Decomposition of mixed pixels in remote sensing images to improve the area estimation of agricultural fields

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Gebbinck, Maurice Stefan klein

Decomposition of mixed pixels in remote sensing images to improve the area estimation of agricultural fields / Maurice Stefan klein Gebbinck. – [S.l. : s.n.], 1998 (Veenendaal: Universal Press). – X, 165 p. : ill. ; 24 cm. – Thesis Katholieke Universiteit Nijmegen. – With refs., index. – With summary in Dutch.
ISBN 90-9012027-0
NUGI 855
Subject headings: Remote sensing / image processing.

About the cover:

> *An artist's impression of digital image enhancement techniques, in which a clear picture emerges from a chaotic array of pixels, almost as a scared bird that flees from its hidden nest in the undergrowth to the safety of the open skies. The illustration was inspired by the motifs found in the print "Sky and Water II" from the famous Dutch graphic artist M.C. Escher (1898–1972).*

Cover design and illustration by Edwin klein Gebbinck, Renkum, 1998.

Decomposition of mixed pixels in remote sensing images to improve the area estimation of agricultural fields

een wetenschappelijke proeve op het gebied
van de Natuurwetenschappen, Wiskunde en Informatica

Proefschrift

ter verkrijging van de graad van doctor
aan de Katholieke Universiteit Nijmegen,
volgens besluit van het College van Decanen
in het openbaar te verdedigen op
vrijdag 27 november 1998
des namiddags om 1.30 uur precies

door

Maurice Stefan klein Gebbinck

geboren op 4 mei 1969 te Westervoort

Promotor:	prof. dr. F.W. Vaandrager
Co-promotor:	dr. Th.E. Schouten

Manuscriptcommissie:

dr. J.J. Settle	University of Reading, United Kingdom
prof. dr. G. Wilkinson	Kingston University, United Kingdom
prof. dr. ir. M. Molenaar	Landbouw Universiteit Wageningen

Preface

This thesis is the result of nearly 10 years of study, work, and fun at the University of Nijmegen. In 1987 I started my study Informatics at the Faculty of Mathematics and Natural Sciences, not knowing exactly what Informatics was but feeling it had a great future. Four years later, however, I had found out that I was fascinated by the ability of the computer to perform certain tasks that are (nearly) impossible to be executed by human beings. A good example of such a task is the classification of multi-dimensional feature vectors, which I studied extensively during my Master's research period at the Biophysics Laboratory of the University Hospital of Nijmegen. Therefore it is no wonder that, after a brief intermezzo in the military service, I took up a related subject, i.e. the decomposition of mixed pixels, as the basis of my PhD research.

When I started my PhD research in January 1994, this time at the Faculty of Mathematics and Informatics, times were quite turbulent. The Computing Science Institute had just gotten a new structure, the professor leading Informatics for Technical Applications—my to be promoter—had just left, and on top of that there was no clear project description. Following the advice of my supervisor Theo Schouten, I took the PhD thesis of Ron Schoenmakers† as a starting point and identified some open ends in his approach. One of the problems of his segmentation methodology for large optical satellite images was the handling of mixed pixels. Since these pixels each cover parts of several objects, it is not only difficult but also incorrect to assign them to a single segment or class. In an attempt to solve this problem I developed the decomposition technique described in Chapter 3, before finding out that many other and simpler (but less accurate) methods had already been proposed. After this, combining decomposition with Ron's segmentation approach was relatively straightforward, although many problems specific to the processing of mixed pixels emerged on the way and had to be solved.

Now this thesis is nearly finished—the Informatics department has meanwhile become part of the Faculty of Natural Sciences, Mathematics and Informatics—it is time to thank the people who have contributed to this thesis in one way or the other. First and foremost I would like to thank my supervisor and co-promoter Theo Schouten, who has been a stronghold throughout my period as a PhD student. He was never too busy to answer my questions or read and comment on my articles, reports and chapters of this manuscript. Together we did a lot of research on various mixed pixel subjects, which resulted in several chapters of this thesis as well as numerous publications [34, 35, 36, 37, 38, 39, 93, 94, 95], and I can only hope that my work inspired him as much as his work inspired me. I am also indebted to Jeff Settle, who voluntarily read the article I sent him and made some valuable suggestions, which led to the improved decomposition method described in Chapter 3 as well as a joint article [39]. Likewise, I want to thank Zheng Kai Liu for the pleasant

collaboration with Theo and me, which resulted in Chapter 4 and a joint article [95], as well as Hubert Voogd, Jeroen Melchiors, and Ron Grazier, whose Master's projects were directly related to this dissertation; especially Ron's contribution—see Chapter 5—has been significant. Equally valuable was the contribution of my promoter Frits Vaandrager, who, although the subject of this thesis is not his speciality, read the entire manuscript minutely and provided a lot of useful comments and advice. Last but not least I would like to thank my brother Edwin klein Gebbinck for designing the front cover, even though he was busy writing his own PhD thesis at the same time. “Bedankt jongen, nou jij nog.”

Apart from the people who contributed directly to this thesis, I also want to express my thanks to those who supported me in any way and made the task of writing this thesis somewhat lighter. First of all I want to thank my parents Ton en Aleid for the unconditional love and support they gave me. “Nu ga ik aan een echte baan beginnen.” I am also grateful to my colleagues of Informatics for Technical Applications, Frits, Marieke, Mariëlle, Marco, Ansgar, David, Bart, André, Angelika, and Jan S, those of the Real-time Systems era, Theo, Louis, Mirèse, Frans, Huub, Hanno, Hans, John, Jeroen, and Jan V., as as well as Toine Tax for providing a pleasant working atmosphere. In particular I want to thank Louis “Dr. Loe” Vuurpijl, with whom I shared an office for the first two years. Everybody who heard one of our conversations in our own style, listened to one of our songs with our own lyrics, or saw the way we encouraged our computers to work faster, must have thought we were completely nuts. In any case, Loe, it was great fun having you around and you motivated me a lot to finish my thesis quickly. Also thanks to Albert, Paul K., and Paul J. for their interesting company at the lunch table, and to Joost, my oldest friend in Nijmegen, who never stops to surprise me. Besides work I also spent a lot of time at the Nijmegen student athletics club ’t Haasje, either following the mawo-, S4-, or S2-training or participating in one of the relay races during the weekend. Many thanks to Joski, Dirk, Janneke, Cheriel, Jasmijn, Kim, Werner, Ernest, Sander, Leonie, Ursula, Jan L., Wilco, Lonneke, Rinus, Paulien, Mieke, Sandra, Yvonne, Jerry, Rob, and all others I forgot to mention. Also thanks to the “mietjes” of the survival group, Aloys “Flying Al”, Gaby “Haile Gebre” and Arthur “Attoe el Toro”; boys, it was fun running through the forest carrying tree-stems and watching the astonished faces of innocent passers-by. Furthermore I want to thank Engelbert, Robert, and Reinaud for the interesting conversations we had after our S4-training, drinking a few “bokjes”, or during one of our monthly S4-meals. Special thanks to Engelbert, with whom I did a lot of S2-training to prepare for the marathons of Eindhoven and Terschelling. Together we must have explored every forest trail for 10 miles around Nijmegen, which was an unforgettable experience. Finally, I want to thank my closest friends, Harvey, René, Remy, Vincent, Karel, Kees and Marco, also known as “de Jongens van Van Zanten”, “de harde kern”, “de white-nighters”, “de super-harde-white-night-zuipkern”, “de Nijmegen kern”, or whatever name is used nowadays. Although they caused me a lot of headaches by taking me to “De Fuik” on Sunday evenings or to “Basje” on arbitrary working days, frequently distracted me from work by calling me for no reason or sending me silly e-mails, and they definitely did not contribute anything to this thesis whatsoever, life would have been not nearly as joyful without them. “Vrienden, bedankt, enne, over 2 jaar ben ik weer terug.”

Ispra, August 30, 1998 Maurice klein Gebbinck

Contents

Chapter 1

Introduction

In this chapter the scientific motivation for writing—and hopefully also for reading—this thesis is given. First it is explained why area estimation of agricultural fields is so important. Next, after a brief introduction to the subject of remote sensing in Section 1.2, some basic image classification principles upon which area estimation is normally based are described. In addition, Section 1.3 explains why the classification approach fails to give an accurate area estimation when dealing with mixed pixels. In the last section, finally, the aims of this thesis are formally stated and the contents of and relationships between the remaining chapters are summarised.

1.1 Area estimation of agricultural fields

For the European Union (EU), it is of great importance to be able to estimate the area of an agricultural field in order to manage its agricultural subsidy system. Under this system, which was introduced by McSharry in 1992, farmers can apply for an income subsidy, whose height is based on the type and acreage of the crops they grow. This way, approximately 75 billion guilders ($ 37,000,000,000) is paid to farmers throughout the EU every year [16]. It is estimated that about 5 billion guilders of this sum is paid wrongly because of applications that were filled out incorrectly [16]. One possibility to commit fraud, for example, is to list the crop type of a field as cole-seed, which is heavily subsidised but is not very profitable, while in fact potatoes, which are not subsidised but return a reasonable profit, are grown. More often, however, a farmer either deliberately or accidentally overestimates the area of (some of) his agricultural fields, for instance by including the surrounding ditches. To reduce the huge amount of money spent on incorrect applications, a reasonable fraction of the subsidy claims must be checked. In order to do so, an accurate estimation of the area covered by each crop grown by the farmer under observation is required.

Apart from combating fraud, the field area estimates can also be used for budget control. By summing the area of all fields of a particular crop over an entire region, a forecast of the total yield per crop can be made. Based on this forecast, the height of the subsidy for each crop type can be determined, and the amount of money that has to be spent on these subsidies can be predicted. Therefore, an accurate estimation of the area of an agricultural field is necessary for the management of the EU's agricultural subsidy system.

In order to operate effectively, the crop area estimation has to be automated. According to the McSharry regulation, the member states are obliged to check at least 5% of all subsidy claims [4]. Since the EU counts approximately 6 million farmers, this task is too large to be carried out by human inspectors on site [16]. A further complication is that on many of the 150 million parcels that are in use several crops are grown in succession each year, making a regular inspection necessary. To cope with these problems, the EU has opted for an automated system based on satellite remote sensing to support the inspectors in their task. Since each remotely sensed image covers a large region, many claims can be investigated in a relatively cost-effective way; only those fields that are marked for possible fraud need to be visited by an inspector, who performs the final check. Furthermore, the inspection can be performed regularly because the satellite passes over the same area on a frequent basis. It is because of these properties that an automated system based on satellite remote sensing is ideally suited for crop area estimation.

1.2 Introduction to remote sensing

The field of remote sensing is very broad and has been described from many angles by numerous authors, e.g. Campbell [19], Lillesand and Kiefer [66], and Sabins [88]. In his book, Campbell tried to identify the central concepts of remote sensing and came up with the following definition:

> *Remote sensing is the practice of deriving information about the earth's land and water surfaces using images acquired from an overhead perspective, using electromagnetic radiation in one or more regions of the electromagnetic spectrum, reflected or emitted from the earth's surface.*

Although this definition certainly does not cover all areas (e.g. meteorological or extra-terrestrial remote sensing), it does serve well as a description of remote sensing in this thesis. In the remainder of this section, several of the above principles will be explained and some additional concepts necessary to understand this thesis will be introduced.

As stated above, remote sensing makes use of electromagnetic radiation. The strongest and best-known source of electromagnetic radiation is our sun, which emits radiation over the entire electromagnetic spectrum—see Table 1.1. Besides this natural source of illumination, which is used for *passive remote sensing*, it is also possible to use an artificial source of electromagnetic radiation, in which case we speak of *active remote sensing*. When the radiation reaches the surface of the earth, part of it will be reflected. Another part will be absorbed and subsequently emitted, mainly in the form of thermal (far infrared) energy. The fraction of the irradiance that is reflected (or absorbed and re-emitted) is dependent on the wavelength and differs for each material, as is illustrated in Figure 1.1. By measuring the amount of electromagnetic radiation that is reflected or emitted and comparing it to the spectral reflectance curves of known materials, information about the earth's land and water surfaces can be derived.

To measure the reflected and emitted radiation, usually an imaging scanner aboard an airplane or satellite is used. Basically, the two types of passive imaging scanners that exist are the mechanical scanner and the pushbroom scanner. The *mechanical scanner*

Division		Wavelengths
Gamma rays		<0.03 nm
X-rays		0.03–3.0 nm
Ultraviolet		3.0–380 nm
Visible		0.38–0.72 μm
Blue		*0.4–0.5 μm*
Green		*0.5–0.6 μm*
Red		*0.6–0.7 μm*
Infrared		0.72–1000 μm
Near infrared	reflected	0.72–1.30 μm
Mid infrared	reflected	1.30–3.00 μm
Far infrared	emitted	3.00–1000 μm
Microwave		0.1–30 cm
Radio		≥30 cm

Table 1.1: *Principal divisions of the electromagnetic spectrum (Campbell [19], corrected).*

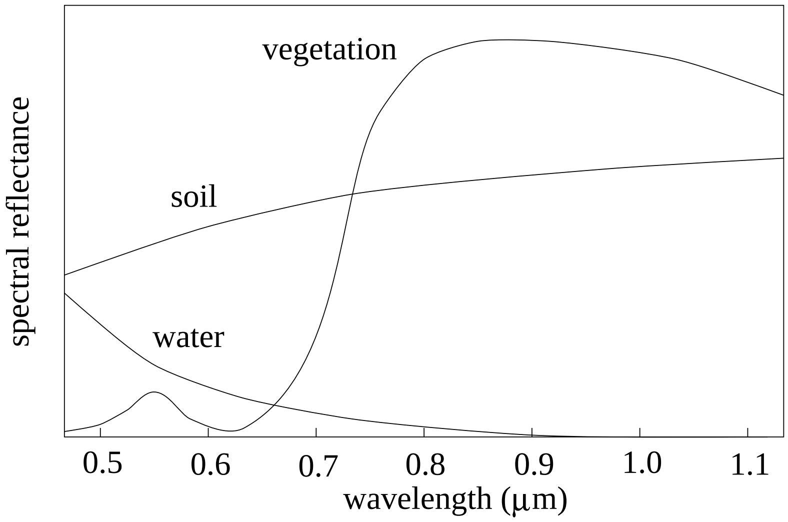


Figure 1.1: *Generalised spectral reflectance curves for three different materials (adopted from Schowengerdt [98]).*

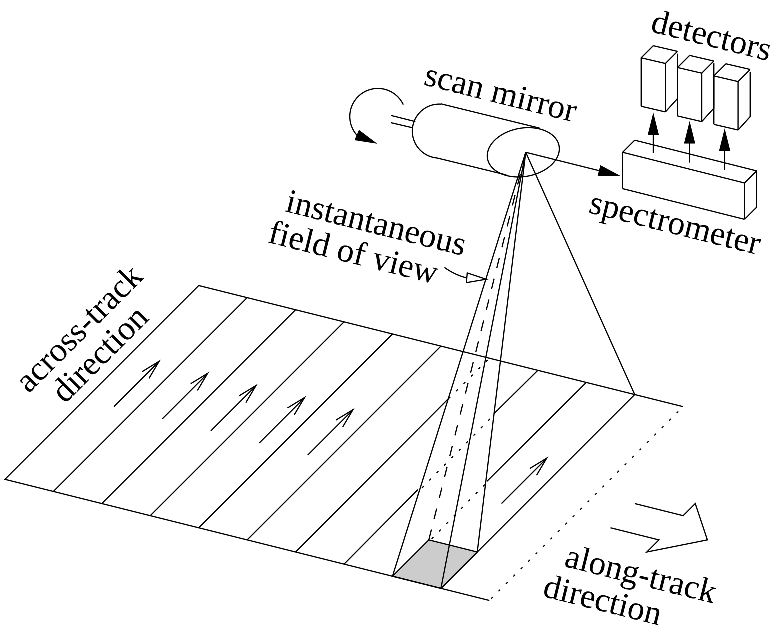


Figure 1.2: *A multispectral mechanical scanner uses a scan mirror to direct the radiation inside the instantaneous field of view towards a spectrometer. Here, the incoming energy is dispersed into a spectrum and led to detectors that are sensitive to specific wavelength bands. Rotation of the scan mirror moves the IFOV across-track, while the along-track movement is provided by the platform motion (after Sabins [88]).*

uses a scan mirror to direct the surface radiation onto an electronic detector, taking a measurement at regular intervals. Figure 1.2 shows an example of a multispectral mechanical scanner. The *pushbroom scanner* uses a linear array of detectors—usually CCDs[1]—to take a number of measurements simultaneously. Apart from these across-track readings, both scanners also take measurements in the along-track direction, which is defined by the platform's motion. With some effort, this two-dimensional grid of measurements can be transformed to a digital image consisting of picture elements or *pixels*. Not only do the corresponding ground locations of the measurements have to be corrected due to factors like the earth's curvature and irregular movements of the scan mirror and the platform (geometric corrections), but the measurements themselves must also be corrected for atmospheric and sensor effects (radiometric corrections). The resolution of the resulting image or series of images, which expresses the level of fine detail that can be distinguished, has four aspects. The *spatial resolution* is the ground area that is represented by a single pixel; this area is approximately equal to the geometrical projection of a single detector element at the earth's surface, which is sometimes called the *instantaneous field of view* (IFOV). The *radiometric resolution* is defined by the number brightness levels that can be distinguished and is dependent on the number of bits into which each measurement is quantified. The *spectral resolution* denotes the width of the wavelength interval at which the electromagnetic radiation is recorded. If a *multispectral* (e.g. Thematic Mapper) or *hyperspectral scanner* (e.g. AVIRIS) is used, which take measurements in a few up to

[1]Charge-Coupled Device.

Scanner	TM[a]	AVHRR[b]	AVIRIS[c]
Platform	Landsat-4/5 satellite	NOAA[d]-7/9/11/12/14 satellite	NASA[e] ER-2 airplane
Scene coverage	185x170 km^2	2399x4752 km^2	11x10 km^2
Image size	6167x5667 pixels	2048x4320 pixels	614x512 pixels
Resolution			
• spatial	$30x30$ m^2 [f]	$1.1x1.1$ km^2	$20x20$ m^2
• radiometric	8 bits	10 bits	12 bits
• temporal	16 days	12 hours	not applicable
• spectral	band 1: 0.45–0.52 μm band 2: 0.52–0.60 μm band 3: 0.63–0.69 μm band 4: 0.76–0.90 μm band 5: 1.55–1.75 μm band 6: 10.4–12.5 μm band 7: 2.08-2.35 μm	band 1: 0.58–0.86 μm band 2: 0.73–1.10 μm band 3: 3.55–3.93 μm band 4: 10.3–11.3 μm band 5: 11.5–12.5 μm	224 bands, each 10 nm wide, over 0.38–2.45 μm

[a] Thematic Mapper.
[b] Advanced Very High Resolution Radiometer, Local Area Coverage (LAC) mode.
[c] Airborne Visible InfraRed Imaging Spectrometer.
[d] National Oceanic and Atmospheric Administration
[e] National Aeronautics and Space Administration
[f] Spatial resolution of band 6 is $120x120$ m^2.

Table 1.2: *Characteristics of a few well-known scanners. The exact specifications may differ for other models carried by different platforms.*

several hundreds of spectral bands, the spectral resolution may well not be unique (c.f. TM bands 3 and 4). The *temporal resolution*, finally, only applies to time series of images and describes how long the interval between two successive recordings of the same scene is. In case the scanner is carried by a satellite, the temporal resolution is determined by the satellite's orbit. The characteristics of a few well-known scanners are listed in Table 1.2.

1.3 Classification of mixed pixels

The most direct approach towards crop area estimation is to classify the remotely sensed image and count the number of pixels per class over the region of interest. Although more refined methods utilising confusion matrices or regression correction exist, see Gallego *et al.* [32] for an overview, they all need an initial estimation of the number of pixels per crop. Therefore, in Section 1.3.1 a brief description of image classification is given; Section 1.3.2 explains why this method does not work for mixed pixels.

1.3.1 Image classification

A well-known way to extract information from an image is *classification*, which is the process of assigning pixels to classes (Campbell [19]). As was explained in the previous section, each pixel (picture element) represents a series of measurements in several spectral bands of the reflectance from a particular ground area. Many of these pixels form groups because they have a uniform brightness in their several spectral bands. These groups are called *spectral classes* and are each characterised by a *spectral signature*, which is the reflectance typical for that class. By assigning each pixel to the class whose spectral signature it resembles most, a classification of the entire image is obtained. However, instead of a pixel's spectral class, a user is more interested in the informational class a pixel belongs to. Whereas spectral classes obviously are inherent to an image, *informational classes* are defined by the user and represent the categories of interest, e.g. crop types in this thesis. Only if it is possible to define links between the spectral classes in the image and the informational classes that are of primary interest, then the image forms a valuable source of information [19].

A useful distinction between classification strategies is the one between supervised and unsupervised classification. *Unsupervised classification* attempts to identify the spectral classes in the image based on the premise that pixels within a given class should be close together in the measurement space, whereas pixels in different classes should be comparatively well separated [90]. One of the more common methods based on this principle is K-means clustering (e.g. see Bishop [13]). Although this procedure requires virtually no human intervention, the analyst must supply a mapping between the spectral and the informational classes afterwards, which, due to natural variations within a class as well as variations in illumination, usually matches several spectral classes with a single informational class. *Supervised classification*, on the other hand, demands much more interaction with the analyst. During the training or calibration stage, the analyst has to define regions on the image that are known—either through fieldwork or because of expert knowledge—to represent certain informational classes. Each of these training areas must be homogeneous with respect to one class, and together the regions must cover the entire range of variability within each information category. Therefore, often subclasses are introduced (e.g. green grass and dry grass), which is related to the aggregation of spectral classes after unsupervised classification. In the subsequent classification stage, all pixels are assigned to the informational (sub)classes they resemble most. A good example of supervised classification is *maximum likelihood classification*, which is further explained in Appendix A.

The accuracy of a classifier can be improved by including the spatial context of a pixel in the classification process. Usually a pixel is assigned to a class solely based on its values measured in the separate spectral bands. This *pixel-based* strategy, however, often results in misclassifications because it is very sensitive to noise and within-class variation. As an alternative, more complex classifiers consider a pixel in the context of its neighbourhood, just like a human interpreter does. This approach allows to take the *texture*—patterns of brightness formed by adjacent pixels—of the landscape into account as well. When dealing with objects like agricultural fields, *field-based classification* can be successfully applied (see for instance Ait Belaid *et al.* [3] or Schotten *et al.* [91]). This type of classifier first needs to determine the field boundaries, either through segmentation of the image (Schoenmakers [90]) or, more directly, by using the geometrical data in a geographical information system (GIS) (Janssen [54]). Next, all pixels belonging to one field are assigned to the same class based on the average reflectance or the majority of the class labels found after classifying the individual pixels. Furthermore, the object can be classified using other features that can be calculated, such as size, shape, and orientation (Stakenborg [106]). Thus, by using a combination of spatial information and domain knowledge, the classification accuracy can be improved.

1.3.2 The mixed pixel problem

In every remotely sensed image, a considerable number of mixed pixels is present. A *mixed pixel* is a picture element representing an area occupied by more than one ground cover type. Basically, there are two situations in which mixed pixels occur. The first case concerns the pixels that are located at the edges of large objects like agricultural fields, for instance. The second case arises when objects are imaged that are relatively small compared to the spatial resolution of the scanner. This can be long linear features such as rivers or highways, but also objects that are small in both dimensions such as farms or ponds, or even bushes in the sparsely vegetated semi-arid rangelands of Australia (see Pech [79]). For a given scanner, the number of mixed pixels greatly depends on the landscape that is imaged. Irons *et al.* [52] reported proportions of probable mixed pixels in TM-images ranging from 29.6% for the category water to 68.3% for grass patches, while Schoenmakers [90] claimed that in some Mediterranean countries of the EU, where the average field size is small, the proportion of mixed pixels can easily be as high as 30%. These figures indicate that mixed pixels have a significant influence on the information that can be derived.

Classification of mixed pixels leads to errors that make the subsequent area estimation inaccurate. These errors are caused by the premise of classification that all pixels are *pure*, i.e. consisting of a single ground cover type, while in fact they are not. The resulting problems are twofold:

1. The mixed pixel is assigned to one class only, while in fact it belongs to two or more classes. As a result, the direct area estimates acquired by pixel counting will be inaccurate. Suppose for instance that a pixel is composed of class A for 60% and class B for 40%. The best decision a classifier can make is to categorise the pixel as class A. However, this decision will lead to an overestimation of the area of A of 0.4 pixel and an equally large underestimation of the area of class B.

2. The mixed pixel may no longer match any of the spectral signatures of the categories present. Even worse, the mixed reflectances of class A and B may resemble the spectral signature of a totally different category C that is not even present within the pixel—see Figure 1.3. In this case, which is a case of *spectral confusion*, the area of both A and B is underestimated, while the area of C is overestimated.

Although it may seem that the above errors diminish the area estimation accuracy only marginally, one should remember that the number of mixed pixels can be very large. Furthermore, it must be noted that both sources of error are systematic instead of random. Suppose for instance that the variation within class A is larger than the variation within class B. Since most classifiers take the within-class variation into account, pixels composed of 75% B and 25% A may well be assigned to class A instead of class B. As a result, the area of A is overestimated much more often than the area of B, which leads to severely

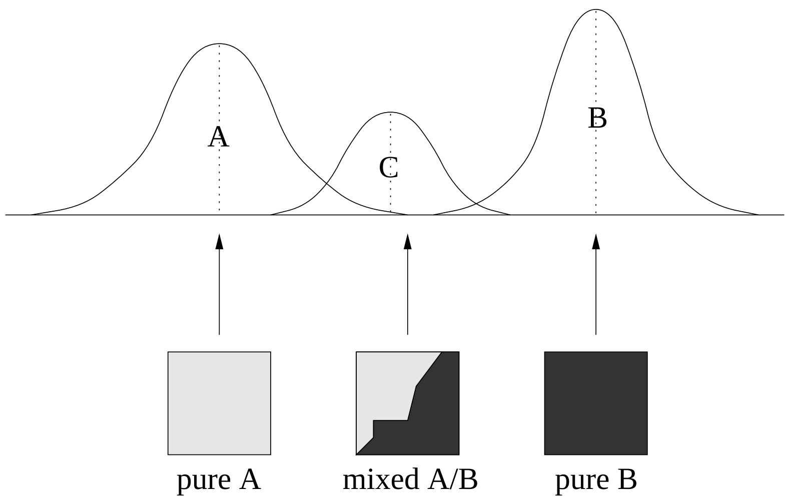


Figure 1.3: *Spectral confusion caused by mixing of ground cover types. The two pure pixels of category A and B are classified correctly as they are best represented by their class distribution. The mixed pixel is misclassified because its reflectance better resembles the spectral signature of the third category C than that of A or B (after Campbell [19]).*

inaccurate area estimates in case two agricultural fields of crops A and B are adjacent. Spectral confusion only adds to this problem as many mixed pixels composed of A and B will be allocated to class C, but the area of a pure pixel belonging to C will never be divided between classes A and B. Therefore, classification of mixed pixels is inappropriate for area estimation because the many small errors that are made do not cancel each other out but, when aggregated, lead to serious overestimation of the area of certain crop types, while the area of others will be underestimated.

The mixed pixel problem is not solved simply by increasing the spatial resolution. In general, the proportion of mixed pixels decreases as the spatial resolution becomes finer, for the smaller pixel size allows more pure pixels to be fit within the object boundaries. In some cases, however, the proportion of mixed pixels can actually increase because the finer detail resolves features not recorded before, thus introducing new spectral classes (Campbell [19]). For example, the image of a forested area, which seemed uniform at coarse resolution, may display individual trees of different species interspersed with open spaces at finer resolution (Woodcock and Strahler [114]). But even if the spectral classes remain the same and the proportion of mixed pixels decreases, the classification results can still deteriorate (Markham and Townshend [67], Irons *et al.* [52]). The main reason for this effect is that at finer resolutions the within-class variation increases as local differences in humidity, elevation, illumination, etc. become more apparent. Another reason is that the increase in spatial resolution usually is achieved at the expense of the spectral or radiometric resolution, because the reduction in received energy due to a smaller IFOV must be compensated for, e.g. by broadening the spectral band at which the reflectance is measured. A further disadvantage of fine spatial resolution is that the number of pixels can become very large, which adds to the costs of processing. Together with the fact that there is a growing interest in data of global coverage recorded on more and smaller spectral bands, new scanners may be designed to have coarser rather than finer spatial resolution.

1.4 Overview of this thesis

In this section an overview of this thesis is given. Section 1.4.1 briefly describes the history of remote sensing and how this thesis relates to previous developments. Next, Section 1.4.2 explains what goals are aimed at and Section 1.4.3 indicates how these goals have been accomplished by summarising the structure of the remaining chapters.

1.4.1 Historical context

In the early days of remote sensing the earth's surface was recorded using photography, at first (1858) from a captive balloon but later (1910) from an airplane. This data was analysed initially by visual examination of prints and transparencies, but with the introduction of the computer just after the Second World War digital image analysis became possible as well (at least in theory). It was not until the launch of the first Landsat satellite in 1972, however, that digital remote sensing data of large regions of the earth's surface became routinely available to a wide range of research institutes, which greatly increased the interest in and development of digital image analysis techniques.

Over the last decades, numerous algorithms to extract ground cover information from remotely sensed images have been designed. Being a well-known statistical technique in other scientific disciplines, (pixel-based) classification was the first method to be applied to the multispectral digital image data. In 1975, Kettig and Landgrebe [62] introduced a field-based classification approach called ECHO as a means to reduce the number of misclassifications due to sensor noise and within-field variation. Since then, many similar methods using a pixel's spatial context have been proposed, but pixel-based classification is still an important tool nowadays. Because the mixed pixel problem cannot be solved using classification, Horwitz *et al.* [51] developed an alternative, *decomposition* technique, which attempts to estimate the abundances of the ground cover types a mixed pixel is composed of. Although this method was proposed already in 1971, it was not followed up to any great degree until the mid 1980s, when Smith *et al.* [103] and Adams *et al.* [2] used it for their geological applications. From then on, interest in spectral unmixing has greatly increased, which resulted in a wide variety of decomposition techniques and application areas. New developments in sensor technology like coarser spatial resolution to achieve global coverage (AVHRR) and finer spectral resolution to record the material characteristics more accurately (AVIRIS) increase the necessity for and possibilities of decomposition, respectively, which only amounts to a further growth of its popularity. For a more detailed overview of related work we refer to the introduction of later chapters in general and the whole of Chapter 2 in particular.

This thesis deals with the use of decomposition for area estimation of agricultural fields. In many aspects it is a follow-up to the thesis of Schoenmakers [90], who, in the frame of the EU programme Monitoring Agriculture with Remote Sensing (MARS), developed a hybrid segmentation method based on a pure pixel model as part of a system to distinguish, identify, and measure the area of crops of significant importance. With the institution of the McSharry agricultural subsidy system in 1993, however, it became more important to get an accurate estimation of the area of an agricultural field instead of the crop area over a larger region, which revealed the necessity of using a mixed pixel model. Quarmby *et al.* [82] already demonstrated that decomposition can be applied for local crop area estimation with good results, but only on a per pixel basis. In this thesis decomposition is combined with aspects borrowed from field-based classification regarding the use of spatial information to improve the area estimates. In addition, a new mixture model is introduced that is better suited for the decomposition of mixed pixels in agricultural scenes. On top of this, several smaller issues related to decomposition are addressed, while keeping the requirements of hyperspectral remote sensing in mind.

1.4.2 Objectives

The aims of this thesis are twofold. On the one hand, we attempt to solve a specific, technical problem related to the area estimation of agricultural fields. On the other hand, we pursue a more general, scientific goal regarding the processing of mixed pixels.

Specific goal

The main objective of this thesis can best be explained by looking at the problems that were identified previously. As was described in the first section, it is of importance to be able to estimate the area of an agricultural field with relatively high accuracy. The usual approach of classifying all pixels and counting the number of pixels per crop is rather inaccurate because of the errors that are introduced by the classification of mixed pixels. Alternatively, decomposition techniques can be applied, after which a mixed pixel can be assigned to several categories proportionally to the fraction of its area covered by each class. The overall improvement of the field area estimates that can be achieved with this decomposition approach is difficult to predict since this varies with the number of mixed pixels present, which in turn depends on fabric of the landscape (viz. the average field size). An estimate of the improved accuracy per mixed pixel, however, can be calculated as follows. Suppose classification always assigns a mixed pixel to the class with the highest proportion. Since most mixed pixels lie on the boundary of two agricultural field and hence consist of only two classes, the proportion of the pixel's area that is assigned incorrectly is uniformly distributed between 0% and 50%, giving an average error of 25% per mixed pixel. Therefore, the main objective of this dissertation can be stated as follows:

> *The primary aim of this thesis is to develop a method for the area estimation of agricultural fields that makes use of mixed pixel decomposition to improve the accuracy of a similar method based on classification with 25% per mixed pixel.*

Since another advantage of decomposition is that due to its mixed pixel model it is less susceptible to spectral confusion than classification, an even better performance may be achieved. The magnitude of this additional improvement in accuracy, however, cannot be predicted as it depends on the spectral signatures of the classes and the precise topography of the agricultural fields in the scene.

General goal

Although this thesis is primarily focused on estimating the area of agricultural fields, most of the techniques and procedures developed can also be applied for other purposes. A very similar application, for instance, is the estimation of the forested area in developing countries to monitor the rate of deforestation (Cross *et al.* [23]). Some examples of applications that are related more distantly are the estimation of the amount of sparse (woodland) vegetation in semi-arid regions (Pech *et al.* [79], Williamson [112]) or deserts (Smith *et al.* [104]), the estimation of water-sediment concentrations (Mertes *et al.* [73], Novo and Shimabukuro [77]), and the estimation of mineral abundances (Drake and Settle [24], Thomson and Salisbury [110], Bryant [18]). However, since the objects of interest in these cases are much smaller than a single pixel, only the general decomposition techniques can be used and not the methods that utilise spatial information based on knowledge of the agricultural domain. Apart from images acquired through satellite remote sensing, the techniques suggested in this thesis may also be applied to industrial images (e.g. to check the production of IC bodies [37]) or medical imagery (e.g. to estimate the amount of tumour tissue), provided that certain conditions regarding the class distributions and the mixing process are fulfilled. Therefore, this dissertation also serves a second purpose:

> *Furthermore, this thesis aims at providing a deeper knowledge about the processing of mixed pixels in general by developing and comparing mixture models as well as methods for mixed pixel detection, decomposition, and related tasks, relatively independent of the specific application and type of imagery used.*

1.4.3 Outline

This thesis consists of two parts, preceded and concluded by a separate chapter. This first chapter serves as an introduction and describes the problem to be solved. The last chapter (Chapter 8) summarises the conclusions and gives some directions for future work.

In Part I, several existing as well as new methods for the decomposition of mixed pixels are discussed. In Chapter 2, the well-known linear mixture model and some of its solutions are described, as well as some other decomposition methods found in the literature. Section 2.2.10 briefly discusses the use of neural networks, which is a relatively new approach; research on this subject carried out in cooperation with Theo Schouten has been published previously in [93] and [94]. Next, Chapter 3 introduces a new variant on the (statistical) linear mixture model that has a better physical foundation, especially when agricultural fields are concerned. The basic idea behind this model was first presented at the 2nd European Symposium on Remote Sensing [34], while a joint paper with Theo Schouten and Jeff Settle covering the entire chapter has been submitted to the *International Journal of Remote Sensing* [39]. Chapter 4, finally, ties up some loose ends concerning decomposition methods based on the linear mixture model. The first section of this chapter discusses some ways to restrict the set of solutions to the ones that are physically sound, while the other section deals with some existing as well as new techniques to derive *endmembers*—a concept of the linear mixture model that is closely related to notion of class (spectral) signatures—directly from the remotely sensed image. The novel approaches to both topics, which were the result of a joint project with Theo Schouten, Zhengkai Liu, and S.W. Chen, have been published previously in [95]. Nearly all of the methods presented in this part can be used for any type of area estimation application needing high accuracy.

The second part aims specifically at the area estimation of agricultural fields. Chapter 5, which in part is based on the Master's project of Ron Grazier [44], discusses two techniques

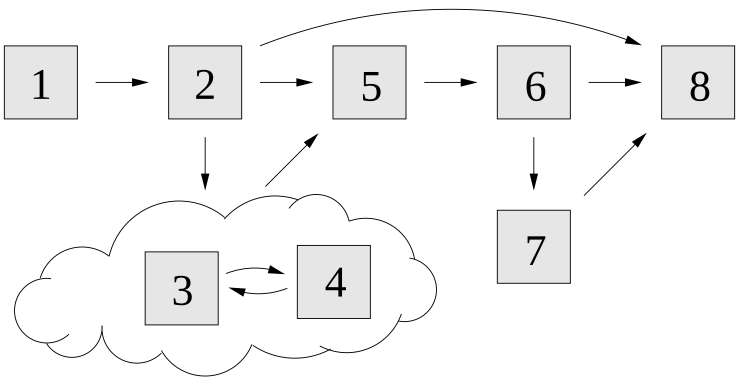


Figure 1.4: *Thesis structure, leading to a recommended order to read the chapters.*

that use spatial information to distinguish mixed pixels from pure pixels. Chapter 6 is the heart of this thesis because it describes a method to make an accurate estimation of the area of an agricultural field. The basis of this method is an agricultural scene model that can be used to determine endmember distributions and probable components of a mixed pixel from the image itself, thus facilitating an accurate decomposition. This work has previously been reported in [35]. In Chapter 7, finally, this method and three other methods are applied to artificial images in order to analyse and compare their performance quantitatively. Furthermore, an experiment using a real remotely sensed image is described by which the conclusions based on the experiment with artificial images can be verified. Both experiments have been published as CSI-reports [35, 36], while a joint paper with Theo Schouten combining this chapter and Chapter 6 has been accepted for publication (subject to revision) in the *International Journal of Remote Sensing* [38]. Although the methods presented in this part are designed specifically to estimate the area of agricultural fields, they may also be applicable when estimating the area of other large objects. An example of such an (industrial) application, which is not included in this thesis, has been reported in [37].

The way this thesis is structured leads to a natural reading order of the chapters, which is shown in Figure 1.4. If, after going through the introduction, the reader only wants to get an idea about the way a well-known decomposition method works, he or she should read (the first part of) Chapter 2 before turning to the concluding chapter. The more interested reader, however, may also want to study any of the Chapters 3–4. If looking for a method to estimate the area of large objects like agricultural fields, one should also read Chapters 5–7; reading this last chapter, however, is optional as it describes two case studies that prove the effectiveness of the area estimation method proposed.

Part I

Decomposition of mixed pixels

Chapter 2

Methods to decompose mixed pixels

The *decomposition* of a mixed pixel entails the unmixing of that pixel to find the properties of its original components; usually, however, only the proportion of the area covered by each component is derived as the single property of interest. For this purpose, several methods have been developed over the last few decades. The best-known technique is linear mixture modelling, which is explained in Section 2.1. A selection of some other methods that were proposed is described in Section 2.2.

2.1 Linear mixture modelling

Linear mixture modelling is the most widely used method for extracting compositional information from remotely sensed images containing many mixed pixels. For instance, it was used to eliminate the effects of topography and illumination in Viking Lander 1 images (Adams *et al.* [2]); to find porphyry copper deposits (Drake and Settle [24]); to measure tropical forest cover (Cross *et al.* [23]); and to estimate the area of agricultural crops (Quarmby *et al.* [82]). Furthermore, it has been applied in numerous other studies estimating the coverage by rangeland vegetation, snow, and urban ground cover types (e.g. [79, 104, 86, 41]). The linear mixture model on which this technique is based is described in Section 2.1.1; Section 2.1.2 explains how this model is used to estimate a mixed pixel's proportions.

2.1.1 The statistical linear mixture model

The *linear mixture model* relates the spectral signature of a combination of a number of classes within the IFOV to the signatures of the individual classes. It was introduced by Horwitz *et al.* [51] in 1971 as a means to increase the crop area estimation accuracy achieved by standard processing techniques. They assume that a *resolution cell*, i.e. the ground area defined by the IFOV, contains many small objects (elements) belonging to c different classes. The spectral signature of each class i is taken to be an n-dimensional Gaussian distribution with mean $\mathbf{m}_i$ and variance-covariance matrix $\mathbf{N}_i$, where n is equal to the number of spectral bands the scanner takes measurements at. Now suppose that a resolution cell contains elements of class i only, which are represented by random variables with mean $\mathbf{m}_i^*$ and variance-covariance matrix $\mathbf{N}_i^*$. If the number of such elements present

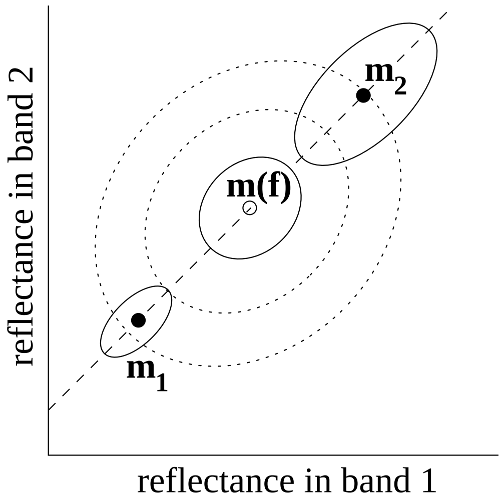


Figure 2.1: *Spectral signatures according to the linear mixture model. The two-dimensional feature space shows the composite mean* $\mathbf{m}(\mathbf{f})$ *in the middle—*$\mathbf{f} = (0.5, 0.5)^T$*—of the two class means* $\mathbf{m}_1$ *and* $\mathbf{m}_2$*. The solid ellipses represent the various variance-covariance matrices by showing the points at a certain (Mahalanobis) distance; the dotted ellipses show the points at four and nine times this distance to* $\mathbf{m}(\mathbf{f})$*.*

within the resolution cell is equal to a_i, then

$$\mathbf{m}_i = a_i \mathbf{m}_i^*, \quad \text{and} \tag{2.1}$$

$$\mathbf{N}_i = a_i \mathbf{N}_i^*, \tag{2.2}$$

provided that the variables are statistically independent. Next, assume that the proportions of the object classes within a resolution cell are defined by $\mathbf{f} = (f_1, \ldots, f_c)^T$ and that consequently the number of elements of class i is $f_i a_i$. As a result, the composite signature of the objects will be given by:

$$\mathbf{m}(\mathbf{f}) = \sum_{i=1}^{c} f_i a_i \mathbf{m}_i^* = \sum_{i=1}^{c} f_i \mathbf{m}_i \tag{2.3}$$

$$\mathbf{N}(\mathbf{f}) = \sum_{i=1}^{c} f_i a_i \mathbf{N}_i^* = \sum_{i=1}^{c} f_i \mathbf{N}_i. \tag{2.4}$$

The latter equation only holds if the random variables associated with elements from different classes are statistically independent as well. Assuming that the conditions for the central limit theorem are satisfied, we can consider Equations 2.3 and 2.4 as the mean vector and variance-covariance matrix, respectively, of a distribution that is multivariate normal—see Figure 2.1. Thus, the composite signature of a combination of classes can be described in terms of the signatures of the individual classes.

The validity of the linear mixture model depends on the type of application. As described in the previous paragraph, the model is based on the assumption that the random variables associated with the elements are statistically independent. With regards to ground cover applications, Settle and Drake [99] translate this prerequisite to the condition that the amount of multiple scattering between the different ground cover types must be negligible, i.e. (nearly) all photons reaching the scanner's sensor have interacted with just one cover type. According to Campbell [19, p. 405], this situation arises when the resolution cell contains two or more surfaces occurring in patches that are large relative to the sensor's resolution. On the other hand, if the component surfaces occur in highly dispersed patterns, mixing is likely to be nonlinear since the probability that radiation is scattered by one cover type and subsequently reflected by another cover type before arriving at the sensor is much higher. When dealing with mixed pixels that occur at the boundaries of agricultural fields, the resolution cell is usually covered by only two or three component surfaces. Therefore, as far as the application described in this thesis is concerned, the linear mixture model appears to be a good abstraction of the mixing process.

2.1.2 Decomposition based on the statistical model

Mixed pixels can be decomposed by inverting the statistical model described previously. To this end, the linear mixture model is usually rewritten in matrix-vector notation:

$$\mathbf{x} = \mathbf{Mf} + \mathbf{e}. \tag{2.5}$$

If the number of spectral bands and ground cover types are given again by n and c respectively, then $\mathbf{x}$ represents an $nx1$ pixel vector or multispectral observation, while, as before, $\mathbf{f}$ denotes the $cx1$ fractions vector with the proportions of the different ground cover types. Each column of (nxc) matrix $\mathbf{M}$ contains the spectrum of a so-called *endmember*, which is the reflectance typical for a resolution cell containing nothing but the cover type of interest. According to Section 2.1.1, the i-th endmember spectrum is equal to the mean vector $\mathbf{m}_i$ of class i, but there exist some other views as well—see Section 4.2 for a more elaborate description of image endmembers and their extraction. The ($nx1$) error vector $\mathbf{e}$, finally, is used to model the statistical fluctuations around the mean value $\mathbf{m}(\mathbf{f})$=$\mathbf{Mf}$ and is assumed to have a multivariate normal distribution with a zero mean and variance-covariance matrix $\mathbf{N}(\mathbf{f})$. The *mixing equations* defined by Equation (2.5) are usually accompanied by two constraints that are obvious in the model of Section 2.1.1, but which should be satisfied explicitly when estimating $\mathbf{f}$. The *sum-to-one constraint* is that a pixel is well-defined by its components, whose proportions should therefore add up to unity:

$$\sum_{i=1}^{c} f_i = 1. \tag{2.6}$$

The other constraint that should be satisfied is the *positivity constraint*, which says that no component of a mixed pixel can make a negative contribution:

$$f_i \geq 0 \quad \text{for} \quad i = 1, \ldots, c. \tag{2.7}$$

Satisfaction of the latter constraint is often difficult and may require some specialised techniques (see Section 4.1). Together, the mixing equations and the constraints describe a model that must be solved[1] for each mixed pixel that is to be decomposed, i.e. given $\mathbf{x}$ and $\mathbf{M}$, one has to determine $\mathbf{f}$ and $\mathbf{e}$ subject to (2.5)–(2.7).

[1]In this thesis, solving a model means solving the system of equations described by the model.

An important characteristic of a decomposition method is the criterion that determines which solution of the linear mixture model is optimal. Suppose that all endmember spectra are linearly independent of each other and consider the system of equations defined by Equations (2.5) and (2.6) (note that (2.7) is an inequality and not an equation). If the error vector $\mathbf{e}$ is disregarded, then this system of $n+1$ linear equations in c unknowns—the f_i—has infinitely many solutions if $c>n+1$, exactly one solution if $c=n+1$, and at most one solution if $c<n+1$. To provide a general solution in the last case, error vectors unequal to zero have to be allowed, which results in an infinity of solutions again ($c+n>n+1$). In contrast to the previous underdetermined system ($c>n+1$), however, it is possible to identify a solution that is optimal in some sense based on the value of the error vector $\mathbf{e}$. The solution that is selected following the popular maximum likelihood approach is the combination of fractions and error vector that has the highest probability. Since $\mathbf{e}$ is assumed to have a multivariate normal distribution with a zero mean and variance-covariance matrix $\mathbf{N}(\mathbf{f})$, the most probable solution is the one that minimises

$$\mathbf{e}^T(\mathbf{N}(\mathbf{f}))^{-1}\mathbf{e} + \ln|\mathbf{N}(\mathbf{f})| = (\mathbf{x}-\mathbf{M}\mathbf{f})^T(\mathbf{N}(\mathbf{f}))^{-1}(\mathbf{x}-\mathbf{M}\mathbf{f}) + \ln|\mathbf{N}(\mathbf{f})| \tag{2.8}$$

(c.f. Equation (A.4), the criterion used for maximum likelihood classification). In this equation, $\ln|\mathbf{N}(\mathbf{f})|$ denotes the natural logarithm of the determinant of $\mathbf{N}(\mathbf{f})$, while the other term represents the *Mahalanobis distance* between pixel $\mathbf{x}$ and point $\mathbf{M}\mathbf{f}$. Thus, among the infinitely many solutions for the linear mixture model, the maximum likelihood approach is able to identify the single, most probable alternative.

Although the maximum likelihood criterion determines which of the solutions is more probable, it does not say how these solutions can be found. Basically, there are two ways to solve the equations of the linear mixture model: brute force and mathematical analysis.

Brute force approximation

The brute force approach is conceptually the least complex as it simply searches the entire solution space. For every $\mathbf{f}$ that satisfies the sum-to-one and positivity constraint, the corresponding error vector $\mathbf{e}$ and variance-covariance matrix $\mathbf{N}(\mathbf{f})$ are calculated, and Equation (2.8) is evaluated; the $\mathbf{f}$ that minimises this cost function is taken as the fractions vector of $\mathbf{x}$. Obviously, this approach leads to an approximation of the optimal $\mathbf{f}$ because the solution space is searched taking discrete steps. Furthermore, it has a considerable computational complexity: if the sampling rate, which equals the reciprocal of the step size, is denoted by r, the maximum likelihood criterion has to be evaluated in the order of r^{c-1} times. By varying the sampling rate, the accuracy of the approximation can be traded off against computational costs.

This method can well be illustrated using *convex geometry* principles for an n-dimensional feature space. According to these principles, all linear combinations of c endmembers whose proportions sum to one are contained in the $(c-1)$-dimensional subspace spanned by those endmembers; if the proportions are also non-negative, the space is further restricted to the simplex defined by the endmembers. The brute force approach is to apply systematic sampling of this simplex in order to determine the point $\mathbf{p}$ that is closest to pixel $\mathbf{x}$. The proportions vector corresponding to $\mathbf{p}$ is subsequently adopted as the proportions vector of $\mathbf{x}$. Figure 2.2 shows a simple example with two endmembers in a two-dimensional space.

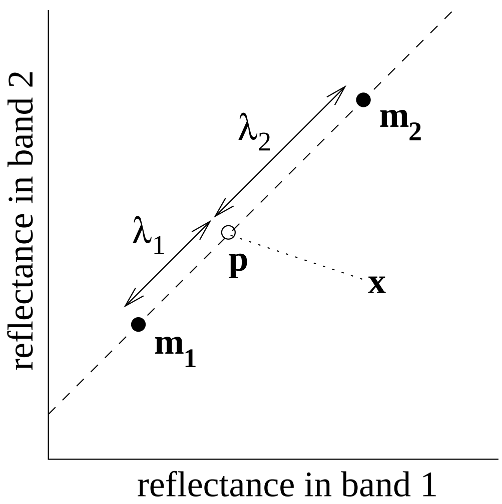


Figure 2.2: *Maximum likelihood solution of the statistical linear mixture model. In this two-dimensional feature space, linear combinations of endmembers* $\mathbf{m}_1$ *and* $\mathbf{m}_2$ *satisfying the sum-to-one constraint occupy the dashed line; those that satisfy the positivity constraint as well lie on* $\mathbf{m}_1\mathbf{m}_2$*. To find the optimal solution, point* $\mathbf{p}$ *is moved along* $\mathbf{m}_1\mathbf{m}_2$ *until the Mahalanobis distance from pixel* $\mathbf{x}$ *to* $\mathbf{p}$ *(plus a log-term) is minimal. The resulting fractions vector* $\mathbf{f}$ *equals* $(\lambda_2/(\lambda_1+\lambda_2), \lambda_1/(\lambda_1+\lambda_2))^T$.

Analytic approximation

Finding the minimum of the maximum likelihood cost function using mathematical analysis is difficult because the variance-covariance matrix $\mathbf{N}(\mathbf{f})$ depends on the fractions vector. However, if all matrices $\mathbf{N}_i$ are equal to a common matrix $\mathbf{N}$, then a closed expression can be derived. Fortunately, as Horwitz *et al.* convincingly argued [51], $\mathbf{N}(\mathbf{f})$ can be approximated by the mean variance-covariance matrix

$$\mathbf{N} = \frac{1}{c}\sum_{i=1}^{c}\mathbf{N}_i \tag{2.9}$$

without much loss of accuracy, as long as the dispersion of the endmembers is relatively large compared to the within-class variation of the individual ground cover types. Now the minimum of Equation (2.8) can be calculated by setting the partial derivatives with respect to each f_i to zero, which gives the unconstrained estimator

$$\hat{\mathbf{f}}_0 = \mathbf{U}\mathbf{M}^T\mathbf{N}^{-1}\mathbf{x} \quad \text{where} \quad \mathbf{U} = (\mathbf{M}^T\mathbf{N}^{-1}\mathbf{M})^{-1}. \tag{2.10}$$

If the sum-to-one constraint is imposed, then a standard Lagrangian analysis gives a slightly different solution (Settle and Drake [99]):

$$\hat{\mathbf{f}} = \hat{\mathbf{f}}_0 + \alpha\left(1 - \mathbf{1}^T\hat{\mathbf{f}}_0\right)\mathbf{U}\mathbf{1}, \quad \text{where} \quad \alpha = \left(\mathbf{1}^T\mathbf{U}\mathbf{1}\right)^{-1} \tag{2.11}$$

and $\mathbf{1}$ denotes the $(c\mathrm{x}1)$ vector consisting entirely of ones. Satisfaction of the positivity constraint cannot be achieved with a similar transformation, but requires some post-processing of the initial estimate or an alternative approach to solve the model's equations—see Section 4.1. The simplest post-processing approach is to set any negative proportions to zero and renormalise the fractions vector. Although this method may cause some extra loss of accuracy, the computational costs are so low that it is often applied. Together, the simple matrix-vector operations of Equation (2.11) and the fast renormalisation method provide a near-optimal solution of the linear mixture model that can be calculated efficiently.

2.2 Other decomposition methods

Apart from the linear mixture modelling approach explained in the previous section, numerous other techniques have been described in the literature. Some of these methods are based on complex non-linear reflectance models derived from the structure of specific plants (e.g. Borel and Gerstl [14], Ray and Murray [83], Jasinski [55]); they will not be considered in this thesis. Another approach that will not be discussed is the use of vegetation indices, which are used to measure biomass; as their ability to discriminate between different crop types is limited and their application in general is restricted to pure instead of mixed pixels (but see Kerdiles and Grondona [60]), vegetation indices are considered unsuitable for mixed pixel decomposition. What remains is a collection of in some cases closely related techniques, which are mainly based on classic statistical principles; a selection of these methods is briefly described in Sections 2.2.1 to 2.2.10.

2.2.1 Simple weighted averaging

The *weighted averaging* method introduced by Marsh *et al.* [68] is a simplified version of the linear mixture modelling approach. It models a mixed pixel as a linear mixture of only two components, thus resulting in the following well-known mixing equations:

$$x_i = f_1 m_{i1} + (1 - f_1) m_{i2} \quad \text{for} \quad i = 1, \dots, n. \tag{2.12}$$

Since Equation (2.12) contains only one unknown (f_1), the exact solution can easily be calculated:

$$\hat{f}_1 = \frac{x_i - m_{i2}}{m_{i1} - m_{i2}}. \tag{2.13}$$

Satisfaction of the positivity constraint is achieved by forcing $f_1 = 0$ if the estimate is negative, and $f_1 = 1$ if the estimate is greater than one. The method computes a separate proportion estimate in each spectral band, which may well be different from the estimates in the other $n-1$ bands. In general, the band with the largest dispersion of the endmembers relative to the within-class variation will yield the best estimate. However, a reliable decision regarding the band that gives the most accurate results can only be made when the correct proportions of a number of calibration pixels are known.

2.2.2 Simple linear regression

A classic statistical technique to find the relation between two (or more) variables is *linear regression*. Marsh *et al.* [68] applied it to unmix pixels consisting of two components, using

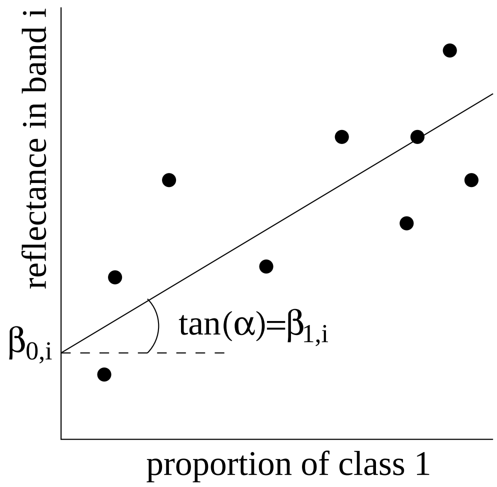


Figure 2.3: *The simple linear regression approach fits a line to a set of calibration samples such that these pixels are approximated best. Based on the slope $\beta_{1,i}$ and the intercept $\beta_{0,i}$ of the line, the unknown proportions of other pixels can be estimated (see Equation (2.15)).*

the following model:

$$x_i = \beta_{0,i} + \beta_{1,i} f_1 \quad \text{for} \quad i = 1, \dots, n. \tag{2.14}$$

Richardson *et al.* [84] used a similar model with an extra term $\beta_{2,i} f_2$ to describe mixtures of three components. The coefficients $\beta_{0,i}$ and $\beta_{1,i}$ are found by calculating the intercept and the slope, respectively, of the line that gives the best representation of a number of calibration pixels—see Figure 2.3. Once $\beta_{0,i}$ and $\beta_{1,i}$ have been determined, the proportion of class 1 can easily be calculated according to

$$\hat{f}_1 = \frac{x_i - \beta_{0,i}}{\beta_{1,i}}. \tag{2.15}$$

This equation coincides with Equation (2.13)—the simple weighted averaging estimator—if $\beta_{0,i} = m_{i2}$ and $\beta_{1,i} = m_{i1} - m_{i2}$, which indicates that the linear regression approach in fact is a method to determine the endmember spectra; therefore, this technique will be discussed in more depth in Section 4.2.3.

2.2.3 Linear discriminant analysis

Discriminant analysis is another classic statistical technique, which traditionally is used for classification purposes. When dealing with two classes, this method transforms a multivariate set of measurements—the pixel—into a single discriminant value, which is related to the likelihood of that pixel belonging to a particular class. While standard linear discriminant analysis uses only the sign of the discriminant value to determine a pixel's class, Marsh *et al.* [68] proposed to use the magnitude of the value as well in order to determine the proportions of the classes a mixed pixel is composed of. The estimator they suggest is

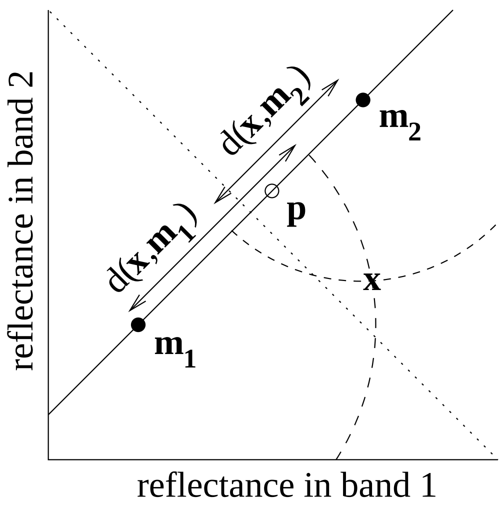


Figure 2.4: *Proportions estimated using linear discriminant analysis. The solid line is the axis of the function* $0.5(d(\mathbf{x},\mathbf{m}_2) - d(\mathbf{x},\mathbf{m}_1))$*; left of the dotted line this function is positive, and pixels will be assigned to class 1 during classification. The dashed arcs show a projection of the M-distances* $(\mathbf{N} = \mathbf{I})$ *between* $\mathbf{x}$ *and the class means, with point* $\mathbf{p}$ *lying in the middle. According to Figure 2.2, the fraction* f_1 *of* $\mathbf{p}$ *can be calculated as* $\{d(\mathbf{x},\mathbf{m}_2) - 0.5(d(\mathbf{x},\mathbf{m}_2) + d(\mathbf{x},\mathbf{m}_1) - d(\mathbf{m}_2,\mathbf{m}_1))\}/d(\mathbf{m}_2,\mathbf{m}_1)$*, which reduces to Equation (2.16).*

a transformed version of the discriminant function $d(\mathbf{x},\mathbf{m}_2) - d(\mathbf{x},\mathbf{m}_1)$:

$$\hat{f}_1 = 0.5 + 0.5\frac{d(\mathbf{x},\mathbf{m}_2) - d(\mathbf{x},\mathbf{m}_1)}{d(\mathbf{m}_2,\mathbf{m}_1)}, \tag{2.16}$$

where d represents the Mahalanobis distance function $d(\mathbf{p},\mathbf{q}) = (\mathbf{p}-\mathbf{q})^T\mathbf{N}^{-1}(\mathbf{p}-\mathbf{q})$ and $\mathbf{N}$ is the pooled variance-covariance matrix accounting for statistical noise. Projection of the discriminant value divided by 2 onto a line joining the class means $\mathbf{m}_1$ and $\mathbf{m}_2$ shows that this method in fact determines a point $\mathbf{p}$, which lies in the middle of the projected Mahalanobis distances to $\mathbf{m}_1$ and $\mathbf{m}_2$, and estimates the proportions of $\mathbf{x}$ by calculating the proportions of $\mathbf{p}$—see Figure 2.4. A generalisation of this method to the three-class case was considered by Foschi [31].

2.2.4 Fuzzy classification

The *fuzzy classification* method proposed by Wang [111] and slightly modified by Maselli *et al.* [69] is an extension of the well-known maximum likelihood classification technique (see Appendix A). Analogous to the linear discriminant analysis estimator presented in the previous section, this method does not merely use the relative order of the class-specific probabilities, it also utilises their actual values to determine the proportions of the classes that make up the mixed pixel. In a fuzzy representation, each pixel is described by a group of *membership grades* which indicate the degree of similarity or strength of membership to the classes considered. Just like the class proportions, the fuzzy membership function values must be positive and sum to one for each pixel. The basic assumption of the fuzzy approach is that membership grades are informative about the subpixel components.

Maselli *et al.* take a membership grade function that is equal to the posterior probability $p(k|\mathbf{x})$ of a pixel $\mathbf{x}$ belonging to a class k and show that in their study the membership grades relate directly to the class proportions with surprising accuracy. According to the Bayesian theorem, therefore,

$$\hat{f}_k = p(k|\mathbf{x}) = \frac{p(k)p(\mathbf{x}|k)}{\sum_{i=1}^{c} p(i)p(\mathbf{x}|i)}, \tag{2.17}$$

where the class-specific probabilities $p(\mathbf{x}|i)$, which are also calculated during maximum likelihood classification, are given by

$$p(\mathbf{x}|i) = 2\pi^{-n/2} \left|\mathbf{N}_i\right|^{-1/2} \exp(-\tfrac{1}{2}(\mathbf{x} - \mathbf{m}_i)^T \mathbf{N}_i^{-1}(\mathbf{x} - \mathbf{m}_i)). \tag{2.18}$$

The prior probabilities $p(i)$ represent the likelihood that a pixel belongs to class i without taking the pixel vector itself into consideration. A simple way to determine them is to calculate the proportion of pixels of class i in a training set representative for the entire scene. Maselli *et al.*, however, compute different prior probabilities for each pixel using frequency histograms derived from training sets with good results.

2.2.5 Fuzzy *c*-means

Similar to fuzzy classification, the *fuzzy c-means* approach (Foody and Cox [29], Foody [25], Atkinson *et al.* [7]) is also based on fuzzy set theory. The algorithm partitions the data in the feature space into c fuzzy groups or classes, where a fuzzy c-means partition U of p pixels is defined by:

$$U = \{u_{ik}|u_{ik} \in [0,1];\ \textstyle\sum_{k=1}^{p} u_{ik} > 0,\ i = 1, \ldots, c;\ \sum_{i=1}^{c} u_{ik} = 1,\ k = 1, \ldots, p\}. \tag{2.19}$$

The membership grades u_{ik} associated with each pixel k are not directly related to the class-specific probabilities as in Section 2.2.4, but are chosen such that

$$\sum_{k=1}^{p}\sum_{i=1}^{c}(u_{ik})^q d(\mathbf{x}_k, \mathbf{m}_i) \tag{2.20}$$

is minimal. In this equation, $d(\mathbf{x}_k, \mathbf{m}_i)$ represents the Mahalanobis distance between the k-th pixel and endmember i, and q is a weighing parameter that indicates the degree of fuzziness; when q=1, a conventional classification is obtained, while a greater value, for instance in the range 1.5–3.0 which has been used in most studies, yields a truly fuzzy classification. The actual algorithm to minimise Equation (2.20) was described by Bezdek *et al.* [12] and will not be repeated here for reasons of brevity. Furthermore, the equations to calculate class fractions from membership grades, which due to the more complex definition of the membership function no longer share a 1:1 relationship, will not be given either. Foody and Cox [29] use linear regression to derive this relationship for each class separately. Since this well-known statistical technique is also described in other parts of this thesis, the reader is referred to Sections 2.2.2 and 4.2.3 for an explanation.

2.2.6 Relaxation

Arai and Terayama [5] proposed a decomposition method that takes compositional information about neighbouring pixels into account. Their approach was motivated by the observation that segments of linear features such as roads often consist of mixed pixels with a similar composition (e.g. 50% road, 20% left-hand side field, 30% right-hand side field). By *relaxing* an initial estimation of the proportions using the proportions of a neighbouring pixel, a better estimate is obtained. Assuming the linear mixture model with an identity variance-covariance matrix, the following expression is to be minimised:

$$(\mathbf{x} - \mathbf{Mf})^T(\mathbf{x} - \mathbf{Mf}) + r(\mathbf{f} - \mathbf{g})^T(\mathbf{f} - \mathbf{g}). \tag{2.21}$$

In the above equation, the $cx1$ vector $\mathbf{g}$ represents the fractions of a connected pixel, and r is a constraint factor that determines the strength of the relaxation effect. Setting to zero the derivative of Equation (2.21) with respect to $\mathbf{f}$ gives the unconstrained estimator

$$\hat{\mathbf{f}} = (\mathbf{M}^T\mathbf{M} + r\mathbf{U}^T\mathbf{U})^{-1}\mathbf{M}^T\mathbf{x}, \quad \text{where} \quad \mathbf{U} = \mathbf{I} - \mathbf{g}\mathbf{1}^T; \tag{2.22}$$

$\mathbf{I}$ and $\mathbf{1}$ denote the (cxc) identity matrix and a ($cx1$) vector consisting entirely of ones, respectively. The optimal value of the constraint factor r depends on the true composition of the pixel—since this is to be estimated, the proportions of the neighbouring pixel can be taken as a candidate mixture—and the Euclidean distance to the neighbouring pixel in feature space. Obviously, r should be decreased (i.e. the relaxation effect should be lessened) as the distance between the pixels increases, because greater differences in spectral reflectance probably mean that the underlying compositions are less similar as well. To provide a fast mechanism for determining r, Arai and Terayama use a look-up table showing the optimum r for all combinations of possible mixtures and Euclidean distances; unfortunately, their article [5] does not describe how to determine this table exactly.

2.2.7 Regularisation

The *regularisation* technique suggested by Settle and Drake (Settle and Drake [99], Drake and Settle [24]) also modifies the linear mixture modelling approach to accommodate for a favoured mixture. The motivation for modifying the standard method is that it fits both the data and the noise contained in each pixel to the model. Although this provides the best proportions estimate of each individual pixel, aggregating these estimates does not necessarily result in the best estimation of the composition of an entire region. Since the aggregated estimate is expected to contain a certain amount noise anyway, the individual estimates may be directed towards some favoured mixture $\mathbf{g}$ as long as the noise level in the overall estimate remains the same. To achieve the desired solution,

$$(\mathbf{x} - \mathbf{Mf})^T\mathbf{N}^{-1}(\mathbf{x} - \mathbf{Mf}) + \lambda(\mathbf{f} - \mathbf{g})^T(\mathbf{f} - \mathbf{g}) \tag{2.23}$$

with λ being a Lagrangian multiplier must be minimised; compared to the relaxation method described in Section 2.2.6, this expression is more general as it includes a variance-covariance matrix $\mathbf{N}$ that does not need to be equal to the identity matrix. The unconstrained, regularised estimator that can be derived from Equation (2.23) is

$$\hat{\mathbf{f}} = \lambda\mathbf{U}\mathbf{g} + \mathbf{U}\mathbf{M}^T\mathbf{N}^{-1}\mathbf{x}, \quad \text{where} \quad \mathbf{U} = (\lambda\mathbf{I} + \mathbf{M}^T\mathbf{N}^{-1}\mathbf{M})^{-1}. \tag{2.24}$$

Settle and Drake argue that $\mathbf{g}$ should be set equal to the proportions vector corresponding to the averaged signal of the pixels in the region. In this case, an optimal value of the parameter λ can be determined; see [99] for a derivation.

2.2.8 Decomposition using higher order moments

The method developed by Bosdogianni *et al.* [15] also aims at identifying the fractions of diverse coverages in a region rather than in individual pixels. Therefore, they propose a mixture model that incorporates class distributions instead of reflectance values, which are used for most of the methods explained in the previous sections. From their linear mixture model it can be derived that the first and second order moments of the probability density function for the mixed pixels in the region are the mean vector $\overline{\mathbf{x}}$ and variance-covariance matrix $\mathbf{N}$ given by

$$\begin{aligned} \overline{\mathbf{x}} &= \mathbf{Mf} \quad \text{and} && (2.25) \\ \mathbf{N} &= \sum_{i=1}^{c} f_i^2 \mathbf{N}_i, && (2.26) \end{aligned}$$

respectively; note that the variance-covariance matrix has squared fractions as weighing factors compared to normal fractions used by the classic linear mixture model. An advantage of the model based on distributions is that it consists of more equations but has the same number of unknowns, which means that the region can be unmixed into more classes than the $c \leq n+1$ limit of the classic model. The optimal fit of the model to the observed mean vector and variance-covariance matrix is defined using the least squares approach:

$$\sum_{i=1}^{n} \left(\sqrt{\frac{p}{N_{ii}}} \left(\overline{x}_i - \sum_{k=1}^{c} f_k m_{ik} \right) \right)^2 + \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\frac{2p}{N_{ij}} \left(N_{ij} - \sum_{k=1}^{c} f_k^2 N_{k_{ij}} \right) \right)^2 \qquad (2.27)$$

should be minimised to find the optimal solution for $\mathbf{f}$. In this equation, $N_{k_{ij}}$ (N_{ij}) denotes the element in the i-th row and j-th column of matrix $\mathbf{N}_k$ ($\mathbf{N}$), and p is equal to the number of pixels in the region. The weighing factors $(p/N_{ii})^{\frac{1}{2}}$ and $2p/N_{ij}$ are introduced to compensate for the different standard errors of the estimators for the mean and variance-covariance of the sample pixels. Bosdogianni *et al.* claim that the minimum of Equation (2.27) can be found by setting the partial derivatives with respect to each f_i to zero and solving the resulting equations; this analytic solution should also work when $f_c = 1 - \sum_{i=1}^{c-1} f_i$ is substituted in order to satisfy the sum-to-one constraint. Satisfaction of the positivity constraint, on the other hand, can only be enforced with a brute force approach similar to the one described in Section 2.1.2.

2.2.9 Probabilistic mixture modelling

In their study of sparsely vegetated semi-arid rangelands, Pech *et al.* [79] applied a method which estimates the fractions of a mixed pixel by the linear combination of the fractions vectors of p pixels in a training set. The weight assigned to each fractions vector $\mathbf{f}_i$ is equal to the likelihood $p(\mathbf{x}|\mathbf{f}_i)$ that pixel $\mathbf{x}$ has the underlying proportions of the i-th calibration sample, divided by the sum of these likelihoods corresponding to the entire set the calibration samples. The resulting estimator

$$\hat{\mathbf{f}} = \frac{\sum_{i=1}^{p} \mathbf{f}_i p(\mathbf{x}|\mathbf{f}_i)}{\sum_{i=1}^{p} p(\mathbf{x}|\mathbf{f}_i)}, \tag{2.28}$$

which Pech *et al.* call the "Lwin-Maritz estimator", is considered to be non-linear because $\hat{\mathbf{f}}$ is not a linear function of $\mathbf{x}$. However, it is still closely related to the classic linear mixture model as the probability that $\mathbf{x}$ originated from $\mathbf{f}_i$ is taken to be

$$p(\mathbf{x}|\mathbf{f}_i) = 2\pi^{-n/2} |\mathbf{N}|^{-1/2} \exp(-\tfrac{1}{2}(\mathbf{x} - \mathbf{M}\mathbf{f}_i)^T \mathbf{N}^{-1}(\mathbf{x} - \mathbf{M}\mathbf{f}_i)), \tag{2.29}$$

which is the probability density function of a Gaussian distribution with mean vector $\mathbf{M}\mathbf{f}_i$. Carlotto [20] suggested a similar method that uses the calibration pixel vector $\mathbf{x}_i$ instead of $\mathbf{M}\mathbf{f}_i$—it also uses a matrix $\mathbf{N}$ that is diagonal, i.e. contains only the variances in each of the n spectral bands—although the derivation of this estimator is based on a slightly different theoretical foundation. An advantage of the probabilistic approach is that there is no limitation on the number of classes a pixel can be unmixed into. Furthermore, both the sum-to-one constraint and the positivity constraint are satisfied automatically as long as the calibration samples satisfy these two constraints as well. Despite the advantages, however, the method has been reported to give worse results than the classic linear mixture modelling approach when applied to a test problem by Brown [17].

2.2.10 Neural networks

In the last decade neural networks have emerged as a powerful classification tool for, amongst others, remotely sensed image data (Benediktsson *et al.* [11], Hepner *et al.* [47], Kanellopoulos *et al.* [57]). An artificial neural network (ANN) is a system of highly connected but simple processing units called neurons, which is inspired by the architecture of the human brain. Calibration of an ANN is done by repeatedly presenting a set of training examples to it, which has the advantage that the class distributions do not have to be known nor need to comply to some parametric model. Another advantage of ANNs compared to classic statistical methods is that multiple data sources can easily be integrated (Civco and Wang [21]). In recent years, ANNs have demonstrated great potential as a method to decompose mixed pixels as well. Foody [26] found that the output of an ANN trained with pure pixels can be regarded as class membership grades, whose strength can be used to derive class proportions. More recently, studies of both Foody [27] and Schouten and Klein Gebbinck [93, 94] showed that ANNs can also be trained successfully using mixed pixels. Apart from being useful when processing an image containing few pure training pixels, the latter approach is also simpler because the network's output can be taken directly as class proportions.

Although many types of ANNs exist, for classification or decomposition of image data mostly feed-forward networks such as the multi-layer perceptron (MLP) are used. A MLP consists of units spread over an input layer, possibly some hidden layers, and an output layer—see Figure 2.5. Each neuron in the input layer is associated with one of the n spectral bands in which a pixel's reflectance is measured, while the output layer contains c units, one

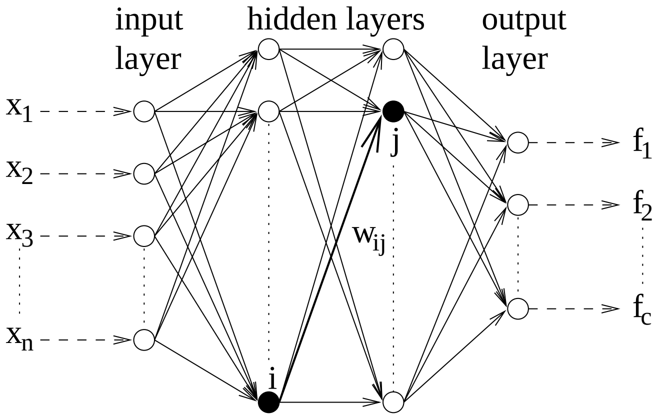


Figure 2.5: *A multi-layer perceptron designed for the decomposition of mixed pixels.*

for each of the estimated class proportions; the number of hidden layers (one or two) and the number of neurons each of these layers contains should be based on the characteristics of the remote sensing data. Between units in successive layers, weighted connections exist, which determine the function computed by the network. Each unit has an internal state or activation level a_j, depending on the output o_i of units in a previous layer, the associated weights w_{ij} of the connections, and the value of a threshold θ_j:

$$a_j = \sum_i o_i w_{ij} - \theta_j. \tag{2.30}$$

This threshold can be seen as an ordinary weight connecting unit j with a neuron having a constant activation level of 1. For hidden units, the output is calculated by passing its activation through a sigmoidal function, like

$$o_j = \frac{1}{1 + \exp(-a_j)}. \tag{2.31}$$

The output of a unit belonging to the input layer, however, is taken to be equal to the corresponding element of the pixel vector currently presented to the network, while for units in the output layer in general a linear activation function is used when mixed pixels are to be decomposed. Using this feed-forward procedure, a pixel vector is transformed into a fractions vector, whose accuracy depends on a correct setting of the weights. The usual approach to adjust the weights is error *backpropagation*, where the difference between the desired and the actual output is computed and fed back through the network, changing the weights in proportion to the error of the neurons they connect; see Appendix B for a detailed description of error backpropagation. This training procedure requires a number of decisions regarding the initial weight distribution, the learning rate, the momentum term, etc. to be made of which the influence cannot be predicted exactly. Although certain rules of thumb exist to set some of these parameters as well as the number of units in the hidden layer(s) (e.g. Kanellopoulos and Wilkinson [58], Klein Gebbinck [33], or Klein Gebbinck *et al.* [40]), tuning the network requires some experience with both ANNs and the specific data set to be processed. A further disadvantage of error backpropagation is that training can be very time-consuming, even if only few trial runs are needed to determine the optimal configuration. Once training is completed, however, the MLP can decompose mixed pixels with a speed comparable to the analytic approximation method for the linear mixture model, but with an accuracy that is somewhat higher (see Schouten and Klein Gebbinck [93]).

2.3 Summary and conclusions

In this chapter twelve methods to decompose mixed pixels were described, which is only a fraction of all methods that have been proposed in the literature. All these methods unmix a pixel to determine the proportion of its area covered by the various ground cover types, although additional information concerning the spectra of the mixed pixel's components, for instance, can be derived as well (see Chapter 3). The most widely used technique is based on the (statistical) linear mixture model, which defines the reflectance of a pixel as the linear combination of the reflectances typical for the contributing ground cover types—the endmembers—weighed by their proportion. By finding the point in the solution space of the model that is closest to the mixed pixel under observation and computing the relative distance of this point to the various endmembers, the proportions of the mixed pixel can be estimated. Of the other methods discussed, some are completely different (decomposition using higher order moments and neural networks, for instance), but most others have some important characteristics with this linear mixture modelling approach in common: either the same procedure is followed but the solution is directed to some favoured mixture (relaxation and regularisation), or the (Mahalanobis) distance between the mixed pixel and each of the endmembers is used to derive the pixel's proportions in a slightly different way (e.g. linear discriminant analysis, fuzzy classification).

An extensive comparison of the various methods for mixed pixel decomposition has not appeared in the literature so far. In part this is due to the fact that some methods are designed for different purposes than decomposing individual mixed pixels, which makes a fair comparison difficult. The estimator using higher order moments, for instance, tries to determine the composition of entire regions, while the relaxation technique specifically aims at decomposing mixed pixels representing roads with the goal to improve the connectivity of separated road segments. Furthermore, most authors compare the results of their method to the results of a classification approach in order to demonstrate the benefits of decomposition. To this end, they use their own data sets, which makes it next to impossible to relate their findings. A few comparative studies, however, do exist, albeit that the number of methods investigated usually is limited. Marsh *et al.* [68] compared the methods based on weighted averaging, simple linear regression, and linear discriminant analysis; they found that on their Landsat-MSS data the last method performed best. On the balloon MSS-like data, on the other hand, this method was clearly outperformed by the simple linear regression approach and, to some lesser degree, by the weighted averaging technique as well. Pech *et al.* [79] found that probabilistic mixture modelling and linear regression of the fractions on the signal—a method not discussed in this chapter—lead to better results than the analytic solution of the linear mixture model. Brown [17], however, reported probabilistic mixture modelling as the only estimator of the same three giving bad results on his data set. A comparison of neural networks, fuzzy c-means, and the analytic linear mixture modelling approach was made by Atkinson *et al.* [7]; according to them, neural networks gave the most accurate estimates, although the worse performance of the other two methods could be the result of poorly defined endmembers due to the lack of pure AVHRR pixels. Schouten and Klein Gebbinck, finally, confirmed that with neural networks results can be obtained that are comparable to the brute force solution of the linear mixture model and better than those achieved by the analytic approach [93] or brute force least-squares (see Section 4.1.2) [94]. A disadvantage of neural networks, however, is that, although they are fast at decomposing mixed pixels, training often takes a long time. In general, when looking for a decomposition method, one should always ask oneself the question whether the expected improvement in accuracy really outweighs the increased computational complexity of a more advanced method. While this may be true in case the decomposition method is designed specifically for the intended application, as is done in Chapter 3, it often turns out that the accuracy with which mixed pixels are decomposed depends much more on data set itself—the dispersion of the endmembers relative to the within-class variation—than on the decomposition method selected. Therefore, the fast and simple analytic solution of the linear mixture model is still used in most studies today.

Chapter 3

Decomposition based on a physical linear mixture model*

In this chapter a new linear mixture model is proposed, which has a more comprehensive physical foundation than the classic linear mixture model described in Section 2.1.1. Because the proposed physical model is more realistic, mixed pixels can be decomposed with higher accuracy. Furthermore, since the new approach explicitly models the components of a mixed pixel, additional information about its composition can be extracted. Section 3.1 describes the physical linear mixture model as well as a variant that is better suited to exploit the increased capabilities of the new model during decomposition. Section 3.2 explains how the physical model can be solved in order to estimate both the proportions and the component spectra of a mixed pixel. Section 3.3 describes an experiment that was designed to make a quantitative comparison between the decomposition methods based on the statistical and the physical linear mixture model. In Section 3.4, finally, the conclusions are presented.

3.1 The physical linear mixture model

Although the classic linear mixture model is a sensible physical model of composite reflectance, it assumes that the many small ground cover patches (called elements in Section 2.1.1) making up a pixel are statistically independent, which is why it is given the predicate "statistical" in this thesis. As a consequence, the fluctuations around the expected value $\mathbf{Mf}$ can be regarded as noise with certain statistical properties, while the error vectors $\mathbf{e}$ of two pixels are considered to be completely independent of one another. In reality, however, this is often not the case: the light and dark spots in an agricultural field, for instance, show that the error vectors of neighbouring (pure) pixels in fact are correlated. In order to explain these correlations—an example can be found in Section 3.3.2—the physical linear mixture model is introduced in Section 3.1.1. Section 3.1.2 describes how the correlations between neighbouring pixels can be modelled such that a more accurate decomposition is achieved.

*This chapter is based on publications [34] and [39].

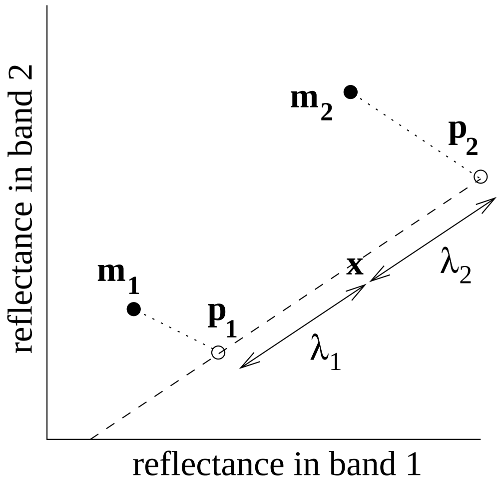


Figure 3.1: *Mixing process according to the physical linear mixture model. The two-dimensional feature space shows a mixture of two classes with global endmember spectra* $\mathbf{m}_1$ *and* $\mathbf{m}_2$*. Pixel* $\mathbf{x}$ *is a linear combination of the component spectra* $\mathbf{p}_1$ *and* $\mathbf{p}_2$*, which deviate from the global spectra due to local variations in physical factors; the proportions of these components are* $\lambda_2/(\lambda_1+\lambda_2)$ *and* $\lambda_1/(\lambda_1+\lambda_2)$*, respectively.*

3.1.1 The basic physical model

Instead of being caused by statistical noise, the lack of an exact solution of the linear mixture model without $\mathbf{e}$ can also be attributed to variations in physical factors such as humidity, soil type, elevation, etc. The scale of the variations in these physical factors usually is larger than a single pixel, which means that adjacent subpixel patches of the same ground cover type will often have a similar reflectance—i.e., they are not statistically independent—and that correlations between neighbouring pixels are likely to be found. According to this view, it is the local variation in the endmember spectra that causes a pixel to deviate from its expected value. Therefore, the *physical linear mixture model*—we give this model the predicate "physical" to stress that the deviations are not of statistical but of physical nature—can be described by the following mixing equations:

$$\mathbf{x} = (\mathbf{M} + \mathbf{E})\mathbf{f}. \tag{3.1}$$

As before, $\mathbf{x}$ represents the n-dimensional pixel, $\mathbf{f}$ the $cx1$ proportions vector, and $\mathbf{M}$ the matrix whose columns contain the c global endmember spectra $\mathbf{m}_i$. The nxc matrix $\mathbf{E}$ consists of columns $\mathbf{e}_i$ describing the local deviations from the endmember spectra that are due to physical factors; taken together, $\mathbf{m}_i + \mathbf{e}_i$ define the spectrum of the i-th component of the mixed pixel. In the physical model, the $\mathbf{e}_i$ are assumed to have a multivariate normal distribution with a zero mean and a variance-covariance matrix $\mathbf{N}_i$. This assumption is identical to the one made for the classic linear mixture model, which can be seen by comparing the statistical distribution of pure pixels of class i in both models. A graphical representation of the physical linear mixture model is given in Figure 3.1, which shows a simple mixture of two classes in two dimensions.

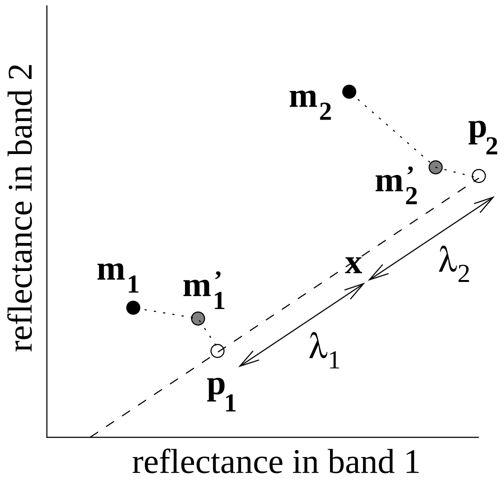


Figure 3.2: *Mixing process according to the environmental linear mixture model. In this model, the distance between the component spectrum* $\mathbf{p}_i$ *and the global endmember spectrum* $\mathbf{m}_i$ *is split into an environmental part—the dotted line from* $\mathbf{m}_i$ *to* $\mathbf{m}'_i$*—and a statistical part—the dotted line from* $\mathbf{m}'_i$ *to* $\mathbf{p}_i$*. If the global endmembers* $\mathbf{m}_i$ *are replaced by their local counterparts* $\mathbf{m}'_i$*, which must be done for each mixed pixel* $\mathbf{x}$ *separately, the physical model and the environmental model are identical (except for the definition of the* $\mathbf{N}_i$*).*

3.1.2 The environmental model

Although the basic physical model provides a realistic explanation for the deviation between the expected and the measured reflectance values, it does not explicitly model the fact that neighbouring pixels are correlated. Consequently, it is difficult to use the information contained in these correlations. To facilitate the exploitation of this information, the error matrix $\mathbf{E}$ of the physical model was split into an environmental and a statistical part, resulting in the *environmental linear mixture model* with the following mixing equations:

$$\mathbf{x} = (\mathbf{M} + \mathbf{E}_1 + \mathbf{E}_2)\mathbf{f}. \tag{3.2}$$

The environmental part $\mathbf{E}_1$ represents the deviations from the global endmember spectra that are typical for the local environment of mixed pixel $\mathbf{x}$. This part is considered to be deterministic as it can be calculated from other pixels in the neighbourhood of $\mathbf{x}$, which share the same deviation. The statistical part $\mathbf{E}_2$ accounts for the small variations between pixels with the same $\mathbf{E}_1$ and for the smooth transitions between pixels with different $\mathbf{E}_1$, which are both due to the more or less gradual changes in (physical) conditions. Each column $\mathbf{e}_{i,2}$ of $\mathbf{E}_2$ is once again assumed to have a multivariate normal distribution with a zero mean and a variance-covariance matrix $\mathbf{N}_i$. It must be noted, however, that the matrices used for the statistical and the physical model differ from the ones used for the environmental model: whereas the former matrices describe the variations in the endmember spectra over the entire image, the latter matrices represent the variations in the endmember spectra of neighbouring pixels. In Section 3.3.1 a specific example is given of how to derive the variance-covariance matrices of the environmental model and how to determine the environmental part of the endmember spectra from the neighbouring pixels when dealing with agricultural scenes. Once these parameters are known, the environmental and the physical model can be unified: substitution of $\mathbf{M}$ with $\mathbf{M}'=\mathbf{M}+\mathbf{E}_1$—so $\mathbf{M}'$ depends on the pixel under observation but is still deterministic—leaves a small statistical term as the only variable factor; compare Figures 3.1 and 3.2 to see the resemblance of the two models.

3.2 Solving the physical model

In order to decompose mixed pixels, the physical linear mixture model has to be solved for the observed reflectance values. In contrast to classic linear mixture modelling (and all other decomposition methods described in Chapter 2), this approach does not only yield an estimate of the component proportions but also provides an estimate of the component spectra. Although these spectra may be unimportant for some applications, it may be a valuable source of information for others. As was the case with the statistical model, the solution of the physical model has to satisfy two constraints: the *sum-to-one constraint* (Equation (2.6)), which says that the proportions must add to unity, and the *positivity constraint* (Equation (2.7)), which demands that each fraction is non-negative. If the positivity constraint is abandoned, one can find an exact solution of the resulting set of equations having a zero error matrix $\mathbf{E}$ for all mixtures where the number of endmembers is at least equal to the number of spectral bands. If the number of endmembers is smaller, a non-zero $\mathbf{E}$ is required, which can be used to select some sort of optimal solution among the infinity of possibilities. The maximum likelihood approach chooses the solution that has the highest probability, i.e. that combination of $\mathbf{E}$ and $\mathbf{f}$ that solves the mixing equations (Equation (3.1)) while minimising

$$\sum_{i=1}^{c} \mathbf{e}_i^T \mathbf{N}_i^{-1} \mathbf{e}_i. \tag{3.3}$$

As can be derived from the previous section, the environmental model can be solved in a similar fashion by substituting the global endmember matrix $\mathbf{M}$ with the local endmembers $\mathbf{M}+\mathbf{E}_1$—remember that the local deviation may differ for each pixel—and using different variance-covariance matrices $\mathbf{N}_i$. Instead of minimising $\mathbf{E}$ itself, we now minimise $\mathbf{E}_2$, which is equal to minimising the differences between the $\mathbf{E}$ of neighbouring pixels. Calculation of the local endmembers and variance-covariance matrices strongly depends on the application domain and can be quite complicated. A way to derive these parameters of the environmental model in case mixed pixels on agricultural field boundaries are to be decomposed is described in Section 3.3.1.

The maximum likelihood approach picks the best solution from a number of alternatives, but it does not come up with a solution by itself. In fact, finding a solution is rather difficult—much more difficult than for the statistical model—because of the complex relations between the $\mathbf{e}_i$ that must be realised in order to comply to Equation (3.1). Currently, two methods for a joint retrieval of $\mathbf{E}$ and $\mathbf{f}$ exist: brute force and analytic approximation.

3.2.1 Brute force approximation

To understand the complex relations between the $\mathbf{e}_i$, a look at the n-dimensional feature space is helpful. According to *convex geometry* principles, pixels that are composed of c

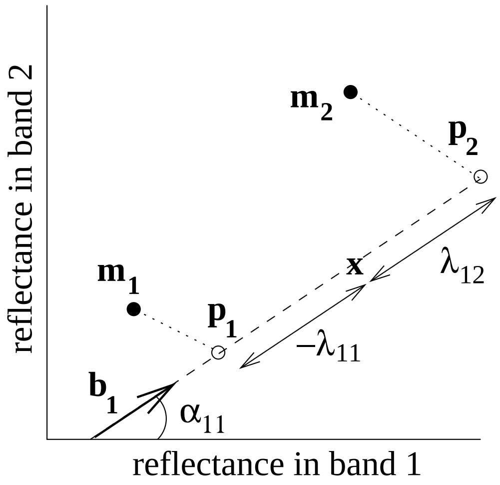


Figure 3.3: *Maximum likelihood solution of the physical linear mixture model. To solve the model, one point* $\mathbf{p}_i$ *per endmember has to be determined, whose Mahalanobis distance to* $\mathbf{m}_i$ *is minimal. The sum-to-one constraint demands that these points lie on a line through* $\mathbf{x}$*; the positivity constraint requires that* $\mathbf{p}_1$ *and* $\mathbf{p}_2$ *lie on either side of* $\mathbf{x}$*. For the subspace defined by basis vecor* $\mathbf{b}_1$*, the optimal position of* $\mathbf{p}_i$ *is given by* λ_{1i}*. The angle* α_{11} *that results in the smallest sum of distances defines the best solution of the model.*

linearly independent endmembers lie in the c-dimensional linear subspace spanned by the vectors $\mathbf{m}_i$ that represent the endmember spectra. If the proportions of the mixed pixel sum to one, this subspace is restricted to the $(c{-}1)$-dimensional subspace containing the $\mathbf{m}_i$ but not (necessarily) the origin of the coordinate system; the latter subspace is further limited to the simplex defined by the $\mathbf{m}_i$ if all fractions are greater than (or equal to) zero. As was described in Section 3.1, the physical linear mixture model attributes the occurrence of pixels outside this simplex to local variations in the endmember spectra. Solving the physical model, therefore, requires that for each endmember a point $\mathbf{p}_i$ is found such that these vectors span a c-dimensional linear subspace containing $\mathbf{x}$. If the sum-to-one constraint is to be satisfied, $\mathbf{x}$ must also be contained in the $(c{-}1)$-dimensional subspace through the $\mathbf{p}_i$ (but not necessarily through the origin), and satisfaction of the positivity constraint demands that $\mathbf{x}$ is contained in the simplex made up of the $\mathbf{p}_i$. Even with these constraints, the number of possible solutions still is infinite. The solution that is favoured by the maximum likelihood approach—the one minimising Equation (3.3)—has the smallest sum of Mahalanobis distances between the $\mathbf{p}_i$ and their corresponding $\mathbf{m}_i$. Examples of this geometric view with two and three endmembers are given in Figures 3.3 and 3.4, respectively. Based on the insights gained from the above analysis, a method to solve the physical linear mixture model can be developed.

It is important to see that once the points $\mathbf{p}_i$ that define a $(c{-}1)$-dimensional subspace containing $\mathbf{x}$ have been determined, a proportions vector that solves the physical model and satisfies the sum-to-one constraint can be derived. In n-space, a linear subspace of dimension $c{-}1$ is spanned by $c{-}1$ linearly independent $nx1$ vectors $\mathbf{b}_j$. Thus, a point $\mathbf{p}_i$ in the linear subspace through $\mathbf{x}$ can be described as

$$\mathbf{p}_i = \mathbf{x} + \lambda_{1i}\mathbf{b}_1 + \cdots + \lambda_{(c-1)i}\mathbf{b}_{c-1} = \mathbf{x} + \mathbf{B}\boldsymbol{\lambda}_i, \tag{3.4}$$

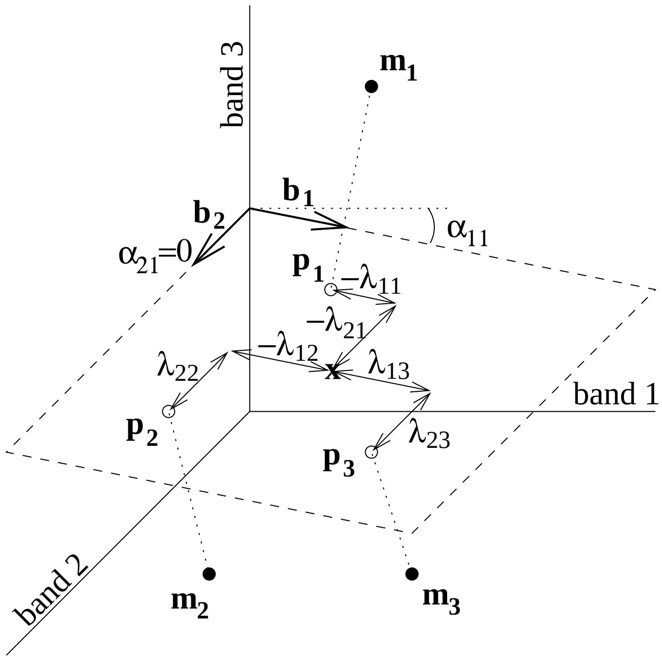


Figure 3.4: *Solution of the physical model in a three-dimensional feature space. The basis vectors* $\mathbf{b}_1$ *and* $\mathbf{b}_2$*, which are controlled by* α_{11} *and* α_{21}*, respectively, define a plane through* $\mathbf{x}$*. Each point* $\mathbf{p}_i$*, which also lies on this plane and is described by* λ_{1i} *and* λ_{2i}*, has a minimal Mahalanobis distance to* $\mathbf{m}_i$*. The optimal solution of the model is given by the combination of* α_{11} *and* α_{21} *that leads to the smallest sum of distances.*

where $\boldsymbol{\lambda}_i$ represents the $(c-1)x1$ column vector $[\lambda_{1i}, \ldots, \lambda_{(c-1)i}]^T$ and $\mathbf{B}$ denotes the $nx(c-1)$ matrix with columns $\mathbf{b}_j$. Since $\mathbf{p}_i = \mathbf{m}_i + \mathbf{e}_i$, substituting the above expression in Equation (3.1) results in a system from which $\mathbf{f}$ can be solved. If this system is expressed in the coordinate system defined by basis vectors $\mathbf{b}_1, \ldots, \mathbf{b}_{c-1}$ and origin $\mathbf{x}$, then

$$f_1\boldsymbol{\lambda}_1 + \cdots + f_c\boldsymbol{\lambda}_c = \Lambda\mathbf{f} = \mathbf{0}, \tag{3.5}$$

with $\mathbf{0}$ denoting the $((c-1)x1)$ null vector and Λ being the matrix consisting of columns $\boldsymbol{\lambda}_i$. This system of equations has at least one non-trivial solution, because c vectors—the $\boldsymbol{\lambda}_i$—of dimension $(c-1)$ always are linearly dependent. Moreover, since the subspace containing the $\mathbf{p}_i$ has to be exactly of dimension $(c-1)$, there will be precisely one solution modulo a scaling factor. If this scaling factor is set such that the sum-to-one constraint is satisfied, and the definition $f_c = 1 - \sum_{i=1}^{c-1} f_i$ is taken, Equation (3.5) can be rewritten as

$$\Lambda'\mathbf{f}' = -\boldsymbol{\lambda}_c, \quad \text{where} \quad \Lambda' = [\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_c, \ldots, \boldsymbol{\lambda}_{c-1} - \boldsymbol{\lambda}_c] \tag{3.6}$$

and $\mathbf{f}'$ is the proportions vector $\mathbf{f}$ minus its last element f_c. This last system is well-known, and its solution is usually given using Cramer's rule (Strang [107]):

$$f_i = \frac{|\Lambda'_i(-\boldsymbol{\lambda}_c)|}{|\Lambda'|}. \tag{3.7}$$

As before, $|\ |$ stands for taking the determinant; $\Lambda'_i(-\boldsymbol{\lambda}_c)$ is the $((c-1)x(c-1))$ matrix Λ' with the i-th column replaced by $-\boldsymbol{\lambda}_c$. A simple example of this formula is the case where a mixed pixel consists of only two components (see Figure 3.3): $\mathbf{f} = (\lambda_{12}/(\lambda_{12} - \lambda_{11}), -\lambda_{11}/(\lambda_{12} - \lambda_{11}))^T$. Although more components lead to a more complex expression, calculation of the proportions vector remains straightforward.

The first step to find the best $\mathbf{E}$ according to the maximum likelihood criterion is to determine the optimal position of the $\mathbf{p}_i$ for a single $(c-1)$-subspace. Let the subspace of interest be given by the matrix $\mathbf{B}$ as described in the previous paragraph. Since $\mathbf{e}_i = \mathbf{p}_i - \mathbf{m}_i$, the deviation from the i-th endmember can be expressed using Equation (3.4):

$$\mathbf{e}_i = (\mathbf{x} - \mathbf{m}_i) + \mathbf{B}\boldsymbol{\lambda}_i. \tag{3.8}$$

As was explained earlier in this section, because of this construction, any choice of the $\boldsymbol{\lambda}_i$ will lead to a solution of the physical linear mixture model. Since no other relations between the $\mathbf{e}_i$ have to be reckoned with, the minimum of Equation (3.3) is found by minimising the individual parts of the summation. For the i-th component this means that, after some manipulation, the minimum of

$$\mathbf{e}_i^T \mathbf{N}_i^{-1} \mathbf{e}_i = (\mathbf{m'}_i - \mathbf{B}\boldsymbol{\lambda}_i)^T \mathbf{N}_i^{-1} (\mathbf{m'}_i - \mathbf{B}\boldsymbol{\lambda}_i) \quad \text{with} \quad \mathbf{m'}_i = \mathbf{m}_i - \mathbf{x} \tag{3.9}$$

has to be determined. Analogous to the derivation of Equation (2.10), successively setting to zero the partial derivatives of this equation with respect to $\lambda_{1i}, \ldots, \lambda_{(c-1)i}$ gives the following estimator:

$$\hat{\boldsymbol{\lambda}}_i = (\mathbf{B}^T \mathbf{N}_i^{-1} \mathbf{B})^{-1} \mathbf{B}^T \mathbf{N}_i^{-1} \mathbf{m'}_i. \tag{3.10}$$

The optimal solution of the physical model for the $(c-1)$-subspace defined by $\mathbf{B}$ can now be calculated simply by substituting the estimated $\boldsymbol{\lambda}_i$ in Equations (3.7) and (3.8).

Before the optimal $\mathbf{E}$ of all $(c-1)$-subspaces can be derived, first the $c-1$ linearly independent basis vectors $\mathbf{b}_j$ have to be defined. One possibility to construct d basis vectors for a n-dimensional space is based on the usual definition of a d-dimensional space with d orthonormal vectors. For each dimension k higher than d, basis vector $\mathbf{b}_j$ is multiplied by $\cos\alpha_{jk}$ and expanded with a term $\sin\alpha_{jk}$. For example, if d=1, $\mathbf{b}_1$ is constructed following

$$[1] \;\rightarrow\; \begin{bmatrix} \cos\alpha_{11} \\ \sin\alpha_{11} \end{bmatrix} \;\rightarrow\; \begin{bmatrix} \cos\alpha_{11}\cos\alpha_{12} \\ \sin\alpha_{11}\cos\alpha_{12} \\ \sin\alpha_{12} \end{bmatrix} \;\rightarrow\; \cdots \quad \text{for } n\text{=1, 2, 3, \ldots ,} \tag{3.11}$$

while in case d=2, $(\mathbf{b}_1, \mathbf{b}_2)$ is formed according to

$$\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right) \;\rightarrow\; \left(\begin{bmatrix} \cos\alpha_{11} \\ 0 \\ \sin\alpha_{11} \end{bmatrix}, \begin{bmatrix} 0 \\ \cos\alpha_{21} \\ \sin\alpha_{21} \end{bmatrix} \right) \;\rightarrow\; \cdots \quad \text{for } n\text{=2, 3, \ldots} \tag{3.12}$$

A graphical representation of these cases is shown in Figures 3.3 (n=2) and 3.4 (n=3), respectively. Unfortunately, except in case d=1, some subspaces are multiply defined, while others are left out. For instance, if d=2 and n=3, all combinations of $\alpha_{11}=\frac{\pi}{2}$ and an arbitrary value of α_{21} result in the same subspace (YZ-plane). On the other hand, no subspace perpendicular to the XY-plane can be formed other than the YZ- and XZ-planes

($\alpha_{21} = \frac{\pi}{2}$). This problem can be solved by modifying the definition of the basis vectors for certain values of the angles α_{jk}. For d=2 and n=3 this means that $(\mathbf{b}_1, \mathbf{b}_2)$ is defined by Equation (3.12) if $\alpha_{11}{\neq}\frac{\pi}{2}$ and $\alpha_{21}{\neq}\frac{\pi}{2}$, and by

$$(\mathbf{b}_1, \mathbf{b}_2) = \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} \cos\alpha_{21} \\ \sin\alpha_{21} \\ 0 \end{bmatrix} \right) \tag{3.13}$$

otherwise. For higher values of d and n similar modifications can be made. Although this construction is rather complex, it does provides a set of linearly independent basis vectors.

The next step in finding the optimal $\mathbf{E}$ is to determine which ($c{-}1$)-subspace gives the lowest value of the maximum likelihood criterion. In order to do so, the expressions for $\mathbf{e}_i$, $\hat{\boldsymbol{\lambda}}_i$, and $\mathbf{b}_j$ have to be substituted in Equation (3.3), and an expression for the α_{jk}—the only variables left—that minimise the resulting equation has to be derived. However, due to the complexity of the expression to be minimised and the conditional definition of the basis vectors, finding a general analytic solution is practically impossible. Therefore, the optimal solution is approximated by checking a large number of ($c{-}1$)-subspaces using brute force. The logical set of subspaces to be checked is the one whose elements are uniformly distributed over the entire n-space, but this set cannot easily be generated. A more obvious set consists of the subspaces that are spread evenly over the $d(n{-}d)$-dimensional parameter space associated with the α_{jk}. This latter set can be created more easily by taking regularly spaced samples of α_{jk} over the interval $(-\frac{\pi}{2}, \frac{\pi}{2}]$ and calculating the corresponding $\mathbf{b}_j$. The computational complexity of this approach is considerable: if the sampling rate is given by r, the maximum likelihood criterion has to be evaluated about $r^{(c-1)(n-c+1)}$ times. By varying the sampling rate, the accuracy of the approximation of $\mathbf{E}$ can be traded off against computational costs.

In the physical linear mixture model, the positivity constraint, which has been ignored so far, can be satisfied quite easily. As was described earlier in this section, satisfaction of this constraint demands that $\mathbf{x}$ is contained in the simplex defined by the $\mathbf{p}_i$. For each ($c{-}1$)-subspace for which this condition does not hold, at least one of the $\mathbf{p}_i$ has to be moved away from its optimal position. Analysis of Equation (3.9) shows that this displacement should be as small as possible, which implies that $\mathbf{x}$ will lie on the hull of the resulting simplex. It is of course possible to determine for each subspace how the $\mathbf{p}_i$ can best be displaced before the cost function (Equation (3.3)) is computed, but the optimal solution that satisfies the positivity constraint can also be found in a more convenient way. The fact that $\mathbf{x}$ lies on the hull of the simplex means that at least one of the proportions is zero. Consequently, we can deal with all non-complying subspaces at once by decomposing $\mathbf{x}$ while leaving out one endmember at the time, and comparing all c solutions with the best solution based on c endmembers that does satisfy the positivity constraint. Since the minimum of Equation (3.3) when using $c{-}1$ components will be at least as high as when c components are used, this recursive step only needs to be taken if the optimal solution found before did not already satisfy the positivity constraint. The second case in which the recursion stops is that c has become equal to one: as $\mathbf{x}$ presumably is a pure pixel, its proportions vector will be equal to 1. Although the computational costs of this approach may be considerable, it is a straightforward way to enforce the positivity of the proportions.

3.2.2 Analytic approximation

As was done for the statistical model, the optimal solution of the physical model can be approximated using a closed formula as well. Once again it is required that all variance-covariance matrices $\mathbf{N}_i$ are identical, which can be accomplished by taking the mean matrix $\mathbf{N} = 1/c\sum_{i=1}^{c}\mathbf{N}_i$. As long as the $\mathbf{N}_i$ are small relative to the dispersion of the endmembers—if not, the estimated solution is probably poor anyway—the resulting loss of accuracy is not very large (Horwitz *et al.* [51]). Under the assumption of equal variance-covariance matrices, the expression to be minimised is $\sum_{i=1}^{c}\mathbf{e}_i^T\mathbf{N}^{-1}\mathbf{e}_i$, subject to the conditions $\mathbf{x} = (\mathbf{M}+\mathbf{E})\mathbf{f}$ and $\sum_{i=1}^{c} f_i = 1$. According to Lagrangian analysis, this boils down to minimising

$$\mathrm{Trace}(\mathbf{E}^T\mathbf{N}^{-1}\mathbf{E}) + 2\boldsymbol{\alpha}^T(\mathbf{x} - (\mathbf{M}+\mathbf{E})\mathbf{f}) + 2\beta(\mathbf{1}^T\mathbf{f} - 1), \tag{3.14}$$

where Trace($\mathbf{A}$) represents the sum of the diagonal entries of matrix $\mathbf{A}$. Furthermore, $\boldsymbol{\alpha}$ stands for an $nx1$ vector of Lagrangian constants, and β denotes another Lagrangian constant. The minimum of Equation (3.14) is found by differentiation and setting the result to zero. Differentiating with respect to $\mathbf{E}$ gives

$$2\mathbf{N}^{-1}\mathbf{E} - 2\boldsymbol{\alpha}\mathbf{f}^T = \mathbf{0}, \quad \text{or, equivalently,} \quad \mathbf{E} = \mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T \tag{3.15}$$

(the nxc matrix $\mathbf{0}$ is filled with zeroes); differentiating with respect to $\mathbf{f}$ results in

$$(\mathbf{M}+\mathbf{E})^T\boldsymbol{\alpha} = \beta\mathbf{1}, \tag{3.16}$$

where $\mathbf{1}$ is a $(cx1)$ vector of ones only. From this last equation, a relation between β and the other Lagrangian constants can be derived:

$$\begin{aligned}
&(\mathbf{M}+\mathbf{E})^T\boldsymbol{\alpha} = \mathbf{1}\beta && \Longrightarrow \quad \{*\mathbf{f}^T\}\\
&\mathbf{f}^T(\mathbf{M}+\mathbf{E})^T\boldsymbol{\alpha} = \mathbf{f}^T\mathbf{1}\beta && \Longrightarrow \quad \{(3.1),\ (2.6)\}\\
&\mathbf{x}^T\boldsymbol{\alpha} = \beta.
\end{aligned} \tag{3.17}$$

Based on Equations (3.15) to (3.17), a fast method for the simultaneous retrieval of $\mathbf{E}$ and $\mathbf{f}$ can be developed.

The general principle followed to derive an estimate of the proportions was to combine the above equations in trying to eliminate the Lagrangian constants until an expression containing only the unknown fractions vector $\mathbf{f}$ was left. Starting with Equation (3.16), one can make the following derivation:

$$\begin{aligned}
&(\mathbf{M}+\mathbf{E})^T\boldsymbol{\alpha} &&= \mathbf{1}\beta && \Longrightarrow \quad \{(3.15),\ (3.17)\}\\
&(\mathbf{M}+\mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T)^T\boldsymbol{\alpha} &&= \mathbf{1}(\mathbf{x}^T\boldsymbol{\alpha}) && \Longrightarrow \quad \{\text{unfold, } \mathbf{N}^T{=}\mathbf{N}\}\\
&\mathbf{M}^T\boldsymbol{\alpha} + (\mathbf{f}\boldsymbol{\alpha}^T\mathbf{N})\boldsymbol{\alpha} &&= \mathbf{1}\mathbf{x}^T\boldsymbol{\alpha} && \Longrightarrow \quad \{\text{rearrange}\}\\
&\mathbf{M}^T\boldsymbol{\alpha} - \mathbf{1}\mathbf{x}^T\boldsymbol{\alpha} &&= -\mathbf{f}\boldsymbol{\alpha}^T\mathbf{N}\boldsymbol{\alpha} && \Longrightarrow \quad \{\text{fold}\}\\
&(\mathbf{M} - \mathbf{x}\mathbf{1}^T)^T\boldsymbol{\alpha} &&= -\mathbf{f}(\boldsymbol{\alpha}^T\mathbf{N}\boldsymbol{\alpha}).
\end{aligned} \tag{3.18}$$

The mixing equations described by Equation (3.1) can be transformed as follows:

$$
\begin{aligned}
(\mathbf{M}+\mathbf{E})\mathbf{f} &= \mathbf{x} && \Longrightarrow \quad \{(3.15)\}\\
(\mathbf{M}+\mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T)\mathbf{f} &= \mathbf{x} && \Longrightarrow \quad \{\text{unfold, rearrange}\}\\
\mathbf{M}\mathbf{f}-\mathbf{x} &= -(\mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T)\mathbf{f} && \Longrightarrow \quad \{(2.6)\}\\
\mathbf{M}\mathbf{f}-\mathbf{x}\mathbf{1}^T\mathbf{f} &= -\mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T\mathbf{f} && \Longrightarrow \quad \{\text{fold}\}\\
(\mathbf{M}-\mathbf{x}\mathbf{1}^T)\mathbf{f} &= -\mathbf{N}\boldsymbol{\alpha}(\mathbf{f}^T\mathbf{f}).
\end{aligned}
\tag{3.19}
$$

Finally, we multiply the above result with $(\mathbf{M}-\mathbf{x}\mathbf{1}^T)^T\mathbf{N}^{-1}$ to combine it with Equation (3.18):

$$
\begin{aligned}
(\mathbf{M}-\mathbf{x}\mathbf{1}^T)^T\mathbf{N}^{-1}(\mathbf{M}-\mathbf{x}\mathbf{1}^T)\mathbf{f} &= (\mathbf{M}-\mathbf{x}\mathbf{1}^T)^T\mathbf{N}^{-1}(-\mathbf{N}\boldsymbol{\alpha}(\mathbf{f}^T\mathbf{f}))\\
&= -(\mathbf{M}-\mathbf{x}\mathbf{1}^T)^T\boldsymbol{\alpha}(\mathbf{f}^T\mathbf{f})\\
&= \mathbf{f}(\boldsymbol{\alpha}^T\mathbf{N}\boldsymbol{\alpha})(\mathbf{f}^T\mathbf{f}).
\end{aligned}
\tag{3.20}
$$

In this last expression the well-known system $\mathbf{A}\mathbf{f}=\lambda\mathbf{f}$ can be recognised, given that $\mathbf{A}=(\mathbf{M}-\mathbf{x}\mathbf{1}^T)^T\mathbf{N}^{-1}(\mathbf{M}-\mathbf{x}\mathbf{1}^T)$ and $\lambda=(\boldsymbol{\alpha}^T\mathbf{N}\boldsymbol{\alpha})(\mathbf{f}^T\mathbf{f})$. The $\mathbf{f}$ that solve this system are the eigenvectors of matrix $\mathbf{A}$, which can be calculated in a straightforward fashion (see Strang [107]). From these alternatives, the $\mathbf{f}$ that minimises $\text{Trace}(\mathbf{E}^T\mathbf{N}^{-1}\mathbf{E})$ is selected; since $\text{Trace}(\mathbf{E}^T\mathbf{N}^{-1}\mathbf{E})=\text{Trace}(\mathbf{f}\boldsymbol{\alpha}^T\mathbf{N}\mathbf{N}^{-1}\mathbf{N}\boldsymbol{\alpha}\mathbf{f}^T)=(\boldsymbol{\alpha}^T\mathbf{N}\boldsymbol{\alpha})(\mathbf{f}^T\mathbf{f})$, the eigenvector corresponding to the smallest eigenvalue λ is considered to be the optimal solution. Subsequently, the error matrix $\mathbf{E}$ is calculated by combining Equations (3.15) and (3.19):

$$
\mathbf{E}=-\frac{(\mathbf{M}-\mathbf{x}\mathbf{1}^T)\mathbf{f}\mathbf{f}^T}{\mathbf{f}^T\mathbf{f}}. \tag{3.21}
$$

Note—expand $(\mathbf{M}+\mathbf{E})\mathbf{f}$—that this expression requires that the sum-to-one constraint is satisfied. Since some standard packages supply eigenvectors that are normalised according to Minkowski's L2-norm, the proportions vector may have to be rescaled. In order to satisfy the positivity constraint, some post-processing is necessary as well. Of the approaches suggested in Section 4.1, the simplest method is to set negative proportions to zero and rescale the fractions vector. Although this approach introduces some further inaccuracy, its computational costs are very low. As a result, the overall method to approximate the optimal solution of the physical linear mixture model is relatively fast.

3.3 Comparing the statistical and the physical model

In order to determine the suitability of the statistical and the physical model—the basic model as well as the environmental variant—several experiments were carried out. In these experiments, the accuracy of each approach was evaluated using artificial images, which were needed to make a quantitative comparison possible. The set-up of the experiments is described in Section 3.3.1, while the results are discussed in Section 3.3.2.

3.3.1 The experimental set-up

Each experiment consisted of three steps. In the first step, an image of an agricultural scene was generated with fields having certain statistical properties. The second step comprised the processing of the image by two methods based on each of the three mixture models. In the third and last step, the results of the six methods were quantified using several accuracy measures.

For the simulation of remotely-sensed images, a modified version of the method introduced by Schouten and Klein Gebbinck [92] was used. First of all, the method, which is described in more detail in Section 6.2.1, was somewhat simplified to facilitate the decomposition of mixed pixels based on the environmental model. One simplification was that no ground cover type was assigned to the edges, which ensured that most mixed pixels consisted of only two classes. Although this made the images somewhat less realistic, it was expected to have only a minor (positive) influence on the accuracy of the environmental approach in particular while the processing time needed by the brute force methods was reduced significantly. Another simplification was that neighbouring fields were not allowed to have the same ground cover type, which ensured that (mixed) pixels located on field boundaries always consisted of more than one class. This restriction did not seriously affect the outcome of the experiments either, because none of the decomposition methods is able to make an accurate division of a mixed pixel's area between two fields of the same ground cover type anyway. The second type of modification to the method concerned the compilation of the class templates. In order to prevent the correlations between neighbouring pixels from being disturbed, each template was generated using a single pixel block clipped out of the middle of a large agricultural field belonging to a particular class. Since these clippings comprised only $15x15$ pixels, several copies had to be mirrored and stitched together to get templates of sufficient size in which the correlations were preserved. This way, four templates labelled A–D were derived from a Landsat-TM image of Portugal, whose statistical distributions are shown in Figure 3.5. All artificial images were made using only two templates at the same time, while the number of bands was reduced from six to two or three. In most cases bands 3 and 5 (plus band 7 for n=3) were selected, because they provided the best discrimination between vegetation types (Sabins [88]). The only exceptions were two images created with templates A′ and C′, which were derived from the six band templates A and C by selecting bands 1 and 2 (plus band 4 in case n=3). With this method, 12 images were generated that each contained 297 mixed pixels to be decomposed.

Each simulated image was processed by six different decomposition methods. These methods can be grouped based on their underlying model—statistical, physical, or environmental—as well as on the method of solution applied—brute force or mathematical analysis. As was described in Sections 2.1.2 and 3.2, the brute force approaches approximate the optimal solution by searching the entire solution space; the analytic methods, on the other hand, approximate the variance-covariance matrices of all classes by their average such that the optimal solution can be expressed as a closed formula. The endmember distributions used for the statistical and the physical model were estimated by calculating the mean vector and variance-covariance matrix of the pixels of each class template. For the environmental model, the endmember distributions had to be calculated totally differently, which will be described in the remainder of this paragraph. Since the simulated images represented agricultural scenes, each mixed pixel was surrounded by pure pixels of the fields it was composed of. The spectral response of such a pure pixel q is approximately equal to one of the local endmember spectra (say i) of the neighbouring mixed pixel p, albeit that it also includes a small statistical term $\mathbf{e}_{i,2,q}$. The only consequence of this minor inconvenience is that, after substituting $\mathbf{x}_q$ for $\mathbf{m}_i + \mathbf{e}_{i,1,p}$, the term $\mathbf{e}_{i,2,p} - \mathbf{e}_{i,2,q}$ instead of $\mathbf{e}_{i,2,p}$ has to be determined in order to solve the model. The distribution of this subtraction sum was estimated based on the differences calculated between all pairs of neighbouring pixels in the corresponding template. As expected, the mean vector of the subtraction sum turned out to be $\mathbf{0}$ for all templates. The only issue that remained was which neighbouring pixels were to be used to approximate the local endmember spectra. This problem was solved by trying all combinations of neighbouring pure pixels—one pixel per class—and selecting the combination that minimised Equation (3.3) (note that $\mathbf{e}_i$ now represents the subtraction sum). Thus, by defining different endmember distributions for all mixed pixels, the environmental approach was unified with the basic physical approach.

The performance of each method was expressed by three accuracy measures. To measure how accurately the proportions of a mixed pixel could be estimated, the pixel's true proportions f_i were compared with their estimates $\hat{f}_i$:

$$\varepsilon_f = \left\langle \frac{1}{2} \sum_{i=1}^{c} \left| \hat{f}_i - f_i \right| \right\rangle . \tag{3.22}$$

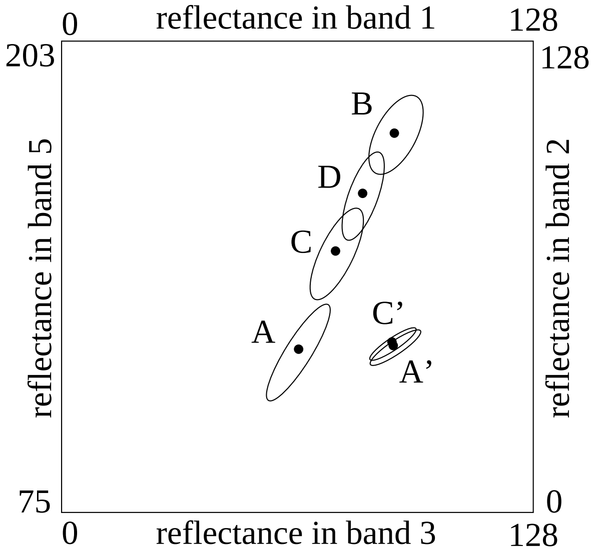
In this equation, $|\ |$ stands for taking the absolute value, and $< >$ denotes that the enclosed expression is averaged over all mixed pixels. The scalar $\frac{1}{2}$ was introduced because overestimation of one class automatically leads to underestimation of the other; due to this scalar the value of ε_f was restricted to the interval [0,1]. The second accuracy measure expressed how large the statistical deviations had to be in order to reconstruct the mixed pixel and is defined by

$$\varepsilon_x = \left\langle \frac{1}{n} \mathbf{e}^T (\mathbf{N}(\mathbf{f}))^{-1} \mathbf{e} \right\rangle \quad \text{or} \quad \varepsilon_x = \left\langle \frac{1}{nc} \sum_{i=1}^{c} \mathbf{e}_i^T \mathbf{N}_i^{-1} \mathbf{e}_i \right\rangle , \tag{3.23}$$

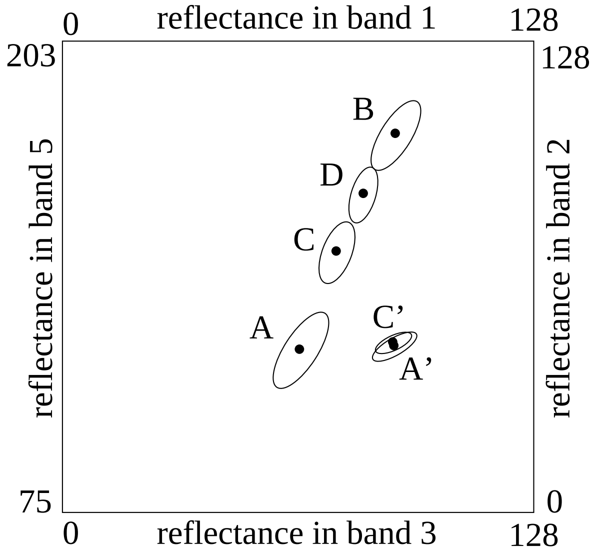
depending on whether the statistical or one of the physical linear mixture models was assumed. Although the magnitude of the deviations is of no direct importance, it may be an indication of the reliability of the reconstruction. Obviously, ε_x cannot be used to compare the methods based on different mixture models. The third and last measure was determined for the methods based on the physical models only and indicated how well the component spectra of a mixed pixel could be estimated by calculating the difference between the true value of the i-th component $\mathbf{p}_i$ and its estimate $\hat{\mathbf{p}}_i$:

$$\varepsilon_{p_i} = \left\langle \frac{1}{n} (\hat{\mathbf{p}}_i - \mathbf{p}_i)^T \mathbf{N}_i^{-1} (\hat{\mathbf{p}}_i - \mathbf{p}_i) \right\rangle . \tag{3.24}$$

Note that in this equation as well as in Equation (3.23) different $\mathbf{N}_i$ were used for the basic physical approach and the environmental approach. With these three measures, a quantitative comparison of the performance of the different methods was made in a straightforward way.



(a) Distributions statistical/physical model.



(b) Distributions environmental model.

Figure 3.5: *Two-dimensional feature space showing six class distributions. A–D are extracted from bands 3 and 5, while A′ and C′ are extracted from bands 1 and 2. (a) Class mean* $\mathbf{m}_i$ *surrounded by the ellipse* $\mathbf{x}^T\mathbf{N}_i^{-1}\mathbf{x} = 4$*, where* $\mathbf{N}_i$ *represents the variation in the pixels themselves. (b) Similar set-up, except that the ellipses now represent the variation in the differences between neighbouring pixels.*

3.3.2 Results and discussion

Two important factors that influence the accuracy of a decomposition method are the relative positions of the endmembers in the feature space and the statistical distribution of the error components. Figure 3.5(a) shows these class characteristics as computed for both the statistical and the basic physical model, while Figure 3.5(b) displays the characteristics as determined for the environmental model. In both figures it can be seen that the dispersion of the endmembers A–D is relatively large, whereas the distance between the endmembers A′ and C′ is relatively small. The latter two classes were defined by selecting bands 1 and 2 instead of the usual bands 3 and 5, which was done in order to simulate an image containing mixed pixels that were difficult to decompose. Addition of a third band—band 7 for A–D, band 4 for A′ and C′—to simulate images of dimension three did not change the relative positions of the endmembers to any great degree. A comparison of Figures 3.5(a) and 3.5(b) clearly reveals that neighbouring pixels are correlated. If two neighbouring pixels p and q had been independent, then the environmental error component they have in common would be zero, and the distribution of the statistical error components $\mathbf{e}_{i,2,p}$ and $\mathbf{e}_{i,2,q}$ would be equal to that of $\mathbf{e}_i$. As a result, the variance-covariance of the subtraction sum $\mathbf{e}_{i,2,p} - \mathbf{e}_{i,2,q}$ calculated for the environmental approach would be twice as large as that of the noise vector $\mathbf{e}_i$ used by both the statistical and the basic physical approach, which would translate to ellipses that were $\sqrt{2}$ times as large. Although this effect indeed was observed if pixels p and q were chosen at random, the fact that the ellipses in Figure 3.5(b) are smaller than those in Figure 3.5(a) demonstrates the correlation in case p and q are neighbours. Based on this collection of endmembers, it was possible to evaluate the performance of the different decomposition methods under a variety of conditions.

Looking at the results of the experiments presented in Tables 3.1 and 3.2, we can make several observations. In general, the accuracy of the proportion estimates decreases as the distance between the endmembers gets smaller, which was to be expected because the influence of the statistical deviations becomes more important. Another anticipated result is that the accuracy of the estimates improves as the number of bands increase (compare Table 3.1 to Table 3.2). An explanation for this phenomenon is that the effect of the statistical deviation in one band is compensated for by opposing deviations in the other bands. As a consequence, however, an exact reconstruction of the spectral response of pixel $\mathbf{x}$ becomes more difficult—ε_x is higher—because the deviations can not all be fitted at the same time. The third and most important observation is that there is a clear order in which the methods are best able to estimate the proportions of a mixed pixel. In all experiments, the methods based on the environmental model were the most accurate, followed at some distance by the approaches that assume the statistical model, which on their turn performed slightly better than the methods based on the basic physical model. This order is apparent especially in the more difficult cases, like the three-dimensional A′C′ experiment for instance (see Table 3.2). Methods based on the same model performed approximately equally well, which is explained by the fact that only simple mixtures of two classes having similar variance-covariance matrices were considered. As a result, the average matrix $\mathbf{N}$ was nearly identical to the $\mathbf{N}_i$ (and thus also to $\mathbf{N}(\mathbf{f})$), which led to similar proportions, similar reconstruction errors ε_x, and similar component estimates.

A further comparison of the performance of the two physical models once again demonstrates that neighbouring pixels are correlated. The big difference in values of ε_x between the basic physical and the environmental approach indicates that in the latter model smaller displacements are needed to reconstruct pixel $\mathbf{x}$; apparently, the true spectra of a pixel's components are approximated well by the spectra of the components of neighbouring pixels. This hypothesis is confirmed by the fact that the accuracy measure ε_p on average shows lower values for the environmental model than for the basic physical model, even when the values for the environmental model are calculated using the variance-covariance matrices of the basic physical model (see Tables 3.1 and 3.2, values given in parentheses). Although the accuracy of the estimated components seems to be a bit disappointing in an absolute sense, it should be noted that the estimates are considerably more accurate than samples drawn at random: the expected value of ε_{p_i} for a random set of samples drawn according to the distribution of class i is equal to 1.0.

As can be seen in Table 3.3, the time needed to process one image differed significantly per method. For each model, the analytic solution was calculated much faster than the brute force approximation, which is not surprising since the computational complexity of the analytic methods is polynomial in n and c, while that of the brute force methods is exponential. When the number of spectral bands increased from two to three, the analytic methods hardly needed more processing time. The processing time of the brute force methods based on the two physical models, on the other hand, increased by a factor of about 100, which matches the sampling rate chosen. The brute force approximation of the statistical model did not require that much extra time, since its complexity is exponential in

mix	model	method	$\varepsilon_f(\%)$	ε_x	ε_{p1}	ε_{p2}
	statistical	brute force	5.8	0.34	–	–
	"	analytic	6.0	0.35	–	–
AB	physical	brute force	5.9	0.27	0.78	0.74
	"	analytic	6.0	0.29	0.85	0.77
	environmental	brute force	5.5	0.06 (0.07)	0.71 (0.73)	0.84 (0.61)
	"	analytic	5.4	0.05 (0.06)	0.75 (0.75)	0.82 (0.61)
	statistical	brute force	13.8	0.32	–	–
	"	analytic	13.8	0.32	–	–
AC	physical	brute force	14.7	0.23	0.77	0.79
	"	analytic	14.5	0.24	0.78	0.77
	environmental	brute force	9.3	0.07 (0.08)	0.63 (0.66)	0.75 (0.44)
	"	analytic	9.3	0.08 (0.08)	0.65 (0.69)	0.77 (0.44)
	statistical	brute force	10.4	0.36	–	–
	"	analytic	10.4	0.36	–	–
BC	physical	brute force	10.7	0.27	0.64	0.78
	"	analytic	10.6	0.27	0.66	0.77
	environmental	brute force	8.4	0.05 (0.03)	0.75 (0.56)	0.86 (0.50)
	"	analytic	8.4	0.05 (0.03)	0.75 (0.56)	0.85 (0.49)
	statistical	brute force	19.3	0.31	–	–
	"	analytic	19.4	0.30	–	–
BD	physical	brute force	20.5	0.20	0.83	0.96
	"	analytic	20.5	0.21	0.83	0.98
	environmental	brute force	13.3	0.05 (0.03)	0.62 (0.46)	0.68 (0.37)
	"	analytic	13.2	0.05 (0.03)	0.62 (0.46)	0.71 (0.38)
	statistical	brute force	23.4	0.33	–	–
	"	analytic	23.4	0.33	–	–
CD	physical	brute force	25.9	0.21	0.84	0.89
	"	analytic	26.2	0.21	0.85	0.91
	environmental	brute force	13.8	0.05 (0.03)	0.78 (0.44)	0.71 (0.36)
	"	analytic	13.9	0.05 (0.03)	0.81 (0.46)	0.73 (0.37)
	statistical	brute force	39.4	0.49	–	–
	"	analytic	39.8	0.49	–	–
A′C′	physical	brute force	45.0	0.25	0.70	0.86
	"	analytic	45.6	0.25	0.73	0.86
	environmental	brute force	27.9	0.06 (0.08)	0.67 (0.77)	0.73 (0.75)
	"	analytic	28.2	0.06 (0.08)	0.66 (0.76)	0.73 (0.75)

Table 3.1: *Performance measured for two band data. Step sizes were 0.01 (statistical) and 0.01π (other). Values in parentheses are calculated using the $\mathbf{N}_i$ of the physical model.*

mix	model	method	ε_f(%)	ε_x	ε_{p1}	ε_{p2}
	statistical	brute force	5.7	0.46	–	–
	”	analytic	5.8	0.46	–	–
AB	physical	brute force	5.7	0.36	0.71	0.66
	”	analytic	5.8	0.37	0.75	0.67
	environmental	brute force	5.0	0.12 (0.16)	0.66 (0.84)	0.71 (0.65)
	”	analytic	5.0	0.11 (0.14)	0.68 (0.86)	0.69 (0.63)
	statistical	brute force	11.0	0.45	–	–
	”	analytic	10.9	0.46	–	–
AC	physical	brute force	12.3	0.34	0.69	0.69
	”	analytic	12.0	0.35	0.70	0.66
	environmental	brute force	9.2	0.12 (0.14)	0.63 (0.84)	0.71 (0.52)
	”	analytic	9.4	0.13 (0.14)	0.64 (0.87)	0.75 (0.54)
	statistical	brute force	10.3	0.43	–	–
	”	analytic	10.3	0.43	–	–
BC	physical	brute force	10.8	0.32	0.61	0.70
	”	analytic	10.8	0.32	0.62	0.70
	environmental	brute force	8.2	0.11 (0.11)	0.64 (0.57)	0.74 (0.52)
	”	analytic	8.0	0.10 (0.10)	0.62 (0.56)	0.73 (0.51)
	statistical	brute force	19.3	0.43	–	–
	”	analytic	19.6	0.43	–	–
BD	physical	brute force	23.9	0.27	0.80	0.89
	”	analytic	24.0	0.30	0.85	0.92
	environmental	brute force	12.7	0.10 (0.08)	0.64 (0.61)	0.65 (0.46)
	”	analytic	12.9	0.10 (0.09)	0.63 (0.60)	0.68 (0.48)
	statistical	brute force	14.9	0.47	–	–
	”	analytic	15.0	0.47	–	–
CD	physical	brute force	17.0	0.33	0.75	0.69
	”	analytic	18.1	0.36	0.77	0.77
	environmental	brute force	11.5	0.10 (0.08)	0.71 (0.50)	0.61 (0.41)
	”	analytic	11.7	0.10 (0.09)	0.73 (0.50)	0.60 (0.41)
	statistical	brute force	36.4	0.53	–	–
	”	analytic	39.5	0.51	–	–
A′C′	physical	brute force	42.5	0.25	0.57	0.89
	”	analytic	47.0	0.28	0.70	0.92
	environmental	brute force	23.7	0.10 (0.12)	0.50 (0.58)	0.73 (0.73)
	”	analytic	23.7	0.10 (0.12)	0.51 (0.60)	0.73 (0.73)

Table 3.2: *Performance measured for three band data. Step sizes were 0.01 (statistical) and 0.01π (other). Values in parentheses are calculated using the* $\mathbf{N}_i$ *of the physical model.*

	statistical brute force	statistical analytic	physical brute force	physical analytic	environmental brute force	environmental analytic
$n = 2$	40.9	1.8	97.3	3.4	771.3	18.7
$n = 3$	42.0	1.8	9386.6	3.6	76523.0	19.8

Table 3.3: *Average CPU-time (in seconds) needed to process one image made up of two classes. The methods were implemented using MATLAB 5.0 and were executed on a SUN SPARCstation-20 running Solaris 2.5.*

c only. Compared to the basic physical model, the brute force solution of the environmental model costs approximately eight times more; the analytic variants differ about factor of only 5.5 due to the relatively greater impact of the evaluation routine which requires a constant amount of time. The computational costs of the brute force methods are the reason that the experiments were limited to at most three dimensions and two classes. The latter restriction, however, also was more convenient for the endmember selection mechanism applied by the environmental approaches.

3.4 Summary and conclusions

In this chapter a new linear mixture model was introduced, which attributes the variations in the spectral response of similarly composed pixels to physical factors instead of random, statistical noise. In this physical model, or better, the more elaborate environmental variant, the correlations that exist between the endmember spectra of neighbouring pixels can be described and exploited to acquire a more accurate estimation of the proportions of a mixed pixel. Solving these physical models can be done using brute force or mathematical analysis but is rather complicated, because it requires estimating the spectral response of each component of the mixed pixel. However, this additional information may be a useful feature for some applications. Based on the characteristics of the component spectra, the proportion of the pixel area covered by each class can be calculated.

To determine the performance of several methods based on the three mixture models (statistical, physical, and environmental), a number of experiments were carried out. In these experiments, artificial images of agricultural scenes were generated using real Landsat-TM data and processed in order to make a quantitative comparison of the accuracy of the decomposition methods possible. It appeared that, independent of the approach followed, the proportions were estimated more accurately as the dispersion of the endmembers got larger; increasing the number of bands also led to a higher accuracy. However, the choice of the mixture model had a considerable influence on the accuracy as well. Methods based on the environmental model, which is a variant of the basic physical model that utilises the fact that neighbouring pixels are correlated, gave the most accurate estimates. The results of the decomposition methods based on the classic statistical model were less good, while those following the basic physical approach, which ignored the pixel correlations altogether, had an accuracy that was even slightly lower. Based on these results, it can be concluded that the physical linear mixture model is better suited to decompose mixed pixels than the classic statistical model, but only if the correlations between neighbouring pixels can be exploited.

In addition to the accuracy of the proportion estimates, the accuracy of the estimated component spectra and the required processing time may be of importance. The components of a mixed pixel could be determined more accurately based on the environmental model than on the basic physical model; the methods based on the statistical linear mixture model did not provide an estimate of the components at all. However, even the components estimated by the environmental approaches did not seem to be very accurate, although this can only be decided in relation to the purpose they are used for. The processing speed of the decomposition methods depended primarily on the method of solution—brute force or mathematical analysis. Whereas the analytic solutions of all models could be calculated very fast due to their polynomial complexity, the methods utilising brute force had an exponential complexity and were much slower. If the methods are ordered according to their underlying models, those based on the statistical model were the fastest, followed by those based on the basic physical model, and those based on the environmental model. Therefore, it can be concluded that the analytic solution of the statistical linear mixture model should be selected if processing speed is of the utmost importance. In case the accuracy of the estimated proportions is the most important, or if the spectral response of the components of a mixed pixel must be extracted, solving the environmental variant of physical linear mixture model is the best choice.

Chapter 4

Linear mixture modelling: tying up the ends*

In this chapter, two loose ends concerning many decomposition methods that are based on the linear mixture model are tied up. Section 4.1 describes how satisfaction of the positivity constraint in particular can be enforced, which in most cases is done *after* a solution of the linear mixture model has been found. In addition, Section 4.2 discusses several methods to determine the endmembers of an image, which must be done *before* the linear mixture model can be solved. Section 4.3, finally, summarises the results and presents the conclusions.

4.1 Satisfaction of the positivity constraint

As described in Section 2.1.2, a solution of the linear mixture model has to satisfy two constraints in order to be physically sound. The sum-to-one constraint, which says that the proportions of a pixel's components must add up to one, can usually be satisfied relatively easily using some sort of scaling operation. The positivity constraint, on the other hand, is much harder to satisfy, because the requirement that each proportion is greater than (or equal to) zero is an inequality that cannot be incorporated in an analytically derived closed solution. This problem can be circumvented of course by using an alternative method of solution such as the brute force approach explained in Section 2.1.2; Settle and Drake [99] also mention Wolfe's quadratic programming procedure [113] and the logit variables approach of Kent and Mardia [59]. However, the computational complexity of these methods may be too high to use them for hyperspectral data. Therefore, most studies simply apply the partially constrained, analytic estimator given by Equation (2.11) and satisfy the positivity constraint in a separate post-processing step like the one described in Section 4.1.1. An alternative to this approach is the method presented in Section 4.1.2, which iteratively adapts the result of an unconstrained least-squares estimator to approximate both the sum-to-one and the positivity constraint without disturbing the original solution to any great degree. To determine the viability of this new approach, we made a small comparison of the various methods, which is described in Section 4.1.3.

*This chapter is based on publication [95].

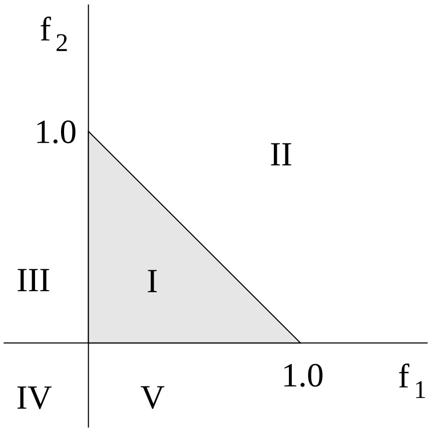


Figure 4.1: *Partitioning of the solution space by the constrained least-squares method. Estimates (f_1,f_2) that lie outside the shaded region need to be recalculated.*

4.1.1 Standard post-processing approaches

One of the first strategies to satisfy the positivity constraint by means of a simple post-processing step was described by Marsh *et al.* [68]. Dealing with mixtures of only two classes, their approximate maximum likelihood technique—see Section 2.2.3—maps a proportion f_1 to 0 if the estimate is negative, while f_1 is taken to be 1 if the estimate was greater than one. Since the proportion of the other class is calculated as f_2=1−f_1, satisfaction of both the sum-to-one constraint and the positivity constraint are guaranteed.

Shimabukuro and Smith [100] suggested a method that can be regarded as an extension of the above strategy to the three class problem. Their constrained least-squares (CLS) method shares some of the characteristics of quadratic programming as it tries to minimise an object function whose parameter values must lie within certain bounds. The basis of the approach of Shimabukuro and Smith is the least-squares estimator (see Section 4.1.2 for more details), which is a simplified version of the well-known linear mixture modelling approach. The expression to be minimised is a function of only two fraction variables, as the sum-to-one constraint is enforced explicitly by taking f_3=1−f_1−f_2. Based on the outcome of the estimator, one or more fractions need to be recalculated in order to satisfy the positivity constraint. Only if both f_1 and f_2 are non-negative and their sum does not exceed 1, the decomposition is accepted. In case (f_1,f_2) indicates a point in one of the other four parts of the solution space depicted by Figure 4.1, several adjustments must be made, which are listed in Table 4.1. For points lying in the part II, the substitution

f_1	f_2	part	recalculate	f_3
positive	positive	I		$1 - f_1 - f_2$
positive	positive	II	f_1, f_2=1−f_1	0
negative	positive	III	f_2 (f_1=0)	$1 - f_2$
negative	negative	IV	(f_1=f_2=0)	1
positive	negative	V	f_1 (f_2=0)	$1 - f_1$

Table 4.1: *Recalculation scheme for the constrained least-squares method.*

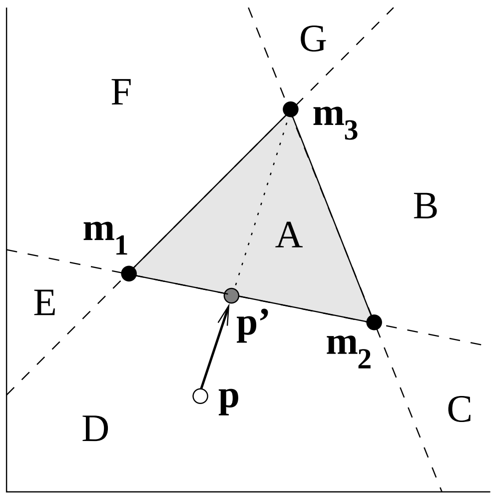


Figure 4.2: *Partitioning of the feature space by the renormalisation method. The fractions estimates corresponding to points outside the shaded region need to be adjusted. Points in part D like* $\mathbf{p}$ *have* $f_3<0$*; setting* $f_3=0$ *and renormalising* (f_1,f_2) *means moving* $\mathbf{p}$ *to* $\mathbf{p'}$*.*

$f_2=1-f_1$ is made prior to recalculating f_1. After recalculation, at least one fraction is 0; if the remaining two fractions still do not satisfy the positivity constraint, the negative fraction is forced to 0 and the other fraction is taken to be 1.

Another post-processing method that can be considered as a generalisation of Marsh's two class strategy for mixtures of arbitrarily many classes was applied in a number of studies (e.g. Settle and Drake [99], Pech *et al.* [79]). The technique is very straightforward as any negative proportions are simply set to zero and the remaining fractions are normalised so that they add up to one. Although this approach may give less accurate results, it does not require the case distinction of the constrained least-squares method, which becomes complicated if the number of classes increases. Compared to the brute force approximation method discussed in Section 2.1.2, the reduction in computational costs is very large, which may make the loss of accuracy acceptable. The difference between the renormalisation method and CLS can well be explained using the feature space representation of a three class problem—see Figure 4.2. According to the theory of convex geometry, all mixtures satisfying the sum-to-one constraint are located on the plane containing the endmembers $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_3$. Similarly, for mixtures on the line through $\mathbf{m}_i\mathbf{m}_j$ holds that $f_i+f_j=1$, in other words, the proportion of the remaining endmember k is zero. This line divides the plane in two parts; the part where $\mathbf{m}_k$ is situated contains all mixtures for which $f_k>0$, while at the other side of the line the mixtures for which $f_k<0$ can be found. As a consequence, all mixtures that satisfy both the sum-to-one constraint and the positivity constraint lie in (or on) the triangle made up of the three endmembers. As described in Section 2.1.2, the brute force method systematically samples this triangle to find the point resulting in the smallest value of the error function. The analytic approach, on the other hand, finds a point $\mathbf{p}$ on the plane—the sum-to-one constraint is satisfied—but not necessarily inside the triangle. Satisfaction of the positivity constraint using the renormalisation approach entails the following adjustments: if one fraction f_k is negative, the point $\mathbf{p}$ is moved towards $\mathbf{m}_k$ until the line $\mathbf{m}_i\mathbf{m}_j$ is reached; if two fractions f_i and f_j are negative, then the point is

outcome estimator			*renormalisation approach*		*constrained least-squares*	
f_1	f_2	f_3	part	renormalise	part	recalculate
positive	positive	positive	A		I	
negative	positive	positive	B	f_2, f_3 (f_1=0)	III	f_2, f_3=1−f_2 (f_1=0)
negative	positive	negative	C	(f_2=1, f_1=f_3=0)	III	f_2, f_3=1−f_2 (f_1=0)
positive	positive	negative	D	f_1, f_2 (f_3=0)	II	f_1, f_2=1−f_1 (f_3=0)
positive	negative	negative	E	(f_1=1, f_2=f_3=0)	V	f_1, f_3=1−f_1 (f_2=0)
positive	negative	positive	F	f_1, f_3 (f_2=0)	V	f_1, f_3=1−f_1 (f_2=0)
negative	negative	positive	G	(f_3=1, f_1=f_2=0)	IV	(f_3=1, f_1=f_2=0)

Table 4.2: *Adjustment rules followed by the renormalisation approach and the constrained least-squares method based on the partitioning given in Figure 4.2.*

moved directly to $\mathbf{m}_k$. Translation of the rules used by CLS to the example of Figure 4.2 results in Table 4.2. This table shows that both post-processing methods actually are very similar: in all cases where the renormalisation approach moves $\mathbf{p}$ in a straight line until $\mathbf{m}_i\mathbf{m}_j$ is reached, CLS calculates which location of $\mathbf{p}$ on $\mathbf{m}_i\mathbf{m}_j$ results in the smallest error. Furthermore, mixtures in part G—and part A of course—are treated exactly the same. Those in parts C and E, however, may be processed differently depending on the location of the endmembers. Figure 4.3 gives an example of the asymmetric behaviour of CLS, which is due to the exceptional handling of endmember $\mathbf{m}_3$. Figure 4.3(a) shows the situation where $\mathbf{p}$, after recalculation of f_1 and f_3 followed by restriction of the values to [0,1], is moved to $\mathbf{m}_1$, which corresponds to the $(1,0,0)^T$ proportions vector found by the renormalisation approach. However, if endmembers $\mathbf{m}_2$ and $\mathbf{m}_3$ are interchanged as shown in Figure 4.3(b), CLS moves $\mathbf{p}$ to a position between $\mathbf{m}_1$ and $\mathbf{m}_3$, which corresponds to a proportions vector of $(f_1,0,1-f_1)^T$ with $0<f_1<1$. Although the latter solution is better as it results in a lower error—the (Euclidean) distance between $\mathbf{p}$ and $\mathbf{p'}$ is smaller than the distance between $\mathbf{p}$ and $\mathbf{m}_1$—the adjustments are different from those made for mixed pixels in G, which are always fully allocated to class 3.

4.1.2 A new method: iterative least-squares

In the last decade, hyperspectral remote sensing has assumed an important place in earth observation research. Hyperspectral instruments collect data in many narrowly defined channels, e.g. AVIRIS takes measurements in 224 spectral bands of 10 nm width each (see Table 1.2). Because of the high data volume that results, special attention has to be paid to the efficiency of the decomposition methods, especially if the positivity constraint is to be satisfied. Solving the linear mixture model with brute force, for instance, is not an option because of the high computational complexity. The CLS method suggested by Shimabukuro and Smith is much faster but rather complicated if more than three classes are involved. Since one advantage of a high number of bands is that in theory mixed pixels can be decomposed in many more classes than the usual number of three to five, which is the maximum for Landsat-TM data [70], this is a serious limitation. Least-squares in combination with the standard renormalisation approach, on the other hand, is at least as fast and conceptually much simpler, but it usually leads to a sub-optimal

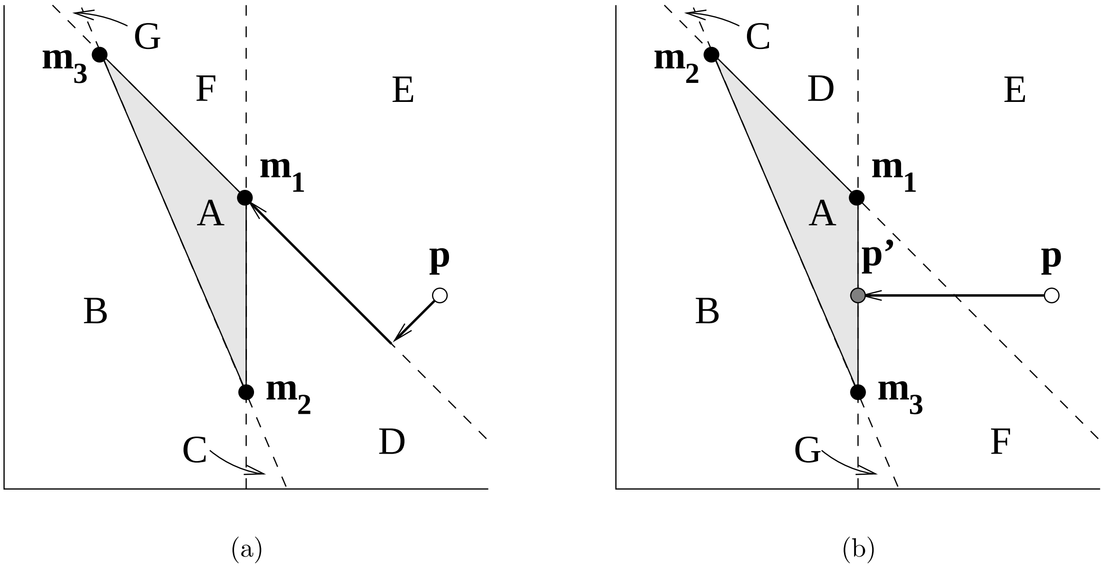


Figure 4.3: *An example of the asymmetry in the constrained least-squares method. (a) Recalculation of f_1 and f_3 moves $\mathbf{p}$ to a position on the line through $\mathbf{m}_1$ and $\mathbf{m}_3$; since $f_3<0$, $\mathbf{p}$ is moved further to $\mathbf{m}_1$. (b) After interchanging $\mathbf{m}_2$ and $\mathbf{m}_3$, recalculation of f_1 and f_3 moves $\mathbf{p}$ to a different position somewhere between $\mathbf{m}_1$ and $\mathbf{m}_3$; equivalent points in part G, however, are always mapped to $\mathbf{m}_3$.*

estimate. Therefore, we developed the iterative least-squares (ILS) method as an accurate, conceptually simple, and relatively fast way to decompose hyperspectral mixed pixels [95].

Like all least-squares methods mentioned before, ILS is based on the linear mixture model given by Equation (2.5). In contrast to classic linear mixture modelling, however, the within-class variations represented by the variance-covariance matrices $\mathbf{N}_i$ are not explicitly accounted for. The standard least-squares approach towards solving this slightly different model is to select the solution that has the smallest square error term, i.e. minimises $\mathbf{e}^T\mathbf{e}$. By setting the partial derivatives of this expression with respect to each f_j to zero, the following estimator can be derived:

$$\mathbf{f}_{ls} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{x}. \tag{4.1}$$

Note that this solution is identical to the one given by Equation (2.10) if the variance-covariance matrix $\mathbf{N}$ is replaced by the nxn identity matrix $\mathbf{I}$. In other words, the least-squares estimator is just a simplification of the classic linear mixture modelling approach for the case that all spectral bands have equal weight and the error term $\mathbf{e}$ is uncorrelated.

In general, the solution given by Equation (4.1) satisfies neither the sum-to-one constraint nor the positivity constraint. In order to comply better with these rules, the function to be minimised is extended with two additional terms:

$$\alpha_0\mathbf{e}^T\mathbf{e} + \alpha_1 g_1(\mathbf{f}) + \alpha_2 g_2(\mathbf{f}), \quad \text{where} \tag{4.2}$$

$$g_1(\mathbf{f}) = \left(1 - \sum_{j=1}^{c} f_j\right)^2 \quad \text{and} \tag{4.3}$$

$$g_2(\mathbf{f}) = \sum_{j=1}^{c} \begin{cases} f_j^2, & f_j < 0 \\ 0, & f_j \geq 0. \end{cases} \tag{4.4}$$

Note that $g_1(\mathbf{f})$ and $g_2(\mathbf{f})$ penalise violations of the sum-to-one and positivity constraint, respectively, and that they are zero in case both constraints are satisfied. Starting from the unconstrained least-squares solution, the iterative least-squares method minimises this cost function by means of the gradient descent approach:

$$f_j^{(k+1)} = f_j^{(k)} - \alpha_0 \frac{\partial \mathbf{e}^T\mathbf{e}}{\partial f_j} - \alpha_1 \frac{\partial g_1(\mathbf{f})}{\partial f_j} - \alpha_2 \frac{\partial g_2(\mathbf{f})}{\partial f_j}. \tag{4.5}$$

The coefficients α_i, which must lie between 0 and 1, determine the weights of the various error terms; by choosing relatively high values for α_1 and α_2, satisfaction of the constraints is emphasised. The iterative process is stopped as soon as the fractions after iteration $k+1$ are approximately equal to those after the k-th iteration: $\sum_{j=1}^{c}(f_j^{(k+1)} - f_j^{(k)})^2 \leq t^2$; by varying the threshold t, the accuracy of the result can be traded of against computational costs. The partial differentials of the different contributions to the overall error are

$$\frac{\partial \mathbf{e}^T\mathbf{e}}{\partial f_j} = \frac{\partial \sum_{i=1}^{n} e_i^2}{\partial f_j} = \sum_{i=1}^{n} 2e_i \frac{\partial e_i}{\partial f_j} \stackrel{(*)}{=} -2 \sum_{i=1}^{n} e_i m_{ij}, \tag{4.6}$$

$$\frac{\partial g_1(\mathbf{f})}{\partial f_j} = 2 \sum_{j=1}^{c} f_j - 2, \quad \text{and} \tag{4.7}$$

$$\frac{\partial g_2(\mathbf{f})}{\partial f_j} = \begin{cases} 2f_j, & f_j < 0 \\ 0, & f_j \geq 0. \end{cases} \tag{4.8}$$

In order to derive these expressions, application of the chain rule was required in several cases. The step marked with $(*)$ was based on the relation $e_i = x_i - \sum_{j=1}^{c} m_{ij} f_j$, which follows directly from the linear mixture model. Substitution of Equations (4.6)–(4.8) in Equation (4.5) defines a way how the aforementioned cost function can be minimised.

The exact procedure followed by the ILS method is a little bit different for reasons of efficiency. As indicated by Equation (4.6), computation of the partial derivative of $\mathbf{e}^T\mathbf{e}$ requires more than n multiplications, which is very time consuming as the number of bands n can be as large as several hundreds. However, this expression can be transformed into a more efficient formula based on the equality $\mathbf{M}^T\mathbf{M}\mathbf{f}_{ls} = \mathbf{M}^T\mathbf{x}$ derived from Equation (4.1). Using matrix-vector notation, we can rewrite Equation (4.6) as

$$\begin{aligned}
\frac{\partial \mathbf{e}^T\mathbf{e}}{\partial \mathbf{f}} &= -2\mathbf{M}^T\mathbf{e} && \{\mathbf{e} = \mathbf{x} - \mathbf{M}\mathbf{f}\} \\
&= -2\mathbf{M}^T(\mathbf{x} - \mathbf{M}\mathbf{f}) && \{\text{unfold, rearrange}\} \\
&= 2\mathbf{M}^T\mathbf{M}\mathbf{f} - 2\mathbf{M}^T\mathbf{x} && \{\mathbf{M}^T\mathbf{x} = \mathbf{M}^T\mathbf{M}\mathbf{f}_{ls}, \text{fold}\} \\
&= 2\mathbf{M}^T\mathbf{M}(\mathbf{f} - \mathbf{f}_{ls}),
\end{aligned} \tag{4.9}$$

which is equivalent to

$$\frac{\partial \mathbf{e}^T\mathbf{e}}{\partial f_j} = \sum_{k=1}^{c} u_{jk}(f_k - f_{ls_k}), \quad \text{where} \quad u_{jk} = 2\sum_{i=1}^{n} m_{ij}m_{ik}. \tag{4.10}$$

Since u_{jk}—the element at row j column k of matrix $\mathbf{U}$=$\mathbf{M}^T\mathbf{M}$—is independent of the current iteration, it needs to be calculated only once and can subsequently be determined quickly by table look-up. Thus, the complexity of a single iteration has become proportional to c, the number of endmembers, which in general is much smaller than the number of spectral bands. Therefore, provided that the number of iterations needed is small, ILS is a fast alternative for the decomposition of mixed pixels in hyperspectral images in cases where satisfaction of the positivity constraint is desired.

4.1.3 A little comparative experiment based on MAIS data

From the previous section it has become clear that the concept of ILS is simple and easily extendible to an arbitrary number of classes. Furthermore, we have shown that the method has the potential to be fast as its complexity depends on the number of classes instead of the number of spectral bands. However, it must still be checked that the number of iterations required is not too large for ILS to be applicable in practice. Besides that, the accuracy of the fractions vector estimated by ILS needs to be determined as well. Therefore, we carried out a little experiment with hyperspectral data in which a comparison between ILS and two other methods based on the least-squares approach was made. The remainder of this section subsequently covers the set-up, results, and discussion of this experiment.

Set-up

Like the test in Chapter 3, this experiment consisted of three steps, two of which are described in more detail here. The first part explains how the artificial data used in this experiment was generated based on the hyperspectral image available. The second step included decomposition of the samples in this data set using the basic least-squares approach with and without renormalisation step, the CLS method, and ILS with different threshold settings. In the third and last step of the experiment, the performance of these methods was evaluated quantitatively using four error measures; a description of these error measures can be found at the end of this section.

Since we had no hyperspectral mixed pixels with supervised information at subpixel level at our disposal, we had to generate a set of artificial samples. The basis of the simulation was a 30 band image acquired with the Modular Airborne Imaging Spectrometer (MAIS) made in China; the bands had a spectral resolution of 20 nm each and ranged from the visible to the infrared part of the spectrum. Each of the $475x512$ pixels was recorded in 8 bits per band and covered an area of $20x20$ m^2. The image, which is shown in Figure 4.4, displays part of the Yellow River Valley in the Ganshu province in China and has water, vegetation, and soil as its main components. For each class, a remote sensing expert who is familiar with the region picked out one pure pixel, which formed our only source of supervised information. Based on these pixels, the entire image was classified using the well-known nearest neighbour algorithm, where pixels that deviated too much from the

Figure 4.4: *Band six of the MAIS image displaying part of the Yellow River Valley.*

supervised examples were allocated to a fourth class called unknown. Next, blocks of $5x5$ pixels belonging to the same class were extracted from the classified image as described in [92]. This way some 16 blocks per class were selected, whose mean vector and variance-covariance matrix—multiplied by four to compensate for the loss of variability due to the selection method—served as the distribution of the corresponding class. To generate the i-th sample of the artificial data set, first its fractions were computed as

$$f_{i\mathrm{mod}3} = r_1, \quad f_{(i+1)\mathrm{mod}3} = r_2(1 - f_{i\mathrm{mod}3}), \quad f_{(i+2)\mathrm{mod}3} = 1 - f_{i\mathrm{mod}3} - f_{(i+1)\mathrm{mod}3},$$

where r_1 and r_2 denote random fractions drawn from a distribution uniform over $[0, 1]$. Next, for each of the three classes a random sample was drawn from the multivariate normal distributions determined before, which were combined with the fractions vector according to the standard mixing equations of the linear mixture model. Finally, a random

Gaussian with μ=0.0 and σ=2.0 was added to each band to model sensor noise, and the resulting intensities were rounded to the nearest integer and truncated to the interval $[0, 255]$. Thus, a set of 16384 mixed pixels was generated, which was used to evaluate the different decomposition methods.

To quantify the performance of each of the decomposition methods, the following four error measures were used:

$$\varepsilon_f = \left\langle \frac{1}{2} \sum_{j=1}^{c} \left| \hat{f}_j - f_j \right| \right\rangle \tag{4.11}$$

$$\varepsilon_x = \sqrt{\left\langle \frac{\mathbf{e}^T \mathbf{e}}{n 256^2} \right\rangle} \tag{4.12}$$

$$\varepsilon_{sum} = \sqrt{\left\langle g_1(\hat{\mathbf{f}}) \right\rangle} \tag{4.13}$$

$$\varepsilon_{pos} = \sqrt{\left\langle g_2(\hat{\mathbf{f}})/c \right\rangle} \tag{4.14}$$

With the exception of ε_f, which was already presented in Section 3.3.1 as a means to assess the accuracy of the estimates, all measures are derived directly from the cost function minimised by the ILS method. Equation (4.12) defines ε_x as the root mean square (RMS) of the error vectors needed to fit the linear mixture model to the pixels in the data set. Note that, apart from the root, ε_x is identical to the definition given by Equation (3.23) if the variance-covariance matrix $\mathbf{N}(\mathbf{f})$ is taken to be $256^2 * \mathbf{I}$. The measures ε_{sum} and ε_{pos} are simple representations of the average violation of the sum-to-one and the positivity constraint, respectively, where the latter measure includes a term $1/c$ to reflect the error per component rather than per pixel. With these four error measures the most important aspects of the performance of ILS and the other decomposition methods can be determined.

Results

The results of the four methods applied to the artificial data set are shown in Table 4.3. For the iterative least-squares method three different threshold settings were used, viz. 0.01, 0.001, and 0.0005. The values of the α were taken to be approximately equal to those determined experimentally by Schouten and Klein Gebbinck [95][1], being α_0=0.20/(n∗256^2), α_1=0.15, and α_2=0.25. As Table 4.3 confirms, the basic least-squares approach naturally is the fastest decomposition method, providing the best fit (lowest ε_x) of the unconstrained model to the data. However, the relatively high values of ε_{sum} and ε_{pos} indicate that the sum-to-one constraint and the positivity constraint are violated rather seriously. Post-processing these estimates with the renormalisation method takes a little extra time and leads to a higher ε_x, but satisfaction of the constraints clearly improves the accuracy of the estimates as ε_f decreases from 9.6% to 8.2%. The extra time needed by CLS approach is about a factor three higher, but this pays off in terms of a better performance: ε_f is reduced to 7.8% and ε_x decreases from 0.036 to 0.031. Compared to CLS, ILS has modestly higher to considerably higher computational costs, depending on the value of the threshold used.

[1]Due to a typo, [95] lists the value of α_0 as half the value actually used.

method	ε_f (%)	ε_x	ε_{sum} (%)	ε_{pos} (%)	iterations	time (s)
least-squares	9.6	0.022	4.7	3.0	–	87.6
renormalise	8.2	0.036	0.0	0.0	–	94.1
CLS	7.8	0.031	0.0	0.0	–	108.8
ILS (0.01)	8.2	0.029	1.4	0.4	2.2	143.6
ILS (0.001)	7.7	0.029	1.2	0.1	8.3	289.9
ILS (0.0005)	7.6	0.028	1.2	0.1	22.5	639.2

Table 4.3: *Performance of the four methods—the figure in parentheses is the threshold setting used by ILS—on the artificial MAIS data. The methods were implemented using MATLAB 5.0 and were executed on a SUN SPARCstation-20 running Solaris 2.5.*

Furthermore, the sum-to-one and positivity constraints are no longer satisfied perfectly but only approximately, albeit that the violation of the constraints is much less severe than it was for the basic least-squares solution without post-processing. On the other hand, the estimates of ILS result in an ε_x that is closer to the optimal value of 0.022 than all other post-processing methods. More important, though, is that the accuracy of the estimates by ILS are surprisingly good as well, ranging from as good as the renormalisation approach (ε_f=8.2% for t=0.01) to slightly better than CLS (ε_f=7.6% for t=0.0005).

In Table 4.3 it can be observed that decreasing the threshold from 0.001 to 0.0005 resulted in only a marginal improvement of the estimates by ILS. Simultaneously, however, the CPU-time required to post-process the data set increased with a factor 2.7, as did the average number of iterations per sample. Presumably, the extra iterations have little effect beyond a certain minimum number. To test this hypothesis, we adapted our ILS implementation to perform a certain number of iterations for all samples instead of using the threshold stopping criterion described in Section 4.1.2. Figure 4.5 shows the results for the first 20 iterations. As expected, ε_f, ε_{sum}, and ε_{pos} decrease sharply at first, but display a much flatter curve after some extra iterations. The value of ε_x rises considerably in the first iteration, but stabilises at an intermediate level soon thereafter as well. To get a better understanding of the number of iterations performed, we also made histograms of the number of iterations required for both t=0.001 and t=0.0005—see Figure 4.6. A comparison of Figures 4.6(a) and 4.6(b) reveals that the general shape of the histogram is the same for both settings. The modal value is slightly higher for t=0.0005, but much less than the difference in the average number of iterations (8.3 vs. 22.5) would suggest. Therefore, it is concluded that lowering the threshold to 0.0005 had little influence other than increasing the computational costs, since it resulted in refining the estimates of a limited number of samples, which had only little effect on the overall performance.

Discussion

The results presented in the previous section make it clear that the estimates of the basic least-squares method can be improved considerably by enforcing the sum-to-one and the positivity constraint. Since the brute force approach towards minimising $\mathbf{e}^T\mathbf{e}$ under these constraints is much too expensive for hyperspectral data, other methods are needed. The renormalisation approach, which is widely used, is simple and fast but gives sub-optimal

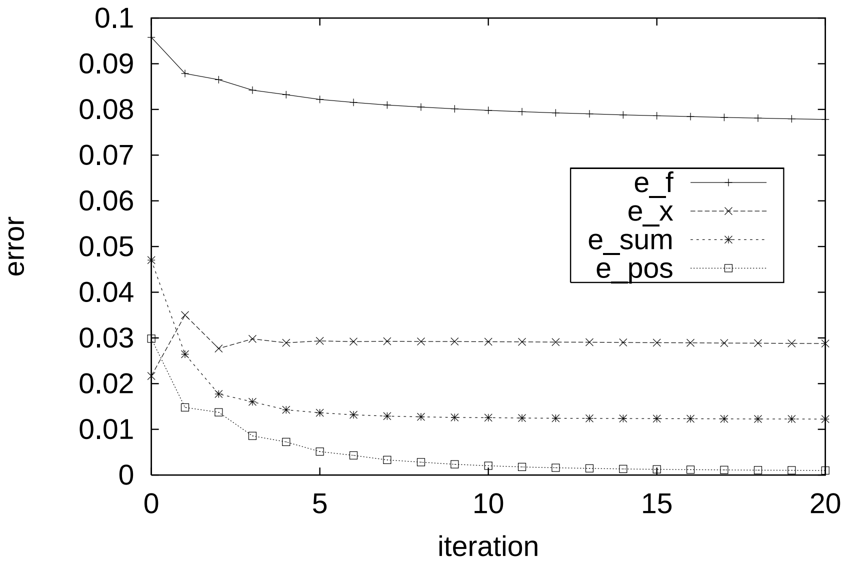
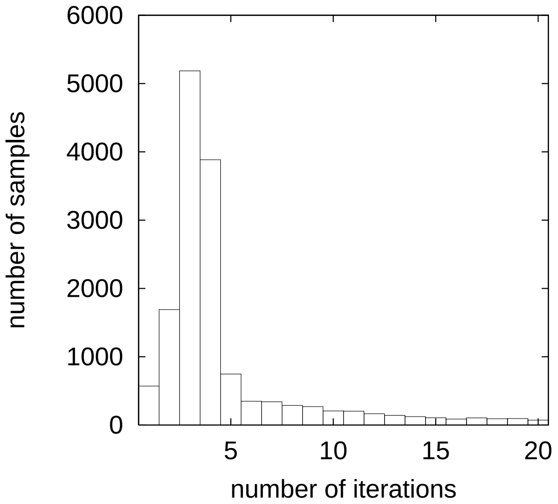
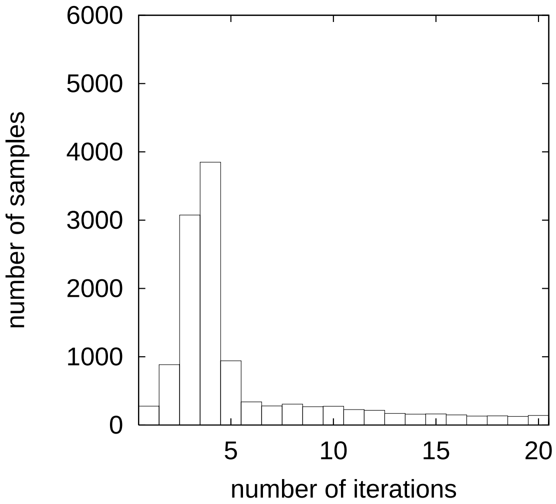


Figure 4.5: *Performance of ILS after executing a number of iterations for all samples.*



(a) Threshold t=0.001.



(b) Threshold t=0.0005.

Figure 4.6: *Histogram of the number of iterations required for several threshold settings.*

estimates. The CLS method is about as fast and more accurate, but the algorithm gets complicated if the number of classes is larger than three. Compared to CLS, ILS requires some extra computations and does not enforce the constraints as rigorously. However, the accuracy that can be achieved by ILS is higher than that of CLS if the threshold is set low enough. This surprising fact is probably due to the sub-optimal behaviour of CLS in case two of the three estimated fractions are negative—see Section 4.1.1. An advantage of ILS is that the accuracy of the estimates can be traded off against computational costs. Furthermore, ILS is conceptually much simpler than CLS, which means that ILS is more suitable to handle mixtures of more than three classes. A possible disadvantage of ILS is that several parameters have to be set, being the weighing factors α_0, α_1, and α_2, and the threshold. In order to determine whether the optimal values of these parameters depend much on the data set or not, further research is needed. Future work could also be directed towards adapting ILS to guarantee stringent satisfaction of the constraints, though this goal can largely be achieved by selecting high values for α_1 and α_2 as well. Anyway, this experiment has shown that ILS potentially is a simple, accurate, and relatively fast method to decompose mixed pixels in hyperspectral images.

4.2 Ways to determine image endmembers

In this section, several methods to determine the endmembers of an image are described. The endmember spectra, which make up the columns of the matrix $\mathbf{M}$ used throughout this thesis, must be known before the decomposition process can take place. Sections 4.2.1 to 4.2.3 discuss three techniques that are usable for applications where the size of the objects in the image can be arbitrarily small. Sections 4.2.4 and 4.2.5 explain two methods that are applicable only if the objects are at least as large as a single pixel or have a size that is even much larger (e.g. an agricultural field). In Section 4.2.6, finally, a new approach is described that is based on K-means clustering; subsequently, this method is evaluated by means of two small experiments.

4.2.1 Spectral libraries

One possibility to determine the image endmembers is to extract them from a spectral library. Such a library, whose use is usually related to geological applications (e.g. Adams *et al.* [2], Smith *et al.* [103]), contains endmember spectra that have been measured either in a laboratory or in the field. For applications in the agricultural domain, however, these libraries are less suitable because they need to account for all processes and factors influencing the data spectra. For example, several spectra corresponding to the same green vegetation type over different backgrounds may have to be included, since multiple scatterings between leaves and a bright soil background increase the near infrared reflectance of the leaves (Bateson and Curtiss [10]). Also, additional spectra may have to be included to cover all stages of a crop's growth or to compensate for the effects of other biological processes such as fluorescence in response to stress. Thus, unless extensive field work is done at the time of image acquisition, access to a very large library is required. A more important problem, though, is that the image and the library need to be calibrated in order to adjust for instrumental drift, atmospheric effects, and illumination intensity. As the necessary corrections may differ for the various parts of the image, which is often the case for images covering a large area (e.g. AVHRR images), this step can be complicated. After the calibration coefficients have been determined, usually the image is converted to match the library. Analogously, however, the reference endmember spectra in the library could be transformed into the endmember spectra of the current image. In recent years, spectral libraries are used primarily to identify the composition of image endmembers (Smith *et al.* [104], Mertes *et al.* [73]). First, the image endmembers themselves are determined using a method such as principal component analysis (see Section 4.2.2). Since the spectra found need not correspond to pure materials in the scene, their identity must subsequently be inferred from reference spectra of known materials. Smith *et al.* [104] express each image endmember as a linear combination of reference endmember spectra, just as mixed pixels are expressed as a linear combinations of image endmembers. In their two-step method, a function incorporating the earlier described calibration and the aforementioned alignment (unmixing) is repeatedly evaluated for different candidate groups of reference spectra until a suitable representation of the image endmembers determined in the first step is found.

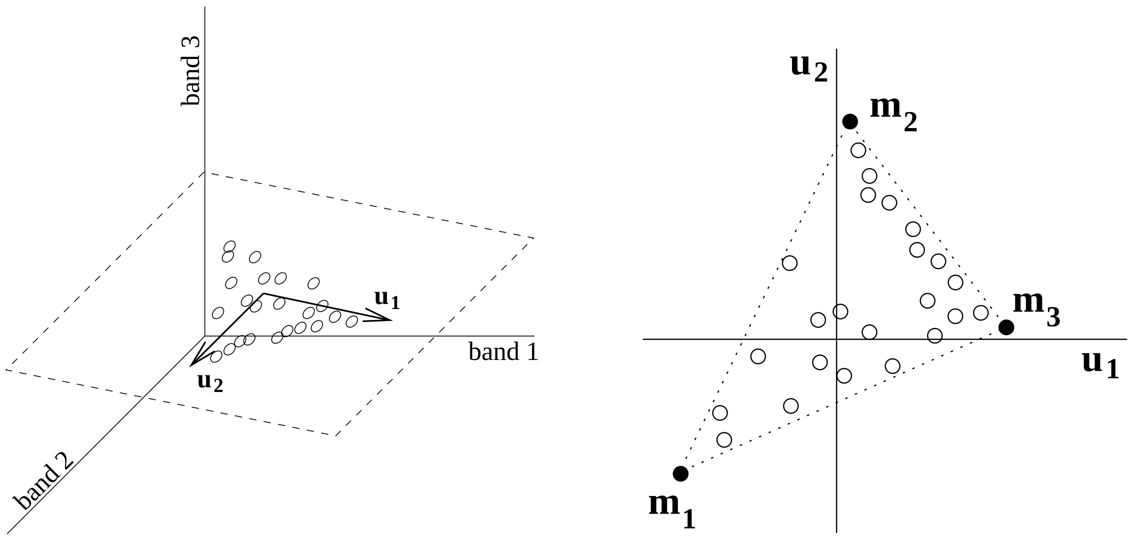
4.2.2 Principal component analysis

Traditionally, the objective of principal component analysis (PCA) is to reduce the dimensionality of a data set while retaining as much of the relevant information as possible. This goal can be achieved by rotating the coordinate system such that most of the variation in the data is found along a limited number of axes, the so-called *principal components*. The axes where the data shows little or no variation are disregarded, which corresponds to restricting the original feature space to a smaller linear subspace. As was described earlier, see Section 3.2.1 for instance, linear mixtures of c ($c<n+1$) classes that satisfy the sum-to-one constraint define a linear subspace as well, being of dimension $c-1$. Therefore, on the condition that mixing accounts for the greater part of the variation in the data, calculating the first $c-1$ principal components is a major step towards determining the c endmember spectra.

Smith *et al.* [103] developed a method based on PCA to determine endmembers from remote sensing data, which was later applied in several other studies, e.g. Cross *et al.* [23] and Bryant [18]. None of these articles explain PCA in any detail; however, a clear description of the mathematical foundations of PCA is given by Bishop [13, pp. 310–313]. In summary, each pixel is represented using a new coordinate system, $\mathbf{x} = \sum_{i=1}^{n} z_i \mathbf{u}_i$, where the $\mathbf{u}_i$ constitute a set of orthonormal basis vectors spanning the original feature space. Next, the dimensionality of the data is reduced by replacing several of the z_i by values that are identical for each pixel—the corresponding $\mathbf{u}_i$ are virtually disregarded—and the sum of square errors between the original pixels and their simplified counterparts is calculated. It can be derived that this error is minimal if the basis vectors satisfy

$$\mathbf{N}\mathbf{u}_i = \lambda_i \mathbf{u}_i, \quad \text{or, equivalently,} \quad \mathbf{U}^{-1}\mathbf{N}\mathbf{U} = \boldsymbol{\lambda}\mathbf{I}, \tag{4.15}$$

where $\mathbf{N}$ denotes the variance-covariance matrix of the data after subtraction of their mean vector. In other words, the eigenvectors of $\mathbf{N}$ define the coordinate system in which most of the variation of the data is found along a minimum number of axes. The eigenvalue λ_i



(a) Finding the principal components. (b) Choosing the endmember spectra.

Figure 4.7: *The PCA approach to determine three endmember spectra. (a) The principal components* $\mathbf{u}_1$ *and* $\mathbf{u}_2$ *are chosen such that the plane they span contains most pixels. (b) Within this plane, the endmembers are chosen such that the triangle they define coincides with the triangular shape of the cloud of pixels.*

represents the variance of the data on the axis $\mathbf{u}_i$ and is commonly regarded as either a primary or a secondary eigenvalue. Whereas the primary eigenvalues are higher in magnitude and account for the variation due to spectral mixing, the sum of secondary eigenvalues should be equivalent to the variances arising from the instrumentation. In many cases, the primary eigenvalues can be clearly separated from the secondary ones, which provides a criterion to test the dimensionality of the data and consequently also the number of endmembers. After discarding the information associated with the secondary eigenvalues, the linear subspace spanned by the endmembers is found.

Once the $c-1$ dimensional subspace has been found, the c endmember spectra must be determined such that the positivity constraint is satisfied. According to the theory of convex geometry, this means that $\mathbf{m}_i$ must be chosen such that the simplex they define contains all elements in the data set. If mixtures of only two classes are considered, then the minimum and maximum values of the first principal component can be taken as the endmember spectra $\mathbf{m}_1$ and $\mathbf{m}_2$. For mixtures of three classes, a scatterplot of the first against the second principal component can be made. As is shown in Figure 4.7, with this scatterplot the smallest triangle—a three-dimensional simplex—containing all pixels in the image can be determined; the vertices of this triangle define the three endmember spectra. When dealing with mixtures of more than three classes, several approaches are possible. Bryant [18] uses a scatterplot of the first two principal components to find that the data has a pentagonal shape. By selecting one pixel near each of the five vertices, five endmember spectra are defined. This approach, however, may fail for example if one of the endmembers is obscured by the other four since its discriminatory features are found along the other three principal components. Bateson and Curtiss [10] use a visualisation technique based on parallel coordinates that permits to extend the set of endmembers with one spectrum at the time. Although this method solves the problem of Bryant's approach, it requires a lot of human effort to judge the acceptability of the endmembers. Furthermore, it is well possible that different users come up with different endmember spectra.

4.2.3 Linear regression of known mixtures

Linear regression is a classic statistical technique that can be used to determine endmember spectra if a set of mixed pixels with known fraction vectors is available. With the growing popularity of AVHRR data, this technique has been applied frequently in the last few years, e.g. by Holben and Shimabukuro [50], Hlavka and Spanner [49], Kerdiles and Grondona [60], and Oleson *et al.* [78]. Due to its coarse spatial resolution, AVHRR data in general contains few if any pure pixels of the classes of interest, which are needed for the extraction of endmembers by the more traditional methods described in Sections 4.2.4 and 4.2.5. However, with the help of relatively high-resolution data such as Landsat-TM imagery, the composition of a number of mixed AVHRR pixels can be estimated, which subsequently can be used to calibrate the linear mixture model for the AVHRR data. This calibration requires that the TM-image is spatially degraded to match the characteristics of the AVHRR scanner, and that it is accurately co-registered with the AVHRR-image. Since both topics are considered to be beyond the scope of this thesis, we refer to Oleson *et al.* [78] for further information.

Multiple linear regression is a generalisation of the simple linear regression method briefly discussed in Section 2.2.2. Just like the difference between linear and non-linear regression is not always clear[2], the terminology simple and multiple regression may be a bit confusing. Simple means that there is only one regressor variable, i.e. f_1 in Equation (2.14), multiple means that there are more than one, for example

$$x_i = \beta_{0,i} + \beta_{1,i} f_1 + \cdots + \beta_{c-1,i} f_{c-1}. \tag{4.16}$$

What may be confusing is that a pixel is a measurement in multiple spectral bands, but this has nothing to do with it. In fact, to determine the endmember spectra, linear regression has to be applied to each of the n bands separately. Another source of confusion is that linear regression tries to find the subspace that best fits a set of calibration pixels. Although the index to this set leads to a more complex notation, this holds for simple as well as multiple regression since both make use of a calibration set. Multiple linear regression in terms of the linear mixture model with sum-to-one constraint is based on the equation

$$\mathbf{y} = \mathbf{F}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad \text{where} \tag{4.17}$$

$$\mathbf{y} = \begin{bmatrix} x_i^{(1)} \\ \vdots \\ x_i^{(p)} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} 1 & f_1^{(1)} & \cdots & f_{c-1}^{(1)} \\ \vdots & \vdots & & \vdots \\ 1 & f_1^{(p)} & \cdots & f_{c-1}^{(p)} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_{0,i} \\ \beta_{1,i} \\ \vdots \\ \beta_{c-1,i} \end{bmatrix} = \begin{bmatrix} m_{ic} \\ m_{i1} - m_{ic} \\ \vdots \\ m_{i(c-1)} - m_{ic} \end{bmatrix},$$

[2]According to Myers [75], a model is linear if it is linear in its parameters β_i, so $y = \beta_0 + \beta_1 x + \beta_2 x^2$ is linear, while $y = \beta_0 + \beta_1 x^{\beta_2}$ is not.

and $\boldsymbol{\varepsilon}$ is a $px1$ vector representing the model error. In this equation, which is specified for band i, the superscript (j) denotes the index into the set of p calibration pixels. For this well-known system, the least-squares solution minimising $\boldsymbol{\varepsilon^T \varepsilon}$ is (c.f. Equation (4.1)):

$$\boldsymbol{\beta} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{y}. \tag{4.18}$$

Once $\boldsymbol{\beta}$ has been calculated for each of the n bands, the endmember spectra $\mathbf{m}_j$ can be compiled from the m_{ij}, which on their turn can be computed by summing the corresponding $\beta_{0,i}$ and $\beta_{j,i}$. For simple linear regression (c=2), the exact same procedure can be used.

4.2.4 Supervised pure pixel data

The use of supervised pure pixels to derive the reflectance typical for the classes of interest has been standard practice in classification applications for many years. Obviously, this approach can be followed only if the size of the objects that are depicted is at least as large as a single pixel. One possibility to determine pure pixels of a particular class is to let a remote sensing expert pick them directly from the image. Although fast and cheap, this method requires that the remote sensing expert is familiar with the terrain and has an up-to-date knowledge regarding the growing stage of the vegetation present. Another possibility that is much more time-consuming, expensive, and probably tedious is to collect them by means of field work, preferably not too long before or after image acquisition. When using a map to determine which locations to visit, one has to choose the number of observations to be made, which sampling pattern to apply, and, given the pattern selected, the spacing of the observations (Campbell [19]). As is known from elementary statistics, the number of observations determines how reliably the characteristics of a class can be estimated. Furthermore, in case of random sampling, many samples have to be taken to make likely that the less frequently occurring classes are represented by a minimum number of observations. The sampling pattern specifies the arrangement of observations. Campbell [19] discusses four different sampling strategies (random, stratified random, systematic, and stratified systematic), which each have their strong and weak points. Related to the sampling pattern is the required spacing of the observations: landscapes with large homogeneous objects can be represented by observations spaced at larger distances than those displaying a finer, more intricate fabric. Once sufficient samples for each class have been taken and the field data have been registered with the image, characteristics such as the class mean and variance-covariance matrix can be calculated. Optionally, the supervised data can be used to find similar pixels probably belonging to the same class in order to refine the class distribution. The pure pixel approach is less appropriate for coarse resolution data such as AVHRR, but it may work if reference images of fine spatial resolution are used. Quarmby *et al.* [82] applied the technique to classify a SPOT-XS image, after which a corresponding AVHRR image was resampled such that pure pixels emerged. Foody and Cox [29] used the same method to classify a Landsat-MSS image, which was then spatially degraded to create pure pixels at AVHRR resolution. From this section and Section 4.2.2, it may be clear that an endmember spectrum is not exactly the same as a class mean vector: whereas an endmember is a single point in the feature space lying at the extreme of the data cloud, a mean vector is a point that represents the average reflectance of some class and is surrounded by pure pixels deviating slightly from the class mean. However, if the within-class variation is small compared to the dispersion of the endmembers, which is a prerequisite for most statistical decomposition methods anyway (see Sections 2.1.2 and 3.2.2), the class means may be a good approximation of the endmember spectra.

4.2.5 Selection of homogeneous areas

Another method to derive class characteristics needed for mixed pixel decomposition is to calculate the mean vector and variance-covariance matrix of selected homogeneous areas. Although many similarities with the method described in Section 4.2.4 exist, an important difference is that this technique can only be applied if the objects in the image are much larger than a single pixel; depending on the resolution, agricultural fields may well qualify as such. Basically, there are two approaches towards selecting training areas directly from the image. The first way, which is followed by Drake and Settle [24] and Gong *et al.* [41] among others, is to let a remote sensing expert who is familiar with the terrain select rectangular patches, each covering (part of) only one object. Campbell [19] lists several concerns that the analyst must reckon with when selecting these areas. First of all, each patch should have a minimum size to provide accurate statistics, but the size should not be too large as large areas include undesirable variation. A related issue is the number of training areas, which for instance depends on the variation within a class and the number of classes to be represented. A third concern is that the area must be localisable based on landmarks present in the image as well as on the map, if a map is used to delineate training areas that are difficult to access. Furthermore, each patch must be placed well within the object bounds to avoid inclusion of mixed pixels, and the areas must be distributed throughout the image in order to capture the diversity within the scene. Last but not least, each area should exhibit a unimodal frequency distribution for each spectral band; if the data displays a, for example, bimodal histogram, the area should be redefined or discarded. Based on all these considerations, Campbell suggests that it is probably better to use a rather large number of small training areas than a few large ones.

The other approach towards selecting homogeneous areas is to make a segmentation of the image. As is described in Chapter 5, there are several ways to go about this automatically, including edge detection and region growing. Whereas the former method looks for discontinuities in the image to determine the object boundaries, the latter method groups adjacent pixels with similar spectral reflectances to determine the object's interior. Once the objects have been established, some of the selection criteria mentioned in the previous paragraph could be enforced using standard image processing techniques; for instance, finding a training area that is placed well within the bounds of an object can be accomplished by shrinking the original object boundaries by several layers of pixels. A disadvantage of this unsupervised approach, though, is that the algorithms define spectral instead of informational classes. To link the spectral classes to the informational classes the user is really interested in, a knowledgeable analyst or supervised data is required.

4.2.6 K-means clustering: experiments with MAIS imagery

In case no supervised information such as a spectral library or a set of pixels—pure or mixed—with known composition is available, unsupervised methods are needed. As discussed before, principal component analysis is a major step in the right direction, but human interaction is still required to determine the endmember spectra. Calculating endmember spectra from selected homogeneous regions, on the other hand, can be done completely automatically, but this method can only be applied if the objects in the image are much larger than a pixel. In this section, it is shown that K-means clustering can be used to determine the endmembers of a hyperspectral image with very little human guidance for objects that are relatively small. After a brief explanation of the K-means clustering algorithm, two experiments are described based on the MAIS image presented in Section 4.1.3. In the first experiment, the endmembers derived by K-means clustering are compared to the endmembers determined by a few other methods. In the other experiment, the influence of the only parameter that needs to be set manually—the value of K—is studied in order to see if its value may be set automatically.

The K-means clustering approach

K-means clustering is a well-known method that traditionally is used for unsupervised classification. It can however also be used to determine endmember spectra because class means, which can be calculated after classification, are closely related to endmembers. Since the classification procedure is meant for pure pixels only, endmember spectra determined with the K-means clustering approach are more accurate as the ratio of pure vs. mixed pixels increases. Therefore, in practice K-means clustering can be applied only if objects in the image cover at least one pixel.

Prior to executing the K-means algorithm, one has to decide on the number K of classes or, in our case, endmembers (so $K{=}c$) to be discerned. Usually the choice of K is based on domain knowledge, but it may be possible to determine it from the image itself—see the second experiment below. The algorithm begins with initialisation of the K cluster means $\mathbf{m}_j$. In [95] several possibilities are described, which nearly all lead to approximately the same solution in the end; we started with $m_{ij}{=}256{*}j/(K{+}1)$, which represent equally spaced flat spectra. Next, each pixel in the image is assigned to the cluster to which its Euclidean distance is minimal. After all pixels have been dealt with, each cluster mean is recomputed by averaging the pixels assigned to it. The cycle of assignation and recomputation is repeated until there is virtually no change in the cluster means: $\sum_{i=1}^{n}\sum_{j=1}^{K}(m_{ij}^{(k+1)} - m_{ij}^{(k)})^2 \leq t^2$ for some threshold t. Finally, as is necessary for any unsupervised classification method, the spectral classes represented by the clusters have to be linked to the informational classes the user is interested in. As described before, this step requires some sort of expert knowledge after all.

If many of the pixels processed by the K-means algorithm are mixed, the cluster means will no longer be good representations of the endmember spectra. A solution to this problem is to post-process the cluster means by separating the pure from the mixed pixels using one of the techniques discussed in Chapter 5. As will be shown in the comparative experiment described later, one possibility is to derive impure endmembers using K-means clustering, decompose all pixels using these spectra, and recalculate the spectrum of endmember j by averaging the pixels with $f_j{\geq}85\%$. Another possibility, which only applies when the size of the objects is large enough, is to locate regions in the image where all pixels belong to the same cluster; by excluding the outer two layers of pixels, which are more likely to be mixed, and replacing the cluster means by the mean vectors calculated based on the remaining pixels, purer endmember spectra are obtained.

Comparison with a few other methods

In the following little experiment, the quality of the endmembers derived by K-means clustering is compared to the quality of endmembers obtained with several other methods in order to test the suitability of the K-means clustering approach for the extraction of endmember spectra from an image. As usual, the division between set-up, results, and discussion of the experiment is made.

Set-up

The first step of the experiment was to apply the different methods to the hyperspectral MAIS image of the Yellow River Valley shown in Figure 4.4. Six methods were used to determine the image's endmembers, two of which used supervised data, whereas the other four were based on unsupervised techniques. The first set of endmember spectra consisted of the three pixels—one per class—selected on-screen by a remote sensing expert familiar with the region. With these spectra, areas of $5x5$ pure pixels all belonging to the same class were identified as described in Section 4.1.3, which were used to calculate the second set of endmember spectra. The third set, also containing three endmembers, was determined using principal component analysis—see Section 4.2.2. The last three sets were derived by applying the K-means (K=3) clustering approach, combined with either (or none) of the post-processing methods mentioned above; one method selected the inner $8x8$ pixels of $10x10$ pixel areas of one class, the other selected pixels having a fraction of at least 85%.

The simplest way to determine the quality of the endmember spectra would be to compare them to the real spectra to see how well they correspond. However, since the real endmember spectra were unknown[3], a different approach was needed. As an alternative, we decomposed all pixels in the MAIS image using each of the six sets derived. If supervised information about the composition of some pixels had been available, we could have taken the accuracy of the decomposition as a quantitative measure to select the best endmember set. Instead, we determined the other three error measures described in Section 4.1.3, i.e. ε_x, ε_{sum}, and ε_{pos} (see Equations (4.12)–(4.14)). The measures ε_x and ε_{sum} indicate how well the pixels can be represented by the plane through the three endmembers used; in the absence of sensor noise and within-class variation, both should be zero if the endmember spectra are perfect. Under the same assumption, the value of ε_{pos} should be zero as well; a non-zero value may be an indication that the endmember spectra are impure (Drake and Settle [24]). Obviously, the value of ε_{pos} is an important measure to assess the quality of the endmembers. Most of the decomposition methods for hyperspectral data suggested in the first half of this chapter, however, always give an ε_{pos} equal to zero. Therefore, iterative least-squares was chosen as the decomposition method to be used in this experiment.

[3] The three supervised pixels picked by the expert were only his best guess.

method	ε_x	ε_{sum} (%)	ε_{pos} (%)
supervised	0.080	4.2	0.2
supervised/areas	0.082	4.8	0.1
PCA	0.137	3.4	0.6
K-means	0.072	2.8	0.6
K-means/areas	0.061	2.6	0.5
K-means/$f \geq 85\%$	0.061	2.7	0.4

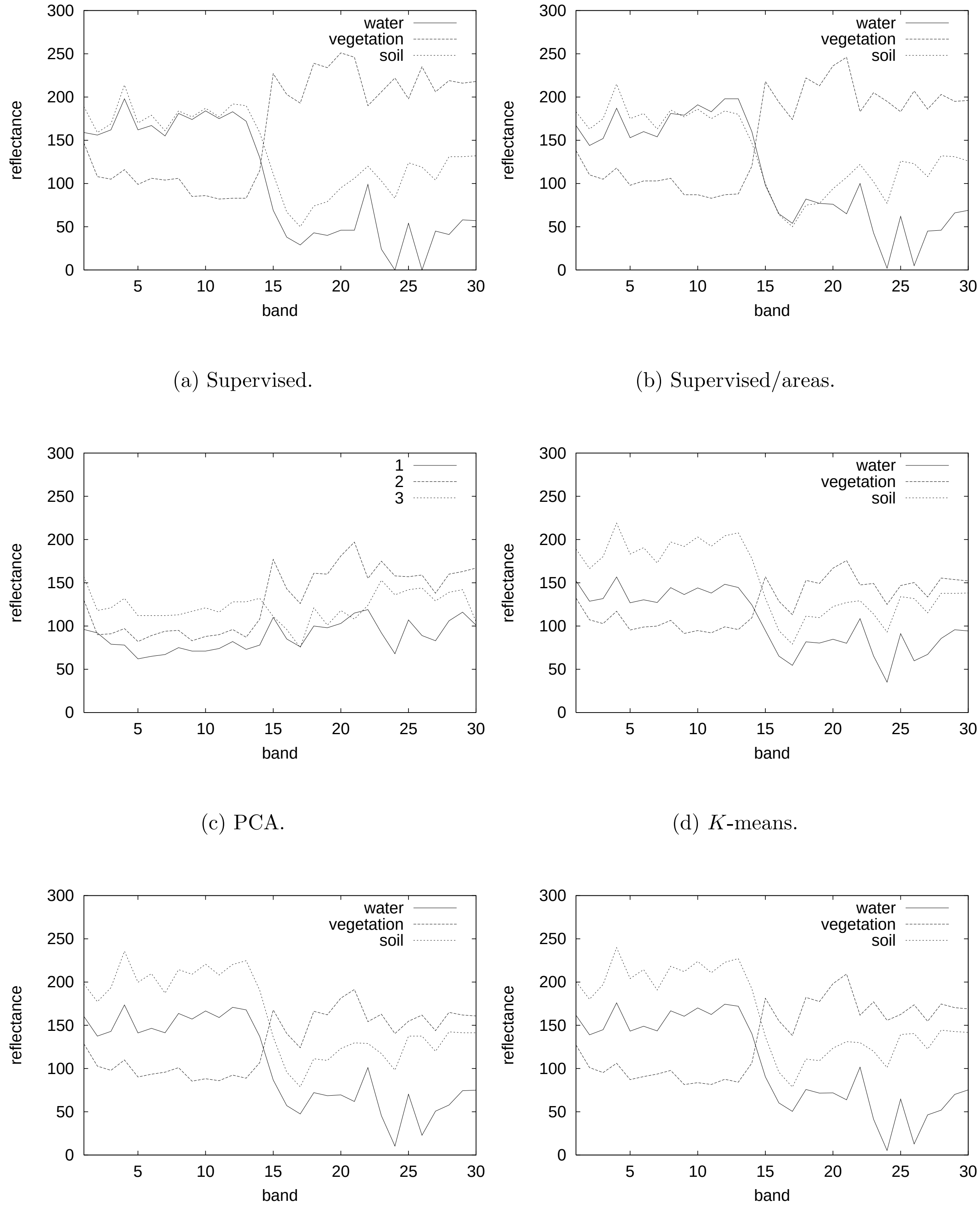
Table 4.4: *Performance of the iterative least-squares method (t=0.001) on the MAIS image for various endmember sets.*

Results

In Figure 4.8 the endmember spectra derived by the various methods are shown. Based on the reflectance of the pixels picked by the remote sensing expert, the other endmember spectra could be labelled as water, vegetation, or soil as well. However, the endmember spectra found by principal component analysis, see Figure 4.8(c), did not resemble the supervised spectra much and consequently were not linked to these informational classes. Table 4.4 presents the values of ε_x, ε_{sum}, and ε_{pos} after decomposition of all pixels using the iterative least-squares method (t=0.001). The results achieved with the PCA endmembers are much worse than those achieved with other endmember sets, which was no surprise because the PCA spectra were quite different. Compared to the supervised methods, the methods based on K-means clustering give endmembers with which the pixels could be reconstructed better: both ε_x and ε_{sum} are considerably lower. However, since the supervised methods resulted in a lower ε_{pos}, their endmembers probably were more pure. Figure 4.9 shows the distribution of the negative fractions which were used to calculate ε_{pos}. The graphs of supervised methods clearly enclose the smallest area, meaning they resulted in a smaller number of negative fractions. Furthermore, their peaks are quite close to 0%, which suggests that most violations of the positivity constraint were minor. The graph corresponding to PCA shows two strange peaks at -1% and just before -2%; this probably indicates that the corresponding spectra were not pure but mixtures of the true endmember spectra—see Figure 4.10 for an explanation of how impure spectra can cause the aforementioned peaks. The graph of the basic K-means clustering approach also shows two extra (less pronounced) peaks, indicating that its spectra were somewhat impure as well. At least one of these peaks is absent in the graphs of the other two methods using K-means clustering; combined with the fact that the latter two graphs enclose smaller areas as well, this confirms that post-processing of the basic K-means spectra indeed led to purer endmember spectra.

Discussion

Although the results presented the previous section are based on a single image, which makes any fact being far from conclusive, several interesting observations can be made. First of all, the endmember spectra derived using supervised data appeared to be better in accordance with the (unknown) true endmember spectra than those obtained with the unsupervised methods. Although the pixels could be reconstructed better with the latter



(a) Supervised.

(b) Supervised/areas.

(c) PCA.

(d) K-means.

(e) K-means/areas.

(f) K-means/$f \geq 85\%$.

Figure 4.8: *Endmember spectra determined using the various methods.*

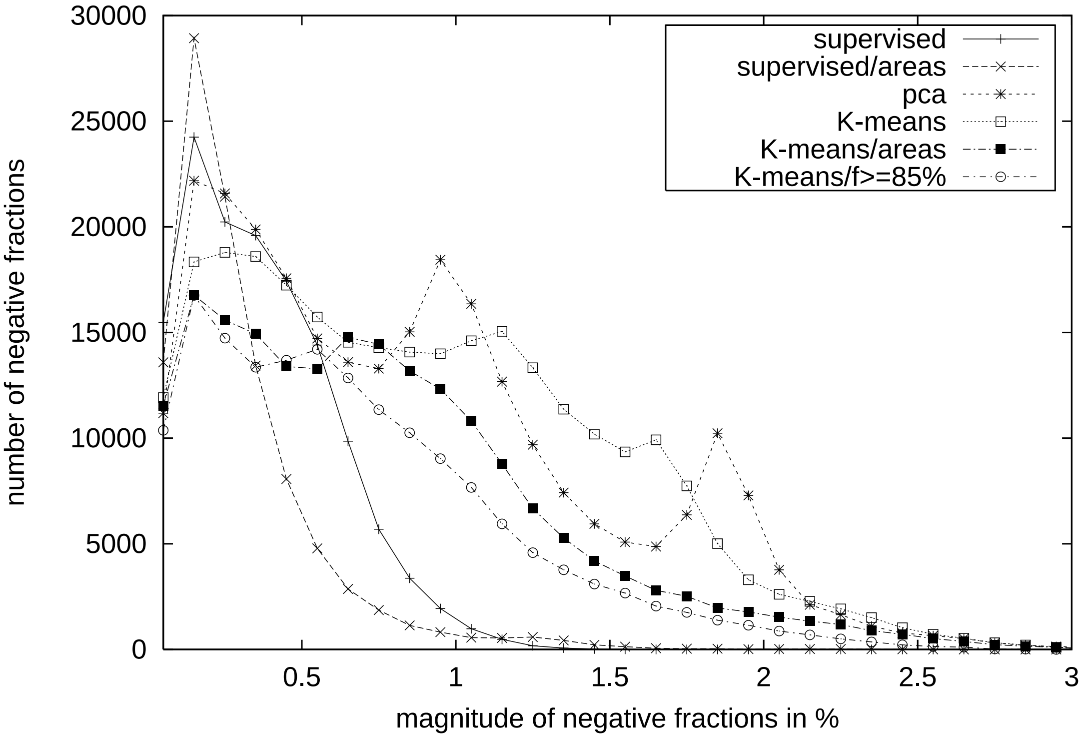


Figure 4.9: *Distribution of the negative fractions for various endmember sets.*

spectra, the former spectra led to less and less serious negative fractions, which is an indication that the spectra were more pure. The use of pure pixel areas identified using supervised data increased the purity of the endmembers slightly. From the unsupervised methods, principal component analysis has the highest potential to determine the true endmember spectra. In our experiment, however, PCA performed poorly, probably because the first two principal components that were used to determine the three endmembers did not contain sufficient information to described the true spectra correctly. The fact that the spectra derived were much flatter and closer to each other than the supervised spectra, as well as the fact that the distribution of the negative fractions displayed two unexpected peaks, suggested that the PCA spectra were mixtures of the true endmember spectra. The same two observations can be made for the endmembers derived with the basic K-means clustering approach. An explanation for the extraction of impure spectra by K-means clustering is simple: the mixed pixels in the image draw the cluster means away from their optimal position. As is shown in Figure 4.10, this leads to endmembers that lie closer together as well as to a larger number of negative fractions; decomposition of the pure pixels may be the cause of the unexplained peaks mentioned earlier. Both post-processing methods applied improve the spectra of K-means clustering by making them purer, which can be inferred from the larger distance between the spectra and the lower value of ε_{pos}. The method selecting pixels with a fraction of at least 85% appeared to perform a little bit better than the method selecting the $8x8$ pixels inside areas of $10x10$ pixels belonging to the same class, but it requires a far greater number of computations. On the other hand, it can be applied successfully if the objects in the image cover about one pixel instead of the

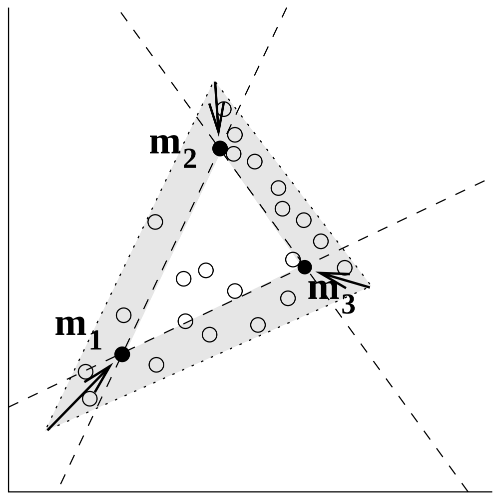


Figure 4.10: *The influence of mixed pixels on K-means clustering. Clustering of mixed pixels draws the cluster means $\mathbf{m}_j$ away from the positions of the true endmembers. Decomposition of pixels in the grey region will result in one or more negative fractions (compare Figure 4.2). Pure pixels of the same class will have approximately the same negative fractions, causing a peak in the distribution of the fractions if there are many.*

block of pixels needed by the area selecting method, which makes it applicable in a wider number of situations. In summary, the endmember spectra of images containing relatively small objects can be determined best using supervised data; in case no such information is available, K-means clustering is a good alternative to the PCA approach as it is more reliable and requires virtually no human guidance.

Varying the number of clusters

A disadvantage of K-means clustering is that the number of classes K has to be decided on in advance. In some cases the value of K can be set by an expert familiar with the terrain, but this knowledge may not always be available. Therefore, for the method to be truly unsupervised, it should be possible to determine K from the image itself, preferably automatically. The feasibility of this approach is studied in the next experiment, of which the set-up, results and discussion are presented below.

Set-up

The MAIS image of the Yellow River Valley, which was also used for the other two experiments described in this chapter, was subjected to K-means clustering using a number of different values of K. The resulting endmember spectra were post-processed using the area selection method rather than the method selecting pixels with high proportions. Although the latter approach proved to perform a little better in the previous experiment, the former method is much faster, especially for larger K. As before, the endmembers were evaluated by applying iterative least-squares decomposition to the entire image, while calculating the error measures ε_x, ε_{sum}, and ε_{pos}. The idea behind this set-up is that, as K increases, new endmembers will eventually be just mixtures of earlier discovered endmembers without

K	ε_x	ε_{sum} (%)	ε_{pos} (%)
2	0.091	3.4	0.4
3	0.061	2.6	0.5
4	0.052	2.5	0.4
5	0.050	2.6	0.4

Table 4.5: *Performance of ILS (t=0.001) for endmember sets found using various K.*

any additional information content. Although the reconstruction of the pixels will always be more accurate if more endmembers can be used, this improvement will eventually be marginal. The value of K that resulted in the last substantial decrease of the reconstruction error is likely to give the correct number of endmembers.

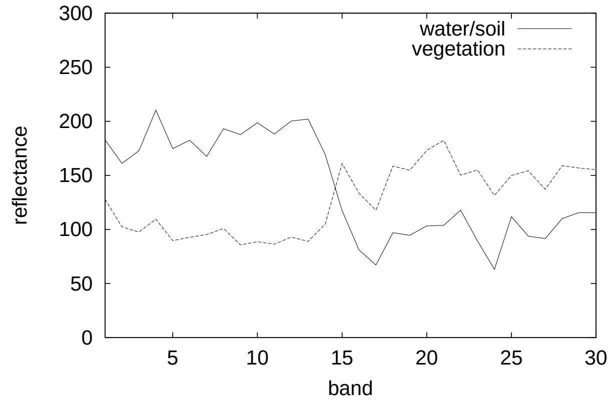
Results

Figure 4.11 displays the endmember spectra derived with the K-means/areas approach for K ranging from two to five. The spectra of water, vegetation, and soil, which were identified by a remote sensing expert during the previous experiment, were found and labelled accordingly for all values of K, except K=2 of course. One of the two spectra extracted in case K=2 could be identified as vegetation, the other appeared to be a mixture of water and soil. Figures 4.11(e) and 4.11(f) show the similarity of the spectra for these three classes, and also confirm that the supposedly mixed spectrum found for K=2 indeed is a mixture of water and soil: the mixed spectrum (the thick solid line) is situated precisely in between the spectra of water and soil. The results of iterative least-squares decomposition (t=0.001) using the different endmember sets are presented in Table 4.5. Clearly, the image is composed of more than two endmembers, since K=3 leads to a much lower ε_x than K=2. The use of a fourth endmember decreases the error a little more, but the fifth endmember derived when K=5 brings no real improvement anymore. The remaining reconstruction errors are most likely due to within-class variation and sensor noise.

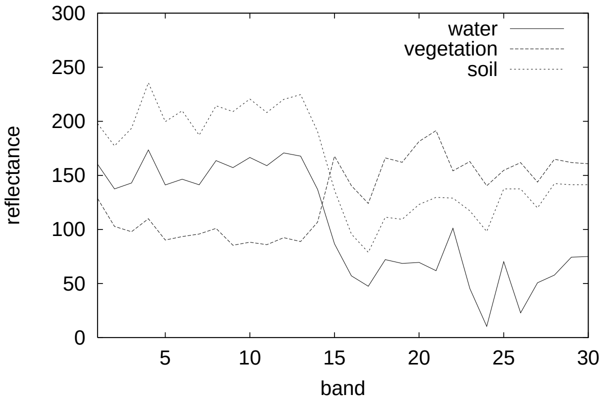
For the task of automatically selecting the value of K, the error measure ε_x calculated for the entire image seems to be a useful criterion. However, repeatedly decomposing the image requires many computations, especially when the number of classes increases. Since the main idea is to increase K until the new spectra are mixtures of the endmember spectra found earlier, the optimal value of K may also be determined directly by decomposing the spectra of K=k using the spectra of K=$k-1$, which is much faster. As before, the reconstruction error will be high if the new spectrum cannot be represented by the previously found spectra, while it will be small if the larger endmember set contains nearly identical

k	ε_x	ε_{sum} (%)	ε_{pos} (%)
3	0.081	3.4	0.2
4	0.034	1.7	0.5
5	0.017	1.2	0.3

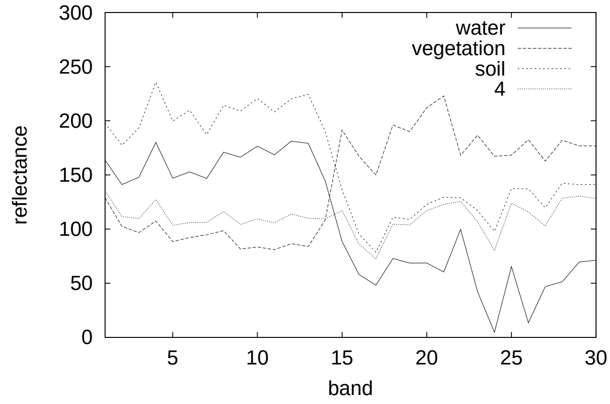
Table 4.6: *Results of iterative least-squares decomposition of the endmember set corresponding to K=k using the set corresponding to K=$k-1$.*



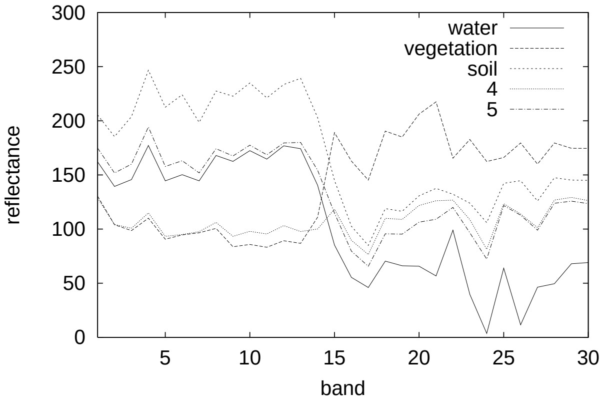
(a) K=2.



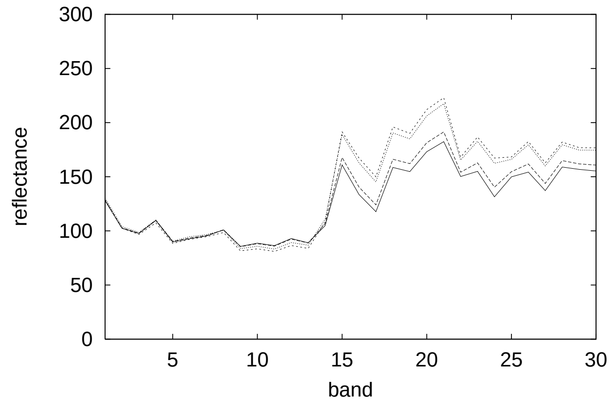
(b) K=3.



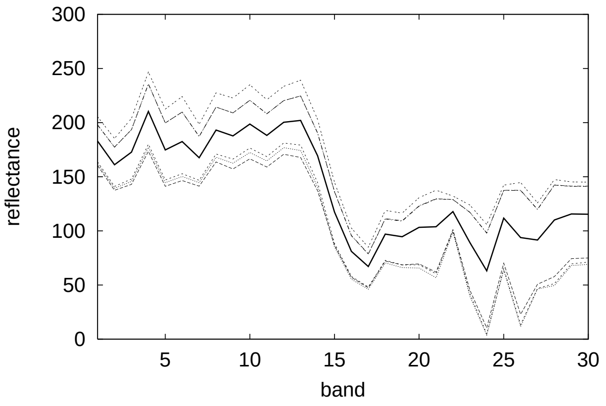
(c) K=4.



(d) K=5.



(e) Vegetation.



(f) Water and soil.

Figure 4.11: *Endmember spectra determined for various values of K.*

endmembers as the smaller set plus one that is a mixture. Table 4.6 shows the results of this approach. As expected, the information content of the additional endmembers decreases as K increases, but the choice of the right K is not as simple as before. However, based on the reconstruction error due to within-class variation and sensor noise expected, it may be concluded that no more than three endmembers are needed: ε_x=0.081 for K=3 is still higher than the expected level of 0.050, but ε_x=0.034 corresponding to K=4 is not.

Discussion

The MAIS image which was used for the experiment appeared to be composed of three distinct endmembers, which already had been identified by a remote sensing expert familiar with the region. This conclusion is based on the fact that nearly identical endmember spectra corresponding to water, vegetation, and soil were found independent of the value for K. The status of the fourth endmember is unclear as it did lead to a somewhat smaller reconstruction error, but it could also be a mixture of the other three. Fieldwork to determine the true composition of the pixels belonging to this cluster can answer this question; a less expensive solution is to compare this spectrum with those in a spectral library to find out whether it corresponds to a known ground cover type or not. In any case, more than four endmembers definitely were not needed to decompose the image; spectrum five clearly was a linear combination of the other endmember spectra. Based on these results, it can be concluded that a reasonable estimate of the number of endmembers can be made based on the image itself, at least for the image used in this experiment. Furthermore, the procedure to determine K can easily be automated by monitoring the value of ε_x. If ε_x is calculated using all pixels in the image, the extraction of extra endmembers can be stopped if the decrease in ε_x is below a certain threshold. If, to speed up the process, not the pixels in the image but the endmember spectra themselves are decomposed to determine ε_x, the procedure can be stopped if ε_x reaches a level due to within-class variation and sensor noise. Further experimentation with other images—for which ideally the true value of K is known—is needed to decide whether the latter approach is truly applicable.

4.3 Summary and conclusions

In this chapter two subjects were dealt with, being satisfaction of the positivity constraint and determination of endmember spectra. Although these problems are in principle independent, they were put in the same chapter because they both need to be solved when applying the classic linear mixture modelling approach.

If the linear mixture model is solved by analytic approximation, which is the usual method to decompose mixed pixels nowadays, satisfaction of the positivity constraint is a problem because it cannot be reckoned with when deriving the solution. This does not hold for other methods of solution such as brute force approximation, but the computational complexity of these approaches often is too high when decomposing hyperspectral data. Therefore, the positivity constraint is usually satisfied by post-processing the analytically derived estimates. Two examples of this approach are the standard renormalisation technique and the constrained least-squares (CLS) method of Shimabukuro and Smith.

Whereas the former method is extremely simple and fast but results in sub-optimal estimates, the latter approach is more accurate but requires a case distinction that gets complicated if the number of endmembers exceeds three. An alternative introduced in this thesis is the iterative least-squares (ILS) method, which adapts the estimates of the standard least-squares estimator in order to enforce the sum-to-one and the positivity constraint to some degree. Since ILS uses gradient descent to minimise a cost function in such a way that the computational complexity depends on the number of endmembers instead of the number of spectral bands, the procedure is well suited to decompose hyperspectral data. To compare the above three methods, an experiment based on artificial MAIS data (n=30) was carried out. As expected, the renormalisation approach proved to be faster but less accurate than CLS. The performance of ILS depended on its threshold setting: as the threshold was lowered, the positivity constraint was satisfied better at the cost of a higher number of computations. A surprising observation was that ILS could achieve an accuracy that was even higher than that of CLS provided the threshold was set low enough. Combined with the fact that its concept is relatively simple, ILS appears to be a decomposition method for hyperspectral data worthy of further investigation.

The other subject dealt with in this chapter is how to determine the image endmembers, which must be known before the linear mixture model can be solved. Apart from using spectral libraries, several techniques based on supervised data can be applied, e.g. linear regression of mixed pixels of known composition and calculation of the mean of pure pixels belonging to the same class. If no supervised data is available, unsupervised methods can be used to determine the endmember spectra from the image itself, like principal component analysis (PCA) and selection of homogeneous areas based on segmentation. Unfortunately, PCA needs human guidance and is difficult if more than three endmembers are involved. The area selection method, on the other hand, requires that the image contains objects that cover a considerable number of pixels each, which may not always be the case. As an alternative, K-means clustering can be applied, because the clusters it finds define unsupervised classes whose means may be taken as endmembers. Compared to the other unsupervised methods, this approach has the advantage that virtually no human guidance is required and that the objects need to have a minimum size of only one pixel. In an experiment based on the same MAIS image used before, the results of several supervised and unsupervised methods were compared. Although the image pixels could be reconstructed less well with the endmembers found by the supervised methods, the spectra themselves appeared to be relatively pure. The performance of PCA was rather poor, probably because too few principle components were used to define the endmember spectra. With the spectra found by K-means clustering, the pixels could be reconstructed well; however, the spectra appeared to be less pure than those of the supervised methods, which was due to the presence of mixed pixels throughout the image. The use of post-processing methods selecting relatively pure pixels improved the purity of the spectra considerably. A disadvantage of the K-means clustering approach is that the number of classes K has to be specified in advance. However, a second experiment showed that, in case K-means clustering is applied to determine image endmembers, the optimal value of K can probably be determined automatically in a fast way. To decide whether the suggested procedure truly is applicable in practice, futher research is needed.

Part II

Area estimation of agricultural fields

Chapter 5

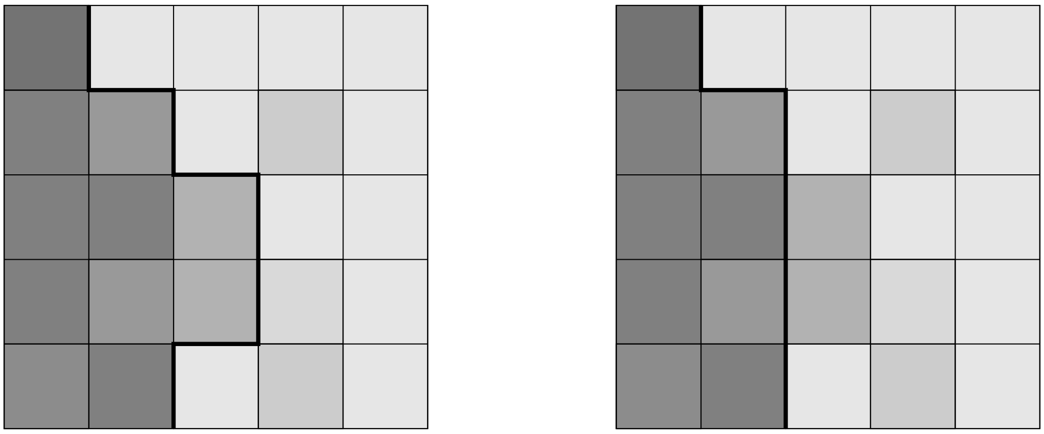
Detection of mixed pixels*

In some cases it is advantageous to discriminate between pure and mixed pixels. One example is the determination of endmember spectra, which can be done more accurately and with greater ease using pure pixels. Another example is area estimation based on counting the pixels (or fractions of pixels) per class: although decomposition is needed for mixed pixels, classification is a more accurate and faster technique for pure pixels. An obvious approach to detect mixed pixels, which was applied in Section 4.2.6, is to decompose all pixels and consider those without one clearly high proportion to be mixed; Wang [111] as well as Foody and Cox [29] suggest using membership grades—acquired with fuzzy classification (Section 2.2.4) or fuzzy c-means (Section 2.2.5), respectively—for the same purpose. However, such an approach is not particularly suitable for the above examples as the endmember spectra need to be known in advance and pure pixels are still decomposed, which is computationally expensive. When dealing with large objects such as agricultural fields, alternative detection methods based on segmentation can be applied, which are reviewed in Section 5.1. Section 5.2 describes an experiment to validate the segmentation approach, while Section 5.3 gives a summary and presents the conclusions.

5.1 Segmentation of agricultural scenes

An agricultural field, just like any other object whose size is large given the resolution of the scanner, can be regarded as a cluster of pure pixels surrounded by a layer of mixed pixels. This knowledge can be used to separate pure from mixed pixels by means of segmentation, which is a well-known field of research in image processing. As Nalwa [76] states informally, the *segmentation* of an image is the division of the image into fragments, or segments, each of which is homogeneous in some sense, but the union of no two adjacent segments is homogeneous in the same sense. The sense in which each segment is usually sought to be homogeneous is that it does not exhibit any abrupt intensity changes in its interior. This definition of segments corresponds nicely with the notion of pure pixel clusters, provided that the variation in intensity within each cluster is smaller than the difference in intensity between any two adjacent clusters. Thus, by using segmentation to extract sufficiently large clusters, pure pixels can be identified. Alternatively, segmentation techniques can be

*This chapter is partially based on Master's thesis [44].



(a) Left region grown first. (b) Right region grown first.

Figure 5.1: *The order dependence problem. Since the central pixel and the one below resemble both the light and the dark region, they are added to the region grown first.*

applied to find the boundaries between the clusters, identifying the mixed pixels themselves. Basically, three types of segmentation methods exist; an example of each category is given in Sections 5.1.1 to 5.1.3.

5.1.1 Region growing

Region growing methods operate by gathering adjacent pixels with similar characteristics into sets called *regions*, i.e. they try to find the pure pixel clusters themselves. The criterion to group pixels in the same set is usually based on spectral similarity, but more complex properties such as conformity to some local texture measure may be used as well. An important issue related to region growing is the order dependence problem: depending on the order in which the pixels are processed, the final segmentation may not always be the same. An example of this problem is shown in Figure 5.1. If the dark region is grown first, the central pixel and its neighbour below will be added to the left segment (Figure 5.1(a)); if, on the other hand, region growing starts with the light region, the same two pixels will end up in the right segment (Figure 5.1(b)). A way to overcome the order dependence problem is the best merge approach, which merges only the single most similar pair of adjacent regions in each iteration. However, care must be taken that order independence is maintained in case multiple pairs having the same, maximum similarity exist.

The simultaneous order independent iterative merging (SOIM) algorithm by Schoenmakers [90] is a segmentation method based on best merge region growing that was designed specifically to handle pairs of regions with identical merging costs. Unlike the best merge algorithms by Beaulieu and Goldberg and by Tilton (see [90]), which in case of ambiguity select an arbitrary pair or the one containing the lowest region label, SOIM merges all pairs with maximum similarity in the same iteration. As is shown by Algorithm 5.1, SOIM uses a dissimilarity rather than a similarity criterion to determine which regions are to be initialise each pixel as a region;

while (true) {
    compute the dissimilarity of all pairs of adjacent regions;
    determine the set of region pairs with minimum dissimilarity;

    if (stopping criterion 5.2 is met) **break**;

    merge all pairs of regions in the set;
    update the statistics of the new regions;
}

Algorithm 5.1: *Simultaneous order independent iterative merging.*

merged. Currently, the criterion is based only on the spectral reflectance of the regions,

$$C_{i,j} = \sum_{k=1}^{n} (m_{ki} - m_{kj})^2, \tag{5.1}$$

although other statistics concerning region size or shape could easily be accommodated for. Equation (5.1) defines the cost of merging regions i and j as the (squared) Euclidean distance between the two region means, which are calculated by averaging the values of the corresponding pixels in each the c spectral bands. The stopping criterion used by SOIM is related directly to the dissimilarity criterion:

$$\sqrt{\frac{\min(C_{i,j})}{n}} > t, \tag{5.2}$$

which boils down to the condition that the RMS distance between the two most similar, adjacent regions must be below (or equal to) some threshold t if the merging process is to continue. What cannot be deduced from Algorithm 5.1 is what course of action is taken if two (or more) of the pairs to be merged have a region in common. In this case, which rarely occurs except in the first iteration, all regions in question are merged into one as they obviously are spatially contiguous. For a more formal description of the SOIM algorithm including optimisation the reader is referred to [90][1].

5.1.2 Edge detection

A segmentation approach that is dual to finding the homogeneous interior of a region is to determine where the region boundaries are located. This strategy is usually implemented using *edge detection* methods, which look for local discontinuities in the grey level distribution of an image. Since in general the strongest *edges* coincide with region boundaries, edge

[1]Note that step 2(ii) of Schoenmaker's Algorithm 5.4 is incomplete; a correct formulation of the set is
$$CS_{nbr} = \{\langle S_u, S_k\rangle \mid \langle S_u, S_k\rangle \in CS \wedge ((\langle S_u, S_v\rangle \in CS_{min} \wedge v \neq k) \vee \langle S_v, S_u\rangle \in CS_{min})\} \cup \{\langle S_k, S_u\rangle \mid \langle S_k, S_u\rangle \in CS \wedge (\langle S_u, S_v\rangle \in CS_{min} \vee (\langle S_v, S_u\rangle \in CS_{min} \wedge v \neq k))\}.$$

$$
\underset{\mathbf{T}_1}{\begin{array}{|c|c|c|}\hline \frac{1}{2\sqrt{2}} & \frac{1}{2} & \frac{1}{2\sqrt{2}} \\ \hline 0 & 0 & 0 \\ \hline -\frac{1}{2\sqrt{2}} & -\frac{1}{2} & -\frac{1}{2\sqrt{2}} \\ \hline \end{array}}
\qquad
\underset{\mathbf{T}_2}{\begin{array}{|c|c|c|}\hline -\frac{1}{2\sqrt{2}} & 0 & \frac{1}{2\sqrt{2}} \\ \hline -\frac{1}{2} & 0 & \frac{1}{2} \\ \hline -\frac{1}{2\sqrt{2}} & 0 & \frac{1}{2\sqrt{2}} \\ \hline \end{array}}
\qquad
\underset{\mathbf{T}_3}{\begin{array}{|c|c|c|}\hline 0 & -\frac{1}{2\sqrt{2}} & \frac{1}{2} \\ \hline \frac{1}{2\sqrt{2}} & 0 & -\frac{1}{2\sqrt{2}} \\ \hline -\frac{1}{2} & \frac{1}{2\sqrt{2}} & 0 \\ \hline \end{array}}
\qquad
\underset{\mathbf{T}_4}{\begin{array}{|c|c|c|}\hline \frac{1}{2} & -\frac{1}{2\sqrt{2}} & 0 \\ \hline -\frac{1}{2\sqrt{2}} & 0 & \frac{1}{2\sqrt{2}} \\ \hline 0 & \frac{1}{2\sqrt{2}} & -\frac{1}{2} \\ \hline \end{array}}
$$

Figure 5.2: *The normalised Frei-Chen edge templates.*

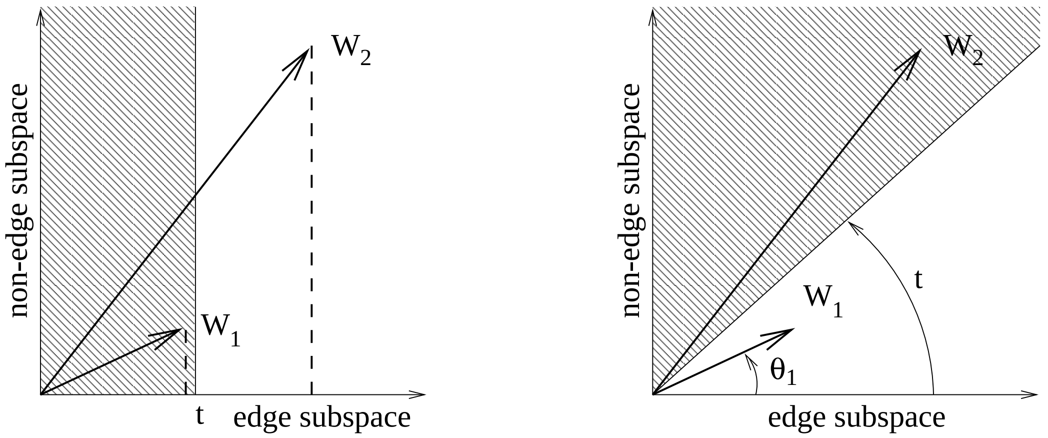
detection can be used to locate the layer of mixed pixels surrounding each pure pixel cluster. In the last decades many edge detection methods have been proposed, which Ballard and Brown [8] classify as mathematical gradient operators, template matching operators, or operators based on a parametric edge model. Though their principles may be very different, (nearly) all these operators regard an edge as having a *direction*, which aligns with the direction of the maximum grey level change, and a *magnitude*, which describes the severity of this change. By calculation of these features for each pixel, edge magnitude and edge direction images are derived, from which one pixel wide boundary structures can be extracted by connecting the most likely edge pixels. Optionally, closed boundaries can be formed by bridging small gaps between the boundary structures found before and pruning those fragments whose ends could not be linked. A detailed description of procedures for contour following, linking, and pruning is given by Schoenmakers [90].

An example of an edge detection method based on template matching is the modified Frei-Chen operator, which is also explained by Schoenmakers [90]. The idea behind this operator is to represent the image in a different coordinate system which permits an easy separation of edge and non-edge features. Calculation of edge information is done for each band separately and is based on windows of $3x3$ pixels. Each window can be conceived as a vector in the nine-dimensional space[2] spanned by the nine natural basis vectors $\mathbf{b}_i$, which consist of all zeroes but a one on position i. The same space, however, is also spanned by templates $\mathbf{T}_1$–$\mathbf{T}_4$ shown in Figure 5.2 completed with five other templates $\mathbf{T}_5$–$\mathbf{T}_9$, which are all normalised and orthogonal to each other. The templates $\mathbf{T}_1$–$\mathbf{T}_4$ form the so-called edge subspace onto which the projection of a window $\mathbf{W}$ representing an ideal edge is equal to $\mathbf{W}$ itself. By calculating the angle θ between $\mathbf{W}$ and its projection or, more efficiently, the ratio r between the squared lengths of $\mathbf{W}$ and its projection ($\cos\theta = \sqrt{r}$), the edge magnitude can be determined:

$$r = \frac{\sum_{i=1}^{4}(\mathbf{W}\cdot\mathbf{T}_i)^2}{\sum_{i=1}^{9}(\mathbf{W}\cdot\mathbf{T}_i)^2} = \frac{\sum_{i=1}^{4}(\mathbf{W}\cdot\mathbf{T}_i)^2}{\mathbf{W}\cdot\mathbf{W}}. \tag{5.3}$$

In this equation, where $\mathbf{W}\cdot\mathbf{T}_i = \sum_{j,k} w_{jk}t_{i_{jk}}$, advantage has been taken of the fact that the $\mathbf{T}_i$ are normalised, which makes the (squared) length of $\mathbf{W}$ under the new basis equal to the (squared) length under the natural basis. As is shown in Figure 5.3, calculating the edge magnitude as an angle (or ratio) provides a better way to discriminate between edge

[2] A possible conversion between window $\mathbf{W}$ and vector $\mathbf{v}$ is $w_{ij}{=}v_{(i-1)n+j}$.



(a) Conventional thresholding. (b) Frei-Chen thresholding.

Figure 5.3: *Thresholding techniques for edge detection. (a) Separation of edge and non-edge pixels based on the gradient magnitude alone fails to detect likely edge pixels with a small gradient magnitude like* $\mathbf{W}_1$*. (b) Taking the angle* θ_1 *between* $\mathbf{W}_1$ *and its projection on the edge subspace as a criterion does mark* $\mathbf{W}_1$ *as an edge; pixels with a large magnitude who do not resemble an edge like* $\mathbf{W}_2$ *are now excluded (adopted from Ballard and Brown [8]).*

and non-edge pixels than taking the gradient magnitude alone. Whereas $\mathbf{W}_1$ has a short projection and would not be considered an edge using the conventional thresholding technique (Figure 5.3(a)), its angle to the edge subspace is sufficiently small to be considered an edge under the Frei-Chen criterion (Figure 5.3(b)). For $\mathbf{W}_2$, whose projection is long but whose resemblance to an edge is small, the opposite holds.

Just like the edge magnitude, the edge direction can also be determined using templates. If the edge direction is defined as the direction in which the transition from low grey levels to high grey level is maximum, angle α—see Figure 5.4—can be calculated as

$$\alpha = \arctan\left(\frac{\mathbf{W}\cdot\mathbf{T}_1}{\mathbf{W}\cdot\mathbf{T}_2}\right). \tag{5.4}$$

In his dissertation [90], Schoenmakers presents a simple algorithm to deal with the situation where the numerator or denominator of the above fraction is zero, as well as to correct α to a value in the interval $[0, 2\pi\rangle$. However, if the edge direction is to be used by a contour following operation later, it is more convenient to convert α into an index to one of the eight neighbours of the central pixel.

5.1.3 Map-based segmentation

The third way by which an image segmentation can be obtained is to derive it directly from ancillary data such as topographical maps. Janssen [54], for instance, used a map onto which farmers indicated their land use to determine segments representing agricultural fields. To the same end, Schotten *et al.* [91] utilised a cadastral map, which apart from parcel boundaries also showed the location of objects like roads and ditches; the field

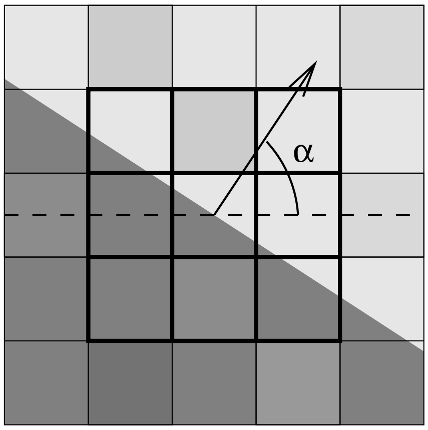


Figure 5.4: *The edge direction α is the direction of maximum change from low (dark) to high (light) grey levels. Mixed pixels are shown at subpixel resolution for reasons of clarity.*

boundaries, which are subject to change several times a year, were determined using a SPOT-XS image of a recent date. The fact that the scene under observation is constantly changing seriously limits the usefulness of maps; since keeping a map up-to-date requires a lot of work, which is only done for certain areas, map-based segmentation cannot always be applied successfully. If, however, the map is stored on a geographical information system (GIS), minor changes to it may be carried through on-screen. Another possibility to obtain an up-to-date map is to create one based on aerial photography or satellite imagery. The resolution of these images—as should be the scale of the maps mentioned before—should be finer than the resolution of the image to be segmented in order to get a segmentation that is accurate at the pixel level[3]. Furthermore, an experienced remote sensing specialist is needed to recognise the object boundaries and digitise them with sufficient accuracy.

To segment an image based on a paper map, a number of steps have to be taken. First the map has to be digitised, which is usually done by placing the map on a digitising table and marking points with a cursor at regular intervals while following the contour of an object. Each time the operator marks a point, a pair of coordinates is sent to a computer, which combines the successive points to polygons. Next, this vector data is *co-registered* with the satellite image using a set of *ground control points* (GCPs) located in both data sources. Good GCPs are features that are as small as a single pixel and which can be identified easily against their background; usually, however, road intersections or edges of land cover parcels are selected—see Figures 5.5(a) and 5.5(b). The actual co-registration is performed by mapping the vector data onto the image by means of an affine transformation; converting the image to match the vector data is possible as well, but the necessary resampling operation may render the pixels unsuitable for subsequent operations (viz. decomposition). An affine transformation scales, rotates, and translates the vector data (see Figure 5.5(c)) by evaluating the same equation for all points (x,y) digitised:

$$\begin{aligned} x' &= \alpha_1 x + \alpha_2 y + \alpha_3, \\ y' &= \alpha_4 x + \alpha_5 y + \alpha_6. \end{aligned} \tag{5.5}$$

[3]Besides the accuracy of the (digitised) map, the quality of the segmentation also depends on the accuracy with which the map and the image can be co-registered.

At least three GCPs are needed to calculate the constants $\alpha_1, \ldots, \alpha_6$, but using a higher number may increase the accuracy of the transformation, provided the quality of the additional GCPs is not significantly lower than the quality of the best three. More important than a high number, though, is that the GCPs are dispersed throughout the image with good coverage near the edges of the scene (Campbell [19]). Another advantage of using more than three GCPs is that an indication of the accuracy of the transformation can be derived, since the p (p>3) GCPs of the vector data after transformation in general do not exactly match the GCPs of the image—see Figure 5.5(d). An obvious measure to express this deviation is the RMS error between the transformed coordinates (x',y') of the vector GCPs and the coordinates (u,v) of the image GCPs:

$$\sqrt{\frac{\sum_{i=1}^{p}((x'_i - u_i)^2 + (y'_i - v_i)^2)}{p}}. \tag{5.6}$$

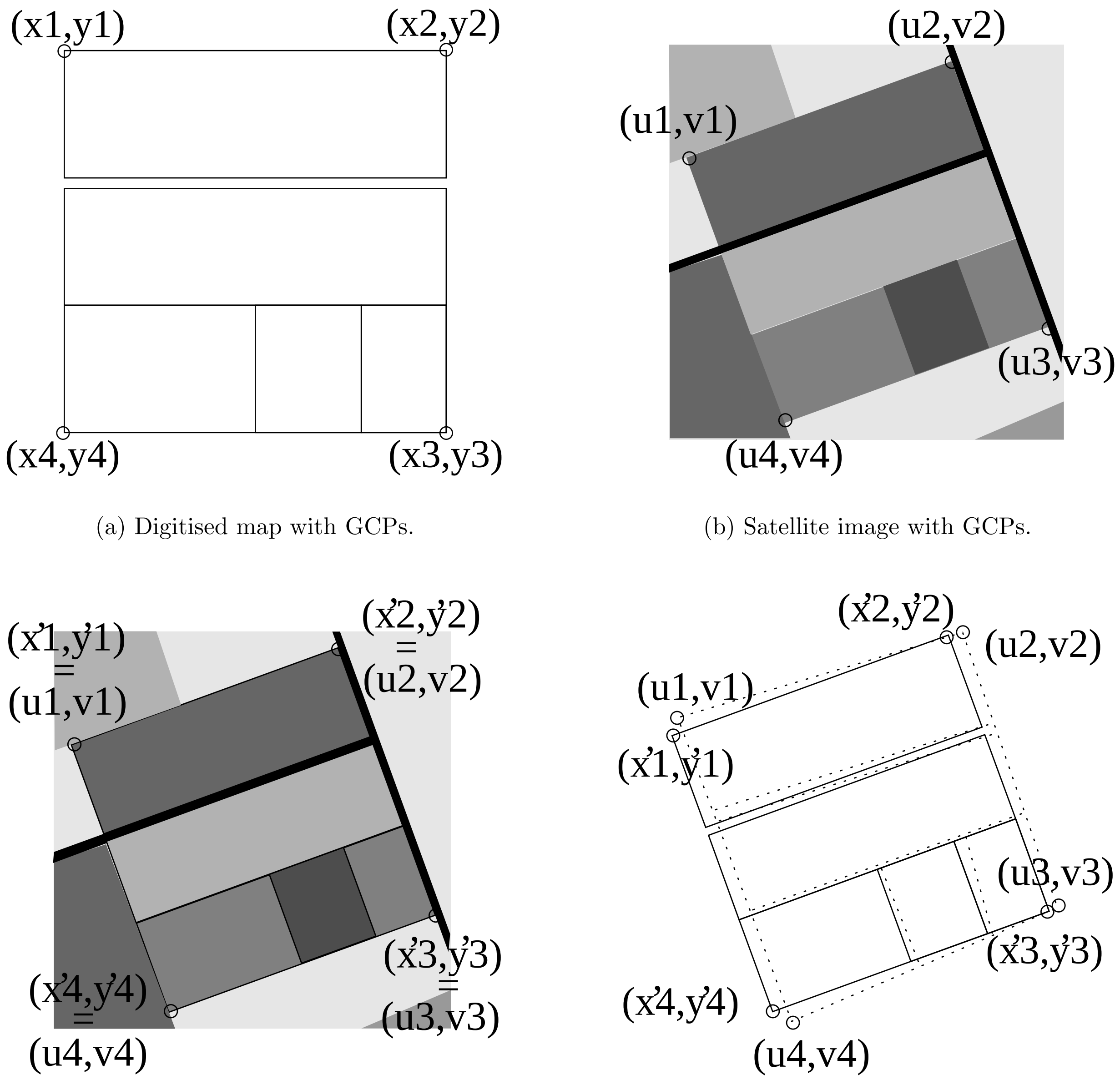
Instead of taking the mean error, one can also calculate the error of each GCP separately; by discarding GCPs giving a relatively large contribution to the overall error, a supposedly more accurate transformation is acquired. After the vector data has been co-registered with the image, it only needs to be converted to a raster data structure to obtain the desired segmentation. Each pixel for which the minimum distance from its centre to the nearest vector is less than a certain threshold value, $\frac{1}{2}\sqrt{2}$ pixel (half the pixel's diagonal) for instance, is considered to contain a boundary and therefore be a mixed pixel. The remaining pixels are regarded to be pure and are given the label of the enclosing polygon.

5.2 Experiments with artificial imagery

To determine how well mixed pixels can be detected using segmentation techniques, two of the methods described previously were applied to an artificial image. The use of artificial imagery was necessary because we had no real image with corresponding ground truth suitable to make a quantitative analysis at our disposal. As a consequence, it was senseless to apply map-based segmentation as transformation of the single available map, which formed the basis of the artificial image, would yield the perfect segmentation. The two methods we did apply were region growing and edge detection, of which the results are presented in Sections 5.2.2 and 5.2.3, respectively. Furthermore, it was investigated whether their combination would lead to better results than those achieved by both methods separately—see Section 5.2.4. A detailed description of the entire experimental set-up can be found in Section 5.2.1, while a brief discussion of the different approaches is provided by Section 5.2.5.

5.2.1 Set-up

The set-up of the experiments was straightforward. First an artificial image of an agricultural scene was generated together with its true segmentation. Next, the different segmentation methods were applied and all pixels were labelled individually as either pure or mixed. Finally, the accuracy of the detection process was determined by comparing the designated label of each pixel with its status according to the true segmentation.



(a) Digitised map with GCPs.

(b) Satellite image with GCPs.

(c) Map co-registered with image.

(d) Co-registration error.

Figure 5.5: *Co-registration of vector data and image data using an affine transformation. (a) In the digitised vector data a number of GCPs are selected. (b) The same GCPs are located in the image as well. (c) By means of an affine transformation, the GCPs* (x_i,y_i) *are transformed to* (x'_i,y'_i) *to match the GCPs* (u_i,v_i)*. (d) If more than three GCPs are used, an exact match may not be possible; in this case, the RMS error is minimised.*

To generate an artificial image, the same procedure as used in Section 3.3.1 was applied, albeit that more complex image characteristics were chosen. Not only did we use 10 instead of 2 different classes and all six bands of our Landsat-TM templates, we also assigned the ground cover type soil to the edges and replaced 1.0% of all pixels by isolated mixed pixels. As before, we made sure that fields having a boundary in common were mapped to different classes; this way, the transition from one field to another became more distinct, which prevented the segmentation methods from seeing two or more fields as one. The end result was virtually identical to Image III, which is shown in Figure 6.9. For an exact description of the image simulation methodology and class templates used, see Section 6.2.1.

The various strategies to detect mixed pixels all need certain parameters to be set. As described in Section 5.1.1, the SOIM algorithm, which was chosen as a representative of the region growing approach, requires the setting of a threshold to define the stopping criterion. After the image has been segmented using a particular threshold, large regions are considered to be pure pixel clusters, while small regions are thought to consist of mixed pixels. Therefore, another threshold needs to be set to define the maximum size of regions containing mixed pixels. The edge detection approach, for which the modified Frei-Chen operator (see Section 5.1.2) was selected, also requires setting a threshold in order to separate edges from non-edge features. Subsequently, edge pixels are labelled as being mixed and non-edge pixels as being pure, which can be done without the need for further parameters. The performance of the edge detection approach may be improved by contour following and edge linking/pruning; depending on the specific implementation, it may be necessary to set additional parameters. A way to combine the region growing and edge detection approaches is to carry out both operations separately and label a pixel as mixed if it is regarded as such by either one or both of the approaches. Although this procedure does not introduce any new parameters, it is unlikely that the parameters for region growing and edge detection can be set independently of each other, which makes the optimisation process more complex.

To determine the optimal parameter settings as well as to compare the different detection methods, a quantitative measure for the detection accuracy is needed. To this end, a confusion matrix expressing the relative performance of a detector was used. As is shown in Figure 5.6, our confusion matrix has $2x2$ entries, denoting the percentage of pure and mixed pixels according to both the true and the derived segmentation. In analogy to medicine, the percentage of pure pixels (healthy patients) that are classified as being pure (healthy) is called the specificity of the detector, while the percentage "true positives"—mixed pixels (ill patients) that are classified as being mixed (ill)—is referred to as the detector's sensitivity. The decision to use relative instead of absolute figures was taken to compensate for the fact that the number of pure pixels (35,215) was much larger than the number of mixed pixels (5,185); if absolute figures were used to measure the overall accuracy, the performance on the mixed pixels would have little influence. Ideally, both the specificity and the sensitivity should be 100%, but in practice this is not achievable at the same time. Depending on the purpose for which pure and mixed pixels are to be separated, maximisation of one of the two percentages may be preferred. If, for instance, endmember spectra are to be determined based on pure pixels, the specificity should be maximised while keeping the sensitivity above a certain minimum value, provided a higher number of pure pixels outweighs the inclusion of some extra mixed pixels with respect to

		detector		
		pure	mixed	
true	pure	specificity	100%-spec.	(100%=35,215 pixels)
	mixed	100%-sens.	sensitivity	(100%= 5,185 pixels)

Figure 5.6: *Relative confusion matrix for two classes. The rows add up to 100%.*

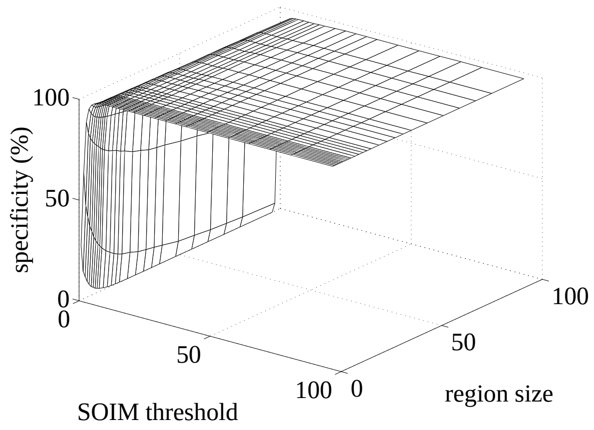
the accuracy of estimated spectra. If, however, mixed pixels are to be detected in order to decompose instead of classify them, it is more important to have a high sensitivity than a high specificity: decomposition of pure pixels may be unnecessarily slow, but classification of mixed pixels will yield inaccurate area estimates. In our experiments we decided to maximise the sum of the specificity and the sensitivity because our goal was to investigate the effectiveness of the different detection methods in general. However, if one focuses on some specific application, it may be necessary to use a different criterion to optimise the detection of mixed pixels.

5.2.2 Results of region growing

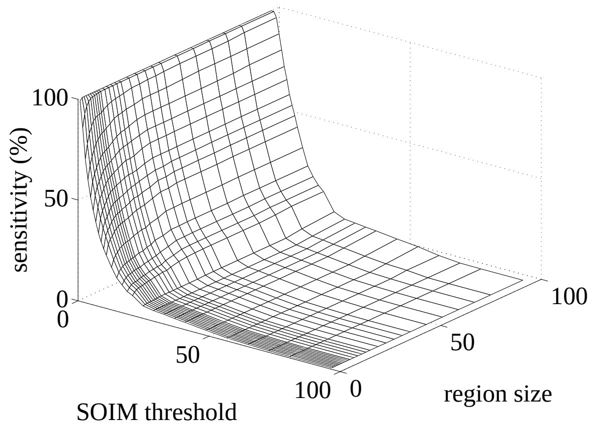
A complication of the region growing approach is that the two parameters that need to be set—the SOIM threshold and the maximum size of regions consisting of mixed pixels—cannot be optimised independently of each other, because the SOIM threshold affects the average region size. Therefore, we conducted an exhaustive search of the parameter space, of which the results are shown in Figure 5.7.

From Figure 5.7(a) it appears that the specificity—the measure for correct recognition of pure pixels—soon reaches 100% when the threshold is raised and the region size is kept constant. This was to be expected as a higher threshold setting (more within-region variation is allowed) results in larger regions, and all regions larger than a certain minimum size are assumed to consist of pure pixels. If the threshold is kept constant and the region size is increased the specificity decreases, although this is visible only at low threshold settings. Figure 5.7(b) shows that the sensitivity displays exactly the opposite pattern. If, for a particular region size, the threshold is raised, then the percentage of mixed pixels that is detected decreases considerable. However, if the region size is increased while keeping the threshold constant, the sensitivity gets slightly higher.

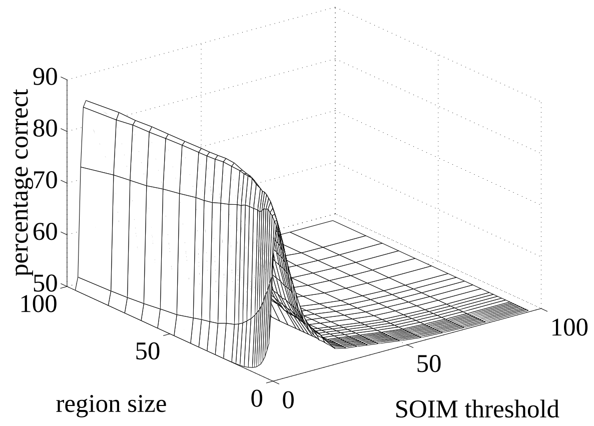
Figures 5.7(c) and 5.7(d) give the average percentage correct, which in our case is defined as the average of the specificity and the sensitivity. For low thresholds the average percentage correct clearly displays the same behaviour as the specificity, whose value obviously varies the most. At higher thresholds the pattern of the sensitivity is followed because the sensitivity still changes considerably while the specificity remains 100%. Furthermore, it turns out that optimising the SOIM threshold is much more important than an optimal setting of the region size. In Figure 5.7(e), which shows the optimal performance for the various threshold settings, it can be seen that a threshold of 3 or 4 gives the best results. As expected, the optimal region size increases as the threshold is raised. Figure 5.7(f) shows the optimal performance for different region sizes. Again it appears that the best thresh-



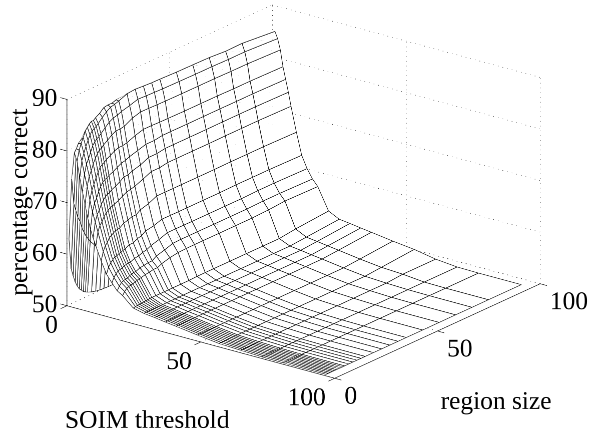
(a) Specificity.



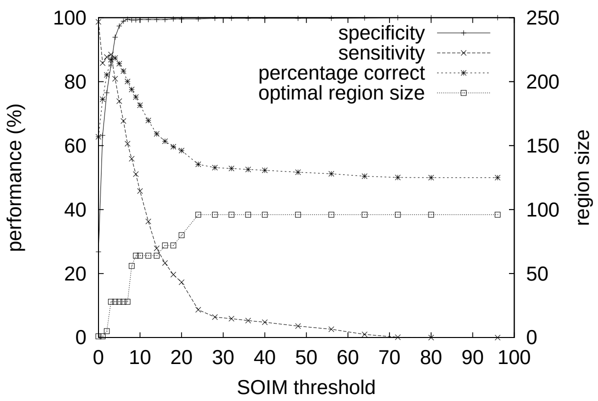
(b) Sensitivity.



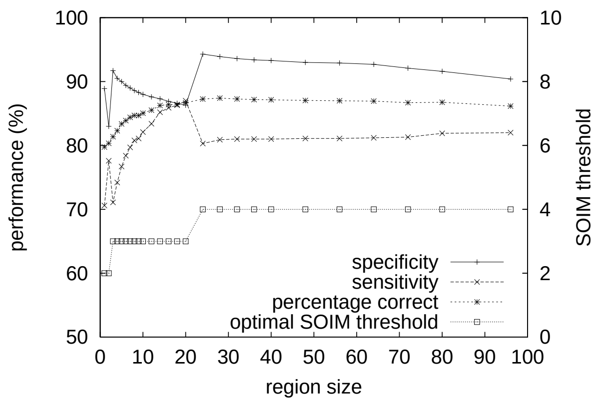
(c) Average percentage correct (view 1).



(d) Average percentage correct (view 2).



(e) Best performance per threshold setting.



(f) Best performance per region size setting.

Figure 5.7: *Performance of region growing for various settings of its two parameters.*

old setting in (nearly) all cases is equal to 3 or 4, while the choice of region size is not so important (note the different range of the left y-axis). All settings considered, a threshold of 4 combined with a region size of 28 proved to result in the highest average percentage correct: 87.4%. By comparing Figures 5.9(a) and 5.9(b), it can be seen that with these settings mixed pixels on field boundaries as well as isolated mixed pixels can well be detected. A good alternative was a SOIM threshold setting of 3 and a maximum region size of 28, which gave an average percentage correct of 86.9%. Although both results seem to be very similar, they turn out to be rather different when the specificity/sensitivity is considered: whereas a threshold of 4 leads to 93.9%/80.9%, a threshold of 3 gives 85.2%/88.5%[4]. As described in Section 5.2.1, the decision which settings to use can be made only when the application requiring detection of mixed pixels is known.

5.2.3 Results of edge detection

The performance of the Frei-Chen edge detector is dependent on the setting of its threshold. Figure 5.8 shows the specificity, sensitivity, and average percentage correct achieved for a wide range of threshold settings[5]. As expected, at low thresholds many pixels are marked as edge and therefore mixed pixels, which leads to a high sensitivity but a low specificity. If the threshold is raised, the sensitivity decreases while the specificity increases, ultimately reaching 100%. The best performance is obtained with a threshold setting of 181, which results in an average percentage correct of 77.7%, a specificity of 76.8%, and a sensitivity of 78.6%. A comparison of Figures 5.9(a) and 5.9(c) demonstrates that mixed pixels on field boundaries are detected reasonably well, although many neighbouring pure pixels are found to be mixed as well. Isolated mixed pixels, however, are recognised considerably less well. As is shown in Figure 5.9(d), often the eight pure pixels surrounding an isolated mixed pixel are regarded as being mixed, while the mixed pixel itself is considered to be pure. This behaviour is caused by the fact that the (modified) Frei-Chen templates $\mathbf{T}_1$–$\mathbf{T}_4$ are designed specifically to detect ideal edge elements. To locate ideal points, which correspond to isolated mixed pixels, other templates are required.

In an attempt to improve the performance, we also carried out some experiments using the options available in the edge detection program at our disposal [90]. The first option was contour following, which thins the edges to 1-pixel wide contours. As described by Schoenmakers [90], this is done by calculating the likelihood of being an edge (LBE) for each pixel based on its edge magnitude and direction, and, starting from virtually sure edge pixels, forming 8-connected chains of pixels having the highest LBE. The result of this operation applied to Figure 5.9(c) is displayed by Figure 5.9(e); although this image is less noisy and appears to agree more with the true segmentation represented by Figure 5.9(a), the performance in fact is worse: the average percentage correct has fallen to 60.2%, with a specificity of 87.6% and a sensitivity of 32.8%. The main cause for this deterioration is that the contours found in many cases lie just next to the true field boundaries. A second reason is that in the true segmentation diagonal borders are 4-connected whereas the contour is constructed as an 8-connected chain; as a result, the label of many edge pixels is erroneously

[4]This difference is reflected by the discontinuities between size 20 and 24 in Figure 5.7(f).

[5]Since the edge magnitudes were mapped to grey levels to create an edge image, the threshold was given a value in the interval [0,255] as well.

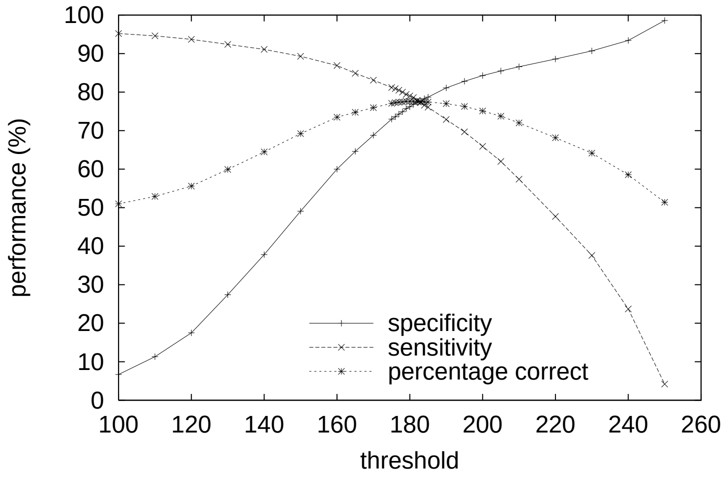
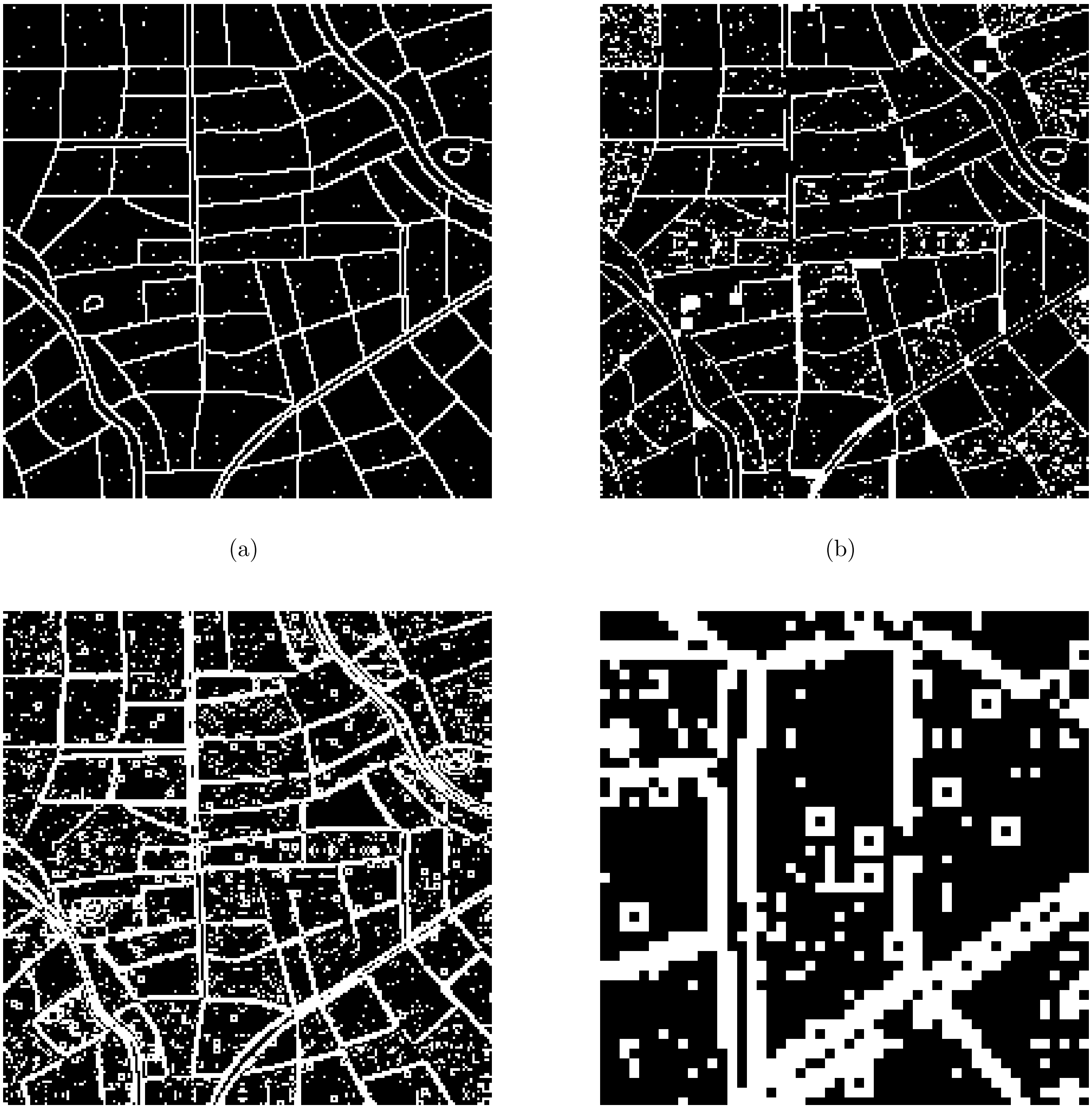


Figure 5.8: *Performance of edge detection for various settings of its threshold.*

changed from mixed to pure. Furthermore, due to within-field variation and the presence of isolated mixed pixels, a large number of small contours are found which do not exist in "reality". It is for this reason that a higher threshold leads to slightly better results: with a setting of 205 the average percentage correct increases to 61.1% (specificity/sensitivity is 91.4%/30.8%).

Another option in Schoenmaker's implementation is edge linking&pruning, where first non-edge pixels in between edge fragments are marked as edges to bridge gaps of one pixel wide, after which dangling edge fragments are deleted to create closed polygons. Besides that, very small regions are removed and care is taken that the polygons created are not wider than one pixel. The result of the edge linking&pruning operation carried out on the image acquired after contour following is displayed by Figure 5.9(f). The most visible difference with Figure 5.9(e) is that all small edge fragments have disappeared, but a number of field boundaries have been removed as well. As a consequence, the sensitivity is somewhat lower (27.6% vs. 32.8%) and the specificity somewhat higher (92.8% vs. 87.6%), but the overall performance has not improved—the average percentage correct is still 60.2%. To avoid the removal of long edge fragments corresponding to field boundaries due to gaps of more than one pixel, the Frei-Chen threshold should be lowered. With a setting of 165 instead of 181 the average percentage correct becomes 60.4%, composed of a specificity of 89.1% and a sensitivity of 31.7%. The edge linking&pruning operation can also be applied directly to the edge image (Figure 5.9(c)), which gives Figure 5.9(g). These results are better than when the contour following step is included, but compared to edge detection alone the performance is still worse instead of better: specificity/sensitivity is 88.9%/42.4%, while the average percentage correct is only 65.7%. This result is rather disappointing, because small edge fragments have been removed while retaining the field boundaries and most diagonal boundaries are now 4-connected instead of 8-connected, which leads to a higher specificity. The sensitivity, however, is considerably lower because the part of the pruning process responsible for the creation of 1-pixel wide polygons often comes up with edges lying just next to the true field boundaries, which is a problem that was also encountered during contour following.



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

Figure 5.9: *Location of the mixed pixels after applying various detection methods. (a) Positions according to the true segmentation. (b) Results of region growing (t=4, s=28). The white squares are due to the way the class templates were generated. (c) Results of edge detection (t=181). (d) Blow-up of the centre right part of (c).*

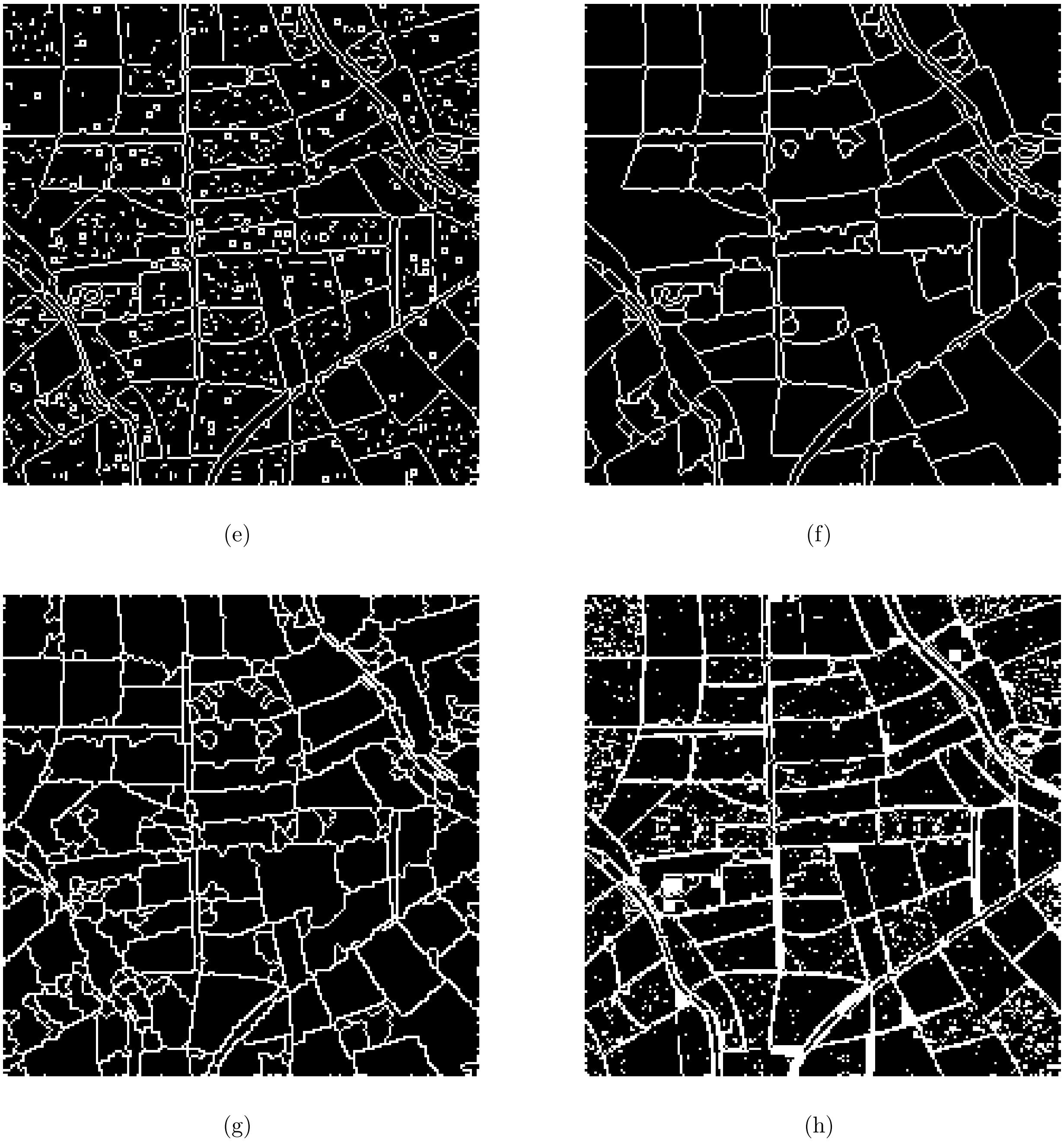


Figure 5.9: *(continued) (e) Results of contour following applied to (c). (f) Results of edge linking&pruning applied to (e). (g) Results of edge linking&pruning applied to (c) directly. (h) Results of combining region growing (t=4, s=28, see (b)) with edge detection (t=195), contour following, and edge linking&pruning.*

method	threshold	percentage correct	specificity (%)	sensitivity (%)
RG	–	87.4	93.9	80.9
RG & ED	240	87.6 (58.5)	87.6 (93.4)	87.7 (23.7)
RG & ED/CF	230	87.7 (59.1)	87.6 (93.4)	87.8 (24.7)
RG & ED/LP	250	87.4 (50.0)	93.8 (99.9)	80.9 (0.1)
RG & ED/CF/LP	195	88.0 (60.4)	88.2 (94.0)	87.8 (26.7)

Table 5.1: *Results of region growing (RG) with a threshold of 4 and a maximum region size of 28, combined with edge detection (ED). The performance of edge detection alone is given in parentheses. Options: CF=contour following, LP=edge linking&pruning.*

5.2.4 Results of a hybrid approach

In the last two sections, both region growing and edge detection proved to be valuable techniques for the detection of mixed pixels. In this section, we try to improve the performance of the most promising approach, region growing, by combining it with the possibly complementary information acquired via edge detection. As was already mentioned in Section 5.2.1, this is done by labelling a pixel as mixed if it was given this label by either region growing or edge detection (or both), i.e. by applying the OR-operator to Figures 5.9(b) and 5.9(c) for instance. This approach can only boost the sensitivity; the specificity will at best be equal to the lowest value achieved by either of the two underlying methods.

As far as region growing is concerned, only the best two images were used, corresponding to a SOIM threshold of 3 or 4 and a maximum region size of 28 (see Section 5.2.2). These images were combined with the results of edge detection using a wide range of threshold settings and applying none, one, or both of the options—contour following and linking&pruning—discussed in Section 5.2.3. It turned out that the results of region growing with the threshold set at 3 could not be improved. This was mainly due to the fact that the sensitivity was already high (88.5%) and could not be increased without allowing a larger decrease in the specificity. The results of combining edge detection and region growing with a SOIM threshold of 4 are presented in Table 5.1. With an optimal setting of the Frei-Chen threshold, a small improvement in the average percentage correct was achieved except when edge detection was directly followed by linking&pruning. In this last case the optimal threshold for edge detection was approximately equal to the maximum value of 255, which provides no additional information since all pixels are given the label pure. In all other cases the sensitivity increased with 7%, while the specificity decreased with a slightly smaller percentage. The thresholds with which these results were reached were higher than the optimal settings found in Section 5.2.3. As a consequence, the specificity of edge detection was about as high as that of region growing (93.9%), which prevented the specificity from becoming too low when the two approaches were combined. The best performance was achieved by combining region growing with edge detection, contour following, and linking&pruning, giving an average percentage correct of 88.0% and a specificity/sensitivity of 88.2%/87.8%. Figure 5.9(h) gives a graphical representation of these results; by finding the subtle differences between this figure and Figure 5.9(b) and matching those with the true segmentation given by Figure 5.9(a), an idea of the beneficial influence of adding information derived using edge detection can be acquired.

5.2.5 Discussion

In the previous sections it was shown that segmentation techniques are able to detect mixed pixels with an accuracy of 85%–90%. It must be noted, however, that these results to some degree are influenced by the use of artificial imagery and standard software.

The success of the segmentation approach depends on a number of image characteristics. The most important aspect is the contrast between the agricultural fields. The more the average grey levels of two neighbouring fields differ, the better the boundary composed of mixed pixels can be located. Another factor is the amount of within-field variation compared to the contrast between fields. If the within-field variation is relatively large, many false edge fragments and isolated mixed pixels will be found. The average field size, finally, also plays an important role when region growing is applied. Since large fields in general lead to the recognition of large pure pixel clusters, regions of mixed pixels, which usually are small, can be detected more easily. By generating an artificial image, we were able to fulfill all these conditions. The resulting image, however, is not unrealistic, except for the fact that neighbouring fields were forced to have different cover types. This last restriction, however, is not very serious since the detection of mixed pixels lying on the boundary of two fields belonging to the same class is of little importance. If these pixels are found to be pure when determining endmember spectra, little is lost because the pixels in fact are nearly pure. If the same error is made when estimating the area of individual fields, the accuracy is hardly affected either since these pixels cannot reliably be decomposed on a field basis anyway. It is for these reasons that we feel that the positive influence of using artificial images on the overall performance is limited and that the results can well be used to assess the suitability of segmentation techniques for mixed pixel detection.

With regard to the segmentation methods applied, several remarks can be made. Region growing implemented by the SOIM algorithm proved to be superior to edge detection using modified Frei-Chen templates. Not only was the average percentage correct about 10% higher, the former approach also groups pure pixels, which can be useful when determining endmember spectra afterwards. This grouping is also possible after edge detection by creating closed polygons and labelling pure pixels using a floodfill algorithm, but since the edges marking field boundaries often show some gaps, this is not very practical. The performance of the edge detection approach, however, does appear to be susceptible of improvement. On the one side, the options contour following and edge linking&pruning, which were developed specifically to segment large satellite images, can be optimised for the detection of mixed pixels. A major increase in performance, for instance, can be achieved by exact alignment of the 1-pixel wide contours with the field boundaries. On the other side, the processing of isolated mixed pixels provides a number of opportunities for improvement as well. Besides the introduction of additional templates matching ideal points, the noisy behaviour due to within-field variation could be reduced by subjecting possible edge pixels that are isolated to a different, higher threshold to determine their label. Another (ad hoc) post-processing option is to find pixels with a probably incorrect label, for instance by locating "window" structures like those displayed in Figure 5.9(d), and change them accordingly. A promising alternative to either region growing or edge detection seems to be a hybrid approach. First, pure pixel clusters are located by applying region growing with a high threshold, which avoids the noisy pattern due to within-field variation. Subsequently, mixed pixels that erroneously were included in these clusters can be removed using the information acquired with edge detection. The result is a more accurate segmentation where, as a bonus, pure pixels are grouped as well. By tuning edge detection to detect mixed pixels, the performance of this combined approach is bound to become even better.

5.3 Summary and conclusions

In this chapter we investigated whether or not segmentation techniques can be used to detect mixed pixels in agricultural scenes. This approach is based on the model of an agricultural field as a homogeneous cluster of pure pixels surrounded by a layer of mixed pixels. Region growing attempts to find these pure pixel clusters by grouping neighbouring, spectrally similar pixels. Edge detection, on the other hand, looks for local discontinuities in order to locate the layer of mixed pixels. A third way to detect mixed pixels is map-based segmentation, but for this approach ancillary information (the map) is required.

To determine the potential of edge detection as well as region growing for the detection of mixed pixels, we carried out a comparative experiment on an artificial though realistic image. The main reason for generating an artificial image was that the ground truth information necessary for a quantitative analysis of the detector's performance could be obtained simultaneously, but this way it could also be guaranteed that the contrast between fields was large relative to the within-field variation, which is a prerequisite for achieving a good segmentation. On this image region growing gave a good performance, labelling approximately 87% of both mixed and pure pixels correctly. As expected, the optimal settings of the two parameters, the SOIM threshold and the maximum size of regions containing mixed pixels, showed a positive correlation. However, since the performance proved to depend much more on a good setting of the threshold than on an optimal setting of the region size, finding the optimum is relatively easy. Edge detection performed less well, which resulted in an average percentage correct of 78%. In addition, pure pixels are not grouped automatically like region growing does, which may be a disadvantage when endmember spectra are to be determined. The main reason for the worse performance is that a lot of false edge pixels due to within-field variation are detected. Furthermore, the implementation of edge detection used was not very well suited to handle isolated mixed pixels. However, a lot of possibilities to improve the performance exist, which may make edge detection as successful as region growing. The best option to achieve an optimal detection of mixed pixels is to combine both approaches. Despite the fact that the results of edge detection were not particularly good, combining them with the results of region growing led to an average percentage correct of 88%, which is slightly higher than optimal performance achieved with region growing alone. The fact that edge detection is dual to region growing apparently leads to complementary information, which can be used to compensate for the flaws region growing. This, in combination with the fact that pure pixels are grouped as well, makes that the hybrid approach probably is the best method to detect mixed pixels when used for applications such as calculation of endmember spectra and area estimation of agricultural fields.

Chapter 6

Data-driven decomposition and its application to artificial images*

Data-driven decomposition (DDD) is a new method introduced in this thesis to make an accurate estimation of the area of an agricultural field based on two ideas. The first thought is that mixed pixels should be split into their composing parts (pure pixels may still be classified). The second idea is that knowledge of the application domain can be used to improve the accuracy of the crop area estimation. In Section 6.1 it is explained how DDD works. Section 6.2 describes an experiment in which the method is evaluated quantitatively based on artificial images. Section 6.3 gives a brief summary and presents the conclusions.

6.1 Data-driven decomposition

In most studies where the crop area is to be estimated, each pixel is classified or decomposed individually without considering its context, using class or endmember distributions derived from training data. Adopting the terminology of Rosin [87], we call this a *model-driven* approach. A *data-driven* approach, on the other hand, derives local (class or endmember) distributions directly from the image data, using spatial information based on knowledge of the application domain. Data-driven decomposition follows the latter approach to implement classic linear mixture modelling (Section 2.1.1, analytic approximation), which was selected because of its relatively good performance combined with low computational costs. Section 6.1.1 shows how DDD uses contextual information related to agricultural fields to "drive" the decomposition process. Section 6.1.2 describes how knowledge of objects other than agricultural fields is incorporated to obtain even more accurate area estimates.

6.1.1 Data-driven aspects

The accuracy of the classic linear mixture modelling estimator can be increased by incorporating some knowledge of the application domain. So far, most studies have used no domain-specific knowledge whatsoever, but have employed the estimator to decompose all image pixels, assuming the same endmember distributions for the entire image, and using

*This chapter is based on publications [35] and [38].

all available endmembers at the same time. Data-driven decomposition, on the other hand, makes use of known spatial relationships between the pixels to locate the mixed pixels that need to be split, to determine local endmember distributions from the image itself, and to select only those endmembers that are probable components of a mixed pixel.

A first way to obtain a more accurate area estimation is to separate the pure pixels from the mixed pixels by making a segmentation of the image. As was described in Chapter 5, any agricultural field that is relatively large given the resolution of the scanner can be regarded as an inner cluster of pure pixels surrounded by a layer of mixed pixels at the field boundaries. Since we know that our images depict agricultural fields, we can discriminate between pure and mixed pixels by locating the pure pixel clusters. These clusters can sometimes be found with the help of ancillary information sources such as topographical maps, see Section 5.1.3, but it is also possible to segment the image using standard image processing techniques. Schoenmakers [90], for instance, developed a hybrid segmentation method that combines region growing (Section 5.1.1) and edge detection (Section 5.1.2) to find pure pixel regions in large optical satellite images. Once the pure pixels belonging to a region, which is considered to be the same as a pure pixel cluster from now on, have been identified, they can be classified individually or as part of the covering fields to achieve a higher accuracy (Janssen [54]). Most remaining pixels are probably mixed and should be decomposed rather than classified. Thus, the accuracy of a crop area estimate can be increased by segmenting the image in order to classify pure and decompose mixed pixels.

Data-driven decomposition is also more accurate than the classic decomposition approach because of the way in which it defines the endmember distributions. Traditionally, the statistical distribution of every ground cover type that was to be distinguished was determined using training pixels, laboratory measurements, or feature space analysis (see Section 4.2). However, since we know that our scene consists of agricultural fields that can be modelled as pure pixel clusters with mixed pixels in between, we can determine more accurate endmember distributions by calculating the statistical distribution of each region. The advantage of taking the distributions of fields instead of crops is that the former endmember distributions, which are in fact crop distributions as well, are better adapted to the local variations in ground slope, elevation, soil type, etc.; as a result, the estimation of the fractions of a mixed pixel will become more accurate too. However, the size of a field may be too small to estimate its distribution reliably. In this case, data-driven decomposition determines the crop type through maximum likelihood classification of the field's mean vector, and selects the corresponding crop distribution from a database with endmember distributions that were determined in a traditional manner after all.

The third way in which data-driven decomposition tries to increase the accuracy of the crop area estimates is to use contextual information for selecting probable endmembers of a mixed pixel. The classic decomposition approach, which decomposes a mixed pixel into all endmembers at the same time, has two drawbacks. The first is that the decomposition is usually sub-optimal because fractions are allocated to incorrect endmembers, and the second is that the number of ground cover types that can be distinguished in the image is severely limited by the number of spectral bands because of the way the linear mixture model is solved. The latter limitation in particular is a serious problem for data-driven decomposition as it regards every agricultural field as a possible endmember. However, by restricting the large set of possible endmembers to a small set of probable endmem-

```
for (each possibly mixed pixel) {
    determine the regions from its 8-connected pure pixels;
    construct a set containing all pairs of two regions;

    for (each element of the set)
        calculate the unreliability of decomposition using this combination;

    if (the lowest unreliability is below the threshold)
        divide the pixel according to the corresponding combination;
    else mark the pixel;
}
```

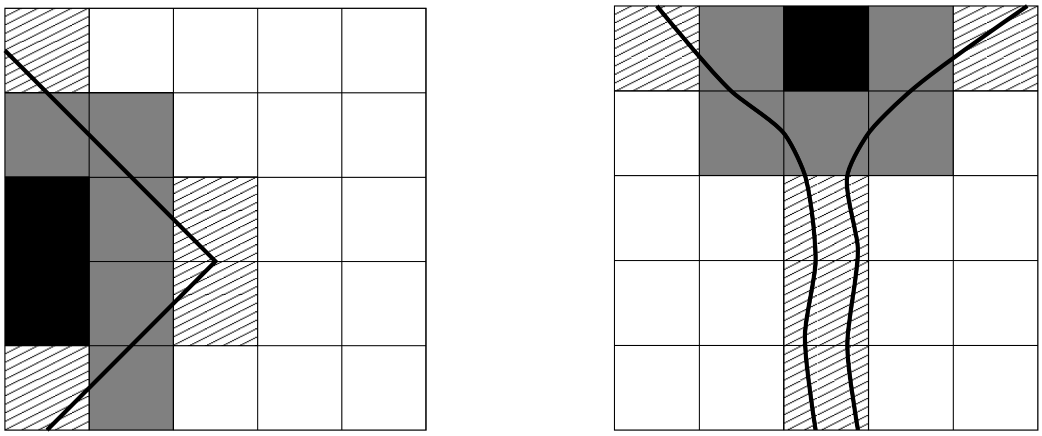
Algorithm 6.1: *First stage of DDD based on the simple scene model.*

bers, which can be different for each pixel, an accurate decomposition using a virtually unlimited number of endmembers can be achieved. According to our simple scene model, the components of a mixed pixel are the agricultural fields that surround it; therefore, the endmembers that are needed to decompose a mixed pixel can be found by searching its neighbourhood for pure pixel clusters. An implementation of this view is given by Algorithm 6.1, which is written in C-like pseudo code. In this algorithm, the smallest possible symmetrical neighbourhood—a $3x3$ window centred at the mixed pixel—is examined to determine the pure pixel clusters. This way only regions containing one or more of the eight pixels directly connected to the mixed pixel are considered, which ensures that these regions remain contiguous even after assignation of mixed pixels. Furthermore, only a small fraction of all possible combinations are tried in order to save time as well as to prevent unlikely class mixtures from being chosen. For instance, combinations of three or more surrounding regions could be investigated, but they would give only marginal improvement because few such pixels exist, and probably only the classes that contribute little to the pixel are wrongly estimated. The (un)reliability measure that is used to distinguish between the different combinations depends on the decomposition method selected. As we selected the classic linear mixture modelling approach, the maximum likelihood criterion defined by Equation (2.8) is an obvious choice[1]. Since this measure equals the Mahalanobis distance between the pixel and its approximation by the decomposition algorithm, a rough indication of a reasonable threshold setting can be derived. Suppose that the classes the mixed pixel is to be decomposed into show no correlation between the different spectral bands. If for each band a deviation of say 2σ is allowed, the M-distance is equal to:

$$\begin{bmatrix} 2\sigma_1 & 2\sigma_2 & \dots & 2\sigma_n \end{bmatrix} \begin{bmatrix} \sigma_1^{-2} & & & 0 \\ & \sigma_2^{-2} & & \\ & & \ddots & \\ 0 & & & \sigma_n^{-2} \end{bmatrix} \begin{bmatrix} 2\sigma_1 \\ 2\sigma_2 \\ \vdots \\ 2\sigma_n \end{bmatrix} = 4n. \tag{6.1}$$

This value is different in case the spectral bands are correlated of course, but a threshold of

[1]Note that $\mathbf{N}(\mathbf{f})$ is substituted by the average of the $\mathbf{N}_i$, which is different for each combination.



(a) (b)

Figure 6.1: *The white and black squares are pure pixels belonging to different fields, the grey and striped squares represent mixed pixels. The fields contributing to the striped pixels cannot be determined by looking at the pure pixels in a* $3x3$ *window.*

four times the number of bands seems to be a reasonable setting. Using Algorithm 6.1 with a threshold setting of $4n$, the true endmembers of most mixed pixels can be determined.

The endmember selection mechanism presented in the previous paragraph can be improved by deferring the decomposition of the more difficult pixels to a later stage. Figure 6.1 depicts two situations in which Algorithm 6.1 fails to find the correct endmembers. In both cases the $3x3$ neighbourhoods of the striped mixed pixels consist of some other mixed pixels and some pure pixels, but they do not contain all contributing fields at the same time. Figure 6.1(a) shows the common problem that mixed pixels hardly ever occur isolated but nearly always appear clustered. Expansion of the neighbourhood with another or even two layers of pixels could solve the problem, but it could also result in fields that are no longer contiguous. For objects like the one in Figure 6.1(b), which represents a river[2] flowing into a lake, this strategy is not applicable since the necessary expansion of the neighbourhood could be arbitrarily large. A much better solution to these problems is given by Algorithm 6.2, which can be considered as the second stage of an overall algorithm started by Algorithm 6.1. In the second stage, the spatial information of the mixed pixels which were processed in the first stage is used to find fields that were not considered before because the neighbourhood did not contain pure pixels of those regions. The second stage is executed iteratively, using new spatial information obtained in the previous iteration. An advantage of this strategy is that the contingency of each field is guaranteed since there always exists a path from the mixed pixel to a pure pixel of each contributing region. Another advantage is that narrow structures can now also be handled without the need to expand the neighbourhood enormously. Although the iterative nature of Algorithm 6.2 may lead to a significant increase in computation time in theory, this proved to be not the case in practice. Not only was the vast majority of the mixed pixels handled by the

[2]Although this is not an agricultural field, we will consider these objects as well; see Section 6.1.2.

```
while (previous iteration decomposed at least one pixel)
    for (each pixel marked) {
        determine new regions from its 8-connected pixels decomposed in last iteration;
        construct a set containing all pairs of two new regions;
        add all pairs of one new and one old region to the set;

        for (each element of the set)
            calculate the unreliability of decomposition using this combination;

        if (the lowest unreliability is below the threshold)
            divide the pixel according to the corresponding combination;
        else
            mark the pixel for the next iteration;
    }
```

Algorithm 6.2: *Second stage of DDD based on the simple scene model. "New" regions have not been discovered before, while "old" regions were found in a previous iteration.*

first stage, most remaining mixed pixels were of the type shown in Figure 6.1(a) and could nearly all be dealt with in the first iteration. Pixels belonging to narrow structures may need more iterations—Figure 6.1(b) needs three—depending on the position of the closest pure pixel in the corresponding cluster, but they are small in number.

In summary, the data-driven decomposition method overcomes three important weaknesses of the classic decomposition approach. The first improvement is that, instead of decomposing all pixels, a discrimination between pure and mixed pixels is made in order to handle them by classification and decomposition, respectively. The second enhancement is achieved by taking local endmembers, whose distributions are well adapted to local crop variations, instead of endmembers that are global to the entire image. The third weakness that is resolved is the necessity of decomposing a mixed pixel into all endmembers simultaneously; our method is able to predict the probable components of a mixed pixel prior to its decomposition. All these improvements make use of the contextual information included in the image. By formalising some knowledge of the application domain into a simple scene model, the spatial relationships between the pixels can be used to make a more accurate estimation of the area of an agricultural field.

6.1.2 Further improvements based on domain knowledge

The accuracy of the data-driven decomposition method can be increased further by incorporating additional knowledge of the application domain in our scene model. So far, a scene was modelled as a collection of adjacent agricultural fields. This simple model, however, is not very accurate since in reality agricultural fields are often bounded by lanes, ditches, strips of grass, etc. Furthermore, usually the scene also contains a number of small objects such as farmhouses, ponds, and groves. Therefore, the scene model is extended with two object types, called "narrow boundary structures" and "isolated objects".

Narrow boundary structures require a different approach than agricultural fields in order to process them accurately. Given that the typical ground resolution of a satellite scanner is 20–30 m, a narrow boundary structure of less than 20 m wide will be depicted as an elongated series of mixed pixels that is one or two pixels wide. Since such a structure does not have a pure pixel cluster, it is impossible to derive its statistical distribution directly from the image as can be done for agricultural fields. Instead, we have to fall back on more traditional techniques, for instance based on training pixels, to determine a few endmember distributions which approximate the distributions of that type of structure. Some examples of endmembers which should be determined because they are components of some well-known boundary structures are: asphalt (roads), soil (uncultivated land and dirt roads), grass (uncultivated land), deciduous forest (tree rows), and water (ditches and canals). Another difficulty due to the lack of a pure pixel cluster is that the presence of a boundary structure cannot be detected, which prevents the endmember selection mechanism introduced in Section 6.1.1 from finding all components of a mixed pixel. Again, the answer is to revert to the classic approach by including all endmembers typical of narrow boundary structures when decomposing a mixed pixel. The problems that arise from this large number of endmembers—infinitely many solutions of the linear mixture model and an inaccurate decomposition—can be solved with some domain-specific knowledge. According to this knowledge, nearly all mixed pixels are formed by one of three possible combinations: two agricultural fields; two agricultural fields and one narrow boundary structure; or one agricultural field and one narrow boundary structure, in case the structure lies in between two pixels. Since it can be derived which agricultural fields are probable components of a particular mixed pixel, and a set of endmembers that covers all types of boundary structures is available, a list of every possible composition can be made. To determine the optimal composition, we once again select the alternative that has the maximum likelihood, i.e. the combination that results in lowest value of Equation (2.8). An implementation of this approach is given by stages 1 and 2 of Algorithm 6.3 [3]. Thus, with a few changes to the DDD algorithm, narrow boundary structures can be processed accurately.

Isolated objects are processed in roughly the same way as narrow boundary structures. Since isolated objects are small not only in width but also in length, they will usually appear as isolated mixed pixels in the image. Just like narrow boundary structures, an isolated object does not result in a pure pixel cluster, which is needed to determine its statistical distribution and to mark it as a probable endmember. Therefore, the distributions of those endmembers that are characteristic of this type of object have to be determined with traditional techniques. The endmember selection strategy that was devised to handle boundary structures, however, should not be used for isolated objects because their variety is too large and their number is too small. As a result of their large variety, many distributions are needed to describe all possible objects, which slows down the decomposition process considerably. More important, though, is that many mixed pixels will be decomposed less accurately because fractions can be assigned to a large number of additional endmembers that are characteristic of isolated objects only, while the probability that the mixed pixel indeed represents an isolated object is small. To solve these problems, data-driven decomposition sets apart all mixed pixels that probably represent isolated objects

[3] Stages 1 and 2 could be merged into one loop, but this is not done for reasons of clarity.

```
for (each possibly mixed pixel) { /* stage 1 */
    determine the regions from its 8-connected pure pixels;
    construct a set containing all pairs of two regions;
    add all pairs of one region and one edge class to the set;
    add all triplets of two regions and one edge class to the set;

    for (each element of the set)
        calculate the unreliability of decomposition using this combination;

    if (the lowest unreliability is below the threshold)
        divide the pixel according to the corresponding combination;
    else
        mark the pixel;
} /* end stage 1 */

while (previous iteration decomposed at least one pixel) /* stage 2 */
    for (each pixel marked) {
        determine new regions from its 8-connected pixels decomposed in last iteration;
        construct a set containing all pairs of two new regions;
        add all pairs of one new and one old region to the set;
        add all pairs of one new region and one edge class to the set;
        add all triplets of two new regions and one edge class to the set;
        add all triplets of one new region, one old region, and one edge class to the set;

        for (each element of the set)
            calculate the unreliability of decomposition using this combination;

        if (the lowest unreliability is below the threshold)
            divide the pixel according to the corresponding combination;
        else
            mark the pixel for the next iteration;
    } /* end stage 2 */

for (each pixel marked) { /* stage 3 */
    determine the regions from all its 8-connected pixels;
    construct a set containing all pairs of one region and one arbitrary class;

    for (each element of the set)
        calculate the unreliability of decomposition using this combination;

    divide the pixel according to the combination with the lowest unreliability;
} /* end stage 3 */
```

Algorithm 6.3: *All three stages of DDD based on the extended scene model.*

and decomposes them in a separate stage—see stage 3 of Algorithm 6.3. The criterion that is used to make this decision is based on the same reliability measure used before: a mixed pixel is regarded as an isolated object only if the (maximum) likelihood of its decomposition into the usual endmembers is too low. The endmember selection strategy that is followed in this case makes use of the knowledge that most of these pixels are a mixture of the isolated object and one of the agricultural fields it is surrounded by. By trying all combinations of one endmember typical of isolated objects and one agricultural field that is located in the pixel's neighbourhood, the optimal decomposition can be found by selecting the solution that has the highest probability. Thus, in a way that is very similar to the way narrow boundary structures are handled, isolated objects are processed efficiently by data-driven decomposition.

In conclusion, the accuracy of the classic decomposition approach can be increased by using the available knowledge of the application domain. Besides our knowledge of agricultural fields, which led to the exploitation of the contextual information in the image, we can also make use of our knowledge of other existing objects. By extending the simple scene model of Section 6.1.1 with narrow boundary structures and isolated objects, much greater realism is introduced. As a result, the accuracy of the area estimation of an agricultural field by data-driven decomposition will be improved further.

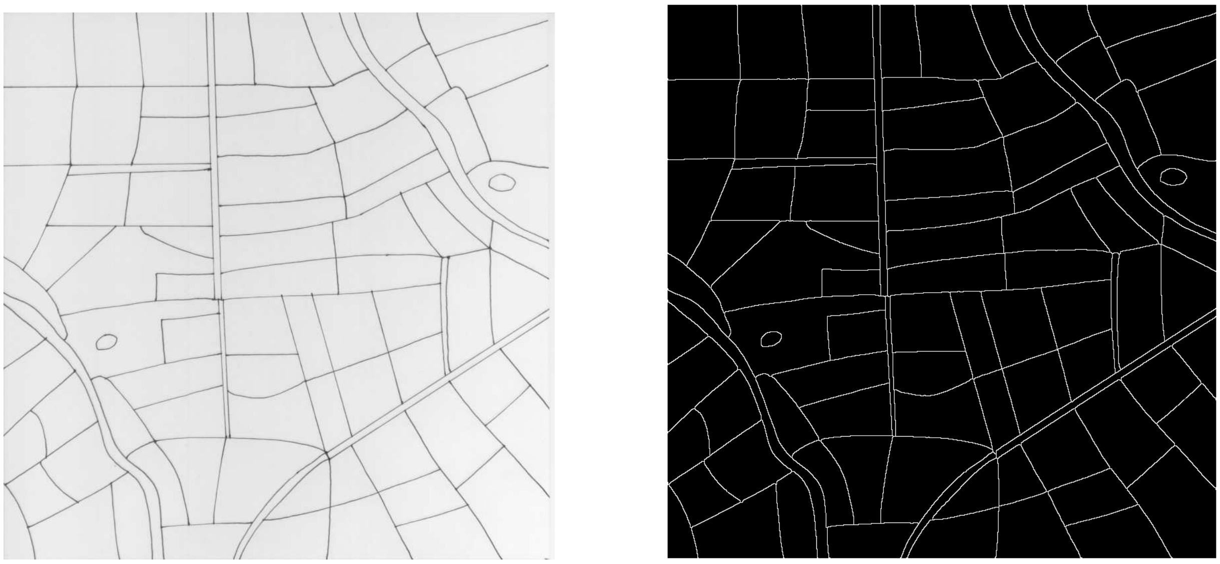
6.2 Application to artificial images

To evaluate the data-driven decomposition method, it was applied to several simulated Landsat-TM images of agricultural scenes. The reason for using artificial instead of real images was that for the latter data no ground truth at subpixel level was available, which is needed for a quantitative analysis of the method. For the simulated images this information is available because of the way they are constructed—see Section 6.2.1. As is described in Section 6.2.2, DDD was applied to images of increasing complexity while monitoring its performance. Based on the results of these experiments, which are discussed in Section 6.2.3, the strong and weak points of the DDD algorithm could be determined.

6.2.1 Simulation of remote sensing images

The generation of artificial satellite images can be done in many ways with varying degrees of realism. A method which preserves both first- and second-order statistics has been described by Schouten and Klein Gebbinck [92] and is briefly summarised in this section.

The first step in generating an artificial image is to draw the boundaries of objects such as agricultural fields, roads, and ponds on a white piece of paper using a black pen. Figure 6.2(a) shows an example, which we used as the basis of our own images. The next step is to digitise the image using an optical scanner, which we set at a modest resolution of 100 dpi and 256 grey levels. After a thresholding operation to produce a binary image, 1-pixel gaps in the edges are filled and the resulting boundaries are thinned to obtain 1-pixel wide closed polygons—see Figure 6.2(b). Next, a standard blob colouring algorithm is applied to label each non-boundary pixel with a number identifying the object it is part of. With this information, a mapping between the objects and the available ground cover types



(a) (b)

Figure 6.2: *First products in the simulation chain. (a) Hand-drawn image scanned at 100 dpi and 256 grey levels. (b) Field boundaries represented as 1-pixel wide closed polygons.*

is made. The boundary or edge pixels can be related to a ground cover type—a so-called *edge class*—as well, thus implementing the concept of narrow boundary structures.

Before we can proceed with the next step of the simulation protocol, we first have to generate the so-called class templates. The idea behind these templates is that each class is represented by a small image of say $100x100$ pixels, which not only captures the distribution of the class—i.e. mean vector and variance-covariance matrix—but also its texture. In our case the templates were derived from a Landsat-TM image of Portugal. First, each pixel is classified using a nearest neighbour algorithm based on a set of supervised training pixels. From the resulting classified image, numerous blocks of $5x5$ pixels having the same label are extracted per class; obviously, these blocks will all be located inside the boundaries of real objects (mainly agricultural fields). With these blocks the templates are constructed using a strategy that places the most common blocks in the middle and arranges the others around them such that the transitions between the blocks are as smooth as possible. Furthermore, several selection criteria are applied to exclude those blocks that deviate too much from the standard; see Schouten and Klein Gebbinck [92] for a detailed description of these exceptions. In the resulting templates the texture of the original Landsat-TM image is preserved very well.

The last steps in the simulation protocol are reducing the resolution of the hand-drawn image and calculating the intensity of the resulting pixels. The spatial resolution is reduced simply by dividing the processed hand-drawn image into blocks of pxp pixels (p=4 in our experiments). Each block is used to create one pixel of the artificial image, which thus may cover a number of classes. The intensity values of each artificial pixel is taken as a linear mixture of the intensities of the classes it is composed of, using weights that are proportional to the number of subpixels covered by each class. In vector notation this approach translates to

$$\mathbf{x}^{(i,j)} = \sum_{k=1}^{c} f_k^{(i,j)}\ \mathbf{t}_k^{(i \bmod s, j \bmod s)}, \tag{6.2}$$

where $\mathbf{x}$ is the pixel vector that is generated for position (i, j), $f_k^{(i,j)}$ denotes the proportion of class k, and $\mathbf{t}_k^{(u,v)}$ represents the pixel vector at the corresponding location of the k-th class template. Since the size s of the (square) templates usually is smaller than the size of the artificial image, the modulo operator is used to make sure the index into the class templates stays within bounds. Discontinuities in the grey level distribution of adjacent pixels that might result as the operator passes through 0 are prevented by extending the templates with mirrored copies in both dimensions. With the above method, the textures of the original Landsat-TM image that were preserved in the class templates are preserved in the artificial image as well.

Optionally, the artificial image can be adapted to incorporate some isolated objects. For this purpose, a number of pixels chosen at random are replaced by isolated mixed pixels. These mixed pixels are assumed to consist of one ground cover type of a nearby field and one totally different ground cover type, where the latter class is given a fraction of at least 75%. The intensity of the resulting pixel is determined in the same manner as explained before. By applying the method described in this section, a realistic artificial image is generated that is comprised of agricultural fields, narrow boundary structures, and isolated objects.

6.2.2 The experimental set-up

In the experiments, three images of increasing complexity were processed by the data-driven decomposition algorithm. In the first image the edges were ignored, which led to simple transitions between the neighbouring regions. In the second image the edges were assigned the ground cover type bare soil, which gave mixed pixels that were more complex. In the third image a number of isolated mixed pixels were added in order to determine the ability to detect these pixels and to evaluate the threshold setting. During the decomposition process, the performance of DDD was monitored using two measures.

The first metric that was calculated measured the accuracy of the decomposition. This measure had to reflect the errors that could be made, which was in the first place that the estimated fraction of a region deviated from the true fraction, and secondly that the chosen region itself was incorrect. However, it was much more practical to calculate the accuracy based on ground cover types than on region numbers. Suppose for instance that two regions with the same ground cover type contribute to the same mixed pixel. While the fractions of the individual regions can never be recovered, calculating the total proportion of that ground cover type is well possible. However, this does require that the ground cover type of each region is determined, which we did through maximum likelihood classification of the region's mean vector (see Appendix A). The above problem could also be circumvented by assigning different ground cover types to neighbouring regions and using edge classes differing from all regions in the entire image, but a measure based on ground cover types was found to be more flexible. Furthermore, with the latter approach the two error types mentioned earlier could be unified: a mixed pixel was considered to be composed of all ground cover types present in the image, taking fractions of classes that did not belong to a pixel's composition equal to zero. The accuracy measure that was calculated for each pixel is Minkowski's L1-metric scaled by $\frac{1}{2}$:

$$e_{acc} = \frac{1}{2} \sum_{i=1}^{c} \left| \hat{f}_i - f_i \right|, \tag{6.3}$$

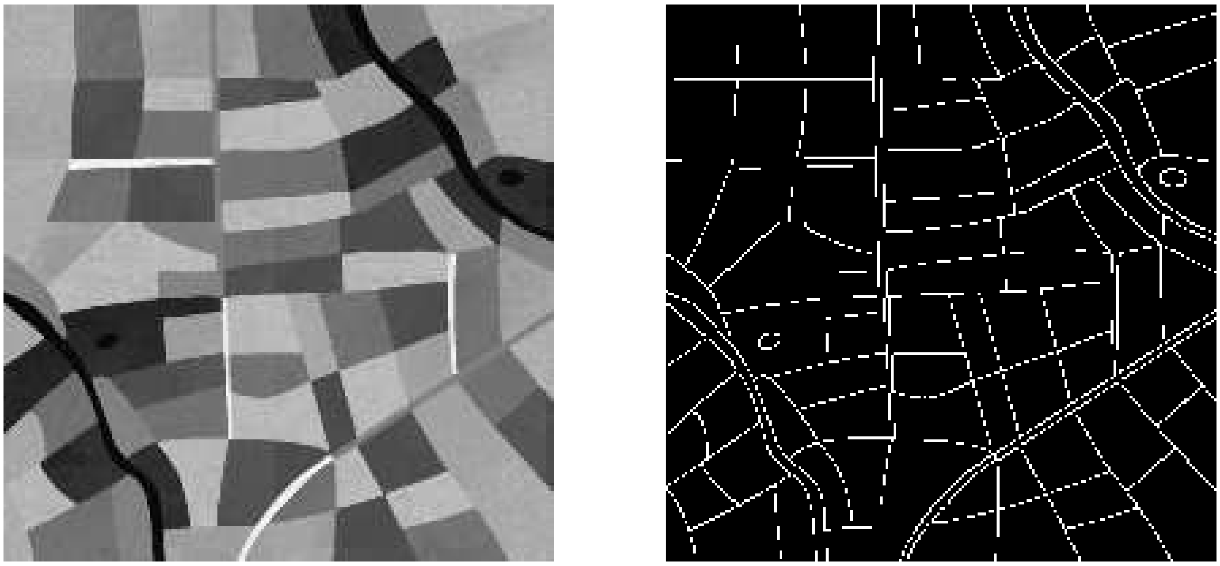
where $\hat{\mathbf{f}}$ and $\mathbf{f}$ represent the estimated and real fractions vector, respectively. This expression is identical to Equation 3.22 except that $\varepsilon_f = \langle e_{acc} \rangle$, i.e. ε_f is the average e_{acc} of all pixels. The errors in accuracy can be presented in a histogram to show the global statistics as well as in an image to view any spatial relations.

The other measure that was computed monitored the (un)reliability of each decomposition. This measure is directly related to the classic linear mixture modelling estimator in that it determines how well the estimator can reconstruct the pixel:

$$e_{rel} = (\mathbf{x} - \mathbf{M}\hat{\mathbf{f}})^T \mathbf{N}^{-1} (\mathbf{x} - \mathbf{M}\hat{\mathbf{f}}). \tag{6.4}$$

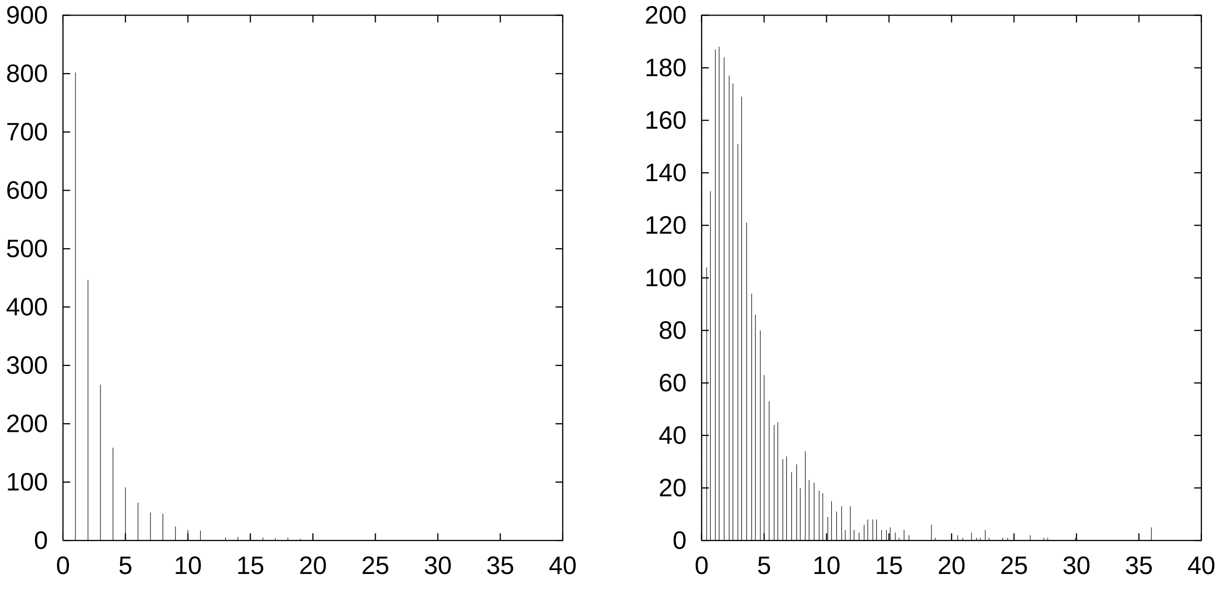
The higher the value of e_{rel}, the more unreliable the decomposition is assumed to be. As was described in Section 6.1.1, this expression is also used to select the probable endmembers of a mixed pixel, hence it is very useful to evaluate DDD. Analogous to the accuracy measure, the results of the reliability measure can be presented using both a histogram and an image. With these means it is possible to assess the threshold setting and to identify certain problem areas.

Last but not least, there are a few choices made during the design of the experiments that need to be mentioned. First of all, we decided to derive a perfect segmentation for each of the satellite images, which was easy because of the way they were generated. This decision was motivated by the fact that the negative influence of a bad segmentation on the performance of DDD would be quite considerable. Since an investigation into optimal segmentation methods was considered to be beyond the scope of this study, we used a perfect segmentation in order to guarantee a proper evaluation of DDD itself. The other decision that was made is that the performance of DDD would only be monitored on mixed pixels. The reasons for this decision were threefold. Firstly, in agricultural scenes the number of pure pixels compared to the number of mixed pixels is so overwhelming that the performance of DDD on the latter ones would hardly be noticeable, while the mixed pixels provide the best possibilities for improvement of the area estimates. Secondly, pure pixels are easily processed by most methods, therefore their performance is not very interesting. Thirdly, the perfect segmentation provides much more information on pure than on mixed pixels, for example the fact that their fractions vectors contain all zeroes except for one place, or which nearby pixels are pure as well and have the same ground cover type. To minimise the positive influence of using an artificial segmentation, therefore, we ignored the pure pixels when evaluating the performance of DDD.



(a) (b)

Figure 6.3: *Image I, in which the edges are ignored. (a) The most contrast-rich band 5. (b) Location of the mixed pixels, which are shown in white.*



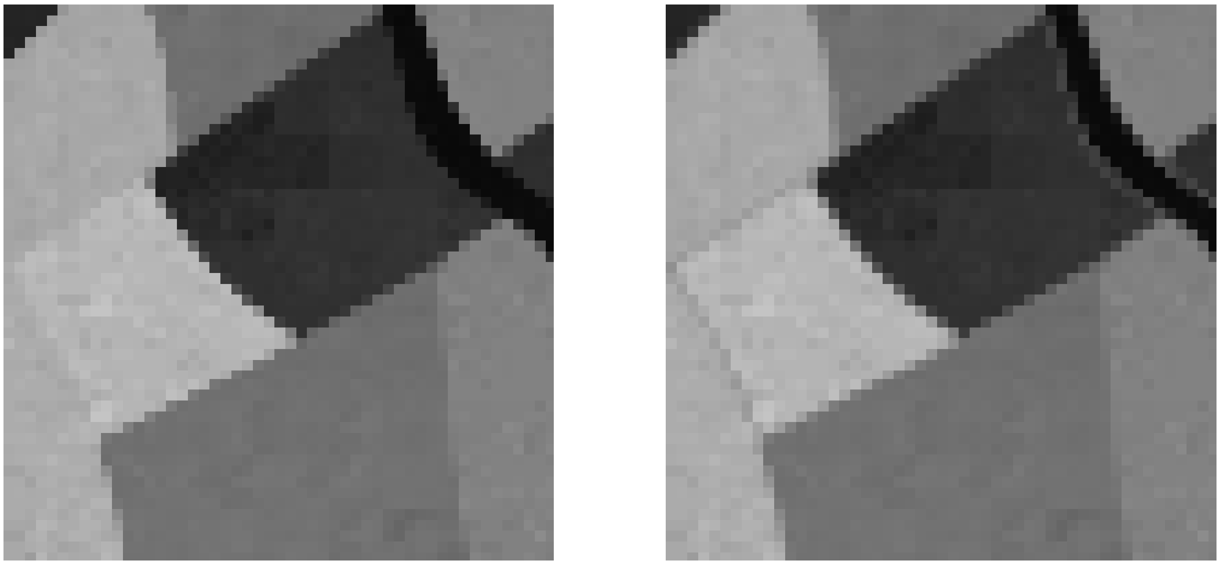
(a) Error in accuracy. (b) Error in reliability.

Figure 6.4: *Statistics collected when decomposing Image I. (a) Histogram of the number of pixels (vertical) vs. the error in accuracy (horizontal). The 40%–100% interval (not shown) contained 8 pixels. (b) Histogram of the reliability values calculated after stage 2. The last bin holds the number of pixels with an e_{rel} of 36.0 and more.*

6.2.3 Results and discussion

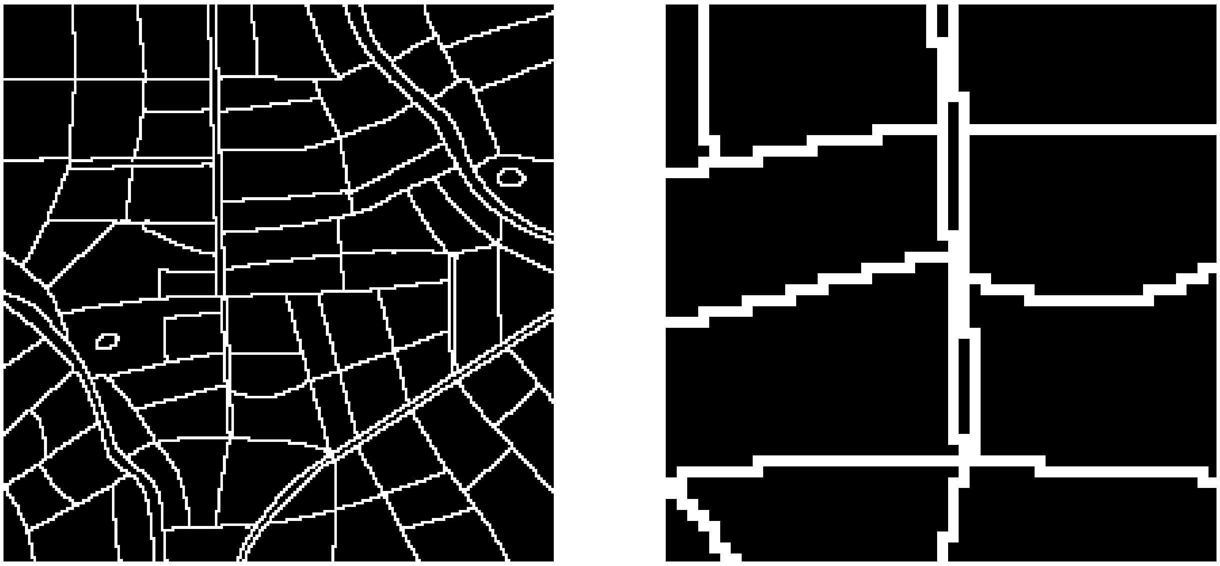
The three images on which the experiments were carried out comprised 203x205 pixels in six spectral bands. These pixels were constructed out of 4x4 subpixels, using 10 different ground cover types. In Image I no ground cover type was assigned to the edges, which resulted in 2479 mixed pixels that were relatively easy to decompose. Figure 6.3(a) displays band 5 of this image, while the positions of the mixed pixels are shown in Figure 6.3(b). Data-driven decomposition of the mixed pixels in Image I resulted in an average error in accuracy of 2.7%. The threshold was set at $4xn$—the number of bands (6)—i.e. 24.0. In stage 2 of the algorithm, two iterations were performed in which subsequently 1 and 0 mixed pixels were processed; 12 pixels were (wrongly) assumed to represent isolated objects. The majority of these pixels consisted of three or more regions, something the algorithm could not handle. Another source of inaccuracy was that some mixed pixels could be decomposed a little bit more reliably using a wrong combination of regions than using the correct combination. A last cause of error was that in a few cases the pixel could be decomposed correctly but with an unreliability that was just above the threshold. As a result, these pixels were subjected to the third stage, which was meant for truly isolated pixels only. Figure 6.4 shows how the errors in accuracy and reliability were distributed. Figure 6.4(b) confirms that a threshold setting of 24.0 was a good choice.

In Image II the edges were assigned the ground cover type soil, which resulted in an image very similar to Image I. The difference between these images can only be seen by zooming in on for instance the lower left corner, as is done in Figure 6.5. Not only did the composition of the mixed pixels become more complex, the number of mixed pixels also increased to 4928—compare Figures 6.3(b) and 6.6(a). As a result, Image II contained several narrow structures consisting largely of mixed pixels, see Figure 6.6(b), which were good test cases for the second stage of DDD. With a threshold setting of 24.0, the mixed pixels were decomposed with an average error in accuracy of 4.9%. For Image II a total of 5 iterations were needed during stage 2, in which subsequently 29, 14, 4, 4 and 0 mixed pixels were decomposed. Iterations 3 and 4 were required to process the narrow structure depicted in Figure 6.6(b), which contained a massive block of mixed pixels of height 10. A total of 5 mixed pixels remained after the second stage and were marked isolated, which in fact they were not. In addition to the reasons already mentioned in the previous paragraph, spectral confusion was another source of error. Several pixels being composed of rice (±90%) and soil (±10%) were estimated to have a much smaller fraction of rice (±60%), and more soil (±20%) and water (±20%). Much more frequent, however, was the confusion between soil, barley and grass. It appeared that the class mean of grass was a linear combination of the class means of soil and barley, leading to spectral confusion for all pixels containing both soil and barley. Since the variance-covariance of grass was higher than the ones of soil and barley, the area covered by grass was systematically overestimated while soil and barley were underestimated. In Figure 6.7 the histograms of the errors in accuracy and the unreliability are presented, while Figure 6.8 shows these measures for each pixel individually. The line fragments in Figure 6.8(a) (e.g. the diagonal line west of the centre) were located at the boundaries between fields of barley and grass. With edges of soil, spectral confusion occurred along the entire border. As can be seen in Figure 6.8(b), these pixels could not be identified based on a low reliability.



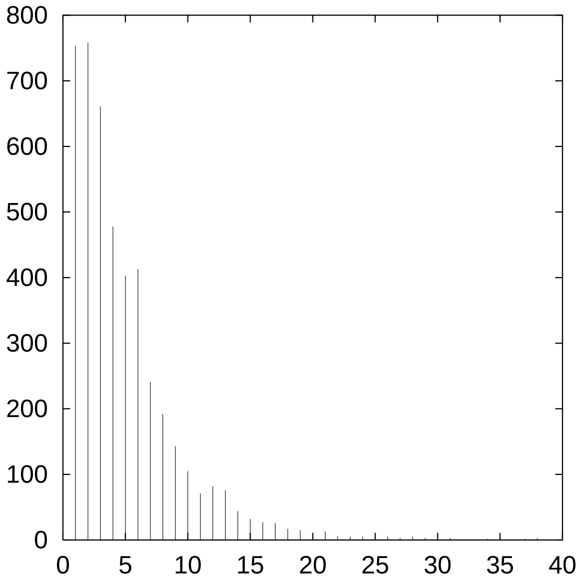
(a) (b)

Figure 6.5: *The difference between Image I and II can be found at the field boundaries. (a) Lower left corner of Image I. (b) Lower left corner of Image II.*

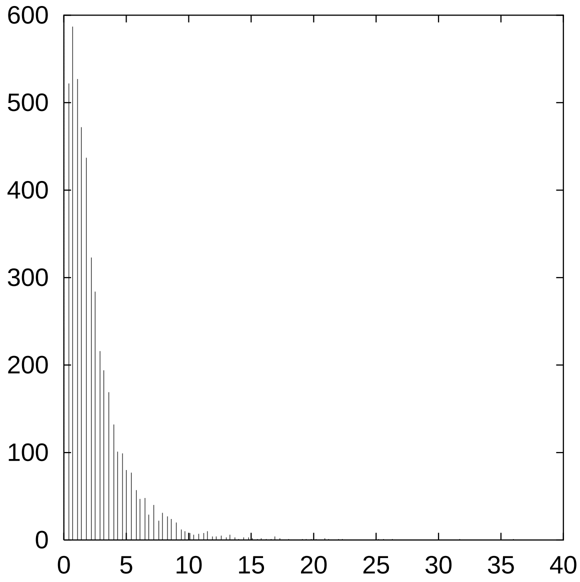


(a) (b)

Figure 6.6: *Image II with edges consisting of soil. (a) Positions of the mixed pixels shown in white. (b) Blow-up of the narrow structure in the south-west quadrant.*

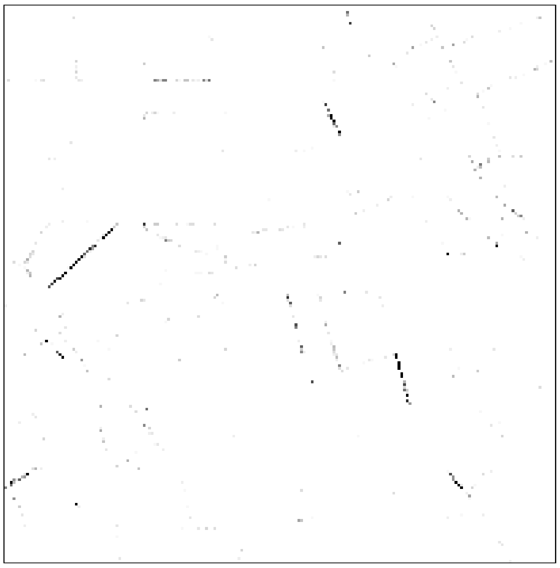


(a) Error in accuracy.

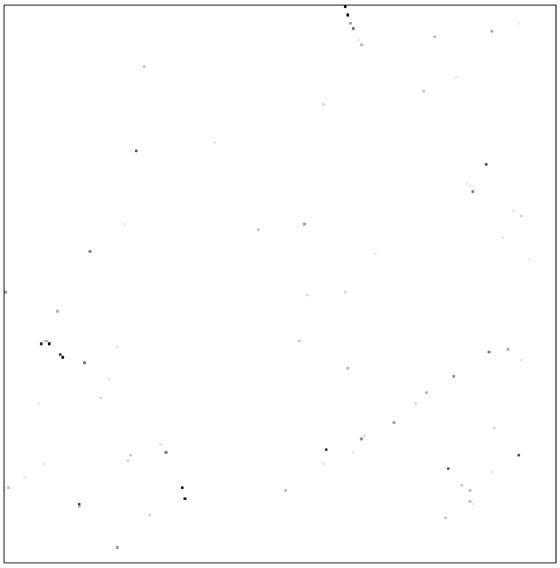


(b) Error in reliability.

Figure 6.7: *Statistics collected for Image II. (a) Histogram of the error in accuracy; the interval cut-off (40%–100%) contained 28 pixels. (b) Histogram showing the number of pixels with a certain (un)reliability calculated after stage 2.*

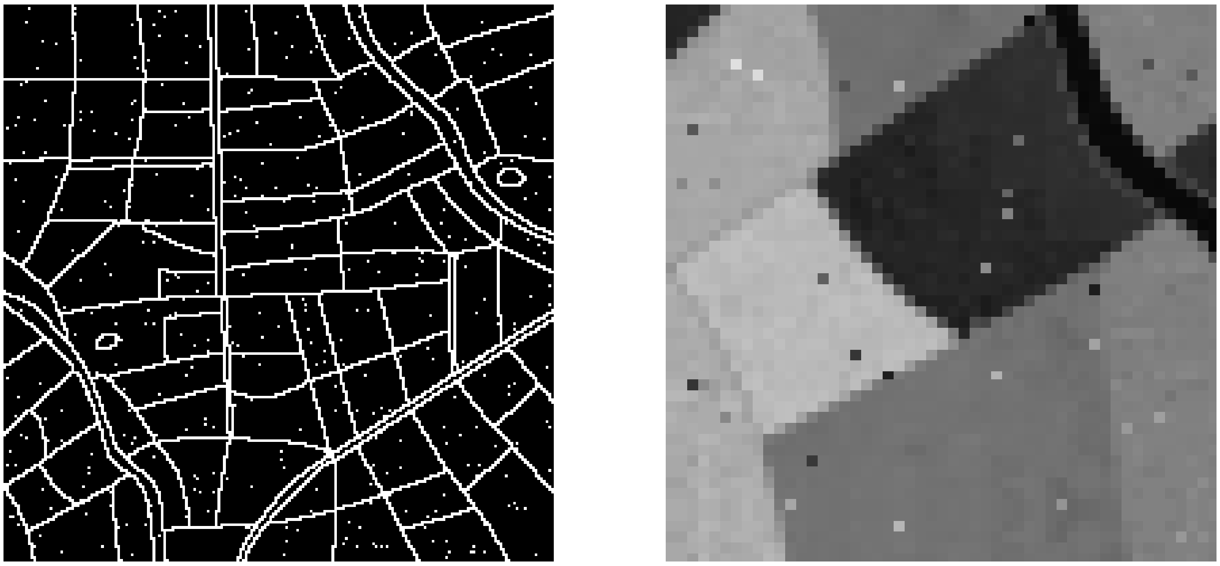


(a) Error in accuracy.



(b) Error in reliability.

Figure 6.8: *Spatial distributions of the performance measures on Image II. (a) Errors in accuracy represented as grey levels. Darker pixels have a higher inaccuracy. (b) Reliability of decomposition in stage 1 or 2; the less reliable the decomposition, the darker the pixel.*

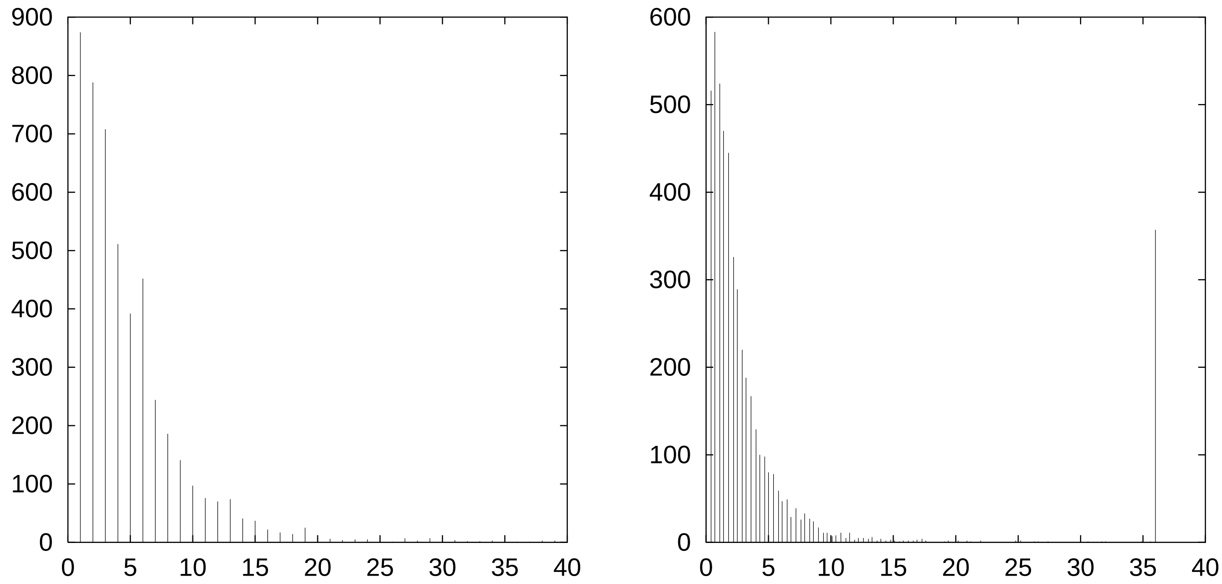


(a) (b)

Figure 6.9: *Image III, which was simulated to contain a number of isolated objects. (a) Location of the mixed pixels shown in white. (b) Lower left corner of Image III.*

Image III was similar to Image II, except for the isolated pixels that were added—see Figure 6.9. An unrealistically high number of 1.0% of the image pixels was selected at random to be replaced by isolated mixed pixels. Of the 416 isolated pixels that were added, 47 replaced pixels that were already mixed, resulting in 5297 mixed pixels to be processed. Data-driven decomposition of these pixels with a threshold setting of 24.0 gave an average error in accuracy of 5.0%. During the 5 iterations of the second stage, 36, 14, 4, 4, and 0 mixed pixels were decomposed, leaving 369 pixels for stage 3. Experiments with images where a more tractable percentage (0.1%) of the pixels was isolated showed that most isolated mixed pixels were recognised as such. Those that were not identified were frequently placed at positions already occupied by mixed pixels. Since their neighbourhoods usually contained multiple regions, often an erroneous decomposition in stage 1 or 2 was accepted because of spectral confusion. Another reason was that some isolated pixels had soil as their main component. Because soil also was the class of the edges and therefore always was considered as a possible component, a correct decomposition could already be achieved in stage 1. As can be seen in Figures 6.10(a) and 6.11(a), isolated mixed pixels could be decomposed as accurately as ordinary mixed pixels. Figures 6.10(b) and 6.11(b) show that with a threshold of 24.0 isolated mixed pixels could easily be identified.

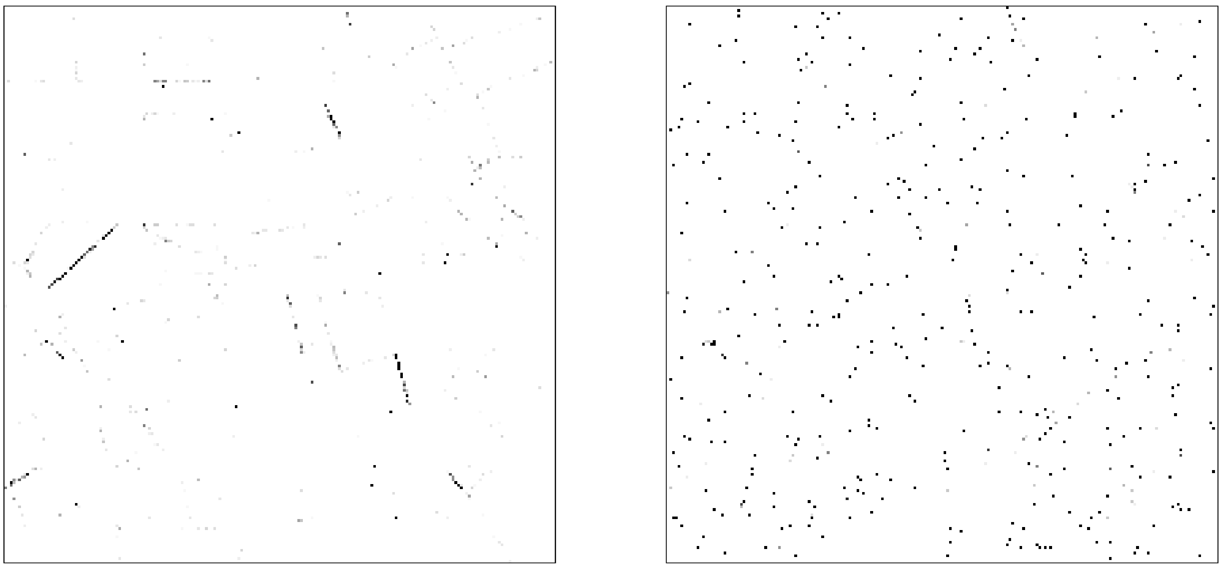
The data-driven decomposition algorithm used a threshold to determine whether a mixed pixel could be decomposed reliably enough or not. If the threshold was set too low, pixels which could be decomposed correctly were marked isolated and erroneously processed in stage 3. On the other hand, if the threshold was set too high, isolated pixels were not recognised but were incorrectly decomposed in stages 1 and 2. Furthermore, mixed pixels with contributing regions that could only be determined in the second stage



(a) Error in accuracy.

(b) Error in reliability.

Figure 6.10: *Statistics collected during decomposition of Image III. (a) Histogram of the number of pixels against the error in accuracy; range 40%–100% contained 41 pixels. (b) Histogram of the measure of (un)reliability after stage 2. The last bin shows the number of pixels having a value of e_{rel} greater than 36.0.*



(a) Error in accuracy.

(b) Error in reliability.

Figure 6.11: *Spatial distributions of e_{acc} and e_{rel} calculated for Image III. (a) Errors in accuracy; higher errors are shown as darker pixels. (b) Reliability of decomposing pixels in stages 1 and 2. Many of the isolated objects can be identified as having a high e_{rel} (black).*

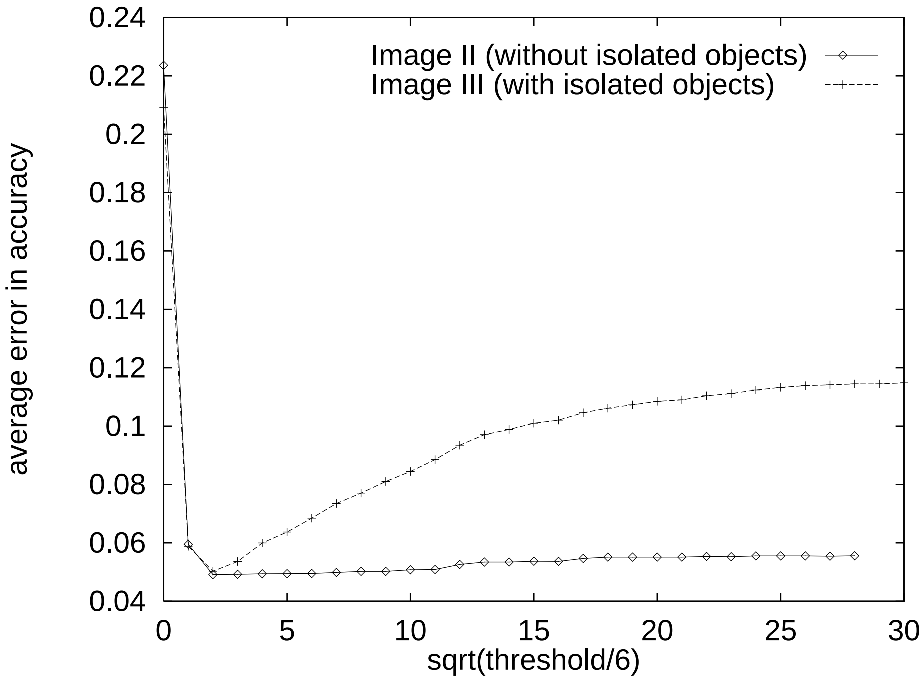


Figure 6.12: *Average error in accuracy after decomposition of Image II (with isolated objects) and Image III (without isolated objects) for different settings of the threshold.*

were decomposed prematurely as well. In Section 6.1.1 it was derived that four times the number of spectral bands might be a good setting for the threshold, something which proved to be the case in the previous experiments. In the last experiment of this section, Images II and III were repeatedly decomposed using different thresholds. The results shown in Figure 6.12 confirm that a threshold of 24.0 (Landsat-TM imagery has 6 bands) was an excellent setting. The effect of prematurely decomposing pixels whose regions could only be determined in stage 2 can be observed in the slightly rising curve of Image II. The much stronger effect of incorrectly decomposing isolated pixels is shown by the curve of Image III.

6.3 Summary and conclusions

In this chapter we presented a method to estimate the area of an agricultural field as accurately as possible. This data-driven decomposition method is a combination of the classic (pixel-based) decomposition approach and the field-based classification approach, and has two underlying concepts. The first concept is that pure pixels can be classified as usual, but mixed pixels must be decomposed such that the area of each pixel can be divided between the contributing fields. The other main concept is the application of domain-specific knowledge based on a scene model comprising agricultural fields, narrow boundary structures, and isolated objects. Aspects of this knowledge are that the spatial relationships between the pixels of an agricultural field can be used to locate mixed pixels that should be decomposed, to determine local endmember distributions from the image, and to select endmembers that are probable components of a mixed pixel. An important advantage of this endmember selection mechanism is that it permits the decomposition of a scene into a far greater number of endmembers than can be considered by classic linear mixture modelling. Further aspects incorporated in our approach are that fields are often bounded by narrow structures with well-known compositions, e.g. roads and ditches, and that isolated objects, which are composed of ground cover types not found in their surroundings, can be detected because their spectral response is not a mixture of the responses of neighbouring pixels. Since all these knowledge rules help to improve the correct allocation of mixed pixels, the crop area estimates by data-driven decomposition are expected to be more accurate than those of traditional crop area estimators.

In order to test the data-driven decomposition method, it was applied to three artificial Landsat-TM images of increasing complexity. From the results on the first image, which contained only simple mixed pixels, it appeared that most mixed pixels could be decomposed in the first stage of DDD. Not only were the correct endmembers derived from the pixel's neighbourhood, the estimation of its proportions was good as well, which led to an average error in accuracy of only 2.7%. The few errors that did occur were the result of two factors: the spectra of the pixel's components deviated significantly from their class mean, which caused a wrong combination of endmembers to look more reliable or the right combination to be not reliable enough, or the mixed pixel was composed of three or more agricultural fields, which DDD could not handle. Only the latter source of error is related to DDD and could be removed by adapting the algorithm, but the more complex decomposition method would probably achieve a lower accuracy on the simpler pixels. The second image contained mixed pixels with a more difficult composition, which resulted in the emergence of several complex narrow structures. These narrow structures could be dealt with very well in stage 2 of DDD: based on a threshold, the iterative algorithm was able to determine whether the correct endmembers had been found already using a minimum number of iterations. Due to the more complex mixtures, spectral confusion turned out to be the third and most important cause of inaccuracy. Although the impact of spectral confusion was considerable—the average error in accuracy increased to 4.9%—DDD cannot be adapted to remove its influence as it has a natural cause. Processing of the third image, which contained a large number of isolated objects, showed that by thresholding the reliability measure the ordinary mixed pixels could be separated well from the isolated mixed pixels. Pixels of the latter type were decomposed incorrectly only in case they were located on a field boundary, because that increased the possibility of spectral confusion. However, the average error in accuracy remained acceptably low (5.0%). Setting the threshold to four times the number of bands, something which was derived mathematically, proved to be an excellent choice.

Data-driven decomposition appears to be a fast and accurate method to estimate the area of an agricultural field. This conclusion, however, is based on experiments with artificial images, which were simulated using the same scene and mixing models that underly DDD itself. Furthermore, to determine whether the two concepts of DDD actually improve the area estimates or not, the results of the method have to be compared with the results achieved by traditional estimators. Therefore, a comparative experiment with preferably genuine satellite images is required, which will be presented in the next chapter.

Chapter 7

Comparison between DDD and three other area estimators*

In this chapter, the performance of the data-driven decomposition method introduced previously is compared to the results of three other area estimators. These other methods, which are described in Section 7.1, are well-known and have been carefully selected to make an evaluation of the two concepts behind DDD possible. Section 7.2 describes a comparative experiment based on artificial images; Section 7.3 discusses the set-up and results of an experiment in which a genuine Landsat-TM image was used. Section 7.4, finally, gives a brief summary of the main results and the conclusions.

7.1 Three related area estimators

As was described in Chapter 6, the two concepts of DDD are decomposition of mixed pixels and application of domain-specific knowledge. To evaluate these concepts, we compared DDD with three other methods in which the first, second, or both concepts are disregarded. Section 7.1.1 describes the most basic method called pixel-based classification. Slightly more advanced is pixel-based decomposition, which decomposes mixed pixels but ignores knowledge of the application domain—see Section 7.1.2. Equally sophisticated in that it considers domain-specific knowledge but does not decompose mixed pixels is field-based classification, which is described in Section 7.1.3. Since most of these methods are still widely used today, a comparison helps to determine the value of data-driven decomposition in practice as well.

7.1.1 Pixel-based classification

The oldest, simplest, and most widespread technique to relate pixels to ground cover types is *pixel-based classification* (PBC)—see for instance Robertson [85], Markham and Townshend [67], Irons *et al.* [52], Mather [70], and Kershaw and Fuller [61]. This approach applies neither concept of DDD, but allocates each pixel to only one of a number of selected classes based solely on the values measured in the different spectral bands. Of the many classifiers

*This chapter is based on publications [35], [36], and [38].

that exist, we chose the maximum likelihood classifier (see Appendix A), because it is well suited to handle data with a multivariate normal distribution such as agricultural fields, and it has a low computational complexity. Each class is characterised by a mean vector and a variance-covariance matrix, which are stored in a database. Since the distributions of the classes are modelled in advance, pixel-based classification is also called model-driven classification, which nicely illustrates the differences with DDD. Clearly, the results of pixel-based classification can easily be transformed to crop area estimates by counting the number of pixels assigned to a particular class. The area of an agricultural field can be determined in a similar fashion provided that the crop type of the field is known. However, if, as usually is the case, other fields of the same crop type are depicted by the same image, only those pixels of that class located in a particular region of interest—the part of the image containing the field under observation but none of the others with the same crop—should be counted.

7.1.2 Pixel-based decomposition

A more recent approach to extract compositional information from remotely sensed images is *pixel-based decomposition* (PBD), which was designed to handle scenes containing many mixed pixels (e.g. see Horwitz *et al.* [51], Adams *et al.* [2], Smith *et al.* [104], Cross *et al.* [23], and Quarmby *et al.* [82]). With this method, each pixel is unmixed individually to determine its true composition without regarding its spatial context or knowledge of the application domain. After decomposition, fractions of the pixel's area are assigned to the different classes proportional to their contribution. For the decomposition task, we selected the analytic solution of the classic linear mixture model described in Section 2.1, mainly because of its high speed combined with a relatively high accuracy. A disadvantage of this method, though, is that it can handle at most $n{+}1$ classes simultaneously, where n represents the intrinsic dimensionality of the data ($n{=}4$ for Landsat-TM data [70]). The spectral signatures of the classes that are needed by this estimator are taken from a database, which is identical to the one used for pixel-based classification. Therefore, pixel-based decomposition is also referred to as model-driven decomposition. Crop area estimates can be determined by summing the fractions allocated to each class. To estimate the area of a particular field, one has to know the field's crop type and possibly its general location in the image in order to sum the appropriate fractions of the right pixels—see Section 7.1.1.

7.1.3 Field-based classification

The third method applied is *field-based classification* (FBC), which was designed to reduce the number of misclassifications by the pixel-based approach due to within-field variation. Although the exact strategy differs for each study in which it was applied (e.g. Stakenborg [106], Ait Belaid *et al.* [3], Janssen [54], Schoenmakers [90], Schotten *et al.* [91]), the common idea is that first the pixels belonging to the same field are identified, after which the field is classified as a whole. The implementation of FBC used in this chapter is based on the same concept as DDD with respect to the application of domain knowledge. This means that FBC uses spatial information to locate mixed pixels, to determine local class

```
/* stage 1 */
for (each possibly mixed pixel) {
    determine the regions from its 8-connected pure pixels;
    construct a set containing all regions;
    add all edge classes to the set;

    for (each element of the set)
        calculate the unreliability of classification to this class;

    if (the lowest unreliability is below the threshold)
        allocate the pixel to the corresponding class;
    else
        mark the pixel;
}

/* stage 2 */
while (previous iteration classified at least one pixel)
    for (each pixel marked) {
        determine new regions from its 8-connected pixels classified in the last iteration;
        construct a set containing all new regions;

        for (each element of the set)
            calculate the unreliability of classification to this class;

        if (the lowest unreliability is below the threshold)
            allocate the pixel to the corresponding class;
        else
            mark the pixel for the next iteration;
    }

/* stage 3 */
for (each pixel marked) {
    construct a set containing all classes;

    for (each element of the set)
        calculate the unreliability of classification to this class;

    allocate the pixel to the class with the lowest unreliability;
}
```

Algorithm 7.1: *Field-based classification based on the extended scene model.*

distributions from the image itself, and to select only those classes the pixel probably belongs to. Furthermore, FBC also uses knowledge about narrow boundary structures and isolated objects to improve the classification. The resulting Algorithm 7.1 is very similar to that of DDD—compare Algorithm 6.3: all regions and other classes that DDD considers as probable components are considered as possible classes by FBC. To emphasise this similarity, field-based classification is also called data-driven classification. The method we selected for the actual classification of pixels was the standard maximum likelihood approach, which was chosen for PBC as well. A natural measure for the (un)reliability of a classification by this method is the Mahalanobis distance between the pixel and the class mean. Just like PBC (and PBD), FBC consults a class database, however, only to obtain the possible class distributions of narrow boundary structures (edge classes) and isolated objects. Despite the use of this database with global crop distributions, FBC—as is DDD—is a field area estimator rather than a crop area estimator; by counting the number of pixels allocated to a region, the area of a field can easily be determined. On the other hand, crop area estimates can be derived with little extra effort by classifying the region's mean vector followed by counting the number of pixels per crop.

7.2 Comparison based on artificial imagery

In order to make a quantitative comparison between DDD and the three other area estimators, ground truth information at subpixel level is required. Since this kind of supervised data was unavailable for any of the real satellite images in our archive, we had to turn to the artificial images once more. The set-up of the comparative experiment is described in Section 7.2.1. In Section 7.2.2, the results are presented and briefly discussed.

7.2.1 Set-up of the experiment

The experimental set-up was straightforward, consisting of three steps. In the first step, several artificial Landsat-TM images were generated using the method described in Section 6.2.1. Next, these images were processed by data-driven decomposition and the three other area estimators, while their performance was monitored using two accuracy measures. Afterwards, the accuracies of the methods were compared such that the concepts behind DDD could be evaluated; by comparing DDD and FBC, for instance, the influence of decomposing mixed pixels could be determined, while a comparison between DDD and PBD revealed the influence of applying domain knowledge.

Two of the images used in the experiment were already described in Section 6.2.3. Image II was based on the scene model with narrow boundary structures but without isolated objects; Image III was identical to Image II but had 1.0 % of its pixels converted to isolated mixed pixels. In addition to Images II and III, we also generated Images IIa and IIIa, which contained only five instead of ten different ground cover types. This restriction of the number of classes was necessary because pixel-based decomposition can handle at most $n+1$ classes simultaneously. Since the intrinsic dimensionality (n) of Landsat-TM data, which our images are based upon, is only four according to Mather [70] as well as our own experience, five ground cover types was the absolute maximum. As a bonus, by comparing the performance on the images with ten classes and those with five classes, the effect of more or less heterogeneous landscapes on the usability of DDD's concepts can be determined.

The accuracy of the crop area estimators was measured at the level of individual pixels as well as the entire region. Since all estimators operated on individual pixels, the direct way to determine their performance was to measure how well the composition of a pixel was estimated. For this purpose, we used the same metric as in Section 6.2.2 (e_{acc}):

$$e_p = \frac{1}{2} \sum_i \left| \hat{f}_{p,i} - f_{p,i} \right| , \tag{7.1}$$

whose value averaged over multiple pixels is denoted by $<e_p>$. As before, i ranges over all ground cover types, which means that the field area estimates by data-driven decomposition and field-based classification had to be transformed to crop area estimates by relating each field to a crop type. It is clear that the methods based on classification will probably give high values of e_p for mixed pixels. However, the crop area estimate for a larger region can still be accurate if opposite errors in pixel area allocation cancel each other out, e.g. overestimating the grassland area of one pixel by 40% is nullified by underestimating the grassland area of two other pixels by 30% and 10%, respectively. Therefore, the accuracy of the crop area estimators was also determined on a larger scale:

$$e_A = \frac{1}{2} \sum_i \left| \hat{A}_i - A_i \right| . \tag{7.2}$$

In this equation, $\hat{A}_i = \sum \hat{f}_{p,i}$ represents the estimated area and $A_i = \sum f_{p,i}$ the true area covered by crop i as calculated from all pixels p belonging to the area of interest. Although in a practical situation the area of interest probably consists of only one or a couple of agricultural fields, we decided to sum the crop area estimates over the entire region depicted by the image because the same effects could be observed in a more convenient way. Whereas the former approach involved a laborious aggregation of the crop area estimates by the pixel-based estimators in particular, the latter approach required no additional processing at all. Just like in Section 6.2.2, pure pixels were excluded from the calculation of e_A (and $<e_p>$) for three reasons: firstly, the results were not very interesting, secondly, their vast number obscured the performance on the mixed pixels, and, thirdly, we wanted to minimise the positive influence of using a perfect segmentation[1]. Compared to Section 6.2.2, this last reason has become even more important, because the perfect segmentation is used by DDD and FBC but not by PBD or PBC. By calculating the two accuracy measures based on mixed pixels only, it was possible to make a fair comparison between the crop area estimation capabilities of the four methods on both a small and a large scale.

7.2.2 Results and discussion

The discussion of the results is organised as follows. First, we describe some details about the way the results were derived. Next, the performance of the four area estimators is

[1]We only want to compensate for the fact that the segmentation is perfect, not for the fact that a segmentation is used, which is an important aspect of the application of domain knowledge.

	reference	DDD	PBD	FBC	PBC
$<e_p>$ (%)	–	4.9	n	38.2	48.0
e_A	–	26.6	o	734.6	1720.2
			t		
$\hat{A}_{sand}$	173.7	175.1		148	112
$\hat{A}_{soil}$	1429.0	1420.6	a	937	350
$\hat{A}_{fresh\ water}$	251.1	252.5	v	115	76
$\hat{A}_{wheat}$	663.6	663.4	a	771	787
$\hat{A}_{barley}$	497.7	489.9	i	686	206
$\hat{A}_{maize}$	485.1	484.4	l	403	371
$\hat{A}_{rice}$	409.5	406.1	a	546	667
$\hat{A}_{vineyards}$	207.8	205.0	b	285	476
$\hat{A}_{grass}$	547.2	569.6	l	715	1142
$\hat{A}_{deciduous\ forest}$	266.0	261.6	e	322	741

Table 7.1: *Performance of the estimators applied to Image II. The column "reference" lists the area of each ground cover type determined from the perfect segmentation. The value of $<e_p>$ is based on proportions expressed as percentages, while the error measure e_A as well as the estimates $\hat{A}_i$ are given in pixel area units.*

compared to determine the value of decomposing mixed pixels. Furthermore, the methods are compared to evaluate the second concept of data-driven decomposition, i.e. the application of domain knowledge. Finally, the performance on images with five classes is compared to that on images with ten classes in order to determine the sensitivity of the different methods to the number of ground cover types.

The performance of the different area estimators on Images II, IIa, III, and IIIa are presented by Tables 7.1–7.4, respectively. For all tables, $<e_p>$ was calculated based on proportions expressed as a percentage of the area of a pixel instead of a fraction. The estimated area per crop $\hat{A}_i$ and the derived error measure e_A are given in pixel area units. Because all figures except $<e_p>$ are absolute values, it must be noted that the number of mixed pixels used to calculate the tables depended on the underlying image: Tables 7.1 and 7.2 were computed using the 4928 mixed pixels contained in Images II and IIa, while Tables 7.3 and 7.4 were based on the 5297 samples extracted from Images III and IIIa. The results of data-driven decomposition and field-based classification were obtained with the threshold set at a value that gave the lowest average error per pixel $<e_p>$. With this setting, the corresponding value of e_A was close to its optimum as well, while minimising e_A, which as a function of the threshold was less smooth, often resulted in a relatively high $<e_p>$. For FBC, the optimal threshold was found to be dependent on the image, being 1734 ($6x17^2$; II), 864 ($6x12^2$; IIa), 726 ($6x11^2$; III), and 384 ($6x8^2$; IIIa); DDD's threshold was always set at 24 ($6x2^2$), which is equal to the value recommended in Section 6.1.1.

A first look at Tables 7.1–7.4 shows that data-driven decomposition was by far the most accurate method in all instances, followed by pixel-based decomposition, and, at some distance, by field-based and pixel-based classification. This order is quite distinct as the differences in accuracy are large, and it is the same regardless of whether the accuracy

	reference	DDD	PBD	FBC	PBC
$<e_p>$ (%)	–	3.9	13.0	32.7	40.6
e_A	–	11.7	203.5	915.1	1409.3
$\hat{A}_{soil}$	1853.8	1844.6	1699.3	1159	820
$\hat{A}_{barley}$	973.2	973.5	957.1	1293	826
$\hat{A}_{maize}$	947.7	943.9	913.6	725	718
$\hat{A}_{vineyards}$	772.6	782.7	776.0	1296	1467
$\hat{A}_{deciduous\ forest}$	383.3	383.3	582.2	455	1097

Table 7.2: *Performance of the estimators applied to Image IIa.*

	reference	DDD	PBD	FBC	PBC
$<e_p>$ (%)	–	5.0	n	41.1	46.4
e_A	–	22.9	o	738.7	1743.7
			t		
$\hat{A}_{sand}$	216.1	217.8		176	155
$\hat{A}_{soil}$	1446.3	1439.2	a	1027	378
$\hat{A}_{fresh\ water}$	290.2	291.4	v	101	87
$\hat{A}_{wheat}$	699.8	700.0	a	770	813
$\hat{A}_{barley}$	542.0	535.9	i	717	243
$\hat{A}_{maize}$	521.5	520.8	l	430	408
$\hat{A}_{rice}$	435.7	433.1	a	675	726
$\hat{A}_{vineyards}$	244.4	243.1	b	330	522
$\hat{A}_{grassland}$	594.1	612.6	l	742	1181
$\hat{A}_{deciduous\ forest}$	309.8	303.1	e	329	784

Table 7.3: *Performance of the estimators applied to Image III.*

	reference	DDD	PBD	FBC	PBC
$<e_p>$ (%)	–	3.9	13.1	35.3	38.9
e_A	–	12.6	204.7	961.9	1426.8
$\hat{A}_{soil}$	1906.0	1898.4	1760.4	1179	872
$\hat{A}_{barley}$	1047.9	1046.7	1025.9	1279	889
$\hat{A}_{maize}$	1017.3	1013.8	978.8	781	782
$\hat{A}_{vineyards}$	847.9	859.0	847.9	1372	1540
$\hat{A}_{deciduous\ forest}$	480.8	479.0	684.1	686	1214

Table 7.4: *Performance of the estimators applied to Image IIIa.*

comparison	II	IIa	III	IIIa	average
DDD/FBC	33.3 (708.0)	28.8 (903.4)	36.1 (715.8)	31.4 (949.3)	32.4 (819.1)
PBD/PBC	–	27.6 (1205.8)	–	25.8 (1222.1)	26.7 (1214.0)
DDD/PBD	–	9.1 (191.8)	–	9.2 (192.1)	9.2 (192.0)
FBC/PBC	9.8 (985.6)	7.9 (494.2)	5.3 (1005.0)	3.6 (464.9)	6.7 (737.4)
DDD/PBC	43.1 (1693.6)	36.7 (1397.6)	41.4 (1720.8)	35.0 (1414.2)	39.1 (1556.6)

Table 7.5: *Difference in performance between the estimators. The figures indicate the improvement achieved by the first method compared to the second method. The values of* $<e_p>$ *are based on percentages, while the* e_A *(given in parentheses) are expressed in pixels.*

was measured on the basis of individual pixels or the entire region. Furthermore, the estimates by DDD seem to be remarkably accurate in an absolute sense, although this observation is of limited value as the results are based on artificial images. To assess the different concepts of DDD, a more detailed comparison is needed, which is presented in Table 7.5. The first concept—decomposition of mixed pixels—can be evaluated by comparing DDD with FBC, or PBD with PBC. It appears that application of this concept results on average in an improvement of $<e_p>$ of 32.4% in the former and 26.7% in the latter case. The first reason for this increase in accuracy was that decomposition did not suffer from intrinsic misallocation of fractions as classification does. For instance, a mixture of 60% A and 40% B can at best be classified as class A, thus resulting in an e_p of 40%, while decomposition can split the pixel and allocate the correct fractions to the two classes. The second reason for the large difference in accuracy was that classification was much more susceptible to spectral confusion than decomposition. As described in Section 6.2.3, sometimes the decomposition approach was inaccurate due to the linear dependency of certain endmember spectra—e.g. soil, barley, and grass. FBC and PBC, however, were affected by spectral confusion much more because they use a pure pixel model to classify mixed pixels. As a result, a mixture of classes A and B often resembled the signature of a totally unrelated class C, whereas this confusion was non-existent in the linear mixture model. Both sources of error systematically overestimate the proportion of some ground cover types and the underestimate that of others. If for instance the variation of class A is larger than that of B, a pixel composed of 50% A and 50% B is nearly always allocated to A instead of B. Another example concerns ground cover types like soil, which are typical for narrow boundary structures. Since the proportion of such a class in a mixed pixel usually is small, it is unlikely that the pixel is ever allocated to that class. Spectral confusion due to the inadequate pure pixel model also results in systematic errors, because a mixture of A and B may well be classified as C, but a pixel of class C will never be divided between A and B. The fact that these errors are not random explains why the e_A of the classification approaches was relatively poor: if the area of a particular ground cover type is systematically overestimated (or underestimated) at the level of individual pixels, its aggregated regional crop area estimate will be as well. Based on all these observations, it can be concluded that decomposing instead of classifying mixed pixels increases the crop area estimation accuracy not only on a pixel basis, but also on a larger scale.

method	II vs. III	IIa vs. IIIa	average
DDD	0.1	0.0	0.1
PBD	–	0.1	0.1
FBC	2.9	2.6	2.8
PBC	-1.6	-1.7	-1.7

Table 7.6: *Difference in performance after addition of isolated objects. Only the values of* $<e_p>$ *based on percentages are given; the differences in* e_A *are omitted because the compared images had a different number of mixed pixels.*

The merits of applying domain-specific knowledge—the second concept of DDD—can be determined by comparing the results of the knowledge-based methods with the results of their pixel-based counterparts. Table 7.5 shows that the average error per pixel achieved by DDD decreased with 9.2% compared to PBD, while the difference in accuracy between FBC and PBC was 6.7% on average. The improvement in performance was mainly due to the mechanism for selecting probable endmembers. As the number of possibilities is much smaller when only a few classes are tried compared to the situation where all classes are considered either simultaneously (PBD) or one after another (PBC), spectral confusion was reduced. Data-driven decomposition resulted in a further increase in accuracy relative to pixel-based decomposition because the selection mechanism also stopped the so-called area leakage: as classic linear mixture modelling in general allocated at least a small fraction to each endmember, part of the area leaked away to classes that did not belong to the actual composition of the pixel. The positive effects of reduced spectral confusion were less clear when comparing the performance of the classification methods on images containing isolated objects. On the one hand, the average error per pixel of PBC decreased with 1.7% (c.f. Table 7.6) when isolated mixed pixels were added, because they could be classified much better than ordinary mixed pixels due to the high proportion ($\geq$ 75%) of one of the components. Field-based classification, on the other hand, was somewhat hindered—$<e_p>$ increased by 2.8% on average, see Table 7.6—because the domain knowledge about isolated objects could not be applied effectively. According to this knowledge, isolated objects have a different ground cover type and therefore a different spectral response than the agricultural fields surrounding them. Ordinary mixed pixels, however, have a spectral response which differs from that of neighbouring pure pixels as well and which cannot be discerned easily from the response of isolated mixed pixels based on the pure pixel model used by classification. As a consequence, the threshold of field-based classification could not be set such that a clear separation between ordinary and isolated mixed pixels was made, which invalidated the mechanism for selecting probable endmembers for a certain number of pixels. This also explains why the optimal threshold setting for FBC depended so much on the image: for images mainly consisting of ordinary mixed pixels the threshold should be set to a high value to guarantee an accurate estimation of the proportions per pixel in the first two stages, but for images containing a lot of isolated mixed pixels the threshold should be lowered to prevent them from being processed before stage 3. The decomposition methods based on the linear mixture model were not affected by the addition of isolated objects, which was reflected by the minutely higher error in accuracy (0.1%) and the fact that the optimal threshold setting for DDD was the same for all images. Therefore, it can

method	IIa vs. II	IIIa vs. III	average
DDD	1.0 (14.9)	1.1 (10.3)	1.1 (12.6)
PBD	–	–	–
FBC	5.5 (-180.5)	5.8 (-223.2)	5.7 (-201.9)
PBC	7.4 (310.9)	7.5 (316.9)	7.5 (313.9)

Table 7.7: *Difference in performance after doubling the number of classes. For PBD no figure could be calculated. The values of* $<e_p>$ *are based on percentages, while the* e_A *(given in parentheses) are expressed in pixel area units.*

be concluded that the use of knowledge of the application domain has a positive influence on the crop area estimation accuracy provided it can be expressed in the pixel model used.

As a final exercise, we compared the performance of the methods on Images IIa and IIIa with that on Images II and III to determine the influence of doubling the number of classes. The results, which are presented in Table 7.7, show that data-driven decomposition was hardly affected by the increased heterogeneity of the landscape. Field-based classification exhibited an increase in $<e_p>$ of 5.7%, while the increase in the average error per pixel of pixel-based classification was even higher (7.5%). For pixel-based decomposition no such figure could be determined, because it could not handle the higher number of ground cover types contained in Images II and III. These results confirm the conclusions drawn earlier in this section regarding the sensitivity of the different estimators to spectral confusion. Since the main impact of doubling the number of classes was that the possibilities for spectral confusion increased, the slightly worse performance of DDD was to be expected. The higher increase of the error per pixel obtained with FBC was caused by the use of a pure instead of a mixed pixel model, which made it more sensitive to spectral confusion. The fact that the increase in $<e_p>$ achieved by PBC was even higher can be explained by the omission of the endmember selection mechanism, whose use limited the number of possibly confusing cases considered by the knowledge-based methods. The only surprising result in Table 7.7 is that the regional crop area estimate (given in parentheses) of FBC showed some improvement, while the average error in accuracy per pixel deteriorated at the same time. However, this was due to opposing instances of spectral confusion, which led to simultaneous under- and overestimation of certain ground cover types. In summary, of the four area estimators applied, data-driven decomposition is the least sensitive to the number of classes, because both its underlying concepts reduce spectral confusion.

7.3 Comparison based on Landsat-TM imagery

A problem of using artificial data is that the necessary model to simulate the data will always be a considerable simplification of reality. A more important weakness of the experiment presented in Section 7.2, though, is that the scene and mixing model used to generate the images were also part of the data-driven decomposition approach itself. Therefore, we carried out an experiment to determine the relative accuracy of data-driven decomposition on a real Landsat-TM image as well. In Section 7.3.1 it is described how the experiment was set up such that a quantitative analysis of the accuracy could be made with the limited information at subpixel level available. A discussion of the results can be found in Section 7.3.2.

7.3.1 Set-up of the experiment

The study site that was selected is located near the village of Zeewolde in the south of the Flevoland polder. The land, which was reclaimed from the lake named "IJsselmeer" in 1968, is flat with an altitude of 3.30 m below sea level and is mainly used for the cultivation of arable crops. A network of roads, canals, and ditches divides the area into lots, which are basic units of approximately $500x1600$ m^2 that can be rented or bought. Many farmers subdivide their lots into several agricultural fields, which are characterised by the cultivation of a single crop. The location of the field boundaries is determined by the farmers themselves and is subject to change every year. In his study of the nearby Biddinghuizen test site, Janssen [54] calculated that the average field size was 6.9 ha. This relatively large size made the area most suitable for field-based methods such as DDD and FBC, since it permitted an accurate calculation of the statistical distribution of the average field.

For a quantitative analysis of the different area estimation methods, supervised topographical data having an accuracy that went beyond the resolution of the scanner was needed. For this purpose, the boundaries of 17 lots were digitised from a 1:10,000 topographical map—see Figure 7.1. If we assume a digitising error of ± 0.5 mm, the boundaries had a positional accuracy of ± 5 m, which was several times higher than the 30 m resolution of the Landsat-TM scanner. Co-registration of the topographical data with a Landsat-TM image of the same region was accomplished by an affine transformation, which translated, rotated, and scaled the vector data. The alternative was to geocode the satellite image in the map's coordinate system, but this method would deteriorate the image quality because it required resampling of the image. The parameters of the affine transformation were calculated using several ground control points (GCPs), which were identified in both the image and the digitised map. The root mean square (RMS) error of resulting transformation was only 0.283 pixel, which was relatively low compared to the values (0.4 and 0.9) that were achieved by Janssen. Furthermore, the factor by which the boundaries were scaled was approximately 29.99, which corresponded with the 30 m resolution of Landsat-TM images very well. Based on these figures as well as on careful visual inspection of the lot boundaries superimposed on the satellite image (c.f. Figure 7.2), it was assumed that co-registration with an error in accuracy of less than one pixel was achieved.

To evaluate the accuracy of the area estimators, several options were considered:

I. Compare the estimated and true proportions per pixel. This approach was preferred because it provides the most detailed information about the performance of the area estimator. Unfortunately, both the resolution of the topographical data and the accuracy of the co-registration were too low to determine the true composition of each pixel. A better co-registration may be achieved by using many more GCPs or by applying a different registration algorithm, while more accurate topographical data may be acquired from the Land Registry in Apeldoorn (the Netherlands), where the "Grootschalige Basiskaart van Nederland", a map with a resolution of 0.5 m, is maintained. However, this data is in general only commercially available, may not be

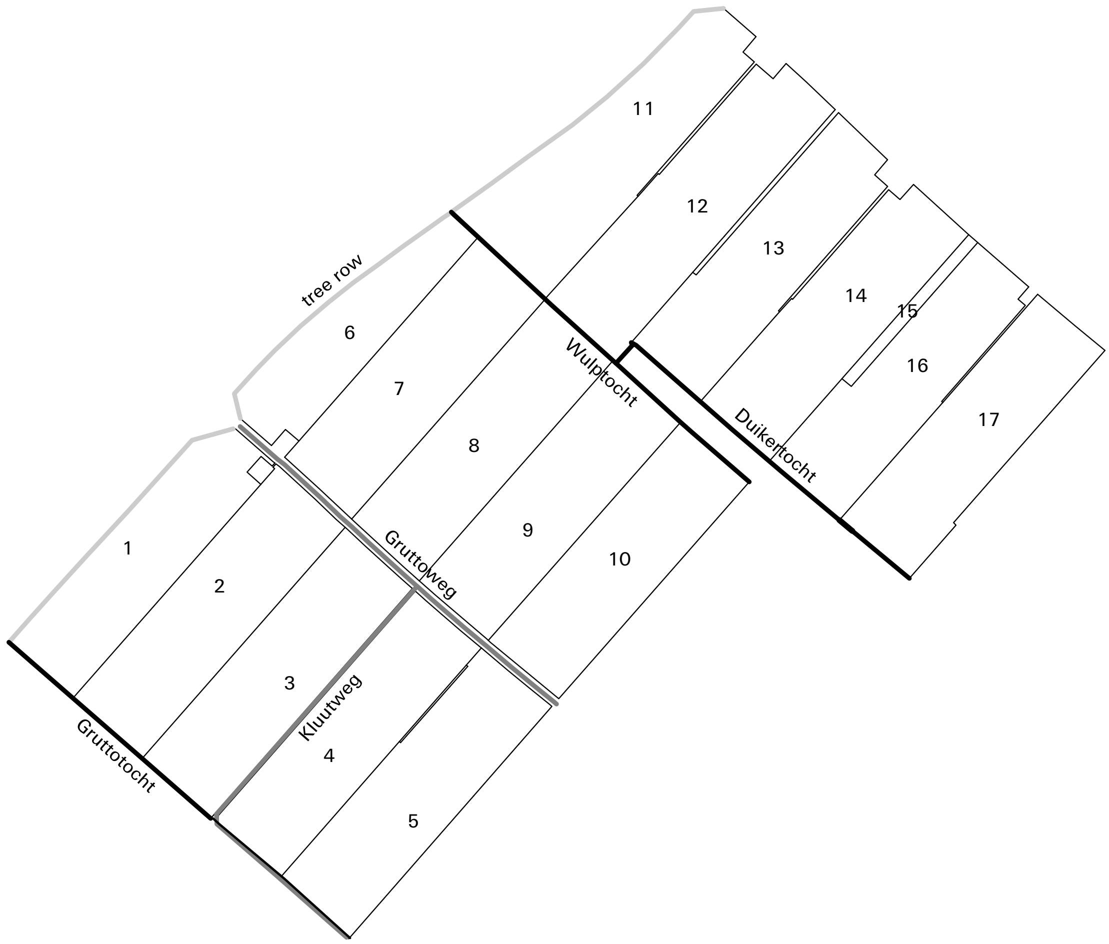


Figure 7.1: *Digitised boundaries of the 17 lots used in the experiment. The lots are separated from each other by ditches or major boundary structures such as tree rows (light grey), asphalt roads (dark grey), and canals (black).*

digitised for the Zeewolde area, and may still be not detailed enough. Furthermore, it remains questionable whether the results will be meaningful, because the fact that the vegetation often gets more sparse towards the field boundaries, where the mixed pixels are situated, is not reflected by the topographical data.

II. Compare the estimated and true area per field. An accurate estimation of the area per field is one of the goals of the EU, because it is needed to check a farmer's subsidy claim. Unfortunately, no topographical map includes field boundaries because they change too frequently. Janssen [54] derived the crop types and field boundaries of his Biddinghuizen test site from maps that were provided by the farmers themselves, but he found that the field boundaries could have been added just as well by means of on-screen digitising. Since the resulting positional accuracy is rather poor, a quantitative analysis on this basis would not be very accurate either.

III. Compare the estimated and true area per crop. Since an estimation of the area per crop is needed to make a prediction of the agricultural production, which is used to determine the subsidy for each crop, this is another goal of the EU. Because the topographical map was our only source of supervised information, the true area per crop had to be derived by summing the areas of all fields on which the crop was being grown. But since the field boundaries could not be located very accurately—see the discussion of alternative II—this approach would not give very reliable results.

IV. Compare the reliability of the estimates with the distribution of the reliability measure based on an image for which the accuracy of the estimates is known as well. If it is found that an estimator can process the mixed pixels as reliably as before, for instance when processing the artificial images of Section 7.2, one might assume that the area estimation accuracy achieved for the real Landsat-TM image also shows a similar pattern. Although this approach can always be used because no topographical information is required, it has some severe drawbacks. One of them is that classification and decomposition use different reliability measures, so no straightforward comparison of their results can be made. Another drawback is that the conclusions are at best indirectly related to the area estimation capabilities of the methods applied.

Although none of the four alternatives seemed to be particularly suitable, a slightly altered Approach II was adopted as the basis for our experiment. Instead of comparing the estimated and true area per field, a comparison was made between the estimated and true area per lot. For this purpose the accuracy of the topographical data was high enough: the true area of a lot could be calculated with an error of only $\pm 2.6\%$[2]. An estimation of the lot area could also be determined relatively easy for field-based methods such as DDD and FBC. Under the assumption that the outer boundaries of the agricultural fields within a lot coincided precisely with the boundaries of the lot itself, the lot area could be calculated by summing the area of all the fields situated within the lot. Therefore, we first carried out a vector-to-raster conversion of the topographical data to find out which pixels belonged to what lot; based on the resulting image, a mapping could be made between the agricultural fields provided by the area estimators and the lots of the topographical data. However, care had to be taken that the lot area thus estimated was compensated for the area allocated to narrow boundary structures inside a lot, as these structures were not present on the topographical map. For example, a strip of grass separating two fields within a lot can be the cause of an area leakage to the edge class grassland, which leads to an underestimation in case the strip of grass is uncharted. Our solution to this problem was to count the number of the pixels located completely inside the lot under observation and add the pixel fractions assigned to one of the fields constituting the lot for each pixel that, according to the vector-to-raster conversion, was located on the lot's boundary. After a simple multiplication—Landsat-TM pixels cover exactly $30x30$ m^2—the area estimate could be compared directly to the true lot area derived from the topographical map.

Apart from data-driven decomposition, the only area estimation method that was applied was field-based classification. The pixel-based approaches PBD and PBC were excluded for three reasons. First of all, the number of endmembers that were present in the

[2]Given an average lot size of $500x1600$ m^2 and a digitising error of ± 5 m, a rectangular lot is underestimated by at most $(1 - (490x1590)/(500x1600))x100\% = 2.6\%$ and overestimated by at most $((510x1610)/(500x1600) - 1)x100\% = 2.6\%$.

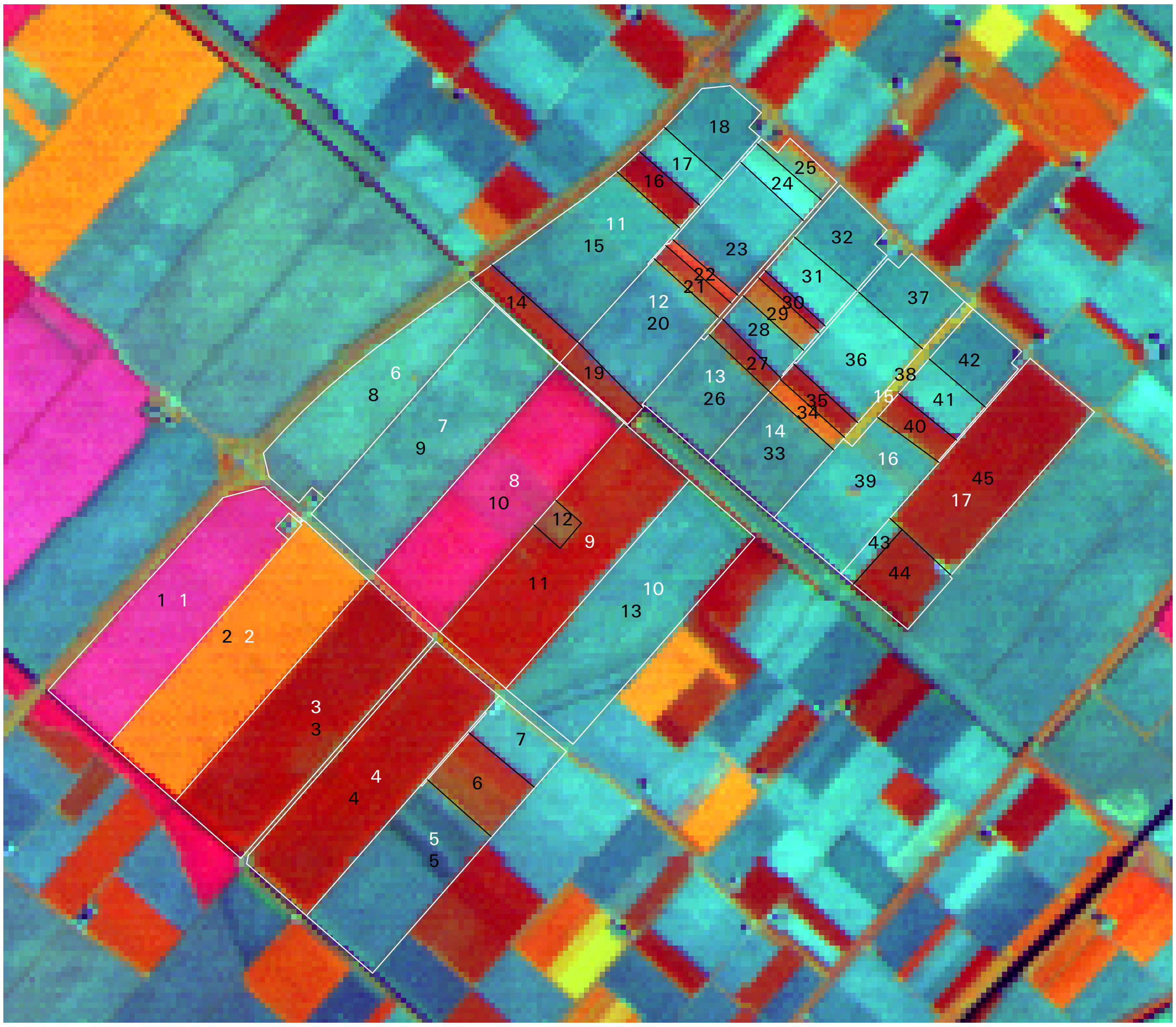


Figure 7.2: *False colour composite (RGB: 453) based on a Landsat-TM image of the Zeewolde study site. Lot boundaries (numbers) are given in white, field boundaries (numbers) are shown in black. Visual inspection suggests a co-registration error of less than one pixel.*

image was too large to be handled by PBD. Secondly, since both pixel-based methods allocated area to ground cover types instead of to agricultural fields, calculation of the area per lot was much more complicated. The third reason was that according to the topographical data, which was used to evaluate the crop area estimators, all pixels inside the boundaries of a lot belonged entirely to that lot; because this view is shared by the knowledge-based methods with regard to agricultural fields, they were favoured during evaluation, while the pixel-based approaches were not. Leaving out PBD and PBC had one other advantage: we could use a supervised segmentation to drive DDD and FBC without biasing the order of most accurate estimators. To obtain this segmentation, the lot boundaries were extended with field boundaries by means of on-screen digitising. After vector-to-raster conversion using a sufficient margin to account for co-registration errors and positional inaccuracy of the field boundaries, a segmentation of high quality was available. It was calculated that 19.4% (15.6%) of the digitised area was occupied by pixels on field (lot) boundaries, which suggested that, even when surveying an area with relatively large fields like the Flevoland polder, the accuracy of the area estimates may be improved considerably by applying techniques specialised in mixed pixels.

In addition to a segmentation, both area estimators needed a class database containing the distribution of all ground cover types present in the image. Since these distributions had to reflect the same bias due to sunlight intensity, atmospheric effects, sensor calibration, etc. as the mixed pixels, they had to be derived directly from the image. The Landsat-TM image of the Zeewolde study site that was used was acquired on 24th May 1990, showing no clouds and less scanner anomalies than more recent images. However, it was taken rather early in the growing season, as a result of which many fields lay fallow, while the remaining fields were covered with only cole-seed, alfalfa, winter wheat, and grass. Based on a 1:25,000 topographical map[3] of the same area, we assumed that the narrow boundary structures were composed of water, forest, grass, soil, and asphalt. Each class was characterised by one distribution in the database, but for some classes several distributions were added in order to cover their spectral variety, e.g. shallow, deep, and canal water; coniferous and three types of deciduous forest ranging from green to brown; and three types of grass ranging from green to yellow. A remote sensing expert determined the distribution of each class by selecting a sufficiently large area completely covered by that class and calculating the mean vector and variance-covariance matrix of the corresponding pixels. Only for the classes canal water and asphalt road no such regions could be found; the distributions of these classes were determined by collecting a number of non-contiguous pure pixels of the same class and calculating the set's statistics.

To simplify a quantitative comparison of the results of the area estimators, it was desirable to have a single measure based on the area estimates of the individual lots. Furthermore, the measure was also needed to select the optimal configuration from a series of trial runs of the experiment in which the threshold setting and the set of edge classes were varied. A measure that meets these requirements was already introduced in Section 7.2.1:

$$e_A = \frac{1}{2} \sum_i \left| \hat{A}_i - A_i \right| . \tag{7.3}$$

[3]Topographical map of the Netherlands, sheet 26D, 1991.

However, the above accuracy measure expresses an error that is slightly different due to the different meaning of A_i and $\hat{A}_i$: whereas in Equation (7.2) A_i and $\hat{A}_i$ represented the true and estimated area covered by the i-th crop, in Equation (7.3) they denote the true and estimated area of the i-th lot, respectively. The scalar $\frac{1}{2}$ could be removed, since underestimation of lot i does not necessarily result in the overestimation of lot j because of the presence of unmapped boundary structures. However, since the scalar had no effect on the order of best area estimators, we saw no reason to remove it either.

7.3.2 Results and discussion

In this section, we try to verify some of the conclusions drawn based on the experiments with artificial images. In the first part, the lot area estimates by DDD and FBC are compared in order to determine the impact of decomposing mixed pixels. DDD's second concept—the application of domain knowledge—could not be evaluated because no fair comparison with the results of PBD and PBC could be made. The other part of this section deals with the errors due to spectral confusion; because of the format of our supervised data, this was the only source of inaccuracy that could be studied.

Estimation of the lot area

In Table 7.8, the area of 17 lots derived from the digitised topographical map are presented, together with their estimates by data-driven decomposition and field-based classification. The set of edge classes that was used in these experiments contained all ground cover types mentioned previously, except for cole-seed, alfalfa, and winter wheat. Leaving one or more classes out of this set resulted in a higher average error, at least as far as DDD was concerned. The threshold setting with which the results were obtained was chosen such that e_A was minimal. While the optimal setting could be determined unambiguously in case of DDD, for FBC the choice was less clear because e_A as a function of the threshold had many local minima that were close to the global minimum. Anyway, even the lowest value of e_A achieved by FBC (222.9) was still more than three times as high as the best value achieved by DDD (66.7). More important, though, was that the decomposition approach gave a more accurate area estimate than classification for 13 of the 17 lots, while in one case—lot 4—their estimates differed by less than one pixel.

To determine whether DDD was affected by other major error sources than spectral confusion, we investigated why FBC was more accurate than DDD in estimating the area of lots 1, 8, and 9. The problem with lot 1, whose area was overestimated by both methods, was that the pixels at the south-west boundary partially covered an undigitised field of the same crop. As a result, the crop area of lot 1 was overestimated during decomposition; the classification approach, on the other hand, allocated these pixels to an incorrect class due to spectral confusion, thus decreasing the already overestimated area. The area of lot 8 was severely underestimated by DDD and slightly overestimated by FBC. A close look at its north-east boundary, however, revealed an uncharted strip of bare soil, which invalidated the requirement that field and lot boundaries should coincide. Consequently, underestimation of the lot area indicated a more accurate crop area estimate. The area of lot 9 was overestimated for about the same reason as lot 1. Because of spectral confusion

	supervised data		*DDD incl. soil*		*FBC incl. soil*		*FBC excl. soil*	
lot	main class	area	area	deviation	area	deviation	area	deviation
1	cole-seed	825.6	840.7	15.1	830	4.4	830	4.4
2	alfalfa	945.0	953.6	8.6	904	-41.0	904	-41.0
3	winter wheat	928.8	933.4	4.6	954	25.2	954	25.2
4	winter wheat	888.7	886.8	-1.9	887	-1.7	887	-1.7
5	bare soil	932.8	932.7	-0.1	904	-28.8	930	-2.8
6	bare soil	617.8	624.2	6.4	590	-27.8	631	13.2
7	bare soil	886.1	866.8	-19.3	810	-76.1	845	-41.1
8	cole-seed	890.7	862.6	-28.2	895	4.3	895	4.3
9	winter wheat	886.4	898.7	12.3	894	7.6	894	7.6
10	bare soil	879.8	884.0	4.2	864	-15.8	914	34.2
11	bare soil	1055.9	1055.5	-0.4	1013	-42.9	1037	-18.9
12	bare soil	1004.6	1009.8	5.2	969	-35.6	999	-5.6
13	bare soil	880.8	879.2	-1.6	840	-40.8	871	-9.8
14	bare soil	901.2	887.4	-13.8	847	-54.2	877	-24.2
15	grassland 3	76.9	75.1	-1.9	84	7.1	94	17.1
16	bare soil	826.1	835.9	9.8	795	-31.1	814	-12.1
17	winter wheat	883.6	883.8	0.2	882	-1.6	897	13.4
e_A		–		66.7		222.9		138.3

Table 7.8: *Lot area estimates by data-driven decomposition and field-based classification for the Zeewolde study site. The column "main class" lists the dominant ground cover type according to a remote sensing expert. The columns under "FBC excl. soil" give the results of classification with soil excluded from the set of possible components of boundary structures. All values are expressed in pixel area units.*

between winter wheat and the mixture of grass and water, a number of pixels situated north-east of the lot's boundary were incorrectly allocated to this lot. But whereas DDD assigned fractions of the boundary pixels themselves to lot 9 as well, FBC allocated them to an incorrect class altogether, which compensated for the overestimation mentioned before. In summary, it appeared that the few cases in which FBC seemed more accurate than DDD were due to spectral confusion as well as the way in which the results were evaluated.

The experiment was repeated under the assumption that soil was not one of the possible components of the narrow boundary structures. We took this step because it was obvious from Table 7.8 that FBC underestimated the area of nearly all lots largely covered by bare soil. Presumably, this approach allocated considerable fractions of these lots to narrow boundary structures composed of soil, which in reality might not even exist. The results of FBC after removing soil of the set of edge classes are presented in the last columns of Table 7.8; the results of DDD are not shown as they deteriorated over the entire line. As expected, FBC underestimated the area of most lots covered by soil less severely or even overestimated them, thus resulting in an e_A that decreased from 222.9 to 138.3, which is only twice the value achieved by DDD. A qualitative comparison of the error per lot, however, showed virtually no improvements: DDD was still more accurate in 12 cases, approximately as accurate (±1 pixel) in two cases, and less accurate in only three cases compared to FBC. Despite the positive effects on the overall performance of FBC, excluding soil as a possible boundary component probably was not legitimate. For one, it led to the (further) overestimation of the area of two lots—numbers 15 and 17—whose main ground cover type was completely different from soil. Furthermore, the performance of DDD decreased dramatically, which also suggested that the image did contain narrow boundary structures composed of soil. Therefore, it was concluded that FBC was much more confused by the slight differences in soil distributions than DDD, which resulted in the allocation of pixels to the edge class soil which in fact belonged to fields that lay fallow.

Based on the above results, we conclude that decomposition of mixed pixels leads to more accurate area estimates of agricultural fields than classification. Compared to the experiments with artificial images, the difference in performance between DDD and FBC was less pronounced due to several reasons. First of all, the number of mixed pixels used to calculate Table 7.8 (2337) was only half the number used in Section 7.2 (4928 and 5297). Furthermore, the indirect method to determine the performance—based on lots instead of fields—obscured the true accuracy of the estimators in some cases (e.g. lot 8), because the requirement that lot and field boundaries coincided was not always met. As a result, accurate area estimates appeared to be inaccurate, while an inaccurate estimator could seem to be more accurate if its deviation was compensated for by the error in the supervised data. Finally, the fact that the mixing model assumed by DDD probably was not as good an abstraction for the real image as it was for the artificial images will also have had a negative influence on its performance. The validity of the scene model, upon which the second concept of DDD—application of domain knowledge—is based, could not be determined since no comparison could be made with the pixel-based approaches. A related issue is to what degree the Zeewolde image was representative of other agricultural scenes to which the area estimators might be applied. As far as the large average field size is concerned, the positive influence probably was limited as the performance on lots with small fields (e.g. lots 12, 13, and 15) was as good as that on large fields (e.g. lots 1–4). A greater diversity of crop types will probably be more beneficial, provided that the new classes do not lead to more spectral confusion with the classes of the narrow boundary structures in particular. On the other hand, considerable variations in terrain altitude will probably decrease the overall accuracy, partially because the distribution of a field's crop type may no longer be derived as reliably as before, and partially because the linear mixture model may be invalidated. Although many factors exist that could be investigated, we are confident that in general the area estimation accuracy of DDD is higher than that of FBC.

Spectral confusion

Based on the available supervised data, spectral confusion was the only source of inaccuracy of DDD and FBC that could studied. In this thesis, the term *spectral confusion* is used to describe the resemblance of one (mixture of) ground cover type(s) to another (mixture of) ground cover type(s). Basically, we distinguish three different kinds of spectral confusion. The first sort occurs during decomposition of mixed pixels based on the linear mixture model; if several linear combinations of classes all result in the same reflectance observed for an actual pixel, errors in area allocation are eminent. Since a pure pixel can be regarded as a mixed pixel of which all classes but one have a zero fraction, this problem can occur with pure pixels as well. The second type of spectral confusion emerges when mixed pixels are classified using a pure pixel model. As a result, a mixed pixel may no longer match any of the spectral signatures of the contributing classes, but that of an unrelated class instead (see Figure 1.3). The third category includes the errors made during the classification of pure pixels based on the pure pixel model. In this case, misclassification is not caused by an inadequate pixel model, but by a deviation from the correct class mean that is too large given the natural variation within that class in relation to the distribution of other classes. The experiment with the Zeewolde image showed only one error of the last type, which was made during classification of the field mean vectors. According to a remote sensing expert, field 22 was covered with grass, but the classification algorithm had a minor preference[4] for alfalfa. The other two types of spectral confusion could be studied by comparing the approximate composition of several narrow boundary structures determined from the 1:25,000 topographical map[3] with their estimated composition by DDD and FBC. Thus, we examined three types of boundary structures: a tree row, asphalt roads, and canals.

The tree row that was studied was located along lots 1, 6, and 11—see Figure 7.1. FBC detected the presence of the boundary structure along the entire line, although some small gaps, where pixels had been allocated to the adjacent fields, were present. Most pixels along lots 6 and 11 were classified as asphalt road instead of forest, while the pixels along lot 1 were predominantly labelled as grassland. Similarly, DDD found the latter part of the tree row to consist of grassland and water, but the pixels along lots 6 and 11 were correctly decomposed into deciduous forest and grass. An explanation for the discovery of grassland could be that the trees along lot 1 had been cut down. However, since lot 1 was covered by cole-seed, while lots 6 and 11 lay fallow, it is more probable that the former mixture gave rise to spectral confusion, while the latter did not.

As can be seen in Figure 7.1, the Zeewolde image contained two main roads: the Kluutweg, which continued along the south-west boundary of lots 4 and 5, and the Gruttoweg. FBC labelled most pixels of the Kluutweg as asphalt road, but several segments of considerable length remained undetected. A large part of the Gruttoweg was classified as grassland; although the topographical map showed that grassland verges were indeed part of this structure, it is questionable whether this was truly their main component. DDD allocated large fractions of the Kluutweg between lots 3 and 4 to the class bare soil instead of asphalt; decomposition of the part along lots 4 and 5, on the other hand, did reveal the presence of the asphalt road. Most pixels of the Gruttoweg were unmixed perfectly: both the asphalt road as well as the grassland verge were identified. In addition to the two main roads, both methods found several asphalt structures inside a lot, where, according to the map, no (permanent) structure was located. Since these road fragments were small, did not connect to other roads, and were mainly located between fields of winter wheat and bare soil, they were probably due to spectral confusion.

The last structures we examined were three canals named Duikertocht, Gruttotocht, and Wulptocht (see Figure 7.1). Although FBC did detect these boundary structures, it allocated nearly all of their pixels to asphalt instead of water; in fact, not a single pixel was classified as water in the entire image. DDD, on the other hand, discovered

[4]The M-distance between the field's mean and alfalfa was 72.3, compared to 76.1 for grassland 2.

class	band 1	band 2	band 3	band 4	band 5	band 7
shallow water	83.5	36.4	31.3	18.2	7.4	4.1
deep water	79.9	33.8	28.3	17.1	7.0	3.9
canal water	81.6	31.6	28.1	33.5	23.3	9.3
coniferous forest	75.4	28.5	25.2	50.3	28.3	9.9
deciduous forest 1	78.5	32.0	28.9	52.3	60.3	26.5
deciduous forest 2	77.3	31.2	26.0	74.3	41.3	12.4
deciduous forest 3	77.8	32.2	27.3	77.8	52.4	17.5
grassland 1	85.4	36.3	32.1	103.8	92.1	34.0
grassland 2	80.6	32.1	27.1	115.4	59.8	18.0
grassland 3	82.7	35.0	31.2	91.2	74.8	27.8
asphalt road	92.2	40.0	42.4	56.2	60.8	32.2
bare soil	96.8	42.2	46.8	55.8	75.3	46.3
cole-seed	86.0	45.7	46.3	111.5	54.3	19.4
alfalfa	82.5	34.9	29.1	119.9	68.9	20.0
winter wheat	80.7	31.6	27.1	90.3	47.6	14.5

Table 7.9: *Class mean vector of all ground cover types present in the Zeewolde image.*

considerable fractions of water in all three canals and even identified some of the ditches separating the lots. Especially the Duikertocht and the part of the Gruttotocht along lots 1–3 were found to consist of water for a large proportion; as mentioned before, the part of the Gruttotocht parallel to the Kluutweg was—understandably—largely allocated to asphalt. The Wulptocht proved to be the most difficult structure to decompose, with pixels containing a fair amount of water found only along lot 9. Because of spectral confusion, deciduous forest was estimated to be the main component of the pixels along lots 6, 7 and 10, whereas the pixels along lot 8 were thought to contain large fractions of winter wheat (the class of field 19, which was adjacent to lot 8).

From these examples it has become clear that both DDD and FBC suffered from spectral confusion, being of type 1 and 2, respectively. The most frequent error was the discovery of asphalt in pixels on boundaries between fields of soil and winter wheat, and in pixels covering two lots of bare soil separated by a ditch. In order to check the possibility of spectral confusion within the linear mixture model, some of these pixels could be decomposed to see if a slightly less probable but more logical composition would be possible as well. Instead of decomposing individual pixels, we decomposed the class mean of asphalt, because this way we could deal with most pixels having a high proportion of asphalt simultaneously. It turned out that unmixing asphalt using two endmembers resulted in a reliability error e_{rel} (see Equation 6.4) of less than 10.0 if and only if one of the components was bare soil. A few of the mixtures closely resembling asphalt were 71.5% bare soil and 28.5% canal water (e_{rel}=2.634), and 80.9% bare soil and 19.1% winter wheat (e_{rel}=4.602). Decomposition in three components also gave a e_{rel} of less than 10.0 only if one of the components was bare soil; 71 of the 78 possible 3-tuples containing soil had a sufficiently low error. Of these, the mixture of 67.2% bare soil, 15.4% shallow water, and 17.4% winter wheat was one of the best alternatives, having an e_{rel} of only 0.946. These results, seen in the light of the two errors mentioned earlier in this paragraph, strongly suggest that spectral confusion occurred during DDD. Although these examples simultaneously explain many of the misclassifications by FBC, the fact that asphalt was likely to be overestimated can also be inferred directly from the class means themselves. In Table 7.9 it can be seen that the spectral signatures of bare soil and asphalt were remarkably similar: the mean value of asphalt was a bit lower than that of soil in all bands except band 4, where the mean value of asphalt was slightly higher. Since all other classes displayed a similar though larger deviation from the mean vector of bare soil, a mixed pixel consisting of bare soil and an arbitrary, not too large fraction of another class was easily confused with asphalt if a pure pixel model was assumed. Given the fact that the Zeewolde image contained many fields of soil, the multitude of pixels that FBC misclassified as asphalt was to be expected. The errors made by DDD, though more diverse, were smaller in number, because spectral confusion within the linear mixture model was much more bound to ground cover types having specific fractions.

7.4 Summary and conclusions

In this chapter, the performance of data-driven decomposition was compared to the results of three other area estimators in order to determine the influence of the two concepts underlying DDD. The other methods—pixel-based decomposition, field-based classification, and pixel-based classification—are well-known from the literature and were selected because they applied only the first (PBD) or the second concept (FBC), or neither (PBC) of the concepts of DDD. Thus, by comparing the estimates by DDD and FBC, for instance, the impact of decomposing mixed pixels could be assessed, while a comparison between DDD and PBD showed the influence of applying domain-specific knowledge.

Application of the different methods to artificial images showed that DDD was much more accurate than the other three estimators. Decomposition of mixed pixels had as first advantage that misallocation of fractions, which is intrinsic to classifying mixed pixels, was counteracted. Furthermore, the errors due to spectral confusion were reduced because the linear mixture model is less sensitive to spectral confusion than a pure pixel model. This difference in sensitivity became especially clear when the number of classes was doubled, which deteriorated the performance of FBC and PBC in particular. The main benefit of using domain knowledge was that the endmember selection mechanism, which was based on this knowledge, led to a further reduction of spectral confusion. If combined with the decomposition approach, this mechanism also prevented small fractions from being allocated to unrelated classes. Addition of isolated mixed pixels to the images revealed that the knowledge regarding isolated objects could not be formalised in the pure pixel model very well. As a result, the optimal threshold setting for FBC differed per image and was difficult to determine. The increase in accuracy achieved by DDD was found not only at the level of individual pixels, but also at a larger scale. The explanation for this result was that the errors due to spectral confusion as well as intrinsic misallocation were not random but caused systematic overestimation of some ground cover types and underestimation of others. Since data-driven decomposition counteracted both sources of error, its crop area estimates were more accurate than those of the other methods for single pixels, agricultural fields, and entire regions.

In order to exclude the possibility that DDD was favoured as it was based on the same mixture and scene models used for image simulation, DDD and FBC were also applied to a real Landsat-TM image. Because of the characteristics of the image and the available supervised data, the pixel-based estimators could not be applied such that a fair comparison of their performance could be made. Compared to the results of FBC, DDD gave a more accurate estimation of the lot area in 13 out of 17 cases, while in one case their estimates were approximately equally well. FBC seemed to be more accurate in the remaining three cases, but this was due to coincidentally opposing instances of spectral confusion or the way in which the results were evaluated. Even after excluding soil as a possible component of narrow boundary structures because FBC, unlike DDD, clearly was confused by the small differences in the various distributions of soil, this qualitative ordering did not change. The only source of inaccuracy that could be studied with our limited ground truth information was spectral confusion. After examining the true and estimated composition of several boundary structures, it appeared that FBC incorrectly classified most of the corresponding pixels as asphalt due to spectral confusion caused by an inadequate pure pixel model. The type of spectral confusion DDD suffered from led to more diverse errors, but the extent of the errors was much less as spectral confusion within the linear mixture model is limited to more specific mixtures.

In general, the findings of the experiment with the artificial images were confirmed by the results of the experiment with the real satellite. Although the merits of using domain-specific knowledge could not be verified as the pixel-based estimators could not be applied, we feel that, based on the former experiment, both this concept and the decomposition of mixed pixels lead to more accurate estimates. Therefore, we conclude that DDD can estimate the area of agricultural fields with higher accuracy than many of the other, well-known methods used today.

Chapter 8

Conclusions

In this final chapter, the conclusions of the study underlying this thesis are presented. Section 8.1 briefly summarises the main findings, while Section 8.2 discusses to what extent the goals set in the introductory chapter have been achieved. Section 8.3, finally, suggests possible improvements of the methods proposed as well as other topics for future research.

8.1 Summary

As described in Chapter 1, the European Union needs to be able to estimate the area of an agricultural field in order to manage its agricultural subsidy system. A cost-effective way to estimate the area of many fields on a frequent basis is to use an automated system that determines the area of each field from satellite remote sensing images. In general, such a system works by allocating each pixel to the class it resembles most, followed by counting the number of pixels per class over the region of interest. This approach, however, results in inaccurate estimates when mixed pixels are involved: since these pixels each cover several ground cover patches belonging to different classes, classifying them will overestimate the area of some classes while that of others will be underestimated. Alternatively, one can try to estimate the composition of a mixed pixel, after which it can be allocated to several categories proportionally to the fraction of its area covered by each class. Under the assumption that most mixed pixels are situated on the boundary of two agricultural fields, the latter approach is expected to have an accuracy that is at least 25% per mixed pixel higher than the former (classification) method. Apart from developing such an accurate, decomposition based method to estimate the area of agricultural fields, which is our main goal, this thesis also aims at providing a deeper knowledge about the processing of mixed pixels in general, i.e., (relatively) independent of the specific application studied and type of imagery used.

As a first step, the literature was studied to find decomposition techniques suitable for our purposes. Chapter 2 reviews twelve methods, which are different in some aspects but have many properties in common as well. The most widely used estimator by far is the analytic solution of the (statistical) linear mixture model; when provided with certain class characteristics, viz. the endmember distributions, this generalised least-squares method is able to determine the proportion of each component from the overall pixel vector measured. So far, no extensive study comparing the different approaches has been carried out. Several smaller studies suggest that with some of the more advanced methods, e.g. artificial neural networks, better results can be achieved, but at the cost of a significant increase in the number of computations. Since the accuracy with which mixed pixels can be decomposed in many cases seems to depend much more on the characteristics of the data itself than on the decomposition method selected, the fast and simple analytic estimator based on the linear mixture model is still used for most applications today.

A way to improve the decomposition accuracy, however, is to use a method that has been tailored to the application domain. In Chapter 3, a new linear mixture model is proposed in which the correlations that exist between the endmember spectra of neighbouring pixels in, for instance, agricultural images can be described. Solving this physical, or more correctly, environmental model entails estimating the spectral response of each component of a mixed pixel in addition to the component proportions. Although this rather complicated operation requires more processing time than solving the statistical model, the estimated component spectra contain information that cannot be extracted using classic decomposition techniques. Furthermore, as was shown experimentally, the accuracy of the proportion estimates obtained with the novel approach is higher than that of the usual analytic estimator mentioned in the previous paragraph, provided the correlations between adjacent pixels are exploited. Especially in case the data set is difficult to decompose, the improvement in accuracy is considerable.

Chapter 4 deals, in the context of hyperspectral remote sensing, with two problems of most analytic decomposition methods based on the linear mixture model. First, several post-processing methods to satisfy the positivity constraint—no fraction may be negative—are described, after which a new, iterative least-squares (ILS) approach is introduced. Compared to the standard renormalisation technique, ILS is considerably more accurate; compared to constrained least-squares (CLS) ILS also prevails, because its concept is simpler. An experiment with hyperspectral data showed that ILS can be more accurate than CLS, provided its threshold is set low enough. A less rigid threshold setting results in a small decrease in accuracy, but also in a significant reduction of computational costs. The second part of Chapter 4 describes five standard methods to determine image endmembers, which must be known before the mixture model equations can be solved. In addition, a new approach based on K-means clustering is explained. An experiment with hyperspectral data made likely that the cluster means found by this unsupervised method can be regarded as relatively pure endmembers, whose quality can be slightly improved with simple post-processing techniques. The endmember spectra of K-means clustering appeared to be less pure than those found by supervised methods, but the latter methods cannot be applied without appropriate ground truth information. Of the unsupervised techniques applied, K-means clustering had the best performance and needed the least human guidance. Another experiment suggested that the value of K, which is the only parameter to be specified in advance, can be determined automatically in a fast way.

Up to this point, all subjects dealt with are related to the decomposition of individual mixed pixels. Starting with Chapter 5, however, we focus on the area estimation of agricultural fields, using knowledge that is specific for this application domain. Chapter 5 is dedicated to the discrimination between pure and mixed pixels based on the perception of an agricultural field as a homogeneous cluster of pure pixels surrounded by a layer of mixed pixels. If no ancillary information such as a topographical map is available, mixed pixels can be detected by finding the local discontinuities in the image that correspond to the layer surrounding each field. Apart from this edge detection approach, mixed pixels can also be recognised as the small regions remaining after region growing, which aims at locating the large pure pixel clusters by grouping neighbouring, spectrally similar pixels. An experiment showed that region growing can label both pure and mixed pixels correctly in 85%–90% of the cases. The performance of edge detection was about 10% lower, but this may be due to the fact that the implementation used was not optimised for the detection of mixed pixels. Inherent to edge detection, however, is that pure pixels are not grouped automatically as is done by region growing, which may be a problem for some applications. A hybrid approach integrating the results of both methods appears to be ideal as it combines two sources of information that to some extent are complementary.

Chapter 6 presents data-driven decomposition (DDD) as a new method to make accurate area estimates of agricultural fields. In correspondence with the primary aim of this thesis, the first concept of DDD is that mixed pixels are decomposed in order to divide their area between the contributing fields; pure pixels are still classified as usual. The other concept is that knowledge of the application domain is used to improve the accuracy of the area estimates. Aspects of this knowledge related to agricultural fields are that spatial information can be used to 1) separate pure and mixed pixels as described in Chapter 5, 2) determine local endmember distributions from the image, and 3) select endmembers that are probable components of a mixed pixel. A big advantage of this last aspect is that the image can be decomposed into a far greater number of endmembers than can be considered by classic pixel-based decomposition. Other aspects included in our approach are that 4) fields are often bounded by narrow structures (e.g. roads, ditches) with well-known compositions, and 5) isolated objects can be detected because their spectral response is not a mixture of that of neighbouring pixels. Experiments with artificial Landsat-TM images representing agricultural fields, narrow boundary structures, and isolated objects demonstrated that DDD can be very accurate: on average only 5% of a mixed pixel's area was allocated incorrectly. This error was largely due to natural phenomena like within-field variation and spectral confusion. The optimal value of the threshold controlling DDD agreed well with its theoretically derived setting.

Chapter 7, finally, describes two experiments designed to determine the influence of the two concepts underlying DDD on the area estimation accuracy. In the first experiment, DDD was compared to field-based classification (FBC), which does not decompose mixed pixels, pixel-based decomposition (PBD), which does not use domain-specific knowledge, and pixel-based classification (PBC), which applies neither of these concepts. On a number of artificial images, the error in accuracy per mixed pixel was on average 4.4% for DDD, 13.1% for PBD, 36.8% for FBC, and 43.5% for PBC. Decomposition of mixed pixels proved to be superior to classification, mainly because misallocation of fractions, which is intrinsic to classification, is largely avoided, but also because errors due to spectral confusion occur less frequently in a mixed than in a pure pixel model. Application of domain-specific knowledge turned out to result in more accurate area estimates as well, primarily because the endmember selection mechanism reduces the possibilities for spectral confusion. Both spectral confusion and intrinsic misallocation lead to systematic overestimation of some classes and underestimation of others; therefore, DDD is also considerably more accurate than the other methods when the results of individual pixels are aggregated to estimate the area of agricultural fields. In the second experiment, DDD was compared to FBC based on a real Landsat-TM image in order to remove possible bias resulting from the image simulation process. Compared to the results of FBC, DDD gave a more accurate estimation of the agricultural lot area in 13 out of 17 cases, while in one case their estimates were approximately equally accurate. FBC seemed to be more accurate in the remaining three cases, but this was caused by coincidentally opposing instances of spectral confusion or the way in which the results were evaluated. Due to the limitations imposed by the ground truth information, the only source of inaccuracy that could be studied was spectral confusion. In accordance with the first experiment, we found that this phenomenon affects FBC much more than DDD, though this conclusion could be verified only qualitatively.

8.2 Evaluation

In retrospect, it can be said that the first objective of this thesis—see Section 1.4.2 for a precise formulation—has been fully accomplished, since a comparison between the accuracy of DDD and FBC (or PBD and PBC) shows an improvement of more than 30%. As discussed before, this increase in accuracy is caused not only by the fact that misallocation of fractions intrinsic to classification is largely avoided, but also because spectral confusion is less apparent in a mixed pixel model. However, it must be noted that the assumption that most mixed pixels lie on the boundary of two agricultural fields and therefore consist of only two classes was not always true, because in many cases a narrow boundary structure was present. Consequently, the largest fraction could have been below 50%, which made an improvement of more than 25% per mixed pixel possible. The influence of using knowledge of the application domain is less pronounced, but unmistakably positive: a comparison between DDD and PBD or between FBC and PBC revealed an increase in accuracy of 5%–10%. Overall, DDD was about 40% more accurate than PBC. Unfortunately, these percentages could only be determined using artificial data, but the experiment based on a real Landsat-TM image made very likely that the accuracy of DDD is relatively high in practice as well. Therefore, it can be concluded that DDD is a method to estimate the area of an agricultural field that is considerably more accurate than most if not all other methods used today.

In addition, we feel that the second goal of this thesis, i.e. providing a deeper knowledge about the processing of mixed pixels, has been achieved as well. The reason for this is that, unlike other studies that have appeared in the literature, this thesis covers the entire processing path, including the separation of pure and mixed pixels, the determination of endmembers, the decomposition of mixed pixels, and the post-processing of the estimated fractions. Furthermore, a new mixture model with which the correlation existing between adjacent pixels in certain images can be described has been introduced, and a number of knowledge rules about the composition of mixed pixels in agricultural satellite images have been formalised. The extent to which the various methods suggested can be applied, however, depends on the type of imagery available and the application domain for which they are used. Many decomposition methods, for instance, assume that the endmember distributions are multivariate normal, while the proposed physical mixture model appeared to be useful only when the local endmember spectra are correlated as well. Also, the detection of mixed pixels was accomplished with standard segmentation techniques, but these can be applied only if the objects of interest are considerably larger than a single pixel. Finally, the knowledge rules mentioned before are specific for the agricultural domain, although some of them may also be valid in other domains. Despite these limitations, the methods discussed in the first part of this thesis can probably be used for most applications, while most of those in the second part can be applied provided the objects in the image are relatively large. Therefore, we hope that this thesis will become a valuable source of information for all kinds of area estimation applications dealing with mixed pixels.

8.3 Future research

In the previous chapters, we have already indicated some possible improvements of this thesis and the methods proposed, which we will summarise below. Chapter 2 signalled the lack of a broad comparative study into the strong and weak points of well-known decomposition methods. Once these results are available, the selection of a suitable decomposition method will be much simpler. In Chapter 3, a method to determine the local endmember spectra $\mathbf{M}+\mathbf{E}_1$ from a pixel's neighbourhood has been described. A disadvantage of this approach, however, is that it only works if the mixed pixel is surrounded by pure pixels, which may not be the case when dealing with objects other than agricultural fields. Therefore, to increase the applicability of the environmental model, a method to estimate $\mathbf{M}+\mathbf{E}_1$ from a number of adjacent mixed pixels needs to be developed. Chapter 4 briefly mentioned that it may be desirable to adapt iterative least-squares (ILS) such that satisfaction of the sum-to-one and positivity constraints is guaranteed. More important, though, is the application of ILS to other images in order to find rules of thumb for setting its threshold, α_0, α_1, and α_2. Likewise, more experiments with other images are needed to decide whether finding the optimal value of K automatically is possible in practice. In Chapter 5, it was noticed that the edge detection implementation was not particularly well suited for our purposes. If the program is tuned for the detection of mixed pixels, it will probably become more accurate, as will be the hybrid approach combining region growing and edge detection. As before, it should be investigated whether rules of thumb for setting the various thresholds and the maximum region size can be determined. Last but not least, the experiment presented in Chapter 7, which made a comparison between DDD and three other area estimation methods based on a real Landsat-TM image, must be repeated when ground truth information of better quality becomes available. Ideally, this supervised data should give details about the composition of individual pixels, but reliable information about the exact area of certain agricultural fields will do as well. With this new data, it will be possible to quantify the improvements in accuracy achieved by DDD in practice. Furthermore, any problem DDD might encounter when handling real instead of artificial objects can be identified more easily. If such a problem results in a serious decrease in accuracy, the DDD algorithm may have to be adapted to deal with it.

In addition, we can give a number of suggestions for future research in new directions. For instance, it would be interesting to integrate some of the methods discussed earlier in this thesis with DDD. Instead of using a map-based segmentation[1], one could also apply

[1] The perfect segmentation of an artificial image is in some way map-based as well.

region growing, possibly in combination with edge detection, to calculate the endmember distributions and detect the mixed pixels to be decomposed. Although the resulting area estimates will probably become somewhat less accurate, they will have been derived fully automatically, provided a good setting of the parameters of the segmentation method can found without human intervention as well. Alternatively, the classic linear mixture modelling approach, which was used by DDD so far, could be replaced by an estimator based on the physical linear mixture model, for example, to see if a further increase in accuracy can be achieved. Another possibility to improve the performance is the introduction of additional endmembers which in fact represent mixtures of already existing endmembers. This way, errors due to spectral confusion may be reduced, because a mixed pixel that was partially allocated to a completely unrelated class before may now be largely assigned to the new endmember that is composed of the pixel's true classes. A different direction, which corresponds to our second objective, is to apply the techniques discussed in this thesis in other areas. Section 1.4.2 lists a number of possible applications which all in one way or the other measure the area (or concentration) per class. However, the decomposition of mixed pixels may also be very useful for the extraction of road networks from satellite imagery, for instance. An even more challenging enterprise is to use the information acquired with mixed pixel decomposition to create an image of a finer spatial resolution. Although some results in this direction have already been published [20, 89], the subject is still that new and relatively unexplored that major contributions to this field are well possible.

Appendix A

Maximum likelihood classification

The most widely used method to classify remote sensing data is *maximum likelihood classification* [13, 19]. This supervised classification technique assumes that the distribution of each class i is multivariate normal, meaning that each distribution is completely described by a mean vector $\mathbf{m}_i$ representing the reflectance typical for that class, and a variance-covariance matrix $\mathbf{N}_i$, which models the within-class variation due to natural variability. In general, these parameters are determined using a training set of pure pixels:

$$\mathbf{m}_i = \frac{1}{p_i}\sum_{j=1}^{p_i}\mathbf{x}^{(i_j)} \quad \text{and} \tag{A.1}$$

$$\mathbf{N}_i = \frac{1}{p_i-1}\sum_{j=1}^{p_i}(\mathbf{x}^{(i_j)}-\mathbf{m}_i)(\mathbf{x}^{(i_j)}-\mathbf{m}_i)^T, \tag{A.2}$$

where $\mathbf{x}^{(i_j)}$ denotes the j-th pixel of class i and p_i equals the total number of pixels belonging to class i. During classification, a pixel of unknown composition is assigned to the class from whose distribution vector $\mathbf{x}$ is most likely to be drawn. These probabilities can be calculated by evaluating the probability density function (PDF) for each class, which in case of multivariate normal distributions is given by ($|\,|$ is the determinant operator):

$$p(\mathbf{x}|i) = 2\pi^{-n/2}\,|\mathbf{N}_i|^{-1/2}\exp(-\tfrac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T\mathbf{N}_i^{-1}(\mathbf{x}-\mathbf{m}_i)). \tag{A.3}$$

As is shown in Figure A.1, for n=2 the PDF resembles a mountain with a peak at $\mathbf{m}_i$ and a slope depending on $\mathbf{N}_i$. Instead of maximising $p(\mathbf{x}|i)$, one often chooses to minimise

$$(\mathbf{x}-\mathbf{m}_i)^T\mathbf{N}_i^{-1}(\mathbf{x}-\mathbf{m}_i) + \ln|\mathbf{N}_i|, \tag{A.4}$$

which equals $-2\ln(p(\mathbf{x}|i))$ minus constants. The first term in Equation (A.4) is the single deciding factor if all variance-covariance matrices are (approximately) equal and is commonly referred to as the *Mahalanobis distance* between $\mathbf{x}$ and $\mathbf{m}_i$. Pixels lying at a particular Mahalanobis distance of the mean of class i make up an ellipse; the shape of these ellipses can be found by intersecting the i-th PDF mountain at the corresponding heights—see Figure A.1.

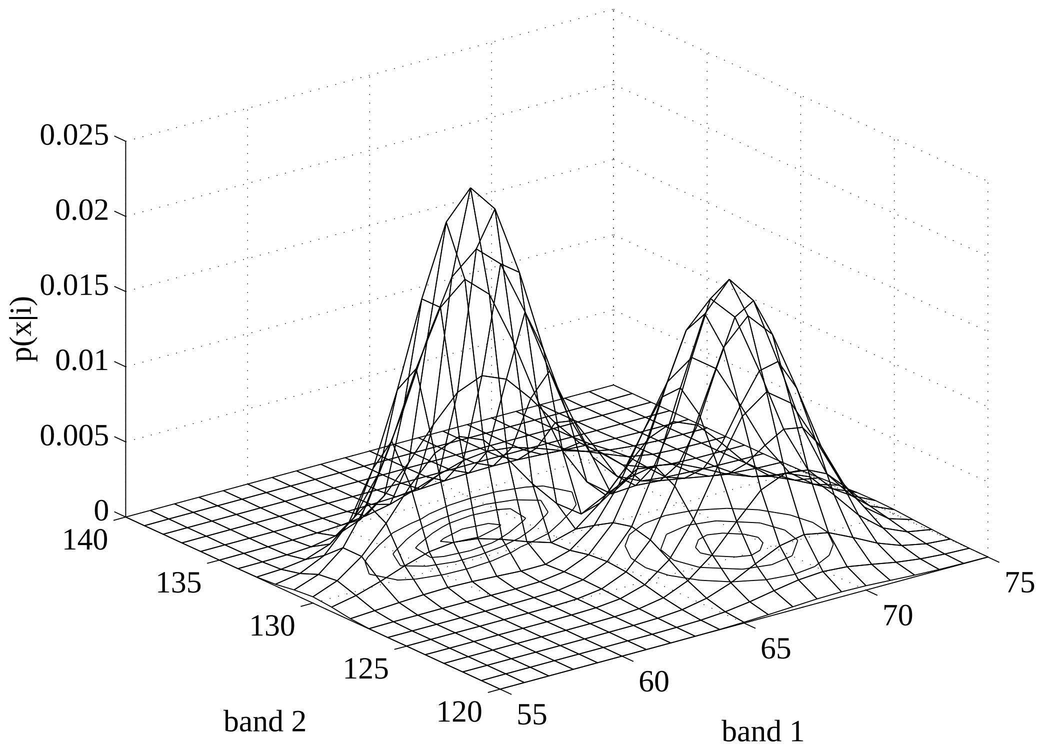


Figure A.1: *Probability density function of two multivariate (n=2) normal distributions. Class 1 is defined by* $\mathbf{m}_1$*=[63; 132] and* $\mathbf{N}_1$*=[16 4; 4 4], while class 2 is given by* $\mathbf{m}_2$*=[69; 126] and* $\mathbf{N}_2$*=[9 0; 0 9]. The ellipses show which pixels lie at the same height of a PDF, i.e. have equal probability of belonging to class i. Consequently, all pixels on a particular ellipse have the same Mahalanobis distance to* $\mathbf{m}_i$*.*

Appendix B

Error backpropagation in a MLP

Backpropagation is a supervised technique of training, which means that the training procedure needs to know the class or composition of the pixels in the training set. If the MLP is used as a classifier for c classes, each pixel of class i is paired up with an c-dimensional target vector where all but one elements are low and the i-th element is high; during the classification stage, the class of an input vector is determined by the element of the output vector that has the highest value. If the network is used to decompose mixed pixels, the target vector of each training pixel is its true proportions vector, and the output of the MLP during the decomposition stage is taken as an estimate of the fractions of the pixel presented. Training the network involves adjusting the weights, which are initialised at small random values, until the network approximates the desired output closely enough.

A measure of how much a network (having weight vector $\mathbf{w}$) used for decomposition is deviating from the desired performance is expressed by the error function

$$E(\mathbf{w}) = \frac{1}{2} \sum_p \sum_i (f_i^{(p)} - o_i^{(p)})^2, \tag{B.1}$$

where $\mathbf{f}$ and $\mathbf{o}$ represent the desired and output fractions vector, i ranges over the output units and p over the training vectors. In this view, training coincides with minimising the error function. This is achieved by iteratively changing the weights using gradient descent:

$$\Delta w_{ij} = -\eta \frac{\partial E(\mathbf{w})}{\partial w_{ij}}, \tag{B.2}$$

where the learning rate η controls the speed of learning. While in Equation (B.2) the weights are adjusted after all training vectors have been presented, i.e. after an entire iteration, usually the weights are changed after presentation of a single vector. If the vectors are drawn at random from the training set, the training procedure is extended with stochastic noise, which improves the performance of the network [48]. For this strategy, it can be derived [48, 13] that in case of a sigmoidal activation function

$$\Delta w_{ij} = \eta o_i \delta_j, \quad \delta_j = \begin{cases} (f_j - o_j) o_j (1 - o_j), & j \text{ is output unit} \\ (\sum_k \delta_k w_{jk}) o_j (1 - o_j), & \text{otherwise.} \end{cases} \tag{B.3}$$

An update rule for the thresholds θ_j can be derived in a similar way. It is this mechanism where the δ of a neuron is calculated using the δs of the neurons in the next layer that has provided the name (error) backpropagation.

One well-known extension of the basic backpropagation procedure, whose use is more or less standard practice nowadays, is the addition of a momentum term:

$$\Delta w_{ij}^{(t+1)} = -\eta \frac{\partial E(\mathbf{w})}{\partial w_{ij}} + \alpha \Delta w_{ij}^{(t)}. \tag{B.4}$$

In this equation, t represents the time at which the network is adjusted and parameter α ($0 \leq \alpha \leq 1$) determines the influence of the momentum term. Thus, changes made in the previous update of the weights are still reckoned with during the present update. The result of this strategy is that in case the gradient of the error function is constant for a longer period, the learning rate is effectively increased from η to approximately $\eta/(1-\alpha)$; think of a ball picking up speed as it rolls down a hill. However, the risk of a high learning rate—the network can show oscillatory behaviour—is avoided because successive contributions from the momentum term will tend to cancel when a minimum of the error function is reached and the sign of the gradient changes. Therefore, using a momentum term is advantageous as it can lead to faster convergence towards the minimum of Equation (B.1) without causing divergent oscillations. There exist many more variations on the training procedure such as weight decay and distortion of the training vectors, but they are not described here. For an explanation of these and other variations the reader is referred to [13].

Bibliography

[1] J.B. Adams, D.E. Sabol, V. Kapos, R.A. Filho, D.A. Roberts, M.O. Smith, and A.R. Gillespie. Classification of multispectral images based on fractions of endmembers: Application to land-cover change in the Brazilian Amazon. *Remote Sensing of Environment*, 52:137–154, 1995.

[2] J.B. Adams, M.O. Smith, and P.E. Johnson. Spectral mixture modeling: A new analysis of rock and soil types at the Viking Lander 1 site. *Journal of Geophysical Research*, 91(B8):8098–8112, 1986.

[3] M. Ait Belaid, G. Edwards, A. Jaton, and K.P.B. Thomson. Post-segmentation classification of images containing small agricultural fields. *Geocarto International*, 3:53–60, 1992.

[4] C. van Alem. Satelliet houdt oogje op landbouwsubsidies. *De Gelderlander*, June 4th, 1996.

[5] K. Arai and Y. Terayama. Label relaxation using a linear mixture model. *International Journal of Remote Sensing*, 13(16):3217–3227, 1992.

[6] K. Arai, Y. Terayama, Y.Ueda, and M. Moriyama. Adaptive least squares method for estimation of partial cloud coverage within a pixel. *International Journal of Remote Sensing*, 16(12):2197–2206, 1995.

[7] P.M. Atkinson, M.E.J. Cutler, and H. Lewis. Mapping sub-pixel proportional land cover with AVHRR imagery. *International Journal of Remote Sensing*, 18(4):917–935, 1997.

[8] D.H. Ballard and C.M. Brown. *Computer Vision.* Prentice-Hall, Inc., Englewood Cliffs, 1992.

[9] F. Baret, J.G.P.W. Clevers, and M.D. Steven. The robustness of canopy gap fraction estimates from red and near-infrared reflectances: A comparison of approaches. *Remote Sensing of Environment*, 54:141–151, 1995.

[10] A. Bateson and B. Curtiss. A method for manual endmember selection and spectral unmixing. *Remote Sensing of Environment*, 55:229–243, 1996.

[11] J.A. Benediktsson, P.H. Swain, and O.K. Ersoy. Neural network approaches versus statistical methods in classification of multisource remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 28(4):540–552, 1990.

[12] J.C. Bezdek, R. Ehrlich, and W. Full. FCM: The fuzzy c-means clustering algorithm. *Computers and Geosciences*, 10:191–203, 1984.

[13] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press Inc., Oxford, 1995.

[14] C.C. Borel and S.A.W. Gerstl. Nonlinear spectral mixing models for vegetative and soil surfaces. *Remote Sensing of Environment*, 47:403–416, 1994.

[15] P. Bosdogianni, M. Petrou, and J. Kittler. Mixed pixel classification in remote sensing. In *Proceedings of the 1st European Symposium on Satellite Remote Sensing, conference on Image and Signal Processing for Remote Sensing*, pages 494–505, 1994. Rome.

[16] M. van den Broek. De koolzaadpolitie kijkt toe uit de ruimte. *Volkskrant*, June 3rd, 1995.

[17] P.J. Brown. Multivariate calibration. *Journal of the Royal Statistical Society*, 44(3):287–321, 1982.

[18] R.G. Bryant. Validated linear mixture modelling of Landsat TM data for mapping evaporite minerals on a playa surface: methods and applications. *International Journal of Remote Sensing*, 17(2):315–330, 1996.

[19] J.B. Campbell. *Introduction to remote sensing*. Taylor & Francis Ltd, London, 2nd edition, 1996.

[20] M.J. Carlotto. Non-linear mixture model and application for enhanced resolution multispectral classification. In *Proceedings of the 1995 IEEE International Geoscience and Remote Sensing Symposium*, pages 1168–1170. IEEE 95CH35770, 1995. Firenze.

[21] D.L. Civco and Y. Wang. Classification of multispectral, multitemporal, multisource spatial data using artificial neural networks. In L.D. Whitmill, editor, *Mapping and Monitoring the Earth's Environments for a Balanced Future*, pages 123–133. ASPRS/ACSM, 1994. Reno.

[22] P.F. Crapper. An estimate of the number of boundary cells in a mapped landscape coded to grid cells. *Photogrammetric Engineering and Remote Sensing*, 50(10):1497–1503, 1984.

[23] A.M. Cross, J.J. Settle, N.A. Drake, and R.T.M. Paivinen. Subpixel measurement of tropical forest cover using AVHRR data. *International Journal of Remote Sensing*, 12(5):1119–1129, 1991.

[24] N.A. Drake and J.J. Settle. Linear mixture modelling of Thematic Mapper data of the Peruvian Andes. In *Proceedings of the EARSeL Symposium*, pages 490–495, 1989. Helsinki, Finland.

[25] G.M. Foody. Ordinal-level classification of sub-pixel tropical forest cover. *Photogrammetric Engineering and Remote Sensing*, 60(1):61–65, 1994.

[26] G.M. Foody. Relating the land-cover composition of mixed pixels to artificial neural network classification output. *Photogrammetric Engineering and Remote Sensing*, 62(5):491–499, 1996.

[27] G.M. Foody. Fully fuzzy supervised classification of land cover from remotely sensed imagery with an artificial neural network. *Neural Computing & Applications*, 5:238–247, 1997.

[28] G.M. Foody. Land cover mapping from remotely sensed data with a neural network: accomodating fuzziness. In I. Kanellopoulos, G.G. Wilkinson, F. Roli, and J. Austin, editors, *Neurocomputation in Remote Sensing Data Analysis: Proceedings of Concerted Action COMPARES*, pages 28–37, 1997.

[29] G.M. Foody and D.P. Cox. Sub-pixel land cover composition estimation using a linear mixture model and fuzzy membership functions. *International Journal of Remote Sensing*, 15(3):619–631, 1994.

[30] G.M. Foody, R.M. Lucas, P.J. Curran, and M. Honzak. Non-linear mixture modelling without end-members using an artificial neural network. *International Journal of Remote Sensing*, 18(4):937–953, 1997.

[31] P.G. Foschi. A geometric approach to a mixed pixel problem: Detecting subpixel woody vegetation. *Remote Sensing of Environment*, 50:317–327, 1994.

[32] F.J. Gallego, J. Delince, and C. Rueda. Crop area estimates through remote sensing: stability of the regression correction. *International Journal of Remote Sensing*, 14(18):3433–3445, 1993.

[33] M.S. klein Gebbinck. Ultrasonic tissue characterization using neural networks. Master's thesis, Department of Informatics, University of Nijmegen, The Netherlands, 1993.

[34] M.S. klein Gebbinck and Th.E. Schouten. Decomposition of mixed pixels. In J. Desachy, editor, *Image and Signal Processing for Remote Sensing II*, pages 104–115. SPIE 2579, 1995. Paris.

[35] M.S. klein Gebbinck and Th.E. Schouten. Accurate area estimation by data-driven decomposition of mixed pixels. Technical Report CSI-R9622, Computing Science Institute, University of Nijmegen, 1996.

[36] M.S. klein Gebbinck and Th.E. Schouten. Application of data-driven decomposition to Landsat-TM images for crop area estimation. Technical Report CSI-R9626, Computing Science Institute, University of Nijmegen, 1996.

[37] M.S. klein Gebbinck and Th.E. Schouten. Area estimation with subpixel accuracy for industrial imaging systems. Technical Report CSI-R9711, Computing Science Institute, University of Nijmegen, 1997.

[38] M.S. klein Gebbinck and Th.E. Schouten. Crop area estimation by data-driven decomposition of mixed pixels in Landsat-TM images. *International Journal of Remote Sensing*, accepted.

[39] M.S. klein Gebbinck, Th.E. Schouten, and J.J. Settle. Decomposition of mixed pixels based on a physical linear mixture model. *International Journal of Remote Sensing*, submitted.

[40] M.S. klein Gebbinck, J.T.M. Verhoeven, J.M. Thijssen, and Th.E. Schouten. Application of neural networks for the classification of diffuse liver disease by quantitative echography. *Ultrasonic Imaging*, 15:205–217, 1993.

[41] P. Gong, J.R. Miller, J. Freemantle, and B. Chen. Spectral decomposition of Landsat Thematic Mapper data for urban land-cover mapping. In *Proceedings of the 14th Canadian Symposium on Remote Sensing*, pages 458–461, 1991. Calgary, Alberta, Canada.

[42] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Addison-Wesley Publishing Company, Inc., 1992.

[43] F. Gonzalez-Alonso and J.M. Cuevas. Remote sensing and agricultural statistics: crop area estimation through regression estimators and confusion matrices. *International Journal of Remote Sensing*, 14(6):1215–1219, 1993.

[44] R. Grazier. Detection of mixed pixels. Master's thesis, Department of Informatics, University of Nijmegen, The Netherlands, 1998.

[45] J. Grunblatt. An MTF analysis of Landsat classification error at field boundaries. *Photogrammetric Engineering and Remote Sensing*, 53(6):639–643, 1987.

[46] A.R. Harrison, S.J. Hurcom, and M. Taberner. Examining spectral mixture modelling using in situ spectroradiometric measurements. In *Proceedings of the SPATIAL DATA 2000 Conference*, pages 242–249, 1991. Christ Church, Oxford, England.

[47] G.F. Hepner, T. Logan, N. Ritter, and N. Bryant. Artificial neural network classification using a minimal training set: Comparison to conventional supervised classification. *Photogrammetric Engineering and Remote Sensing*, 56(4):469–473, 1990.

[48] J. Hertz, A. Krogh, and R.G. Palmer. *Introduction to the Theory of Neural Computation*. Addison-Wesley Publishing Company, Inc., 1991.

[49] C.A. Hlavka and M.A. Spanner. Unmixing AVHRR imagery to assess clearcuts and forest regrowth in oregon. *IEEE Transactions on Geoscience and Remote Sensing*, 33(3):788–795, 1995.

[50] B.N. Holben and Y.E. Shimabukuro. Linear mixing model applied to coarse spatial resolution data from multispectral satellite sensors. *International Journal of Remote Sensing*, 14(11):2231–2240, 1993.

[51] H.M. Horwitz, R.F. Nalepka, P.D. Hyde, and J.P. Morgenstern. Estimating the proportions of objects within a single resolution element of a multispectral scanner. In *Proceedings of the 7th International Symposium on Remote Sensing of Environment*, pages 1307–1320, 1971. Ann Arbor, Michigan.

[52] J.R. Irons, B.L. Markham, R.F. Nelson, D.L. Toll, D.L. Williams, R.S. Latty, and M.L. Stauffer. The effects of spatial resolution on the classification of Thematic Mapper data. *International Journal of Remote Sensing*, 6(8):1385–1403, 1985.

[53] E. Ishiguro, M.K. Kumar, Y. Hidaka, S. Yoshida, M. Sato, and M. Miyazato andJ.Y. Chen. Use of rice response characteristics in area estimation by LANDSAT/TM and MOS-1 satellites data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 48(1):26–32, 1993.

[54] L.L.F. Janssen. *Methodology for updating terrain object data from remote sensing data*. PhD thesis, Wageningen Agricultural University, The Netherlands, 1994.

[55] M.F. Jasinski. Estimation of subpixel vegetation density of natural regions using satellite multispectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 34(3):804–813, 1996.

[56] C.O. Justice, B.L. Markham, J.R.G. Townshend, and R.L. Kennard. Spatial degradation of satellite data. *International Journal of Remote Sensing*, 10(9):1539–1561, 1989.

[57] I. Kanellopoulos, A. Varfis, G.G. Wilkinson, and J. Mégier. Classification of remotely-sensed satellite images using multi-layer perceptron networks. *Artificial Neural Networks*, 1:1067–1070, 1991.

[58] I. Kanellopoulos and G.G. Wilkinson. Strategies and best practice for neural network image classification. *International Journal of Remote Sensing*, 18(4):711–725, 1997.

[59] J.T. Kent and K.V. Mardia. Spectral classification using fuzzy membership models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):659–671, 1988.

[60] H. Kerdiles and M.O. Grondona. NOAA-AVHRR NDVI decomposition and subpixel classification using linear mixing in the Argentinean Pampa. *International Journal of Remote Sensing*, 16(7):1303–1325, 1995.

[61] C.D. Kershaw and R.M. Fuller. Statistical problems in the discrimination of land cover from satellite images: a case study in lowland Britain. *International Journal of Remote Sensing*, 13(16):3085–3104, 1992.

[62] R.L. Kettig and D.A. Landgrebe. Classification of multispectral image data by extraction and classification of homogeneous objects. In *Proceedings of the Symposium on Machine Classification of Remotely Sensed Data*, pages 2A1–2A11, 1975.

[63] E.J. van Kootwijk, H. van der Voet, and J.J.M. Berdowski. Estimation of ground cover composition per pixel after matching image and ground data with subpixel accuracy. *International Journal of Remote Sensing*, 16(1):97–111, 1995.

[64] F. van der Laan, M. Hassani, B. van de Pas, and A. van Dijk. Landbouwfraude-identificatie met teledetectie eindelijk door nederlandse bedrijven uitgevoerd. *Remote Sensing Nieuwsbrief*, 18(63):38–40, 1995.

[65] R.K. Lennington, C.T. Sorensen, and R.P. Heydorn. A mixture model approach for estimating crop areas from Landsat data. *Remote Sensing of Environment*, 14:197–206, 1984.

[66] T.M. Lillesand and R.W. Kiefer. *Remote sensing and image interpretation*. John Wiley, New York, 3rd edition, 1994.

[67] B.L. Markham and J.R.G. Townshend. Land cover classification accuracy as a function of sensor spatial resolution. In *Proceedings of the 15th International Symposium on Remote Sensing of Environment*, pages 1075–1085, 1981. Ann Arbor, Michigan.

[68] S.E. Marsh, P. Switzer, and W.S. Kowalik. Resolving the percentage of component terrains within single resolution elements. *Photogrammetric Engineering and Remote Sensing*, 46(8):1079–1086, 1980.

[69] F. Maselli, A. Rodolfi, and C. Conese. Fuzzy classification of spatially degraded Thematic Mapper data for the estimation of sub-pixel components. *International Journal of Remote Sensing*, 17(3):537–551, 1996.

[70] P.M. Mather. Theoretical problems in image classification. In JA Clark and MD Stevens, editors, *Application of RS in Agriculture*, pages 127–135. Butterworths, London, 1990.

[71] K.R. McCloy and K.A. Hall. Mapping the density of woody vegetative cover using Landsat MSS digital data. *International Journal of Remote Sensing*, 12(9):1877–1885, 1991.

[72] F. van der Meer. Spectral unmixing of Landsat Thematic Mapper data. *International Journal of Remote Sensing*, 16(16):3189–3194, 1995.

[73] L.A.K. Mertes, M.O. Smith, and J.B. Adams. Estimating suspended sediment concentrations in surface waters of the Amazon river wetlands from Landsat images. *Remote Sensing of Environment*, 43:281–301, 1993.

[74] M.D. Metzler and R.C. Cicone. Assessment of technologies for classification of mixed pixels. In *Proceedings of the 17th International Symposium on Remote Sensing of Environment*, pages 1015–1021, 1983. Ann Arbor, Michigan.

[75] R.H. Myers. *Classical and modern regression with applications*. PWS-Kent Publishing Company, Boston, 2nd edition, 1990.

[76] V.S. Nalwa. *A guided tour of computer vision.* Addison-Wesly Publishing Company, 1993.

[77] E.M. Novo and Y.E. Shimabukuro. Spectral mixture analysis of inland tropical waters. *International Journal of Remote Sensing*, 15(6):1351–1356, 1994.

[78] K.W. Oleson, S. Sarlin, J. Garrison, S. Smith, J.L. Privette, and W.J. Emery. Unmixing multiple land-cover type reflectances from coarse spatial resolution satellite data. *Remote Sensing of Environment*, 54:98–112, 1995.

[79] R.P. Pech, A.W. Davies, R.R. Lamcroft, and R.D. Graetz. Calibration of LANDSAT data for sparsely vegetated semi-arid rangelands. *International Journal of Remote Sensing*, 7(12):1729–1750, 1986.

[80] V. Pestemalci, U. Dinc, I. Yegingil, M. Kandirmaz, M.A. Cullu, N. Ozturk, and E. Aksoy. Acreage estimation of wheat and barley fields in the province of Adana, Turkey. *International Journal of Remote Sensing*, 16(6):1075–1085, 1995.

[81] N. Quarmby, G. Foody, and I. MacLaren. Classifying land use. *Image Processing*, 6(3):28–30, 1994.

[82] N.A. Quarmby, J.R.G. Townshend, J.J. Settle, K.H. White, M. Milnes, T.L. Hindle, and N. Silleos. Linear mixture modelling applied to AVHRR data for crop area estimation. *International Journal of Remote Sensing*, 13(3):415–425, 1992.

[83] T.W. Ray and B.C. Murray. Nonlinear spectral mixing in desert vegetation. *Remote Sensing of Environment*, 55:59–64, 1996.

[84] A.J. Richardson, C.L. Wiegand, H.W. Gausman, J.A. Cuellar, and A.H. Gerbermann. Plant, soil, and shadow reflectance components of row crops. *Photogrammetric Engineering and Remote Sensing*, 41(11):1401–1407, 1975.

[85] T.V. Robertson. Extraction and classification of objects in multispectral images. In *Proceedings of the Annual Symposium on Machine Processing of Remotely Sensed Data*, pages 3B–27–3B–34, 1973. Purdue University, West Lafayette, Indiana.

[86] W. Rosenthal and J. Dozier. Automated mapping of montane snow cover at subpixel resolution from the LANDSAT Thematic Mapper. *Water Resources Research*, 32(1):115–130, 1996.

[87] P.L. Rosin. Refining region estimates for post-processing image classification. In J. Desachy, editor, *Image and Signal Processing for Remote Sensing*, pages 214–224. SPIE 2315, 1994. Rome.

[88] F.F. Sabins. *Remote sensing.* Remote Sensing Enterprises, Inc., 2nd edition, 1987.

[89] W. Schneider. Land use mapping with subpixel accuracy from LANDSAT TM image data. In *Proceedings of the 25th International Symposium on Remote Sensing and Global Environmental Change*, pages II–155–II–161, 1993. Graz, Austria.

[90] R.P.H.M. Schoenmakers. *Integrated methodology for segmentation of large optical satellite images in land applications of remote sensing*. PhD thesis, Department of Informatics, University of Nijmegen, The Netherlands, 1995.

[91] C.G.J. Schotten, W.W.L. van Rooy, and L.L.F. Janssen. Assessment of the capabilities of multi-temporal ERS-1 SAR data to discriminate between agricultural crops. *International Journal of Remote Sensing*, 16(14):2619–2637, 1995.

[92] Th.E. Schouten and M.S. klein Gebbinck. Quality measures for image segmentation using generated images. In J. Desachy, editor, *Image and Signal Processing for Remote Sensing II*, pages 411–422. SPIE 2579, 1995. Paris.

[93] Th.E. Schouten and M.S. klein Gebbinck. Fuzzy classification of spectra. In *Neural networks: Best practice in Europe*, pages 198–201. SNN, 1997. Amsterdam.

[94] Th.E. Schouten and M.S. klein Gebbinck. A neural network approach to spectral mixture analysis. In I. Kanellopoulos, G.G. Wilkinson, F. Roli, and J. Austin, editors, *Neurocomputation in Remote Sensing Data Analysis: Proceedings of Concerted Action COMPARES*, pages 79–85, 1997.

[95] Th.E. Schouten, M.S. klein Gebbinck, Z.K. Liu, and S.W. Chen. Fast algorithm for spectral mixture analysis of imaging spectrometer data. In J. Desachy, editor, *Image and Signal Processing for Remote Sensing III*, pages 198–207. SPIE 2955, 1996. Taorminia.

[96] Th.E. Schouten, M.S. klein Gebbinck, R.P.H.M. Schoenmakers, and G.G. Wilkinson. Finding thresholds for image segmentation. In J. Desachy, editor, *Image and Signal Processing for Remote Sensing*, pages 90–101. SPIE 2315, 1994. Rome.

[97] Th.E. Schouten, M.S. klein Gebbinck, J.M. Thijssen, and J.T.M. Verhoeven. Ultrasonic tissue characterisation using neural networks. In *Proceedings of the Conference on Artificial Neural Networks*, pages 110–112, 1993. Brighton.

[98] R.A. Schowengerdt. *Techniques for Image Processing and Classification in Remote Sensing*. Academic Press, Inc., 1983.

[99] J.J. Settle and N.A. Drake. Linear mixing and the estimation of ground cover proportions. *International Journal of Remote Sensing*, 14(6):1159–1177, 1993.

[100] Y.E. Shimabukuro and J.A. Smith. The least-squares mixing models to generate fraction images derived from remote sensing multispectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 29(1):16–20, 1991.

[101] R. Sigman and M. Craig. Potential utility of Thematic Mapper data in estimating crop areas. In *Proceedings of the 15th International Symposium on Remote Sensing of Environment*, pages 1057–1064, 1981. Ann Arbor, Michigan.

[102] R.B. Singer and T.B. McCord. Mars: Large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In *Proceedings of the 10th Lunar and Planetary Science Conference*, pages 1835–1848, 1979.

[103] M.O. Smith, P.E. Johnson, and J.B. Adams. Quantitative determination of mineral types and abundances from reflectance spectra using principal components analysis. *Journal of Geophysical Research*, 90:C797–C804, 1985.

[104] M.O. Smith, S.L. Ustin, J.B. Adams, and A.R. Gillespie. Vegetation in deserts: I. a regional measure of abundance from multispectral images. *Remote Sensing of Environment*, 31:1–26, 1990.

[105] M.O. Smith, S.L. Ustin, J.B. Adams, and A.R. Gillespie. Vegetation in deserts: II. environmental influences on regional abundance. *Remote Sensing of Environment*, 31:27–52, 1990.

[106] J.H.T. Stakenborg. Per-field classification of a segmented SPOT simulated image. In *Proceedings of the Symposium on Remote Sensing for Resources Development and Environmental Management*, pages 73–78, 1986. Enschede, The Netherlands.

[107] G. Strang. *Linear algebra and its applications.* Academic Press, New York, 2nd edition, 1980.

[108] J.M. Thijssen, H. Huisman, M.S. klein Gebbinck, J.T.M. Verhoeven, and Th.E. Schouten. Neural networks for the echographic diagnosis of diffuse liver diseases. In *Proceedings of the International Conference on Artificial Neural Networks*, page 948, 1993. Amsterdam.

[109] G. Thomas, S.E. Hobbs, and M. Dufour. Woodland area estimation by spectral mixing: applying a goodness-of-fit solution method. *International Journal of Remote Sensing*, 17(2):291–301, 1996.

[110] J.L. Thomson and J.W. Salisbury. The mid-infrared reflectance of mineral mixtures (7–14 μm). *Remote Sensing of Environment*, 45:1–13, 1993.

[111] F. Wang. Fuzzy supervised classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 28(2):194–201, 1990.

[112] H.D. Williamson. Estimating sub-pixel components of a semi-arid woodland. *International Journal of Remote Sensing*, 15(16):3303–3307, 1994.

[113] P. Wolfe. The simplex method of quadratic programming. *Econometrica*, 27:382–398, 1959.

[114] C.E. Woodcock and A.H. Strahler. The factor of scale in remote sensing. *Remote Sensing of Environment*, 21:311–332, 1987.

[115] H.P. Wu and R.A. Schowengerdt. Improved estimation of fraction images using partial image restoration. *IEEE Transactions on Geoscience and Remote Sensing*, 31(4):771–778, 1993.

Acronyms

ANN	*Artificial Neural Network*
AVHRR	*Advanced Very High Resolution Radiometer*
AVIRIS	*Airborne Visible InfraRed Imaging Spectrometer*
CCD	*Charge-Coupled Device*
CLS	*Constrained Least-Squares*
CPU	*Central Processing Unit*
CSI	*Computing Science Institute*
DDD	*Data-Driven Decomposition*
ECHO	*Extraction and Classification of Homogeneous Objects*
EU	*European Union*
FBC	*Field-Based Classification*
GCP	*Ground Control Point*
GIS	*Geographical Information System*
IFOV	*Instantaneous Field Of View*
ILS	*Iterative Least-Squares*
LAC	*Local Area Coverage mode*
LBE	*Likelihood of Being an Edge*
MAIS	*Modular Airborne Imaging Spectrometer*
MARS	*Monitoring Agriculture with Remote Sensing*
MLP	*Multi-Layer Perceptron*
MSS	*Multi-Spectral Scanner*
NOAA	*National Oceanic and Atmospheric Administration*
NASA	*National Aeronautics and Space Administration*
P	*Panchromatic*
PBC	*Pixel-Based Classification*
PBD	*Pixel-Based Decomposition*
PCA	*Principal Component Analysis*
PDF	*Probability Density Function*
RGB	*Red Green Blue*
RMS	*Root Mean Square*
SOIM	*Simultaneous Order-independent Iterative Merging*
SPOT	*Satellite Pour l'Observation de la Terre*
TM	*Thematic Mapper*
XS	*multi("X")Spectral*

Index

Samenvatting

Voor de Europese Unie (EU) is het van groot belang de oppervlakte van agrarische velden automatisch te kunnen schatten. Sinds 1992 is de hoogte van de subsidie die een agrariër bij de EU kan aanvragen namelijk afhankelijk van het soort en de oppervlakte van het gewas dat hij verbouwt. Jaarlijks geeft de EU ongeveer 75 miljard gulden uit aan landbouwsubsidies, waarvan naar schatting zo'n 5 miljard gulden wordt besteed aan onjuiste aanvragen. Zo kan het gebeuren dat een agrariër opgeeft koolzaad—gesubsidieerd maar niet erg winstgevend—te verbouwen, terwijl hij in werkelijkheid aardappels—winstgevend maar niet gesubsidieerd—verbouwt. Het komt echter vaker voor dat hij de oppervlakte van zijn velden (al dan niet bewust) overschat, bijvoorbeeld door de sloten rondom een veld mee te tellen. Om het enorme bedrag dat onterecht wordt uitgegeven zo klein mogelijk te houden, zijn de lidstaten verplicht minstens 5% van alle aanvragen te controleren. Aangezien de EU 6 miljoen agrariërs telt die samen 150 miljoen percelen bebouwen is het ondoenlijk deze taak door controleurs in het veld te laten uitvoeren. Daarbij komt nog dat op veel percelen meerdere gewassen per jaar worden verbouwd, zodat er regelmatig gecontroleerd moet worden. Vandaar dat de EU gekozen heeft voor een geautomatiseerd systeem dat gebruik maakt van *remote sensing* beelden die vanaf een satelliet worden opgenomen. Elk beeld beslaat een groot gebied, zodat veel aanvragen op een relatief goedkope manier onderzocht kunnen worden: alleen die velden waarvan de oppervlakte schatting (door het systeem) niet overeenkomt met de opgave (door de agrariër) hoeven door een controleur te worden bezocht. Bovendien kan de controle geregeld herhaald worden omdat de satelliet met een zekere regelmaat over hetzelfde gebied komt.

Het doel van dit proefschrift, zie **hoofdstuk 1**, kan het best uitgelegd worden aan de hand van het huidige systeem voor oppervlakteschatting. In het algemeen werkt zo'n systeem door elke *pixel* toe te wijzen aan de klasse waar hij het meeste op lijkt. De oppervlakte per gewas wordt vervolgens bepaald door het aantal pixels per klasse te tellen. Deze aanpak geeft echter onnauwkeurige resultaten wanneer het beeld veel gemengde pixels bevat: aangezien een gemengde pixel meer dan één gewas beslaat, zal zijn *classificatie* het oppervlak van de gekozen klasse overschatten en dat van de andere klassen onderschatten. Een andere aanpak is het achterhalen van de samenstelling van de gemengde pixel, waarna zijn oppervlak verdeeld wordt over meerdere klassen. Onder de aanname dat de meeste gemengde pixels op de grens van twee agrarische velden liggen, is de nauwkeurigheid van de laatstgenoemde *decompositie* aanpak naar verwachting gemiddeld 25% per gemengde pixel nauwkeuriger. Behalve het ontwikkelen van zo'n nauwkeurige, op decompositie gebaseerde oppervlakteschattingsmethode voor agrarische velden, probeert dit proefschrift tevens meer kennis over het verwerken van gemengde pixels in het algemeen te verschaffen, dat wil zeggen (relatief) onafhankelijk van de bestudeerde toepassing en de gebruikte beelden.

Als een eerste stap is de literatuur bestudeerd om te kijken of reeds bestaande technieken gebruikt kunnen worden. **Hoofdstuk 2** bespreekt een 12-tal verschillende decompositiemethoden, ook wel schatters genaamd, die op sommige punten veel op elkaar lijken. De veruit meest gebruikte schatter is de analytische oplossing van het *statistische lineaire mixmodel*. Aan de hand van de zogenaamde *endmember distributies*, welke min of meer overeenkomen met het gemiddelde en de variantie van de verschillende klassen, kan deze gegeneraliseerde kleinste-kwadraten methode voor elke pixel de bijdrage van de verschillende klassen berekenen. Helaas is er tot nu toe nog geen uitgebreide vergelijking van de verschillende methoden in de literatuur verschenen. Sommige kleine studies suggereren wel dat met de meer geavanceerde methoden wat betere resultaten kunnen worden bereikt, maar deze methoden vereisen in het algemeen veel meer berekeningen. Aangezien de nauwkeurigheid waarmee pixels ontbonden kunnen worden in veel gevallen meer af lijkt te hangen van de karakteristieken van de data dan van de gebruikte decompositiemethode, wordt de eenvoudige, snelle analytische schatter voor het lineaire mixmodel vandaag de dag nog steeds gebruikt voor de meeste toepassingen.

De nauwkeurigheid van het decompositieproces kan verbeterd worden door de methode toe te spitsen op het toepassingsgebied. In **hoofdstuk 3** wordt een nieuw lineair mixmodel voorgesteld waarin de correlaties die bestaan tussen de endmember spectra van naburige pixels in bijvoorbeeld agrarische beelden kunnen worden beschreven. Het oplossen van dit *fysische*, of beter, *omgevingsmodel* houdt in dat behalve het ingenomen oppervlak ook de spectrale reflectie van elke component van de gemengde pixel geschat wordt. Hoewel deze gecompliceerde operatie meer rekentijd kost dan het oplossen van het statistische model, vertegenwoordigen de geschatte component spectra informatie die niet verkregen kan worden met klassieke decompositie technieken. Bovendien bleek uit een experiment dat de proporties die geschat zijn volgens de nieuwe aanpak nauwkeuriger zijn dan de proporties die verkregen zijn met de standaard schatter. Vooral in het geval de data moeilijk te ontbinden is, kan de verbetering in nauwkeurigheid aanzienlijk zijn.

Hoofdstuk 4 behandelt twee problemen die zich voordoen bij analytische decompositiemethoden die gebaseerd zijn op het lineaire mixmodel. Eerst worden enkele bestaande post-processing methoden beschreven die de geschatte proporties van de componenten zo aanpassen dat ze altijd positief zijn, waarna een nieuwe, kleinste-kwadraten aanpak genaamd *iterative least-squares* (ILS) wordt geïntroduceerd. In vergelijking met renormalisatie is ILS aanzienlijk nauwkeuriger, terwijl in vergelijking met *constrained least-squares* (CLS), een andere, bekende kleinste-kwadraten methode, ILS eenvoudiger is. Een experiment liet zien dat ILS tevens nauwkeuriger is dan CLS indien de in te stellen drempelwaarde laag genoeg is. Het tweede deel van het hoofdstuk beschrijft een vijftal standaard methoden om de endmember distributies uit het beeld te bepalen. Daarnaast wordt een nieuwe methode gebaseerd op *K-means clustering* uitgelegd. Experimenteel werd aangetoond dat de gemiddelde reflectie van de clusters die door deze methode worden gevonden beschouwd kunnen worden als relatief pure endmembers. De kwaliteit van deze endmembers kan naderhand nog iets verbeterd worden met simpele post-processing technieken. De endmember spectra gevonden door het K-means cluster algoritme leken minder puur te zijn dan die gevonden met behulp van *supervised* methoden, maar deze kunnen alleen worden toegepast als informatie over de werkelijke structuur van het beeld bekend is. Van de toegepaste *unsupervised* technieken leverde K-means clustering de beste resultaten. Een tweede experiment suggereerde dat de waarde van K, de enige parameter die nog moet worden ingesteld, automatisch en op een snelle manier bepaald kan worden.

Hadden alle onderwerpen tot nu toe te maken met het ontbinden van individuele gemengde pixels, de rest van dit proefschrift richt zich op de oppervlakteschatting van agrarische velden op grond van kennis specifiek voor dit domein. **Hoofdstuk 5** is gewijd aan het onderscheiden van pure en gemengde pixels, waarbij een agrarisch veld gezien wordt als een homogeen cluster van pure pixels omgeven door een rand van gemengde pixels. Gemengde pixels kunnen opgespoord worden door het vinden van lokale discontinuïteiten in het beeld die overeenkomen met de rand om elk veld. Behalve deze *edge detectie* aanpak kunnen gemengde pixels ook herkend worden als de kleine gebiedjes die overblijven na *region growing*, een methode die probeert de grote pure pixel clusters te lokaliseren door naburige pixels met eenzelfde spectrale reflectie te groeperen. Een experiment liet zien dat region growing zowel pure als gemengde pixels in 85%–90% van de gevallen correct labelt, terwijl de prestatie van edge detectie ongeveer 10% lager lag. Een hybride aanpak die de resultaten van beide methoden integreert lijkt ideaal te zijn omdat het twee soorten informatie combineert die in zeker opzicht complementair zijn.

Hoofdstuk 6 presenteert *data-driven decompositie* (DDD) als een nieuwe methode om nauwkeurige oppervlakteschattingen van agrarische velden te maken. Het eerste principe achter DDD is dat gemengde pixels worden ontbonden om vervolgens hun oppervlak te verdelen tussen de velden die ze representeren. Pure pixels worden nog steeds geclassificeerd zoals gebruikelijk. Het andere principe is dat kennis van het toepassingsgebied gebruikt wordt om de nauwkeurigheid van de oppervlakteschattingen te verbeteren. Aspecten van deze kennis met betrekking tot agrarische velden zijn dat ruimtelijke informatie gebruikt kan worden om 1) pure en gemengde pixels te scheiden, 2) lokale endmember distributies uit het beeld zelf te halen, en 3) endmembers te selecteren die waarschijnlijk onderdeel uitmaken van een gemengde pixel. Een groot voordeel van dit laatste punt is dat het beeld in een veel groter aantal endmembers ontbonden kan worden dan mogelijk is met de klassieke decompositie aanpak. Verdere domeinkennis die in deze methode is verwerkt is dat 4) velden vaak begrensd worden door smalle structuren met een bekende samenstelling (bijvoorbeeld wegen of sloten) en 5) geïsoleerde objecten (bijvoorbeeld boerderijen, vennetjes) opgespoord kunnen worden omdat hun spectrale reflectie geen mix is van die van naburige pixels. Experimenten met artificiële Landsat-TM beelden demonstreerden dat door DDD gemiddeld slechts 5% van het oppervlak per gemengde pixel aan het verkeerde veld werd toegewezen. Deze foutjes hadden grotendeels natuurlijke oorzaken zoals variatie in de spectrale reflectie binnen een veld en *spectrale verwarring*, waar sprake van is als bepaalde klassen of combinaties van klassen op elkaar lijken. Verder bleek dat de experimenteel gevonden optimale instelling van de drempel die de werking van DDD controleert goed overeenkwam met het theoretisch voorspelde optimum.

Hoofdstuk 7 tenslotte beschrijft twee experimenten om de invloed van de twee principes achter DDD op de nauwkeurigheid van de oppervlakteschattingen te bepalen. In het eerste experiment werd DDD vergeleken met *field-based classificatie* (FBC), waarbij gemengde pixels niet ontbonden worden, *pixel-based decompositie* (PBD), waarbij geen gebruik wordt gemaakt van domeinkennis, en *pixel-based classificatie* (PBC), waarbij van geen van beide principes gebruik wordt gemaakt. Over een aantal artificiële beelden was de fout per pixel gemiddeld 4.4% voor DDD, 13.1% voor PBD, 36.8% voor FBC en 43.5%

voor PBC. Decompositie van gemengde pixels bleek voornamelijk nauwkeuriger te zijn dan classificatie omdat het verkeerd toewijzen van gedeeltes van een gemengde pixel—hetgeen inherent is aan de classificatie aanpak—grotendeels wordt vermeden, maar ook omdat fouten als gevolg van spectrale verwarring minder vaak voorkomen in een gemengd pixelmodel dan in een model voor pure pixels. Ook het toepassen van domeinkennis bleek nauwkeurigere oppervlakteschattingen op te leveren, hoofdzakelijk omdat het selectiemechanisme voor endmembers de mogelijkheden voor spectrale verwarring beperkt. Aangezien zowel spectrale verwarring als het verkeerd toewijzen van pixelfracties tijdens classificatie leidt tot het systematisch overschatten van sommige klassen en het onderschatten van andere, geeft DDD ook een aanzienlijk nauwkeuriger schatting van het oppervlak van een agrarisch veld als de resultaten voor individuele pixels worden gesommeerd. In het tweede experiment werd DDD vergeleken met FBC op grond van een echt Landsat-TM beeld om de mogelijke invloed van het gebruik van artificiële beelden te elimineren. Vergeleken met de resultaten van FBC gaf DDD een nauwkeuriger schatting van de oppervlakte voor 13 van de 17 percelen, terwijl in een geval hetzelfde resultaat werd behaald. De overige 3 gevallen konden eenvoudig worden verklaard. Net als in het eerste experiment werd gevonden dat spectrale verwarring een veel groter probleem vormt voor FBC dan voor DDD, alhoewel deze conclusie alleen op grond van een kwalitatieve vergelijking kon worden getrokken.

Gezien de resultaten van de vergelijkende experimenten kan geconcludeerd worden dat het hoofddoel van dit proefschrift—het verbeteren van de nauwkeurigheid met 25% per gemengde pixel door decompositie in plaats van classificatie toe te passen—gehaald is. Immers, de nauwkeurigheid van DDD is ruim 30% hoger dan die van FBC, evenals de nauwkeurigheid van PBD in vergelijking met die van PBC. Ook het toepassen van domeinkennis leidt tot enige verbetering, aangezien de nauwkeurigheid van DDD ten opzichte van PBD, net als die van FBC in vergelijking met PBC, zo'n 5%–10% hoger ligt. De combinatie van beide principes geeft een verbetering van ongeveer 40%, zoals de vergelijking tussen DDD en PBC aangeeft. Hoewel deze resultaten behaald zijn op artificiële data bevestigde een experiment met echte Landsat-TM data dat DDD ook in werkelijkheid aanzienlijk nauwkeuriger is dan een soortgelijke methode gebaseerd op classificatie. Naast het hoofddoel is ook het nevendoel van dit proefschrift bereikt, namelijk het verkrijgen van een beter inzicht in het verwerken van gemengde pixels in het algemeen. In de verschillende hoofdstukken zijn namelijk een groot aantal aspecten van het verwerken van gemengde pixels besproken en gerelateerd, waarbij tevens een aantal nieuwe modellen en methoden zijn geïntroduceerd. Hiervan kunnen de technieken die in het eerste deel besproken zijn waarschijnlijk voor de meeste toepassingen worden gebruikt, terwijl de methoden in het tweede deel in het algemeen gebruikt kunnen worden zodra de objecten in het beeld relatief groot zijn. Aangezien kennis die zeer specifiek is voor het agrarische domein slechts sporadisch wordt gebruikt en eenvoudig weggelaten kan worden, wordt dit proefschrift hopelijk een waardevolle bron van informatie voor iedereen die zich bezig houdt met het verwerken van gemengde pixels in het kader van oppervlakteschatting.

Curriculum vitæ

4 mei 1969
Geboren te Westervoort.

5 juni 1987
Diploma VWO (cum laude), Thomas a Kempis College, Arnhem.

september 1987–maart 1993
Studie Informatica (incl. 6 studentassistentschappen), Faculteit der Wiskunde en Natuurwetenschappen (later Faculteit der Wiskunde en Informatica), Katholieke Universiteit Nijmegen.

26 augustus 1988
Propadeutisch diploma Informatica (cum laude), Faculteit der Wiskunde en Natuurwetenschappen, Katholieke Universiteit Nijmegen.

september 1991–maart 1993
Afstudeerproject bij het Biofysisch Laboratorium, afdeling Oogheelkunde, Academisch Ziekenhuis Nijmegen St. Radboud.

26 maart 1993
Doctoraal diploma Informatica (cum laude), Faculteit der Wiskunde en Informatica, Katholieke Universiteit Nijmegen.

april 1993–december 1993
Vervulling militaire dienstplicht, Korps Commando Troepen, Roosendaal.

januari 1994–april 1998
Assistent-in-Opleiding bij het zwaartepunt Informatica voor Technische Toepassingen, Computing Science Institute, Faculteit der Wiskunde en Informatica, Katholieke Universiteit Nijmegen.

mei 1998–heden
Marie Curie Research Training Grant holder, Strategy and Systems for Space Applications unit (voorheen Centre for Earth Observation), Space Applications Institute, Joint Research Centre (European Commission), Ispra, Italy.