

UNIVERSITEIT ANTWERPEN

FYSICA

MASTERPROEF

---

**Spectrale variabiliteit in niet-lineaire ontmenging van  
hyperspectrale beelden**

---

*Auteur:*

Thorvald Dox

*Promotor:*

Paul Scheunders

*Copromotor:*

Rob Heylen



# Inhoudsopgave

<b>Abstract (english)</b>	<b>3</b>
<b>Abstract (Nederlands)</b>	<b>4</b>
<b>Structuur van deze thesis</b>	<b>5</b>
<b>Inleiding</b>	<b>6</b>
0.1 Reflectantie . . . . .	7
0.2 Atmosferische correctie . . . . .	8
0.3 Toepassingen . . . . .	8
0.4 Alina-dataset . . . . .	9
<b>1 Spectraal ontmengen</b>	<b>10</b>
1.1 Resolutie . . . . .	10
1.2 Lineair ontmengen . . . . .	11
1.2.1 Minimaliseren van reconstructie-error . . . . .	12
1.3 Niet-lineaire interacties . . . . .	14
1.4 Variabiliteit . . . . .	14
<b>2 Niet-lineariteit<sup>[5]</sup></b>	<b>15</b>
2.1 Bilineair ontmengen . . . . .	15
2.1.1 Vrijheidsgraden . . . . .	16
2.2 Multilineair ontmengen . . . . .	16
2.2.1 Berekening . . . . .	16
2.2.2 Reflectantie vs albedo . . . . .	19
2.2.3 Ondergrens van P waarde . . . . .	19
<b>3 Endmember-variabiliteit<sup>[6]</sup></b>	<b>20</b>
3.1 Multiple endmember spectral mixture analysis . . . . .	20
3.1.1 randvoorwaarden . . . . .	20
3.1.2 Werking en implementatie van MESMA . . . . .	21
3.1.3 vrije keuze van het ontmengalgoritme . . . . .	22
3.2 Suboptimaal algoritme . . . . .	23
3.2.1 Hoekminimalisatie . . . . .	23

<b>4</b>	<b>Spectrale variabiliteit bij niet-lineaire ontmenging</b>	<b>25</b>
4.1	Multilineair MESMA . . . . .	25
4.2	Semi-lineair model . . . . .	26
4.2.1	Theoretische controle met gesimuleerde data . . . . .	26
4.3	multilineair AAM . . . . .	30
4.3.1	correct geselecteerde materialen . . . . .	30
<b>5</b>	<b>Experimentele vergelijking van verschillende methodes</b>	<b>34</b>
5.1	Looptijd en reconstructie-error . . . . .	34
5.2	De vergeleken methodes . . . . .	34
5.3	Bespreking van de resultaten . . . . .	40
5.3.1	Lineair vs semilineair . . . . .	40
5.3.2	Semi-lineair vs multilineair . . . . .	40
5.3.3	P-afhankelijkheid . . . . .	40
5.3.4	Bespreking P-ondergrens . . . . .	41
5.3.5	Bepreking multilinair AAM vs semilinair model . . . . .	41
	<b>Bibliografie</b>	<b>42</b>

# Abstract

# Abstract

# Structuur van deze thesis

In de inleiding worden de fundamenteën van aardobservatie beschreven. Dit hoofdstuk is vooral gebaseerd op *Fundamentals of remote sensing*<sup>[3]</sup>.

Het eerste hoofdstuk introduceert het onderwerp van deze thesis: spectrale ontmenging. Eerst wordt het lineaire spectrale ontmengingsmodel beschreven, welke het meest gebruikte model is. Dan worden twee belangrijke problemen beschreven die niet met het lineaire model kunnen worden behandeld: niet-lineaire effecten en endmember variabiliteit.

In hoofdstuk 2 en 3 worden 2 state-of-the-art methodes uit de literatuur beschreven voor niet-lineair spectrale ontmenging en endmember variabiliteit. Het multilineair model -beschreven in hoofdstuk twee - is een methode voor niet-lineaire spectrale ontmenging, en wordt uitvoerig beschreven in: *A multilinear mixing model for nonlinear spectral unmixing*<sup>[5]</sup>. In hoofdstuk 3 wordt de methode die endmember variabiliteit in rekening brengt beschreven, deze wordt behandeld in: *Hyperspectral unmixing with endmember variability via alternating angle minimization*<sup>[6]</sup>.

Er zijn in de literatuur momenteel nauwelijks of geen methoden gekend die endmember variabiliteit in rekening brengen bij niet-lineaire ontmenging. Daarom heb ik in deze masterproef onderzoek gedaan naar de combinatie van niet-lineaire ontmenging en endmember variabiliteit. In het derde hoofdstuk beschrijf ik een aantal methodes die gelijktijdig rekening houden met beide concepten. Deze methodes zijn gebaseerd op combinaties van de 2 methodes uit hoofdstuk 2 en 3.

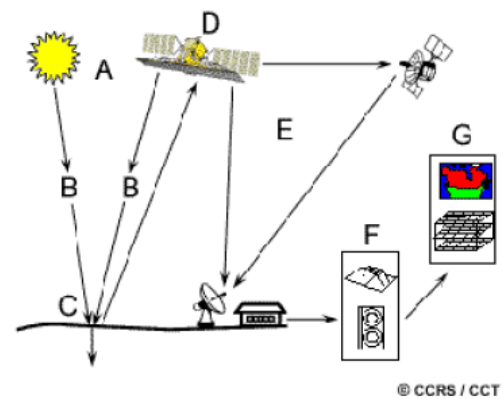
In hoofdstuk 5 worden de ontwikkelde methodes en een aantal varianten hierop gevalideerd op een experimentele dataset: de Alina dataset<sup>[4]</sup>.

# Inleiding

Aardobservatie is de wetenschap van het bepalen van informatie over het aardoppervlak vanop een afstand. Dit wordt gedaan door het meten en vastleggen van gereflecteerde of uitgezonden elektromagnetische golven en het verwerken, analyseren en toepassen van deze informatie.

(*Fundamentals of remote sensing*<sup>[3]</sup> p5)

Aardobservatie begint bij een lichtbron<sup>1</sup>. Dit is vaak de zon, maar dit proces is gelijkaardig voor andere lichtbronnen. De lichtbron zendt een lichtstraal uit, die eerst door de ruimte en vervolgens door de atmosfeer reist, totdat deze contact maakt met een materiaal op het aardoppervlak en hierop een of meermaals reflecteert. Deze reflectie verandert de eigenschappen van de lichtstraal. De gereflecteerde lichtstraal reist opnieuw door de ruimte, totdat deze in contact komt met de detector op een satelliet. Deze detector zet de lichtstraal om in elektrische signalen, die verzonden worden vanaf de satelliet naar het aardoppervlak, waar deze geanalyseerd kunnen worden. Dit volledige proces wordt afgebeeld op figuur 1.

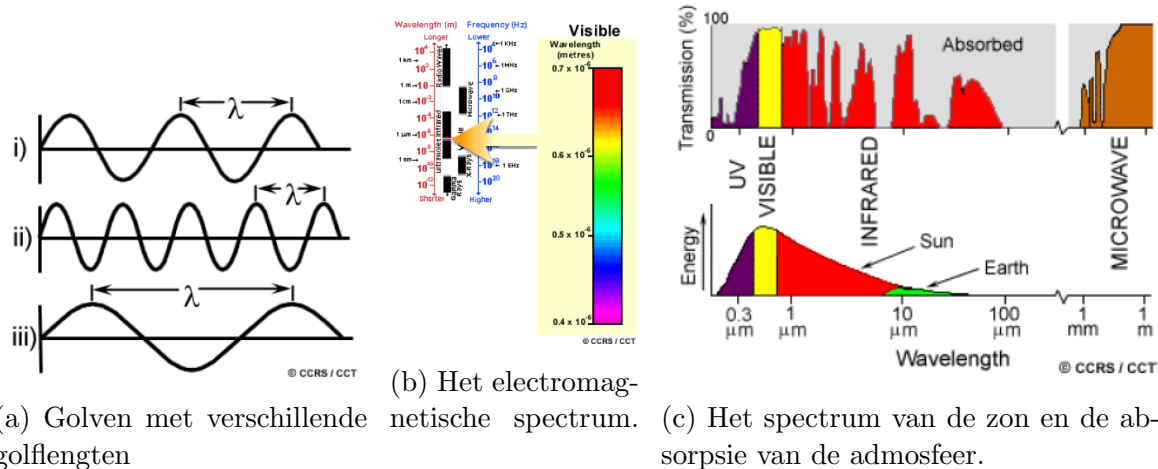


Figuur 1: Aardobservatie Deze afbeelding komt uit *Fundamentals of remote sensing*<sup>[3]</sup>

## Elektromagnetische golven

Een lichtstraal is een elektromagnetische golf, met als een van de belangrijkste eigenschappen voor aardobservatie de golflengte. Voor een vlakke golf is deze golflengte de afstand tussen twee opeenvolgende cycli, zoals te zien in figuur 2. Echter, een lichtstraal is bijna nooit een vlakke golf, maar een combinatie van verschillende vlakke golven met verschillende golflengten. De intensiteit van een golf in functie van de golflengte wordt het spectrum genoemd. Soms wordt in plaats van golflengte ook frequentie gebruikt, deze is eigenlijk de snelheid van het licht gedeeld door de golflengte.

<sup>1</sup>In het geval dat men geïnteresseerd is in het thermisch infrarood spectrum, is er geen lichtbron nodig aangezien materialen op kamertemperatuur van nature dit soort licht uitstralen, ten gevolge van black body radiation.



Figuur 2: Elektromagnetische golven. Deze afbeeldingen komen uit *Fundamentals of remote sensing*<sup>[3]</sup>

### Hyperspectrale camera's

Gewone beelden van een camera bestaan uit drie kleuren, elk passende bij een specifieke golflengte. De keuze voor deze drie kleuren, namelijk rood, groen en blauw, is een gevolg van de beperkingen van het menselijk oog. Het zijn de kleuren die een mens kan waarnemen. Om dit beeld digitaal te vertalen, moet het beeld opgedeeld worden in kleine, even grote, vierkante elementen, genaamd pixels. Elk van deze pixels bevat voor elke kleur een waarde, die de respectievelijke intensiteit van alle kleuren in dit vierkant weergeeft. Hyperspectrale camera's werken in essentie op dezelfde manier, alleen detecteert deze camera niet alleen de hoeveelheid rood, groen en blauw, maar een groot aantal kleuren, genaamd banden. Door het grotere aantal banden in vergelijking met die van een gewone camera, kan hier meer informatie uit gehaald worden. De zon zendt vooral licht uit in het infrarood, het zichtbaar licht en het ultraviolet en daarom worden er vooral banden gebruikt uit dit spectrum, zoals te zien in figuur 2c.

Het doel van spectrale analyse is de analyse van de materialen op het aardoppervlak, gegeven het spectrum van deze pixel. Verschillende methoden hiervoor worden beschreven in deze thesis.

## 0.1 Reflectantie

Wanneer een lichtstraal invalt op een materiaal reflecteert dit een gedeelte van het licht en absorbeert het de rest. Het spectrum van deze lichtstraal die gereflecteerd wordt noemt men de reflectantie. Deze is afhankelijk van het materiaal en van de golflengte van het licht.

In deze thesis worden hiervoor drie aannames gemaakt. Ten eerste: de gereflecteerde lichtstraal heeft dezelfde golflengte als het invallende licht. Ten tweede: de intensiteit is van de gereflecteerde straal recht evenredig met de intensiteit van de invallende straal. Ten derde: de reflectie van een lichtstraal van een specifieke golflengte wordt niet beïnvloed door licht van andere frequenties. Deze drie aannames zijn correct in de klassieke fysica, maar er



zitten correcties van kwantummechanische effecten op. Deze zijn echter klein genoeg zodat ze als deel van de ruis beschouwd kunnen worden.

De intensiteit van de uitgaande straal kan geschreven worden als volgt:

$$E_{out}(\lambda) = R_{\lambda}E_{in}(\lambda) \quad (1)$$

Dit is een stelsel van vergelijkingen met een vergelijking voor elke waarde van  $\lambda$ , die overeenkomt met een band in het spectrum. Zowel de intensiteiten als de reflectiecoëfficiënten kunnen samengesteld worden tot een vector, en dan geldt

$$\mathbf{x} = \mathbf{R} \odot \mathbf{y} \quad (2)$$

met  $\odot$  het Hadamard ofwel elementwijs product. De vector  $R$  is de reflectie van het materiaal. Deze neemt in beschouwing dat er meerdere interacties kunnen gebeuren in het materiaal. Een reflectie die alleen een enkelvoudige interactie meeneemt, noemt men het albedo.

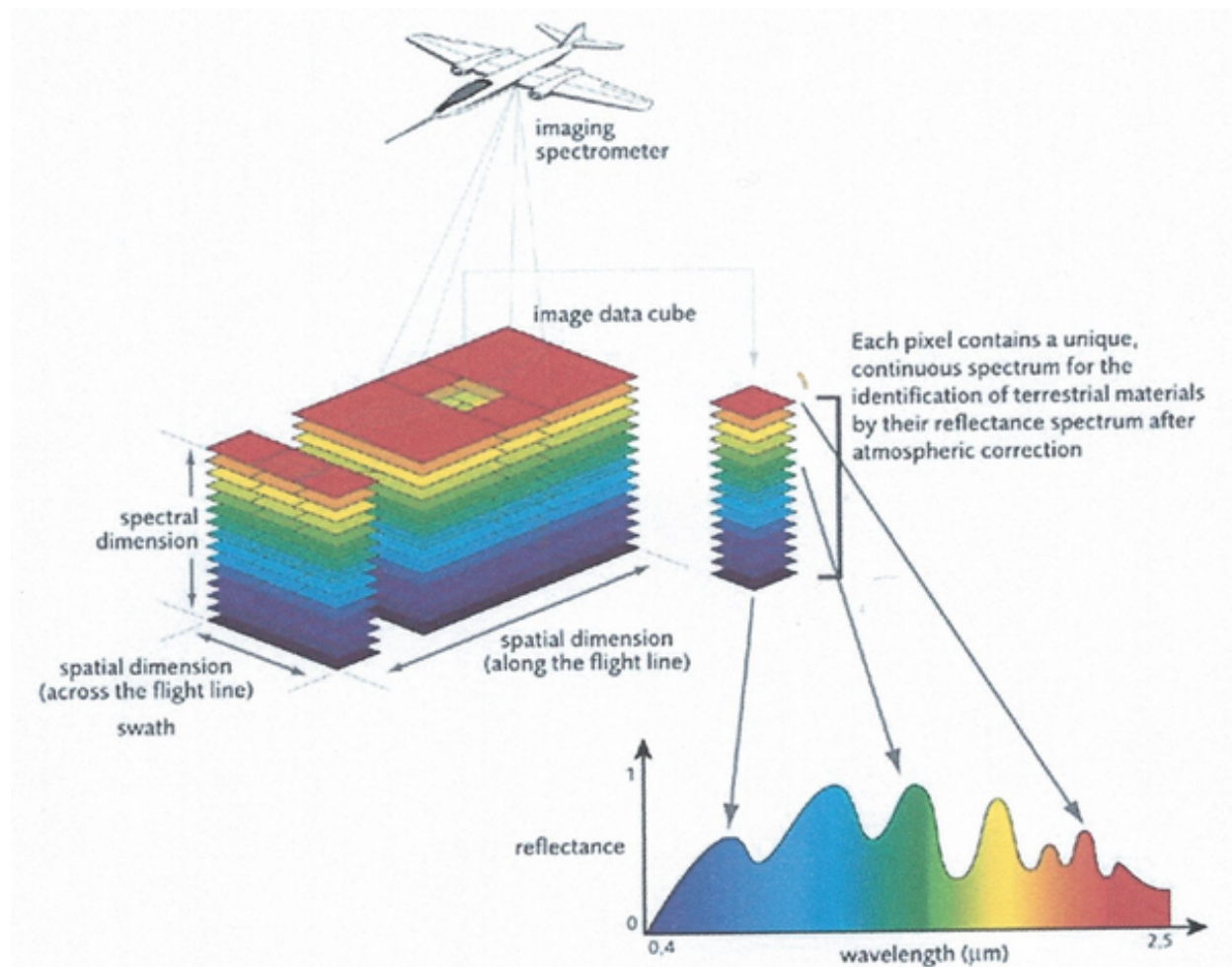
## 0.2 Atmosferische correctie

Wanneer de lichtbron uniform en genormaliseerd is, is de reflectie gelijk aan de reflectantie. Alleen is de gebruikte lichtbron - meestal is dit de zon - niet uniform. Verder beïnvloeden verschillende effecten de lichtstraal<sup>[3]</sup> zoals verschuiving, atmosferische verstrooiing, Rayleigh verstrooiing, Mie verstrooiing, nonselectieve verstrooiing, atmosferische absorptie, ozon absorptie,  $CO_2$ -absorptie en interpixel verstrooiing. De gebruikte datasets in deze thesis zijn door een algoritme gecorrigeerd voor al deze effecten, en de beelden kunnen dus beschouwd worden als verlicht door een uniforme lichtbron, waarbij de gedetecteerde lichtstraal alleen beïnvloed is door reflecties en Gaussische ruis.

## 0.3 Toepassingen

Een veelgebruikte toepassing van aardobservatie is het in kaart brengen van landbouwgewassen<sup>[3]</sup>. Men kan niet alleen het soort gewas op een akker in kaart brengen, maar ook de gezondheid en verwachte oogst van verschillende gewassen. In het verleden werd het in kaart brengen van gewassen gedaan door steekproeven met de hand vanop de grond te nemen, maar aardobservatie is nauwkeuriger, goedkoper en kan gestandaardiseerd worden. Enerzijds is deze informatie nuttig voor de landbouwers zelf, omdat deze aan de hand van de data kunnen bepalen welke gewassen het beste groeien op welke plaats en wanneer er het best gezaaid of geoogst wordt, aan de andere kant is deze informatie ook nuttig voor overheden en verzekeraars, omdat deze kunnen nagaan welke planten er geplant worden, voorspellen wat de oogst gaat zijn en de schade bepalen na een droogte of storm.

Aardobservatie wordt ook toepast in de geologie. Van de aardbodem kan niet alleen het soort gesteente aan de oppervlakte in kaart gebracht worden, maar ook de formatie van deze gesteentes en zelfs de gesteentes liggend onder de aardbodem. Een van de voor de hand liggende toepassingen hiervan is de mijnbouw; een andere, het voorspellen van aardbevingen,



Figuur 3: Opbouw van een hyperspectraal beeld. Deze afbeelding komt uit *Fundamentals of remote sensing*<sup>[3]</sup>

aardverschuivingen en vulkanisme, wat inhoudt dat aardobservatie gebruikt kan worden voor het plannen van wegen, gebouwen en andere structuren.

## 0.4 Alina-dataset

[4]

De *Alina-dataset* is een dataset die verder in deze thesis gebruikt wordt. Deze is een deel uit een hyperspectraal beeld uit Long-beach, Missouri, USA, en is genomen met de Avari-sensor. Het oorspronkelijke beeld is vrij groot, maar hier er is een deel van genomen van  $13 \times 19$  pixels. Dit stuk van het beeld bevat vier verschillende endmembers: een geasfalteerde weg, een voetpad naast de weg, een bos van eikenbomen en een grasveld. Voor elk van deze endmembers is er een bibliotheek van spectra, gemeten met een *hand-held ASD spectrometer*, met respectievelijk 10,10,10 en 50 spectra. Zowel de pixels van het beeld en de spectra uit de bibliotheken hebben 53 banden.

# Hoofdstuk 1

## Spectraal ontmengen

### 1.1 Resolutie

Een pixel in een hyperspectraal beeld bevat een of meerdere materialen. Indien elke pixel maar een enkel materiaal zou bevatten, zou elk spectrum van elke pixel vergeleken kunnen worden met een referentiespectrum. Echter, het is theoretisch onmogelijk om een camera te maken met een resolutie die groot genoeg is om dit te verwezenlijken. De hoeveelheid licht die op de camera valt is immers beperkt, en moet verdeeld worden over alle pixels en de verschillende banden. Dit betekent dat wanneer de pixels te klein worden, elke pixel te weinig licht krijgt en de signaal-ruisverhouding te laag wordt, en aangezien licht bestaat uit fotonen, is er een limiet aan de gevoeligheid van de camera.

De enige methode om de gevoeligheid te verhogen is bijgevolg een langere belichtingstijd. Hierdoor vergroot de ruis en dit is niet haalbaar voor een satelliet in een baan rond de aarde - zelfs als het praktisch haalbaar is om de resolutie kleiner te maken, ligt het probleem vooral in de fractalische eigenschappen van vele objecten.

Als bijvoorbeeld, bij het observeren van een bos een grove resolutie gebruikt wordt, vallen er verschillende soorten bomen in een enkele pixel. Als de pixels kleiner worden, kunnen bomen al van elkaar onderscheiden worden, maar de takken van die bomen nog niet. Zelfs een blad bestaat nog uit verschillende materialen, aangezien de nerven van het blad uit een ander materiaal bestaan dan de bladmoes. Ook bij mineralen heeft men eenzelfde probleem: een gesteente bevat verschillende mineralen, maar om deze van elkaar te kunnen onderscheiden moet de pixelgrootte van microscopisch niveau zijn.

Hierom is het van belang om het spectrum van een pixel, dat een combinatie is van de spectra van de verschillende materialen in deze pixel, te kunnen “ontmengen”. Hierbij is men specifiek geïnteresseerd in de abundantie van elk materiaal: een getal dat het gedeelte van de lichtstralen afkomstig van dat specifiek materiaal in een pixel weergeeft. Aangezien dit een goede maat is voor de aanwezige hoeveelheid van een bepaald materiaal in een pixel, is dit de belangrijkste parameter.

## 1.2 Lineair ontmengen

Bij lineair ontmengen wordt uitgegaan van het *linear mixing model*. Een lichtstraal valt in op een pixel op een enkel wel bepaald materiaal, interageert eenmalig met dit materiaal en valt daarna op de detector. Het spectrum van de teruggekaatste lichtstraal alleen afhangt van de reflectantie van het materiaal, de *endmember* genaamd. De totale reflectantie van een gemeten pixel is het gewogen gemiddelde van de verschillende endmembers, waarbij het gewicht van elke endmember bepaald wordt door het gedeelte van de lichtstralen dat door dit materiaal gereflecteerd wordt. Dit gedeelte is de abundantie. In symbolen geeft dit:

$$\mathbf{x} = \sum_i a_i \mathbf{e}_i \quad (1.1)$$

Waarbij  $\mathbf{x}$  het gemeten spectrum is,  $\mathbf{e}_i$  de endmembers en  $a_i$  de abundanties. Omdat het mixing model slechts een benadering is van de werkelijkheid en omdat er op een experimentele meting altijd een vorm van ruis zit, kan het exacte spectrum nooit gevonden worden. Er wordt daarom gezocht naar het reconstructiespectrum dat het dichtst bij het werkelijke spectrum ligt. Hierbij wordt verondersteld dat de ruis normaal verdeeld is. Dit is een eenvoudig gevolg van het centrale-limiettheorema, dat zegt dat de som van een groot aantal kansvariabelen normaal verdeeld is.

Noem  $\mathbf{x}$  het gemeten spectrum, en  $\mathbf{y}$  het reconstructiespectrum, met  $\boldsymbol{\eta}$  de ruis, dan is

$$\mathbf{x} = \mathbf{y} + \boldsymbol{\eta} \quad (1.2)$$

De kans dat  $\mathbf{x}$  gemeten wordt, gegeven dat  $\mathbf{y}$  het reconstructiespectrum is, is  $f(\boldsymbol{\eta})$ , waarbij  $f$  de normale verdeling is. Aangezien de normale verdeling afhankelijk is van de kwadratische norm, zijn we alleen geïnteresseerd in de kwadratische norm van de ruis. De normale verdeling is ook groter wanneer deze kwadratische norm kleiner is, dus moet de norm van de ruis geminimaliseerd worden om de kans te maximaliseren.

Deze norm kan berekend worden gebruik makend van vergelijking 1.2.

$$|\boldsymbol{\eta}|^2 = |\mathbf{x} - \mathbf{y}|^2 \quad (1.3)$$

De uitdrukking  $|\mathbf{x} - \mathbf{y}|^2$  noemt men het reconstructie-error. Het is4 deze uitdrukking die geminimaliseerd moet worden bij ontmenging.

### 1.2.1 Minimaliseren van reconstructie-error

Voor het lineair model wordt er vanuit gegaan dat de endmembers gekend zijn, maar de abundanties en de ruis niet. Als de ruis wordt toegevoegd aan vergelijking 1.1 dan geldt:

$$\mathbf{x} = \sum_i a_i \mathbf{e}_i + \eta \quad (1.4)$$

Het minimaliseren van de reconstructie-error, gebruik makend van vergelijking 1.3:

$$\operatorname{argmin}_{a_1 \dots a_p} \left| \mathbf{x} - \sum_{i=1}^p a_i \mathbf{e}_i \right|^2 \quad (1.5)$$

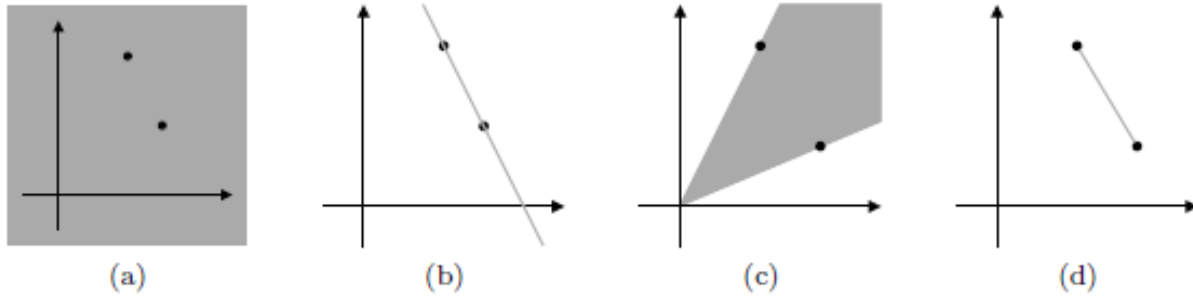
Indien de abundanties  $a_i$  vrije reële waarden zouden zijn, zou dit minimum eenvoudig berekend kunnen worden, door een projectie te nemen opgespannen door het vlak van de endmembers. Deze projectie kan genomen worden als volgt. Neem  $E = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n]$  een matrix met als kolommen de verschillende endmembers. De projectie op de deelruimte opgespannen door de elementen van  $E$  kan gevonden worden aan de hand van de *More-Penrose-inverse* matrix.

$$\mathbf{a} = (E^T E)^{-1} E^T \mathbf{x} \quad (1.6)$$

Doordat de abundanties gedeeltes van lichtstralen zijn die op een materiaal botsen, zijn er twee extra voorwaarden. Enerzijds moeten de abundanties positief zijn, aangezien men geen negatieve hoeveelheid lichtstralen kan hebben - dit noemt men de niet-negativiteitsvoorwaarde; anderzijds kan elke lichtstraal maar op een enkel materiaal botsen, dus is de som van alle gedeeltes één - dit noemt men de eenheidssom-voorwaarde.

#### **Niet-negativiteit**

De niet-negativiteitsvoorwaarde geeft weer dat de abundanties allemaal positief moeten zijn, dus groter of gelijk aan nul. Dit betekent dat men niet moet projecteren op de deelruimte, maar op een polyhedrale kegel hierin. Een tweedimensionaal voorbeeld, waarvan het systeem zonder voorwaarde de volledige ruimte is, is geïllustreerd in figuur 1.1. Hiervoor kan een aangepast algoritme gebruikt worden, namelijk het “nonnegative least-squares curve fitting” of `lsqnonneg` algoritme. Dit algoritme is veel berekeningsintensiever dan het *penrose-inverse* algoritme.



Figuur 1.1: Voorbeeld van de niet-negativiteitsvoorwaarde aan de hand van een model met twee endmembers. (a) is het model zonder voorwaarden. (b) bevat de eenheidssomvoorwaarde, (c) de niet-negativiteitsvoorwaarde en (d) beide voorwaarden. Deze figuur komt uit<sup>[?]</sup>.

### Eenheidssom

De eenheidssom-voorwaarde geeft weer dat de abundanties moeten sommeren tot één. Dit zorgt ervoor dat de mogelijke abundanties geen deelruimte meer opspannen, maar een hypervlak, aangezien het nulpunt geen element meer is, zoals te zien in figuur 1.1. Dit kan worden opgelost door een van de endmembers als schaduw te beschouwen. Dit verschuift het hypervlak over de vector  $e_1$  zodanig dat het nulpunt deel wordt van de hyperruimte en dit terug een deelruimte is.

Deze transformatie houdt in dat de matrix  $E$  getranslateerd wordt naar  $[e_2 - e_1; e_3 - e_1; \dots; e_n - e_1]$  en het gemeten spectrum naar  $x - e_1$ . Na deze transformatie kan het *Penrose-inverse* algoritme worden toegepast, zodat men de abundanties  $a_2, a_3, \dots, a_n$  krijgt. De laatste abundantie  $a_1$  kan gevonden worden door de eenheidssomvoorwaarde te eisen, zodat  $a_1 = 1 - \sum_{i=2}^n a_i$ .

Het doorvoeren van beide bovenstaande voorwaarden zorgt ervoor dat de oplossingsruimte een simplex opspant, en de oplossing hiervan kan gevonden worden door de beide bovengenoemde correcties door te voeren. Het algoritme dat deze correcties doorvoert noemt men het *Fully constrained least-squares unmixing model*. Het algoritme dat enkel de eenheidssomvoorwaarde meeneemt, maar niet de niet-negativiteitsvoorwaarde, noemt men het *sum-to-one constrained least-squares unmixing model*.

Volgende secties beschrijven twee belangrijke problemen voor spectraal ont mengen. Hiervoor wordt een oplossing beschreven in de rest van deze thesis.

## 1.3 Niet-lineaire interacties

In vorige sectie is men ervan uitgegaan dat elke lichtstraal maar een keer interageert met een enkel materiaal. Dit is een goede benadering als men vlakke gebieden met duidelijk verdeelde endmembers beschouwd. Dit geldt echter niet meer voor geometrische structuren zoals gebouwen of vegetatie, waarbij enkelvoudige interacties zeldzaam zijn en de meerderheid van de lichtstralen meermaals interageert. Hetzelfde geldt voor gebieden met mineralen, waarbij lichtstralen meerdere keren interageren met een enkele korrel en effecten zoals transmissie belangrijk worden.

In deze thesis wordt geen rekening gehouden met transmissie-effecten en interpixel-interacties. Dit laatste is het fenomeen waarbij een lichtstraal interageert met materialen in twee of meer verschillende pixels. Wel wordt er rekening gehouden met niet-lineaire effecten in een enkele pixel, wat inhoudt dat een lichtstraal meerdere keren kan interageren met dezelfde of verschillende materialen in een enkele pixel.

## 1.4 Variabiliteit

In dit hoofdstuk werd ervan uitgegaan dat elke endmember een enkel vast spectrum heeft. In werkelijkheid is dit niet het geval. Enerzijds worden materialen met verschillende spectra beschouwd als een enkel materiaal. Bijvoorbeeld de bovenkant en de onderkant van een blad, die een verschillend spectrum hebben, zien we als een materiaal. Ook door de mens gemaakte materialen, zoals beton en asfalt, kunnen verschillende spectra hebben. Meestal is het niet interessant om een onderscheid te maken tussen deze verschillende materialen. Aan de andere kant kan het spectrum van een enkel materiaal ook verschillen dankzij verschillen in belichting en atmosfeer. Deze beide effecten zorgen ervoor dat een enkele endmember meerdere spectra heeft. Dit concept noemt men variabiliteit. Hierdoor word elke endmember beschreven door een bibliotheek van spectra, in plaats van een enkel spectrum. Hoe kan worden ontmengd, rekening houdend met variabiliteit staat beschreven in sectie 3.1.

# Hoofdstuk 2

## Niet-lineariteit<sup>[5]</sup>

### 2.1 Bilineair ontmengen

Het bilineair model beschrijft dat elke lichtstraal één of twee keer kan reflecteren in een enkele pixel. Van dit model bestaan meerdere varianten. Het Fan model<sup>[5]</sup> voert de bilinaire term in als volgt.

$$\mathbf{x} = \sum_i a_i \mathbf{e}_i + \sum_{i=1}^{p-1} \sum_{j=i}^p a_i a_j \mathbf{e}_i \odot \mathbf{e}_j + \boldsymbol{\eta} \quad (2.1)$$

Dit geeft in de praktijk geen goede resultaten, omdat het bilineair model is ontwikkeld voor objecten met een ingewikkelde geometrie, maar deze methode bevat geen parameters of informatie over deze geometrie. Dit probleem kan worden opgelost door gebruik te maken van het veralgemeend bilineair model<sup>[5]</sup>, dat beschrijft dat de endmembers worden gemengd als volgt:

$$\mathbf{x} = \sum_i a_i \mathbf{e}_i + \sum_{i=1}^{p-1} \sum_{j=i}^p \gamma_{ij} a_i a_j \mathbf{e}_i \odot \mathbf{e}_j + \boldsymbol{\eta} \quad (2.2)$$

Een alternatief is het model van Nascimento en Somers, dat de abundaties mengt als volgt:

$$\mathbf{x} = \sum_i a_i \mathbf{e}_i + \sum_{i=1}^{p-1} \sum_{j=i}^p b_{ij} \mathbf{e}_i \odot \mathbf{e}_j + \boldsymbol{\eta} \quad (2.3)$$

Dit is equivalent aan het vorige model, behalve dat de eenheidssomvoorwaarde  $\sum_i a_i + \sum_{i,j} b_{ij} = 1$  wordt in plaats daarvan  $\sum_i a_i = 1$

Deze beide model hebben een ander probleem, namelijk dat voor een model met  $p$  materialen, dit model  $p^2$  extra vrijheidsgraden bevat. Daardoor is er geen garantie dat het model het juiste resultaat weergeeft, zoals beschreven in sectie 2.1.1.



### 2.1.1 Vrijheidsgraden

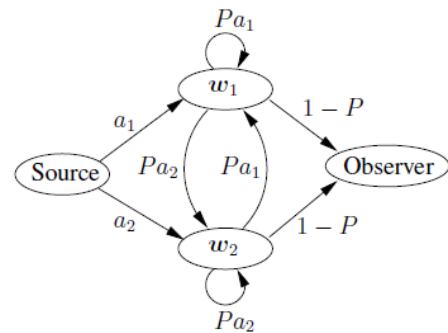
In een optimalisatieprobleem noemt men het aantal reële continue variabelen waarvan een systeem afhankelijk is de vrijheidsgraden. Wanneer het aantal te bepalen vrijheidsgraden groter is dan het aantal gegeven parameters, dan is het probleem ondergedefinieerd. Dit houdt in dat er meerdere oplossingen voor de vrije parameters mogelijk zijn waarvoor aan alle voorwaarden voldaan is. Als dit gebeurt, is een gevonden oplossing niet met zekerheid de juiste oplossing, en geeft het model verkeerde oplossingen terug, die lijken te voldoen aan het systeem.

Van wegen deze problemen met het bilineaire model, wordt er overgegaan naar het multilineaire model dat sterker fysisch onderbouwd is, en dat meer dan twee reflecties meeneemt. Dit model wordt beschreven in de volgende sectie.

## 2.2 Multilineair ontmengen

Het multilineaire model gaat ervan uit dat een lichtstraal meerdere keren kan reflecteren in een enkele pixel. Dit model is gebaseerd op *A multilinear mixing model for nonlinear spectral unmixing*<sup>[5]</sup>.

De inkomende lichtstraal valt in op een materiaal binnen een pixel. Wanneer deze invalt op het materiaal heeft deze een kans  $P$  om opnieuw in te vallen op hetzelfde of een ander materiaal in dezelfde pixel, en een kans  $(1 - P)$  om te reflecteren naar de detector. De kans voor de gereflecteerde lichtstraal om in te vallen op een specifiek materiaal is gelijk aan diezelfde kans voor een lichtstraal afkomstig van de bron, namelijk de abundantie van de materiaal. Het spectrum van de lichtstraal verandert bij elke interactie zoals beschreven in sectie 0.1. Dat wil zeggen dat de uitgaande lichtstraal het elementwijs product is van de invallende lichtstraal en het albedo van het materiaal. Dit hele proces kan worden voorgesteld als een Markov-ketting (zie figuur 2.1).



Figuur 2.1: Markov-ketting van het multilineaire model

### 2.2.1 Berekening

In deze sectie noemen we het "pad" van een lichtstraal de volgorde waarop de verschillende materialen reflecteren. Dit kan genoteerd worden als volgt:

$$\mathcal{P}(i_1, i_2, i_3, \dots, i_R) \quad (2.4)$$

Deze uitdrukking geeft de kans weer dat een lichtstraal reflecteert op materiaal  $i_1$ , daarna op  $i_2$ , op  $i_3$  en zo voort, totdat deze uiteindelijk interageren met materiaal  $i_R$ .  $R$  is hier het totaal aantal materialen waarop gereflecteerd wordt.

De kans op dit pad kan worden uitgerekend aan de hand van het hiervoor genoemde Markov-proces<sup>[5]</sup>.

$$\mathcal{P}(i_1, i_2, i_3, \dots, i_R) = \prod_{k=1}^{R-1} P_{i_k} (1 - P_{i_R}) \prod_{k=1}^R a_{i_k} \quad (2.5)$$

Waarbij  $P_{i_k}$  de reflectiekans van materiaal  $i_k$  is. Het spectrum van de uitgaande straal ten gevolge van een specifiek pad is  $\bigodot_{k=1}^R \mathbf{w}_{i_k}$ , waarbij  $\mathbf{w}_{i_k}$  het albedo is van materiaal  $i_k$ , en  $\bigodot$  het elementwijs product. Hiermee kan de bijdrage van een specifiek pad -de kans vermenigvuldigt met het spectrum- berekend worden als volgt:

$$\mathbf{x}_{(i_1, i_2, i_3, \dots, i_R)} = \prod_{k=1}^{R-1} P_{i_k} (1 - P_{i_R}) \bigodot_{k=1}^R (a_{i_k} \mathbf{w}_{i_k}) \quad (2.6)$$

Gesommeerd over alle mogelijke paden geeft dit het totale reconstructiespectrum. Deze som over alle paden kan berekend worden door de som te nemen over alle mogelijke waarden van  $R$  voor een enkel pad, en dan als  $R$  -het aantal interacties- gekend is, kan er voor elke interactie gesommeerd worden over alle materialen. Wanneer alle bijdragen van elk pad gesommeerd zijn over alle paden, geeft dit het totale spectrum dat de detector meet, en dit is gelijk aan:

$$\mathbf{x} = \sum_{R=1}^{\infty} \left( \sum_{i_1}^p \dots \sum_{i_R}^p \right) \prod_{k=1}^{R-1} P_{i_k} (1 - P_{i_R}) \bigodot_{k=1}^R (a_{i_k} \mathbf{w}_{i_k}) \quad (2.7)$$

Het spectrum afkomstig van een specifiek pad wordt:

$$\mathbf{x}_{(i_1, i_2, i_3, \dots, i_R)} = \prod_{k=1}^{R-1} P_{i_k} (1 - P_{i_R}) \bigodot_{k=1}^R (a_{i_k} \mathbf{w}_{i_k}) \quad (2.8)$$

$$= \bigodot_{k=1}^{R-1} (P_{i_k} a_{i_k} \mathbf{w}_{i_k}) \odot (1 - P_{i_R}) a_{i_R} \mathbf{w}_{i_R} \quad (2.9)$$

$$\mathbf{x} = \sum_{R=1}^{\infty} \left( \sum_{i_1}^p \dots \sum_{i_R}^p \right) \bigodot_{k=1}^{R-1} (P_{i_k} a_{i_k} \mathbf{w}_{i_k}) \odot (1 - P_{i_R}) a_{i_R} \mathbf{w}_{i_R} \quad (2.10)$$

$$\begin{aligned} &= \sum_{i=1}^p (1 - P_i) a_i \mathbf{w}_i + \sum_{i=1}^p \sum_{j=1}^p (1 - P_i) a_i \mathbf{w}_i \odot P_j a_j \mathbf{w}_j \\ &+ \sum_{i=1}^p \sum_{j=1}^p \sum_{k=1}^p (1 - P_i) a_i \mathbf{w}_i \odot P_j a_j \mathbf{w}_j \odot P_k a_k \mathbf{w}_k \\ &+ \sum_{i=1}^p \sum_{j=1}^p \sum_{k=1}^p \sum_{l=1}^p (1 - P_i) a_i \mathbf{w}_i \odot P_j a_j \mathbf{w}_j \odot P_k a_k \mathbf{w}_k \odot P_l a_l \mathbf{w}_l \\ &+ \dots \end{aligned} \quad (2.11)$$

$$= \mathbf{z} + \mathbf{y} \odot \mathbf{z} + \mathbf{y} \odot \mathbf{y} \odot \mathbf{z} + \dots \quad (2.12)$$

$$= \mathbf{z} + \mathbf{y} \odot \mathbf{x} \quad (2.13)$$

$$= \frac{\mathbf{z}}{1 - \mathbf{y}} \quad (2.14)$$

waar

$$\mathbf{y} = \sum_{i=1}^p P_i a_i \mathbf{w}_i \quad (2.15)$$

$$\mathbf{z} = \sum_{i=1}^p (1 - P_i) a_i \mathbf{w}_i \quad (2.16)$$

zodat

$$\mathbf{x} = \frac{\sum_{i=1}^p (1 - P_i) a_i \mathbf{w}_i}{1 - \sum_{i=1}^p P_i a_i \mathbf{w}_i} \quad (2.17)$$

Dit is een uitdrukking voor het multilineair mengen van endmembers waarbij de reflecties endmember-afhankelijk is. Indien men geïntereerd is in een onafhankelijke reflecties, kan men de formule hiervoor bekomen door te stellen dat  $P_i = P \forall i$  zodat het reconstructiespectrum gelijk wordt aan:

$$\mathbf{x} = \frac{(1 - P) \sum_{i=1}^p a_i \mathbf{w}_i}{1 - P \sum_{i=1}^p a_i \mathbf{w}_i} \quad (2.18)$$

wat exact overeenkomt met de formule in de literatuur<sup>[5]</sup>.

Om aan de hand van dit model te ontmengen, moet nog steeds gebruik gemaakt worden van het minimaliseren van de reconstructie-error.

$$\operatorname{argmin}_{a_1 \dots a_p; P_1 \dots P_p} \left| \mathbf{x} - \frac{\sum_{i=1}^p (1 - P_i) a_i \mathbf{w}_i}{1 - \sum_{i=1}^p P_i a_i \mathbf{w}_i} \right|^2 \quad (2.19)$$

Aangezien deze vergelijking niet lineair is en bijgevolg ook geen hypervlak opspant, kan deze uitdrukking niet geminimaliseerd worden door middel van projecties, en zijn er hier geavanceerde minimalisatietechnieken nodig. In deze thesis hiervoor de functie `fmincon` uit *matlab*<sup>[7]</sup> gebruikt. Deze neemt de uitdrukking  $\left| \mathbf{x} - \frac{\sum_{i=1}^p (1 - P_i) a_i \mathbf{w}_i}{1 - \sum_{i=1}^p P_i a_i \mathbf{w}_i} \right|^2$  in en geeft hier een minimum voor terug. Dit heeft als nadeel dat het draaien van deze functie op de experimentele data later beschreven in deze thesis, een looptijd heeft van ongeveer 0.1 seconden per pixel, in tegenstelling tot het berekenen van pseudo-inversen, wat ongeveer 50 microseconden in beslag neemt.

## 2.2.2 Reflectantie vs albedo

Het albedo  $w_i$  uit vorige sectie kan niet exact gemeten worden. In theorie kan dit afgeleid worden uit de reflectie van een meting met een enkele endmember. Dit kan bepaald worden door vergelijking 2.17 te inverteren, zodat:

$$w_i = \frac{e_i}{e_i P_i + 1 - P_i} \quad (2.20)$$

waarbij de deling de elementwijze deling is.

Hierbij wordt verondersteld dat  $P_i$  gekend is, wat in werkelijkheid niet het geval is. Daarom wordt verondersteld dat  $P_i \approx 0$  wat neerkomt op een benadering die stelt dat  $w_i \approx e_i$ .

## 2.2.3 Ondergrens van P waarde

$P$  is per definitie een getal groter dan nul, omdat kansen tussen nul en één liggen. Echter, het is mogelijk om in vergelijking 2.17 een negatieve waarde in te voeren. Dit komt neer op een verandering van de randvoorwaarden in het `fmincon` algoritme. In plaats van dat de grenzen  $[0, 1]$  zijn, worden de grenzen nu  $[-1, 1]$ . Een negatieve reflectiekans is niet fysisch, maar er bestaan twee mogelijke verklaringen voor het toelaten van negatieve  $P$  waarden, waarbij  $P$  niet meer als kans wordt beschouwd.

De eerste mogelijke verklaring - zoals beschreven in de vorige sectie - is dat het albedo slechts een benadering is. Het vervangen van het albedo door de reflectie is equivalent aan het veronderstellen dat de reflectiekans van het referentiemateriaal gelijk is aan nul. Dit zorgt dat de waarde van  $P$  die gemeten wordt lager is dan in werkelijkheid. Als de reflectiekans van de pixel kleiner is dan de reflectiekans van het referentiemateriaal, zorgt dit voor een negatieve gemeten reflectiekans, hoewel de werkelijke reflectiekans nog steeds positief is. Merk op dat de reflectiekans  $P$  niet kan vervangen worden door  $P - P_R$  - met  $P_R$  de reflectiekans van het referentiemateriaal - omdat het systeem niet lineair afhangt van  $P$ .

De tweede mogelijke verklaring is het niet meenemen van interpixelinteracties. Deze interacties geven extra hogere-ordeterminen mee aan het systeem. De enige hogere-ordeterminen die het multilineair model meeneemt hebben machten van  $P$  als coëfficiënten, dus wordt de waarde van  $P$  aangepast naar niet werkelijke waarden.

# Hoofdstuk 3

## Endmember-variabiliteit<sup>[6]</sup>

### 3.1 Multiple endmember spectral mixture analysis

Om te ont mengen rekening houdend met variabiliteit wordt elke endmember, in plaats van als een enkel spectrum, gedefinieerd als een bibliotheek van verschillende spectra. Het extraheren van deze spectra uit de metingen ligt buiten het bestek van deze thesis: er wordt gewoon vanuit gegaan dat de endmembers gekend zijn. De experimentele data beschouwd in deze thesis bevatten spectra die vanop de grond gemeten zijn aan de hand van een spectrograaf, dus exact gekend zijn.

Het *multiple endmember spectral mixture analysis* ofwel *MESMA*-algoritme gaat ervan uit dat elk materiaal in elke pixel naar een enkel spectrum reflecteert, maar dit spectrum kan veranderen tussen verschillende pixels. Dit algoritme ontmengt de gegeven spectra over alle modellen, waarbij een model een verzameling is van spectra, gehaald uit de bibliotheken van hun specifieke materialen. Elk model bevat een verzameling spectra, en aan de hand van *sum-to-one constrained least square unmixing* kan een pixel worden ontmengd aan de hand van deze spectra. Uiteindelijk wordt het model met de laagste reconstructie-error geselecteerd dat voldoet aan de niet-negativiteitsvoorwaarde en de eenheidssom-voorwaarde. De abundanties van dit model zijn de abundanties van de materialen in de pixel.

Merk op dat er altijd minstens een model is dat voldoet aan beide voorwaarden. De eenheidssomvoorwaarde is voldaan voor alle modellen dankzij het gebruikte ontmengalgoritme. De modellen met maar een endmember hebben bijgevolg een abundantie van 1, en voldoen per definitie ook aan de niet-negativiteitsvoorwaarde.

#### 3.1.1 randvoorwaarden

Wanneer een pixel wordt ontmengd waarbij een endmember volgens *sum-to-one constrained least-squares unmixing model* een negatieve abundantie zou hebben, is deze abundantie in *Fully constrained least-squares unmixing model* gelijk aan nul. Indien deze pixel zou worden ontmengd aan de hand van de endmembers met niet negatieve abundanties, zijn de abundanties van de voordien negatieve endmembers bij definitie nul, en worden de overige abundanties bepaald door het minimaliseren van de reconstructie-error. Dit geeft de zelfde resultaten als bij het *Fully constrained least-squares unmixing model*. Dit betekent dat het computationeel

zware *Fully constrained least-squares unmixing model* kan worden vervangen door het veel snellere *sum-to-one constrained least-squares unmixing model*, gegeven dat men ontmenkt aan de hand van alle deelverzamelingen.

### 3.1.2 Werking en implementatie van MESMA

#### ***deelverzameling van alle materialen***

Als eerste moet er gelopen worden over alle mogelijke deelverzamelingen van materialen. Voor elke deelverzameling geldt dat een enkel materiaal daar ofwel een element van is, ofwel niet. Hierdoor zijn er  $2^p$  mogelijke deelverzamelingen. Er kan geïtereerd worden over alle deelverzamelingen door te itereren over de natuurlijke getallen van nul tot  $2^p - 1$ . Elke waarde van deze iterator kan geschreven worden als een binair getal, waarbij elk cijfer bepaalt of het materiaal vervat zit in de deelverzameling of niet. De waarde nul is de lege deelverzameling, deze wordt overgeslagen en de iterator begint bij een.

#### ***Itereren over alle bibliotheken***

Voor elke van deze deelverzameling van materialen, moet geïtereerd worden over alle mogelijk spectra van elk materiaal. Er zijn verschillende methoden om te itereren over verschillende verzamelingen, zoals bijvoorbeeld de *Banker series*, maar in deze thesis wordt gebruik gemaakt van overvloedige tellers<sup>[6]</sup>. Dit houdt in dat men een vector gebruikt als index, waarbij elk element verwijst naar een enkel materiaal, en de waarde van dit element verwijst naar de index van het spectrum dat gekozen werd voor dit materiaal. Bij elke lus wordt de vector gewijzigd zoals weergegeven in 3.1.

---

#### **Algoritme 3.1** Overvloedige tellers

---

```

1: de vector begint met overal 1
2: while True do
3:   verhoog het eerste element met 1
4:   for voor elk element do
5:     if Indien het element hoger is dan zijn maximum (het aantal spectra voor het
       respectievelijke materiaal) then
6:       zet het element op 1
7:       verhoog het volgende element met 1
8:       indien dit element het laatste element is, beëindig het algoritme.
9:     end if
10:   end for
11: end while

```

---

## Ontmengen van de selectie

In de vorige secties zijn er materialen en hun spectra geselecteerd. Hierna kan de pixel worden ontmengd zoals in vorig hoofdstuk, gebruikt makend van het *sum-to-one constrained least-squares unmixing model*. Dit model houdt geen rekening met de niet-negativiteitsvoorwaarde, en kan bijgevolg negatieve abundanties hebben. Indien negatieve abundanties voorkomen wordt de selectie overgeslagen, en wordt er overgegaan naar de volgende selectie van spectra en materialen. Het is onmogelijk dat op die manier alle selecties worden verwijderd, aangezien bij deelverzamelingen met een enkel materiaal dit materiaal dankzij de eenheidssomvoorwaarde abundantie 1 heeft. Het hele proces wordt beschreven in algoritme 3.2.

---

### Algoritme 3.2 MESMA

---

```

1:  $\eta = +\infty$ 
2: for Voor alle deelverzamelingen do
3:   for Voor alle combinaties van spectra do
4:     Ontmeng gebruik makend van het sum-to-one constrained least-squares unmixing
       model
5:     if  $\exists \text{Abundantie} < 0$  then
6:       continue
7:     end if
8:     if reconstructie-error  $< \eta$  then
9:        $\eta = \text{reconstructie-error}$ 
10:       $a_i = \text{abundanties}$ 
11:    end if
12:  end for
13: end for
14:  $a_i$  zijn nu de correcte abundanties

```

---

### 3.1.3 vrije keuze van het ontmengalgoritme

Hierboven is gebruikt gemaakt van het lineaire ontmengalgoritme, maar dit hoeft niet het geval te zijn. In algoritme 3.2 kan de regel *Ontmeng gebruik makend van het sum-to-one constrained least-squares unmixing model* vervangen worden door elk algoritme naar keuze. Indien men geïnteresseerd is in niet lineaire effecten, kan hier gebruik gemaakt worden van het multilineair model. De *Alina-dataset* gebruikt in deze thesis, bevat voor de deelverzameling van alle materialen  $50 \times 10 \times 10 \times 10 = 50000$  mogelijkheden, en het ontmengen van een pixel aan de hand van vier spectra duurt ongeveer 0.1 seconden. Hierdoor neemt dit algoritme meer dan 2 uur in beslag voor elke pixel. Het beeld bevat ongeveer 400 pixels, dus de totale looptijd van dit algoritme is ongeveer 25 dagen. Zelfs als *Sum-to-one constrained least-square unmixing* gebruikt wordt, levert dit een langdurig proces.

## 3.2 Suboptimaal algoritme

Bij *MESMA* moet een pixel ontmengd worden in functie van elke deelverzameling van endmembers, voor elk materiaal. Er bestaat een efficiënter proces dat werkt als volgt: een startpositie wordt gekozen van willekeurige bibliotheekelementen en daarna wordt herhaaldelijk voor een materiaal het geselecteerde spectrum vervangen door een andere spectrum van ditzelfde materiaal, met de minimale reconstructie-error. Hierbij worden de spectra van de andere materialen constant gehouden. Dit wordt gedaan voor elk materiaal, en verscheidene keren herhaald. Doordat de reconstructie-error kleiner wordt na elke stap, bestaat er een suboptimaal minimum. Wanneer dit bereikt wordt, kan voor geen enkel materiaal de reconstructie-error verlaagd worden door een ander spectrum te nemen. Dit is niet per definitie gelijk aan de oplossing van *mesma*, omdat dit een lokaal minimum vindt in plaats van een globaal minimum. Men is in elke stap niet geïnteresseerd in de werkelijke reconstructie-error, alleen in het element dat de laagst mogelijke reconstructie-error heeft.

### 3.2.1 Hoekminimalisatie

Het lineair ontmengen gebeurt door het gemeten spectrum te projecteren op het hypervlak opgespannen door de endmembers. Noemen we dit hypervlak  $S$ , dan kan de reconstructie-error herschreven worden als:

$$|\mathbf{x} - \text{proj}_S(\mathbf{x})| \quad (3.1)$$

Noemen we  $F$  het hypervlak opgespannen door endmembers  $\{e_1, \dots, e_{p-1}\}$ , zijnde het hypervlak opgespannen door alle endmembers behalve  $e_p$  dan geldt:

$$|\mathbf{x} - \text{proj}_S(\mathbf{x})| = |\mathbf{x} - \text{proj}_F(\mathbf{x})| \sin \theta \quad (3.2)$$

met

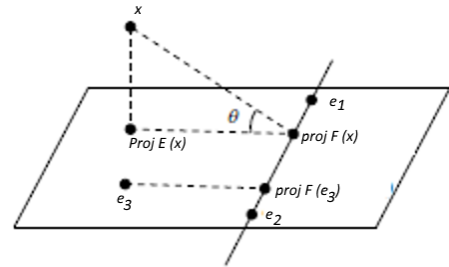
$$\theta = \min(\alpha, \pi - \alpha)$$

$$\alpha = \arcsin \left( \frac{|\mathbf{x} - \text{proj}_S(\mathbf{x})|}{|\mathbf{x} - \text{proj}_F(\mathbf{x})|} \right) \quad (3.4)$$

In deze laatste uitdrukking kan de rol van  $e_p$  en  $\mathbf{x}$  worden verwisseld, dus als er gesteld wordt dat  $G$  het hypervlak opgespannen door  $\{e_1, \dots, e_{p-1}, \mathbf{x}\}$  is, geldt dat:

$$\alpha = \arcsin \left( \frac{|\mathbf{e}_p - \text{proj}_G(\mathbf{e}_p)|}{|\mathbf{e}_p - \text{proj}_F(\mathbf{e}_p)|} \right) \quad (3.5)$$

Aangezien in functie van  $\mathbf{e}_p$ ,  $|\mathbf{x} - \text{proj}_F(\mathbf{x})|$  constant is, zal de minimalisatie van  $\sin \theta$  de reconstructie-error minimaliseren. Deze methode noemt men hoekminimalisatie of *alternating angle minimisation*



Figuur 3.1: Voorbeeld van projecties van verschillende vlakken in drie dimensies, waarbij  $e_3$  de endmembers is waarvoor het spectrum gekozen moet worden. Deze figuur komt uit *Hyperspectral unmixing with endmember variability via alternating angle minimization*<sup>[6]</sup> (3.3)



Het voordeel van deze methode is dat de reconstructie-error niet uitgerekend moet worden voor de verschillende combinaties, maar alleen voor de hoek tussen twee vectoren, waarvan een enkele vector op voorhand wordt berekend. Dit is computationeel veel efficiënter dan het voorgaande systeem.

Het volledige proces staat beschreven in algoritme 3.3.

---

**Algoritme 3.3** *Alternating Angle Minimalisation*


---

```

1: kies een aantal iteraties  $K$ 
2: for Voor elke deelverzameling van materialen do
3:   Stel  $q$  het aantal materialen
4:   Stel  $e_j^i$  het  $j$ -de spectrum uit de bibliotheek van het  $i$ -de materiaal.
5:   Selecteer een willekeurige index set  $I_j$ 
6:   for doe  $K$  keer do
7:     for voor  $i$  elk materiaal do
8:       Definieer  $F$  het hypervlak opgespannen door  $\{e_{I_1}^1, \dots, e_{I_{i-1}}^{i-1}, e_{I_{i+1}}^{i+1}, \dots, e_{I_q}^q\}$ 
9:       Definieer  $G$  het hypervlak opgespannen door bovenstaande endmembers en  $\mathbf{x}$ 
10:      for  $n \in [1, \dots, \text{aantal spectra}]$  do
11:         $p_n = \arcsin \left( \frac{|e_n^i - \text{proj}_G(e_n^i)|}{|e_n^i - \text{proj}_F(e_n^i)|} \right)$ 
12:        if  $(e_n^i - \text{proj}_F(e_n^i))(x - \text{proj}_F(x)) < 0$  then
13:           $p_n = \pi - p_n$ 
14:        end if
15:      end for
16:    end for
17:     $I_n = \text{argmin}_n(p_n)$ 
18:  end for
19:   $E = [e_{I_1}, \dots, e_{I_p}]$ 
20:  Abundantie en reconstructie-error kunnen bepaald worden aan de hand van het Fully constrained least-squares unmixing model met als endmembers  $E$ 
21: end for
22: geef het resultaat terug uit vorige stap met de laagste reconstructie-error.

```

---

# Hoofdstuk 4

## Spectrale variabiliteit bij niet-lineaire ontmenging

### 4.1 Multilineair MESMA

Volgens het optimaal *MESMA*-algoritme wordt ontmengd aan de hand van elke mogelijke combinatie van endmembers voor alle materialen. Dit ontmengalgoritme kan vrij gekozen worden. Niet-lineaire effecten en variabiliteit kunnen bijgevolg gecombineerd worden door het multilineair model in te voegen. Dit zorgt echter -zoals beschreven in vorig hoofdstuk- voor een te lange looptijd. Dit model wordt verder in deze thesis *multilineair MESMA* genoemd. Het voorgenoemde *MESMA* algoritme wordt vanaf nu *lineair MESMA* genoemd. Merk op dat deze methode niet gebruikt kan worden om niet-lineariteit in te voeren in hoekminimalisatie, aangezien deze laatste uitgaat van projecties op hypervlakken.

---

**Algoritme 4.1** multilineair MESMA

---

```
1:  $\eta = +\infty$ 
2: for Voor alle combinaties van spectra do
3:   Ontmeng gebruik makend van het multilineair model
4:   if reconstructie-error  $< \eta$  then
5:      $\eta =$  reconstructie-error
6:      $a_i =$  abundanties
7:   end if
8: end for
9:  $a_i$  zijn nu de correcte abundanties
```

---

**Algoritme 4.2** semilineair MESMA

---

```

1:  $\eta = +\infty$ 
2: for Voor alle deelverzamelingen do
3:   for Voor alle combinaties van spectra do
4:     Ontmeng gebruik makend van het sum-to-one constrained least-squares unmixing
       model
5:     if  $\exists \text{Abundantie} < 0$  then
6:       continue
7:     bepaal reconstructie-error
8:     end if
9:   end for
10: end for
11: Selecteer spectra voor indices voor de laagste reconstructie-error
12: Ontmeng gebruik makend van het multilineair model

```

---

## 4.2 Semi-lineair model

Om te looptijd van het algoritme te verkorten, wordt in plaats van elke combinatie multilineair te ontmengenen, elke combinatie lineair ontmengd aan de hand van *sum-to-one least-square unmixing*. Hierbij is men niet geïnteresseerd in de abundanties van elke combinatie, alleen in de indices - die aanduiden welke combinatie men beschouwd - en de reconstructie-error. De combinatie met de laagste reconstructie-error wordt teruggegeven en het systeem wordt eenmalig multilineair ontmengd aan de hand van de bekomen indices. Deze methode wordt verder *Semilineair MESMA* genoemd, en is een benadering van *multilineair MESMA*. Het is niet evident dat deze methode goede resultaten geeft, aangezien ontmengd wordt aan de hand van een ander model dan dat waarmee de spectra van de endmembers gekozen werden.

### 4.2.1 Theoretische controle met gesimuleerde data

De beste methode om na te gaan of *semilineair MESMA* een goede benadering is van *multilineair MESMA*, is ze allebei testen op experimentele data, en de resultaten ervan vergelijken. Hoewel dit gedeeltelijk gedaan wordt in volgend hoofdstuk, is dit wegens de looptijd van *multilineair MESMA* een langdurig proces. In dit hoofdstuk wordt in plaats hiervan een theoretisch model gemaakt waarvan de ontmenging van *multilineair MESMA* gekend is. Dit model wordt ontmengd aan de hand van *semilineair MESMA* en *lineair MESMA*. Dit theoretisch model maakt gebruik van de bibliotheek van de *Alina*-dataset maar niet van de spectra van de pixels.

### **Genereren van de data**

Eerst worden voor elke pixel willekeurige abundanties en een vaste reflectiekans gekozen. Het genereren van vier willekeurige abundanties die vervolgens worden genormeerd om te voldoen aan de eenheidssomvoorwaarde, geeft geen uniforme verdeling, en deze bevoordeelt gelijkmatige verdelingen. Daarom wordt gebruik gemaakt van de Diricletverdeling, welke een uniforme kansverdeling beschrijft in een simplex. Ten tweede wordt voor elk van de materialen een willekeurig spectrum gekozen uit de respectievelijke bibliotheek. Daarna worden deze materialen gemengd aan de hand van het multilineair model. Hierdoor is de reconstructie-error van *multilineair MESMA* exact 0. Ten slotte wordt het verkregen spectrum ontmengd aan de hand van zowel *semilineair MESMA* als *lineair MESMA*. Dit wordt gedaan voor verschillende gegeven reflectiekansen. Voor elke gegeven reflectiekans wordt de procedure 100 maal uitgevoerd, en de aangegeven waarden en foutenvlaggen zijn respectievelijk het gemiddelde en de variantie. De uiteindelijk data zijn te zien in figuur 4.1.

### **Correct geselecteerde materialen**

In figuur 4.2a staat het aantal juist geselecteerde spectra in functie van de gegeven reflectiekans. Deze geeft weer welk deel van de indices in *lineair* of *semilineair MESMA* - die voor beide modellen hetzelfde zijn - gelijk zijn aan de gegeven indices van het multilineair model. Zelfs voor een lage reflectiekans zoals 0.1 zijn maar 1 op 10 indices correct. Echter, het is mogelijk dat er spectra bestaan voor een specifiek materiaal die sterk lijken op andere spectra, en die daarom - hoewel deze verschillende indices hebben - grotendeels dezelfde resultaten geven. In theorie kan dit nagegaan worden door de hoek te bepalen tussen het werkelijke en het bepaalde spectrum, maar zoals te zien in figuur 4.2c, zijn de fouten op deze data groter dan de data zelf, en kan hieruit niets worden afgeleid.

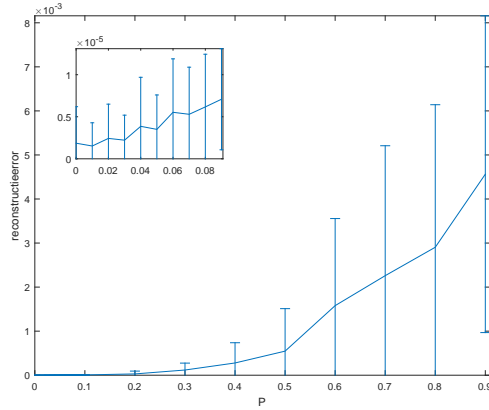
### **Reflectiekans**

Figuur 4.2b geeft de gemeten reflectiekans van *semilineair MESMA* weer. Aangezien men een lineaire benadering gebruikt die er vanuit gaat dat  $P = 0$ , zal de gemeten reflectiekans lager zijn dan de werkelijke reflectiekans.

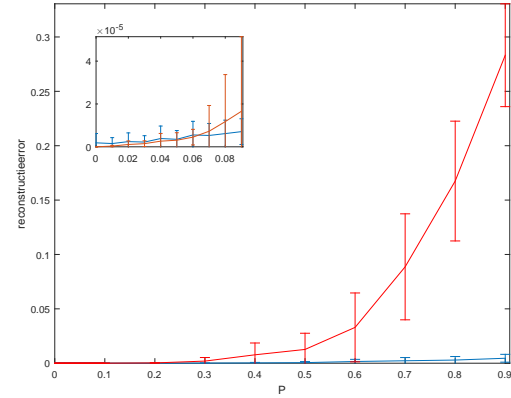
### **Reconstructie-error**

Figuur 4.1 beschrijft de reconstructie-error. De figuren 4.1a en 4.1b beschrijven de reconstructie-error, waarbij in figuur 4.1a de waarden van *lineair MESMA* niet weergegeven zijn, en in figuur 4.1b wel. Merk op dat de waarden van *semilineair MESMA* nog altijd op figuur 4.1b staan, maar dat deze zeer dicht bij de x-as liggen. De grafiek linksonder bevat dezelfde informatie, maar gebruikt de logaritmische schaal. Hetzelfde geldt voor de grafiek rechtsonder, maar deze vermeldt de waarden in functie van de gemeten reflectiekans in plaats van de werkelijke reflectiekans.

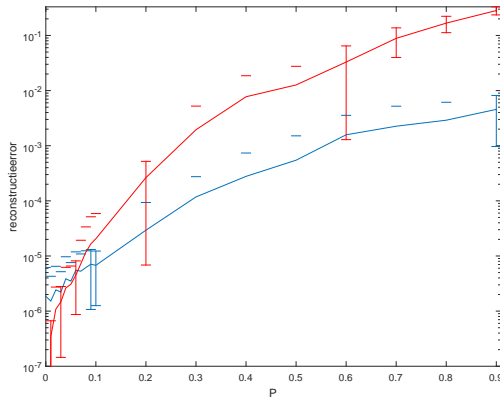
In deze vier grafieken is te zien dat de reconstructie-error van het *semilineair MESMA* twee grootte-orde kleiner is dan het lineair model.



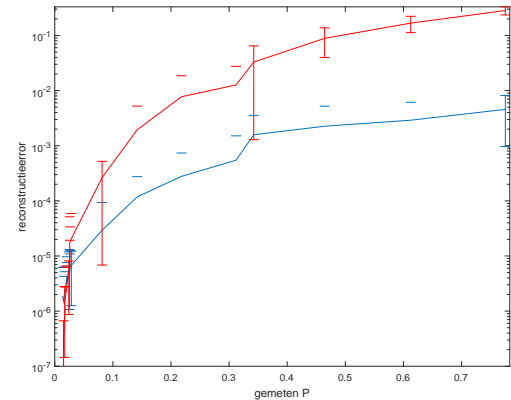
(a) Reconstructie-error van het semilineaire model in functie van de  $P$  waarde.



(b) Reconstructie-error van het lineaire en semilineaire model in functie van de  $P$  waarde.

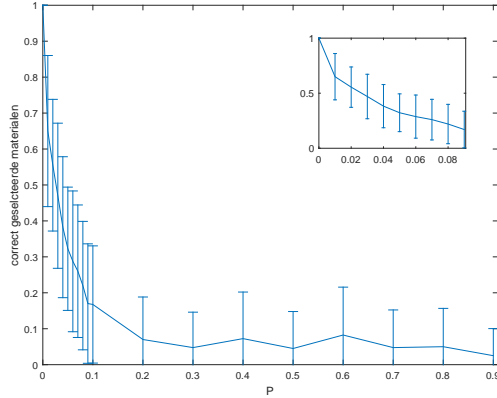


(c) Reconstructie-error van het semilineaire model in functie van de  $P$  waarde. De y-as is logaritmic

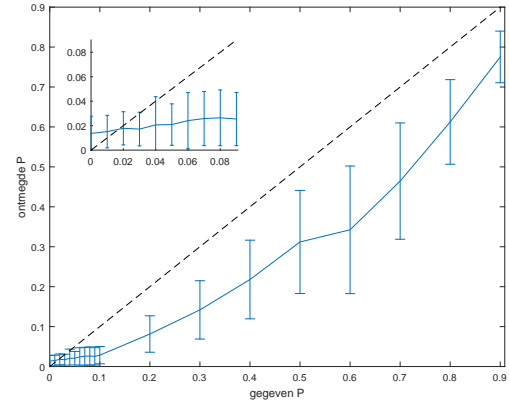


(d) Reconstructie-error van het semilineaire model in functie van de gemeten  $P$  waarde. De y-as is logaritmic

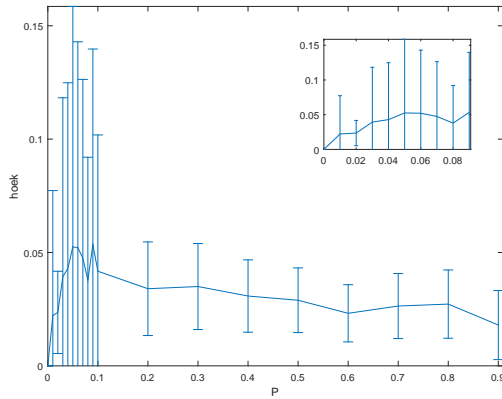
Figuur 4.1: Vergelijking tussen lineair, semilineair en multilineair MESMA. Voor elke  $P$  waarde zijn er 100 pixels berekend. Rood is het lineaire model en blauw het semilineaire model. Merk trouwens op dat wanneer het eindpunt van de onderste foutenvlag onder nul ligt, alleen de bovengrens getekend is en niet de foutenvlag zelf op de logaritmische grafieken. Deze figuren zijn gemaakt met *matlab*<sup>[7]</sup>



(a) Aantal correct geselecteerde materialen. Deze zijn hetzelfde voor het lineair en multilineair model.



(b) De gemeten  $P$  waarde voor het semilineair model in functie van de gegeven  $P$  waarde van het multilineair model. De stippelijn is de eerste bissectrice.



(c) Gevonden hoek in functie van de reflectiekans. Men ziet dat de foutenvlaggen veel groter zijn dan de data zelf. Deze figuur is gemaakt met *matlab*<sup>[7]</sup>

Figuur 4.2: Vergelijking tussen lineair, semilineair en multilineair MESMA. Voor elke  $P$  waarde zijn er 100 pixels berekend. Rood is het lineaire model en blauw het semilineaire model. Deze figuren zijn gemaakt met *matlab*<sup>[7]</sup>

### Maximale reflectie

Op de grafieken is te zien dat de reconstructie-error van het semilineaire model exact nul wordt als  $P = 1$ . Deze situatie zal in werkelijkheid niet voorkomen en is ook niet fysisch relevant. Als een lichtstraal een kans 1 heeft om terug op het materiaal te vallen, zal deze nooit op de detector vallen. Uit vergelijking 2.18 volgt:

$$\mathbf{x} = \frac{\sum_{i=1}^P (1 - P_i) a_i \mathbf{w}_i}{1 - \sum_{i=1}^P P_i a_i \mathbf{w}_i} \quad (4.1)$$

Vullen we  $P \rightarrow 1$  in dan krijgt men

$$\mathbf{x} = 0 \quad (4.2)$$

Dit geldt zowel voor de het reconstructiespectrum als het gemengde spectrum, en de reconstructie-error is dus de norm van de nulvector namelijk nul.

## 4.3 multilineair AAM

Aangezien *semilinear mesma* gebruikt maakt van *sum-to-one least-sqaure unmixing* kan hoekminimalisatie gebruikt worden om dit algoritme te versnellen, op analoge wijze als dit gebruikt werd voor *linear mesma*. De theoretische controle verloop gelijkaardig aan de vorige sectie, alleen gebeurt de selectie in het semilineair model en het lineair model aan de hand van hoekminimalisatie. Aangezien beide algoritmen voor de selecties gebruik maken van lineaire ontmenging, kan deze benadering doorgevoerd worden. Er is niets aangepast aan het genereren van de data, dus om de gegeven data terug te krijgen, moet nog steeds *optimal multilineair mesma* gebruikt worden.

De resultaten van de theoretische controle zijn te zien in figuren 4.3 en 4.4. De resultaten voor de reconstructie-error, de hoeken tussen de materialen en de gemeten  $P$ -waarden zijn hetzelfde als in sectie 4.2.1. Er is alleen een verandering in het aantal correct geselecteerde materialen.

### 4.3.1 correct geselecteerde materialen

Wanneer de reflectiekans klein is, is er geen verschil meer tussen de optimale lineaire, semilineaire en multilineaire *mesma*-algoritmen, dus gaat in de limiet het gedeelte correct gesimuleerde materialen naar 1. Dit is niet meer het geval voor hoekminimalisatie. Hoekminimalisatie is suboptimaal, dus geeft niet altijd de gegenereerde spectra terug. Hierdoor gaat de limiet van het gedeelte correct geselecteerde materialen niet naar 1, maar gaat deze naar een waarde rond 0.25. Ditzelfde effect gebeurt ook voor de reconstructie-errors. In voorgaande sectie was de limiet voor  $P \rightarrow 0$  gelijk aan nul, en nu is dit niet altijd het geval. Maar aangezien de gemiddelden nog steeds veel kleiner zijn dan de varianties, is dit in de grafieken alleen zichtbaar in de foutenvlaggen. In vorige sectie waren de foutenvlaggen van de reconstructie-error voor  $P = 0$  nul - op numerieke fouten na - omdat de gesimuleerde

waarden consequent gelijk zijn aan de gemeten waarden. Dit is niet meer het geval voor hoekminimalisatie.

---

**Algoritme 4.3** *multilinear Alternating Angle Minimalisation*


---

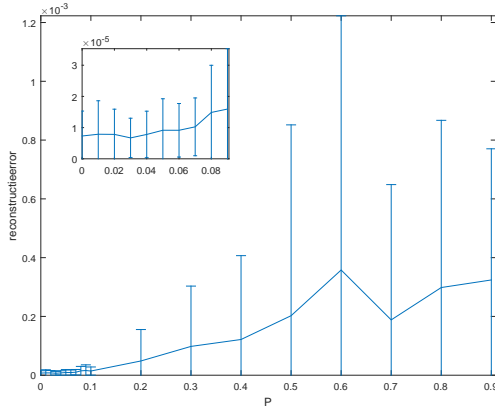
```

1: kies een aantal iteraties  $K$ 
2: for Voor elke deelverzameling van materialen do
3:   Stel  $q$  het aantal materialen
4:   Stel  $e_j^i$  het  $j$ -de spectrum uit de bibliotheek van het  $i$ -de materiaal.
5:   Selecteer een willekeurige index set  $I_j$ 
6:   for doe  $K$  keer do
7:     for voor  $i$  elk materiaal do
8:       Definieer  $F$  het hypervlak opgespannen door  $\{e_{I_1}^1, \dots, e_{I_{i-1}}^{i-1}, e_{I_{i+1}}^{i+1}, \dots, e_{I_q}^q\}$ 
9:       Defineer  $G$  het hypervlak opgespannen door bovenstaande endmembers en  $\mathbf{x}$ 
10:      for  $n \in [1, \dots, \text{aantal spectra}]$  do
11:         $p_n = \arcsin \left( \frac{|e_n^i - \text{proj}_G(e_n^i)|}{|e_n^i - \text{proj}_F(e_n^i)|} \right)$ 
12:        if  $(e_n^i - \text{proj}_F(e_n^i))(x - \text{proj}_x F) < 0$  then
13:           $p_n = \pi - p_n$ 
14:        end if
15:      end for
16:    end for
17:     $I_n = \text{argmin}_n(p_n)$ 
18:  end for
19:   $E = [e_{I_1}, \dots, e_{I_p}]$ 
20:  reconstructie-error kan bepaald worden aan de hand van het Fully constrained least-squares unmixing model met als endmembers  $E$ 
21: end for
22: Neem de indices met de laagste reconstructie-error uit vorige stap
23: bereken de abundanties en reflectiekans door multilineair te ont mengen.

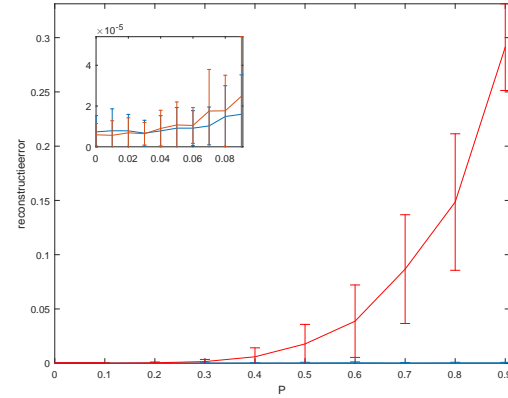
```

---

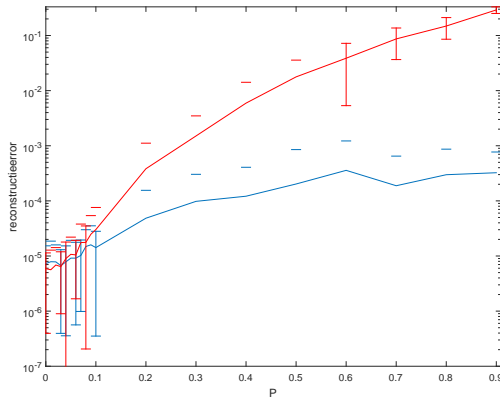




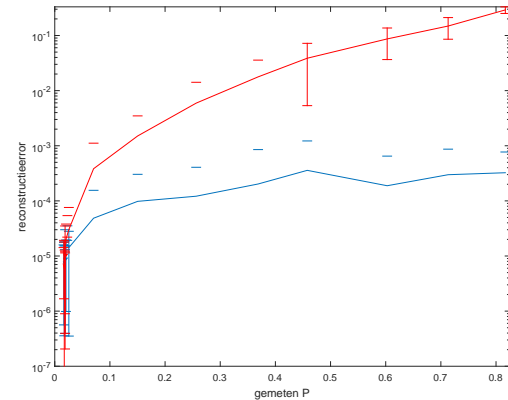
(a) Reconstructie-error van het semilineaire model in functie van de  $P$  waarde.



(b) Reconstructie-error van het lineaire en semilineaire model in functie van de  $P$  waarde.

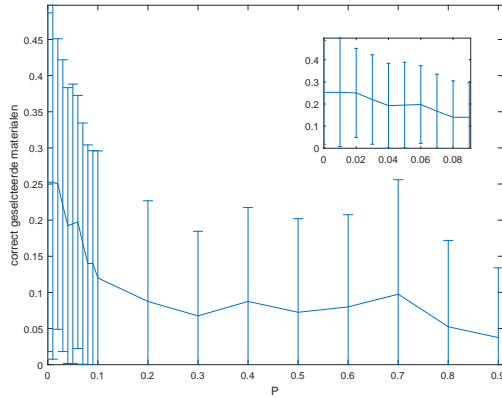


(c) Reconstructie-error van het semilineaire model in functie van de  $P$  waarde. De y-as is logaritmisch

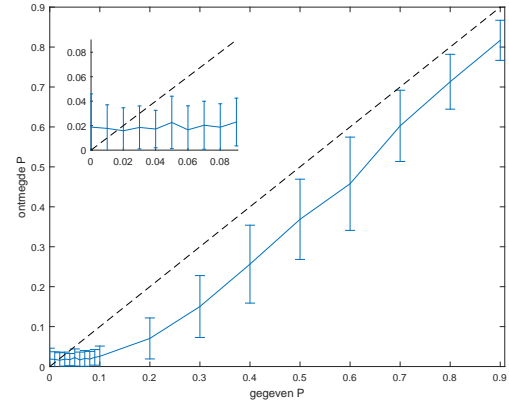


(d) Reconstructie-error van het semilineaire model in functie van de gemeten  $P$  waarde. De y-as is logaritmisch

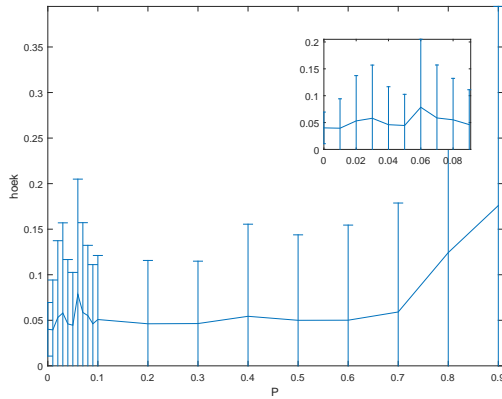
Figuur 4.3: Vergelijking tussen lineair, semilineair en multilineair MESMA. Voor elke  $P$  waarde zijn er 100 pixels berekend. Rood is het lineaire model en blauw het semilineaire model. Merk trouwens op dat wanneer het eindpunt van de onderste foutenvlag onder nul ligt, alleen de bovengrens getekend is en niet de foutenvlag zelf op de logaritmische grafieken. Deze figuren zijn gemaakt met *matlab*<sup>[7]</sup>



(a) Aantal correct geselecteerde materialen. Deze zijn hetzelfde voor het lineair en multilineair model.



(b) De gemeten  $P$  waarde voor het semilineair model in functie van de gegeven  $P$  waarde van het multilineair model. De stippelijn is de eerste bissectrice.



(c) Gevonden hoek in functie van de reflectiekans. Men ziet dat de foutenvlaggen veel groter zijn dan de data zelf. Deze figuur is gemaakt met *matlab*<sup>[7]</sup>

Figuur 4.4: Vergelijking tussen lineair, semilineair en multilineair MESMA. Voor elke  $P$  waarde zijn er 100 pixels berekend. Rood is het lineaire model en blauw het semilineaire model. Deze figuren zijn gemaakt met *matlab*<sup>[7]</sup>

# Hoofdstuk 5

## Experimentele vergelijking van verschillende methodes

In dit hoofdstuk worden verschillende modellen toegepast op de *Alina*-dataset om deze te vergelijken op verschillende aspecten. Ook worden verschillende opties voor een model - bijvoorbeeld het wel of niet toepassen van de niet-negativiteitsvoorwaarde - met elkaar vergeleken. Ook worden alleen modellen beschouwd die variabiliteit bevatten.

### 5.1 Looptijd en reconstructie-error

Een algoritme heeft twee belangrijke aspecten. Ten eerste moet het zo nauwkeurig mogelijk zijn. Dit komt overeen met een zo laag mogelijke reconstructie-error. Ten tweede moet het algoritme efficiënt zijn, dus een zo kort mogelijke looptijd. Deze twee aspecten zijn meestal niet verenigbaar. Een eenvoudig algoritme zoals lineair ontmengen heeft een korte looptijd maar een grote reconstructie-error; een complex algoritme zoals *multilineair MESMA* heeft een lage reconstructie-error, maar een lange looptijd. In praktijk wordt de overweging gemaakt hoe nauwkeurig of hoe snel de data moeten worden verwerkt, en aan de hand daarvan wordt een algoritme gekozen.

### 5.2 De vergeleken methodes

In dit hoofdstuk worden vijf methoden vergeleken. In tabel 5.2 staan de secties en de pseudocodes die gebruikt worden in deze thesis.

	$P > 0$	$P > -1$
afhankelijke $P$	Real	Free
onafhankelijke $P$	Strict	Strong

Tabel 5.1: Benamingen van de verschillende parameteropties voor multilineair ontmengen

algoritme	sectie	algoritme
lineair MESMA	3.1	3.2
semi-lineair MESMA	4.1	4.1
multi-lineair MESMA	4.2	4.2
lineair AAM	3.2	3.3
multilineair AAM	4.3	4.3

Tabel 5.2: De vijf beschouwde methoden

x-pos	3	5	9	12	14	17
real multilineair	6776.9	6246.9	6009.1	6063.3	6097.4	6060.3
free multilineair	6597.8	6533.0	6490.3	6492.1	5707.1	6495.9
strict multilineair	5746.3	5486.6	6132.5	6050.9	5405.2	5676.8
strong multilineair	5603.9	5461.7	5284.3	5412.6	5699.3	5553.0
lineair	2.0421	2.0435	2.0213	2.0154	2.0175	2.0103
strict semilineair	2.0459	2.1301	2.0659	2.0590	2.0527	2.0735
strong semilineair	2.0324	2.1236	2.0509	2.0482	2.0637	2.0726
free semilineair	2.1488	2.1105	2.1255	2.1304	2.0722	2.1466
real semilineair	2.0982	2.1208	2.0752	2.0880	2.0628	2.1169
AAM	0.1086	0.0543	0.0538	0.0543	0.0533	0.0536
strict AAM ML	0.0922	0.1567	0.0980	0.1007	0.0805	0.1143
strong AAM ML	0.0876	0.1564	0.0833	0.0829	0.1018	0.0906
free AAM ML	0.1638	0.1477	0.1629	0.1713	0.1129	0.1618
real AAM ML	0.1625	0.1473	0.1129	0.1147	0.1024	0.1377

Tabel 5.3: Looptijd van de verschillende algoritmen voor verschillende pixels, in seconden. Deze zijn allen gekozen voor een pixel met  $y = 3$ . Deze pixels bevatten voor de meerderheid respectievelijk: bos; bos en asfalt; asfalt; asfalt en voetpad; voetpad en grass; grass. Deze indeling is gemaakt op basis van wat zich werkelijk op de grond bevindt, en kan bijgevolg verschillen van de gemeten abundanties.

We vergelijken niet alleen de verschillende methodes, maar ook, waar toepasselijk, de verschillende opties van het multilineair model. Ten eerste wordt de afhankelijkheid met de onafhankelijkheid van de reflectiekans van het materiaal vergeleken. Ten tweede wordt het al dan niet toelaten van negatieve reflectiekansen bekeken. De optie die het meest fysisch relevant is, namelijk dat reflectiekansen afhankelijk zijn en positief, wordt *real* genoemd. De optie met het meeste vrijheid, waarbij de reflectiekansen afhankelijk zijn en negatieve waarden kunnen aannemen wordt *free* genoemd. De tegengestelde optie, met een onafhankelijke positieve reflectiekans wordt *strict* genoemd en de overgebleven optie, met onafhankelijke al dan niet negatieve kansen wordt *strong* genoemd.

Er worden gebieden geselecteerd van pixels die volgens de groundtruth ongeveer dezelfde samenstelling hebben. Voor elk gebied wordt de gemiddelde reconstructie-error en looptijd berekend, en deze worden uitgemiddeld. Voor de modellen gebruik makend van *multilineair mesma* wordt -vanwege de lange looptijd- slechts een enkele pixel gekozen. Deze pixel wordt

x-pos	3	5	9	12	14	17
real multilinear	0.0016	0.0020	0.0045	0.0058	0.0535	0.0070
free multilinear	0.0016	0.0020	0.0011	0.0018	0.0060	0.0062
strict multilinear	0.0017	0.0020	0.0045	0.0058	0.0535	0.0070
strong multilinear	0.0017	0.0020	0.0011	0.0018	0.0060	0.0063
linear	0.0835	0.0023	0.0045	0.0058	0.0535	0.0070
strict semilinear	0.0023	0.0020	0.0045	0.0058	0.0535	0.0070
strong semilinear	0.0023	0.0020	0.0014	0.0021	0.0078	0.0070
free semilinear	0.0022	0.0020	0.0013	0.0021	0.0077	0.0066
real semilinear	0.0023	0.0020	0.0045	0.0058	0.0535	0.0070
AAM	0.0835	0.0023	0.0045	0.0058	0.0535	0.0070
strict AAM ML	0.0022	0.0020	0.0045	0.0058	0.0535	0.0077
strong AAM ML	0.0022	0.0020	0.0014	0.0021	0.0078	0.0070
free AAM ML	0.0021	0.0020	0.0013	0.0021	0.0077	0.0066
real AAM ML	0.0022	0.0020	0.0045	0.0058	0.0535	0.0070

Tabel 5.4: Reconsturctie-error van de verschillende algoritmen voor verschillende pixels. Deze zijn allen gekozen voor een pixel met  $y = 3$ . Deze pixels bevatten voor de meerderheid respectievelijk: bos;bos en asfalt;asfalt;asfalt en voetpad; voetpad en grass; grass. Deze indeling is gemaakt op basis van wat zich werkelijk op de grond bevind, en kan bijgevolg verschillen van de gemeten abundanties.

x-pos	3	5	9	12	14	17
real multilinear	0.0189	0.8820	1.0000	1.0000	1.0000	0.9056
free multilinear	0.0186	0.8695	0.2490	0.3136	0.1129	0.8054
strict multilinear	0.0200	0.8932	1.0000	1.0000	1.0000	0.9058
strong multilinear	0.0200	0.8932	0.2490	0.3137	0.1129	0.8213
linear	1.0000	1.0000	0.9999	0.9998	1.0000	0.9056
strict semilinear	0.0276	0.8960	1.0000	1.0000	1.0000	0.9058
strong semilinear	0.0276	0.8960	0.3060	0.3631	0.1460	0.9049
free semilinear	0.0268	0.8773	0.2946	0.3574	0.1447	0.8554
real semilinear	0.0271	0.8935	1.0000	1.0000	1.0000	0.9056
AAM	1.0000	1.0000	0.9999	0.9998	1.0000	0.9056
strict AAM ML	0.0268	0.8960	1.0000	1.0000	1.0000	1.0000
strong AAM ML	0.0268	0.8960	0.3060	0.3631	0.1460	0.9049
free AAM ML	0.0256	0.8773	0.2946	0.3574	0.1447	0.8554
real AAM ML	0.0260	0.8935	1.0000	1.0000	1.0000	0.9056

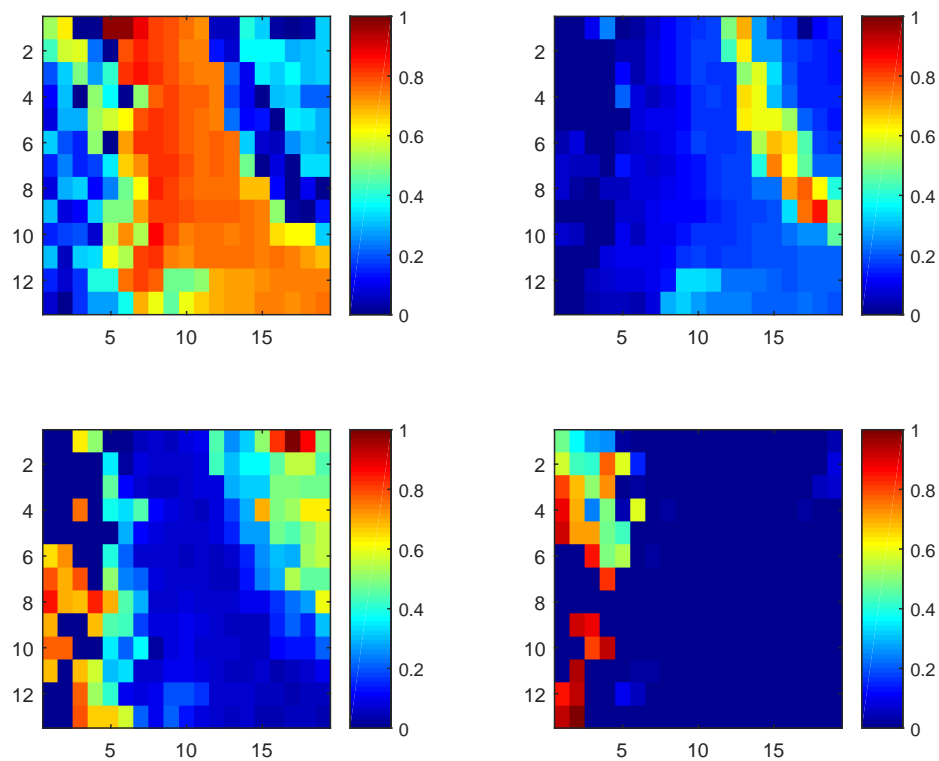
Tabel 5.5: Reconsturctie-error van de verschillende algoritmen voor verschillende pixels, genormaliseerd met de grootste error voor elke pixel. Deze zijn allen gekozen voor een pixel met  $y = 3$ . Deze pixels bevatten voor de meerderheid respectievelijk: bos;bos en asfalt;asfalt;asfalt en voetpad; voetpad en grass; grass. Deze indeling is gemaakt op basis van wat zich werkelijk op de grond bevind, en kan bijgevolg verschillen van de gemeten abundanties.

lineair	2.0095	2.0195	2.0027	2.0003	2.0025	2.0128
strict semilineair	2.0615	2.0809	2.0646	2.0533	2.0381	2.0633
strong semilineair	2.0354	2.0541	2.0409	2.0442	2.0403	2.0466
free semilineair	2.0934	2.1167	2.1152	2.1110	2.1104	2.1032
real semilineair	2.0728	2.0918	2.0747	2.0549	2.0534	2.0742
AAM	0.0559	0.0553	0.0550	0.0550	0.0548	0.0548
strict AAM ML	0.1092	0.1187	0.1124	0.1010	0.0933	0.0944
strong AAM ML	0.0891	0.0974	0.0930	0.0970	0.0963	0.0870
free AAM ML	0.1520	0.1645	0.1659	0.1667	0.1568	0.1480
real AAM ML	0.1274	0.1402	0.1264	0.1293	0.1090	0.1115

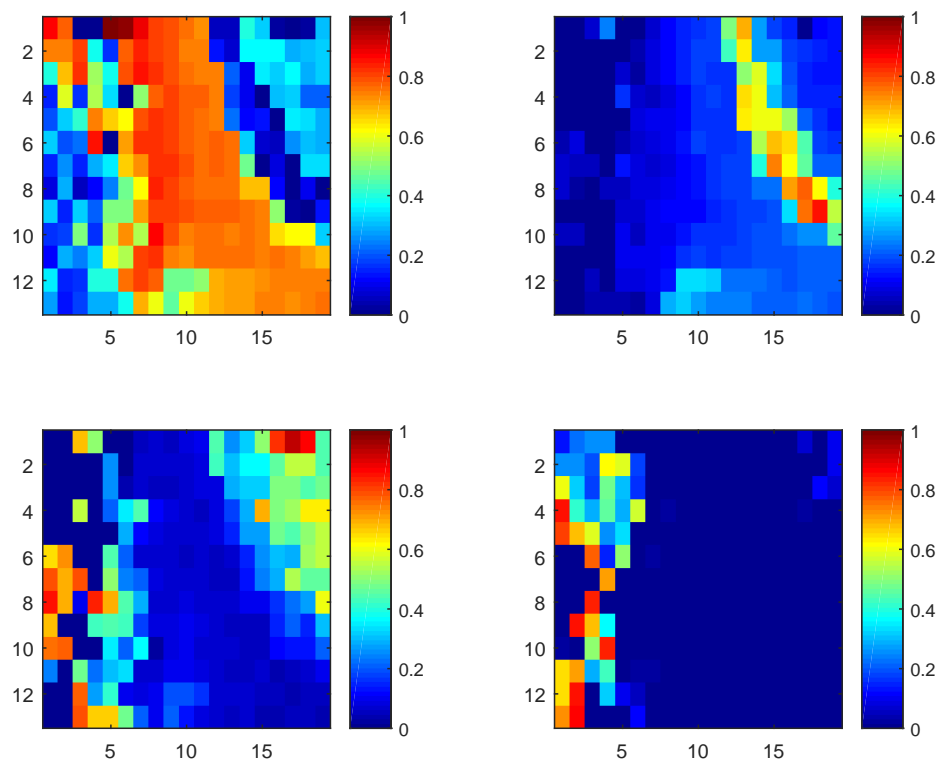
Tabel 5.6: Looptijd van de verschillende algoritmen voor verschillende pixels, in seconden. Deze zijn uitgemiddeld voor een gebied voor  $y$  gaande van 3 tot 8. Deze pixels bevatten voor de meerderheid respectievelijk: bos; bos en asfalt; asfalt; asfalt en voetpad; voetpad en grass; grass. Deze indeling is gemaakt op basis van wat zich werkelijk op de grond bevindt, en kan bijgevolg verschillen van de gemeten abundanties.

lineair	0.0156	0.0127	0.0036	0.0093	0.0255	0.0201
strict semilineair	0.0030	0.0027	0.0036	0.0093	0.0255	0.0201
strong semilineair	0.0030	0.0027	0.0018	0.0023	0.0057	0.0099
free semilineair	0.0028	0.0026	0.0018	0.0023	0.0055	0.0092
real semilineair	0.0030	0.0027	0.0036	0.0093	0.0255	0.0201
AAM	0.0156	0.0127	0.0036	0.0093	0.0257	0.0202
strict AAM ML	0.0030	0.0026	0.0036	0.0093	0.0257	0.0202
strong AAM ML	0.0029	0.0026	0.0019	0.0023	0.0055	0.0104
free AAM ML	0.0028	0.0025	0.0018	0.0023	0.0053	0.0098
real AAM ML	0.0029	0.0026	0.0036	0.0093	0.0257	0.0202

Tabel 5.7: Reconsturctie-error van de verschillende algoritmen voor verschillende pixels. Deze zijn uitgemiddeld voor een gebied voor  $y$  gaande van 3 tot 8. Deze pixels bevatten voor de meerderheid respectievelijk: bos; bos en asfalt; asfalt; asfalt en voetpad; voetpad en grass; grass. Deze indeling is gemaakt op basis van wat zich werkelijk op de grond bevindt, en kan bijgevolg verschillen van de gemeten abundanties.



Figuur 5.1: Abundanties van de verschillende waarden gebruik gemaakt van *strict semilineair mesma*. Deze is exact hetzelfde als *strict multilineair AAM* Deze afbeelding is gemaakt met MATLAB<sup>[7]</sup>



Figuur 5.2: Abundanties van de verschillende waarden gebruik gemaakt van *linear mesma*. Deze is exact hetzelfde als *AAM* Deze afbeelding is gemaakt met MATLAB<sup>[7]</sup>



vergeleken met de andere reconstructie-errors van andere methode voor deze zelfde pixel. In tabel 5.3, 5.4 en 5.5 staan, voor alle beschouwde methodes en opties, de reconstructie-error en de looptijd voor een enkel pixel voor alle modellen, en in tabellen 5.6 en ?? staat de gemiddelde looptijd en de reconstructie-error voor gebieden voor de modellen net relatief korte looptijd.

## 5.3 Bespreking van de resultaten

### 5.3.1 Lineair vs semilineair

Een methode waarbij de abundantie lineair berekend wordt vergeleken met een methode die dit multilineair doet, gegeven dat de selectie van de spectra gebeurt op lineaire manier. Concreet betekent dit dat het lineair met het semilineair model vergeleken wordt en het hoekminimalisatiemodel met het multilineair hoekminimalisatiemodel. Voor alle pixels geldt dat de looptijd ongeveer gelijk is, aangezien de bottleneck ligt bij het selecteren van de spectra en niet bij het ontmengen. In de pixels met gras of asfalt is de reconstructie-error bij beide methoden gelijk. De reflectiekans is laag, dus het lineaire model is een goede benadering. Echter, bij de pixel met bebossing, waar de reflectiekans hoog is, geeft het semilineair model een beter resultaat.

### 5.3.2 Semi-lineair vs multilineair

Ten tweede vergelijken we het semi-lineair model met het multilineair model. In alle pixels geven beide modellen een vergelijkbare reconstructie-error, zelfs in pixels waar deze modellen een sterke verbetering geven op het lineair model. De looptijd van het semilineair algoritme is veel kleiner dan die van het multilineair algoritme.

### 5.3.3 P-afhankelijkheid

Om het belang van endmemberafhankelijkheid te bepalen, worden algoritmes die deze onafhankelijkheid wel meenemen vergeleken met algoritmes die dit niet meenemen. In het algoritme zit het enige verschil in de randvoorwaarde van het `fmincon` algoritme, en is de looptijd dus gelijk, wat ook beaamd wordt in het experiment. Men verwacht ook een sterke reductie in de reconstructie-error, omdat enerzijds dit model fysisch correcter is, en anderzijds dit meer vrijheidsgraden heeft. Echter, het experiment geeft slechts een verwaarloosbare reductie in de reconstructie-error. Aangezien hiervoor een verdubbeling van het aantal vrijheidsgraden nodig is, vormt P-afhankelijkheid geen verbetering.

### 5.3.4 Bespreking P-ondergrens

Het aanpassen van de ondergrens van  $P$  geeft geen verschil in de looptijd, de enige verandering in het algoritme gebeurt in de randvoorwaarden van `fmincon`. Voor de gevallen met bos of gras geeft dit een vergelijkbare reconstructie-error, aangezien het aanpassen van randvoorwaarden geen invloed heeft op resultaten die niet in de buurt van de rand liggen. Echter, voor asfalt, waarvoor geldt dat  $P = 10^{-5}$ , ligt  $P$  dicht bij de rand, en verwacht men dus een verbetering in de reconstructie-error na een verandering in de ondergrens, wat beaamd wordt door het experiment. Echter, voor sommige gebieden geeft dit resultaten terug waar  $P \approx -1$ , en geeft deze sterk verschillende resultaten terug in vergelijking met de andere modellen. Dit kan wijzen op een vorm van overfitting, waarbij resultaten die niet werkelijk voorkomen bij toeval goede resultaten geven.

### 5.3.5 Bepreking multilineair AAM vs semilineair model

In tegenstelling tot wat voorspeld werd in sectie 4.2, geeft in het experiment hoekminimalisatie bijna altijd dezelfde resultaten als het lineair model, en elk semilineair model bijna altijd dezelfde resultaten als elk respectievelijk multilineair hoekminimalisatie model. Niet alleen dezelfde reconstructie-error, maar omdat deze dezelfde indices kiest, maar ook dezelfde abundaties en reflectiekansen. Doordat voor sommige pixels de suboptimale en optimale spectra niet hetzelfde zijn, zorgt dit ervoor dat het gemiddelde een kleine hoeveelheid hoger is voor AAM. Echter, de looptijd is veel korter.

Gebruikt makend van de vorige vergelijkingen is het meest nuttige model een semilineair model waarbij de reflectiekans onafhankelijk is van de endmember en tussen  $-1$  en  $1$  ligt. Dit is *strong multilineair AAM*.

# Bibliografie

- [1] Jose M. Bioucas-dias, Antonio Plaza, Gustavo Camps-Valls, Paul Scheunders, Nasser M. Nasrabadi, and Jocelyn Chanussot. Hyperspectral remote sensing data analysis and future challenges. *Geoscience and remote sensing magazine*, pages 6–36, June 2013.
- [2] Jose M. Bioucas-dias, Antonio Plaza, Nicolas Dobigeon, Mario Parente, Qian Du, Pual Grader, and Jocelyn Chanussot. Hyperspectral unmixing overview: Geometrical, statistical and sparse regression based approaches. *Journal of selected topics in applied earth opservation and remote sensing*, 5(2):354–376, April 2012.
- [3] Canada centre for remote sensing. *Fundamentals of remote sensing*. Natural Resources Canada.
- [4] Xiaoxiao Du, Alina Zare, Paul Gader, and Dmitri Dranishnikov. Spatial and spectral unmixing using the beta compositional model. *Journal of selected topics in applied earth opservation and remote sensing*, 7(6):1994–2003, June 2014.
- [5] Rob Heylen and Paul Scheunders. A multilinear mixing model for nonlinear spectral unmixing.
- [6] Rob Heylen, Alina Zare, Paul Gader, and Paul Scheunders. Hyperspectral unmixing with endmember variability via alternating angle minimization.
- [7] MATLAB. *version 8.5.0 (R2015a)*. The MathWorks Inc., Natick, Massachusetts, 2015.