introduction of 'smart cameras' with on-board image processing and communication hardware offers interesting possibilities for a distributed implementation of the proposed method. However, to be applicable in practical smart camera networks, a method has to deal with computational, latency and bandwidth constraints. Therefore we adapt the proposed occupancy calculation method in several ways. We modify it into a low data rate version such that wireless communication with the cameras becomes possible. Moreover we

drastically simplify the processing such that an implementation in hardware becomes more straightforward.
The work presented in this chapter has been performed in collaboration with my colleague Marleen Morbee and therefore some of the concepts presented here also appear in her PhD thesis. However, in Marleen's thesis the emphasis lies on an efficient calculation and usage of scan-lines. This possibility is only briefly introduced in this dissertation (in Section 4.5.2). In this thesis the aspect of data fusion is treated more elaborately. I.e., different alternatives for fusing the ground occupancy data are proposed and studied (cfr. Section 4.4, Section 4.5.1 and Section 4.6).
The remainder of this chapter is organized as follows. We start with a general introduction on occupancy maps and we elaborate on the data fusion aspect in Section 4.1. An introduction to the used Dempster-Shafer theory of evidence is presented in Section 4.2. The problem of occupancy map calculation is defined more precisely in Section 4.3, after which the proposed method is explained in Section 4.4. In Section 4.5 we study adaptations of this method. Results are discussed in Section 4.6 and we end with a conclusion in Section 4.7.

4.1 Occupancy Maps and Data Fusion

An occupancy map provides a top view of a scene and indicates which parts are occupied by people or objects. Such maps are important in many applications such as surveillance, smart rooms, video conferencing and sport game analysis. Camera networks offer an attractive non-intrusive and flexible tool for this purpose. They do not require people to wear dedicated gear, nor the environment to be equipped with special sensors other than cameras, which are often part of the existing infrastructure, especially in security applications.
In recent years, foreground silhouettes in multiple camera views have been increasingly used to estimate the probability of ground occupancy. Two basic approaches exist. Bottom-up methods transfer the information in the different camera images to a common reference plane using camera image-floor homographies [Delannay et al., 2009]. Top-down approaches extract occupied ground positions by comparing a generative model of the objects in the scene with the actual foreground silhouettes observed in the camera views [Alahi et al., 2009; Fleuret et al., 2008].
For both approaches the mathematical laws for the fusion of data from different cameras had not been considered explicitly before the work presented in this chapter was published [Morbee et al., 2010a]. In the following we focus on this data fusion aspect within a bottom-up method and show that Dempster-Shafer based fusion of camera information leads to significantly more accurate occupancy maps. For the basket ball dataset of De Vleeschouwer and Delannay [2009], the total mass of occupancy evidence is 1.12 to 10.34 times more concentrated around the ground truth player positions than for the methods of Delannay et al. [2009] and Fleuret et al. [2008], as will be discussed in Section 4.6.

In the probabilistic occupancy map (POM) method of Fleuret et al. [2008], for each view the conditional distribution of the observed background subtraction image given the true object positions is a function of a distance measure between the background subtraction image and the image obtained from a generative model. Information from different views is fused by multiplying these conditional distributions. This strategy is problematic in the typical case of imperfect foreground detection: a badly detected foreground region in even a single view can easily result in a missed occupancy detection.
In [Delannay et al., 2009], each camera produces a confidence value for the occupancy of each ground position $\mathbf{x}$ by back-projecting the foreground silhouettes to a common reference plane using camera image-floor homographies. The aggregated ground occupancy map is obtained by summing the camera confidences and by normalizing by the number of cameras that actually view $\mathbf{x}$.
In this work, unlike the summing [Delannay et al., 2009] and POM [Fleuret et al., 2008] fusion strategy, we use Dempster-Shafer (DS) based fusion to exploit the fact that if a hypothesis of (non-)occupancy is corroborated by different cameras, a higher belief should be assigned to it. Moreover, the DS theory of evidence allows to distinguish between equal probability of occupancy and non-occupancy, and lack of knowledge, e.g., when an object is (partially) outside a camera viewing frustum.
In the next section we introduce the main concepts of DS theory. Afterwards we describe the proposed occupancy calculation method in detail.

4.2 Dempster-Shafer Theory of Evidence

The DS theory of evidence provides a theoretical basis to combine evidence from different sources to arrive at a degree of belief in a number of propositions. Formally, an exhaustive set of mutually exclusive propositions constitutes a frame of discernment Ω. The subsets A of Ω are called propositions, the singleton subsets ω of Ω are elementary propositions and the power set, denoted as 2^Ω, is the set of all possible subsets A of Ω. A basic belief assignment (BBA) is a mapping m from 2^Ω to $[0,1] \subset \mathbb{R}$ such that $\sum_{A\subseteq\Omega} m(A) = 1$ and $m(\emptyset) = 0$. $m(A)$ expresses how much an agent believes in proposition A alone, with no further assumption about any proper subset of A. A particular instance of a BBA is called a body of evidence. The basic probability allotted to Ω is a measure of the belief that has not been assigned to any of the proper subsets of Ω. It can be interpreted as the remaining uncertainty about the propositions. Complete ignorance is represented by $m(\Omega) = 1$.
Consider as an example a camera network observing a car. The car can be either an orange Toyota, a red Honda or a blue Honda. The frame of discernment Ω in this case contains the mutually exclusive and exhaustive hypotheses {orange Toyota}, {red Honda} and {blue Honda}. There is also an aggregated hypothesis {Honda}. A first camera analyzes the color histograms of the cars as they appear in the image. Because the lighting conditions are unknown,

Table 4.1: Example bodies of evidence in a network of two cameras observing a car.

m	Ω	{orange Toyota}	{Honda}	{red Honda}	{blue Honda}
m_1	0.2	0.3	0.0	0.5	0.0
m_2	0.5	0.2	0.3	0.0	0.0
$m_{1,2}$	0.12	0.31	0.07	0.49	0.0
$m_{1\wedge 2}$	0.14	0.21	0.08	0.56	0.0

these observations do not suffice to conclude with certainty which is the color of the car, but there are strong indications that it is red. The resulting body of evidence m_1 could therefore be the one indicated on the first line of Table 4.1. Note that some basic belief (i.e., 0.2) is not assigned to any of the hypotheses to account for uncertainty in the measurements. This can arise for example when the color information of the car is degraded by specular reflections.
Suppose a second camera is not a color camera. In the captured gray-scale image, the evidence for the brand of the car is gathered by comparing the observed shape of the car with shapes in a database. Clearly, with this evidence gathering mechanism no direct evidence can be obtained for the hypotheses {red Honda} and {blue Honda}. We can however obtain evidence for the aggregated hypothesis {Honda}, and for the hypothesis {orange Toyota}. Assume the car is half occluded. The remaining shape is compared with the shapes stored in the database, and the resulting matching scores could for example give rise to the body of evidence m_2, listed on the second line of Table 4.1.
Assume two pieces of evidence give rise to two bodies of evidence m_1 and m_2. These provide different assessments for the propositions in the same frame of discernment. To aggregate the information from these two sources, we need a rule of combination.
The best known and most common combination rule is Dempster's rule of combination:

$$m_1 \oplus m_2(C) = \begin{cases} \sum_{A,B|A\cap B=C} \frac{m_1(A)m_2(B)}{1-K} & \text{if } C \neq \emptyset \\ 0 & \text{if } C = \emptyset. \end{cases} \tag{4.1}$$

where $C \subseteq \Omega$ and K is the amount of conflict between the two bodies of evidence, measured by

$$K = \sum_{A,B|A\cap B=\emptyset} m_1(A)m_2(B). \tag{4.2}$$

The denominator in Eq. 4.1 is a normalizing factor. It has the effect that conflict is completely ignored. This rule considers that the different evidence sources are reliable, i.e., that their output is correct. It leads to a specialization of the basic belief: each time a new piece of information is accepted, the basic belief assigned to a proposition A is redistributed over the subsets of A [Denoeux, 2008].

In our car example, the fusion of the two bodies of evidence using Eq. 4.1 leads to the fused body of evidence $m_{1,2}$, listed in Table 4.1.
When comparing m_1 to $m_{1,2}$, we note that the basic belief assigned to Ω has been redistributed over its subsets. The same holds for m_2, where additionally a large part of the basic belief assigned to the aggregated hypothesis {Honda} has been shifted towards the more specific hypothesis {red Honda}.
Dempster's rule assumes that m_1 and m_2 are distinct, i.e., that the sources that produced the evidence are independent. In [Denoeux, 2008] a cautious conjunctive rule is proposed to combine bodies of evidence that are not distinct. To present it here, we must introduce two more definitions. The commonality function and the conjunctive weight function associated with a BBA m are defined as, respectively

$$\forall A \subseteq \Omega : q(A) = \sum_{B|B \supseteq A} m(B); \tag{4.3}$$

$$\forall A \subset \Omega : w(A) = \prod_{B|B \supseteq A} q(B)^{(-1)^{|B|-|A|+1}} \tag{4.4}$$

Now let m_1 and m_2 be two non-dogmatic bodies of evidence (which means $m_1(\Omega) \neq 0$ and $m_2(\Omega) \neq 0$). Their combination with the cautious conjunctive fusion rule is denoted as $m_1 \circledcirc m_2$ and is defined as the body of evidence with weight function $w_1(A) \wedge w_2(A), \forall A \subset \Omega$, where $\wedge$ denotes the minimum operator. The resulting body of evidence $m_{1\wedge 2}$ for our car example is listed on the last line of Table 4.1.
This rule is derived from the principle of least commitment: of all bodies of evidence that could result from the combination of the inputs m_1 and m_2, the least informative one is chosen (see [Denoeux, 2008] for a discussion on how to compare the information content of two bodies of evidence). Note that as a consequence, if the bodies of evidence *are* distinct and they are combined using the cautious rule, the result will be less informative than if Dempster's rule is used.

4.3 Problem Formulation

Consider a network of N cameras and let the ground plane of the observed scene be discretized in resolution cells $\mathbf{x}$. We wish to assign a real value to each cell that expresses our confidence that the cell is occupied by a foreground object. Foreground objects are objects of interest, i.e., objects that are not part of the background of the scene. In typical applications such objects are persons, cars, luggage, etc.
The choice of the discretization resolution should depend on the resolution of the cameras. Because cameras have a limited resolution, different points on the ground plane can be projected on the same image pixel. Suppose we group all ground plane points that get projected on the same pixel in one resolution cell. The size of these cells varies with the distance between the cell and the

camera. Indeed, a translation of one pixel in the image corresponds to a certain translation on the ground plane. The size of the translation on the ground plane is large for ground plane regions far away from the camera and small for close-by regions. Thus, the resolution cells close to the camera will be small, and those far away will be large.

Ground plane regions close to one camera can be far away from another camera. For combining occupancy information from several cameras, a discretization in camera-dependent resolution cells is not very practical because information from cells with different sizes would have to be combined. In this work we therefore opt for a regular grid of fixed-size resolution cells in the ground plane. Alternatively, polygonal resolution cells adapted to the geometry of the cameras could be used.

The highest resolution occupancy information is obtained when the size of the cells is chosen as the minimal cell size of all camera-dependent resolution cells, i.e., as the size of the cell closest to its corresponding camera. In many practical cases this leads to an occupancy map that is unworkably large. In this work the discretization resolution is therefore chosen as a compromise between resolution and size of the occupancy map.

4.4 Dempster-Shafer based Occupancy Calculation

In our method, for each ground position $\mathbf{x}$ the mutually exclusive and exhaustive hypotheses that $\mathbf{x}$ is either occupied ($\{occ_{\mathbf{x}}\}$) or not ($\{nocc_{\mathbf{x}}\}$) constitute the frame of discernment $\theta_{\mathbf{x}} = \{occ_{\mathbf{x}}, nocc_{\mathbf{x}}\}$. The information from each view i, $1 \leq i \leq N$, is considered a distinct piece of evidence and we denote the BBA representing this evidence by m_i. We now explain how we define the BBA in our method.

Let H be the typical height of a person and consider a rectangular cuboid $\mathbf{C}_{\mathbf{x}}$ with height H and cell $\mathbf{x}$ as base. We assume that both the internal and external calibration parameters of the cameras are known, which includes knowledge about the position of the ground plane. If the cuboid $\mathbf{C}_{\mathbf{x}}$ lies completely outside the viewing frustum of camera i, this camera cannot provide any information about the occupancy of $\mathbf{x}$. The BBA is then $m_i(\{occ_{\mathbf{x}}\}) = 0$, $m_i(\{nocc_{\mathbf{x}}\}) = 0$ and $m_i(\theta_{\mathbf{x}}) = 1$. Otherwise, the projection of this cuboid into camera view i defines an image region $R^i_{\mathbf{x}}$. An example of such a region is marked by the white line in Fig. 4.1. Fig. 4.2 illustrates the introduced notations.

We gather evidence about the (non-)occupancy of the cells by independently segmenting each view into background and foreground using any state-of-the-art foreground detection algorithm, and by determining in each region $R^i_{\mathbf{x}}$ the fraction of background pixels $b^i_{\mathbf{x}}$ and of foreground pixels $f^i_{\mathbf{x}}$. Of course $b^i_{\mathbf{x}} + f^i_{\mathbf{x}} = 1$. The evidence $m_i(\{nocc_{\mathbf{x}}\})$ of camera i for the hypothesis $\{nocc_{\mathbf{x}}\}$ is $b^i_{\mathbf{x}}$.

For $m_i(\{occ_{\mathbf{x}}\})$ the situation is more complicated: because of the limited resolution of the cameras, cuboids $\mathbf{C}_{\mathbf{x}}$ and $\mathbf{C}_{\mathbf{x}'}$ centered in different cells $\mathbf{x}$ and $\mathbf{x}'$

Figure 4.1: An example of a region $R^i_{\mathbf{x}}$ with $H = 2$ m is marked with a white line on the player of the dark team at the front right in the image.

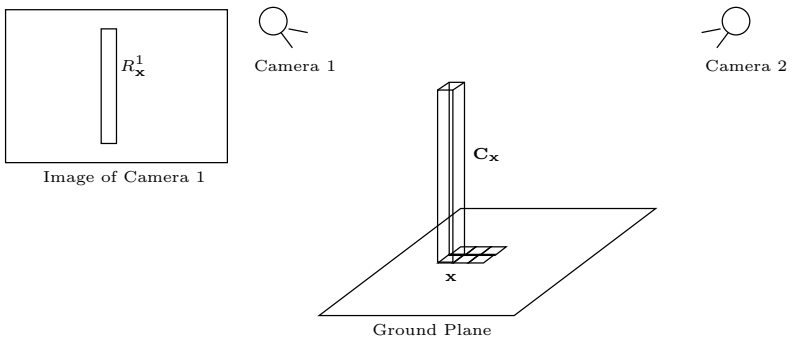
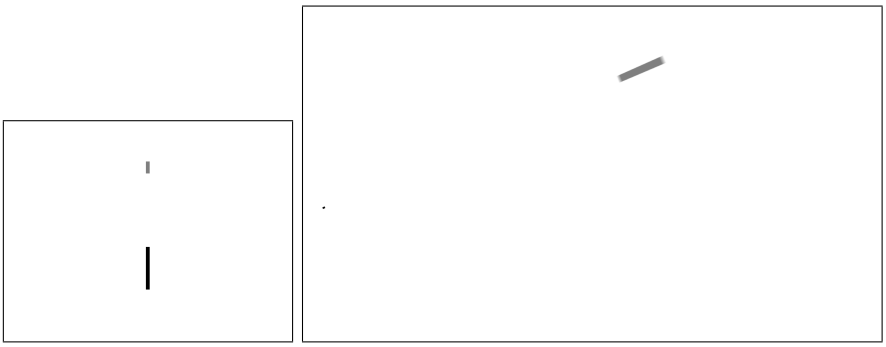


Figure 4.2: The projection of a rectangular cuboid $\mathbf{C_x}$ with height H and cell $\mathbf{x}$ as base into camera view 1 defines an image region $R^1_{\mathbf{x}}$.



(a) Camera image. (b) Occupancy map.

Figure 4.3: (a) A camera image with two regions $R^i_{\mathbf{x}}$, one marked in black and one in gray. (b) Occupancy map with the $G^i_{\mathbf{x}}$ cells $\mathbf{x}$ for which the cuboid $\mathbf{C_x}$ is projected onto the black region $R^i_{\mathbf{x}}$ marked in black, and the $G^i_{\mathbf{x}}$ cells $\mathbf{x}$ for which the cuboid $\mathbf{C_x}$ is projected onto the gray region $R^i_{\mathbf{x}}$ marked in gray.

may be projected onto completely coinciding image regions $R^i_{\mathbf{x}}$ and $R^i_{\mathbf{x}'}$. This is illustrated in Fig. 4.3. Let us first consider the gray (upper) region in the camera image in Fig. 4.3a. Consider two cells $\mathbf{x}$ and $\mathbf{x}'$ lying in the gray ground occupancy map region shown in Fig. 4.3b. The projections of the cuboids $\mathbf{C_x}$ and $\mathbf{C_{x'}}$ in the camera image define the image regions $R^i_{\mathbf{x}}$ and $R^i_{\mathbf{x}'}$ respectively (see Fig. 4.2). For any two cells $\mathbf{x}$ and $\mathbf{x}'$ lying in the gray ground occupancy map region, the regions $R^i_{\mathbf{x}}$ and $R^i_{\mathbf{x}'}$ completely coincide with each other and form the gray image region shown in Fig. 4.3a. In other words, all the cells $\mathbf{x}$ for which the cuboid $\mathbf{C_x}$ is projected onto the gray image region are marked in gray in Fig. 4.3b. This is also illustrated for a second image region, marked in black in Fig. 4.3a. All the cells $\mathbf{x}$ for which the cuboid $\mathbf{C_x}$ is projected onto the black image region are marked in black in the ground occupancy map shown in Fig. 4.3b. Note that in the occupancy map shown in Fig. 4.3b the number of gray cells is a lot higher than the number of black cells. This is because the gray cells are far away from the camera, whereas the black cells are situated close to the camera. This is also the reason why in the image (Fig. 4.3a) the black region $R^i_{\mathbf{x}}$ is a lot larger than the gray region $R^i_{\mathbf{x}}$, even though both regions correspond to the projection of equally-sized cuboids $\mathbf{C_x}$.

Let $G^i_{\mathbf{x}}$ be the number of cells $\mathbf{x}$ for which the cuboids $\mathbf{C_x}$ are projected onto coinciding regions $R^i_{\mathbf{x}}$. If $G^i_{\mathbf{x}} > 1$, the evidence of occupancy collected in $R^i_{\mathbf{x}}$ may be attributable to a person occupying only part of the cells with coinciding $R^i_{\mathbf{x}}$. Because of the reprojection geometry, these $G^i_{\mathbf{x}}$ cells will be approximately laid out in a trapezoid on the ground plane, which we approximate by a square $\mathbf{S}$ with side length $\sqrt{G^i_{\mathbf{x}}}$. The black and the gray regions in the occupancy map shown in Fig. 4.3b are two such trapezoids.

Assuming a person occupies a square of W^2 cells on the ground plane, this person can be in $(\sqrt{G^i_{\mathbf{x}}}+W-1)^2$ different positions with respect to the square

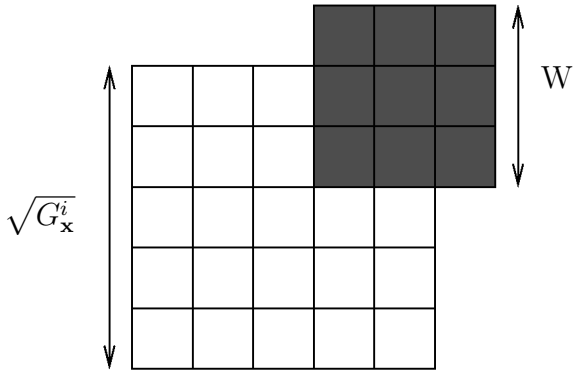


Figure 4.4: Example of a square approximation $\mathbf{S}$ of $G^i_{\mathbf{x}} = 25$ resolution cells $\mathbf{x}$ for which the cuboid $\mathbf{C_x}$ is projected onto the same region $R^i_{\mathbf{x}}$. A person, represented here by the gray square with $W^2 = 9$ resolution cells, can assume $(\sqrt{G^i_{\mathbf{x}}} + W - 1)^2$ different positions such that it overlaps with $\mathbf{S}$. Hence, if $R^i_{\mathbf{x}}$ is completely part of the foreground, there is a probability of $W^2/(\sqrt{G^i_{\mathbf{x}}} + W - 1)^2$ that a particular cell is actually occupied by a foreground object.

$\mathbf{S}$ (see Fig. 4.4). A particular cell $\mathbf{x}$ in the square $\mathbf{S}$ is only occupied in W^2 of all these positions. When observing a fraction $f^i_{\mathbf{x}}$ of foreground pixels, the evidence of individual cells being occupied is smaller than $f^i_{\mathbf{x}}$. If for example $f^i_{\mathbf{x}} = 1$, we are sure that at least one of the $G^i_{\mathbf{x}}$ cells $\mathbf{x}$ for which the cuboids $\mathbf{C_x}$ are projected onto coinciding regions $R^i_{\mathbf{x}}$ is occupied. In this case, the probability that it is occupied is $W^2/(\sqrt{G^i_{\mathbf{x}}} + W - 1)^2$. Hence we scale the fraction $f^i_{\mathbf{x}}$ of foreground pixels with the factor $g^i_{\mathbf{x}} = W^2/(\sqrt{G^i_{\mathbf{x}}} + W - 1)^2$ to obtain the evidence of occupancy of the $G^i_{\mathbf{x}}$ cells $\mathbf{x}$ for which the cuboids $\mathbf{C_x}$ are projected onto coinciding regions $R^i_{\mathbf{x}}$ as $m_i(\{occ_{\mathbf{x}}\}) = g^i_{\mathbf{x}} f^i_{\mathbf{x}}$.
With $m_i(\{occ_{\mathbf{x}}\})$ and $m_i(\{nocc_{\mathbf{x}}\})$ defined, $m_i(\theta_{\mathbf{x}}) = 1 - m_i(\{occ_{\mathbf{x}}\}) - m_i(\{nocc_{\mathbf{x}}\})$.
The distinct pieces of evidence collected by the N views about each cell $\mathbf{x}$ are fused by iteratively applying Dempster's rule of combination (Eq. 4.1). More precisely, let us denote the body of evidence obtained after fusing the occupancy evidence from n cameras as m^n_{fused} and let us initialize m^0_{fused} as $m^0_{\text{fused}}(\theta_{\mathbf{x}}) = 1$, $m^0_{\text{fused}}(\{occ_{\mathbf{x}}\}) = 0$ and $m^0_{\text{fused}}(\{nocc_{\mathbf{x}}\}) = 0$. If the information of the cameras i is fused in the numerical order of their index i, then we can express this iterative fusion process as

$$m^i_{\text{fused}}(C) = m^{i-1}_{\text{fused}} \oplus m_i(C) \tag{4.5}$$

for $C \subseteq \Omega$ and $i = 1 \ldots N$. As for Dempster's rule of combination the order in which the evidence is fused does not matter, the camera indexing can be chosen freely in Eq. 4.5.
This fusion process must be performed for each resolution cell in the occupancy map. We denote the fused evidence of occupancy for all occupancy map cells as $m(\{occ\})$.

Note that the presented algorithm assumes that the people in the scene are not occluded by any objects such as furniture in the scene or objects blocking part of the view of cameras (e.g., cables). If such occluders are present and they can be detected by a scene modeling algorithm, their presence can be easily taken into account by setting $m_i(\theta_{\mathbf{x}}) = 1$ for all occluded cells.

4.5 Adaptations

In a smart camera network the nodes use their on-board image processing hardware to extract from the captured images the necessary observations for occupancy calculation and then transmit this data to fuse it with the data of other cameras. In the method described in Section 4.4, the data transmitted by each camera i will be either the foreground silhouette image or the occupancy evidence $m_i(\{occ_{\mathbf{x}}\})$ and $m_i(\{nocc_{\mathbf{x}}\})$. If the cameras communicate wirelessly, it is very important that the amount of data to be transmitted is kept low. E.g., in the ZigBee specification, the data rate is limited to 20 to 250 kbit/s. However, these are gross rates. Due to overhead, the net maximal data rate is even lower, i.e., about 100 kbit/s. Wireless communication also requires a lot of power, so less communication prolongs battery life.
In Section 4.5.1, we discuss how to adapt the method described in Section 4.4 such that the amount of transmitted data is reduced. This low data rate version of the method is more suited for application in wireless smart camera networks. To further facilitate the use of the proposed occupancy calculation method in such networks, we also explain in this section how we can adapt the method to lower the computational and memory load of the fusion process. In Section 4.5.2 we hint at the possibility of reducing the computational and memory burden of the algorithm even more by reversing the order of the operations executed on the cameras.

4.5.1 Low Data Rate and Low Load Version

We wish to avoid transmitting either the foreground silhouette image or the occupancy evidence $m_i(\{occ_{\mathbf{x}}\})$ and $m_i(\{nocc_{\mathbf{x}}\})$ of each camera, as it is needed in the method described in Section 4.4. To this end, we propose the following data reduction strategy.
Consider the typical silhouette image in Figure 4.5a. Note that the vertical direction in 3D (i.e., the direction perpendicular to the ground plane) nearly coincides with the vertical direction in the image. We say that *verticality* is nearly preserved. This is because this camera, as it is the case for many cameras, is mounted such that the horizontal image axis is nearly parallel to the ground plane. If the computational power at the camera side allows this, the image can be transformed such that verticality is exactly preserved. This homography transformation can be derived directly from the camera calibration data. In the following, we assume that this image transformation has been performed. If the computational power is not sufficient to perform this trans-

(a) (b) (c)

Figure 4.5: Example of (a) a background/foreground segmentation F, (b) a scan-line H, and (c) a column-wise extended scan-line $\hat{F}$.

formation, the application of the subsequent method on an image will lead to similar results as long as verticality is reasonably well preserved.

We exploit this preservation of verticality to make a crude foreground approximation. Let the pixel value of the foreground image $F(x, y)$ be 1 when the corresponding image pixel has been detected as foreground and 0 otherwise. At each camera we add $F(x, y)$ (transformed such that verticality is preserved) along its columns to a 1D horizontal line and threshold it to obtain the scan-line H:

$$H(x) = \begin{cases} 1 & \text{if } T < \sum_{y=1}^{h} F(x, y) \\ 0 & \text{if } T \geq \sum_{y=1}^{h} F(x, y) \end{cases} \tag{4.6}$$

with h the image height. The threshold T should be chosen in accordance with the size at which a person in the scene appears in the image. An example of a scan-line is shown in Figure 4.5b. Cameras transmit their (run-length coded) scan-lines instead of the full background/foreground segmented image. If we assume there are at most B distinguishable objects in an image and the number of bits needed to encode start and end point of each object on the scan-line is at most $2\lceil \log_2 w \rceil$ bits, where w is the image width, then the payload of this transmission can be approximated by $2\lceil \log_2 w \rceil B$ bit. For example, for an image with width=352 and for 5 detected objects, this number amounts to 90 bit.

In essence this approach amounts to approximating the foreground mask F of the original image with $\hat{F}$, the column-wise extension of a scan-line to a 2D image. An example of $\hat{F}$ is shown in Figure 4.5c. In the subsequent text this image is only used to facilitate the explanation of the proposed algorithms. In a real implementation the algorithms should operate directly on the scan-lines. The accuracy of the approximate foreground mask $\hat{F}$ can be improved by dividing the image in tiles and by computing a scan-line for each tile. Combining the introduced horizontal scan-line with a vertical scan-line can further im-

prove the accuracy of the foreground approximation. Note that the calculation of vertical scan-lines comes at an extra computational cost because of the summation of the foreground pixels along a different direction. In this thesis we wish to provide a proof of concept of the scan-line approach by studying the case of one horizontal scan-line per image. The suggested extensions of this main approach are expected to improve the performance of the method, but bring about a higher communication and computational cost. The balance between these aspects must be fine-tuned for particular camera set-ups.

With the goal of further increasing the suitability of the low data rate occupancy calculation method for usage in smart camera networks, we propose an occupancy calculation technique that has a very low computational and memory load. This method consists of back-projecting for each camera the foreground approximation $\hat{F}$ to a common reference plane parallel to the ground plane and fusing these camera occupancy maps with a logical AND operation. The height of the common reference plane H_{AND} should be chosen between 0 and the typical height of a person.

Conceptually, this technique is related to the shape-from-silhouette technique of [Laurentini, 1994] to construct visual hulls. With this technique, within a cuboid-shaped volume V^3 in the 3D space of the observed scene

$$V^3 = [X_1, X_2) \times [Y_1, Y_2) \times [Z_1, Z_2) \subset \mathbb{N}^3, \tag{4.7}$$

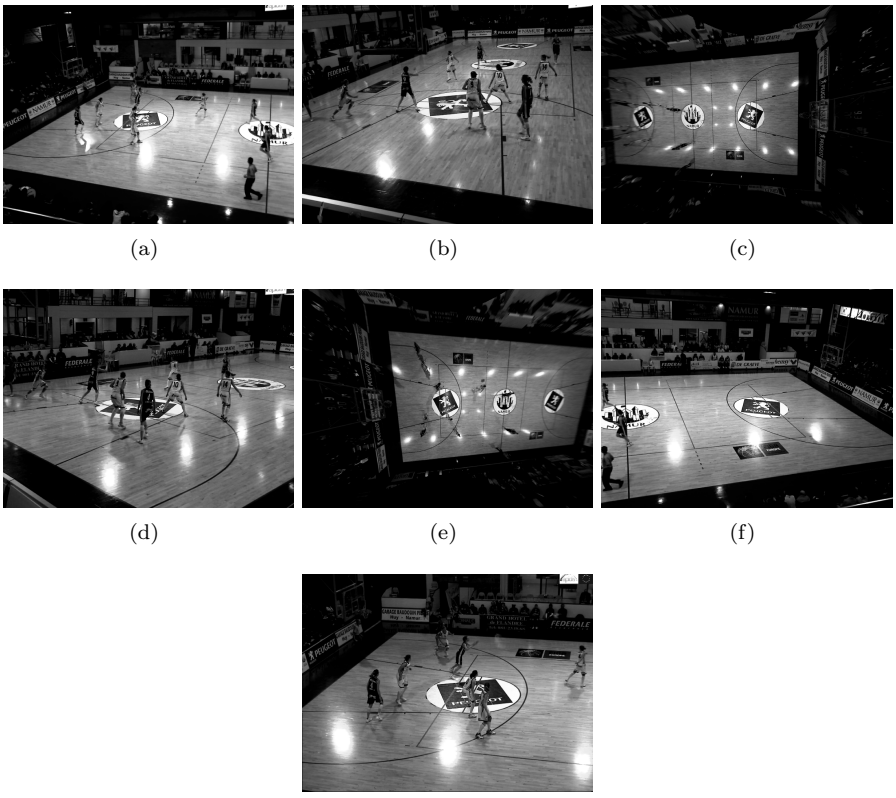
a voxel $\mathbf{j} \in V^3$ assumes the value 0 when it is observed as empty in at least one of the views. In our case this happens when the voxel is part of the reprojected background region in the scan-line based foreground approximation $\hat{F}$ from at least one of the cameras. All other voxels have value 1. Intersecting this visual hull with the plane at height H_{AND} parallel to the ground plane yields us the desired occupancy map.

The quality of the thus obtained occupancy map depends on the quality of the foreground approximation $\hat{F}$. This quality is influenced by the camera set-up and it is better when the objects appear large in the camera image.

4.5.2 Foreground Detection on Scan-Lines Version

In Section 4.5.1 the foreground detected in a camera image is reduced to a scan-line. One can reverse the order of these two operations and first reduce the image to a line by column-wise summing and detect foreground on this line. The feasibility of this has been demonstrated in [Tessens et al., 2009].

On the output of the foreground detection on the scan-line, the method of Section 4.5.1 can be applied to calculate ground occupancy. This system is discussed extensively in the PhD thesis of my colleague Marleen Morbee and is therefore not elaborated on in this work.



(a) (b) (c)

(d) (e) (f)

(g)

Figure 4.6: Undistorted camera views of the APIDIS data set.

4.6 Results

4.6.1 Test Data

To evaluate the proposed method and its adaptations, we use two data sets. The first one is the publicly available basketball data set from the European project APIDIS [De Vleeschouwer and Delannay, 2009]. It consists of seven synchronized and calibrated video streams from five cameras with partially overlapping views distributed around the court, and two top-mounted cameras with fish eye lenses. The views are shown in Fig. 4.6. The videos are processed at a resolution of 800×600 and at 25 fps. The size of the field is $15m \times 28m$. There are on average 12 persons on the field. Ground truth target positions have been made available for 60 frames recorded at 1 s intervals within the time interval 18:47 until 18:48. As most cameras point to the left half of the court, only positions in that half are considered for the evaluation.
The second data set is from an indoor scene of 5m×4m observed by $N = 10$ web

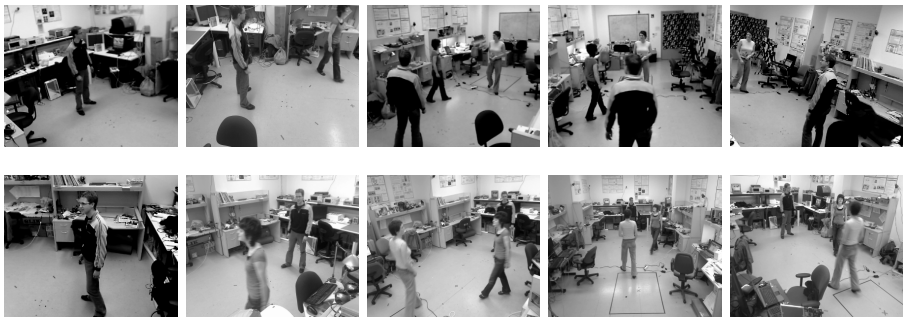


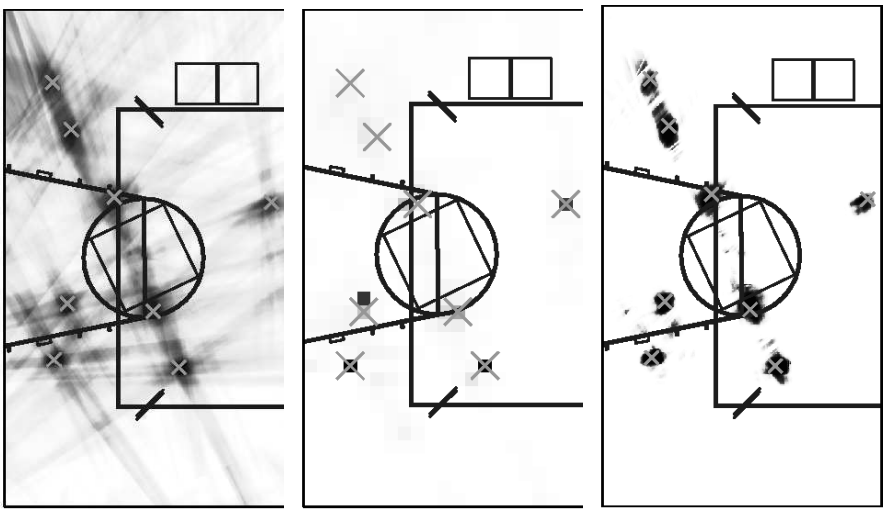
Figure 4.7: Camera views of the WSNL data set.

cameras. The camera views are shown in Fig. 4.7. Approximately 8 minutes of footage (2400 frames) in which two, three and four persons appear, have been recorded at 5 frames per second and at CIF resolution (352×288). Ground truth ground plane positions of the tracked persons have been generated for every fifth frame (1 s intervals). This has been done by manually checking the output of the multi-camera person detection algorithm of [Delannay et al., 2009] and correcting it where necessary.

The average height of a person is set to $2m$, as in [Delannay et al., 2009]. In the rare case of conflicts in the fusion process, all evidence is transferred to $m(\theta_{\mathbf{x}})$. In the first data set we consider square resolution cells with an area of $(0.02m)^2$ and we detect the foreground with an algorithm based on mixture of Gaussians modeling [Stauffer and Grimson, 2000] with elementary shadow removal [Kaewtrakulpong and Bowden, 2001].

The choice of the foreground detection algorithm is important because it is a fundamental building block in the proposed system. Thanks to the different viewpoints of the cameras in a network, gross foreground detection errors, such as the ones introduced by the foreground approximation proposed in Section 4.5.1, can be filtered out to some extent. However, the danger lies in errors that simultaneously occur in all cameras, such as the appearance of shadows or local or global lighting changes. For this reason it is important to use an effective and accurate foreground detector.

In the second data set we take resolution cells of $(0.04m)^2$. The cameras in this set-up have quite an unstable automatic gain control, which is a typical property of very cheap cameras. In a network made up of many cameras, the price of the cameras is indeed an issue and is best kept low. This data set is therefore an interesting test case to assure that our algorithms are not only suited for usage with high-end industrial cameras. Because of the unstable gain control, extreme lighting changes of the observed scene are frequent. For this reason we use a background foreground segmentation algorithm for these sequences that can quickly adapt to such changes [Li et al., 2003] with elementary shadow removal [Kaewtrakulpong and Bowden, 2001].



(a) Sum Method (b) POM Method (c) Proposed Method

Figure 4.8: (a) The aggregated [Delannay et al., 2009], (b) the probabilistic [Fleuret et al., 2008] and (c) the proposed evidential occupancy map for the frames of Fig. 4.6. White corresponds to low confidence/probability/evidence of occupancy, black to high. The crosses indicate the ground truth player positions.

4.6.2 Occupancy from Full Foreground Images

In this section we evaluate the method described in Section 4.4 on the first data set.

The right panel of Fig. 4.8 shows $m(\{occ\})$ in part of the left half of the court for the frames shown in Fig. 4.6. The left panel in Fig. 4.8 shows the aggregated occupancy map obtained as in [Delannay et al., 2009], the middle one the probabilistic occupancy map of [Fleuret et al., 2008] with cell width set to $0.4m$ (other widths yield less accurate results). The map obtained by DS fusion is more representative of the actual occupancy of the field because it shows very clearly defined peaks at the target positions, and very few ghost objects or interference strokes between objects. This is less the case for the methods of Delannay et al. [2009] and Fleuret et al. [2008].

Let the total mass (TM) be the sum over all cells of the occupancy evidence for the proposed method (TM $= \sum_{\forall \mathbf{x}} \mathrm{m}(\{\mathrm{occ}_{\mathbf{x}}\})$), of the aggregated occupancy confidence for the method of Delannay et al. [2009], and of the occupancy probability for the method of Fleuret et al. [2008]. In Fig. 4.9, we plot for our method and the method of Delannay et al. [2009] the percentage of TM that lies within a disc with diameter d around a ground truth target position as a function of d for all the frames in which ground truth target positions are available. For the method of Fleuret et al. [2008] this evaluation method yields poor results because it uses a generative person model that is designed such

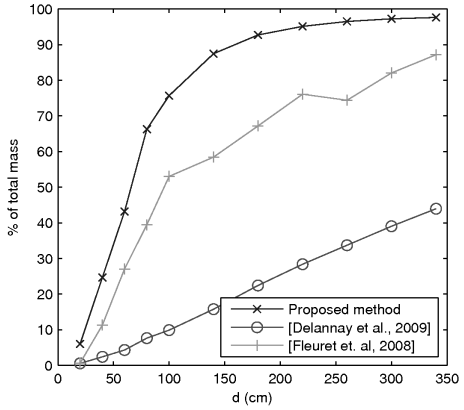


Figure 4.9: For the first environment, the percentage of the total mass within a disc with diameter d around a ground truth target position (for the proposed method and the method of [Delannay et al., 2009]), or within cells with width d actually occupied by a target (for method [Fleuret et al., 2008]).

that the size of the resolution cells should approximate the expected size of the objects to detect. This cell size is significantly larger than in our method and the method of Delannay et al. [2009]. Therefore, for fair comparison we plot for the method of Fleuret et al. [2008] for different cell widths d the percentage of TM that is generated in cells that are actually occupied by a target.

From this graph we conclude that in the proposed method the mass of occupancy evidence is more concentrated around the ground truth positions than the mass of occupancy confidence of method [Delannay et al., 2009] and the mass of occupancy probability of method [Fleuret et al., 2008]. This is obvious from the ratio between the percentage of total mass of our method and the method of Delannay et al. [2009] and Fleuret et al. [2008]. For [Delannay et al., 2009], this ratio ranges from $24.64\%/2.38\% = 10.34$ for $d = 40$cm to $97.61\%/43.96\% = 2.22$ for $d = 340$cm, and reaches 7.67 for a typical diameter of $1m$ for sports players. For [Fleuret et al., 2008], it ranges from $6.02\%/0.65\% = 9.22$ for $d = 20$cm to $97.61\%/87.15\% = 1.12$ for $d = 340$cm, and reaches 1.43 for $d = 1m$. In other words, the ground occupancy map obtained using the proposed method is more accurate than using the methods of Delannay et al. [2009] and Fleuret et al. [2008]. This is beneficial for direct use or for further analysis of the map.

The proposed method is about a factor of six more complex than the method of Delannay et al. [2009]. Indeed, fusing the bodies of evidence of two cameras requires 17 operations per cell. For N cameras this boils down to $17(N - 1)$ operations, compared to $3N + 1$ operations required for [Delannay et al., 2009]. Due to the iterative nature of the algorithm of Fleuret et al. [2008], its complexity is a factor in the order of hundreds higher than that of the proposed method.

Experiments for computation time measurement were performed on an AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set. The performed computations were floating-point computations. The method of Delannay et al. [2009] and the proposed method are implemented in Matlab code, and all mentioned execution times are averages over calculations performed on 10 frames of the test data. Fusing the ground occupancy maps of the cameras with the proposed method took 0.93 s on average, compared to 0.11 s for the method of Delannay et al. [2009]. The ratio between these two times is higher than the theoretically expected factor of six. Obtaining the ground occupancy map per camera for side view cameras took 4.86 s for the method of Delannay et al. [2009]. For the proposed method, an additional 1.48 s are needed because evidence of occupancy *and* of non-occupancy needs to be calculated for each resolution cell. So in total this amounts to 6.34 s. Due to the higher computational requirements to process the top view images captured with fish-eye lenses, obtaining a ground occupancy map from such a camera took 35.54 s on average.
For the method of [Fleuret et al., 2008], the freely available c++ implementation of the authors was used. As this method is based on an algorithm that needs to converge, the computation time is image dependent. On the same processor as described above, averaged over 60 frames, with cell width set to $0.4m$, the method took 17.59 s per frame to execute.

4.6.3 Comparison of Data Rates

In this section we investigate how many bits are required for communication when scan-lines are transmitted, as in the method discussed in Section 4.5.1, as opposed to full foreground images, which is needed in the method described in Section 4.4.
First we discuss the data rate in the method described in Section 4.4. The number of bits required to represent a foreground image depends on the image size, which is l by w. Therefore the number of required bits is lw. This number can be minimized by compression. We assume that PNG compression is used, which is especially suited for the sharp transitions in silhouette images, and that the average compression rate is $\rho = 0.02$. So, the required number of bits after compression can be approximated by $lw\rho$.
In the method discussed in Section 4.5.1, only scan-lines and not full foreground images are transmitted. As mentioned in Section 4.5.1, the amount of bits needed to encode a start or end point of an object in such a scan-line is at most $M = \lceil \log_2 w \rceil$ bit. If we assume there are at most B distinguishable objects in an image frame, the payload of the transmission of a scan-line can be approximated by $2MB$ bit. Alternatively, the scan-line can be run-length coded, which leads to an even smaller payload.
In Table 4.2 we give some numerical examples to compare the required bits. For the image size, we assume that $l = 352$ by $w = 288$. So, we have $M = 9$. For full foreground images, the number of bits is fixed and does not depend on the number of objects. In the table we indicate the average number of bits

Table 4.2: Number of bits required to represent a full foreground image and a scan-line.

	Full foreground image	Full foreground image (compressed)	Scan-line
Required bits	lw	$\pm lw\rho$	$2MB$
5 objects	101376	2151	90
10 objects	101376	2151	180

needed to represent the foreground images in the second data set after PNG compression. For the scan-lines, we need the number of objects in the room. We assume that there are at most 10 objects, and give the maximum number of required bits. However, in a realistic situation objects may occlude each other, so the number of objects visible in each image frame is usually smaller than the number of objects in the room. The cases when $B = 5$ and $B = 10$ are listed in Table 4.2. It can be observed that transmitting scan-lines instead of full foreground images significantly decreases the communication overhead.

4.6.4 Occupancy from Scan-line Approximations

In this section we compare the performance of the low data rate and low load version of the proposed method as it is discussed in Section 4.5.1 with the Dempster-Shafer based method on full foreground images of Section 4.4. The threshold T to obtain the scan-lines is set to one tenth of the image height: $T = h/10$ and H_{AND} is chosen $H_{AND} = 1.29$.
We first apply the methods of Sections 4.4 and 4.5.1 to the first data set. To obtain a scan-line based foreground approximation from the top view images (Figs. 4.6c and e), we transform the detected foreground image $F(x, y)$ to polar coordinates $F'(r, \theta)$ with the optical center of the image as origin and compute the scan-line as

$$H'(\theta) = \begin{cases} 1 & \text{if } T < \sum_{r=1}^{r_{max}} F'(r, \theta) \\ 0 & \text{if } T \geq \sum_{r=1}^{r_{max}} F'(r, \theta) \end{cases} \tag{4.8}$$

The extension of the scan-line $H'(\theta)$ transformed back to the Euclidean image coordinate system yields the desired scan-line based foreground approximation. In Fig. 4.10, we plot for the investigated methods the percentage of TM that lies within a disc with diameter d around a ground truth target position as a function of d for all frames for which ground truth is available. We observe that the occupancy evidence calculated from scan-line based foreground approximations is a lot less concentrated around the ground truth player positions than when the full foreground images are used.
This is also apparent in Fig. 4.11. Fig. 4.11b shows $m(\{occ\})$ in part of the left half of the court for the frames shown in Fig. 4.6, calculated from scan-

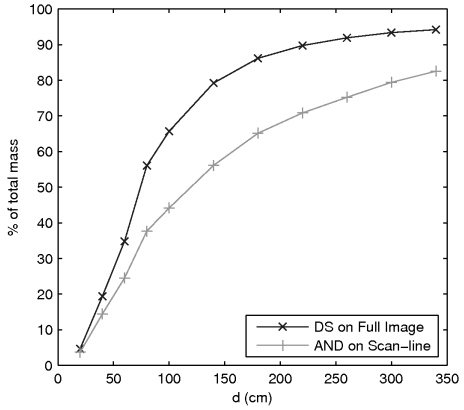


Figure 4.10: For the first environment, the percentage of the total mass within a disc with diameter d around a ground truth target position for the methods of Sections 4.4 and 4.5.1.

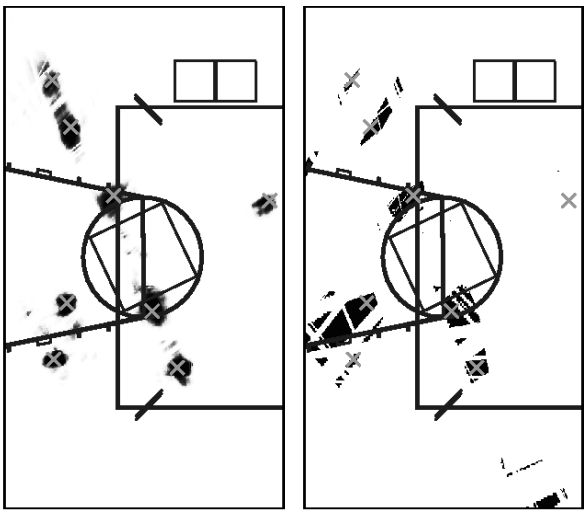
line based foreground approximations using AND-fusion. There is quite some clutter in the left part of the figure, and a completely missed occupancy region in the right part.

To facilitate comparison with the proposed method on full foreground images, Fig. 4.8c has been included in Fig. 4.11a. The reason for the disappointing performance of the method operating on scan-lines is the poor approximation quality of $\hat{F}$. Indeed, as is apparent from Fig. 4.6, the persons in the scene appear small in the images in this data set.

We expect $\hat{F}$ to be a better approximation of F in the second data set because persons appear bigger in the camera images (see Fig. 4.7). Fig. 4.12a shows $m(\{occ\})$ for the frames shown in Fig. 4.7 calculated from full foreground images. Fig. 4.12b shows the occupancy map obtained from scan-line based foreground approximations using the low data rate and low load method. Both methods lead to satisfactory results and comparable amounts of clutter. It appears that in this set-up the low data rate occupancy calculation method is competitive with its full data rate counterpart.

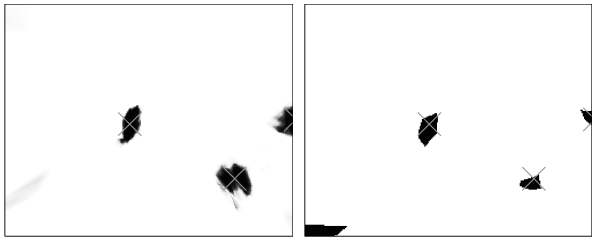
This is confirmed by the numerical results of Fig. 4.13. In this graph we plot for all frames in this set-up for which ground truth is available the percentage of TM that lies within a disc with diameter d around a ground truth target position as a function of d. We observe that the occupancy evidence produced by the low data rate and low load version of the method is less concentrated around the ground truth target positions than the occupancy evidence obtained with the method operating on full foreground images. However, the performance drop is less pronounced than in the first data set.

The calculation of the maps $m_i(\{occ_{\mathbf{x}}\})$ and $m_i(\{nocc_{\mathbf{x}}\})$ for each camera is expected to be quicker than when these maps are computed from full foreground images because the calculation of $b^i_{\mathbf{x}}$ and $f^i_{\mathbf{x}}$ becomes trivial and because of bet-



(a) DS on Full Image (b) AND on Scan-line

Figure 4.11: The occupancy map obtained using the methods of (a) Section 4.4, (b) Section 4.5.1 for the frames of Fig. 4.6. White corresponds to low evidence of occupancy, black to high. The crosses indicate the ground truth player positions.



(a) DS on Full Image (b) AND on Scan-line

Figure 4.12: The occupancy map obtained using the methods of (a) Section 4.4 and (b) Section 4.5.1 for the frames of Fig. 4.7. White corresponds to low evidence of occupancy, black to high. The crosses indicate the ground truth player positions.

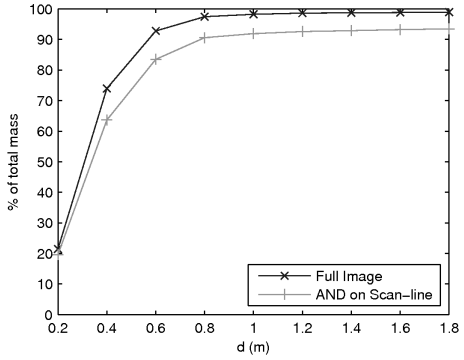


Figure 4.13: For the second environment, the percentage of the total mass within a disc with diameter d around a ground truth target position for the methods of Sections 4.4 and 4.5.1.

ter data locality in the scan-line than in the full image. Experiments to measure this were performed on the same AMD Athlon 64 3400+ 2.40 GHz processor. The mentioned execution times are averages over calculations performed on 10 frames of the test data. In this case computation of the ground occupancy maps per camera took 3.68 s on average. This is indeed less than the 6.34 s needed for full images. The fusion process of the ground occupancy maps of the different cameras is independent of how the individual ground occupancy maps are calculated. Hence, no computation time differences are measured for the fusion process.

4.6.5 Real-time Demonstrator

We have implemented the method of Section 4.4 in a camera network installed at Hogeschool Gent to calculate ground occupancy in real time. The network consists of four progressive CCD color cameras with a resolution of 1024×768, each connected to an Intel Core 2 Duo/1.86GHz processor. Each camera plus computer simulates a smart camera. A base station with the same processor completes the network. The cameras observe a scene of 6m×4m. The resolution cells $\mathbf{x}$ have a size of 0.5cm×0.5cm.

Each camera $i = 1 \ldots 4$ performs foreground detection based on mixture of Gaussians modeling [Stauffer and Grimson, 2000] with elementary shadow removal [Kaewtrakulpong and Bowden, 2001] and calculates for all $\mathbf{x}$ $m_i(\{occ_\mathbf{x}\})$, where $g^i_\mathbf{x}$ is always set to one. $m_i(\{occ\})$ is transmitted over an Ethernet cable to the base station.

As $g^i_\mathbf{x}$ is always one, $m_i(\theta_\mathbf{x})$ depends only on the calibration parameters of the camera (i.e., the viewing range) and is stored at the base station. The base station calculates $m_i(\{nocc_\mathbf{x}\})$ as $m_i(\{nocc_\mathbf{x}\}) = 1 - m_i(\{occ_\mathbf{x}\}) - m_i(\theta_\mathbf{x})$. The occupancy maps of the single cameras $m_i(\{occ\})$ are fused using Dempster's rule of combination to obtain the final occupancy map $m(\{occ\})$.

The base station starts calculating the occupancy map $m(\{occ\})$ as soon as it has received a new $m_i(\{occ\})$ from all four cameras since the last time $m(\{occ\})$ was calculated. However, to make the system resilient against transmissions getting lost or the occupancy calculation of single cameras being delayed, the base station is also programmed to operate at a minimal frame rate fps_{min}. If after a time $1/\text{fps}_{\text{min}}$ it has not received data from all cameras yet, the last received $m_i(\{occ\})$ is used as the current one for all cameras i from which no data was received. In our system, fps_{min}=2fps.
In this demonstrator additionally some area of the ground plane is marked as a forbidden zone. People walking in the forbidden zone trigger an alert. The alert is triggered as soon as one third of the total mass TM is in the forbidden zone.
The system currently operates at 2 to 3 fps. The bottleneck is the calculation of $m_i(\{occ\})$ at the camera side. A more efficient implementation with integral images to calculate $b^i_{\mathbf{x}}$ and $f^i_{\mathbf{x}}$, and larger resolution cells $\mathbf{x}$, would be straightforward ways to speed up calculations. This would also help reduce the latency of the system, which currently amounts to about 1 s.
Fig. 4.14 shows a picture of the demonstrator in use. A video explaining its operation is also available [Tessens and Morbee, 2010]. The carpet marks the observed scene. People are allowed to walk on the light gray track and the dark gray carpet is the forbidden zone. The projector screen on the right shows the alert level on the left - green means no alert at this moment - and the occupancy map on the right. Yellow indicates high evidence of occupancy. The black track represents the allowed zone and the blue regions the forbidden area. The system latency clearly shows up in Fig. 4.14. Indeed, the region of high occupancy evidence corresponding to the left person on the projector screen matches the location where the person was standing about 1 s prior to the current scene. The right person has been stationary for the past second and is therefore shown at the correct location.

4.7 Conclusion

We have described a new method to calculate occupancy maps using multiple cameras. In particular, we have shown how the performance of a method requiring only forward projections from the image to the ground plane can be significantly improved by Dempster-Shafer based fusion of the single view ground occupancy maps. Experiments and a comparison with the state-of-the-art show clear improvements in the fused ground occupancy maps in terms of concentration of the occupancy evidence around ground truth person positions. We have also demonstrated the effectiveness of the proposed method in a four camera network operating in real time.
We have modified this method into a low data and low load version for use in a smart camera network. This version requires that the persons in the scene appear sufficiently large in the camera views. If this is the case, cameras can send only scan-lines of the detected foreground, not the full foreground image.

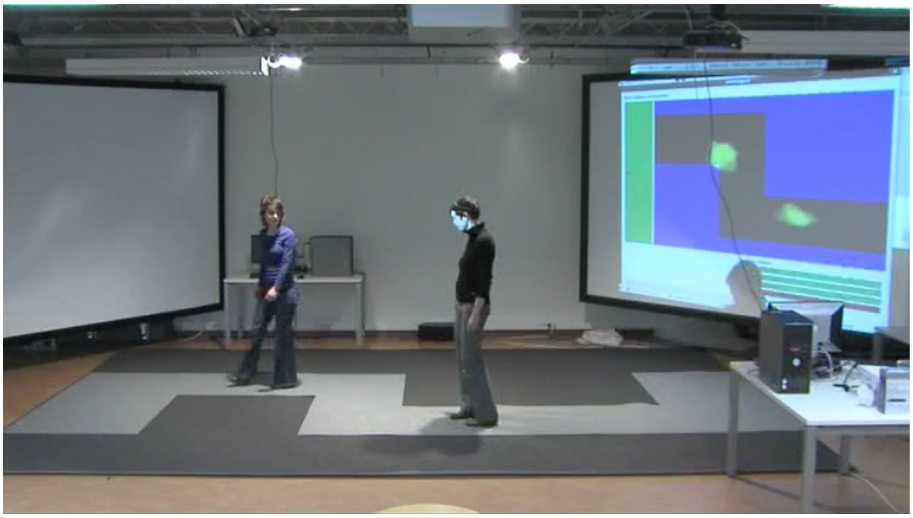


Figure 4.14: Real-time demonstrator in use. The carpet marks the observed scene. People are allowed to walk on the light gray track and the dark gray carpet is the forbidden zone. The projector screen on the right shows the alert level on the left - green means no alert at this moment - and the occupancy map on the right. Yellow indicates high evidence of occupancy. The black track represents the allowed zone and the blue regions the forbidden area.

At the receiver side a scan-line based foreground approximation serves as a good basis to calculated ground occupancy.

5

View Selection for Observability and 3D Shape Reconstruction

In a camera network with overlapping viewing frustums, observations from different nodes are usually highly correlated, resulting in redundant data to be processed. A sensor management system that can fully exploit all available information in the network while keeping the redundancy under control is beneficial, and from a practical point of view often necessary. A possible way to avoid redundant processing is to select a limited number of cameras for each network task. Only these cameras do processing and data is transmitted only between the relevant cameras. With smart cameras such a distributed implementation is feasible. In some cases a central sensor may also be involved, but this is not always necessary. Putting only some cameras to work saves camera and network resources and facilitates multi-tasking where different optimally chosen subsets execute different tasks, e.g., observe specific persons.
In this chapter we study view selection for observing people in a scene and for reconstructing their 3D shape. In applications such as human behavior observation, pose extraction and person identification, the main information content of the joint network observation can be summarized into a limited number of views at each time instant. Depending on the acceptable information loss associated with this data reduction, the selection can be narrowed down to one principal view.
In the next chapter, we will take a more general approach to sensor selection in camera networks.

5.1 Introduction

In this chapter, our interest lies in selecting a limited number of cameras from a network such that this subset constitutes a complete view of the persons in the scene, i.e., that we have a frontal view of one or more persons and that

we can reconstruct their 3D shape. We propose a low data rate method that is designed to be implemented in a distributed way on smart cameras. These allow to extract from the captured images the necessary observations for view selection using distributed computing, thus eliminating the need to collect the image data at a central point. This diminishes the required communication bandwidth within the network, which allows the cameras to work wirelessly, and spreads the computational burden over the camera nodes, resulting in a scalable system. We further design an algorithm to reduce the computational burden of the selection decision and to make the method applicable at high frame rates. We also discuss the practical issues to operate such a network including the network communication protocol.
The remainder of this chapter is organized as follows. Section 5.2 discusses literature related to this work. In Section 5.3, we elaborate on the setup of the system for which we devise our methods and on the assumptions we make. Section 5.4 sketches the layout of the algorithm. Methods for principal view and helper camera selection are explained in Sections 5.5 and 5.6 respectively. The operation time frame is treated in Section 5.7. The performance of the method is discussed in Section 5.8 and conclusions are presented in the last section.

5.2 Related Work

Viewpoint selection has been studied in the fields of computer graphics and robot navigation (see for example [Vázquez et al., 2003] and [Roberts and Marshall, 1998]). The methods developed in these fields require an accurate model of the observed shape(s) and have difficulties coping with the background present in natural scenes, as they were all designed for artificial circumstances. More directly related to this work is [Feris et al., 2007], where a single camera collects key frames of people in surveillance video based on face detections.
View selection for observability is treated in [Daniyal et al., 2010; Jiang et al., 2008; Kelly et al., 2009; Li and Bhanu, 2009; Morbee et al., 2008]. The authors in [Daniyal et al., 2010] assign a score to the content of each view by measuring the activity level, the number of objects, events, etc. The size of the bounding box of an object is used as a quality of view measure in [Jiang et al., 2008], where dynamic programming is used to optimize the selection over time. The object size and centrality in the camera image are considered in [Kelly et al., 2009], complemented by a face detection measure in [Li and Bhanu, 2009; Morbee et al., 2008].
In this work, besides relying on face detection, we extract features such as object size and visibility from a 3D analysis of the scene, reducing the sensitivity of the algorithm to spurious detections in single cameras.
Algorithms for automatically selecting a subset of cameras within practical camera networks have been designed for several other purposes than view selection for observing people and 3D shape reconstruction. In [Soro and Heinzelman, 2007] and [Yu et al., 2007], the authors investigate camera selection within

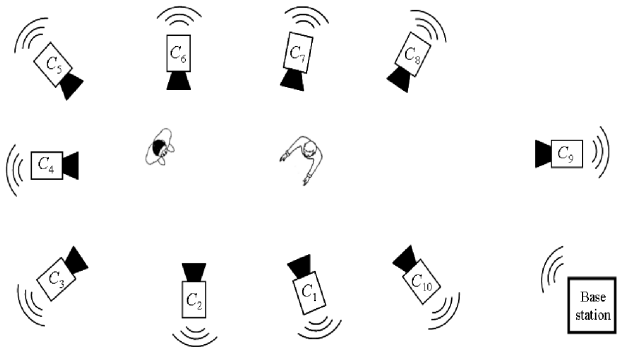


Figure 5.1: Scheme of the system setup.

wireless vision networks of battery-powered nodes under lifetime constraints for user-specified viewpoint synthesis. In [Matsui et al., 2001; Yang et al., 2004], bandwidth and computational issues are considered when cameras within a network are tasked in order to minimize the number of active cameras [Matsui et al., 2001] while determining the occupied space in the scene [Yang et al., 2004]. Also, a related topic is treated in [Bramberger et al., 2005], where real-time allocation of tasks in networks of smart cameras is studied.
We propose a low data rate method with greedy optimization to select a subset of cameras that constitute a complete view on the people in a scene.

5.3 System Setup and Notations

The system we consider consists of multiple smart camera sensors that observe a room with persons inside. A scheme of the system setup is depicted in Fig. 5.1. The smart camera sensors are battery powered and communicate with each other through wireless channels. Their positions and orientations are fixed and calibrated. If the internal and external calibration parameters are available at each time instant, the proposed algorithm can also be applied in a mobile camera network. A base station is deployed to receive the observations from the camera sensors and is responsible for coordinating all sensors in the network. The cameras are denoted by C_i for $i = 1, \ldots, N$, with N the total number of sensors. The complete collection of cameras is the set $\mathbf{C} = \{C_1, \ldots, C_N\}$ where $|\mathbf{C}| = N$. The image captured by the i-th camera at a certain time instant t is denoted by $\mathbf{X}_i(t)$. The different persons or objects are denoted by O_j for $j = 1, \ldots, L$, with L the total number of objects in the scene.
The goal of the proposed algorithm is to select a set of cameras $\mathbf{S} \subseteq \mathbf{C}$, where $|\mathbf{S}| = n \leq N$, that provides a frontal view of as many persons in the scene as possible, and that allows to reconstruct the volume in 3D space occupied by the people in the scene as accurately as possible for the given number of selected cameras. In the remainder of this chapter, we will refer to determining

the volume in 3D space occupied by a person as 3D shape reconstruction. The reconstructed 3D shapes of people are useful as input for higher level algorithms such as pose or gesture recognition or 3D rendering and the frontal view provides an overview for observing the scene. This frontal view is expected to be the view preferred by a human observer among all available views because people usually like to see the front side of a person.
Ideally, the number of selected views n should be updated dynamically as a function of a task related quality measure and a communication and/or computational cost criterion. However, in this work, the focus lies on the criteria for camera selection and the selection process itself, and the number of selected views n is kept fixed. The formulation of a task related quality measure is the subject of Chapter 6 of this thesis. The definition of a computational cost criterion is treated in the PhD thesis of my colleague Marleen Morbee, who has also studied the dynamic updating of the number of cameras selected for a task.
The camera selection decision is based on a limited amount of information that the cameras locally extract from the observed images and transmit to the base station. The base station runs the camera selection algorithm based on the received data and broadcasts the selection result to all the camera sensors. Only the selected cameras send their complete image to the base station. The remaining $N - n$ cameras do not send any image data. At the base station, the images can be watched, stored or processed further.
The selected set of cameras contains two types of cameras:

- The key or principal camera: the camera with the view that contributes most to the desired observation of the scene at a certain time instant, i.e., that captures a frontal view of one or more people in the scene. The key camera is indicated by K.

- One or more helper cameras: cameras with views that complement the selected key view and that together with the key view allow to reconstruct the 3D shape of the persons in the scene as accurately as possible.
  The $n-1$ helper cameras are indicated by W_k where $k = 1, \ldots, n-1$.

Note that $\mathbf{S} = \{K\} \cup \{W_1, \ldots, W_{n-1}\}$. The total selected view subset constitutes a significantly more efficient scene representation than the totality of the available views.
Although the transmission of an image is now delayed by the time it takes the base station to make and communicate its selection decision, the time gain resulting from not having to transmit complete images from all nodes ensures that the observation frequency of this system can be considerably higher than that of one without view selection. For example transmitting a 352×288 JPEG-compressed color image of 50kB (corresponding to a compression rate of 0.17) using 100kbit/s (about the maximal net data rate achievable under the ZigBee specification) takes 4.0 s.
To reduce the time of the camera selection, it is important to lower the time needed for the base station to collect the input data from the nodes. This is

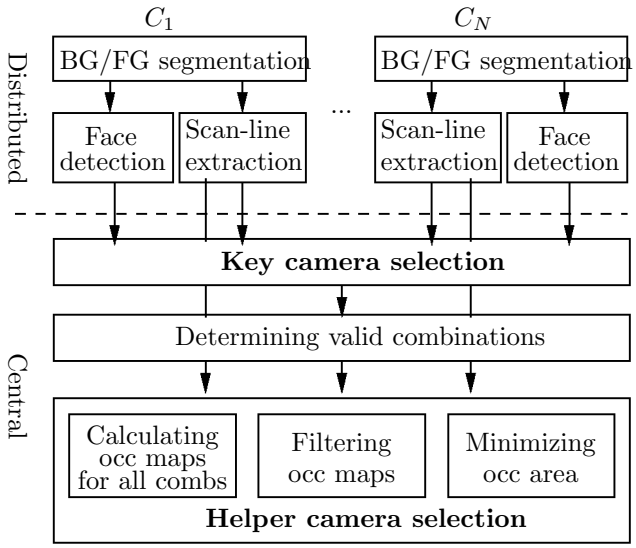


Figure 5.2: Block diagram for camera selection.

why it is of paramount importance that the nodes send only small amounts of *processed* information as input for the camera selection algorithm.
It is possible to augment the observation frequency by not running the view selection algorithm for every frame but applying a selection decision to several frames. Also, one can determine the selection at a certain time instant based on the observation data of a previous time instant. These frame rate increasing strategies have an impact on the accuracy of the camera selection as necessary switches of the selection will be delayed. These issues will be discussed in Section 5.7.

5.4 Algorithm Architecture

The algorithm block diagram is depicted in Fig. 5.2. We now explain the main building blocks.

5.4.1 Distributed Processes

In the first phase of the algorithm, the nodes process the observed images to yield only the information necessary for the base station to determine the camera selection. The lower the amount of data that needs to be transmitted, the quicker this decision can be made and the higher the achievable observation frequency.
Each smart camera C_i independently runs the following algorithms on its image $\mathbf{X}_i(t)$ captured at a certain time instant t. In a first step, we segment the

foreground (FG) $\mathbf{F}_i(t)$ and the background (BG) $\mathbf{B}_i(t)$ of the frames $\mathbf{X}_i(t)$. Foreground objects are objects of interest, i.e., objects that are not part of the background of the scene. In our application these are the persons moving in the scene. This implies that people or objects that come to a standstill and do not move during a predetermined time (the length of which depends on the parameters of the foreground detection algorithm) will inevitably become background objects. In another application the foreground may for example be people behaving abnormally in a crowd. We use the method of [Li et al., 2003] to detect foreground objects. This method uses a Bayesian decision rule to classify pixels as background or foreground based on features extracted from long-term image statistics as well as the temporal difference between the current and the previous frame.

Then, we detect the frontal faces with the object detector that was initially proposed in [Viola and Jones, 2001] and then improved in [Lienhart and Maydt, 2002]. At the core of this method is a cascade of complex classifiers. Each complex classifier consists of several simple classifiers that detect specific Haar-like features. An image region is classified as being a face if the region has passed all classification stages of the cascade. To speed up processing and to lower the number of false detections, we restrict the face detection to the foreground regions of the frame. If even more efficient processing is needed, the face detection processing could take into account the face detections of the previous time instance.

At each time instant t, the face detector returns the following values: $f_i(t)$ and $Q_i^l(t)$, $l = 1, \ldots, f_i(t)$. $f_i(t)$ is the number of faces detected in the frame $\mathbf{X}_i(t)$. $Q_i^l(t)$ is a measure of the quality of the l^{th} detected face. The lower this measure, the less certain the detection. In our implementation, we assume that the number of simple classifiers in the face detector that have detected the feature which they were trained to detect is such a measure.

The face detection measures $Q_i^l(t)$ of all detected faces are added into one general score

$$Q_i(t) = \sum_{l=1}^{f_i(t)} Q_i^l(t), \tag{5.1}$$

which is sent to the base station. With proper quantization if needed, this score can be represented by at most one byte.

Additionally, as in Section 4.5.1, we project in each camera C_i the segmented foreground $\mathbf{F}_i(t)$ onto a horizontal line. This line is called scan-line and an example is shown in Figure 5.3b. All cameras send their (run-length coded) scan-lines to the base station.

Assuming at most B distinguishable objects in an image frame, the payload of this transmission can be approximated by $2\lceil \log_2 w \rceil B$ bit, where w is the image width (see Section 4.5.1). For example, for an image with width=352 and for 5 detected objects, this number amounts to 90 bit. Together with the output of the face detector, only 98 bits are transmitted per frame.

In the remainder of this chapter, we will leave out the time variable t when talking about the observations and processing of the current time instant, in

(a) (b) (c)

Figure 5.3: Example of (a) a background/foreground segmentation ($\mathbf{B}_i$ and $\mathbf{F}_i$), (b) a scan-line, and (c) a column-wise extended scan-line ($\mathbf{B}_{i,\mathrm{sc}}$ and $\mathbf{F}_{i,\mathrm{sc}}$).

order not to overload the notations. We will again use the time variable when previous observations are taken into account.

5.4.2 Central Processes

At the base station, we extend the received scan-lines to very rough approximations of the background and foreground regions $\mathbf{B}_{i,\mathrm{sc}}$ and $\mathbf{F}_{i,\mathrm{sc}}$ (see Fig. 5.3). Using the foreground approximations from all cameras we calculate an occupancy map $\mathbf{O_C}$ with the method of Section 4.5.1.
From this occupancy map, we extract a number of cues on which we base our camera selection.
The selection of a subset of cameras to observe the scene efficiently starts with the determination of the key or principal view. In Section 5.5 we present two methods to determine this view. To complement the key view and to allow 3D shape reconstruction, one can decide to select additional helper views that complement the view of the key camera. This is discussed in Section 5.6.

5.5 Principal View Determination

5.5.1 Face Detection Only

In a first method for principal view determination the face detection score of Eq. 5.1 is used to select the principal view [Morbee et al., 2008]. To deal with spurious face detections and to obtain smoothness over time, the decision on the key camera for time instant t not only depends on the current face detection output, but also on the previous observations. For each camera C_i, the temporally filtered face detection score $S_i(t)$ is an exponentially weighted moving average of the current observation and the previous temporally filtered face detection score $S_i(t-1)$, with $S_i(0) = 0$:

$$S_i(t) = \alpha \sum_{l=1}^{f_i(t)} Q_i^l(t) + (1-\alpha) S_i(t-1), \forall t \geq 1 \tag{5.2}$$

where α is a constant between 0 and 1 that determines the importance of previous observations. Then, the key camera at time instants $t \geq 1$ is

$$K(t) = \arg\max_{C_i} S_i(t) \tag{5.3}$$

5.5.2 Face Detection and Occupancy Map Cues

In a second key camera selection method we combine the face detection scores of all views with knowledge about the scene layout that we extract from the occupancy map. More precisely we determine the position and velocity of each detected object O_j, with $j = 1 \dots L$. Velocities are determined by calculating the distance covered by each object from the previous to the current frame. Armed with this information and with the output $Q_i(t)$ of the face detector on each camera, we assess the suitability of each camera to be assigned the role of key camera.
We propose different factors to determine this suitability.

- The *visibility* ν_{ij} of each object O_j in the view of camera C_i: This measure takes on value 1 if the center of mass of the object lies within the viewing range of the camera and 0 otherwise. The viewing range of each camera is determined from the calibration data.

- The *moving direction* of each object O_j *relative to the viewing direction* $\mathbf{\Psi}_i$ of camera C_i: With $\mathbf{V}_j$ the velocity of object O_j, negative values of $G_{ij} = \mathbf{V}_j \cdot \mathbf{\Psi}_i$ (with $\cdot$ denoting the scalar product between two vectors) indicate that the object is moving towards the camera. In this work, we assume that an observed person's body is oriented in the direction of his or her movement. As we wish to obtain frontal views of the observed persons, we introduce a binary value γ_{ij} which is 1 when G_{ij} is negative and 0 otherwise.

- The *distance* D_{ij} between the center of mass of object O_j and the camera center of C_i: This distance is normalized by dividing it by the maximal possible distance $D_{\max}$ between an object in the observed space and a camera center. To avoid evaluating square roots, we always work with the square of distances. If an observed person's body is oriented towards a camera - which can be ascertained when the velocity vector points towards the camera or if the person's face is detected - a small distance between camera and object is desirable.

- The *speed* $\|\mathbf{V}_j\|$ at which each object O_j is moving: If this speed is very small, we assume that we cannot conclude anything about the body orientation of the observed person as (s)he might be standing still or rotating around his or her axis. The binary value μ_j indicates if the speed exceeds a certain threshold K_S, in which case $\mu_j = 1$. Otherwise $\mu_j = 0$. This measure is camera independent.

- The output Q_i of the *face detector* on each camera C_i: As the face detection score of each camera is the sum of the scores of all faces detected in its view, it is not linked to a particular object. In this method, we do not temporally filter the output of the face detector (as in Section 5.5.1), because the occupancy map related factors also used to assess the suitability of a camera to be the key camera already have a temporally smoothing effect on the key selection.

We summarize these factors into a score for each camera C_i:

$$S_i = K_Q Q_i + \sum_{j=1}^{L} \nu_{ij}\gamma_{ij}\mu_j \Big(-K_G G_{ij} + K_D(1 - \frac{D_{ij}^2}{D_{max}^2}) \Big), \tag{5.4}$$

where K_Q, K_G and K_D are positive tuning parameters that weight the contribution of each factor. In the current system, these parameters have been optimized experimentally and then fixed. This tuning was done manually on a very limited number of frames. K_Q is chosen such that if a face is detected, the term $K_Q Q_i$ is much larger than the other terms in Eq. 5.4. This ensures that if a face is detected in only one view, this view will be selected as the principal view. The parameters K_G and K_D have been chosen such that $K_G > K_D$. The reason is that unlike D_{ij}, G_{ij} is indicative of the frontal view of a person because the assumption that a person's body is oriented in the direction of his or her movement is more likely to be valid if G_{ij} is large. In this way, if no faces have been detected in any of the views, the view which is most likely to provide a frontal view of one or more persons in the scene will be selected as the principal view, even if other cameras observe the persons from closer. If more than one view is equally likely to provide a frontal view of one or more persons in the scene, the camera which observes the persons from the closest distance is chosen.
The dynamic adaptation of the value of the tuning parameters (e.g., based on a probabilistic modeling of the scene dynamics) would make the algorithm more universally applicable and flexible. In Chapter 6, we introduce a more general and theoretically founded way of evaluating quality-of-view measures for camera selection.
Note that the score S_i can never assume negative values, but it can be zero if no faces are detected in the camera i and if the observations extracted from the occupancy map are inconclusive. The latter case occurs

- if no objects are visible in any of the cameras,
- if all objects move away from the cameras in which they are visible, or
- if all objects move at speeds below the threshold K_S.

To obtain smoothness over time, the decision on the key camera for time instant t not only depends on the current observations, but also on those obtained at previous time instants. The default choice for the key camera $K(t)$ at time

Input: $S_i, i = 1 \dots N$ (the scores from the different cameras)
Output: $K(t)$ (the key camera for this time instant)

$K(t) \leftarrow K(t-1)$
$S_{max} \leftarrow 0$ and $R_{key}(t) = \text{NRQ}$
for $i = 1$ to N **do**
    if $S_i > S_{max}$ **then**
        $S_{max} \leftarrow S_i$
        $R_{key}(t) \leftarrow C_i$
    end if
end for
if $R_{key}(t) \neq \text{NRQ}$ **then**
    if $R_{key}(t) = R_{key}(t-1)$ **then**
        $K(t) \leftarrow R_i(t)$
    end if
    if $K(t) \notin \{R_{key}(t), \dots, R_{key}(t-T)\}$ **then**
        $K(t) \leftarrow R_i(t)$
    end if
end if

algorithm 1: Principal View Determination

instant t is the previous key camera $K(t-1)$. It is then possible for all cameras to place a request to take over the role of key camera. Let the acronym NRQ denote 'No ReQuest'. The camera placing the request (the requester) is denoted by $R_{\text{key}}(t)$, with $R_{\text{key}}(t) \in \mathbf{C} \cup \{\text{NRQ}\}$. If all scores are zero, no request is placed and $R_{\text{key}}(t) = \text{NRQ}$. Otherwise, the camera with the highest score $S_i(t)$ at time instant t places the request :

$$R_{\text{key}}(t) = \begin{cases} \text{NRQ} & \text{if } \forall C_i, S_i(t) = 0 \\ \arg\max\limits_{C_i} S_i(t) & \text{otherwise} \end{cases} . \tag{5.5}$$

This request is granted if the same camera also placed a request at time instant $t-1$ or if the current key camera has not placed a request during the past T frames. The parameter T should be chosen as a function of the frame rate and of the maximal time delay which the user would allow for switching away from a principal view that is not suited anymore, when there is no other view that has placed two subsequent requests to take over the role of principal view. In this way, excessive switching between cameras that are equally suitable to provide the principle view is averted, while simultaneously avoiding that alternating requests from such cameras prevents the role from being passed on to a more suitable camera than the current key camera. Also, the delay for a necessary switch of principal view is limited to one frame at most.

This algorithm is summarized in Algorithm 1.

5.6 Helper Camera Selection

Among the remaining $N-1$ cameras we choose helper cameras, W_k, where $k = 1, \ldots, n-1$, with n the total number of selected views. These helper cameras also transmit their image to the base station. They complement the image data from the already selected key camera K and allow to reconstruct the 3D shape of the people in the scene. The 3D shapes of people, i.e., the volume they occupy in 3D space, can for example be recovered using the shape-from-silhouette technique [Laurentini, 1994]. With this technique, within a cuboid-shaped volume V^3 in the 3D space of the observed room

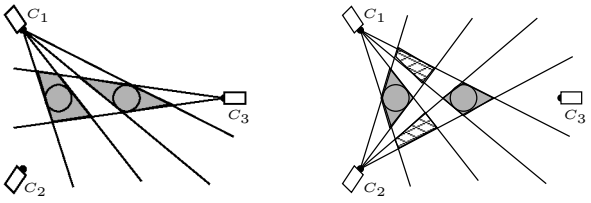
$$V^3 = [X_1, X_2) \times [Y_1, Y_2) \times [Z_1, Z_2) \subset \mathbb{N}^3, \tag{5.6}$$

the visual hull $\mathbf{H}(\mathbf{j})$, $\mathbf{j} \in V^3$, assumes value 0 when the voxel $\mathbf{j}$ is observed as empty by at least one of the selected cameras. This is the case when it is part of the reprojected BG region from at least one of the selected cameras. All other voxels have value 1. The 3D shape reconstruction can be further refined, e.g., by identifying skin color in the selected views to locate hands and faces, or by fitting appearance models. The reconstructed shapes, together with the image data available at the base station, can serve as input for, e.g., pose recognition or 3D rendering algorithms.

To determine which helper cameras to select, we assume that the occupancy maps are (very crude) 2D ground plane shape approximations of the objects in the scene and that the subset that yields the minimal occupied area provides the best 3D shape reconstruction (see Figure 5.4). Indeed, the more resolution cells are observed as empty around the objects in the scene, the better the 2D shape reconstruction and the more the selected subset observes them from different viewing directions. Note that the shape of people that can be reconstructed based on the selected cameras is a 3D volume and not just a 2D approximation, which is what the occupancy map provides us with.

Our approach consists of the following steps. First, the base station determines all the valid candidate subsets $\mathbf{S} \subseteq \mathbf{C}$, for which $|\mathbf{S}| = n$ and $K \in \mathbf{S}$, with K determined as in Section 5.5. At this time, all the cameras have already sent their scan-lines to the base station, which has used them for principal view selection. For each candidate subset $\mathbf{S}$, we now use only the scan-lines from cameras in the subset: $C_i \in \mathbf{S}$ to reconstruct the occupancy map $\mathbf{O_S}$ for that candidate subset with the method of Section 4.5.1.

Subsequently, this occupancy map is filtered to remove *ghost* areas. These are parts of the occupancy map that do not represent real objects but result from an insufficient number of used cameras n (see figure 5.4). The map that is used as a filtering mask is a dilated version $\mathbf{O}_\mathbf{C}^{\text{filt}}$ of the ideal occupancy map $\mathbf{O_C}$, reconstructed from the scan-lines of the complete set of cameras $\mathbf{C}$ (as calculated in Section 5.4.2). In this way, we ensure that we base our camera selection only on the shape approximation of objects that are also detected when all N cameras are selected and the influence of ghost areas is minimized.



(a) Occupancy map when C_1 and C_3 are selected.

(b) Occupancy map when C_1 and C_2 are selected.

Figure 5.4: Occupancy maps when specific cameras are selected. Detected objects are marked in gray. The ghost regions in (b), marked with a pattern, are filtered out with the knowledge of the occupancy map calculated from all three cameras. As the dark gray area around the circular objects is smaller in (b) than in (a), C_2 adds more shape information than C_3 and is assumed to provide a better complementary view to C_1 than C_3.

The size of the filtered occupied area $\mathbf{A}(\mathbf{S})$, with

$$\mathbf{A}(\mathbf{S}) = \sum_{\forall \mathbf{j} \in P^2} \mathbf{O_S}(\mathbf{j}) \mathbf{O_C^{filt}}(\mathbf{j}), \tag{5.7}$$

is considered the camera selection criterion in our algorithm. Thus we select from all candidate subsets the final subset $\mathbf{S}_n$ that yields the minimal occupied area $\mathbf{A}(\mathbf{S})$:

$$\mathbf{S}_n = \arg\min_{\forall \mathbf{S}} \mathbf{A}(\mathbf{S}). \tag{5.8}$$

This subset constitutes a significantly more efficient scene representation than the totality of the available views. Only the selected n cameras transmit their full image to the base station.
The number of candidate subsets is $\binom{N-1}{n-1}$. If for example in a network of $N = 10$ cameras we wish to select $n = 3$ cameras, $\binom{9}{2} = 36$ candidate subsets need to be checked. If in a network twice this size, i.e., $N = 20$, we wish to select twice as many cameras, i.e., $n = 6$, we need to check $\binom{19}{5} = 11628$ candidate subsets. Clearly the number of candidate subsets quickly grows with increasing camera network size. This means that performing an exhaustive search over all candidate subsets to identify the optimal one which minimizes the occupied area becomes computationally very demanding for larger networks.
We therefore propose a greedy algorithm that starts from the set selected at the previous time instant to select those $w = n-1$ helper cameras $\{W_1, \ldots, W_{n-1}\}$ that add most shape information to the image data from the key camera selected at the current time instant. The algorithm consists of two steps:

- removing from the set of the previous time instant those cameras that add least shape information to the image data from the key camera selected at the current time instant;

- adding to this reduced set those cameras that add most shape information to the image data from the key camera selected at the current time instant.

This approach allows to keep the selected camera set updated as the scene changes over time, while exploiting the temporal smoothness of these scene changes to avoid an exhaustive search over all possible camera sets.

Assume that the algorithm reevaluates the selection status of at least u cameras to obtain the camera set for the current time instant. The more cameras have their selection status reevaluated, i.e., the higher u, the more the selection can be adapted to possible scene changes, but also the higher the computational burden of the selection.

Let us assume that the cameras selected at time instant $t-1$ form the set $\mathbf{S}$. The algorithm first combines the key camera K selected at time instant t with the selected camera set of the previous time instant $t-1$ to form a new set $\mathbf{S}' = \mathbf{S} \cup \mathbf{K}$. Initially, at time instant $t = 0$, we start the algorithm with all cameras selected, which means that $\mathbf{S}$ is equal to $\mathbf{C}$.

The first part of the algorithm greedily removes r cameras one at a time from the set $\mathbf{S}'$ such that the remaining selected cameras yield the minimal occupied area. The resulting camera set $\mathbf{S}''$ includes the remaining selected cameras. Given the number u, the number of cameras that will be first removed from the set $\mathbf{S}'$ is

$$r = \begin{cases} u, & \text{if } |\mathbf{S}'| = n, \\ |\mathbf{S}'| - n + u, & \text{if } |\mathbf{S}'| > n. \end{cases} \tag{5.9}$$

In other words, the number of removed cameras r equals $N - n + u$ after initialization (because then $\mathbf{S}' = \mathbf{C}$), r equals $u + 1$ if the key camera selected for the current time instant was not part of the selected set of the previous time instant (then $|\mathbf{S}'| > n$), and r equals u otherwise.

The second part of the algorithm greedily adds $u = n - |\mathbf{S}''|$ cameras one at a time to $\mathbf{S}''$ such that the set of selected cameras yields the minimal occupied area. We denote the final selection solution as $\hat{\mathbf{S}}_n$. The pseudo-code of the algorithms are summarized in Algorithm 2 and 3.

In case of the greedy algorithm, the number of subsets that needs to be checked is $r(N-1)$. If for example in a network of $N = 10$ cameras we wish to select $n = 3$ cameras starting from a set of this size, and we reevaluate the status of $u = 2$ cameras, 18 subsets need to be checked. This is half the 36 candidate subsets that need to be checked when using the optimal algorithm. If in a network twice this size, i.e., $N = 20$, we wish to select twice as many cameras, i.e., $n = 6$, starting from a set of this size, and we reevaluate the status of $u = 5$ cameras, we need to check 95 subsets. This is roughly $1/122$ of the 11628 candidate subsets that need to be checked when using the optimal algorithm Clearly the advantage of using the greedy instead of the optimal algorithm is larger in larger camera networks.

Input: $\mathbf{S}'$ (a set of currently selected cameras)
Output: $\mathbf{S}''$ (a set of cameras after removing)
A denotes the size of the occupancy area
A_{min} denotes the minimal size over all sets tested so far

for $m = 1$ to r **do** (remove r cameras)
  $A_{min} \leftarrow +\infty$
  for each camera C_i in $\mathbf{S}'$ **do**
    if $C_i \neq K$ **then**
      $\mathbf{S}' \leftarrow \mathbf{S}' \backslash \{C_i\}$
      $A \leftarrow \mathbf{A}(\mathbf{S}')$, the size of occupancy area given cameras $\mathbf{S}'$
      if $A < A_{min}$ **then**
        $\mathbf{S}'' \leftarrow \mathbf{S}'$
        $A_{min} \leftarrow A$
      end if
      $\mathbf{S}' \leftarrow \mathbf{S}' \cup \{C_i\}$
    end if
  end for
  $\mathbf{S}' \leftarrow \mathbf{S}''$
end for

algorithm 2: Greedy Selection Algorithm - Removing

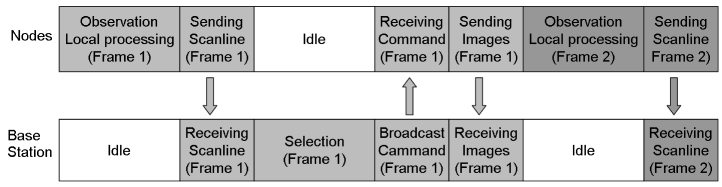


Figure 5.5: The time frame of the basic operation scheme.

5.7 Operation Time Frame

The basic operation time frame is shown in Fig. 5.5 where different colors indicate operations on different image frames (i.e., captured at different time instances). The sensor nodes first make observations and process the images locally. The main operations this processing encompasses are background subtraction and face detection. Then, each node sends its scan-line to the base station. After receiving the scan-line from all nodes, the base station runs the greedy selection algorithm and broadcasts the result. Finally, the selected nodes transmit their images to the base station, after which the nodes start making new observations for the next frame and a new cycle starts.

From Fig. 5.5, it can be observed that both the base station and camera nodes have idle time slots, which increases the interval between observations. In order to increase the observation frequency, we propose an interleaving scheme as

Input: $\mathbf{S}''$ (a set of cameras after removing)
Output: $\hat{\mathbf{S}}$ (a set of new selected cameras)
A denotes the size of the occupancy area
A_{min} denotes the minimal size over all sets tested so far

```
1: Ŝ ← S''
2: for m = 1 to n − |S''| do (add n − |S''| cameras)
3:     A_min ← +∞
4:     for each camera C_i in C do
5:         if C_i ∉ S'' then
6:             S'' ← S'' ∪ {C_i}
7:             A ← A(S''), the size of occupancy area given cameras S''
8:             if A < A_min then
9:                 Ŝ ← S''
10:                A_min ← A
11:            end if
12:            S'' ← S''\{C_i}
13:        end if
14:    end for
15:    S'' ← Ŝ
16: end for
```

algorithm 3: Greedy Selection Algorithm - Adding

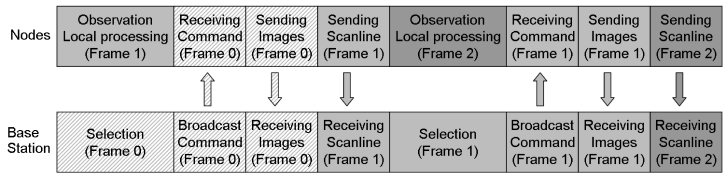


Figure 5.6: The time frame of the interleaving operation scheme.

shown in Fig. 5.6. In this scheme, the operations on different image frames are interleaved to minimize the idle time. While the nodes are making observations for Frame 1 (marked by light gray in Fig. 5.6), the base station decides on the camera selection based on the observations of a previous frame (Frame 0, marked by light gray stripe pattern). After the selection is completed, each node receives the broadcast from the base station and the selected ones transmit their image frames (Frame 0), and once the image frames (Frame 0) are sent, each node starts sending the scan-line of frame (Frame 1).

Although the interleaving operation scheme increases the observation frequency, it increases the delay between the observation of a frame and the same frame received at the base station. To decrease this delay, we propose the advanced operation scheme shown in Fig. 5.7. In this scheme, the camera nodes receive selection results from Frame 0 (marked by light gray stripe

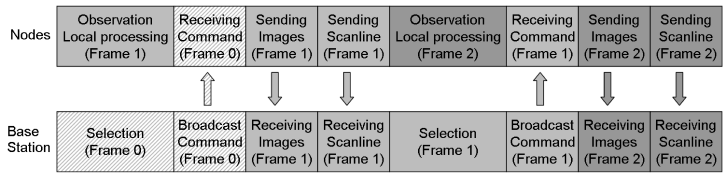


Figure 5.7: The time frame of the advanced operation scheme.

pattern in Fig. 5.7) right after making the observations for Frame 1 (marked by light gray). Instead of sending the image frames (Frame 0) as in the interleaving scheme, the selected nodes now transmit image frames (Frame 1). In other words, we assume that the difference between successive observations is small. Under this assumption, the base station can select the current camera set based on a previous observation. This scheme is useful when the frame rate of the system is sufficiently high with respect to the scene changes in the room, such that successive observations result in similar selection results.

5.8 Results

In this section, we assess the performance of the proposed camera selection methods for observability and 3D shape reconstruction.

Experimental data for testing the method on, was recorded with a camera network set up as described in Section 5.3. One to four persons were present in the scene.

In this work, we use sequences captured in the second multi-camera set-up described in Section 4.6.1. To briefly recapitulate: this is an indoor scene observed by $N = 10$ web cameras. The camera views can be seen in Fig. 5.11. Sequences have been recorded at 5 frames per second and at a CIF resolution (352×288).

The resolution cells of the occupancy map have a side length of $0.04m$. The structuring element for the dilation to obtain the filters $\mathbf{O}_{\mathbf{C}}^{\text{filt}}$ (see Section 5.6) and $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$ (see later, in Section 5.8.2) is a square of 11×11 with the origin at its center.

The tuning parameters in Eq. 5.5 are set to $K_Q = 1$, $K_G = 2$ and $K_D = 1$. The threshold for the speed is $K_S = 0.08$ m/frame $= 0.4$ m/s. The temporal filtering parameters of the key selection are set to $\alpha = 0.05$ and $T = 4$. These parameters have been manually tuned on a very limited number of frames. K_S has been chosen as a very small speed. α has been set to a small value, such that previous observations are weighted heavily. When the value of α is increased, the principal view will be switched more frequently. This will also happen if T is set to a smaller value than the current $T = 4$.

To evaluate the quality of the principal view selected by the methods of Section 5.5, we use sequences labeled by human observers as a benchmark. To evaluate

Figure 5.8: Example of ambiguity when choosing the best observation of the person. Both images display a nearly frontal view of the person. In the left one, the face is tilted somewhat more towards the camera, while in the right view the person appears slightly bigger. Both views can therefore be considered equivalent.

the accuracy of the helper camera selection, we compute the visual hulls **H** of the people in the scene based on the selected cameras using the shape-from-silhouette technique [Laurentini, 1994]. The voxel volume V^3 is set to $[0, 200) \times [0, 100) \times [0, 50) \subset \mathbb{N}^3$, where each voxel is a cube with edges of $0.04m$.

5.8.1 Principal View Quality

Which view provides the best observation of a person is in many cases not clearly defined, even for a human observer (see for example Fig. 5.8). For this reason, at each time instant up to three views can have the label of being a view that provides a good observation of the persons in a scene. If the scene is empty, none of the views is labeled as principal view.
In our experiments we distinguish between four scenarios, depending on the number of people in the scene.
Table 5.1 indicates the percentage of frames in which the view selected as key view by the methods based on face detection cues only (Section 5.5.1) and based on face detection and occupancy map cues (Section 5.5.2) were labeled as a principal one by a human observer. The total number of labeled frames is indicated in the second column. Comparing the results from the method based on face detection cues only and the method based on face detection and occupancy map cues, we conclude that including knowledge about position and velocity of the observed objects in the principle view determination provides a powerful means to boost the hit rate.
We can also observe in Table 5.1 that the principal view selection based on face detection and occupancy map cues achieves a good hit rate for small numbers of people in the scene, or in other words, that it very often selects the view which also a human observer judges as providing a good observation of the persons

Table 5.1: Percentage of frames in which the view selected by the method based on face detection cues only (Section 5.5.1) and the method based on face detection and occupancy map cues (Section 5.5.2 were labeled as a principal one by a human observer.

Scenario	# frames	Key as in Section 5.5.1	Key as in Section 5.5.2
1 persons	316	46	70
2 persons	297	51	71
3 persons	376	50	55
4 persons	262	51	53

in a scene. Correct and incorrect selections of the principal view mainly occur in bursts. Most correct selections are made when there is a view that clearly stands out as being the one providing the best view on the scene, also to a human observer. Errors arise during the transitions between such clear-cut cases.
For more persons, the hit rate drops. In these cases, determining the principal view becomes more ambiguous, as more than one camera might have a good frontal view of different persons. Adding information about the occupancy of the scene does not resolve this inherent ambiguity. The performance difference between principal view selection based on face detection only and based on face detection and occupancy map cues is therefore minimal. Selecting more than one key view would be a possible solution in this case.
A demo video illustrating principal view selection in a network of 10 cameras can be found online at [Tessens et al., 2008a].

5.8.2 Optimal Helper Camera Selection

In this section we assess the helper camera selection algorithm in its optimal (exhaustive search) implementation.
To evaluate how well the optimal helper camera selection observes the people in the scene from different viewing directions and consequently provides a good 3D shape reconstruction, we reconstruct the visual hull for each frame from the foreground *silhouettes* $\mathbf{F}_i$ of the selected camera subset $\mathbf{S}_n$, with $\mathbf{S}_n$ as in Eq. 5.8. We will denote this hull by $\mathbf{H}_{\mathbf{S}_n}$. The reconstruction of a person at a particular time instant can be seen in Fig. 5.9b for a selection of n=3 cameras. In Fig. 5.9c, the reconstruction of the person at the same time instant for a different selection of n=3 cameras is shown. Note that these FG silhouettes $\mathbf{F}_i$ are *not* available at the base station when the selection decision has to be made nor used in the actual method, only their approximate versions $\mathbf{F}_{i,\mathrm{sc}}$.
We determine at each time instant the number of voxels d_n that are different between the hull reconstructed from the selected subset and a benchmark hull. If the detected foreground silhouettes in the views are correct, the visual hull reconstructed from the whole set $\mathbf{C}$ of ten available cameras is the best possible hull we can reconstruct because it includes information from the highest num-

ber of different viewing angles. A missed foreground detection in one of the views has as an effect that some voxels are mistakenly considered unoccupied. Leaving out the faulty view avoids this, but also has as an effect that a lot of voxels are mistakenly considered occupied because the information of this viewing angle is lost. In this work we take the visual hull reconstructed from the whole set $\mathbf{C}$ of ten available cameras, denoted by $\mathbf{H_C}$, as the benchmark visual hull.

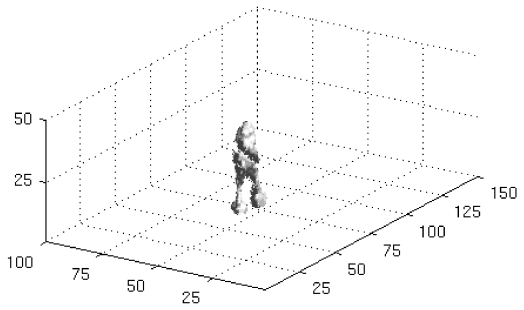
Fig. 5.9a shows an example of a visual hull $\mathbf{H_C}$ reconstructed from all cameras. We can clearly recognize the shape of a person in this visual hull. We can still discern the person in the visual hull reconstructed from three selected views in Fig. 5.9b and c, but we also observe ghost volumes. These are parts of the visual hull that do not represent real objects but result from an insufficient number n of used cameras. Ghost volumes can be seen as the 3D version of the ghost areas described in Section 5.6. The total ghost volume is smaller for the camera selection of Fig. 5.9c, but the quality of the reconstructed person is better for the camera selection of Fig. 5.9b, where you can, e.g., discern the legs of the person.

We wish to evaluate how well a subset of cameras performs in accurately reconstructing the shape of the person from the images transmitted to the base station, and not the ghost volumes. We assume that ghost volumes can be filtered out, e.g., based on temporal information or based on the occupancy map calculated from all scan-lines. To exclude the disturbing influence of ghost volumes on the evaluation of a camera subset, we only take differences between the visual hull reconstructed from the selected subset and the benchmark hull into account within $\mathbf{H_C^{filt}}$, which is the dilated version of $\mathbf{H_C}$:

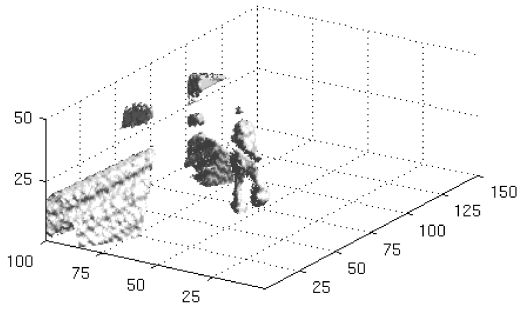
$$d_n = \sum_{\forall \mathbf{j} \in V^3} \left[\left(\mathbf{H_C^{filt}}(\mathbf{j}) \mathbf{H_{S_n}}(\mathbf{j}) \right) - \mathbf{H_C}(\mathbf{j}) \right]. \tag{5.10}$$

The dilation of $\mathbf{H_C}$ is performed by an image dilation in each plane parallel to the ground plane. The dilation is performed by iteratively dilating the image five times with a structuring element that is a square of 3×3 with the origin at its center. The visual hull $\mathbf{H_C}$ reconstructed from all cameras is plagued by ghost volumes as little as is achievable with the available cameras (compare for example the number of ghost volumes in Fig. 5.9a and Figs. 5.9b or c). Thus filtering with $\mathbf{H_C^{filt}}$ helps us to focus on the interesting objects in the scene. At the same time, due to the dilation operation, we still consider the whole object as reconstructed by the subset. The amount of extra volume within the filtered hull (in other words d_n) gives us an insight in how well the selected subset observes the persons in the scene from different directions and allows to accurately reconstruct their 3D shape. In the case of Fig. 5.9, d_n will thus be smaller for Fig. 5.9b than for Fig. 5.9c, because in Fig. 5.9b the quality of the reconstructed person is better.

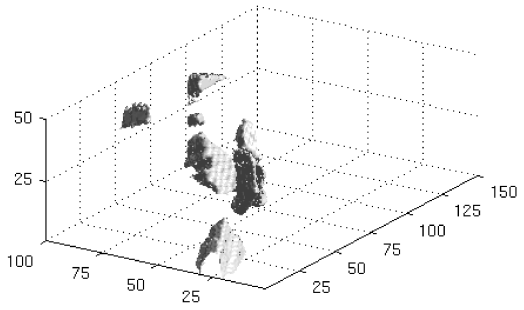
To select helper cameras, we start from a key camera and select additional views such that the 3D shape of the people in the scene can be reconstructed as accurately as possible. As the full 3D shapes are not available during the



(a) Visual hull reconstructed from all views.



(b) Visual hull reconstructed from three views, selected such that the quality of the reconstructed person is high. Observe the large ghost volumes to the left and behind the person.



(c) Visual hull reconstructed from three views, selected such that the total volume of the visual hull is low. Observe the poor quality of the reconstructed person despite the relatively small ghost volumes around the person.

Figure 5.9: Visual hull at a particular instant in time, reconstructed from (a) all views, (b) three views, selected such that the quality of the reconstructed person is high, and (c) three views, selected such that the total volume of the visual hull is low. We observe that minimizing the total amount of ghost volume does not necessarily lead to a better quality of the reconstructed person.

Table 5.2: Mean voxel difference d_n (Eq. 5.10) when helper cameras are selected starting from three key camera assignment methods (no key camera assigned, key camera assigned using only face detection cues, see Section 5.5.1, and using face detection and occupancy map cues, see Section 5.5.2) for four different scenarios. In the second row we indicate the total number of frames over which the average is calculated. The average voxel volume of $\mathbf{H_C}$ is shown in the third row. Rows 3-5 are the results for $n = 3$ and rows 6-8 for $n = 6$.

Scenario		1 person	2 persons	3 persons	4 persons
# frames		1629	2213	826	290
$\sum_{\forall \mathbf{j} \in V^3} \mathbf{H_C}(\mathbf{j})$/# frames		615.61	2450.99	4584.35	8079.53
$\hat{d}_3$	No key	842.70	2945.63	5364.68	7630.01
	Key as in Sect. 5.5.1	1204.29	3365.90	7349.20	10036.74
	Key as in Sect. 5.5.2	894.42	2755.85	5816.58	8917.38
$\hat{d}_6$	No key	298.60	1095.09	1476.38	1361.51
	Key as in Sect. 5.5.1	348.29	755.97	1326.16	1356.97
	Key as in Sect. 5.5.2	310.36	750.32	1306.34	1415.02

selection process, the occupancy map area as in Eq. 5.7 is our selection criterion. If cameras are selected with no prior assignment of a key camera, all possible combinations of n out of N cameras are valid and the camera subset that leads to the occupancy map with the smallest area is guaranteed to be found. This is not the case when we start our selection from a key camera, as the camera subset that leads to the occupancy map with the smallest area might not include the selected key camera and thus the 'optimal' subset might be excluded from the valid combinations (i.e., the combinations that contain the key camera). Note that the lack of prior key camera assignment drastically increases the computational burden of the algorithm and eliminates the guarantee that the view is selected that contributes most to the desired observation of the scene, i.e., that captures a frontal view of one or more people in the scene.

We compare the accuracy of the reconstructed visual hull when n cameras are selected without prior key camera assignment with n cameras selected using the proposed method of helper camera selection starting from a key camera. In Table 5.2, we list for three methods (helper cameras selected starting from no key camera assigned, key camera assigned using only face detection cues, see Section 5.5.1, and using face detection and occupancy map cues, see Section 5.5.2) the mean value of the number of different voxels d_n, denoted $\hat{d}_n$ over all frames of the sequences with a certain scenario, both for $n = 3$ and $n = 6$. The lower this number, the higher the quality of the observation with the selected camera subset. The number of frames available per scenario is indicated in the second row, and the average voxel volume of the benchmark hull $\mathbf{H_C}$ in the third row as a reference.

First of all, we observe that the number of voxels in the benchmark hull is of the same order as the mean voxel difference between this hull and the hulls reconstructed from a subset of cameras when $n = 3$. For $n = 6$ it is a fraction

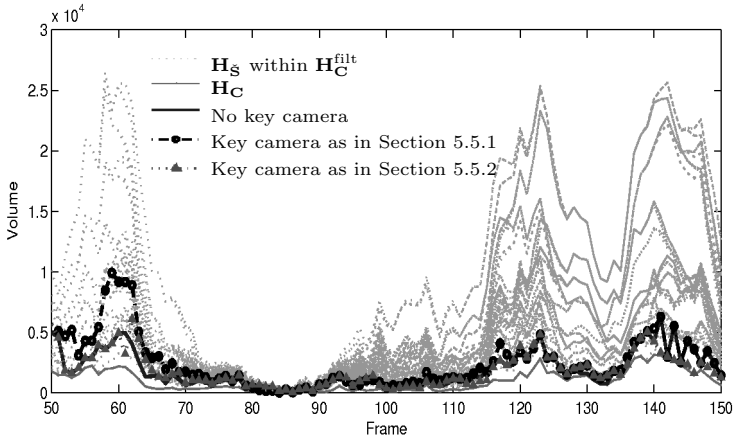
of this number. We conclude that for accurate 3D shape reconstruction, a sufficient number of cameras, e.g., six, needs to be selected. Mean voxel differences are smaller for helper camera selection with the principal view determined using face detection and occupancy map cues (Section 5.5.2) than for selection where the choice of the key camera is only based on the face detection scores (Section 5.5.1). Helper camera selection with a key camera determined using face detection and occupancy map cues (Section 5.5.2) yields similar results as helper camera selection without prior key camera assignment. Occasionally, it even outperforms that method. This is possible because the occupancy map area is only an approximation of the 3D shape of the people present in the scene. The subset of cameras that minimizes the occupancy area does not necessarily lead to the solution that gives the best visual hull.

To illustrate the necessity of view selection, we show in Figure 5.10 the selection performance when selecting $n = 3$ cameras from 10. For a representative sequence of the scenarios with one and four persons, we plot per frame the volume (in number of voxels) of the visual hull from all possible subsets $\check{\mathbf{S}} \subset \mathbf{C}$, with $|\check{\mathbf{S}}| = n = 3$, contained within $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$ (green dotted lines). Note that there are more possible subsets $\check{\mathbf{S}}$ than there are candidate subsets for which $K \in \mathbf{S}$. As a reference, for each frame the number of voxels of the benchmark visual hull $\mathbf{H}_{\mathbf{C}}$ is also indicated (solid magenta line).

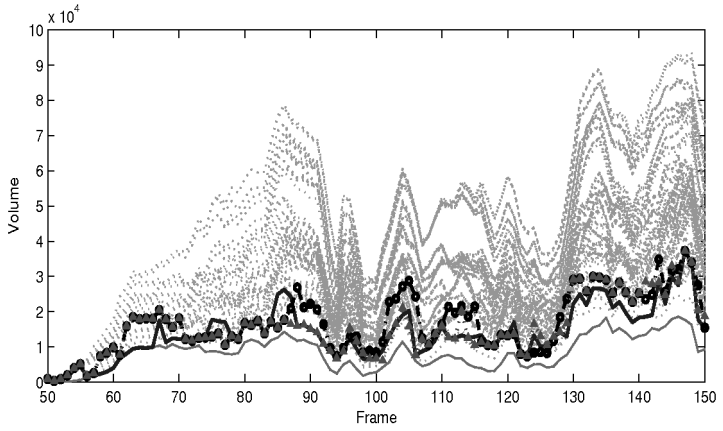
The number of voxels per frame of the visual hull reconstructed from a selected set of cameras S_3 (obtained as in Eq. 5.8) within $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$, are drawn as the thicker lines. The solid blue line indicates the camera selection when no key camera is assigned. The dash-dotted black line with round markers is the subset selection with the key camera selection based on face detection only (as in Section 5.5.1). The dotted red line with triangular markers is the camera subset selection with the principal camera selection based on face detection and occupancy map cues (Section 5.5.2).

This graph indicates that, regardless of how the principle view is determined, the optimal helper camera selection method of Section 5.6 selects from all possible subsets one that is always close to the best possible subset. Indeed, the curves of all methods are close to the lower envelope of the curves of all possible subsets. A second observation is that the curves corresponding to camera selection without prior key camera assignment and with principal view determination using face detection and occupancy map cues (Section 5.5.2) mostly coincide and that both methods lead to lower visual hull volumes than the same selection method but with the principal view determined using face detection cues only (Section 5.5.1).

Figure 5.11 shows a visual example of the selection of $n = 3$ cameras from 10 using optimal helper camera selection with the principal view determined using face detection and occupancy map cues (Section 5.5.2). We display the views of all the cameras $C_1, \ldots, C_{10}$. To give an insight into the system setup, we depicted in the bottom-right corner a top view of the scene, which indicates the relative positions of the ten cameras and the person in the scene. The selected key camera C_3 is marked by a magenta bounding box. This camera was chosen



(a) 1 person



(b) 4 persons

Figure 5.10: Selection performance for 100 frames of a representative sequence of the scenarios with (a) one and (b) four persons. The number of cameras in the subset is $n = 3$. For each frame, we plotted the volume (in number of voxels) of the visual hulls reconstructed from all possible subsets $\check{\mathbf{S}}$ within $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$, (green dotted lines), of the benchmark hull $\mathbf{H}_{\mathbf{C}}$ (solid magenta line) and of the hull $\mathbf{H}_{\mathbf{S}_3}$ within $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$. The camera set $\mathbf{S_3}$ has been selected using different principal view selection strategies: no key camera assigned (solid blue line), key camera assigned using only face detection cues (dash-dotted black line with round markers), and using face detection and occupancy map cues (dotted red line with triangular markers). The lower this volume, the less redundant the selected views.

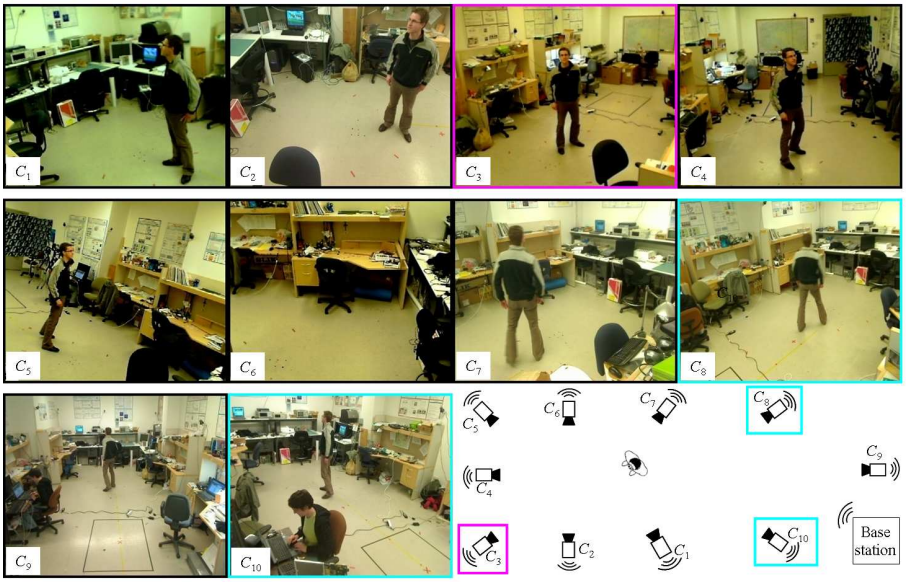


Figure 5.11: Example of the selection of 3 out of 10 cameras. The views of the 10 cameras $(C_1, \ldots, C_{10})$ are shown. In the bottom-right corner, we depicted a top view of the scene which shows its geometry and the positions of the cameras and person. The selected key camera C_3 is marked by a magenta bounding box and the helper cameras C_8 and C_{10} by a cyan bounding box.

to be the key camera by the principal view determination method of Section 5.5.2. The helper cameras C_8 and C_{10} are marked by a cyan bounding box and are selected using the optimal method of Section 5.6. We can observe from the displayed views that the selected principle view contributes most to the observation of the person, while the helper cameras complete the observation. Note that the person sitting at the desk in camera view C_{10} is operating the start and the end of the capturing of the sequence. This person is immobile during the whole sequence and is assumed to be part of the background.
A demo video illustrating the application of principal view selection in camera selection can be found online at [Tessens et al., 2008a].

5.8.3 Greedy vs. Optimal Helper Camera Selection

In this section we compare the accuracy of the greedy helper camera selection with its optimal counterpart. The helper camera selection starts from a key view selected using face detection and occupancy map cues (Section 5.5.2).
We first reconstruct the 3D visual hull $\mathbf{H}_{\hat{\mathbf{S}}_n}$ based on the foregrounds $\mathbf{F}_i$ from the cameras in the greedy solution set $\hat{\mathbf{S}}_n$. We also reconstruct the 3D visual hull $\mathbf{H}_{\mathbf{C}}$ based on the foreground $\mathbf{F}_i$ from all cameras $C_i \in \mathbf{C}$. The visual hull $\mathbf{H}_{\mathbf{C}}$ is considered the correct 3D shape of the objects and serves as the

Table 5.3: Mean voxel difference for the optimal and greedy selection methods for four different scenarios and for n=6 selected cameras. In the second column we indicate the total number of frames over which the average is calculated. The average voxel volume of $\mathbf{H_C}$ is shown in the third column.

Scenario	# frames	average voxel volume	$\hat{d}_6^{\text{optimal}}$	$\hat{d}_6^{\text{greedy}}$
1 person	1629	615.61	310.36	342.15
2 persons	2213	2450.99	750.32	790.72
3 persons	826	4584.35	1306.34	1369.96
4 persons	290	8079.53	1415.02	1508.18

performance baseline. Finally, we reconstruct the 3D visual hull $\mathbf{H}_{\mathbf{S}_n}$ based on the foreground $\mathbf{F}_i$ from the cameras in the optimal solution set $\mathbf{S}_n$.

Given the reconstructed visual hulls at each time instance, we calculate the number d_n^{greedy} of voxels that are different between the greedy solution visual hull $\mathbf{H}_{\hat{\mathbf{S}}_n}$ and the benchmark visual hull $\mathbf{H_C}$ within $\mathbf{H}_{\mathbf{C}}^{\text{filt}}$. For the optimal solution, we also calculate the number of different voxels and denote it by d_n^{optimal}.

In Table 5.3, we compare for the greedy and optimal methods the mean value of the number of different voxels, denoted by $\hat{d}_n^{\text{greedy}}$ and $\hat{d}_n^{\text{optimal}}$, over all frames of the sequences with a certain scenario. The lower this number, the higher the quality of the observation with the selected camera subset. The number of frames available per scenario is indicated in the second column, and the average voxel volume of $\mathbf{H_C}$ in the third column. In these experiments, $n = 6$ cameras were selected among 10 cameras, and in each time frame the selection status of at least $u = 2$ cameras was reevaluated. We observe that the optimal and greedy methods yield similar results.

To measure the computation time gained by using the greedy method instead of the optimal one, we performed experiments on an AMD Athlon 64 3400+ 2.40 GHz processor using the SSE (Streaming SIMD extensions) instruction set. The performed computations were floating-point computations, and both methods are implemented in Matlab code. Averaged over 30 frames of the test set, the optimal method took 2.26 s to execute, whereas the greedy method required 0.39 s. In the optimal method, for each frame $\binom{9}{5} = 126$ candidate camera subsets need to be checked to determine the optimal set. The greedy method checks 18 camera subsets per frame. The ratio $126/18 = 7$ is of the same order of magnitude as the experimentally measured time ratio between the two methods ($2.26/0.39 = 5.8$).

5.8.4 Reduction of Delay

When the advanced operation scheme (Fig. 5.7) is applied, the selection decision at time instant t is based on observations of the previous time instant $t - 1$. In order to evaluate the impact of this shift on the accuracy, we pro-

Table 5.4: Mean voxel difference for the optimal and greedy selection methods with one frame delay. In the second column we indicate the total number of frames over which the average is calculated. The average voxel volume of $\mathbf{H_C}$ in shown in the third column.

Scenario	# frames	average voxel volume	$\hat{d}_6^{\text{optimal}}$	$\hat{d}_6^{\text{greedy}}$
1 persons	1619	619.41	316.28	368.23
2 persons	2203	2459.22	780.91	826.29
3 persons	822	4606.66	1344.37	1410.31
4 persons	289	8107.48	1422.11	1522.69

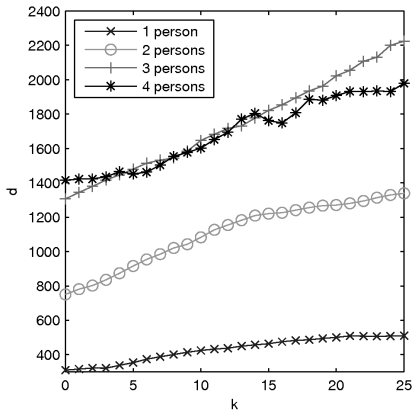


Figure 5.12: Mean voxel difference for the optimal selection method for four different scenarios as a function of the delay k between observation and selection decision. $n = 6$ cameras were selected.

cess the observations in a similar way as in the previous section. Only now, at time instant t the foreground silhouettes $\mathbf{F}_i$ from which the visual hull was reconstructed for accuracy evaluation were selected based on the selection of the previous time instant $t - 1$.

The experimental results are shown in Table 5.4 where $n = 6$ cameras were selected among 10 cameras, and in each time frame the selection status of at least $u = 2$ cameras was reevaluated. Comparing Tables 5.3 and 5.4, the accuracy is comparable in both cases. We conclude that the introduced delay has almost no impact on the performance.

We also investigated the delay impact on the performance when the delay k between observation and selection decision is more than one frame. The performance over different delays k is plotted in Fig. 5.12. It can be observed that delays of up to 5 frames result in only a minor drop in quality. Therefore, we can further reduce the data transmission by transmitting the scan-lines every k frames instead of all frames. In other words, we use the same selection results for every k frames for small k.

5.9 Conclusion

In this chapter, we have presented a method to determine which sensor subset in a smart camera network has the best view on the persons in a scene and allows to reconstruct their 3D shape as accurately as possible. The algorithm consists of two types of processes. The *distributed* processes run on the smart cameras themselves and strongly reduce the amount of data that needs to be sent over the network to the base station to a couple of tens of bits per node. At the base station the *central* camera selection takes place. In order to choose an appropriate key camera, this algorithm takes into account the number of faces detected by each of the cameras, and the velocity and positions of the objects relative to the viewing direction and viewing angle of the cameras. This principal view can be complemented with additional views that complete the observation and that allow to reconstruct the 3D shape of the people in the scene. To select these additional views we use the occupancy map as a crude 2D shape approximation of the people in the scene.
Experimental results on human-labeled sequences show that the selected principal view is equal to the view selected by a human observer in a high number of cases for a limited number of people in the scene. Also, we showed that this view together with the additional views gives a good approximation of the 3D shape compared to the best achievable 3D shape with the selected amount of cameras. Additionally, it is shown that the principal camera selection is a good starting point for the selection of additional views, since it greatly reduces the computational complexity, while still allowing the reconstruction of the 3D shape of the objects to be almost as accurate as in the optimal subset selection case (without principal view determination).
Moreover, a greedy camera selection algorithm was proposed for real time network operation. We used 3D shape reconstruction to compare the proposed greedy algorithm and optimal selection algorithm. Experimental results showed that the proposed algorithm provides a performance very close to the optimal results. Also, two different network operation protocols were proposed. The first scheme aims to improve the sensor observation frequency and the second scheme decreases the delay between view observation and image transmission. Experimental results verified that the proposed protocols improve observation frequency and latency without degrading much the performance of the 3D shape reconstruction.
A possible improvement of the proposed principal view selection method is to take dynamic occlusions into account. These occur when several persons are present in the scene, and one person blocks the view of a camera on another person. Such an improvement can be achieved by basing the assessment of the visibility of an object in a camera view (see Section 5.5.2) not only on the static viewing range of the camera, but also on the dynamic scene configuration. A drawback of this procedure is that the calculation of the visibility becomes computationally much more demanding. The method for helper camera selection naturally takes into account occlusions.

6

Camera Contribution Quantification for Sensor Selection

As discussed in the previous chapter, an important challenge in smart camera networks with correlated views is keeping data redundancy under control without discarding useful information. We propose to do this by selecting a limited number of cameras for each network task and to process information only on these cameras and transmit data only between these.

A crucial component in an effective camera selection system is quantifying the contribution of one or more cameras to the accomplishment of a task. This allows to appropriately allocate available network resources such that the best possible task performance is achieved. The contribution of a camera set depends on the observation perspective of the camera(s) and on the scene configuration, which is subject to change over time. In the case of view-correlated nodes, the event of interest may be simultaneously observed by several sensors, but not all cameras are equally suited to perform the task at hand.

In this chapter we introduce a unifying approach to integrate quality of view measures, such as, e.g., the ones for observability discussed in the previous chapter, in a criterion founded on generalized information theory. The proposed criterion is not limited to a specific type of task and can be applied to a wide range of vision problems. As a proof of concept, we use it for camera selection in a network in which multiple targets are tracked.

The work presented in this chapter has been performed in collaboration with my colleague Marleen Morbee and therefore some of the concepts presented here also appear in her PhD thesis. However, the two theses elaborate on different aspects and applications of the proposed technique. In this work, we propose and thoroughly study a method of quantifying the quality of one or more cameras to the accomplishment of a task and its effect on the final quality of the accomplished task. This method is used as a tool to select cameras for a single task at a time. In my colleague's dissertation, the potential of

this framework for task assignment is explored. More precisely, a technique to distribute *several* tasks over the network cameras in an optimal way with respect to the achievable frame rate are proposed in her thesis. Solutions to the related optimization problem are also investigated.
The rest of this chapter is organized as follows. In Section 6.1 we discuss related work. A formal problem formulation is provided in Section 6.2. Section 6.3 introduces the proposed camera set suitability value to quantify the contribution of one or more cameras to the accomplishment of a task, which is applied to tracking in Section 6.5. Results and conclusions are presented in Sections 6.6 and 6.7 respectively.

6.1 Related Work

Sensor selection in networks of range restricted sensors is a well studied problem. An overview can be found in [Rowaihy et al., 2007]. Many methods focus on localization and tracking applications. [Chu et al., 2002; Zhao et al., 2002] introduced information-driven sensor querying. In [Ertin et al., 2003] and [Liu et al., 2003], the maximum mutual information principle is proposed as a criterion for sensor selection. This principle has been reworked into an entropy-based heuristic in [Wang et al., 2005]. [McIntyre and Hintz, 1996; Schmaedeke and Kastella, 1998] also use entropy to value sensors.
Cameras have received special attention in literature because they are not range restricted in the classical sense that their sensing range limit only depends on the distance to the sensor. Camera selection for tracking and localization has been studied in [Denzler et al., 2003; Ercan et al., 2006; Gupta et al., 2007; Isler and Bajcsy, 2005; Pahalawatta and Katsaggelos, 2004; Snidaro et al., 2003; Sommerlade and Reid, 2008]. These methods will be discussed in more detail in Section 6.5. An overview of camera sensor planning for robustly detecting object features can be found in [Tarabanis et al., 1995]. View selection for object recognition using an information-theoretic criterion was proposed in [Denzler and Brown, 2002]. View selection for optimal observability is treated in [Daniyal et al., 2010; Jiang et al., 2008; Kelly et al., 2009; Li and Bhanu, 2009; Park et al., 2006; Tessens et al., 2008b]. These methods assign view quality measures based on activity level, size and centrality of the object in the view, etc. [Vázquez et al., 2003] uses viewpoint entropy calculated from a polyhedral scene model to select a minimal set of views for image-based rendering. In [Yang et al., 2004] cameras are tasked to determine the occupied space in the scene while minimizing the number of active cameras.
In this work we present a novel, more general framework for camera selection. We introduce a unifying approach to integrate quality of view measures in a criterion founded on generalized information theory. Because this criterion is derived from the Dempster-Shafer theory of evidence [Dempster, 1968; Shafer, 1976], it naturally handles common problems in camera networks such as partial or incomplete visibility of objects or events due to limited fields of view or occlusion.

6.2 Problem Formulation

Consider a network of N cameras i, $1 \leq i \leq N$, potentially involved in the execution of a task. Let S denote the set of cameras actually selected to perform this task. Some camera sets are more suited for the task than others. For example, if a person is mostly occluded in one camera view, that camera may be less useful in determining the person's position. To express this property, we associate a *suitability value* $v(S)$ with each set.

Let Γ denote the set of all possible selections S. It is often useful to impose restrictions on the camera sets we consider, such as limiting the number of cameras in the set. Let Γ' be the restricted set of admissible selections. The optimal selection S^* is the set $S \in \Gamma'$ that maximizes the suitability value $v(S)$:

$$S^* = \arg\max_{S \in \Gamma'} v(S). \tag{6.1}$$

The main goal of this work is to define and study an effective camera set suitability value $v(S)$. This is the topic of the next section. The algorithm for finding the optimal set based on this criterion will be discussed in Section 6.4.

6.3 A Generalized Information-Theoretic Suitability Value

6.3.1 Quantification of Task-Related Information

In a camera sensor network all tasks basically involve information gathering. The more information relevant to a task a camera set can acquire, the more suited it is to perform this task. The set containing all cameras can always gather the maximal amount of task relevant information available in the network. However, out of computational and communication efficiency reasons, it is useful to select a smaller camera set for a task. In this work, the camera set size is limited by manually fixing an upper limit for each experiment. Ideally, the set size would be dynamically adapted according to a computational and/or communication cost criterion. This dynamic adaptation of the camera set size is studied in the PhD thesis of my colleague Marleen Morbee.

In this thesis we focus on quantifying the task-related information contained in the observations of a camera set, which is a key issue in designing a value $v(S)$ which reflects the suitability of the set S for the task at hand.

In information theory, information is specified in terms of the entropy associated with a random variable. In this work we therefore define a camera network task more precisely as discovering the value of a realization of a random variable X using a subset of cameras. E.g., in the tracking example treated in Section 6.5, X designates within which range of ground positions the target is located.

6.3.2 Classical Information-Theoretic Approach

Suppose X can assume any of the N_X values in the finite set $\{x_1, x_2, \ldots, x_{N_X}\}$. Without any observations only prior knowledge about the probability distribution of X is available. Let $p(X)$ denote the prior distribution of X. Let us now denote the observations of a camera set S as O_S. Given the observations from all cameras in the set, the probability distribution of X can be updated to $p(X|O_S)$.
The uncertainty associated with the value of X, and hence the information content of the observations O_S, is expressed by the entropy $H(X|O_S)$. In [Denzler et al., 2003; McIntyre and Hintz, 1996; Schmaedeke and Kastella, 1998; Sommerlade and Reid, 2008; Wang et al., 2005; Zhao et al., 2002] this is the criterion used for sensor selection. To assess how suited a camera set S is for the task, we must evaluate $H(X|O_S)$ and consequently $p(X|O_S)$.
To determine $p(X|O_S)$ we apply Bayes' rule:

$$p(X|O_S) = \frac{p(O_S|X)p(X)}{p(O_S)}. \tag{6.2}$$

We assume the observations of the different cameras in the set S are conditionally independent from each other, meaning that for a given state of the variable X the observation process of each camera is an independent process. When the influences of general conditions on all camera observations simultaneously (e.g., scene lighting changes) are ignored, this is a reasonable assumption. $p(O_S|X)$ can then be obtained as

$$p(O_S|X) = \prod_{i \in S} p(O_i|X). \tag{6.3}$$

Combining Eqs. 6.2 and 6.3 we obtain

$$p(X|O_S) = \frac{p(X)}{p(O_S)} \prod_{i \in S} p(O_i|X). \tag{6.4}$$

The probabilities $p(O_i|X)$ relating the camera observations to the state of X can be modeled based on the physical properties of the cameras. As mentioned previously, $p(X)$ must be specified based on prior information. This must be either modeled, estimated from training data or be determined empirically.
It is interesting to note at this point that in a camera network it frequently occurs that a sensor can only yield partial information or even no information at all about a task (represented by X). This happens when all or part of the events relevant to the task are occluded or occur outside of the camera viewing frustum. Consider for example that the task X is to determine the color of a person's shirt and trousers. If only the person's shirt is visible to the camera, it can yield only partial information about the task X. If the person is not visible at all, this camera produces no information about the task. In these cases classical probability theory has to resort to priors which can be difficult to obtain, and if badly modeled, introduce misleading information in the system.

Imprecise probability theory provides an extension to its classical counterpart and is able to explicitly represent the absence or incompleteness of information using lower and upper probabilities. A well known mathematical theory that implements the concept of imprecise probabilities through belief functions is the Dempster-Shafer (DS) theory of evidence [Dempster, 1968; Shafer, 1976]. A brief overview of this theory has been presented in Section 4.2. In what follows, we use this theory to obtain the desired suitability value $v(S)$, after which we make a comparison with the classical information-theoretic approach.

6.3.3 Generalized Information-Theoretic Approach

The concepts from information theory as they were introduced for classical probability theory cannot be straightforwardly transferred to imprecise probability theory. To this end, generalized information theory was developed [Klir, 1991]. In generalized information theory, information is defined in terms of uncertainty reduction.

Uncertainty comprises several aspects: *probabilistic uncertainty* is generated by the randomness of a system, whereas *unspecificity* arises when there is evidence for a proposition that aggregates several elementary propositions but not for the elementary propositions themselves. Unspecificity can be mathematically expressed by the generalized Hartley (GH) measure [Abellan and Moral, 2000]:

$$GH(\mathrm{m}) = \sum_{\mathrm{A} \subseteq \Omega} \mathrm{m(A)} \log_2 |\mathrm{A}|, \tag{6.5}$$

where $|A|$ denotes the cardinality (number of elements) of the set A and $m(A)$ is the basic belief assigned to the hypothesis A (see Section 4.2). If $|A| = 1$, i.e., if A is an elementary subset, there is no unspecificity and $m(A) \log_2 |A| = 0$. As the hypothesis A aggregates more and more elementary propositions, $\log_2 |A|$ gets bigger. A body of evidence with only basic belief assigned to the elementary propositions does not contain any unspecificity uncertainty and $GH(\mathrm{m}) = 0$. For a body of evidence in which $m(\Omega) = 1$, their is no specific evidence at all and $GH(\mathrm{m}) = \log_2 |\Omega|$, which is the maximal uncertainty that can be present in a body of evidence.

The generalization of Shannon (GS) entropy to characterize probabilistic uncertainty is defined through an *aggregated uncertainty*, AU, which unites both unspecificity and probabilistic uncertainty: $GS(\mathrm{m}) = AU(\mathrm{m}) - GH(\mathrm{m})$. To define the AU present in a BBA m, we first define $\mathcal{D}$, a set of probability mass functions $p(\omega)$ on the finite set Ω that are *consistent* with m, as follows [Klir and Wierman, 1999]:

$$\mathcal{D} = \{p(\omega) | \omega \in \Omega, p(\omega) \in [0,1], \sum_{\omega \in \Omega} p(\omega) = 1, \sum_{B \subseteq A} m(B) \leq \sum_{\omega \in A} p(\omega) \text{ for all } A \subseteq \Omega \text{ and B } \subseteq A\}. \tag{6.6}$$

The AU is defined as [Klir and Wierman, 1999]

$$AU(\mathrm{m}) = \max_{\mathrm{p}\in\mathcal{D}} \left[-\sum_{\omega\in\Omega} \mathrm{p}(\omega) \log_2 \mathrm{p}(\omega) \right]. \tag{6.7}$$

It is the maximal Shannon entropy of any probability mass function $p(\omega)$ within $\mathcal{D}$. An efficient algorithm for computing Eq. 6.7 is available in [Klir and Wierman, 1999].
In what follows we will use the aggregated uncertainty, which joins probabilistic uncertainty and unspecificity, to characterize the uncertainty in a BBA m.
Applying our definition of a network task of Section 6.3.1 to the DS theory, we formulate each task as assessing the validity of a set of elementary propositions that form a frame of discernment Ω. Each camera set S gathers evidence about the propositions within the power set 2^{Ω}, leading to a BBA m_S. The smaller the aggregated uncertainty in m_S, the more informative the observations of the set and the better suited this set is for the task. Let $|\Omega|$ denote the number of elementary propositions in the frame of discernment Ω. The maximal AU in a BBA m_S equals $\log_2 |\Omega|$. It is for example obtained when $m_S(\omega) = 1/|\Omega|, \forall\omega \in \Omega$. We define our camera set suitability value for a task as

$$v(S) = 1 - \frac{AU(\mathrm{m_S})}{\log_2 |\Omega|}. \tag{6.8}$$

A camera set that is very suitable for a task will thus have a suitability value close to one, whereas unsuitable sets will have a value of zero.
In the next section we apply the proposed camera set suitability value to camera selection in a network in which multiple persons are tracked. First we discuss the relationship of this value with measures from classical information theory.

6.3.4 Comparison with Classical Information-Theoretic Approach

Let us call a BBA in which $\sum_{\omega\in\Omega} m(\omega) = 1$ a Bayesian BBA. In this case the proposed suitability value reduces to a well known information-theoretic measure. Indeed, the generalized Hartley uncertainty is zero in this case:

$$AU(\mathrm{m_S}) = GS(\mathrm{m_S}) = -\sum_{\omega\in\Omega} \mathrm{m_S}(\omega) \log_2 \mathrm{m_S}(\omega). \tag{6.9}$$

In other words, the aggregated uncertainty in the BBA m_S in this case coincides with the Shannon entropy of the random variable X representing the task. Recall that $H(X|O_S)$ is the Shannon entropy of X after using the observations of all cameras in the set S. Substituting Eqs. 6.8 and 6.9 in Eq. 6.1 we have

$$S^* = \arg\max_{S\in\Gamma'} \big(1 - \frac{H(X|O_S)}{\log_2 |\Omega|}\big) = \arg\min_{S\in\Gamma'} H(X|O_S). \tag{6.10}$$

This is the minimal entropy criterion used for sensor selection for tracking in [Denzler et al., 2003; McIntyre and Hintz, 1996; Schmaedeke and Kastella, 1998; Sommerlade and Reid, 2008; Wang et al., 2005; Zhao et al., 2002].
Now suppose we want to enlarge the camera set S by one camera and we look for the most informative one. Eq. 6.1 then simplifies to

$$S^* = \underset{i \in [1,N] | S \cup \{i\} \in \Gamma'}{\arg\max} v(S \cup \{i\}). \tag{6.11}$$

As $H(X|O_S)$ is independent of the added camera i

$$S^* = \underset{i \in [1,N] | S \cup \{i\} \in \Gamma'}{\arg\max} \big(H(X|O_S) - H(X|O_{S \cup \{i\}})\big). \tag{6.12}$$

In information theory, $H(X|O_S) - H(X|O_{S \cup \{i\}})$ is the mutual information $I(O_i; X|O_S)$ measuring the reduction in uncertainty about X when the observing camera set is enlarged from S to $S \cup \{i\}$. This criterion has been used to select sensors in a tracking context [Ertin et al., 2003; Liu et al., 2003], but also to select views for object recognition [Denzler and Brown, 2002].
The camera set suitability value proposed in this work is more general than the classical information-theoretic entropy or mutual information criteria as it can also handle non-Bayesian belief structures. The absence of (complete) information, which frequently occurs in a camera network, can be easily incorporated in the DS evidence structure, but is more difficult to handle in a Bayesian context without prior knowledge. For instance, if only a person's arm is visible in a camera, some information about the person's position can be deduced from this, but a lot of localization uncertainty will remain. This is not easily modeled using Bayesian reasoning, whereas the DS based formulation of such partial knowledge is quite natural. We will illustrate this strength in Section 6.5. First we will discuss the solution of the optimization problem of Eq. 6.1.

6.4 Greedy Optimization

As the set S can only assume a discrete number of values, Eq. 6.1 is a discrete constrained optimization problem. An exhaustive search over all possible values of S guarantees that the optimal solution of Eq. 6.1 is found. In a network of N cameras, Γ contains 2^N possible camera subsets. Only sets in $\Gamma' \subseteq \Gamma$ need to be evaluated. The nature of the imposed constraints will dictate the exact number of elements in Γ', but for large networks with many cameras, an exhaustive search quickly becomes unacceptably slow. Assume for example that a set of 5 cameras needs to be selected out of 10 cameras. In this case, $\binom{10}{5} = 252$ camera sets need to be evaluated. Better solution methods are proposed in the domains of integer programming and combinatorial optimization [Tsang and Voudouris, 1998].
In this work, we adopt a greedy optimization heuristic to solve the optimization problem of Eq. 6.1. We start from an empty camera set. In each iteration, we

Input: Observations about X of all cameras
Output: S^* (the optimal set to perform a task)

$S \leftarrow \emptyset$, $S^* \leftarrow \emptyset$, $V_{max} \leftarrow 0$, $a \leftarrow$ true
while a **do**
    $a \leftarrow$ false (continue loop only if allowed sets
that increase the suitability value can be formed)
    for $i = 1$ to N **do**
        $S \leftarrow S \cup \{i\}$
        if $S \in \Gamma'$ **then**
            Construct m_S based on observations about X of cameras in S
            $V \leftarrow 1 - \frac{AU(\mathrm{m_S})}{\log_2 |\Omega|}$, the suitability value
            if $V > V_{max}$ **then**
                $a \leftarrow$ true
                $S^* \leftarrow S$
                $V_{max} \leftarrow V$
            end if
        end if
        $S \leftarrow S \backslash \{i\}$
    end for
    $S \leftarrow S^*$
end while

algorithm 4: Greedy Optimization.

identify the sensors which lead to an admissible camera set $S \in \Gamma'$ when added to the current set. Among these, we select the one which increases the set suitability value most. This process is iterated until none of the remaining sensors that lead to an admissible camera set $S \in \Gamma'$ increase the set suitability value. Algorithm 4 shows the pseudo-code of this optimization.
If there is more than one network task, we search for an optimal set of cameras for each task independently of the other task(s). Performing this optimization jointly offers interesting possibilities to distribute the tasks among the cameras according to some practical criteria (such as equal spread of load, or minimization of the required communication) while controlling the associated changes in the quality with which the tasks are performed. This matter is not treated in this work.

6.5 Application to Camera Selection for Tracking

In this application example we consider a multi-camera system that observes a scene containing multiple persons. The goal of the system is to track the persons, i.e., to determine their position on the ground plane at each time instant.

6.5.1 Related Work

As already mentioned in Section 6.1, camera selection for tracking and localization has been studied before. [Isler and Bajcsy, 2005] approaches camera selection for localization as a geometric problem and minimizes the uncertainty area obtained by intersecting the reprojection cones of cameras. In [Pahalawatta and Katsaggelos, 2004] the information utility of a camera is characterized by the trace of the covariance matrix of the posterior distribution of the object state in an Kalman filtering framework. Also in the context of tracking using a Kalman filter, the authors of [Denzler et al., 2003] adopt an information-theoretic approach to control the focal length of a camera based on the uncertainty associated with the target position. This is done by minimizing the expected entropy of the state conditioned on the observation. In [Sommerlade and Reid, 2008] this approach is extended to account for the appearance of new targets, leading to an active scene exploration system. The authors in [Snidaro et al., 2003] base their view selection on a quality measure for the appearance of a tracking target in an image. The previous methods cannot effectively take occlusion into account - a frequent problem in tracking - without significant reformulation of the algorithms.

In [Gupta et al., 2007] cameras are selected especially to avoid occlusion (and confusion - people being visible behind the target) in a localization task. This is achieved by determining the probability of visibility of each part of a person model in each camera based on probabilistic estimates of the poses of other people in the scene. This determines the order in which the object positions and poses should be inferred. [Ercan et al., 2006] handles occlusion in a similar way, albeit in 2D, by weighting error contributions with the probability of occlusion, calculated from the prior of the occluding object. Furthermore an essentially geometric approach is followed to minimize the localization error of an object given its prior position distribution and the camera noise parameters.

The limited fields of view of cameras are usually dealt with by ignoring the contributions of cameras in which the target is not completely visible [Denzler et al., 2003; Isler and Bajcsy, 2005; Snidaro et al., 2003; Sommerlade and Reid, 2008]. This method discards valuable information, as a large part of the target may still lie inside the camera viewing frustum. In the methods of [Ercan et al., 2006; Gupta et al., 2007] the limited fields of view of cameras are naturally dealt with by determining the visibility of objects in the cameras. However, in [Ercan et al., 2006] the camera foreground images are vertically summed and thresholded prior to determining the visibility of objects, which makes it impossible to differentiate between full and partial visibility at the horizontal image boundaries because an object will be classified as visible even if a large part of it is projected below or above the image. This method can therefore only handle partial visibility at the vertical boundaries of the camera images. E.g., the information of a camera in which only a person's head is visible is not valued less than a camera in which the whole person is visible. This is especially a problem in set-ups where the cameras are close to the observed objects.

Our approach differs from the existing literature in several ways. The camera selection method is suited to be used in combination with a tracker based on particle filtering. Particle filters are powerful tools that can model multiple hypotheses, making them robust, and that can handle non-linear motion and noise models. Moreover, the proposed method selects cameras by measuring the impact of the quality of the appearance of objects in the camera image on the localization uncertainty. This approach links a generalized information-theoretic criterion for camera selection (similar to [Denzler et al., 2003; Sommerlade and Reid, 2008]) with taking the impact of occlusion and confusion of multiple targets on the localization into account (similar to [Gupta et al., 2007]). As our selection criterion is founded on the Dempster-Shafer theory of evidence, problems of absent or incomplete information (partial or complete invisibility due to limited fields of view, occlusions) are naturally handled.

6.5.2 Camera Set Suitability Value for Tracking

We consider tracking each person as a separate network task. For each person we determine at each time instant which camera set is most suited to track it. We do this by solving Eq. 6.1 using the camera set suitability value of Eq. 6.8. Because for each person the optimization is performed independently of the other tracking tasks, the camera sets selected for the different persons may or may not overlap.

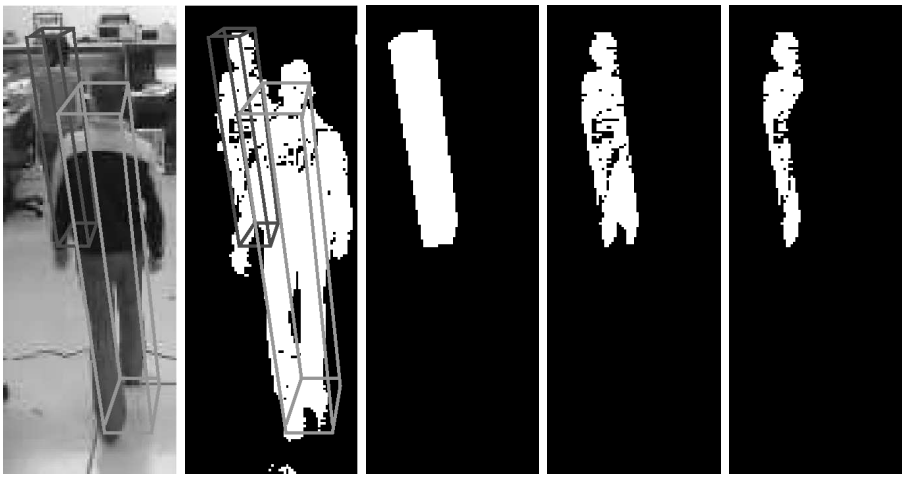
To be able to use the camera set suitability value of Eq. 6.8, we reformulate the tracking task as determining the validity of an exhaustive and mutually exclusive set of hypotheses. We do this by dividing the ground plane in the vicinity of the tracked person in $G-1$ discretization cells (we will explain how in Section 6.5.5.1). There is also a part of the ground plane area in which we cannot gather observations (the area outside the viewing range of all cameras in the network), or in which we do not expect the tracked person to be. This part of the ground plane area makes up another cell X_G. The frame of discernment Ω is made up of the elementary propositions $\omega_g = \{x \in X_g\} \in \Omega$, $g = 1 \dots G$. In other words, the elementary proposition ω_g is the hypothesis that the position x of the tracked person lies in the cell X_g. The inclusion of the cell X_G in Ω makes the set of hypotheses exhaustive. A camera set that can with low uncertainty locate the person in one of these cells, without being hampered by limited fields of view, occlusions, heavily cluttered foreground segmentations or other error sources, will be assigned a high suitability value. A set that is not certain in which cell the person is, gets a low suitability value.

Note that the discretization of the ground positions is only necessary to determine a suitable camera set to perform the tracking task. The tracking as such can be performed with one of the many existing multi-camera multi-people tracking algorithms. This tracking does not need to operate on discretized ground positions. In this work we opt for the tracker in [Munoz-Salinas et al., 2009], which is an extension of the Bayesian particle filter to the DS theory of evidence. It combines the strength of a classical particle filter to handle non-linear and non-Gaussian motion and error models with the power of the

DS theory to elegantly model uncertainty and absence of knowledge without having to specify any priors or conditionals. This latter property is particularly advantageous in camera networks, where limited fields of views and occlusions frequently pose problems.
Instead of dividing the ground plane in discretization cells, it is also possible to divide the 3D volume in the vicinity of the tracked person into a number of discretization cells. This would allow to also localize the person vertically. This would be useful in a scenario where people are likely to move in this direction, e.g., because the terrain is not flat or because they jump or climb on objects. In this work however, we assume that people move mainly on a flat ground plane.
In the next subsection we briefly present the main aspects of the algorithm of [Munoz-Salinas et al., 2009]. Only the elements used in our camera selection method for tracking are highlighted. For a comprehensive description of its operation, we refer the reader to [Munoz-Salinas et al., 2009]. The remainder of the section will discuss the details of our camera selection method for tracking.

6.5.3 Tracking Using Evidential Filters

Consider for each tracking target at time t a set of positions x_l, $1 \leq l \leq L$, on the ground plane of a 3D scene. Compliant with the established terminology used in tracking literature, we will call these positions particles. For each particle the hypotheses that the tracked person is present at this position ($\{present\}$) or not ($\{\neg present\}$) are investigated. These two hypotheses constitute the frame of discernment Θ associated with this particle. To gather evidence, each camera i makes an observation and translates it for each particle into a body of evidence m_i^l through the BBA defined as follows.
A 3D model of a person is assumed to be standing at position x_l. The 3D model is a cuboid with ground plane centered at x_l and with the dimensions of an average adult (see Fig. 6.1a for two examples). Observations for a particle are gathered over this 3D model. One of the advantages of this procedure is that it offers some robustness against occlusions that vary with height.
Let $Vis(x_l)$ be the percentage of the 3D model inside the viewing frustum of a camera. If $Vis(x_l) = 0$, nothing is known about the hypotheses and $m_i^l(\Theta) = 1$. Otherwise the 3D model is projected onto the camera view, defining a region *pm* of pixels. In Fig. 6.1c this region is shown for the model furthest from the camera. Any foreground detection algorithm from literature can be used to segment the image in foreground regions, that contain the objects of interest in the scene, and background regions. The foreground pixels lying within *pm* form a region *fpm* (Fig. 6.1d).
If the scene contains multiple people, other persons may block the view on the person at position x_l. By placing 3D models at the estimated positions of all other persons in the scene, the region *vpm*, being the part of *fpm* not occluded by other people, is estimated (Fig. 6.1e). The measure $Nocclu(x_l)$ is proportional to the number of pixels in *vpm* relative to the number of pixels in *fpm*. How much evidence can be gathered about the hypotheses depends on both



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

Figure 6.1: (a) Projection of the wire frame of two 3D models into a camera image. (b) Detected foreground and projection of the wire frame of the 3D models. (c) Image region within 3D model projection of furthest person (*pm*). (d) Foreground within 3D model projection of furthest person (*fpm*). (e) Unoccluded foreground within 3D model projection of furthest person (*vpm*).

the visibility of the tracking target and on its level of occlusion. The authors in [Munoz-Salinas et al., 2009] therefore define $m_i^l(\Theta) = 1 - \text{Vis}(x_l)\text{Nocclu}(x_l)$. Observations that are considered evidence for the hypothesis that the tracked person is present at position x_l are

- the presence of pixels that are part of the foreground within the projected 3D model region *pm*. This observation is captured by the measure $Occu(x_l)$;
- a small distance between the center of mass of the detected foreground region *fpm* and the projected model *pm* in the image of a camera. The degree to which this is the case is represented by $Cent(x_l)$;
- a small difference in color within the region *vpm* to the color histogram model of the tracked person kept by the camera. This similarity is represented by $Cd(x_l)$. Complete similarity is represented by $Cd(x_l)$=1.

The basic belief assigned to the $\{present\}$ hypothesis is defined as

$$m_i^l(\{present\}) = (1 - m_i^l(\Theta))\text{Occu}(x_l)\text{Cent}(x_l)\text{Cd}(x_l).$$

Then, by the definition of a basic probability assignment, $m_i^l(\{\neg present\}) = 1 - m_i^l(\{present\}) - m_i^l(\Theta)$.

The bodies of evidence m_i^l from different cameras are fused using the fusion rule of Eq. 4.1 (or using the cautious conjunctive rule if the observations are not independent), resulting in a BBA m^l for each particle x_l. Each particle x_l gets assigned a relevance that is proportional to $m^l(\{present\})$ and the target is estimated to be at the position with maximal relevance. Once the new position estimate is known, all cameras in which the target is visible update their color model of the target (used in the calculation of Cd). Then a new set of particles for time instant $t+1$ is generated using the classical condensation algorithm [Isard and Blake, 1998] in which the resampling probability of the particles is equal to their relevance and the propagation step assumes a random walk movement of the particles following a Gaussian distribution $N(0, \sigma_{\text{prop}})$.
For more details on the tracking algorithm, the reader is referred to [Munoz-Salinas et al., 2009].

6.5.4 Camera Selection for Tracking

We now propose a method to assess the suitability of a camera set for tracking a particular person. As discussed previously, we consider the frame of discernment $\Omega = \{\omega_1, \omega_2, \ldots, \omega_g, \ldots, \omega_G\}$, where $\omega_g = \{x \in X_g\}$ and X_g, $g \in [1, G]$, are discretization cells on the ground plane. We derive the suitability of a set S for tracking a target from the certainty with which it can locate the target in one of the cells. The set suitability value $v(S)$ is calculated from the body of evidence m_S that contains the evidence for each of the 2^G propositions in Ω. In the following we explain how we define the BBA m_S that translates camera observations into evidence supporting the propositions in Ω.
The observations about a single cell X_g can provide direct evidence for only two hypotheses: the target is in this cell (ω_g) or it is not ($\Omega \backslash \omega_g$). Combining evidence from different cells using one of the combination rules of Section 4.2 allows us to draw indirect conclusions about some hypotheses for which no direct evidence can be gathered because, as explained in Section 4.2, applying these rules leads to a specialization of the basic belief (i.e., basic belief is redistributed over the subsets of each proposition). Indeed, if there is evidence supporting the hypothesis that the target is not in cell X_g and other evidence that it is not in $X_{g'}$, then the hypothesis that it is in any of the other cells becomes more likely. To model this intuitively plausible evidence gathering process, we consider the assessment of the hypotheses in Ω based on the observations about a single cell X_g as a separate body of evidence, denoted as m_S^g.
To gather evidence for the propositions in Ω, we could perform observations and extract evidence from them. However, for the tracking algorithm we already need to perform some observations. For reasons of efficiency, we make use of these observations performed for the tracking algorithm to extract evidence for the propositions in our frame of discernment Ω. We take

$$m_S^g(\omega_g) = \max_{\forall l \in [1,L] | x_l \in X_g} m_S^l(\{present\}), \tag{6.13}$$

where m_S^l is obtained by fusing the bodies of evidence m_i^l for all cameras in the considered set: $i \in S$. Eq. 6.13 expresses that the basic belief that the tracked person is in cell X_g equals the highest evidence of presence measured in the particles that lie in this cell. The quality of this approximation depends on the sampling density in the cell. The basic belief for the hypothesis that the target being tracked is not in cell X_g is defined as the minimal evidence of absence measured in any of the particles that lie in this cell:

$$m_S^g(\Omega \backslash \omega_g) = \min_{\forall l \in [1,L] | x_l \in X_g} m_S^l(\{\neg present\}). \tag{6.14}$$

Note that this is equivalent with

$$m_S^g(\Omega \backslash \omega_g) = 1 - \max_{\forall l \in [1,L] | x_l \in X_g} [m_S^l(\{present\}) + m_S^l(\Theta)].$$

This implies that when we have full information about all particles in the cell (i.e., $m_S^l(\Theta) = 0$ for all $l \in [1, L]$ for which $x_l \in X_g$), $m_S^g(\Omega \backslash \omega_g) = 1 - m_S^g(\omega_g)$. For example if there is no evidence that the target is in this cell because $m_S^l(\{present\} = 0$ for all particles in the cell, i.e., $m_S^g(\omega_g) = 0$, then we are sure that the target is not in this cell: $m_S^g(\Omega \backslash \omega_g) = 1$. If nothing is known about the presence or absence of the target at all particles in the cell (i.e., $m_S^l(\Theta) = 1$ for all $l \in [1, L]$ for which $x_l \in X_g$), $m_S^g(\Omega \backslash \omega_g) = 0$.
Observations about X_g can only provide direct evidence for the hypotheses ω_g and $\Omega \backslash \omega_g$. Therefore $m_S^g(A) = 0$ for all proper subsets of Ω except for ω_g and $\Omega \backslash \omega_g$. By the definition of a basic probability assignment then $m_S^g(\Omega) = 1 - m_S^g(\omega_g) - m_S^g(\Omega \backslash \omega_g)$.
The cell X_G never contains any particles because it is in the part of the ground plane area in which we do not gather observations, either because we cannot or because we do not expect the tracking target to be there. Because no direct evidence about the presence or absence of the target in X_G can be gathered $m_S^G(\Omega) = 1$ and $m_S^G(A) = 0, \forall A \subset \Omega$. If there are no particles in another cell $X_{g'}$, $g' \in [1, G-1]$, this cell de facto assumes the same role as X_G. We therefore merge such a cell with X_G and remove the hypothesis $\omega_{g'}$ from the frame of discernment.
The body of evidence m_S is obtained by fusing the bodies of evidence m_S^g from all cells. Distinct pieces of evidence can be combined using Dempster's rule of combination (Eq. 4.1). This is not possible if the evidence is not independent. Dependencies between the evidence of different cells can arise from two sources:

- for some particles the 3D model used in the evidence gathering process of [Munoz-Salinas et al., 2009] extends into adjacent cells. If the intersection between the models from which evidence is gathered in either Eq. 6.13 or 6.14 for different cells is not empty, the evidence of these cells is not distinct. The probability with which this occurs can be minimized by choosing an appropriate discretization scheme. This will be further discussed in Section 6.5.5.1;

- the projections into the camera views of the 3D model associated with particles in different cells can overlap. However, as soon as evidence is gathered from at least two cameras with sufficiently different viewing angles, the projections cannot overlap in all views and the dependence of the evidence sources will be small.

Non-distinct pieces of evidence should be combined using the cautious conjunctive rule of [Denoeux, 2008]. Unfortunately, as mentioned in Section 4.2, the result of fusing distinct bodies of evidence with this rule is less informative than if Dempster's rule (Eq. 4.1) is used. It is therefore important to establish to what extent the possible dependence between the evidence sources of different cells actually manifests itself in a practical scenario. In Section 6.6.3 we will ascertain that Dempster's rule can be safely used to combine the bodies of evidence m_S^g from all cells if S contains at least two cameras.
When we have obtained m_S, we can use Eq. 6.8 to calculate the suitability $v(S)$ of a camera set S for tracking a specific target.

6.5.5 Practical Scheme for Tracking with Selected Cameras

In the following, we discuss the practical issues that need to be considered when implementing our algorithm in a practical smart camera network tracking scenario.

6.5.5.1 Discretization Scheme

A design choice which influences the suitability value of a camera set is the discretization scheme of the ground positions. This discretization is only used to assess if a set of cameras can make a sound estimate of the position of the tracking target and not for the tracking as such.
We assume the target is at its estimated position (or a prediction thereof, as will be explained in Section 6.5.5.2). Around this position we center a discretization cell which we call the center cell. Its center is the estimated target position. It is shaped and sized such that the 3D model placed at the estimated target position is completely disjunct with a 3D model placed in any particle in another cell. Hence, the minimal allowed side length of the center cell is twice the 3D model side length. The rationale is that camera sets that localize the target in the center cell and also clearly observe that the target is not present in the other cells are very suitable to perform the tracking. In the schemes Fig. 6.2a, b and c, the side length of the center cell is the minimal allowed side length. In Fig. 6.2d the center cell is larger than this minimal allowed size and in Fig. 6.2e it is smaller. Various possible divisions of the space around the center cell into other discretization cells are proposed in Figure 6.2a-c.
The influence of the scheme choice on the camera selection algorithm will be discussed in Section 6.6.4.

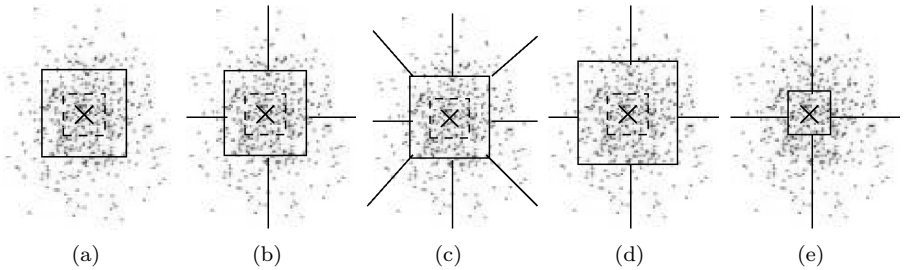


Figure 6.2: Discretization schemes of the ground positions. The dots are the particle positions, the cross indicates the estimated target position, the dashed line delineates the contour of the person 3D model centered at the estimated target position and the full lines indicate the discretization cell borders. (a), (b) and (c) show various possible divisions of the space around the center cell, which has the minimal allowed side length of twice the 3D model side length. In (d) the center cell is larger than this minimal allowed size and in (e) it is smaller.

6.5.5.2 Avoiding Costly Data Transmissions

To calculate the suitability of a camera set, Eqs. 6.13 and 6.14 need to be evaluated, which requires observations about all particles to be made by all cameras and to be transmitted to some point of central processing. However, as will be explained more thoroughly in Section 6.5.5.3, we wish to save camera and network resources by making and transmitting fewer observations.

To this end, we determine for each tracking target at a base station (which can coincide with one of the cameras) which set of cameras is most suited to make and transmit observations about this target. This selection is not based on observations of the current time instant. Instead, as will be explained below, it is made by assuming temporal smoothness between subsequent frames of the positions of the tracking targets. This is a valid assumption if the moving speed of the targets is not too high compared to the frame rate of the system. The base station broadcasts the camera selection decision to all cameras. Only the selected sensors actually make and transmit real observations of the scene.

To make the camera selection decision at the base station, the camera images are not available, nor any of the observations of the current frame (in fact, no observations have been made yet, also not on the cameras). Of course, these images or observations could be transmitted by the cameras, but it is exactly these costly observations and transmissions that we wish to avoid.

Therefore, the selection decision is based on simulated observations of 3D models placed at the *predicted* target positions. Alternatively, the observations of the previous time frame could be used to base the selection on. However, these are only available for the cameras that were selected at the previous time instant, since only the selected cameras have transmitted their observations to the base station. To keep the input data of the selection algorithm homogeneous for all cameras, we prefer to use simulated observations.

To predict a tracking target position we assume that the target does not move appreciably between subsequent frames. The higher the frame rate and the lower the target's speed, the more reasonable this assumption is. As will be apparent from the results section, Section 6.6, this assumption of temporal smoothness leads to satisfying camera selection results, even when people make many sudden and fast movements, such as in a sequence of a basketball game. Since for the simulated observations no background/foreground silhouettes of the cameras are available, *fpm* is set equal to *pm* for all particles. The simulated regions *vpm* for a particular target are obtained by placing 3D models at the estimated positions of all other persons in the scene. From these input regions, *Vis*, *Nocclu*, *Occu* and *Cent* are derived for all particles. For the simulated observations no color information is available, so for all particles the simulated observation of *Cd* is simply set to 1.
The discretization of the ground positions according to one of the schemes described in Section 6.5.5.1 is also based on a prediction of the target position. The camera selection decision is broadcast to all cameras and only the selected sensors actually make and transmit real observations of the scene. Based on these observations, the tracking algorithm estimates the target's current position. Based on this position, the selection decision for the following frame is calculated, and so on.

6.5.5.3 Computation and communication

The proposed method of identifying the most suitable camera set for a tracking task can be very useful in a practical smart camera network to save computational and communication resources. In the tracking algorithm of [Munoz-Salinas et al., 2009], as in most tracking methods described in literature, most resources are used for observation gathering rather than for other parts of the methods. The cost of estimating the target state is negligible in comparison. When the tracking decision can be based on the observations of only a selected set of sensors, a substantial saving of resources is achieved. This is especially the case in wireless camera networks, where saving communication and processing power is essential in enabling battery operation or prolonging battery life. In such systems lowering communication bandwidth is also a very important factor in reducing system latency because transmissions usually occur sequentially (cfr. the carrier sense, multiple access/collision avoidance channel access mode used in the ZigBee specification). The non-selected cameras can be used for other network tasks or can be left idle. Depending on the foreground segmentation method, it may be necessary that they keep capturing images to update their background model.
In this section we analyze which operations are saved by tracking with a selection of only N' cameras instead of with all N cameras. We also discuss which extra operations are necessary for the sensor selection itself as compared to tracking with all cameras, where no such selections need to be computed. The numbers are listed in Table 6.1. To provide a rough estimate of the relative computation time of the different operations, their computation time on an In-

Table 6.1: Analysis of extra and saved operations when tracking with a selection of N' out of N cameras instead of with all N cameras. L is the number of particles. The average computation time of each operation on an Intel Core i7 920/2.67GHz processor is also listed.

Operations	Baseline: All N Cams	# Extra Operations	# Saved Operations	Computation Time (ms)
pm	NL	0	0	0.04
fpm	NL	0	$(N-N')L$	0.12
vpm	NL	NL	$(N-N')L$	0.13
Vis	NL	0	0	0.13
Nocclu	NL	NL	$(N-N')L$	0.18
Occu	NL	NL	$(N-N')L$	0.19
Cent	NL	NL	$(N-N')L$	0.42
Cd	NL	0	$(N-N')L$	1.86
Selection	0	1	0	f(G,L,N,N') (see text)

tel Core i7 920/2.67GHz processor is listed (averaged over 200 000 instances).

Additional savings are possible by not running the camera selection algorithm for every frame but applying a selection decision to several frames. This will have an impact on the tracking accuracy as necessary changes of the selected camera sets will be delayed. If the frame rate of the system is sufficiently high with respect to the scene changes in the room, such that successive observations result in similar selection results, the performance drop will be minimal (cfr. also Section 5.8.4).

Let us now analyze the operations necessary to obtain the simulated observations. The predicted target position is the estimated target position in the previous frame (see Section 6.5.5.2). It does not require extra operations to obtain. The same holds for the regions *pm* and their derived measures *Vis* because they are the same for simulated and real observations. The regions *fpm* for the simulated observations are taken equal to the regions *pm*, so also for those no extra computations are needed. Extra operations are needed to calculate the simulated observations *Nocclu*, *Occu* and *Cent* in N cameras and for L particles, and the regions *vpm* as part of their input. As for the simulated observations no color information is available, *Cd* does not need to be computed.

When tracking with only the selected cameras, only $N'L$ actual observations (instead of NL when tracking with all cameras) of *Nocclu*, *Occu*, *Cent* and *Cd*, and their necessary input, the regions *fpm* and *vpm*, need to be calculated so their computation is saved $(N-N')L$ times.

For the selection algorithm we need to determine for L particles in which of the G discretization cells they lie. Then, each iteration of the greedy optimization of Algorithm 4 to solve Eq. 6.1 involves these major steps on lines 8 and 9 of

the algorithm:

1. fusing evidence of selected cameras;
2. for G cells: obtaining the maximal evidence of presence and the minimal evidence of absence to construct m_S^g;
3. fusing the m_S^g from all G cells to obtain m_S;
4. determining the aggregated uncertainty in the obtained body of evidence m_S;
5. calculating the value as in Eq. 6.8.

In a non-optimized implementation, the number of operations exponentially rises in steps 3 and 4 with the number of cells G because there are 2^G hypotheses in the power set of Ω. It is therefore important to keep G low (we advise that G is smaller than 10). The number of loops that has to be executed in Algorithm 4 depends on the constraints imposed on the desired camera set. Assume for example that the finally selected set contains N' cameras. In this case $\sum_{i=0}^{N'-1}(N-i)$ loops need to be run through. For typical parameters $G = 6$, $L = 50$, $N = 8$ and $N' = 3$, the selection algorithm took on average 3.86 ms during 10000 executions in a non-optimized c++ implementation on the mentioned Intel Core i7 920/2.67GHz processor. To approximate the computation time of the selection algorithm for other parameters N and N', we neglect the time needed to determine for L particles in which of the discretization cells they lie and we assume that each loop in this algorithm takes a fixed time t_{loop}. The total time for selection was 3.86 ms for $N = 8$ and $N' = 3$, from which we estimate $t_{loop} = 0.18$ ms.
To assess if from a purely operational point of view it is favorable to perform tracking with a selection of cameras instead of with all of them, one has to draw the balance between the computations saved over the entire network and the extra computations spent compared to the baseline scenario where all cameras are used. In the baseline scenario, *pm*, *fpm*, *vpm*, *Nocclu*, *Occu*, *Cent* and *Cd* have to be computed NL times, i.e., for all particles on every camera. Note that the calculation time of *Cd* is higher than for *Nocclu*, *Occu* and *Cent*. These latter basically involve pixel counting operations, whereas *Cd* requires the creation and comparison of a color histogram. The color histogram is created in the Yuv color space, so a conversion between the RGB and the Yuv color spaces is necessary. For this reason the number of operations per pixel is higher for the calculation of *Cd* than for *Nocclu*, *Occu* and *Cent*. The fraction of saved computations is

$$\frac{2.90(N-N')L - 0.92NL - 0.18\sum_{i=0}^{N'-1}(N-i)}{(0.04+0.12+0.13+0.13+0.18+0.19+0.42+1.86)NL}$$

which we approximate by $0.95(N-N')/N - 0.30$ because $\sum_{i=0}^{N'-1}(N-i) \ll NL$. Thus as long as $N' < 0.68N$ the total number of computations in the entire

Table 6.2: Example of extra and saved computation time as a percentage of the total baseline time for typical parameters $G = 6$, $L = 50$, $N = 8$ and $N' = 3$. The total baseline time is the computation time when tracking with all cameras.

Operations	Saved Comp. for G=6, L=50, N=8, N'=3 (% of Total Baseline Time)
pm	0.00%
fpm	2.55%
vpm	-1.54%
Vis	0.00%
Nocclu	-2.24%
Occu	-2.28%
Cent	-5.13%
Cd	37.89%
Selection	-0.31%
Total percentage of saved computation time	28.93%

network is smaller when tracking with a selection of cameras instead of with all of them, also taking the overhead calculations to determine the selection into account. In Table 6.2 we indicate the savings for typical parameters $G = 6$, $L = 50$, $N = 8$ and $N' = 3$. Note how the computation savings are mainly concentrated in the calculation of *Cd*.

Independently of whether the savings balance is positive or negative, an advantage of tracking with a selection of cameras instead of with all of them is that it is possible to design the network such that the overhead computations of the selection are performed on a base station, which can be made to have a higher performance or to have access to more power than the smart cameras. In such a design there is always a saving in computations at the side of the smart cameras.

Particularly important for wireless camera networks is that communication with $N - N'$ cameras is saved. In such networks, this is a very important factor in reducing the latency of the system and in saving communication power and bandwidth. Assume for example that in a network of 10 cameras 3 cameras are selected for tracking a target. Let us further assume we can neglect the small amount of resources needed to broadcast the selection decision. This is a reasonable assumption, as the number of bits needed to represent the selection decision is a lot smaller than the number of bits required to represent the camera observations. In this example, the communication power and bandwidth savings amount to 70% per target.

6.6 Results

In this section we discuss the performance of the camera selection method for tracking as proposed in Section 6.5.

6.6.1 Test Data

We use natural video sequences recorded in three different environments for our evaluation.

The first environment is an indoor scene of $7m$ by $9m$ observed by $N = 8$ IP cameras. Approximately 3 minutes of footage (900 frames) in which two persons appear have been recorded at 5 fps and at QVGA resolution (320×240). Only the starting points of these recordings have been synchronized.

The second environment is the one from the publicly available basketball dataset from the European project APIDIS [De Vleeschouwer and Delannay, 2009], already used in Section 4.6. To recapitulate: in these sequences a basketball court is observed by seven synchronized and calibrated cameras (see Fig. 6.7). The videos are processed at 25 fps and at a resolution of 800×600. The size of the field is $15m \times 28m$. There are on average 12 targets on the field. We have used the images recorded in the time interval 18:47 until 18:50 (4500 frames). As most cameras point to the left half of the court, only positions in that half are considered for the evaluation.

The third environment is the indoor scene of $5m$ by $4m$ observed by $N = 10$ web cameras already used in Sections 4.6 and 5.8. The camera views are shown in Fig. 6.10. Approximately 8 minutes of footage (2400 frames) in which two, three and four persons appear have been recorded at 5 frames per second and at CIF resolution (352×288). Only the starting points of these recordings have been synchronized.

Foreground detection in the first and second environment is done using an algorithm based on mixture of Gaussians modeling [Stauffer and Grimson, 2000] with elementary shadow removal [Kaewtrakulpong and Bowden, 2001]. Because the cameras of the third environment have quite an unstable automatic gain control, extreme apparent lighting changes of the observed scene are frequent and we use a background foreground segmentation algorithm that can adapt especially quickly to such changes [Li et al., 2003], combined with the same shadow removal as for the other environments. The size of the 3D model box is set to $0.5m \times 0.5m \times 1.7m$.

For the sequences of the first and the third environment ground truth ground plane positions of the tracked persons have been generated for every fifth frame (1 s intervals). This has been done by manually checking the output of the multi-camera person detection algorithm of [Delannay et al., 2009] and correcting it where necessary. For the APIDIS sequence, ground truth target positions have been made available at 1 s intervals.

6.6.2 Evaluation Metrics

For each frame for which ground truth target positions are available, we determine the root mean squared error (RMSE) of the estimated target positions with respect to the ground truth positions and average them over all tracked targets and all frames. We also count the number of times a tracker loses its target. In the average RMSE computation we exclude the large error due to losing a target. After each loss the tracking is reinitialized at the correct position and tracking resumes.
A person is considered lost if none of the particles of its tracker is closer to the ground truth position than twice the maximal standard deviation σ_{prop} of the propagation of the particles, plus half the side length of the 3D person model box. The idea is that in this case the target is not likely to be recovered anymore by a propagation of the particles. In [Munoz-Salinas et al., 2009] the maximal $\sigma_{\text{prop}} = 2s/\text{fps}$, where s is the speed with which the targets are assumed to move and fps is the frame rate at which the system operates. In our first and third environment the frame rate is 5 fps and the speed is assumed 1m/s. In the APIDIS environment the frame rate is 25 fps and the moving speed of the basket ball players is assumed 5m/s. Both scenarios lead to $2\sigma_{\text{prop}} + \text{sidelength\_3Dbox}/2 = 1.05\text{m}$.
To assess the computational load at the camera side of the network in the following experiments, we determine at each time instant the number of times a camera has to collect observations for tracking one of the targets. We compare this number with the number of times a camera has to collect observations for tracking one of the targets when all targets are tracked with all cameras (i.e., the number of targets multiplied by the number of cameras).
In the following experiments a set of cameras is selected for each person independently of the sets selected for other persons. As a consequence, these sets may or may not overlap. We call a camera that is selected to track at least one target an *active* camera. Communication is saved with all non-active cameras. A lot of overlap between sets selected for different tracking targets is beneficial from a communication point of view, but it also increases the instantaneous computational burden for some cameras because they have to calculate *pm*, *fpm*, *vpm*, *Nocclu*, *Occu*, *Cent* and *Cd* for several targets. To monitor these two aspects, we determine at each time instant the number of active cameras as a fraction of all cameras and the maximum number of targets assigned to one camera as a fraction of all targets. These measurements will be conducted in Section 6.6.6 on the experiments of Section 6.6.5. A more in-depth study of the consequences for the frame rate and the battery life time of the system is beyond the scope of this work.

6.6.3 Distinctness of Cell Evidence

In a first experiment we analyze the distinctness of the evidence of the cells for the various discretization schemes of Fig. 6.2. We focus on the distinctness of the evidence of the center cell from the evidence of the other cells. A similar

analysis can be performed for each cell, but for conciseness it is not included here as it leads to similar conclusions.
If the evidence of the different cells is distinct, we can use Dempster's rule (Eq. 4.1) for combining the bodies of evidences m_S^g of the different cells X_g, $g \in [1, G]$ to obtain m_S. Otherwise we must use the cautious conjunctive rule of [Denoeux, 2008].
Evidence is distinct if it is produced by independent sources. A formal definition of the concept of distinct evidence was provided in [Smets, 1992]. In this work, we use the practical method that has been proposed in [Quost et al., 2008] to assess the dependence between sources. Quost *et al.* measure the distance between the bodies of evidence m_1 and m_2 produced by two sources. The smaller this distance, the greater the dependence. The distance metric introduced in [Jousselme et al., 2001] is adopted, which is defined as:

$$d(m_1, m_2) = \sqrt{(\frac{(m_1 - m_2)D(m_1 - m_2)^T}{2})}, \tag{6.15}$$

with an element $D_{A,B}$ of matrix D defined as

$$D_{A,B} = \frac{|A \cap B|}{|A \cup B|}, \forall A \subseteq \Omega, B \subseteq \Omega, A \neq \emptyset, B \neq \emptyset. \tag{6.16}$$

This distance is normalized, which means that $0 \leq d(m_1, m_2) \leq 1$.
The viewing angles of the cameras in the set S determine to what extent the projections into the camera views of the 3D model associated with particles in different cells can overlap. If these projections overlap, the evidence is not produced by independent sources and we expect the evidence not to be distinct. We want to focus on the evidence which we expect not to be distinct from the evidence of the center cell. To this end, in each frame, we determine which other cell $g = g^*$ has the evidence that is least distinct from that of the center cell ($g = 1$):

$$g^* = \underset{g \in [2,G]}{\arg\min}\, d(m_S^1, m_S^g). \tag{6.17}$$

In Fig. 6.3 we plot for the various discretization schemes of Fig. 6.2 the distance $d(m_S^1, m_S^{g^*})$ averaged over all frames as a function of the number of cameras in the set S. The number of particles is set to $L = 50$. Clearly for all discretization schemes the average distance $d(m_S^1, m_S^{g^*})$ is smallest when S contains only one camera and it almost reaches its maximal value, namely 1, as soon as S contains at least two cameras. This indicates that the bodies of evidence of the different cells are approximately distinct as soon as S contains at least two cameras. This matches our expectations of Section 6.5.4 with regard to the overlap of the projections of the 3D model in the views of the cameras in S. It also justifies using Dempster's rule (Eq. 4.1) for combining the bodies of evidences m_S^g of all cells to obtain m_S if the evidence stems from at least two cameras. If S contains only one camera, it is more prudent to use the cautious conjunctive rule of [Denoeux, 2008] to fuse the bodies of evidence m_S^g of the different cells X_g, $g \in [1, G]$ as the bodies of evidences m_S^g of the different cells may not be totally distinct.

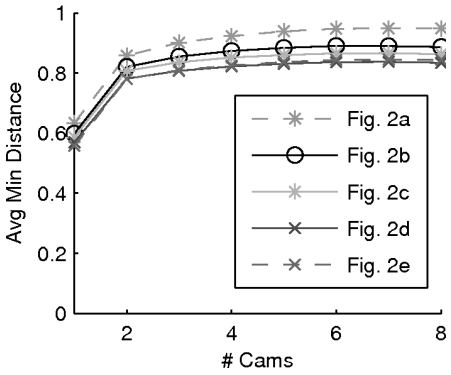


Figure 6.3: For various discretization schemes, the distance $d(m_S^1, m_S^{g^*})$ averaged over all frames as a function of the number of cameras in the set S used to gather the evidence.

6.6.4 Influence of Parameters

In a second experiment we assess the influence of the number of particles L and the discretization scheme of the ground plane on the tracking performance. We use the sequences from the first environment for this. For each person a set of maximally three cameras is selected independently of the set selected for the other person. We test the discretization schemes of Fig. 6.2, with the side length of the center cell twice the width of the 3D model box, i.e., 1m, for the schemes of Fig. 6.2a-c, 3 times this width, i.e., 1.5m for the scheme of Fig. 6.2d and exactly this width, i.e., 0.5m for the scheme of Fig. 6.2e.

In Table 6.3 and Fig. 6.4 we show how the tracking performance changes with varying numbers of particles and for different discretization schemes. As a reference, we also include results for tracking with all eight cameras.

As a general trend, we observe that the tracking performance improves with rising numbers L of particles. This improvement is largest in the range $[10, 30]$. For tracking with all cameras the average RMSE remains constant at approximately 0.15m for $L \geq 30$. For all discretization schemes the average RMSE and the number of target losses also level out for $L \geq 30$. This leads us to the important observation that for these schemes it is not necessary to use more particles than in the case where all cameras are used.

From Fig. 6.4 we further observe that in spite of the large reduction of active cameras from eight to three, the rise in average RMSE is limited, varying from less than a centimeter (for $L = 100$ and discretization scheme of Fig. 6.2d) to a maximum of less than ten centimeters (for $L = 10$ and discretization scheme of Fig. 6.2e). The differences between the proposed discretization schemes are small. The schemes of Fig. 6.2a and of Fig. 6.2c perform worst in terms of average RMSE for $L \geq 10$ and number of target losses. The scheme of Fig. 6.2c has many cells, requiring observations from many different viewing angles to correctly discern between the presence or absence of the target in

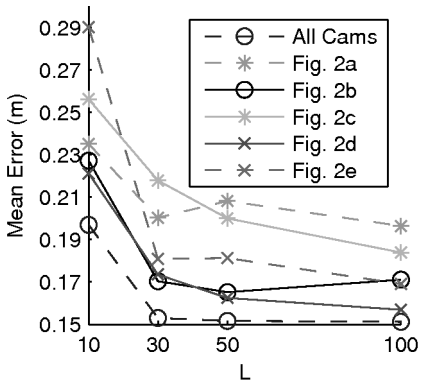


Figure 6.4: Average RMSE as a function of the number of particles L when tracking with all or with a selection of cameras obtained using different discretization schemes.

each of them. If such observations are not available, uncertainty is introduced into the body of evidence m_S, and the camera selection algorithm is unable to make a sound choice. The scheme of Fig. 6.2a has only one outer cell. We solve Eq. 6.1 with a greedy optimization algorithm, adding cameras one by one. With this discretization scheme, it is impossible to identify which is a suitable camera to first add to the selection set. Indeed, because of camera projection geometry, it is impossible for one camera to observe that the target is absent in this entire cell (except for a top view camera positioned directly above the tracking target). For this reason, this discretization scheme hampers the search for a suitable camera set.

Reducing the size of the center cell of the discretization scheme decreases tracking performance, as is shown in Fig. 6.4 by the curve of the discretization scheme of Fig. 6.2e. In this scheme the 3D model placed in some particles in the non-center cells picks up evidence of the presence of the tracked person at the estimated target position, causing uncertainty in the body of evidence m_S. Increasing the size of the center cell (see Fig. 6.4, curve of Fig. 6.2d) does not have a major influence on the tracking performance. Indeed, this does not violate the rationale that a suitable camera set for tracking should observe that the center cell contains the tracking target and the target is not present in the other cells. On the contrary, the increased center cell size allows for some error in the prediction of the target position as there is some buffer for the 3D model of the particles in the other cells not to intersect with the tracked target. The size should however not be chosen too large with respect to the spread of the particles, otherwise the non-center cells will be insufficiently sampled.

We conclude that a number of particles of at least thirty and a discretization scheme with a center cell with side length at least twice the 3D model side length and four other cells (i.e., Fig. 6.2b or d) are good parameters for our selection algorithm. We will use the discretization scheme of Fig. 6.2d in subsequent experiments, and choose the number of particles $L = 50$.

Table 6.3: Number of target losses as a function of the number of particles L when tracking with all or with a selection of cameras obtained using different discretization schemes.

	All	Scheme of				
	cams	Fig. 6.2a	Fig. 6.2b	Fig. 6.2c	Fig. 6.2d	Fig. 6.2e
L=10	0	2	0	5	2	3
L=30	0	2	0	3	0	0
L=50	0	2	0	2	0	0
L=100	0	1	0	1	0	0

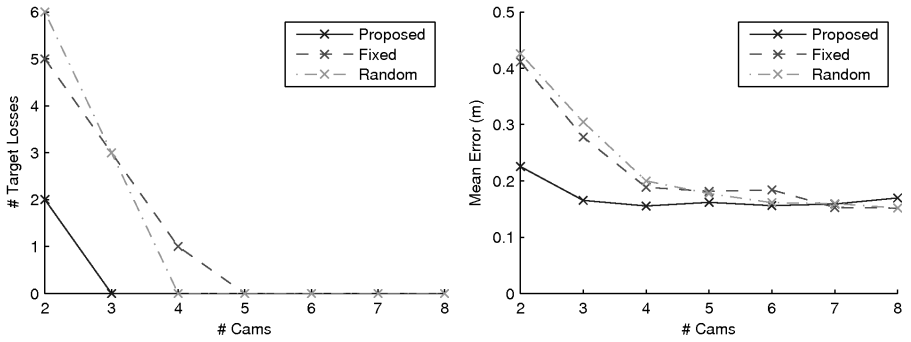
6.6.5 Tracking Performance

Using the camera selection scheme described in Section 6.5.5, we now track persons using camera sets with varying size limits. For each person a set of cameras is selected independently of the sets selected for other persons. We use the discretization scheme of Fig. 6.2b and choose the number of particles $L = 50$.

We compare the tracking performance of this method with tracking using a subset of cameras that remains fixed throughout the sequence. This fixed set is the same for all targets and has been chosen as the best performing one among all possible fixed sets. We also compare with tracking using a set of cameras that is randomly chosen in each frame for each person.

Although the main strength of the use of generalized instead of classical information theory in this work is to provide a tool for easily modeling the impact of observation quality on the localization certainty of multiple tracking targets, it would be very interesting to assess if the generalized information-theoretic approach provides specific advantages over a method using classical information-theory. Unfortunately, we are not aware of any camera selection systems from literature with which we could compare the proposed method. Indeed, the methods for tracking based on classical information theory presented in Section 6.3.4 are not able to handle the complex scenarios in our test data. Most of them are not suited for camera sensors but instead are designed for sensors that produce simpler output (for example, direction-of-arrival sensors). The ones designed for camera tracking ([Denzler et al., 2003; Sommerlade and Reid, 2008]) have only been demonstrated on video sequences with one person in the scene, and it is unclear how they can handle occlusions in the case of multiple persons. A comparison of generalized and classical information-theoretic approaches is therefore not provided in this work.

Fig. 6.5 shows the results for our first environment. One can clearly see that with the proposed camera selection method the tracking can be performed with as few as three cameras per person without substantial tracking quality loss. For a set of two cameras, the number of target losses and the average RMSE are larger than in the all camera case. The proposed camera selection method outperforms the fixed and random camera selection schemes. The performance gain is larger for smaller sets. For larger sets all methods perform equally well.



(a) (b)

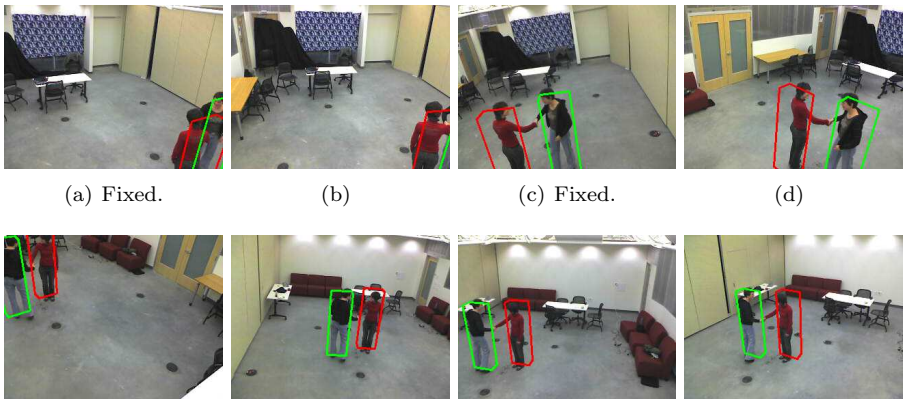
Figure 6.5: (a) Number of target losses and (b) average RMSE for different selection schemes as a function of the size limit of the selected camera sets for the first environment.

Note that the proposed method occasionally selects sets that are smaller than the set size limit.

A tracking result in this environment with a set of three cameras, selected using the proposed method, is shown in Fig. 6.6. For both persons the camera views shown in Fig. 6.6f-h were selected. These are the only views in which the persons are completely visible. Note how each camera has only a partial view of the room. A random selection of cameras may therefore very well have a bad view of the tracking targets. This also makes it impossible to find a fixed camera set that tracks people well at all times. In this case the best performing fixed set was Fig. 6.6a, c and g. Especially the view of Fig. 6.6a is an unfortunate source of information for this configuration of the tracking targets.

A tracking result in our second environment for the proposed camera selection method with a set size limit of three cameras is shown in Fig. 6.7. For this environment, a well performing fixed camera set exists because many cameras have a nearly complete view of the left half of the court. In particular, for three cameras, the views of Fig. 6.7a, c and e performed best. While this set guarantees a good overall view of the tracking targets at all times, at specific time instances such as this one more close-up views can be useful for some targets. Notice for example the two players indicated in green and cyan standing very close to each other under the basketball ring. The one with the cyan wire frame is occluded completely in Fig. 6.7c. The proposed method selects the views of Fig. 6.7a, b and e to track this player, thus replacing the bad view of Fig. 6.7c with the much better view of Fig. 6.7b. For the player with the green wire frame it selects the views of Fig. 6.7b, d and g, three close-up views.

Numerical results for this environment are shown in Fig. 6.8. Overall the proposed method outperforms the fixed and random camera selection schemes. Again the performance gain is larger for smaller sets, especially compared to



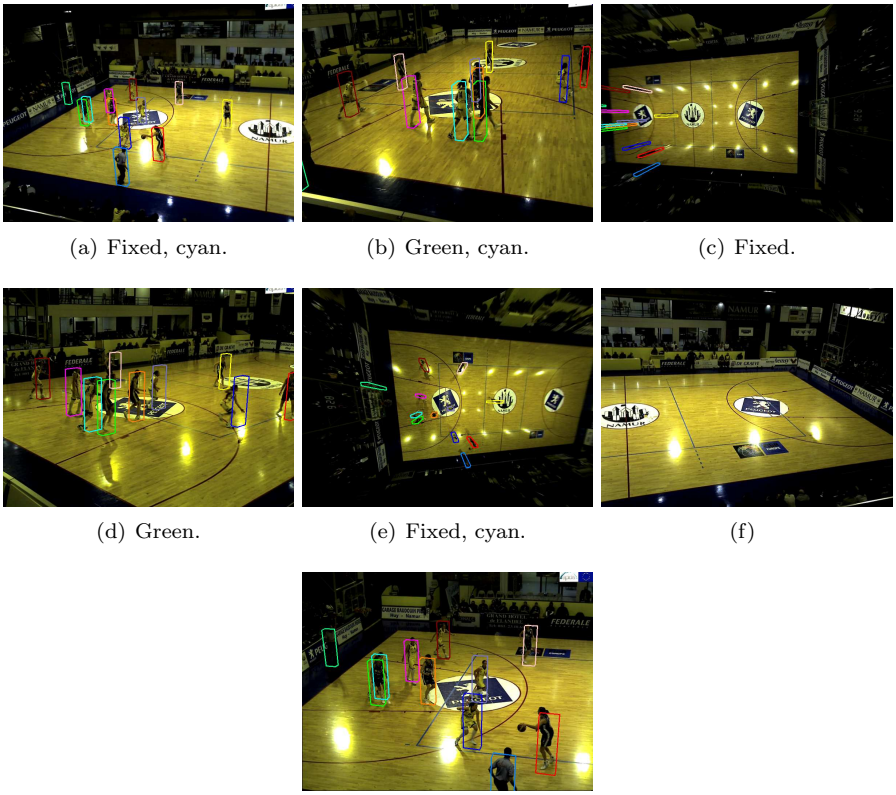
(a) Fixed. (b) (c) Fixed. (d)

(e) (f) Red, green. (g) Fixed, red, green. (h) Fixed, red, green.

Figure 6.6: Camera views in the first environment. The views which are part of the fixed selection of three cameras, or of the selection of three cameras determined using the proposed method for the person with the red or green wire frame are marked by fixed, red and green respectively.

the random selection scheme. When only two cameras are selected, a random choice of cameras often turns out to be an unfortunate choice. The fixed selection of two cameras performs quite well because the two top views in this case cover the entire field and offer a good overall view of all tracking targets at all times. However, the proposed method outperforms the fixed selection method in terms of number of target losses because it can also select more close-up views of the targets than the top views. When the set size limit equals the total number of cameras, namely seven, the proposed method outperforms the other selection schemes in terms of number of target losses. The fixed and random selection schemes in this case always boil down to tracking all targets with all cameras. The proposed method is able to selectively choose only the views that are suitable for tracking a target. Indeed, in some cases it is better not to use the information of a view because it is misleading. This happens for example when the target is mostly occluded, or when a lot of players appear close together. These cases are identified by the proposed method and the corresponding view is then not selected.

Results for the third environment are shown in Fig. 6.9. For the average RMSE the same conclusion as for the previous environments can be drawn, namely that the proposed method outperforms the others in terms of average RMSE for small cameras sets and display equal performance for larger sets. The number of target losses is in this environment clearly a lot lower than for the random and fixed selection methods. This is because the cameras in this setup have narrow viewing frustums. This increases the importance of dynamic camera selection, as it is not possible to select a fixed or random set with an overview of the scene.

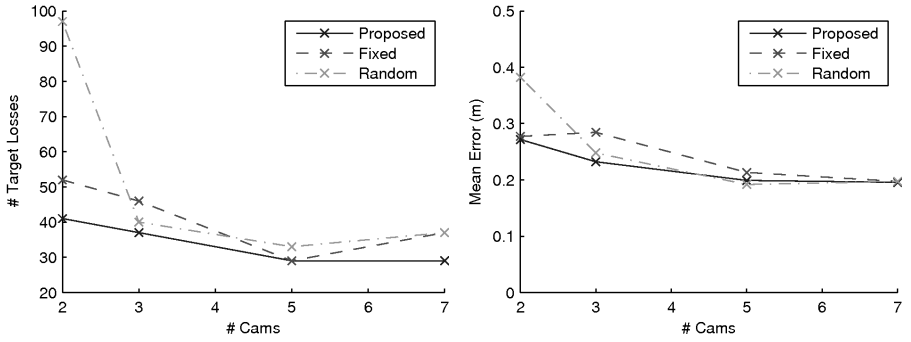


(a) Fixed, cyan. (b) Green, cyan. (c) Fixed.

(d) Green. (e) Fixed, cyan. (f)

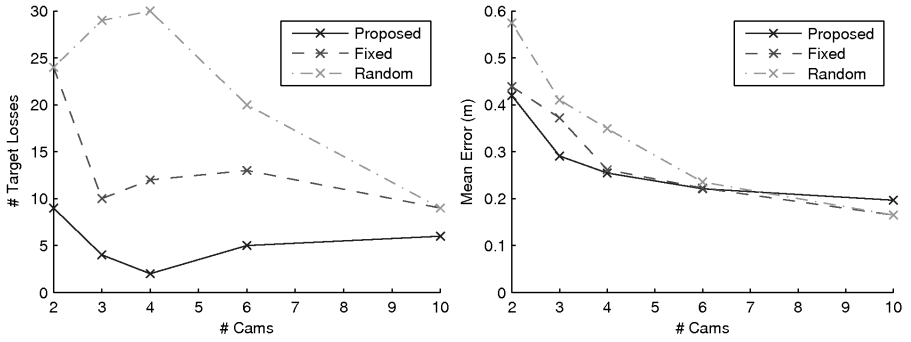
(g) Green.

Figure 6.7: Undistorted camera views in the second environment. The views that form the fixed selection of three cameras, or the selection of three cameras determined using the proposed method for the person with the green or cyan wire frame are marked by fixed, green and cyan respectively.



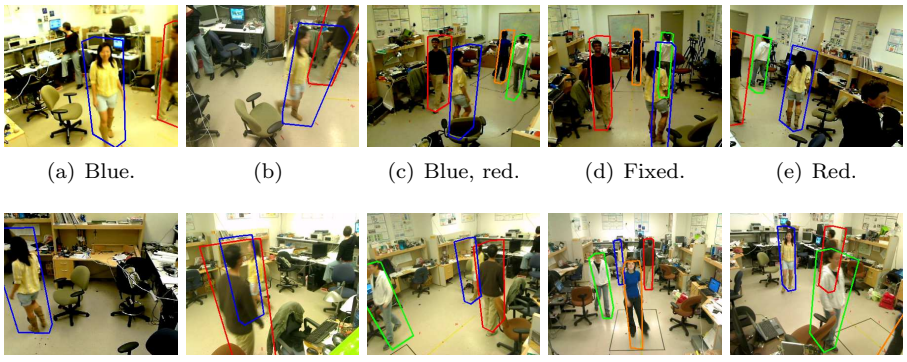
(a) (b)

Figure 6.8: (a) Number of target losses and (b) average RMSE for different selection schemes as a function of the size limit of the selected camera sets for the second environment.



(a) (b)

Figure 6.9: (a) Number of target losses and (b) average RMSE for different selection schemes as a function of the size limit of the selected camera sets for the third environment.



(a) Blue. (b) (c) Blue, red. (d) Fixed. (e) Red.

(f) (g) (h) (i) Fixed, red. (j) Fixed, blue.

Figure 6.10: Camera views in the third environment. The views which are part of the fixed three camera selection, or of the selection of three cameras determined using the proposed method for the person with the blue or red wire frame are marked by fixed, blue and red respectively.

Fig. 6.10 shows a tracking result in this environment using a set of three cameras, selected using the proposed method. The camera views in this environment are very diverse. Some cameras provide an overview of the scene, e.g., Fig. 6.10d and i, whereas some focus on a small part of it, e.g., Fig. 6.10a, b and f. A random selection of cameras in such a setup often leads to poor tracking results. The best performing fixed camera set includes the overview views of Fig. 6.10d, i and j. The proposed method has more flexibility and can also take advantage of the close-up views. E.g., for the person with the blue wire frame the algorithm selected the views of Fig. 6.10a, c and j, and for the one with the red wire frame the views of Fig. 6.10c, e and i. Note that, in order to yield accurate localization, the selected cameras have very different viewing angles. Views with the same viewing direction but that look from different sides (such as Fig. 6.10c and Fig. 6.10g) are not selected simultaneously. E.g., for the red wire frame, in combination with the view of Fig. 6.10c the view of Fig. 6.10e and not of Fig. 6.10g is selected, even if in Fig. 6.10e the person is only partially visible.

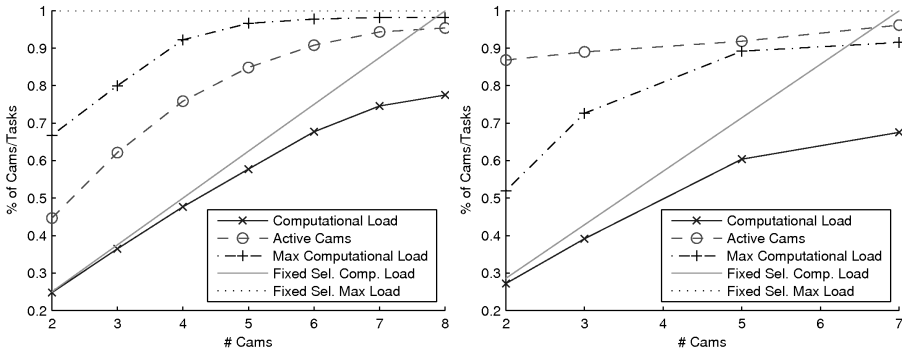
6.6.6 Impact on Computation and Communication

We now assess the computational load at the camera side of the network in the experiments of the previous Section 6.6.5. Let λ denote the computational load of one camera collecting observations for tracking one target, i.e., for calculating *pm*, *fpm*, *vpm*, *Nocclu*, *Occu*, *Cent* and *Cd* for this target. For ease of comparison, in this section we disregard the dependence of λ on the image content and we assume that λ is a fixed number. Let there be T targets t and let the camera set selected for tracking t be denoted as S_t. The computational

load Λ_i of a camera i is then:

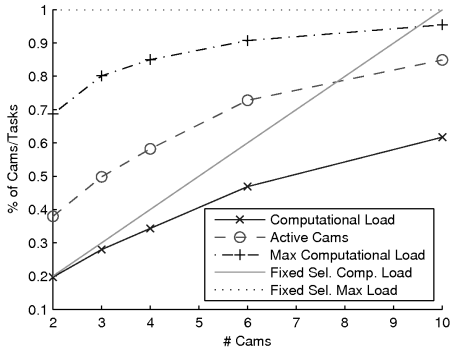
$$\Lambda_i = \sum_{t|i \in S_t} \lambda. \tag{6.18}$$

The average computational load of the cameras is $\sum_{i=1}^{N} \Lambda_i / N$.
When all targets are tracked with all cameras, $\Lambda_i = T\lambda$ for all cameras $i \in [1, N]$. We express the average computational load of the cameras as a fraction of the computational load of a camera when all targets are tracked with all cameras: $\sum_{i=1}^{N} \Lambda_i / (NT\lambda)$. In Fig. 6.11 we evaluate how this expression evolves as a function of the size limit imposed on the selected camera sets for the different environments for the proposed camera selection method. When a fixed set of cameras is chosen, this expression is simply the ratio of the size of the selected set and the number N of cameras. This curve is also plotted in Fig. 6.11.
As can be expected, we observe that for small sets the average computational load on the cameras is almost the same when the set is selected using the proposed algorithm as when it is fixed. For larger sets, it often happens that the proposed method selects fewer cameras than the imposed set size limit because none of the non-selected cameras increase the set suitability value. We therefore observe a relative drop in computational load as compared to the fixed set scenario for larger sets.
In Fig. 6.11 we have also plotted for the proposed camera selection algorithm the average fraction of active cameras in each environment as a function of the size limit of the selected sets. Recall that we call a camera *active* as soon as it is selected to track at least one target, i.e., as soon as $\Lambda_i > 0$. For the first and third environment which contain fewer targets than cameras this fraction depends heavily on the number of selected cameras per target. Especially for small sets the fraction of active cameras is a lot smaller than 1 and hence a lot of communication with cameras can be saved. For large set sizes, approximately 80 to 90% of all cameras are active. In the second environment, where twelve people are tracked, even for small selected sets all but one camera are active. This one camera is pointing at the right half of the court and is therefore mostly inactive. Note that the computational load of each active camera Λ_i is on average always only a fraction of what it would be when all targets are tracked by all cameras. For the fixed set scenario the number of active cameras always equals the set size.
Also plotted for the proposed camera selection method as a function of the size limit of the selected sets is the average maximal instantaneous computational load of the cameras as a fraction of the maximum possible load, i.e., $\max_{i \in [1,N]} \Lambda_i / T$. In the fixed set scenario the chosen cameras always track all targets and this fraction is always one. It is also always one when all cameras track all targets. For the second environment we note that the large number of active cameras at each time instant entails a spread of the computational load over the cameras. Indeed, the maximal computational load measured on the cameras in the network is substantially lower than when tracking all targets



(a) First environment

(b) Second environment



(c) Third environment

Figure 6.11: Average computational load of the cameras, average number of active cameras and average maximal instantaneous computational load for the proposed method and for fixed sets measured in the experiments of Section 6.6.5.

with all cameras or with a fixed set of cameras. For the other environments a similar drop in average maximal load can be observed, especially for small sets.

6.7 Conclusion

A crucial component in an effective camera selection system is quantifying the contribution of one or more cameras to the accomplishment of a task. In this chapter we have presented a novel, general framework to evaluate the quality with which a subset of cameras accomplishes a network task. The proposed set suitability value is derived from the Dempster-Shafer theory of evidence and can be applied to a wide range of vision problems.
As a proof of concept, we have used it for sensor selection in a camera network in which multiple targets are tracked. This method has been tested on thousands of frames in different environments and allows to track persons using as little as three cameras with the same accuracy as when using all available seven, eight or ten cameras. When tracking with two cameras, there is only a minor performance drop. The proposed method clearly outperforms other camera selection schemes for tracking.
The quantification of the contribution of a camera set to a task offers an instrument to distribute multiple tasks among cameras according to some practical criteria, while controlling the associated changes in the quality with which the tasks are performed. Interesting constraints for practical camera networks involved in the execution of multiple tasks include limiting the instantaneous load of a camera, and limiting per frame the number of cameras that need to communicate observation data, such that a frame rate goal can be achieved. To increase the battery life time of the entire camera network, it would be useful to spread the computational and communication load equally over all cameras across time. This could for example be achieved by associating the selection of a camera for a task with a cost that varies with the remaining battery life of the camera.
The impact of limiting the instantaneous load of a camera on the quality of the performed network tasks has been assessed in the thesis of my colleague Marleen Morbee. Studying the other practical constraints would be useful future research to expand the work presented in this chapter.

7

Conclusions

This thesis has dealt with fusion and selection of information in visual systems. The developed algorithms evolved from techniques for visual data selection and fusion at the pixel level to methods for reasoning about the importance of observations and ways of combining them into a useful output product at a higher level of abstraction.

7.1 Overview of Contributions

We have considered two types of visual systems: conventional light microscopes and camera networks.

7.1.1 Depth of Field Extension in Microscopy

A conventional light microscope has a limited depth of field. For this reason, it is often not possible to acquire an image of a 3D object in which all parts of the object appear in focus. A standard technique to virtually extend the depth of field of a microscope is to record an image 'stack' of a 3D object. The distance between the image sensor and the object varies in each image, such that a set of images called slices is obtained in which each time a different part of the object is in focus. Clearly this technique results in an image stack that contains very useful information (sharp images of all object parts), but unfortunately also a lot of information that is irrelevant, namely blurred image regions, or redundant, i.e., sharp image regions that appear in several slices of the stack.
The storage, processing and transmission of this irrelevant and/or redundant data leads to a waste of resources such as storage capacity, processing power and transmission bandwidth. Reducing irrelevance and redundancy in the data is therefore of paramount importance.
In Chapter 2 we have proposed a technique for selecting and fusing all information of interest in an image stack for depth of field extension into a single output image that contains all in-focus parts of the object. More precisely we have exploited the directional sensitivity of the curvelet transform to produce

high quality fusion results, both on real microscopy data and on artificially generated image stacks. The average performance gain over our test set is 3.23 dB over the state-of-the-art complex wavelet-based technique of [Forster et al., 2004] and 7.88 dB over the common pixel domain variance-based method. Moreover, we have shown that adding consistency and spatial smoothness checks to this curvelet-based image fusion method generally leads to better fusion results. For real test data, imposing these constraints leads to a reduced number of artifacts in the fused image.
Additionally, we have hinted at the potential this method holds as a depth from defocus technique by identifying which slice contains a sharp image of each object part.
Noise, present in all image capturing systems, has a disturbing effect on the proposed image fusion technique. In Chapter 3 we have proposed several solutions to temper its influence on the fusion process. We have shown that imposing the assumptions of spatial smoothness within and consistency between the curvelet decomposition sub-bands has a regularizing effect and improves the fusion quality. We have also pointed out that denoising the slices in the curvelet domain prior to fusion is an alternative solution.
In order to develop a curvelet-based denoiser, we have investigated the differences in statistical behavior between curvelet coefficients containing a significant noise-free component and those in which no signal of interest is present. We have developed the *ProbShrinkCurv* denoising method for curvelets, which is an adaptation of the wavelet-based *ProbShrink* denoising method [Pižurica and Philips, 2006]. To this end, we have put the knowledge gained from our statistical study to use in the design of an appropriate local spatial activity indicator (LSAI) for this new method.
ProbShrinkCurv outperforms its wavelet-based counterpart and produces results that are both visually competitive with and numerically close to those of state-of-the-art denoisers.
Using *ProbShrinkCurv* to denoise the curvelet coefficients of the noise-contaminated slices prior to fusion improves the fusion result considerably. The average gain over our test set amounts to 3.59 dB when no checks are performed and 2.20 dB when smoothness and sub-band consistency are imposed. The best fusion results are obtained when denoising prior to fusion is combined with a fusion process in which spatial smoothness and sub-band consistency constraints are imposed.

7.1.2 Data Fusion and Selection in Camera Networks

Camera networks with overlapping fields of view are the second type of visual systems that we have treated in this PhD work. Because such networks present different views on the same scene, they have substantial advantages over a single fixed viewpoint camera. E.g., in scene monitoring, camera networks can alleviate occlusion problems; in gesture recognition, cues coming from different viewpoints can lead to a more robust decision; in free viewpoint television, the

quality of the rendered intermediate views benefits from a larger number of cameras.

Recent hardware developments have made 'smart cameras' possible. These are cameras with on-board processing and communication hardware. They allow for the construction of more flexible and scalable camera networks because the required image processing can be distributed over the cameras. The collaborative processing of the output data of the smart cameras can take place either in a base station or on one of the cameras.

Data processing in a smart camera network entails some specific challenges. The hardware embedded with the image sensor is usually especially designed for image processing (high degree of parallelization), which is an advantage, but it also has some limitations in terms of memory and processing power. If the amount of output data of the smart cameras is kept low, wireless operation becomes possible. This is an advantage for the flexibility of the system. Battery operation is in this case also desirable, which again restricts the number of computations and data transmissions.

The algorithms for camera networks developed in this thesis have all been designed with a view to their possible implementation in smart camera networks, either as they are or in a modified, more light-weight form. To this end, attention has been paid to issues such as data rates and computational load.

When the cameras in a network observe the same event or subject from different viewing perspectives, this not only increases the amount of useful information. A large part of the data produced by the network is redundant or even irrelevant. To reduce the huge amount of data produced by camera networks to workable proportions, techniques that reduce irrelevance and redundancy in the data are of paramount importance. We have followed two main approaches to tackle this challenge: information fusion, which combines relevant data from different sources into a single output product, and information selection, which identifies which data is most valuable for a specific task.

In Chapter 4 we have focused on the fusion of occupancy data from different cameras to obtain a 2D overview of the occupancy of a scene, called an occupancy map. We have proposed a new method based on Dempster-Shafer based fusion of single view ground occupancy maps to combine this information. Experiments and a comparison with the state-of-the-art show clear improvements in the fused ground occupancy maps in terms of concentration of the occupancy evidence around ground truth person positions. We have also demonstrated the effectiveness of the proposed method in a four camera network operating in real time.

To facilitate the implementation of this method in smart camera networks, we have modified it into a low data rate and low load version. This version requires that the persons in the scene appear sufficiently large in the camera views. If this is the case, cameras can send compact scan-lines of the detected foreground, instead of the full foreground image.

Chapters 5 and 6 consider the problem of selecting data of interest in a camera network. Chapter 5 introduces a practical method to select the best views

for observing people in a scene and their shapes. In Chapter 6 we approach camera selection in a more theoretical way and then apply it to multicamera multiperson tracking.

Chapter 5 presented a method to determine which sensor subset in a smart camera network has the best view on the persons in a scene. It consists of distributed and central processes. To choose an appropriate key camera the algorithm takes into account the number of faces detected by each of the cameras, and the velocity and positions of the objects relative to the viewing direction and viewing angle of the cameras. This principal view can be complemented with additional views that complete the observation and that allow to reconstruct the shape of the people in the scene. To select these additional views we use the occupancy map as a crude 2D shape approximation of the people in the scene.

Moreover, a greedy camera selection algorithm was proposed for real time network operation. Experimental results show that the proposed algorithm provides a performance very close to the optimal results. Also, two different network operation protocols were proposed. The first scheme aims to improve the sensor observation frequency and the second scheme decreases the delay between view observation and image transmission. Experimental results show that the proposed protocols improve observation frequency and latency without degrading much the performance of the 3D shape reconstruction.

A crucial component in an effective camera selection system is quantifying the contribution of one or more cameras to the accomplishment of a task. We have presented a novel, general framework to evaluate the quality with which a subset of cameras accomplishes a network task in Chapter 6. The proposed set suitability value is derived from the Dempster-Shafer theory of evidence and can be applied to a wide range of vision problems.

As a proof of concept, we have used it for sensor selection in a camera network in which multiple targets are tracked. This method has been tested on thousands of frames in different environments and allows to track persons using as little as three cameras with the same accuracy as when using all available seven, eight or ten cameras. When tracking with two cameras, there is only a minor performance drop. The proposed method clearly outperforms other camera selection schemes for tracking.

7.2 Directions for Future Research

The main limitation of the algorithms proposed in this thesis is that they all require controlled circumstances to function properly. This is not a major drawback in microscopy, as there it is relatively easy to control the environment. The algorithms for camera networks, however, would greatly benefit from being versatile and robust against disturbing influences. In their current form, all methods of information selection and fusion in camera networks presented in this thesis

- require the network to be fully calibrated. This means that the internal calibration parameters of the cameras must be known (i.e., the focal distance, pixel density of the image sensor and the pixel coordinates of the optical center), as well as the position of the optical center in the 3D world and the viewing direction. This puts important constraints on the application possibilities of these networks. Calibrating a camera network is time-consuming and requires some skill. Keeping a network calibrated means that the cameras must be mounted very solidly and occasional calibration updates must be performed. Some methods of camera self-calibration exist in literature, but they are only suited for cameras with a large overlap of their viewing ranges. Furthermore they rely on the detection of features in the camera views. Detecting features is not easy if the scene contains large homogeneous regions (such as empty floors and walls);

- assume the frames of the different cameras are synchronized, i.e., captured at the same time instant. In practice, without hardware synchronization, frames are never captured at the exact same time instant. Ideally the capturing time difference must be limited to a fraction of a second. In the absence of major network congestion this is automatically the case in real-time systems with a sufficiently high frame rate. In off-line systems synchronized video streams are more difficult to obtain and require soft- or hardware time synchronization between the cameras during capture;

- rely on the output of a foreground detector. Foreground detection is very sensitive to scene lighting and changes in the scene background. Both must remain more or less constant for the foreground detector to work properly. Some falsely detected foreground patches are naturally filtered out by combining information from different cameras. However, important scene lighting changes or changes in the background disturb all cameras simultaneously and cause the methods to fail.

An important research goal for the future is to develop methods that are more robust in the sense that they also function in less controlled circumstances, such as under varying lighting conditions or slowly changing calibration parameters. A possible way of achieving this is by feeding some system level information back to the basic image processing algorithms. E.g., slow deterioration of the calibration parameters should be detectable at the system level. Based on the observations by a camera network of a single person, the network should deduce one location of the person. If this is not the case, the calibration of the cameras needs updating. If this can be detected, small changes can be accounted for. Another example is that if an entire scene is suddenly detected as being occupied, this hints at problems during the foreground detection process and the parameters of this algorithm should be adjusted accordingly.
For all proposed methods there is room for improvement in the way evidence is gathered. In image fusion for depth of field extension the activity level measurement that indicates if an object part is in or out of focus has been

the subject of ample research. Region-based approaches offer most unexplored possibilities. For occupancy calculation the method proposed in this thesis considers local evidence of the presence of foreground objects. A path to explore is the use of a generative person model in the evidence gathering process. Such a generative model exploits prior knowledge of how a person appears in a camera view. It not only allows to assess if foreground objects are observed where their presence is expected, but also to verify that background is observed everywhere else. This can provide the algorithm with important additional information to improve its performance.
Another important direction for future research is the incorporation of temporal information in the algorithms. Currently all methods operate on a frame by frame basis, processing information from each single frame separately. Filtering approaches surely hold the potential to improve the proposed algorithms by incorporating assumptions about temporal smoothness.
One of the main contributions of this thesis is the development of algorithms that deal with information at the network level. Very little research has been done in this field by the image processing and computer vision research community and a lot of directions for future research are still wide open. An aspect that definitely requires extensive further investigation is the communication schemes to be used in practical camera networks. If wireless communication is considered, this would best be developed within the framework of the ZigBee standard. We have only slightly touched this theme in this dissertation. The development of more autonomous algorithms at the camera level would greatly benefit the flexibility and scalability of camera networks but this also presents some important challenges to the design of the algorithms because not all information is available to all agents at all times.

7.3 Summary of Contributions

To summarize, the main contributions of this thesis are:

- a novel image fusion method to extend the depth of field of optical systems such as conventional light microscopes. This method uses the curvelet transform to distinguish between in-focus and blurred image regions. Using this method we have improved image fusion results for depth of field extension in terms of PSNR by several dBs [Tessens et al., 2007a,b];
- a statistical study of curvelet coefficients, based on which we have presented a novel denoising method, inspired by a recent wavelet domain method *ProbShrink*. The new method outperforms its wavelet-based counterpart and produces results that are close to those of state-of-the-art denoisers [Tessens et al., 2006b,c, 2008c]). This denoising method has been shown to improve fusion results on image stacks that are contaminated with noise;
- a novel method to calculate ground occupancy maps by fusing ground occupancies from each view separately according to the Dempster-Shafer

theory of evidence. The method yields very accurate occupancy detection results and in terms of concentration of the occupancy evidence around ground truth person positions it outperforms the state-of-the-art probabilistic occupancy map method and fusion by summing [Morbee et al., 2008, 2010a; Tessens et al., 2008b];

- a novel method to effectively select camera views for observing people in a scene and reconstructing their 3D shape in a network of smart cameras. Only low data rate information is required to be sent over wireless channels since the image frames are locally processed by each sensor node before transmission [Lee et al., 2008; Tessens et al., 2008b];

- a novel, general framework to quantify the quality with which a subset of cameras accomplishes a network task. This is a crucial component in effective sensor selection schemes. The proposed set suitability value is derived from the Dempster-Shafer theory of evidence and can be applied to a wide range of vision problems. We have used this method for sensor selection in camera networks in which multiple people are tracked. The proposed method clearly outperforms other camera selection schemes for tracking in terms of average position error and number of target losses [Tessens et al., 2010].

In total, the research during this PhD resulted in two publication in international peer-reviewed journals [Morbee et al., 2010a; Tessens et al., 2008c]. One article is under review [Tessens et al., 2010] and one in preparation [Morbee et al., 2010b]. A patent application has been submitted [Morbee and Tessens, 2010]. Furthermore thirteen conference papers have been published at international conferences [Lee et al., 2008; Morbee et al., 2007a,b, 2008, 2009; Soleimani et al., 2010; Tessens et al., 2006a,b,c, 2007a,b, 2008b, 2009].

References

Abellan, J. and Moral, S. (2000). A non-specificity measure for convex sets of probability distributions. *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, 8(3):357 – 367.

Abramovich, F., Sapatinas, T., and Silverman, B. (1998). Wavelet thresholding via a bayesian approach. *J. R. Statist. Soc. B*, 60:725–749.

Abramovich, F. and Sapatinas, T. (1999). Bayesian approach to wavelet decomposition and shrinkage. In Muller, P. and Vidakovic, B., editors, *Bayesian Inference in Wavelet Based Models*, volume 141 of *Lecture Notes in Statistics*, pages 33–50. Springer-Verlag, New York.

Abramovich, F., Besbeas, P., and Sapatinas, T. (2002). Empirical bayes approach to block wavelet function estimation. *Computational Statistics and Data Analysis*, 39(4):435 – 451.

Alahi, A., Boursier, Y., Jacques, L., and Vandergheynst, P. (2009). Sport players detection and tracking with a mixed network of planar and omnidirectional cameras. In *Proceedings of ACM/IEEE ICDSC*, pages 1–8, Como, Italy.

Alecu, A., Munteanu, A., Pižurica, A., Philips, W., Cornelis, J., and Schelkens, P. (2006). Information-theoretic analysis of dependencies between curvelet coefficients. In *Proceedings of the ICIP*.

Boie, R. A. and Cox, I. J. (1992). An analysis of camera noise. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(6):671–674.

Boubchir, L. and Fadili, J. M. (2005a). Modélisation statistique multivariée des images dans le domaine de la transformée de curvelet. In *Proceedings - 20th GRETSI Symposium on Signal and Image Processing*, pages 233 – 236.

Boubchir, L. and Fadili, J. M. (2005b). Multivariate statistical modeling of images with the curvelet transform. In *Proceedings - 8th International Symposium on Signal Processing and its Applications, ISSPA 2005*, volume 2, pages 747 – 750, Sydney, Australia.

Bramberger, M., Rinner, B., and Schwabach, H. (2005). A method for dynamic allocation of tasks in clusters of embedded smart cameras. In *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 2595 – 2600, Waikoloa, HI, United States.

Candès, E. J. and Donoho, D. L. (1999). Ridgelets: a key to higher-dimensional intermittency. *Phil. Trans. R. Soc. London A.*, pages 2495–2509.

Candès, E. J., C, E. J., and Donoho, D. L. (2000). *Curves and Surfaces*, chapter Curvelets - A Surprisingly Effective Nonadaptive Representation For Objects with Edges, pages 105–120. Vanderbilt University Press.

Candès, E. (2001). The curvelet transform for image denoising. In *IEEE International Conference on Image Processing*, volume 1, pages 7 –, Thessaloniki.

Candès, E. J. and Donoho, D. L. (2004). New tight frames of curvelets and optimal representation of objects with piecewise C2 singularities. *Commun. Pure and Appl. Math.*, 57:219–266.

Candès, E., Demanet, L., Donoho, D., and Ying, L. (2006). Fast discrete curvelet transforms. *Multiscale Modeling and Simulation*, 5(3):861–899.

Chang, S. G., Yu, B., and Vetterli, M. (2000a). Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532 – 1546.

Chang, S. G., Yu, B., and Vetterli, M. (2000b). Spatially adaptive wavelet thresholding with context modeling for image denoising. *IEEE Transactions on Image Processing*, 9(9):1522 – 1531.

Chipman, H. A., Kolaczyk, E. D., and McCulloch, R. E. (1996). Signal denoising using adaptive bayesian wavelet shrinkage. *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pages 225 – 228.

Chu, M., Haussecker, H., and Zhao, F. (2002). Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. *International Journal of High Performance Computing Applications*, 16(3):293 – 313.

Clyde, M., Parmigiani, G., and Vidakovic, B. (1998). Multiple shrinkage and subset selection in wavelets. *Biometrika*, 85(2):391–401.

da Cunha, A. L., Zhou, J., and Do, M. N. (2006). The nonsubsampled contourlet transform: Theory, design, and applications. *IEEE Transactions on Image Processing*, 15(10):3089 – 3101.

Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. (2007). Image denoising by sparse 3d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8).

Daniyal, F., Taj, M., and Cavallaro, A. (2010). Content and task-based view selection from multiple video streams. *Multimedia Tools and Applications*, 46:235–258.

De Vleeschouwer, C. and Delannay, D. (2009). Basket ball dataset from the European project APIDIS. `http://www.apidis.org/Dataset/`.

Delannay, D., Danhier, N., and De Vleeschouwer, C. (2009). Detection and recognition of sports(wo)men from multiple views. In *Proceedings of ACM/IEEE ICDSC*, pages 1–7, Como, Italy.

Dempster, A. P. (1968). A generalization of Bayesian inference. *Journal of the Royal Statistical Society, Series B*, 30:205–247.

Denoeux, T. (2008). Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence. *Artificial Intelligence*, 172(2-3):234 – 264.

Denzler, J. and Brown, C. M. (2002). Information theoretic sensor data selection for active object recognition and state estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:145–157.

Denzler, J., Zobel, M., and Niemann, H. (2003). Information theoretic focal length selection for real-time active 3-d object tracking. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 400. IEEE Computer Society.

Do, M. N. and Vetterli, M. (2005). The contourlet transform: An efficient directional multiresolution image representation. *IEEE Transactions on Image Processing*, 14(12):2091 – 2106.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455.

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613 – 627.

Donoho, D. (1999). Wedgelets: nearly minimax estimation of edges. *The Annals of Statistics*, pages 859–897.

Ephraim, Y. and Malah, D. (1984). Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-32(6):1109 – 1121.

Ercan, A. O., Gamal, A. E., and Guibas, L. (2006). Camera network node selection for target localization in the presence of occlusions. In *In SenSys Workshop on Distributed Cameras*, pages 1–6.

Ertin, E., Fisher, J. W., and Potter, L. C. (2003). Maximum mutual information principle for dynamic sensor query problems. *LNCS*, 2003(2634):405 – 416.

Feris, R., Tian, Y.-L., and Hampapur, A. (2007). Capturing people in surveillance video. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, United States.

Fleuret, F., Berclaz, J., Lengagne, R., and Fua, P. (2008). Multi-camera people tracking with a probabilistic occupancy map. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(2):267–282.

Forster, B., Van De Ville, D., Berent, J., Sage, D., and Unser, M. (2004). Complex wavelets for extended depth-of-field: A new method for the fusion of multichannel microscopy images. *Microscopy Research and Technique*, 65(1-2):33–42. http://bigwww.epfl.ch/publications/forster0404.html.

Guerrero-Colon, J. A. and Portilla, J. (2005). Two-level adaptive denoising using gaussian scale mixtures in overcomplete oriented pyramids. In *Proceedings - International Conference on Image Processing, ICIP*, volume 1, pages 105 – 108, Genova, Italy.

Gupta, A., Mittal, A., and Davis, L. S. (2007). Cost: An approach for camera selection and multi-object inference ordering in dynamic scenes. *Computer Vision, IEEE International Conference on*, 0:1–8.

Irie, K., McKinnon, A. E., Unsworth, K., and Woodhead, I. M. (2008). A technique for evaluation of ccd video-camera noise. *IEEE Trans. Circuits Syst. Video Techn.*, 18(2):280–284.

Isard, M. and Blake, A. (1998). Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28.

Isler, V. and Bajcsy, R. (2005). The sensor selection problem for bounded uncertainty sensing models. In *2005 4th International Symposium on Information Processing in Sensor Networks, IPSN 2005*, volume 2005, pages 151 – 158.

Jansen, M. and Bultheel, A. (2001). Empirical bayes approach to improve wavelet thresholding for image noise reduction. *Journal of the American Statistical Association*, 96(454):629–639.

Jiang, H., Fels, S., and Little, J. (2008). Optimizing multiple object tracking and best view video synthesis. *IEEE Transactions on Multimedia*, 10(6):997–1012.

Johnstone, I. M. and Silverman, B. W. (2005). Empirical bayes selection of wavelet thresholds. *The Annals of Statistics*, 33(4):1700–1752.

Jousselme, A.-L., Grenier, D., and Éloi Bossé (2001). A new distance between two bodies of evidence. *Information Fusion*, 2(2):91 – 101.

Kaewtrakulpong, P. and Bowden, R. (2001). An improved adaptive background mixture model for real-time tracking with shadow detection. In *Proceedings of 2nd European Workshop on Advanced Video Based Surveillance Systems*, volume 5308, pages 149–158.

Kelly, P., Ó Conaire, C., Kim, C., and O'Connor, N. E. (2009). Automatic camera selection for activity monitoring in a multi-camera system for tennis. In *Third ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–8, Como, Italy.

Klir, G. (1991). Generalized Information-Theory. *Fuzzy Sets and Systems*, 40(1):127–142.

Klir, G. and Wierman, M. J. (1999). *Uncertainty-based information: elements of generalized information theory.* Physica-Verlag/Springer-Verlag, Heidelberg and New York.

Laurentini, A. (1994). The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(2):150–162.

Lee, H., Tessens, L., Morbee, M., Aghajan, H., and Philips, W. (2008). Suboptimal camera selection in practical vision networks through shape approximation. In *Advanced Concepts for Intelligent Vision Systems 2008*, pages 266–277.

Li, H., Manjunath, B., and Mitra, S. (1995). Multisensor image fusion using the wavelet transform. *Graphical Models and Image Processing*, 57(3):235 – 245.

Li, L., Huang, W., Gu, I. Y. H., and Tian, Q. (2003). Foreground object detection from videos containing complex background. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 2–10, New York, NY, USA. ACM.

Li, S., Kwok, J.-Y., Tsang, I.-H., and Wang, Y. (2004). Fusing images with different focuses using support vector machines. *Neural Networks, IEEE Transactions on*, 15(6):1555 –1561.

Li, Y. and Bhanu, B. (2009). Task-oriented camera assignment in a video network. In *International Conference on Image Processing*, pages 3473–3476.

Lienhart, R. and Maydt, J. (2002). An extended set of haar-like features for rapid object detection. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–900–I–903 vol.1.

Liu, J., Reich, J., and Zhao, F. (2003). Collaborative in-network processing for target tracking. *Eurasip Journal on Applied Signal Processing*, 2003(4):378 – 391.

Mallat, S. (1998). *A Wavelet Tour of Signal Processing, Chapter X, Section 10.2.* Academic Press.

Maloney, L. (1999). *Physics based approaches to modeling surface color perception*, pages 387–416. Cambridge University Press.

Matsui, T., Matsuo, H., and Iwata, A. (2001). Dynamic camera allocation method based on constraint satisfaction and cooperative search. In *Proceedings of 2nd International Conf. on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, volume 8, pages 955–964.

McAulay, R. J. and Malpass, M. L. (1980). Speech enhancement using a soft-decision noise suppression filter. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-28(2):137 – 145.

McIntyre, G. A. and Hintz, K. J. (1996). Information theoretic approach to sensor scheduling. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 2755, pages 304 – 312.

Mihcak, M. K., Kozintsev, I., Ramchandran, K., and Moulin, P. (1999). Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Processing Letters*, 6(12):300 – 303.

Morbee, M., Tessens, L., Prades-Nebot, J., Pižurica, A., and Philips, W. (2007a). A distributed coding-based extension of a mono-view to a multi-view video system. In *3DTV Conference*, pages 1 – 4. Digital Object Identifier 10.1109/3DTV.2007.4379387.

Morbee, M., Tessens, L., Quang Luong, H., Prades-Nebot, J., Pižurica, A., and Philips, W. (2007b). A distributed coding-based content-aware multi-view video system. In *2007 1st ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC*, pages 355 – 362, Vienna, Austria.

Morbee, M., Tessens, L., Lee, H., Philips, W., and Aghajan, H. (2008). Optimal camera selection in vision networks through shape approximation. In *International Workshop on Multimedia Signal Processing*, pages 50–55.

Morbee, M., Tessens, L., Kleihorst, R., Aghajan, H., and Philips, W. (2009). Phd forum: Dempster-shafer based camera contribution evaluation for task assignment in vision networks. In *Proceedings of the Third ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–2.

Morbee, M. and Tessens, L. (2010). An optical system for occupancy sensing, and corresponding method. EPO Patent Application EP10164483.9.

Morbee, M., Tessens, L., Aghajan, H., and Philips, W. (2010a). Dempster-shafer based multi-view occupancy maps. *IET Electronic Letters Journal*, 46.

Morbee, M., Tessens, L., Aghajan, H., and Philips, W. (2010b). Dempster-Shafer based task assignment in vision networks. *In preparation*, x(x):xxx–xxx.

Moulin, P. and Liu, J. (1999). Analysis of multiresolution image denoising schemes using generalized gaussian and complexity priors. *IEEE Transactions on Information Theory*, 45(3):909 – 919.

Munoz-Salinas, R., Medina-Carnicer, R., Madrid-Cuevas, F., and Carmona-Poyato, A. (2009). Multi-camera people tracking using evidential filters. *International Journal of Approximate Reasoning*, 50(5):732 – 749.

Pahalawatta, P. V. and Katsaggelos, A. K. (2004). Optimal sensor selection for video-based target tracking in a wireless sensor network. In *in Proc. International Conference on Image Processing*, pages 3073–3076.

Park, J., Bhat, P. C., and Kak, A. C. (2006). A look-up table based approach for solving the camera selection problem in large camera networks. In *Proc. of Workshop on Distributed Smart Cameras, in conjunction with ACM SenSys06*, pages 1–5.

Petrovic, V. and Xydeas, C. (2000). On the effects of sensor noise in pixel-level image fusion performance. In *Proceedings of the 3rd International Conference on Information Fusion*, volume 2, pages 14–19, Paris, France.

Pižurica, A. and Philips, W. (2006). Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising. *IEEE Transactions on Image Processing*, 15(3):654 – 665.

Po, D. D.-Y. and Do, M. N. (2006). Directional multiscale modeling of images using the contourlet transform. *IEEE Transactions on Image Processing*, 15(6):1610 – 1620.

Portilla, J., Strela, V., Wainwright, M. J., and Simoncelli, E. P. (2003). Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338 – 1351.

Quost, B., Denoeux, T., and Masson, M.-H. (2008). Adapting a combination rule to non-independent information sources. In *Proc. of IPMU'08*, pages 448–455.

Roberts, D. R. and Marshall, A. D. (1998). Viewpoint selection for complete surface coverage of three dimensional objects. In *Proc. of the British Machine Vision Conference (BMVC)*, Southampton, England.

Romberg, J. K., Choi, H., and Baraniuk, R. G. (1999). Bayesian tree-structured image modeling using wavelet-domain hidden markov models. *Proceedings of SPIE - The International Society for Optical Engineering*, 3816:31 – 44.

Rowaihy, H., Eswaran, S., Johnson, M., Verma, D., Bar-Noy, A., Brown, T., and La Porta, T. (2007). A survey of sensor selection schemes in wireless sensor networks. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 6562.

Schmaedeke, W. and Kastella, K. (1998). Information based sensor management and IMMKF. In *Proceedings of SPIE - The International Society for Optical Engineering*, volume 3373, pages 390 – 401.

Sendur, L. and Selesnick, I. W. (2002). Bivariate shrinkage with local variance estimation. *IEEE Signal Processing Letters*, 9(12):438 – 441.

Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press.

Simoncelli, E. P. and Adelson, E. H. (1996). Noise removal via bayesian wavelet coring. *IEEE International Conference on Image Processing*, 1:379 – 382. Bayesian wavelet coring;Wiener filter;.

Smets, P. (1992). The concept of distinct evidence. In *IPMU'92*, pages 789–794.

Snidaro, L., Niu, R., Varshney, P. K., and Foresti, G. L. (2003). Automatic camera selection and fusion for outdoor surveillance under changing weather conditions. In *Proc. of the IEEE Conference on Advanced Video and Signal Based Surveillance*, page 364.

Soleimani, S., Rooms, F., Tessens, L., and Philips, W. (2010). Image fusion using blur estimation. In *Proc. of the ICIP*.

Sommerlade, E. and Reid, I. (2008). Information theoretic active scene exploration. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1397–1403.

Soro, S. and Heinzelman, W. B. (2007). Camera selection in visual sensor networks with occluding objects. In *Proceedings of ACM/IEEE First International Conference on Distributed and Smart Cameras (ICDSC)*, Vienna, Austria.

Starck, J.-L., Candès, E. J., and Donoho, D. L. (2002). The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670 – 684.

Stauffer, C. and Grimson, W. (2000). Learning patterns of activity using real-time tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):747–757.

Tarabanis, K., Allen, P., and Tsai, R. (1995). A survey of sensor planning in computer vision. *IEEE Transactions on Robotics and Automation*, 11(1):86–104.

Teelen, K. (2010). *Geometric Uncertainty Models for Correspondence Problems in Digital Image Processing*. PhD thesis, Ghent University and University College Ghent.

Tessens, L., Kehl, R., Pižurica, A., Van Gool, L., and Philips, W. (2006a). A real-time optical head tracker based on 3D prediction and correction. In *Proc. of SPS-DARTS 2006 (the second annual IEEE Benelux/DSP Valley Signal Processing Symposium*, pages 39–42.

Tessens, L., Pižurica, A., Alecu, A., Munteanu, A., and Philips, W. (2006b). Modeling curvelet domain inter-band image statistics with application to spatially adaptive image denoising. In *Proceedings of ProRISC - Program for Research on Integrated Systems and Circuits - Workshop 2006*, pages 208–213, Veldhoven, the Netherlands.

Tessens, L., Pižurica, A., Alecu, A., Munteanu, A., and Philips, W. (2006c). Spatially adaptive image denoising based on joint image statistics in the curvelet domain. In Truchetet, F. and Laligant, O., editors, *Wavelet Applications in Industrial Processing IV*, volume 6383 of *1*, page 63830L. SPIE.

Tessens, L., Ledda, A., Pižurica, A., and Philips, W. (2007a). Extending the depth of field in microscopy through curvelet-based image fusion under smoothness and consistency constraints. In *Proc. of SPS-DARTS 2007 (the third annual IEEE Benelux/DSP Valley Signal Processing Symposium*, pages 29–33.

Tessens, L., Ledda, A., Pižurica, A., and Philips, W. (2007b). Extending the depth of field in microscopy through curvelet-based frequency-adaptive image fusion. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume I, pages 861–864, Honolulu, Hawaii, USA. ISSN: 1520-6149 (ISBN: 1-4244-0728-1).

Tessens, L., Morbee, M., Lee, H., Philips, W., and Aghajan, H. (2008a). Principal view determination demos. `http://telin.ugent.be/~ltessens/index.php?id=30`.

Tessens, L., Morbee, M., Lee, H., Philips, W., and Aghajan, H. (2008b). Principal view determination for camera selection in distributed smart camera networks. In *2008 2nd ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC*, pages 225–234, Stanford, USA.

Tessens, L., Pižurica, A., Alecu, A., Munteanu, A., and Philips, W. (2008c). Spatially adaptive image denoising through modeling of curvelet domain statistics. *Journal of Electronic Imaging*, 17(3).

Tessens, L., Morbee, M., Kleihorst, R., Aghajan, H., and Philips, W. (2009). Efficient approximate foreground detection for low-resource devices. In *Proceedings of the Third ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–8.

Tessens, L. and Morbee, M. (2010). Video of occupancy map demonstrator. `http://telin.ugent.be/~ltessens/demoISYSS/`.

Tessens, L., Morbee, M., Aghajan, H., and Philips, W. (2010). Dempster-shafer based camera contribution quantification for sensor selection in vision networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. under review.

Tsang, E. and Voudouris, C. (1998). Constraint satisfaction in discrete optimisation.

USC-SIPI (last visited 11 Nov 2009). Image database, http://sipi.usc.edu/services/database/.

Valdecasas, A., Marshall, D., Becerra, J., and Terrero, J. (2001). On the extended depth of focus algorithms for bright field microscopy. *Micron*, 32:559 – 569.

Vázquez, P.-P., Feixas, M., Sbert, M., and Heidrich, W. (2003). Automatic view selection using viewpoint entropy and its application to image-based modelling. *Computer Graphics Forum*, 22(4):689–700.

Vidakovic, B. (1998). Nonlinear wavelet shrinkage with bayes rules and bayes factors. *Journal of the American Statistical Association*, 93(441):173–179.

Vidakovic, B. and Ruggeri, F. (2001). Bams method: Theory and simulations. *Sankhya*, 63(2):234.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features.

Wang, H., Yao, K., and Estrin, D. (2005). Information-theoretic approaches for sensor selection and placement in sensor networks for target localization and tracking. *Journal of Communications and Networks*, 7(4):438–449.

Yang, D., Shin, J., Ercan, A. O., and Guibas, L. (2004). Sensor tasking for occupancy reasoning in a camera network. In *Proc. of IEEE/ICST Workshop on Broadband Advanced Sensor Networks*.

Yu, C., Soro, S., Sharma, G., and Heinzelman, W. (2007). Lifetime-distortion trade-off in image sensor networks. In *Proceedings of International Conference on Image Processing (ICIP)*, volume V, pages 129–132, San Antonio, Texas, USA.

Zhang, Z. and Blum, R. S. (1999). Categorization of multiscale-decomposition-based image fusion schemes with a performance study for a digital camera application. *Proceedings of the IEEE*, 87(8):1315 – 1326.

Zhao, F., Shin, J., and Reich, J. (Mar 2002). Information-driven dynamic sensor collaboration. *Signal Processing Magazine, IEEE*, 19(2):61–72.

Zitova, B. and Flusser, J. (2003). Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000.