

# UNIVERSITÉ DE LILLE

## THÈSE

pour obtenir le grade de :

DOCTEUR DE L'UNIVERSITÉ DE LILLE

dans la spécialité

AUTOMATIQUE, GÉNIE INFORMATIQUE, TRAITEMENT DU SIGNAL ET DES IMAGES

par

**Anis Amziane**

---

### **Attributs de texture extraits d'images multispectrales acquises en conditions d'éclairage non contrôlées—Application à l'agriculture de précision.**

---

Thèse soutenue le 18 octobre 2022 devant le jury composé de :

- Pr. David Rousseau, Université d'Angers (Rapporteur)
- Pr. Christelle Gée, Institut AgroSup Dijon (Rapporteuse)
- Pr. Gilles Roussel, Université du Littoral-Côte d'Opale (Président)
- Pr. Christine Fernandez-Maloigne, Université de Poitiers (Examinatrice)
- Dr. Aurélien Duménil, Chambre d'Agriculture de la Somme (Examineur)
- Pr. Ludovic Macaire, Université de Lille (Directeur de Thèse)
- Dr. Olivier Losson, Université de Lille (Co-encadrant)
- Dr. Benjamin Mathon, Université de Lille (Co-encadrant)

**Laboratoire CRISTAL**





**UNIVERSITÉ DE LILLE**  
**DOCTORAL THESIS**

A thesis submitted in fulfillment of the requirements  
for the degree of  
DOCTOR OF PHILOSOPHY

BY  
ANIS AMZIANE

---

**Texture features extracted from multispectral images  
acquired under uncontrolled illumination  
conditions—Application to precision farming.**

---

Thesis defended on October 18, 2022 in front of the jury composed of:

- Pr. David Rousseau, Université d'Angers (reviewer)
- Pr. Christelle Gée, Institut AgroSup Dijon (reviewer)
- Pr. Gilles Roussel, Université du Littoral-Côte d'Opale (president)
- Pr. Christine Fernandez-Maloigne, Université de Poitiers (examinor)
- Dr. Aurélien Duménil, Chambre d'Agriculture de la Somme (examinor)
- Pr. Ludovic Macaire, Université de Lille (supervisor)
- Dr. Olivier Losson, Université de Lille (co-supervisor)
- Dr. Benjamin Mathon, Université de Lille (co-supervisor)

**CRISTAL laboratory**



*In memory of my grandfather,  
Amziane Hamatache*



## *Acknowledgements*

The work described in this document has been carried out within the Color Imaging team of the CRISAL laboratory, at the university of Lille, France. This work has been co-funded by the university of Lille, the Hauts-de-France region, and the Chambre d'Agriculture de la Somme. It would not have been completed without the support and help of many people.

First of all, I would like to express my gratitude to my supervisors: Pr. Ludovic Macaire, Dr. Olivier Losson, and Dr. Benjamin Mathon. Thank you for your confidence, guidance, and investment in the supervision of my thesis. I have learned so much from you over the past four years. Also, I would like to thank Dr. Aurélien Duménil from the Chambre d'Agriculture de la Somme for his contribution and investment in the image acquisition campaign, and Laurence Delbarre from Ir-DIVE platform for providing technical support and essential materials for my experiments.

Special thanks to my thesis committee members for their time and effort in evaluating this work: Pr. Gilles Roussel, Pr. Christelle Gée, Pr. David Rousseau, and Pr. Christine Fernandez-Maloigne. Thank you for the stimulating discussion and insightful comments/questions during the thesis defense.

Further, I thank all my colleagues from the Image Group and the Faculty of Science and Technology (P2-building) for the discussions and friendly atmosphere during the coffee breaks. I also thank my friends Djamel and Djafar for encouraging me, and Melissa for her unconditional support.

Last but not least, I would like to express my deepest gratitude to my beloved grandparents, my parents, and my sister for their prayers and constant support throughout my entire life.



UNIVERSITÉ DE LILLE

*Abstract*

CRIStAL laboratory

Doctor of Philosophy

**Texture features extracted from multispectral images acquired under uncontrolled illumination conditions—Application to precision farming.**

by Anis AMZIANE

**P**RECISION spraying aims to fight weeds in crop fields while reducing herbicide use by exclusive weed targeting. Among available imaging technologies, multispectral (multishot) cameras sample the scene radiance according to narrow spectral bands in the visible and/or near infrared domains and provide multispectral radiance images with many spectral channels. The main objective of this work is to develop an automatic recognition system of crop and weed plants in field conditions based on multispectral imaging.

In this manuscript, we describe the formation of multispectral radiance images under the Lambertian surface assumption, and provide a formalization of the linescan multispectral camera used in this study. We then propose an original multispectral image formation model that takes illumination variation during image acquisition into account. From our image formation model, we propose a method to estimate the reflectance as an illumination-invariant spectral signature. The quality of reflectance estimated by our method is evaluated against state-of-the-art methods, and its contribution to supervised crop/weed recognition is demonstrated. As spectral bands associated to the acquired channels may be redundant or contain highly correlated spectral information, we select the best spectral bands for crop/weed identification. We then use them to specify a single-sensor (snapshot) camera model suited for outdoor crop/weed recognition. Finally, we propose an original approach based on a convolutional neural network for spatio-spectral feature extraction from multispectral images at reduced computation costs. Extensive experiments show the contribution of our approach to outdoor crop/weed recognition.





## Résumé

Laboratoire CRIStAL

Docteur

**Attributs de texture extraits d'images multispectrales acquises en conditions d'éclairage non contrôlées—Application à l'agriculture de précision.**

Anis AMZIANE

**L**A pulvérisation de précision vise à lutter contre les adventices des cultures tout en ciblant exclusivement les plantes indésirables pour réduire les quantités utilisées d'herbicides. Parmi les technologies d'imagerie disponibles, les caméras multispectrales (multishot) échantillonnent la radiance de la scène selon plusieurs bandes spectrales étroites dans les domaines du visible et/ou du proche infrarouge et fournissent des images multispectrales de radiance composées de plusieurs canaux spectraux. L'objectif principal de ce travail est de développer un système de reconnaissance automatique des cultures et des adventices en plein champ, basé sur l'imagerie multispectrale.

Dans ce manuscrit, nous décrivons la formation d'images multispectrales de radiance sous l'hypothèse de surface lambertienne, et fournissons une formalisation de la caméra multispectrale à balayage linéaire utilisée dans cette étude. Nous proposons ensuite un modèle original de formation d'images multispectrales qui prend en compte la variation de l'éclairage pendant leur acquisition. À partir de ce modèle, nous proposons une méthode pour estimer la réflectance des espèces en présence, qui en est une signature spectrale invariante à l'éclairage. La qualité de la réflectance estimée par notre méthode est évaluée par rapport aux méthodes de l'état de l'art, et nous montrons sa contribution à la reconnaissance supervisée des cultures et des adventices. Comme les bandes spectrales associées aux canaux acquis peuvent être redondantes ou contenir des informations spectrales fortement corrélées, nous sélectionnons les meilleures bandes spectrales pour l'identification des cultures et adventices. Nous les utilisons ensuite pour spécifier un modèle de caméra mono-capteur

(de type instantané, ou snapshot) adapté à la reconnaissance des cultures et adventices en plein champ. Enfin, nous proposons une approche basée sur un réseau de neurones convolutifs pour l'extraction d'attributs spatio-spectraux à partir d'images multispectrales à coût calculatoire réduit. Des expériences approfondies montrent la contribution de notre approche à la reconnaissance des cultures et adventices en extérieur.

# Contents

|          |                                                  |           |
|----------|--------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                              | <b>1</b>  |
| 1.1      | Precision farming                                | 4         |
| 1.1.1    | Overview                                         | 4         |
| 1.1.2    | Challenges                                       | 5         |
| 1.2      | State of the art in crop/weed imaging            | 6         |
| 1.2.1    | Remote sensing                                   | 6         |
| 1.2.2    | Proximal sensing                                 | 7         |
| 1.2.3    | State of the art about crop/weed recognition     | 9         |
| 1.2.4    | Limitation and challenges                        | 13        |
| 1.3      | CA80 project                                     | 14        |
| 1.3.1    | Aims and planning                                | 15        |
| 1.3.2    | Examined crops and weeds                         | 16        |
| 1.4      | Thesis contributions and structure               | 19        |
| 1.4.1    | Contributions                                    | 19        |
| 1.4.2    | Manuscript structure                             | 20        |
| <b>2</b> | <b>Multispectral imaging</b>                     | <b>23</b> |
| 2.1      | Introduction                                     | 24        |
| 2.2      | Illumination and surfaces                        | 25        |
| 2.2.1    | Bands and wavelengths                            | 25        |
| 2.2.2    | Radiance and reflectance                         | 27        |
| 2.3      | Multispectral radiance image acquisition devices | 27        |
| 2.3.1    | Multispectral radiance image                     | 28        |
| 2.3.2    | Snapshot devices                                 | 29        |
| 2.3.3    | Multishot devices                                | 30        |

|                                                                        |           |
|------------------------------------------------------------------------|-----------|
| <b>2.4 Multispectral radiance image acquisition by Snapscan camera</b> | <b>33</b> |
| 2.4.1 Snapscan camera . . . . .                                        | 33        |
| 2.4.2 Radiance measurement . . . . .                                   | 35        |
| 2.4.3 Frame acquisition . . . . .                                      | 36        |
| 2.4.4 Stripe assembly . . . . .                                        | 38        |
| 2.4.5 Image formation model for Snapscan camera . . . . .              | 39        |
| <b>2.5 Conclusion</b> . . . . .                                        | <b>40</b> |
| <b>3 Reflectance estimation</b>                                        | <b>43</b> |
| 3.1 Introduction . . . . .                                             | 44        |
| 3.2 State-of-the-art about reflectance estimation . . . . .            | 44        |
| 3.2.1 Reflectance estimation under constant illumination . . . . .     | 46        |
| 3.2.2 Reflectance estimation under varying illumination . . . . .      | 48        |
| 3.3 Reference reflectance estimation . . . . .                         | 50        |
| 3.3.1 Assumption validation . . . . .                                  | 51        |
| 3.3.2 Reflectance estimation . . . . .                                 | 55        |
| 3.3.3 Spectral correction . . . . .                                    | 57        |
| 3.3.4 Negative value removal . . . . .                                 | 58        |
| 3.4 Reflectance estimation under varying illumination . . . . .        | 59        |
| 3.4.1 White diffuser in the scene . . . . .                            | 60        |
| 3.4.2 Vignetting correction . . . . .                                  | 61        |
| 3.4.3 Row-wise (rw) based reflectance estimation . . . . .             | 63        |
| 3.4.4 Optimized row-wise (orw) based reflectance estimation . . . . .  | 66        |
| 3.5 Conclusion . . . . .                                               | 68        |
| <b>4 Reflectance estimation assessment</b>                             | <b>71</b> |
| 4.1 Introduction . . . . .                                             | 73        |
| 4.2 Multispectral vegetation databases . . . . .                       | 73        |
| 4.3 Multispectral dataset acquisition . . . . .                        | 74        |
| 4.3.1 Experimental setup . . . . .                                     | 74        |
| 4.3.2 Vegetation pixel extraction and labelling . . . . .              | 77        |
| 4.4 State-of-the-art methods adapted to the Snapscan . . . . .         | 77        |
| 4.4.1 Illumination-based reflectance estimation . . . . .              | 78        |

|            |                                                                                   |            |
|------------|-----------------------------------------------------------------------------------|------------|
| 4.4.2      | Training-based reflectance estimation . . . . .                                   | 80         |
| <b>4.5</b> | <b>Estimated reflectance fidelity . . . . .</b>                                   | <b>83</b>  |
| 4.5.1      | Metrics . . . . .                                                                 | 83         |
| 4.5.2      | Results . . . . .                                                                 | 84         |
| 4.5.3      | Reflectance estimated at vegetation pixels . . . . .                              | 90         |
| 4.5.4      | Computation cost . . . . .                                                        | 92         |
| <b>4.6</b> | <b>Reflectance features for beet/weed detection and identification . . . . .</b>  | <b>93</b>  |
| 4.6.1      | Learning and test pixel extraction . . . . .                                      | 93         |
| 4.6.2      | Supervised classification . . . . .                                               | 94         |
| 4.6.3      | Supervised classification by boosting approach . . . . .                          | 97         |
| 4.6.4      | Evaluation metrics . . . . .                                                      | 98         |
| 4.6.5      | Classification experiments . . . . .                                              | 99         |
| <b>4.7</b> | <b>Conclusion . . . . .</b>                                                       | <b>105</b> |
| <b>5</b>   | <b>Dimension reduction . . . . .</b>                                              | <b>107</b> |
| 5.1        | Introduction . . . . .                                                            | 108        |
| 5.2        | Feature transform . . . . .                                                       | 108        |
| 5.2.1      | Data-driven methods . . . . .                                                     | 109        |
| 5.2.2      | SSF-based method . . . . .                                                        | 110        |
| 5.3        | <b><math>K</math>-dimensional vs. RGB-NIR reflectance feature space . . . . .</b> | <b>111</b> |
| 5.3.1      | From $K$ -dimensional to RGB-NIR feature space . . . . .                          | 112        |
| 5.3.2      | Learning and test pixel extraction . . . . .                                      | 113        |
| 5.3.3      | Classification protocol . . . . .                                                 | 113        |
| 5.3.4      | Results . . . . .                                                                 | 115        |
| 5.3.5      | NIR feature contribution . . . . .                                                | 117        |
| 5.4        | Feature selection . . . . .                                                       | 118        |
| 5.4.1      | State of the art . . . . .                                                        | 119        |
| 5.4.2      | Band selection . . . . .                                                          | 121        |
| 5.4.3      | Global band selection . . . . .                                                   | 124        |
| 5.4.4      | MSFA design . . . . .                                                             | 125        |
| 5.5        | Conclusion . . . . .                                                              | 127        |

|                                                         |            |
|---------------------------------------------------------|------------|
| <b>6 Raw texture features for crop/weed recognition</b> | <b>129</b> |
| 6.1 Introduction                                        | 130        |
| 6.2 Multispectral texture features                      | 130        |
| 6.2.1 Texture features from multispectral images        | 131        |
| 6.2.2 Handcrafted features                              | 132        |
| 6.2.3 Deep learning-based features                      | 133        |
| 6.2.4 MSFA texture features                             | 133        |
| 6.2.5 CNN-based raw texture features                    | 134        |
| 6.3 Texture feature for crop/weed classification        | 134        |
| 6.3.1 Patch extraction                                  | 135        |
| 6.3.2 Compared texture features                         | 136        |
| 6.3.3 CNN Training                                      | 139        |
| 6.4 Classification results and discussion               | 139        |
| 6.4.1 CA $4 \times 4$                                   | 140        |
| 6.4.2 CA $5 \times 5$                                   | 140        |
| 6.4.3 Discussion                                        | 142        |
| 6.5 Conclusion                                          | 144        |
| <b>Conclusion</b>                                       | <b>147</b> |
| <b>A GBM steps and LightGBM features</b>                | <b>153</b> |
| A.1 GBM steps for multiclass problem                    | 153        |
| A.2 LightGBM features                                   | 158        |
| A.2.1 Exclusive Feature Bundling (EFB)                  | 158        |
| A.2.2 Gradient-based One-Side Sampling (GOSS)           | 160        |
| <b>Bibliography</b>                                     | <b>163</b> |

# List of Abbreviations

| Abbreviations related to multispectral imaging/acquisition systems |                                                                                                |
|--------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| <b>VIS</b>                                                         | Visible (spectral) domain.                                                                     |
| <b>NIR</b>                                                         | Near infrared (spectral) domain.                                                               |
| <b>VIS-NIR</b>                                                     | Visible and Near infrared (spectral) domain.                                                   |
| <b>CIE</b>                                                         | Commission Internationale de l'Éclairage.                                                      |
| <b>SPD</b>                                                         | Spectral Power Distribution.                                                                   |
| <b>RSPD</b>                                                        | Relative <b>SPD</b> .                                                                          |
| <b>SSF</b>                                                         | Spectral Sensitivity Function.                                                                 |
| <b>FWHM</b>                                                        | Full Width at Half Maximum.                                                                    |
| <b>FSR</b>                                                         | Free Spectral Range.                                                                           |
| <b>FOV</b>                                                         | Field Of View.                                                                                 |
| <b>B, G, R</b>                                                     | Blue, Green, and Red channels.                                                                 |
| <b>RGB-NIR</b>                                                     | Image composed of <b>RGB</b> and <b>NIR</b> channels.                                          |
| <b>CFA</b>                                                         | Color Filter Array.                                                                            |
| <b>MSFA</b>                                                        | Multispectral Filter Array.                                                                    |
| <b>CA 4 × 4</b>                                                    | Designed <b>MSFA</b> with <b>16</b> bands in the <b>VIS</b> and <b>NIR</b> (spectral) domains. |
| <b>CA 5 × 5</b>                                                    | Designed <b>MSFA</b> with <b>25</b> bands in the <b>VIS</b> and <b>NIR</b> (spectral) domains. |
| <b>PPI</b>                                                         | Pseudo Panchromatic Image.                                                                     |
| <b>PPID</b>                                                        | <b>PPI</b> Difference.                                                                         |
| <b>WB</b>                                                          | Weighted Bilinear.                                                                             |
| <b>NDVI</b>                                                        | Normalized Difference Vegetation Index.                                                        |
| <b>UGV</b>                                                         | Unmanned Ground Vehicle.                                                                       |
| <b>UAV</b>                                                         | Unmanned Aerial Vehicle.                                                                       |



| <b>Abbreviations related to reflectance estimation</b>       |                                                     |
|--------------------------------------------------------------|-----------------------------------------------------|
| <b>MAE</b>                                                   | <b>Mean Absolute Error.</b>                         |
| <b>KLD</b>                                                   | <b>Kullback-Leibler Divergence.</b>                 |
| <b>rw</b>                                                    | <b>Row-wise method.</b>                             |
| <b>orw</b>                                                   | <b>Optimized rw method.</b>                         |
| <b>wa</b>                                                    | <b>White-average method.</b>                        |
| <b>ms</b>                                                    | <b>Max-spectral method.</b>                         |
| <b>wn</b>                                                    | <b>Wiener method.</b>                               |
| <b>dwd</b>                                                   | <b>Double White Diffuser method.</b>                |
| <b>Abbreviations related to supervised classification</b>    |                                                     |
| <b>QDA</b>                                                   | <b>Quadratic Discriminant Analysis.</b>             |
| <b>LDA</b>                                                   | <b>Linear Discriminant Analysis.</b>                |
| <b>SVM</b>                                                   | <b>Support Vector Machines.</b>                     |
| <b>K-SVM</b>                                                 | <b>Kernel SVM.</b>                                  |
| <b><math>k</math>-NN</b>                                     | <b><math>k</math>-Nearest Neighbors.</b>            |
| <b>RF</b>                                                    | <b>Random Forest.</b>                               |
| <b>MRF</b>                                                   | <b>Markov Random Field.</b>                         |
| <b>GBM</b>                                                   | <b>Gradient Boosting Machine.</b>                   |
| <b>LGBM</b>                                                  | <b>Light GBM.</b>                                   |
| <b>EFB</b>                                                   | <b>Exclusive Feature Bundling.</b>                  |
| <b>GOSS</b>                                                  | <b>Gradient-based One-Side Sampling.</b>            |
| <b>LBP</b>                                                   | <b>Local Binary Patterns.</b>                       |
| <b>M-LBP</b>                                                 | <b>MSFA-based LBP.</b>                              |
| <b>CNN</b>                                                   | <b>Convolutional Neural Network.</b>                |
| <b>SGD</b>                                                   | <b>Stochastic Gradient Descent.</b>                 |
| <b>Abbreviations related to spectral dimension reduction</b> |                                                     |
| <b>PCA</b>                                                   | <b>Principal Component Analysis.</b>                |
| <b>K-PCA</b>                                                 | <b>Kernel PCA.</b>                                  |
| <b>t-SNE</b>                                                 | <b>t-Distributed Stochastic Neighbor Embedding.</b> |
| <b>PLS-DA</b>                                                | <b>Partial Least Square Discriminant Analysis.</b>  |
| <b>mRMR</b>                                                  | <b>Minimum-Redundancy Maximum-Relevance.</b>        |

|             |                                       |
|-------------|---------------------------------------|
| <b>SHAP</b> | <b>SHapley Additive exPlanations.</b> |
| <b>SFS</b>  | <b>Sequential Forward Selection.</b>  |



# List of Symbols

| Symbols related to multispectral images          |                                                                                     |
|--------------------------------------------------|-------------------------------------------------------------------------------------|
| $\lambda$                                        | Wavelength.                                                                         |
| $E(\lambda)$                                     | Relative spectral power distribution of illumination.                               |
| $\Omega$                                         | Spectral domain.                                                                    |
| $b$                                              | Spectral band index.                                                                |
| $p$                                              | Pixel of a multispectral image.                                                     |
| $s$                                              | Surface element observed by a pixel.                                                |
| $T^b(\lambda)$                                   | Spectral sensitivity function of band $b$ .                                         |
| $A_p(\lambda)$                                   | Optical attenuation of the lens for pixel $p$ .                                     |
| $R_p(\lambda)$                                   | Spectral reflectance at $\lambda$ measured by the photosensor at $p$ .              |
| $Q(\cdot)$                                       | Quantization function.                                                              |
| $\lambda^b$                                      | Central wavelength of band $b$ .                                                    |
| Symbols related to multispectral image formation |                                                                                     |
| $q$                                              | Point in the image plane.                                                           |
| $x_q, y_q$                                       | Coordinates of point $q$ in the camera 2D coordinate system.                        |
| $t$                                              | Acquisition time.                                                                   |
| $E_t(\lambda)$                                   | RSPD of illumination associated to acquisition time $t$ .                           |
| $\mathcal{L}_{t,q}$                              | Incident light stimulus at acquisition time $t$ at point $q$ in the image plane.    |
| $t_q^b$                                          | Acquisition time at which the $b$ -th optical filter observes $\mathcal{L}_{t,q}$ . |
| $\mathcal{S}_{t,q}$                              | Radiance measured by the sensor at $q$ at acquisition time $t$ .                    |
| $f_t$                                            | Frame acquired at time $t$ .                                                        |
| $v$                                              | Pixel step between two successive frame acquisitions.                               |
| $\delta$                                         | Frame acquisition period.                                                           |
| $f_{dark}$                                       | Dark frame.                                                                         |

|                               |                                                                                                       |
|-------------------------------|-------------------------------------------------------------------------------------------------------|
| $f_t^b$                       | Frame stripe in $f_t$ that contains the spectral information associated to spectral band $b$ .        |
| $f_{t,p}$                     | Radiance value stored at pixel $p$ in frame $f_t$ .                                                   |
| $E_{t_p^b}(\lambda)$          | RSPD of illumination associated to the frame that records the measured radiance $\mathcal{S}_{t,q}$ . |
| $\mathbf{I}^{(B)}$            | $B$ -channel multispectral radiance image.                                                            |
| $\mathbf{I}^{(B)}[\text{WD}]$ | $B$ -channel multispectral radiance image of a white diffuser.                                        |
| $\mathbf{I}^{(B)}[\text{CC}]$ | $B$ -channel multispectral radiance image of a ColorChecker.                                          |
| $I^b$                         | Channel of index $b$ of a $B$ -channel multispectral radiance image.                                  |
| $I_p^b$                       | Intensity value at pixel $p$ associated to spectral channel $I^b$ .                                   |
| $I_p^b[\text{WD}]$            | Intensity value at pixel $p$ associated to spectral channel $I^b[\text{WD}]$ .                        |
| $I_p^b[\text{CC}]$            | Intensity value at pixel $p$ associated to spectral channel $I^b[\text{CC}]$ .                        |
| $x, y, b$                     | Axes of a multispectral image.                                                                        |
| $X, Y, B$                     | Size of a multispectral image, according to $x, y, b$ axes.                                           |
| $\tau$                        | Integration time used to acquire the object image $\mathbf{I}^{(B)}$ .                                |
| $\tau_{\text{wd}}$            | Integration time used to acquire the white diffuser image $\mathbf{I}^{(B)}[\text{WD}]$ .             |
| $\rho_{\text{wd}}$            | Diffuse reflectance coefficient of the white diffuser.                                                |

---

### Symbols related to spatial and spectral correction

---

|                              |                                                                                                                                                                         |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $\mathbf{C}^{(B)}$           | $B$ -channel multispectral image of correction factors.                                                                                                                 |
| $C_p^b$                      | Vignetting correction factor for pixel $p$ associated to channel $I^b$ .                                                                                                |
| $\mathcal{H}$                | $11 \times 11$ averaging filter.                                                                                                                                        |
| $\tilde{C}_p^b$              | Denoised vignetting correction factor for pixel $p$ associated to channel $I^b$ .                                                                                       |
| $\underline{I}^b[\text{WD}]$ | Median intensity value of the $m$ white diffuser pixels with the highest values at channel $I^b[\text{WD}]$ of the white diffuser image $\mathbf{I}^{(B)}[\text{WD}]$ . |
| $\tilde{\mathbf{I}}^{(B)}$   | Spatially corrected radiance image with $B$ channels.                                                                                                                   |
| $\tilde{I}^b$                | Spatially corrected radiance channel of index $b$ .                                                                                                                     |
| $\mathbf{M}$                 | Spectral correction matrix.                                                                                                                                             |
| $\tilde{\mathbf{I}}^{(K)}$   | Spatio-spectrally corrected radiance image with $K$ channels.                                                                                                           |
| $\tilde{I}_p^k$              | Intensity value at pixel $p$ associated to spatio-spectrally corrected channel $\tilde{I}^k$ .                                                                          |

| Symbols related to filter sensitivities and illumination |                                                                                              |
|----------------------------------------------------------|----------------------------------------------------------------------------------------------|
| $\delta\lambda^b$                                        | Full width at half maximum of band $b$ .                                                     |
| $\Delta\lambda$                                          | Free spectral range.                                                                         |
| $\mathcal{F}$                                            | Overall finesse (spectral resolving power) of the Fabry-Perot optical filters.               |
| $\overline{\mathbb{E}}$                                  | Average transmitted energy of the Fabry-Perot filters.                                       |
| $D_{KL}[\cdot]$                                          | Kullback-Leibler divergence.                                                                 |
| Symbols related to reflectance estimation                |                                                                                              |
| $\mathcal{WD}$                                           | White diffuser zone.                                                                         |
| $\mathcal{WS}$                                           | Square area of $\mathcal{WD}$ used by wa reflectance estimation method.                      |
| $\mathcal{WP}$                                           | ColorChecker white patch.                                                                    |
| $\hat{\mathbf{R}}_*^{(B)}$                               | Estimated $B$ -channel multispectral reflectance image by estimation method $*$ .            |
| $\hat{\mathbf{R}}_*^{(K)}$                               | $K$ -channel (spectrally corrected) multispectral reflectance image.                         |
| $\hat{R}_{*,p}^k$                                        | Estimated reflectance at pixel $p$ associated to channel $\hat{R}_*^k$ .                     |
| $\hat{\mathbf{R}}_{\text{ref}}^{(K)}$                    | Multispectral (reference) reflectance image estimated under controlled illumination.         |
| $\hat{\mathbf{R}}_{\text{ref}}^{(K)}[CC]$                | Multispectral reflectance image of the ColorChecker estimated under controlled illumination. |
| Symbols related to row-wise-based reflectance estimation |                                                                                              |
| $E_{t_{yp}^b}(\lambda)$                                  | Relative spectral power distribution of illumination associated to the row of pixel $p$ .    |
| $\tilde{I}_{\mathcal{WD},yp}^b$                          | Median value of the $m$ highest pixel values that represent $\mathcal{WD}$ in $y_p$ .        |
| $\mu_{\text{ref}}^k$                                     | Average reference reflectance values over all learning patches for channel $k$ .             |
| $\mu_{\text{rw}}^k$                                      | Average rw-based reflectance values over all learning patches for channel $k$ .              |
| $\hat{b}_0^k$                                            | Bias coefficient for rw-based reflectance channel $k$ .                                      |
| $\hat{b}_1^k$                                            | Scale coefficient for rw-based reflectance channel $k$ .                                     |
| $\hat{\mathbf{R}}_{\text{rw}}^{(K)}$                     | rw-based $K$ -channel reflectance image.                                                     |
| $\hat{\mathbf{R}}_{\text{orw}}^{(K)}$                    | orw-based $K$ -channel reflectance image.                                                    |
| Symbols related to reflectance estimation methods        |                                                                                              |
| $\tilde{\mathbf{R}}^{(B)}$                               | Coarse estimation of a $B$ -channel reflectance image.                                       |
| $\tilde{\mathbf{R}}'^{(B)}$                              | Row-wise rescaling of $\tilde{\mathbf{R}}^{(B)}$ image.                                      |

|                                                        |                                                                                                                                                              |
|--------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $\alpha_{y_p}^b$                                       | Illumination scaling factor computed at the row $y_p$ of $p$ .                                                                                               |
| $\bar{I}_{\mathcal{WD}, y_p}^b$ [WD]                   | Average value over the row of $p$ within the white diffuser subset $\mathcal{WD}$ in channel $I^b$ of the full-field white diffuser image.                   |
| $\bar{I}_{\mathcal{WD}, y_p}^b$                        | Average value over the row of $p$ within the white diffuser subset $\mathcal{WD}$ in channel $I^b$ of the scene image.                                       |
| $\beta_{\mathcal{WP}}^b$                               | Average value over the white patch subset $\mathcal{WP}$ in channel $\tilde{R}^b$ .                                                                          |
| $\rho_{\mathcal{WP}}^b$                                | Diffuse reflection factor of the white patch for the spectral band centered at $\lambda^b$ measured by a spectroradiometer in laboratory.                    |
| $\mathbf{T}_{\text{ref}}$                              | Reference reflectance vectors (from $\hat{\mathbf{R}}_{\text{ref}}^{(K)}[CC]$ ) of the learning patches horizontally stacked to form a $K \times 12$ matrix. |
| $\mathbf{T}_{\text{rad}}$                              | Radiance vectors (from $\hat{\mathbf{R}}_{\text{ref}}^{(K)}[CC]$ ) of the learning patches horizontally stacked to form a $K \times 12$ matrix.              |
| $\mathbf{G}$                                           | Estimation matrix of Wiener method.                                                                                                                          |
| $\mathcal{P}^l$                                        | Set of learning ColorChecker patches.                                                                                                                        |
| $\mathcal{P}_j^l$                                      | $j$ -th, $j = 1..12$ , learning patch of $\mathcal{P}^l$ .                                                                                                   |
| $\hat{\mathbf{R}}_{\text{ref}, \mathcal{P}_j^l}^{(K)}$ | $K$ -dimensional reference reflectance spectrum for learning patch $\mathcal{P}_j^l$ .                                                                       |

---

### Symbols related to reflectance estimation assessment

---

|                                                        |                                                                                       |
|--------------------------------------------------------|---------------------------------------------------------------------------------------|
| $\mathcal{P}^t$                                        | Set of test ColorChecker patches.                                                     |
| $\mathcal{P}_j^t$                                      | $j$ -th, $j = 1..12$ , test patch of $\mathcal{P}^t$ .                                |
| $\hat{\mathbf{R}}_{\text{ref}, \mathcal{P}_j^t}^{(K)}$ | $K$ -dimensional reference reflectance spectrum for testing patch $\mathcal{P}_j^t$ . |
| $\Delta\theta [\cdot]$                                 | Angular error.                                                                        |
| $\text{MAE}[\cdot]$                                    | Mean absolute error.                                                                  |
| $\overline{\text{MAE}}^k [\cdot]$                      | Channel-wise mean absolute error.                                                     |

---

### Symbols related to RGB-NIR simulation

---

|                                |                                                                                |
|--------------------------------|--------------------------------------------------------------------------------|
| $\mathbf{R}_{\text{rw}}^{(4)}$ | RGB-NIR (4-channel) image simulated from the $K$ -channel (rw-based) one.      |
| $T^l(\lambda)$                 | Normalized spectral sensibility function of the $l$ -th RGB-NIR camera filter. |

---

### Symbols related to supervised classification

---

|                              |                            |
|------------------------------|----------------------------|
| $\mathcal{S}^{\text{learn}}$ | Learning image/sample set. |
| $\mathcal{S}^{\text{test}}$  | Test image/sample set.     |
| $\mathcal{C}^i$              | $i$ -th class.             |

|                          |                                                                    |
|--------------------------|--------------------------------------------------------------------|
| $N_C$                    | Number of classes.                                                 |
| $\mathcal{N}^l$          | Number of learning pixels.                                         |
| $\mathcal{N}^t$          | Number of test pixels.                                             |
| $\mathcal{N}^{t_i}$      | Number of test pixels for class $\mathcal{C}^i$ .                  |
| $\Psi$                   | The quadratic discriminant function.                               |
| $\Sigma_{\mathcal{C}^i}$ | Covariance matrix of class $\mathcal{C}^i$ .                       |
| $y_s$                    | True label of a test sample $\mathbf{x}_s \in \mathbb{R}^K$ .      |
| $\hat{y}_s$              | Predicted label of a test sample $\mathbf{x}_s \in \mathbb{R}^K$ . |
| $D$                      | Number of selected features/bands.                                 |
| $\mathcal{B}^{(K')}$     | Reduced set of $K'$ ( $\ll K$ ) features.                          |
| $\mathcal{B}^{(D)}$      | Reduced set of $D$ ( $\ll K'$ ) features.                          |

---

### Symbols related to gradient boosting machine learning

---

|                             |                                                                                                |
|-----------------------------|------------------------------------------------------------------------------------------------|
| $M$                         | Number of boosting iterations.                                                                 |
| $\epsilon$                  | Learning rate.                                                                                 |
| $\gamma^i$                  | Per-class initialization value $\in [0, 1]$ .                                                  |
| $L$                         | Differentiable loss function for vectorial inputs.                                             |
| $\phi$                      | Differentiable loss function for scalar inputs.                                                |
| $\theta$                    | Weak learner (ensemble model).                                                                 |
| $\theta(\cdot)$             | Decision function (a classification logit).                                                    |
| $\hat{\theta}^i$            | Final weak learner specialized in class $\mathcal{C}^i$ .                                      |
| $\Theta$                    | Multiclass learner induced by $N_C$ ensemble models.                                           |
| $\Theta(\mathbf{x}_j)$      | $N_C$ -dimensional vector of classification logits provided by multiclass learner $\Theta$ .   |
| $\mathbf{y}_j$              | True label associated to sample $\mathbf{x}_j$ encoded as a one-hot $N_C$ -dimensional vector. |
| $\mathcal{P}(\mathbf{x}_j)$ | $N_C$ -dimensional probability vector associated to sample $\mathbf{x}_j$ .                    |
| $r_{j,m}^n$                 | Pseudo residual associated to sample $\mathbf{x}_j$ , iteration $m$ , and class $n$ .          |
| $T$                         | Number of terminal nodes.                                                                      |
| $\mathcal{R}_{t,m}^n$       | Terminal node associated to iteration $m$ and class $n$ .                                      |
| $\gamma_{t,m}^n$            | Output value associated to terminal node $\mathcal{R}_{t,m}^n$ .                               |





# Introduction

## Contents

---

|                                                    |           |
|----------------------------------------------------|-----------|
| <b>1.1 Precision farming</b>                       | <b>4</b>  |
| 1.1.1 Overview                                     | 4         |
| 1.1.2 Challenges                                   | 5         |
| <b>1.2 State of the art in crop/weed imaging</b>   | <b>6</b>  |
| 1.2.1 Remote sensing                               | 6         |
| 1.2.2 Proximal sensing                             | 7         |
| 1.2.3 State of the art about crop/weed recognition | 9         |
| 1.2.4 Limitation and challenges                    | 13        |
| <b>1.3 CA80 project</b>                            | <b>14</b> |
| 1.3.1 Aims and planning                            | 15        |
| 1.3.2 Examined crops and weeds                     | 16        |
| <b>1.4 Thesis contributions and structure</b>      | <b>19</b> |
| 1.4.1 Contributions                                | 19        |
| 1.4.2 Manuscript structure                         | 20        |

---

AGRICULTURE is one of the most essential areas that contribute to the economic growth of a country, and is the primary food source for a growing world population. To fulfill the increasing demand of population, crop quality and production quantity must be preserved. Unfortunately, crop fields are infested by weed plants that unintentionally grow alongside crops and compete with them in light and limited resources such as water and nutrients. When weeds become dominant, they may also cause shadows that alter the quality (*wavelength*) and amount (energy) of light reaching crops, leading to shade avoidance responses such as etiolation, leaf shortening or curling, and delayed flowering [69]. The ubiquitous nature of weeds affects crops and leads to crop yield reduction and crop quality alteration. According to the Food and Agriculture Organization, between 26% and 40% of the world's potential crop production is annually lost, and this loss could double without crop protection practices [133]. Moreover, crops are also subject to several diseases, such as *Cercospora* leaf spot that especially affects beet plants and alters their quality, leading to catastrophic loss in sucrose production when the disease is not handled rapidly. At an advanced stage, no cure is available, and the only treatment is to remove the affected parts of the plant so that the disease does not propagate to other plants.

In order to protect crops, a continuous inspection is required. Basically, farmers manually remove weeds and diseased plants in their fields. This task is laborious, time-consuming, and intractable in large crop fields. Therefore, to effectively control weeds and diseases in agricultural fields, herbicide and pesticide are sprayed by farmers (see Fig. 1.1). Uniform spraying does not require a detailed knowledge about weed or infected plant distribution in the field. It treats soil, crops, and weeds in the same way and with the same dose of herbicide/pesticide, regardless of its success in preventing diseases, controlling weeds and increasing crop productivity. In France, as elsewhere in the world, agrochemicals are sprayed in massive quantities (see Fig. 1.2) to get rid of plant diseases and weeds. The massive use of agrochemicals results in environmental pollution and soil degradation. It also impacts water quality, and can be harmful to human health [151] and various living organisms [174]. Moreover, weeds are not uniformly distributed within crop fields, and



FIGURE 1.1: Uniform spraying of a crop field.

FIGURE 1.2: Worldwide pesticide sale turnover<sup>2</sup> (green plot) and herbicide volume (in tonne) sold in France 2011–2019<sup>3</sup> (red plot).

agrochemicals can largely be saved if spraying is applied in a more localized manner [61]. In Europe, with the growing desire to tend towards a sustainable and less chemical-dependent agriculture, several regulations have emerged after the 2010 ENDURE international conference [150] in order to limit the usage of agrochemicals. Specifically, France projects to cut the use of pesticides by 50% for the next few years according to the Ecophyto II plan<sup>1</sup>. Therefore, considerable investments in *precision farming* are made to achieve this goal.

<sup>1</sup> <https://agriculture.gouv.fr/le-plan-ecophyto-quest-ce-que-cest>

<sup>2</sup> <https://fr.statista.com/infographie/11599/chiffre-affaires-pesticides-produits-phytosanitaires-dans-le-monde-et-par-region/>

<sup>3</sup> <https://www.statista.com/statistics/1068660/sales-herbicides-defanant-anti-foam-france/>

## 1.1 Precision farming

### 1.1.1 Overview

Precision farming, also known as site-specific management [142] is the application of advanced technologies and principles to manage spatial and temporal variability associated with all aspects of agricultural production [32]. Precision farming aims to increase yield production while reducing the negative impact of agrochemicals on the environment and ecosystems. Several agricultural applications, such as pasture systems [12], soil and yield monitoring [134, p. 143], and viticulture or horticulture management [195], are improved using precision farming. In the last decade, with the growing concern about food security and sustainable crop production, several precision farming applications try to deploy robots and sensors in fields to automatically identify diseased leaves or to separate crops from weeds for high-level interventions.

Precision spraying is one of the promising solutions being currently investigated to achieve this goal. It aims to exclusively spray weeds or diseased leaves/plants with an adequate herbicide dose according to weed density or disease stage. This approach is more efficient and less toxic than uniform spraying. It can achieve a drastical reduction of agrochemicals since it treats each plant individually.

Oberti et al. [132] conduct selective spraying experiments of *powdery mildew* disease on grapevine plants in a greenhouse. They use a robot equipped with a vision-based disease detection system and a pesticide sprayer. The system performs spraying whenever diseased spots are detected. As a result, 65% to 85% reduction of pesticide is achieved in comparison with conventional uniform spraying. Søgård and Lund [169] conduct indoor experiments about precision spraying of weeds. Their experiments show that, when selectively spraying weeds with micro-doses of a broad-spectrum systemic herbicide (glyphosate), the amount of herbicides can be reduced by two orders of magnitude compared to uniform spraying. Similarly, Utstumo et al. [179] perform indoor trials in a carrot field infested by four types of weeds (grass and three dicot species). They use a robot equipped by a drop-on-demand (DoD) system that deposits herbicide droplets on the detected weed leaves. They report that weeds can be effectively controlled using only 7.6  $\mu\text{g}$  of glyphosate and 0.15  $\mu\text{g}$

of iodosulfuron herbicide per plant. They also report that the herbicide amount can be reduced by more than 90% compared to classical spraying approaches.

In 2017, the French Ministries of Agriculture and of Ecological Transition, in partnership with the Ministry of Research and the French National Research Agency (ANR), launched the *Rose challenge*<sup>4</sup>. It is organized by the French National Research Institute for Agriculture, Food and Environment (INRAE), and the National Metrology and Testing Laboratory (LNE). The purpose of this challenge is to bring cooperation between academic and industrial research teams to develop projects for innovative technological solutions to reach the Ecophyto II objectives. Among the funded research projects in the Rose challenge is the WeedElec<sup>5</sup> project, that aims to overcome the automatic weed detection/identification problem using image processing. It combines aerial weed detection by a drone and a robot equipped with delta arms<sup>6</sup> fitted with a high-voltage electrical weeding tool. This project provides a public dataset<sup>7</sup> that contains color images of different crops and weeds as well as a weed detection approach based on convolutional neural networks (CNNs) [27].

According to the aforementioned studies, precision farming can indeed optimize the application of agrochemicals in crop fields. However, the practical adoption of precision farming techniques such as precision spraying still faces several challenges [21], foremost among which is probably the automatic localization of the areas to spray (or destroy).

### 1.1.2 Challenges

To efficiently perform selective spraying, the targeted areas (weeds or diseases) must be accurately located within the field. Moreover, in some cases, their type and density must be also determined in order to select the agrochemicals and the adequate spraying dose. This is not a trivial task since most weeds and crops share a common visual appearance. A given weed or crop plant is subject to several visual variations caused by external factors, such as variable illumination, water stress, or nutrients quality, which makes weed detection difficult. Moreover, color and texture of leaf surfaces also vary during their growing cycle, which further increases variability in

---

<sup>4</sup> <https://challenge-rose.fr/en/home/>

<sup>5</sup> <https://challenge-rose.fr/en/projet/weedelec2017-2/>

<sup>6</sup> <https://github.com/Agroecology-Lab/Open-Weeding-Delta>

<sup>7</sup> <https://zenodo.org/record/3906501>

plant visual appearance.

Non-imaging sensors, such as spectroradiometers, provide local spectral measurements of target surfaces in specific spectral *bands*. On the other hand, imaging sensors (typically, cameras) provide spatial (shape/texture) and spectral (color/spectra) measurements that allow for a global interpretation of the scene, hence its visual inspection. Such sensors are widely used in precision farming [176]. They can be used for the analysis of crop yield and biomass, water stress, soil properties, moisture and clay content, disease detection and crop/weed recognition.

In this study, we only focus on the crop/weed recognition problem by analysis of their visual aspect.

## 1.2 State of the art in crop/weed imaging

Two types of approaches have been proposed to address the crop/weed recognition problem, namely *remote* and *proximal* sensing.

### 1.2.1 Remote sensing

Early works on crop/weed mapping are based on remote vegetation sensing, which usually relies on high-altitude platforms, such as satellites and aircrafts, to acquire images of vegetation [9, 94, 96, 140]. Satellites (e.g., the SPOT, AVHRR, AVIRIS, ...) are equipped with multispectral sensors and provide images with high spectral resolution but coarse spatial resolution. For instance, the AVIRIS sensor provides images composed of 224 *channels* associated to spectral bands in the range 400–2500 nm, where each pixel covers an area of 20 m<sup>2</sup> on the ground. It is difficult to determine the optimal spatial resolution for weed mapping applications because it depends on various factors, such as growth stage, height, and density of vegetation. Despite outstanding improvements in aerial and satellite imaging, especially regarding the spatial resolution (finer than 1 m<sup>2</sup>/pixel with GeoEye and WorldView satellites), images may still not carry enough textural information to analyze vegetated surfaces precisely. A higher spatial resolution remains necessary to detect or identify weeds from crops.

Aircrafts that embed multispectral sensors can provide images with spatial resolution between 0.25 and 4 m<sup>2</sup>/pixel, depending on the scene-to-sensor distance. As for

satellite imaging, aerial imaging for crop/weed detection is also subject to several barriers, such as: tedious (or expensive) image acquisition, atmospheric perturbations, etc. [129]. In addition, it mainly relies on the available spectral signature at each pixel. Indeed, pixels of images acquired from a high-altitude platform observe large surface elements of the field. Thus, a single pixel may be associated to different scene surfaces. This is more likely to occur in case of overlapping vegetation. Spectral signatures are then mixed and spectral unmixing of surface elements is performed in order to retrieve the correct spectral signatures [20]. High-altitude aerial imaging systems are mostly adapted to observe fields where crops and weeds are grouped in large and dense patches and when identifying weed species is not required. Weed can be then detected if it has specific growth spectral signatures. In order to improve spatial resolution, proximal sensing approaches should be considered.

### 1.2.2 Proximal sensing

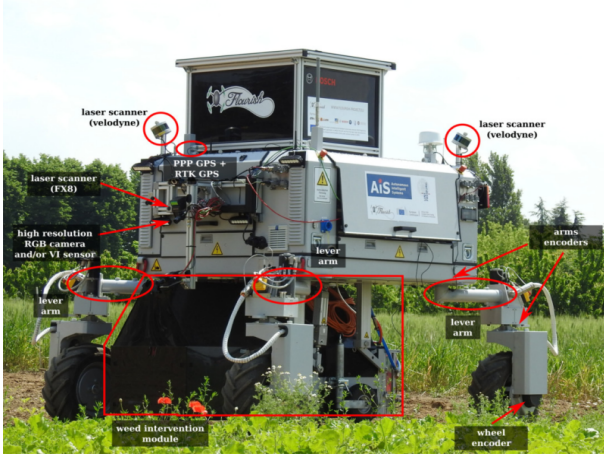
Proximal sensing of crops and weeds involves sensors (spectrometers) or cameras mounted on tractors, unmanned ground vehicles (UGVs), or low-altitude unmanned aerial vehicles (UAVs). UGVs such as the Thorvald II [64], BoniRob [145], AgeBot II [16], Ladybird [184], or the Avo robot from Ecorobotix<sup>8</sup>, can be used for both image acquisition and in-field interventions. They can embed different kinds of tools (such as selective sprayers, mechanical weeding, cameras, lasers, etc.) to provide a per-plant monitoring (see Fig. 1.3(a, b)). However, because the field of view (FOV) of UGVs is usually restricted to a small area of the scene, monitoring large crop fields may be relatively slow. They are also more expensive than UAVs, that are relatively cheap and easier to deploy in outdoor [101]. UAVs are used to monitor crop fields at larger scales (see Fig. 1.3(c)) since they can cover large areas in a shorter time in contrast to UGVs. However, because UAVs are more subject to vibrations and stability problems, not all cameras can be embedded on them, and the acquired images may undergo some corrections (such as image alignment in case of multi-sensor based cameras).

Cameras originally deployed in agricultural fields are either monochromatic or color [130].

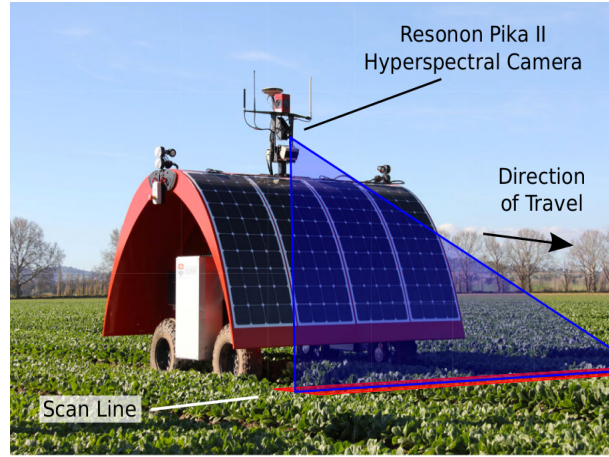
---

<sup>8</sup> <https://www.ecorobotix.com/en/avo-autonomous-robot-weeder/>





(a) Bosch BoniRob farming UGV [145].



(b) Ladybird UGV [184].

(c) DJI Phantom 4 UAV<sup>9</sup> [109].

FIGURE 1.3: Two UGV systems (a, b) and a UAV (c) used for outdoor crop/weed monitoring. The BoniRob robot (a) uses RGB and RGB-NIR cameras for weed detection and tracking. The Ladybird robot uses a multispectral linescan camera for weed detection. (c) The DJI Phantom 4 quad-copter (standard version) uses an RGB camera for image acquisition.

Color cameras are usually equipped with three different optical filters that are sensitive to the visible (VIS) domain of light. Thanks to one shot, they capture light along the red (R), green (G), and blue (B) spectral bands of the VIS domain, and provide color images composed of three RGB channels. RGB-NIR cameras provide information in the near-infrared (NIR) domain that is combined with color information to compute vegetation indices, such as the normalized difference vegetation index (NDVI) [15, 43, 48, 76]. However, spectral information is averaged over wide wavelength ranges in both color and RGB-NIR cameras, resulting in a lack of detail in specific narrow spectral bands. Combining imaging and spectrometry has led to a

<sup>9</sup> <https://www.dji.com/fr/phantom-4>

new generation of optical sensing technologies generically referred to as *multispectral imaging*.

Multispectral cameras sample the scene *radiance* (amount of light reflected by an object in a given direction) according to many spectral bands of the VIS and NIR domains, and provide a multispectral radiance image that is composed of multiple channels. Each channel contains the scene spectral information associated to a specific narrow spectral band. From this image, features are extracted by a specific descriptor, and analyzed by a supervised classifier to discriminate crops and weeds. For feature extraction, we can distinguish *handcrafted* and *deep learning*-based approaches.

### 1.2.3 State of the art about crop/weed recognition

#### 1.2.3.1 Handcrafted features for crop/weed recognition

One of the earliest precision spraying systems was proposed in 1999 by Lee et al. [97]. The authors describe a robot equipped with an RGB camera that acquires images of vegetation under artificial illumination (halogen lamps). In order to discriminate tomato plants from weeds, they compute shape features (area, elongation, compactness, length/perimeter ratio, etc.) that are analyzed by a Bayesian classifier for decision. In field conditions, this system is able to detect about 76% of the tomato plants, whereas only 48% of the weeds are detected. Haug et al. [72] use a two-band camera mounted on an autonomous field robot equipped with artificial lighting to acquire images in the visible red and NIR domains. In each acquired radiance image, the NDVI is computed at each pixel. The authors propose a keypoint classification approach and analyze only a small set of pixels located at sparse spatial coordinates in each NDVI image. At these pixels, they compute shape (perimeter, area, compactness, solidity, convexity, etc.) and statistical features (mean, median, standard deviation, etc.) that are analyzed by a random forest (RF) classifier. To refine and recover the final pixel classification map, they use Markov random field (MRF) smoothing and nearest neighbor interpolation. Lottes et al. [107] use a camera that samples four broad spectral bands to acquire RGB-NIR images under controlled illumination (halogen spots), and compute (illumination-dependent) shape and spectral features of vegetation. Louargant et al. [111] use a multispectral camera

that samples four key spectral bands for NDVI computation. Their weed detection method combines prior knowledge about the spatial distribution of plants in the field (i.e., between-row pixels are considered as weeds and within-row pixels as crops) and spectral information. *Reflectance* (intrinsic optical property of materials) is estimated from the multispectral radiance images using a reference panel (device that reflects all incident light) that measures the downwelling illumination during radiance image acquisition. Learning crop and weed pixels are first selected thanks to the between/within-row assumption. Then, a supervised classifier is trained with reflectance features estimated at the selected learning pixels. Bosilj et al. [22] propose the *attribute profiles* descriptor to extract texture features over a  $72 \times 72$  pixel grid. They use a RF classifier and compare the discrimination power of attribute profiles features against local binary patterns (LBP) and histogram of oriented gradients (HOG) features. They show that attribute profiles provide the best crop and weed classification performance on RGB-NIR images of the Sugar Beets 2016<sup>10</sup> and Carrots 2017<sup>11</sup> datasets.

### 1.2.3.2 Deep learning for crop/weed recognition

Alternatively to the *handcrafted* feature extraction approach, several authors propose to use convolutional neural networks (CNNs) for crop/weed recognition [110, 124, 125, 153, 154]. CNNs can learn complex features describing non-linear relationships between the different classes. Potena et al. [144] propose a shallow CNN and a deep one to analyze RGB-NIR radiance images. They first separate vegetation from background thanks to a thresholding of the NDVI. They apply the trained shallow CNN to a small patch centered at each vegetation pixel to refine vegetation detection. Then, vegetation pixels are classified as crops or weeds by the deep CNN that analyzes larger patches. In [124], a CNN composed of 8 layers is used to analyze RGB-NIR images for weed detection. First, vegetation pixels are isolated using the NDVI. Then, vegetation blobs (subsets of connected vegetation pixels) are built as patches (with different spatial resolution according to blob shape and vegetation density),

<sup>10</sup> <https://www.ipb.uni-bonn.de/data/sugarbeets2016/>

<sup>11</sup> <https://lcas.lincoln.ac.uk/wp/research/data-sets-software/crop-vs-weed-discrimination-dataset/>

and rescaled so that each patch fits in a  $64 \times 64$  pixel window. Blob-wise classification is fast since it learns from vegetation blobs (patches) and classifies each blob by considering it as a small (rescaled) patch. It however assumes that vegetation plants do not overlap. All pixels in a vegetation blob are associated to the same class, otherwise, blob-wise classification may provide inaccurate results. In [153], a downward-facing RGB-NIR camera is mounted on a UAV to acquire radiance images of vegetation at 2 m of altitude. In order to learn complex features and to achieve semantic detection of crop (sugar beet) and weeds, the authors use an encoder-decoder CNN model based on the state-of-the-art *SegNet* framework [14]. The latter is trained with different combinations of channels (panchromatic, Red and NIR, Red and NIR and NDVI) and reaches a weed detection performance of 80% in terms of F1-score. Despite of the good results obtained by CNN-based crop/weed detection approaches, they also show decreasing performances when the feature distribution of plants in the test images changes due to growth stage and when test images come from different fields. To generalize these models, data augmentation [11, 125] and prior information about weed distribution in the field [106, 111] may be considered.

### 1.2.3.3 Multispectral image analysis for crop/weed recognition

Because some weeds are visually very similar to crops (e.g., in color and texture), such as bean and datura (see Fig. 1.4(a)), or beet and goosefoot (see Fig. 1.4(b)), their detection from color image analysis only may be difficult, especially in field conditions where illumination varies and plants shade each other. As crops and weeds also often overlap, it is difficult to compute shape features. Therefore, some researchers use multispectral devices with finer spectral resolution. In [49], a multishot multispectral device (composed of an imaging spectrograph and a camera) is used to acquire radiance images of vegetation with a spectral resolution of 35 nanometers (nm) in the range 435–1000 nm. In order to bypass illumination conditions, spectral reflectance is estimated thanks to a reference panel included in the scene before each image acquisition. The latter is then used as a feature to train different classifiers (k-nearest neighbour (k-NN), minimal distance, classification and regression tree (CART), multi-layer neural network with non-linear mapping (MLNLM)). The best performance is achieved by the MLNLM classifier with an accuracy of 80.1%





FIGURE 1.4: Two crop/weed RGB images: bean/datura (a) and beet/goosefoot (b). Red dots show the locations of weed plants that can hardly be distinguished from their respective crop.

for sugar beet crop and of 91.4% for weed class. In [168], a multishot (linescan) multispectral camera (integrating a spectrograph and a matrix detector) that acquires radiance images with a spectral resolution of 10 nm between 360 nm and 1010 nm is used to detect weeds in soybean fields. The radiance spectrum measured at each pixel is normalized so that the minimum radiance value is zero and the average value is one. The normalized radiance may be transformed by principal component analysis (PCA). Either the best fifteen features selected (by step-wise feature selection based on Wilks' lambda criterion) from normalized radiance or the best seven features selected from normalized-transformed radiance are used to train the linear discriminant analysis (LDA) classifier and a neural network (NN). The PCA-NN pipeline (NN classifier trained with normalized-transformed radiance) provides the best results with a classification accuracy between 90.2% and 99.1% according to the weed type.

Wendel and Underwood [184] use a multishot (linescan) multispectral camera mounted on a UGV to acquire radiance images with 244 channels in the VIS and NIR domains (between 390 and 887.4 nm). Vegetation pixels are extracted by thresholding NDVI images. Radiance is normalized as in [168], then smoothed by *Savitzky-Golay* filtering [158]. Then, PCA is performed on the training data for feature extraction and the 20 most explanatory features are retained. Pixels are classified thanks to support

vector machine (SVM) and LDA classifiers. In order to learn additional (new) spectral signatures and to adapt the classifiers to changing vegetation appearance without manually generating training data, the system relies on the between/within-row assumption about crop and weed distribution. The two classifiers provide similar results and achieve an overall crop/weed detection performance between 85% and 88% in terms of F1-score.

Despite that multishot multispectral cameras acquire high spectral resolution images, it is still difficult to use them in outdoor conditions. As these cameras build the multispectral image from several shots, they are subject to different perturbations that may occur during image acquisition, such as illumination variation and wind, which may alter the quality of the spectral signatures.

#### 1.2.4 Limitation and challenges

In order to distinguish between crops and weeds, state-of-the-art crop/weed recognition systems rely on the following approaches:

- **Shape features:** Several studies extract shape features in order to discriminate weeds from crops [107, 182, 185]. They can be efficient at early growth stages when leaves of crop and weed plants do not interfere. However, at an advanced growth stage, when crop and weed leaves occlude each other, shape features become irrelevant.
- **Between/within-row assumption:** To increase classification performances of the classifier, one often assumes within-row plants as crops and between-row plants as weeds [71, 76, 107, 109]. This assumption is however not always verified because weeds can grow within crop rows (see Fig. 1.4). Moreover, when the camera is mounted on a UGV, the FOV is much more bounded than with a UAV and crop rows may not be well detected.
- **UGV with controlled illumination:** Most UGV-based systems acquire vegetation radiance images under artificial light to avoid the naturally varying illumination of skylight [16, 72, 76, 106]. This makes them more cumbersome and dependent on a particular illumination.

- **UAV under skylight:** In the case of UAV-based systems, images are acquired under skylight. Therefore, in order to be robust against illumination conditions, a large training dataset of radiance images needs to be acquired under various illuminations. Moreover, labeling large datasets is very tedious and expensive (because it requires experts in agronomy) [4]. Furthermore, learning from high-dimensional datasets is problematic and requires high computational power.
- **Deep learning:** The acquired radiance images can be fed into a deep CNN for feature extraction and pixel classification. Encoder-decoder CNN models, such as SegNet [14], can take the whole image as input and provide an output as a pixel-wise classification map [154]. However, there is no guarantee that the model will be robust against different kinds of illuminations. Moreover, training such models with high spatial and spectral resolution radiance images requires a lot of memory and high computational power.
- **Multispectral imaging:** Multispectral linescan cameras have been used for weed recognition [168, 184] because they acquire high spectral resolution images. However, they build a multispectral image from successive *frames* acquired at different times, so, they have not been widely adopted in precision farming. Hence, using them in field conditions is still a challenge because image acquisition is sensitive to illumination variation. Moreover, feature extraction from multispectral images is problematic because of their dimensionality.

The CA80 project has emerged to address these challenges, and develop an automatic crop/weed recognition system based on multispectral imaging.

### 1.3 CA80 project

Farmers are waiting for solutions that would enable them to take further steps towards the reduction of herbicides, and to achieve the objectives of Ecophyto II plan. Mechanical weeding, the main alternative to agrochemicals, has been progressively adopted for beets, sunflowers and corns. However, it remains limited in its ability to treat large crop fields. To investigate solutions that improve weed control applications and reach the objectives of Ecophyto II plan, the Chambre d'Agriculture de

la Somme<sup>12</sup> (CA80) initiated the project at the end of 2016, co-financed by the CA80, the French government, and the Hauts-de-France region. They focus on proximal sensing of weeds for agro-ecological weeding. Their ambition is to reduce the use of herbicides in field crops and vegetable production by more than 80% while maintaining a high rate of weed eradication.

### 1.3.1 Aims and planning

The CA80 project is based on a multidisciplinary approach (imaging, information technology, agronomy, machinery) that associates the CRISAL laboratory of the University of Lille and the CA80. The research works carried out in this thesis since October 2018 result from the collaboration between the Color Imaging Team<sup>13</sup> of CRISAL and engineers of the CA80.

Based on field experiments, the CA80 project aims to develop a process ranging from the location and qualification of weeds, to the application of the most sustainable weeding possible. Its main tasks are:

- **Protocol for outdoor image acquisition:** Outdoor vegetation image acquisition by multishot multispectral devices is subject to several challenges, such as variable illumination or shadows. Therefore, an adequate image acquisition protocol should be established.
- **Acquisition of a crop/weed multispectral radiance image dataset:** Once images have been collected according to the established protocol, they should be stored and organized following an efficient management. The CA80 has already developed a graphical user interface for an efficient archiving of the dataset thanks to several parameters such as acquisition date, plant species, growth stage, hydric or nutritional stresses, disease type, etc. An expert in agronomy should also analyze the images to build the ground truth (pixel labels) associated to each of them.
- **Creation of a database of vegetation spectral signatures:** The radiance images in the dataset should be analyzed to estimate reflectance. Region of interests

---

<sup>12</sup> <https://hautsdefrance.chambre-agriculture.fr/chambre-agriculture-somme/>

<sup>13</sup> <https://color.univ-lille.fr/>



(ROIs) are defined based on the vegetation ground truth to extract reflectance spectra of the plants as their spectral signatures.

- **Supervised crop/weed recognition:** This task is to design an illumination-robust crop/weed classification system thanks to the analysis of the extracted reflectance signatures and/or of extra texture features.
- **Spectral band selection for crop/weed recognition:** To reduce the dimension of the multispectral images, and to specify a low-cost sensor dedicated to outdoor crop/weed recognition, the relevant spectral bands for crop/weed classification should be selected.

### 1.3.2 Examined crops and weeds

The Hauts-de-France region is one of the most important agricultural regions in France, with about 67% of its area dedicated to agriculture. It is a leading producer of different crops such as wheat, beet, chicory, bean, and peas<sup>14</sup>.

The Chambre d'Agriculture de la Somme has an experimental greenhouse in parcels (see Fig. 1.5(a)) located near Amiens, France, where they carried out an image acquisition campaign of vegetation under skylight in April 2017.



(a) Parcel with two aisles of cultivation.

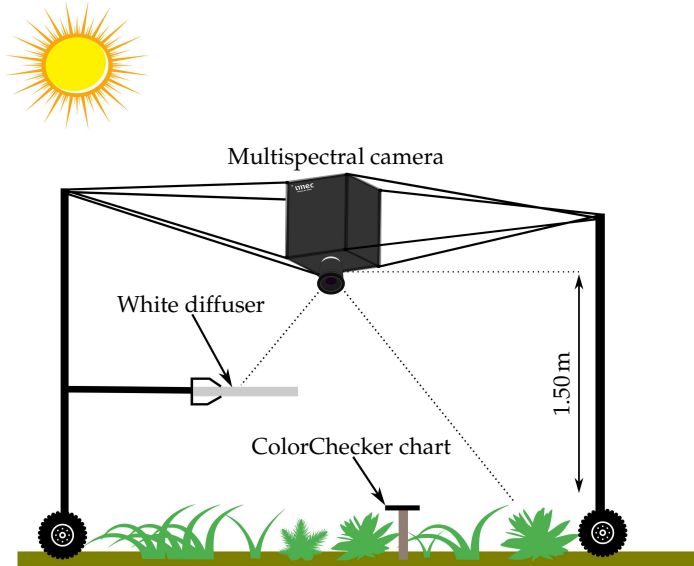


(b) Experimental greenhouse over the parcel shown in (a).

FIGURE 1.5: The parcel (a) and greenhouse (b) used for plant cultivation and image acquisition experiments.

<sup>14</sup> [https://draaf.hauts-de-france.agriculture.gouv.fr/IMG/pdf/memento\\_hauts-de-france-edition2020\\_2-11-2020\\_cle0b35c6-1.pdf](https://draaf.hauts-de-france.agriculture.gouv.fr/IMG/pdf/memento_hauts-de-france-edition2020_2-11-2020_cle0b35c6-1.pdf)

The experimental greenhouse is organized into several micro-plots allowing the establishment of rooted crops. It contains a custom-designed guided moving frame (see Fig. 1.5(b)) to be equipped with a camera and or a spectroradiometer. The CA80 Project considers several crops that are representative and somewhat specific to the Hauts-de-France region such as sugar beet, green bean, wheat, barley, canola, and peas. In this study, we focus on three crop types that represent one of the main crop productions of the region, namely (sugar) beet (*Beta vulgaris*), wheat (*Triticum* sp.), and (green) bean (*Phaseolus vulgaris*). These plants are implanted in greenhouse under skylight along with three weed species that commonly infest them, namely thistle (*Cirsium arvense*), goosefoot (*Chenopodium* sp.), and datura (*Datura stramonium*). We assess the discrimination of these species using multispectral imaging. To acquire outdoor crop/weed images, an acquisition system based on a linescan multispectral camera has been designed (see Fig. 1.6(a)). The camera is mounted on top of the acquisition system (about 1.5 m from the ground) and observes vegetation from nadir (see Fig. 1.6(b)). It also observes a white diffuser for reflectance estimation and a ColorChecker chart for estimated reflectance assessment.



(a) Acquisition setup.



(b) Scene observed from nadir (RGB rendering of an acquired multispectral image).

FIGURE 1.6: Multispectral image acquisition.

A second main image acquisition campaign has been conducted in 2019 in order to acquire outdoor radiance images of vegetation. These images are used in this study



as a support to investigate solutions for weed recognition. Figure 1.7 shows 9 multi-spectral images that illustrate the considered plant species, rendered as RGB images with D65 illuminant.

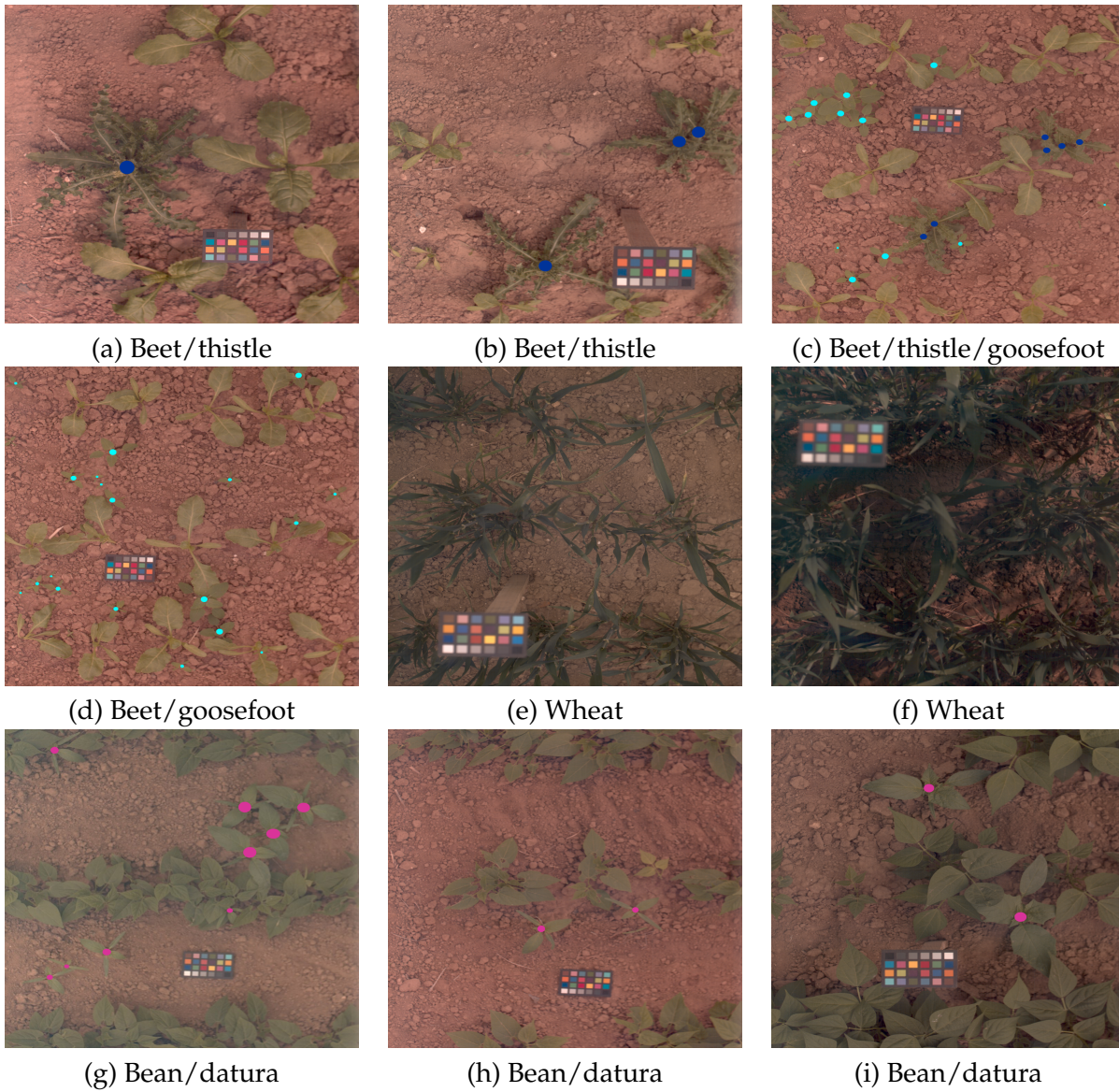


FIGURE 1.7: Samples of the plant species (at various growth stages) considered in this study. Dots show weed plant: thistle in blue (a, b, c), goosefoot in cyan (c, d), and datura in magenta (g, h, i). Other plants are crop: beet (a, b, c, d), wheat (e, f), and bean (g, h, i). As we can see some crop/weed pairs such as beet/goosefoot and bean/datura are barely distinguishable because they have similar colors and textures. Although textures and shapes of thistle and wheat leaves are different, they have a slightly darker green color than beet and bean ones.

## 1.4 Thesis contributions and structure

### 1.4.1 Contributions

The main objective of this work is to develop an automatic recognition system of the considered crop and weed plants in field conditions. Specifically, this thesis investigates the deployment of a multispectral camera in outdoor conditions. The ultimate goal is to design illumination-robust classification methods and to specify a dedicated sensor for the discrimination of the considered plants. The main contributions are:

- **Multispectral image database acquisition:** Because there is no publicly available dataset that contains high spectral resolution images associated to the considered vegetation species, we build a new database from the acquired vegetation images and annotate it with the segmentation ground truth. The targeted species are beet, wheat, and green bean for crops, and thistle, goosefoot, and datura for weeds.
- **Multispectral image formation model:** Radiance image acquisition by a linescan camera follows a specific process that is influenced by variable illumination conditions. Thus, the classical image formation model is not adapted to model the acquisition process. Therefore, we propose an original image formation model for multispectral image acquisition by a linescan camera [7].
- **Reflectance estimation:** To compute features that are invariant to illumination conditions and tend towards specific spectral signatures of vegetation, we propose to compute the reflectance at each pixel. However, because radiance acquired by linescan cameras is sensitive to variable illumination, state-of-the-art reflectance estimation methods are not suited to images acquired with this type of camera. From our image formation model, we propose a method to estimate reflectance [6, 7].
- **Supervised crop/weed recognition based on reflectance features:** A supervised classifier is trained using estimated reflectance features. To assess their discrimination power, they are compared against reflectance features estimated by state-of-the-art methods [7, 8].

- **Band selection:** A supervised feature selection approach is applied to reduce the dimension of our images. It selects the most discriminant spectral bands for crop/weed recognition.
- **Spatio-spectral texture feature extraction:** We propose an original method based on a CNN to extract texture features directly from low dimensional multispectral images. These features are fed into a supervised classifier for training and classification.

### 1.4.2 Manuscript structure

The rest of the manuscript is organized as follows.

In Chapter 2 we describe the formation of multispectral radiance images under the Lambertian surface assumption and the different devices that can be used to acquire such images. We then provide a detailed description of the multispectral camera used in this study. Because radiance multispectral images are acquired under varying illumination, we propose an original multispectral image formation model that takes the variation of illumination conditions into account.

In chapter 3, we estimate the reflectance as an illumination-invariant spectral signature. First, we present state-of-the-art methods that can be used to estimate the reflectance from multispectral images. We then introduce the reference state-of-the-art method for reflectance estimation and describe our proposed method for reflectance estimation under varying illumination.

Chapter 4 focuses on estimated reflectance assessment. The quality of reflectance estimated by our method is evaluated against state-of-the-art methods, and its contribution to supervised crop/weed recognition is demonstrated.

Chapter 5 addresses the dimension reduction issue. The acquired multispectral images are composed of a high number of spectral channels, whose analysis is memory- and time-consuming. Moreover, spectral bands associated to these channels may be redundant or contain highly correlated spectral information. Therefore, we select the best spectral bands for crop/weed classification and use them to specify a camera suited for crop/weed recognition.

---

Chapter 6 deals with the problem of spatio-spectral feature extraction from multispectral images. We propose an approach that extracts both spatial and spectral information at reduced computation costs based on a CNN. Its contribution to crop/weed recognition is demonstrated.



# Multispectral imaging

## Contents

---

|                                                                        |           |
|------------------------------------------------------------------------|-----------|
| <b>2.1 Introduction</b>                                                | <b>24</b> |
| <b>2.2 Illumination and surfaces</b>                                   | <b>25</b> |
| 2.2.1 Bands and wavelengths                                            | 25        |
| 2.2.2 Radiance and reflectance                                         | 27        |
| <b>2.3 Multispectral radiance image acquisition devices</b>            | <b>27</b> |
| 2.3.1 Multispectral radiance image                                     | 28        |
| 2.3.2 Snapshot devices                                                 | 29        |
| 2.3.3 Multishot devices                                                | 30        |
| <b>2.4 Multispectral radiance image acquisition by Snapscan camera</b> | <b>33</b> |
| 2.4.1 Snapscan camera                                                  | 33        |
| 2.4.2 Radiance measurement                                             | 35        |
| 2.4.3 Frame acquisition                                                | 36        |
| 2.4.4 Stripe assembly                                                  | 38        |
| 2.4.5 Image formation model for Snapscan camera                        | 39        |
| <b>2.5 Conclusion</b>                                                  | <b>40</b> |

---



## 2.1 Introduction

THE trichromatic approach has been dominating for a long time in color representation to mimic the human color vision. With the development of technology, merging conventional imaging techniques with spectrophotometers properties is possible through multispectral imaging sensors that acquire the radiance of a scene with a high spectral and spatial resolution. The first applications of this technology were dedicated to satellite remote sensing for environmental monitoring, geological research, or mineral mapping. Increasing the number of spectral bands to more than three is an active research topic with a great potential for a plethora of applications, such as quality assessment of agro-food products [30, 46, 80], art studies [187], medical imaging [135], or texture and material classification [79, 121]. During the last decade, multispectral imaging has gained much attention and is currently a very promising technology being investigated in precision farming applications like pest and disease detection [114], plant classification [72], and weed detection [99, 184].

Section 2.2 presents the basics of multispectral imaging and how radiance results from the interaction between the spectral properties of illumination and objects. Section 2.3 introduces the main devices that can be used to acquire multispectral radiance images. To acquire outdoor multispectral radiance images of plant parcels in a greenhouse under skylight, we use a multishot camera called the “Snapscan” [141] that is presented in Sec. 2.4.1. To build a multispectral image, this camera acquires several frames. We detail how it measures radiance (see Sec. 2.4.2) and formalize frame acquisition by the Snapscan (see Sec. 2.4.3). We explain how the multispectral radiance image is obtained from frame stripes in Sec. 2.4.4, and we propose a specific model of multispectral image formation in Sec. 2.4.5. This model shows that when outdoor illumination varies during frame acquisitions, the spatio-spectral correlation assumption does not hold.

## 2.2 Illumination and surfaces

We first introduce the notions of spectral band and illumination in Sec. 2.2.1. Then, we specify the key notions of radiance and reflectance, and explain how radiance is related to illumination and reflectance in Sec. 2.2.2.

### 2.2.1 Bands and wavelengths

The electromagnetic spectrum of light is composed of different waves, each being identified by its wavelength. Figure 2.1 shows the VIS and NIR domains in the electromagnetic spectrum. A spectral sensing instrument collects light energy within specific domains of the electromagnetic spectrum.

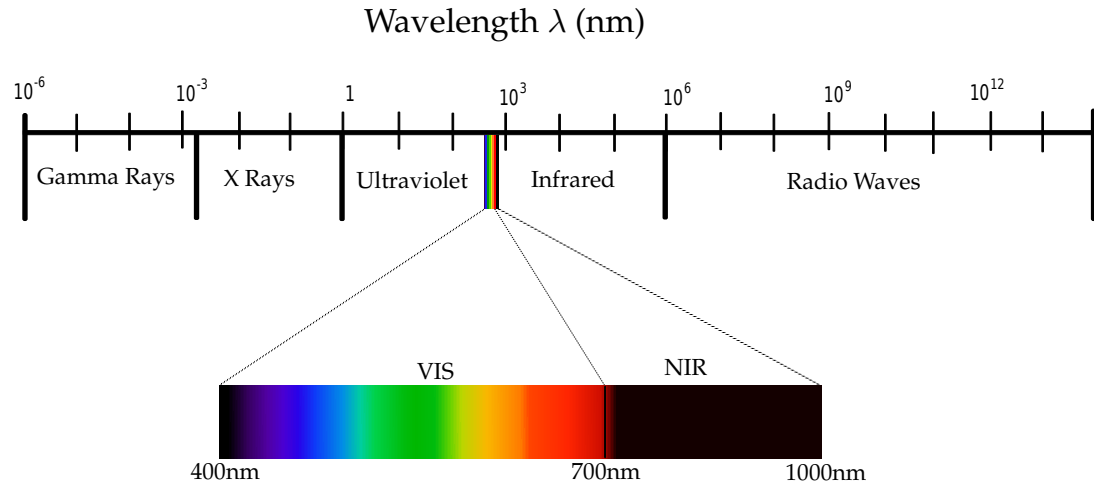


FIGURE 2.1: Visible and near infrared domains in the electromagnetic spectrum.

Each domain is associated to a band-pass filter simply referred to as a *band*, that is embedded in the camera. For example, the wavelengths between 400 nm and 450 nm might correspond to one (spectral) band whose *bandwidth* is  $\approx 50$  nm. The bandwidth of a band depends on the spectral sensitivity function (SSF) of the optical filter. Specifically, it is the wavelength range through which light passes the filter, defined by the full width at half maximum (FWHM) of the filter SSF.

The scene is enlightened by a light source called illumination, that is characterized by different electromagnetic waves spreading at light speed. The spectral power distribution (SPD) (in  $W \cdot sr^{-1} \cdot m^{-2}$ ) of a light source is the amount of photons emitted by unit surface per unit solid angle with respect to the wavelength. Light sources are often characterized by their relative SPD (RSPD)  $E(\lambda)$  (with no unit), obtained

as their SPD in a certain spectral domain  $\Omega$  normalized by that of a reference wavelength (the wavelength where SPD is maximal for example, as in Fig. 2.2). Several light sources called *illuminants* have been normalized by the International Commission on Illumination (Commission Internationale de l'Éclairage, CIE) [62]. In 1931, the CIE defined three standard illuminants (A, B, and C). The RSPD of illuminant A theoretically represents an incandescent tungsten filament lamp whose temperature is about 2856 K. Illuminants B and C represent direct sunlight and average daylight with a temperature of approximately 4900 K and 6800 K. In 1964, the CIE recommended a new set of illuminants (prefixed by the letter D) to represent additional variations of daylight throughout the VIS and ultraviolet (UV) domains (as far as 300 nm). The most commonly used D illuminant in color analysis is D65 that represents typical daylight with a temperature of about 6500 K. However, D65 illuminant is difficult to reproduce in practice. It can only be approximated by daylight simulators based on fluorescent lamps such as the broadband F7 illuminant for instance. In addition to these illuminants, we can also distinguish the equienergetic illuminant E whose RSPD does not depend on the wavelength, that is also commonly considered as a theoretical reference in color analysis.

Note that as these CIE standard illuminants are not defined along the NIR domain of the spectrum, they may not be adapted for multispectral imaging that usually considers the NIR domain. Figure 2.2 displays the RSPD  $E(\lambda)$ , for all  $\lambda \in [400 \text{ nm}, 1000 \text{ nm}]$  of alternative (computed or measured) illuminants used in [175] to characterize a multispectral camera.

Incident light is either absorbed, reflected, or refracted according to the optical properties of the material surface. *Specular* reflection occurs when the incident light hits a smooth (mirror-like) surface, in which case the reflection angle (measured according to the normal) is equal to that of the incident one. *Diffuse* reflection occurs in the case of rough surfaces when ideal diffusion surfaces (a.k.a. Lambertian surfaces) uniformly reflect the incident light in all directions. In this study, we only consider the Lambertian diffuse reflection model, so that the reflected radiance does not depend on the angle of view.

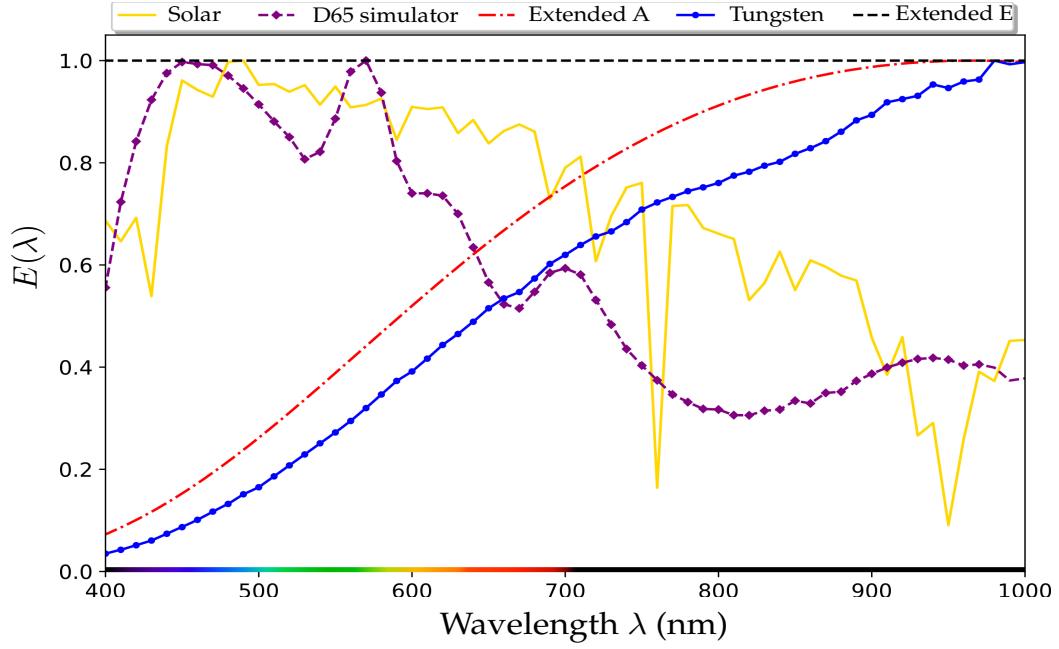


FIGURE 2.2: Computed RSPDs of extended A and E illuminants, and measured RSPDs of solar emission at ground level, of a D65 simulator, and of a practical tungsten realization of A illuminant in the VIS ( $\lambda \in [400 \text{ nm}, 700 \text{ nm}]$ ) and NIR ( $\lambda \in [700 \text{ nm}, 1000 \text{ nm}]$ ) domains.

### 2.2.2 Radiance and reflectance

The amount of light reflected by a surface element  $s$  according to wavelength  $\lambda$  is called *radiance* and is defined as the product between the RSPD  $E(\lambda)$  of the illumination and the reflectance spectrum  $R_s(\lambda)$  of the material at  $s$ , as shown in Fig. 2.3.

*Reflectance* is an intrinsic property of a material that depends on the pigments of which it is made. Each surface element  $s$  is characterized by a reflectance spectrum (reflection function of the wavelength) usually normalized between 0.0 and 1.0. Ideal cases are the perfect black chart for which  $R_s(\lambda) \equiv 0$ , and the perfect white diffuser for which  $R_s(\lambda) \equiv 1$  at any element  $s$  of their surface.

## 2.3 Multispectral radiance image acquisition devices

In this section, we first explain how radiance is measured and stored in a pixel of a multispectral radiance image, then we present the main devices that can be used to acquire such an image.

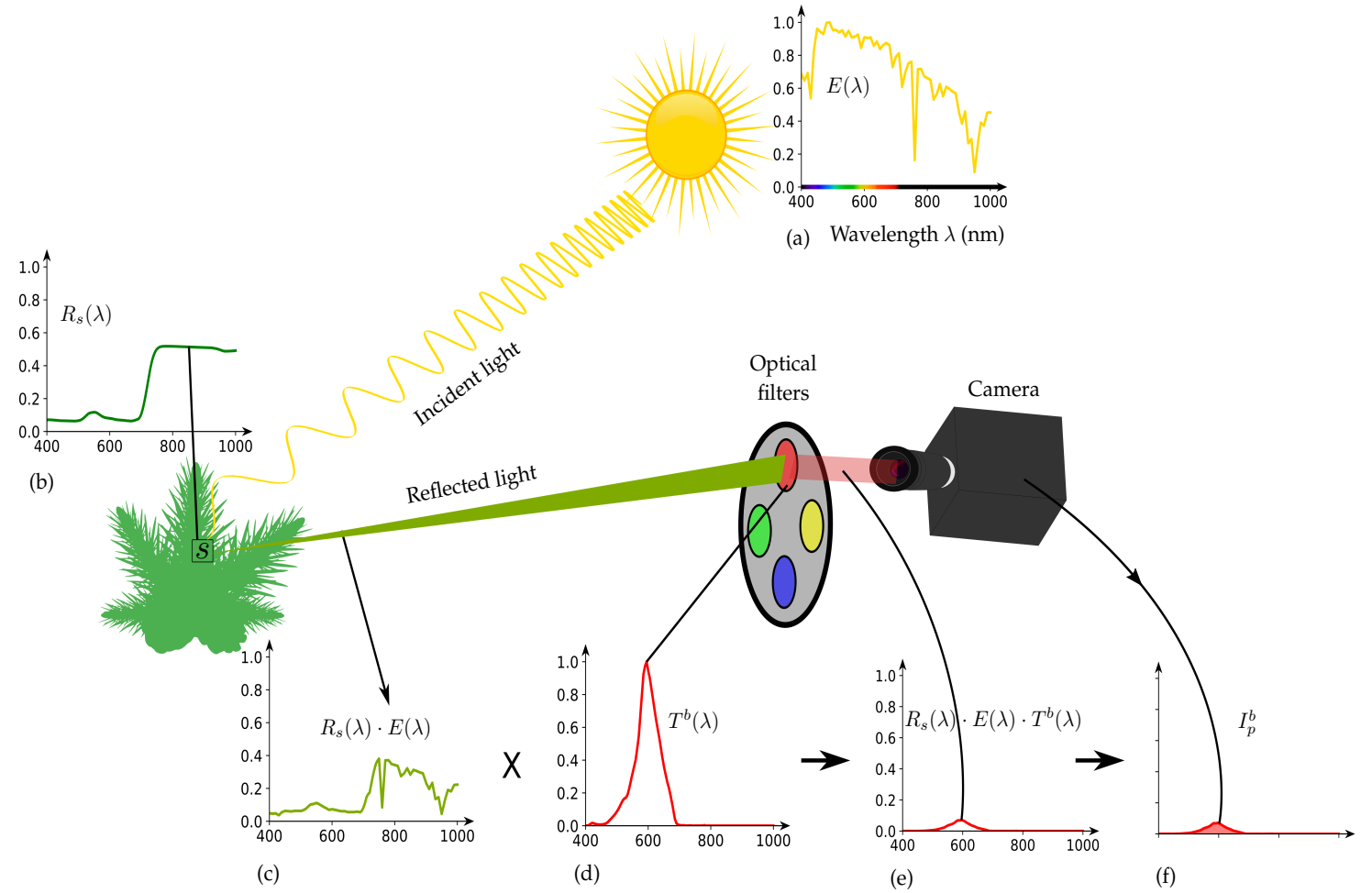


FIGURE 2.3: Interaction between light and surface (Lambertian model). (a) Illumination (solar) RSPD, (b) Reflectance spectrum, (c) Radiance spectrum. (d) Spectral sensitivity function (SSF) of the red filter centered at  $\lambda^b = 600$  nm. (e) Filtered radiance. (f) Intensity value  $I_p^b$ .

### 2.3.1 Multispectral radiance image

The radiance reflected by an object in a given direction can be observed by a digital camera. The latter embeds a photosensitive surface that is composed of a grid of sites covered with spectrally-sensitive filters, that convert the incoming radiance (amount of photons) into an analog signal. This signal is digitized in binary coding by an analog-to-digital converter, which provides a spatially discrete digital image as a two-dimensional matrix of  $X \times Y$  picture elements called pixels. Each pixel  $p$  is associated to a value that represents the quantity of photons reflected by the surface element  $s$  of the scene in the spectral domain of the filter band. A multispectral image is composed of  $B$  spectral channels,  $B > 3$ , and can be seen as a rectangular cuboid (often called cube for short) of size  $X \times Y$  pixels  $\times B$  channels. A spectral

channel  $I^b$  is the representation of the scene radiance filtered according to a given bandpass filter centered at wavelength  $\lambda^b$ , and characterized by its spectral sensitivity function (SSF)  $T^b(\lambda)$  (see Fig. 2.3(d))<sup>15</sup>. Assuming ideal optics, the radiance at a given pixel  $p$  in channel  $I^b$  is expressed by the classical image formation model as:

$$I_p^b = Q \left( \tau \int_{\Omega} E(\lambda) \cdot R_p(\lambda) \cdot T^b(\lambda) d\lambda \right), \quad (2.1)$$

where  $\tau$  is the image integration time and  $Q$  is a quantization function.

A multispectral image composed of a high number of (typically, more than 25) spectral channels is generally referred to as a *hyperspectral image*, even if no consensus exists in the literature for this term. In the following, we stick to the multispectral adjective whatever the number of channels.

Two categories of devices can be distinguished for multispectral image acquisition: “snapshot” (multi-sensor or filter array-based) and “multishot” (tunable filter or illumination-based, push-broom, spatio-spectral linescan) devices (see Fig. 2.5).

### 2.3.2 Snapshot devices

A snapshot device builds the multispectral image from a single shot [67]. This technology provides images composed of few channels. It can acquire multispectral images at video frame rate, and is therefore adapted to observe moving objects. We can distinguish two types of snapshot devices: multi-sensor and single-sensor devices. Multi-sensor snapshot devices acquire a fully-defined multispectral image thanks to dichroic beam splitters that selectively redirects the incoming radiance onto  $B$  sensors according to the wavelength. However, they are expensive, cumbersome, and can sample only a very small number of spectral bands [67]. Single-sensor snapshot devices embed a multispectral filter array (MSFA) (see Fig. 2.4), like the widely-used Bayer color filter array in color imaging, to spatio-spectrally sample the incoming radiance according to the sensor element’s location (see Fig. 2.5(f)) [25, 95]. Each filter in the MSFA is sensitive to a specific narrow spectral band, so that each pixel of the acquired image contains a single radiance value. The fully-defined multispectral radiance image must (afterwards) be estimated thanks to a demosaicing procedure [122].

<sup>15</sup> Note that  $T^b(\lambda)$  includes the sensitivity (spectral response curve) of the sensor itself.

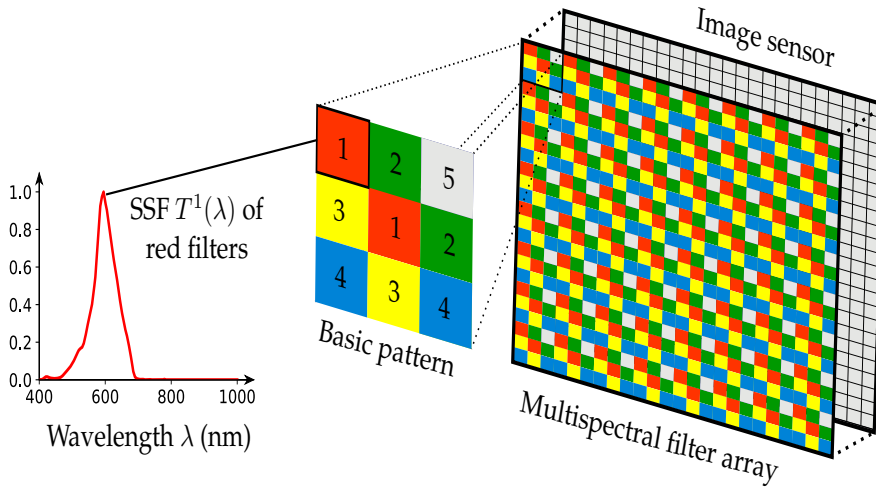


FIGURE 2.4: Image sensor covered by an MSFA. In this example, the  $3 \times 3$  MSFA basic and periodic pattern is sensitive to five different spectral bands. Each pixel in the raw image provided by the sensor contains a single intensity value, and demosaicing is required to estimate the fully-defined 5-channel image.

The computed tomography imaging spectrometer (CTIS) [38] requires no spatio-spectral scan of the scene to acquire a multispectral radiance image, and can therefore also be considered as a snapshot device. To collect radiance in several wavelengths in one shot, CTIS cameras use a diffraction grating and a focal plane array (FPA) to obtain a set of projections. The central projection results from the diffracted scene radiance called the  $0^{th}$ -order of diffraction. This projection can be seen as the summation of the scene radiance over all spectral bands. It contains the scene spatial information. The surrounding projections result from a non-diffracted scene radiance called the  $1^{th}$ -order of diffraction (see Fig. 2.5(g)). They partially overlap and are spatially organized as a spread-out deck of cards. They mostly contain the scene spectral information. The provided 2D image that contains all the projections can be seen as a compressed version of the scene radiance. Because it does not represent a spatio-spectral scan of the scene, each of its pixels cannot be directly associated to a surface element of the scene. A specific reconstruction procedure is needed to provide the fully-defined radiance cube [68].

### 2.3.3 Multishot devices

Multishot devices build the multispectral image from several and successive acquisitions [18, 58, 188]. Although restricted to still scenes, they provide images with

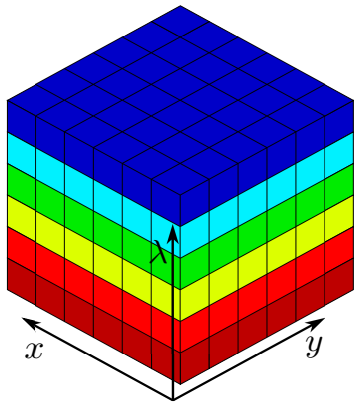
a high spectral resolution, and allow to analyze material surfaces in many narrow spectral bands. Multishot technologies can be categorized according to whether they rely on a channel-wise, row-wise, or pixel-wise scan of the scene.

The tunable filter-based technology [58] acquires one fully-defined channel at a time (see Fig. 2.5(b)) by changing the optical filter in front of the camera mechanically (e.g., thanks to a filter wheel) or electronically (e.g., thanks to liquid crystal or acousto-optic tunable filters [58]). Alternatively, this kind of scan can also be achieved using tunable illumination technology [136]. The scene is successively illuminated with  $B$  narrow-band LED illuminations in order to provide the  $B$  channels [138].

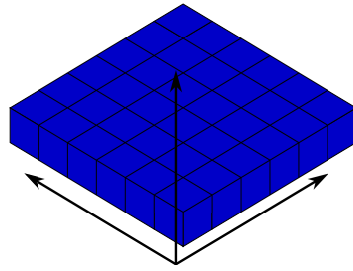
Other scanning technologies can acquire all channels pixel by pixel or row by row. The whisk-broom technology [178] acquires all channels at a single pixel (see Fig. 2.5(c)), therefore  $X \times Y$  acquisitions are needed to build the multispectral image. The push-broom (also called linescan) technology [66] proceeds similarly, but scans the scene in a row-wise manner. At each time, a single row of  $X$  pixels is acquired for all the  $B$  channels so that  $Y$  acquisitions are required to build the entire cube (see Fig. 2.5(d)). Both approaches proceed by spatially scanning the scene to provide the fully-defined cube. This can be done by moving either the scene or the camera (using a translation stage) according to the direction orthogonal to the acquired rows.

Spatio-spectral scanning devices [40] provide a 2-D representation (frame) of the scene of size  $X \times B$  pixels at each time (see Fig. 2.5(e)). The  $b$ -th pixel row contains the spectral information associated to the  $b$ -th channel. The multispectral image is provided thanks to a specific reconstruction procedure from  $Y$  acquired frames of size  $X \times B$  pixels. Recently, a new type of spatio-spectral linescan device has emerged that does not require to move the scene or the camera. Indeed, the sensor moves inside the camera in order to scan a still scene. This feature is particularly interesting to deploy the device in field conditions. In this thesis, we use such a spatio-spectral linescan camera called the *Snapscan* [141] for multispectral radiance image acquisition. Further details about this camera are given in the next section.

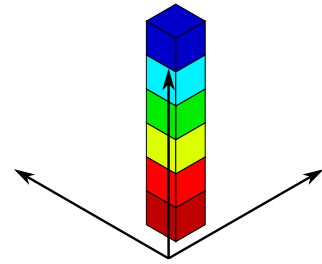




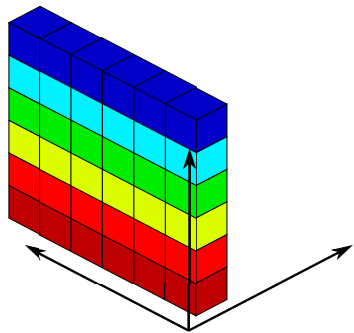
(a) Multispectral image



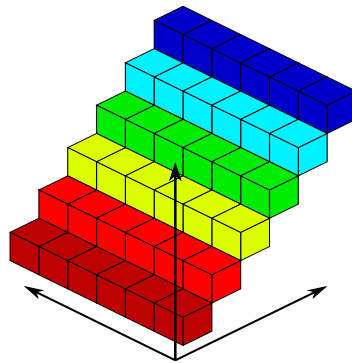
(b) Tunable filter or illumination



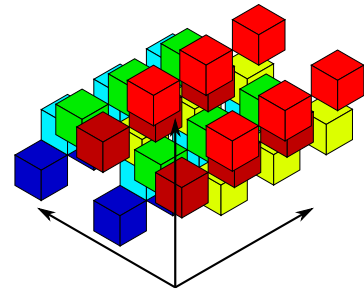
(c) Whisk-broom



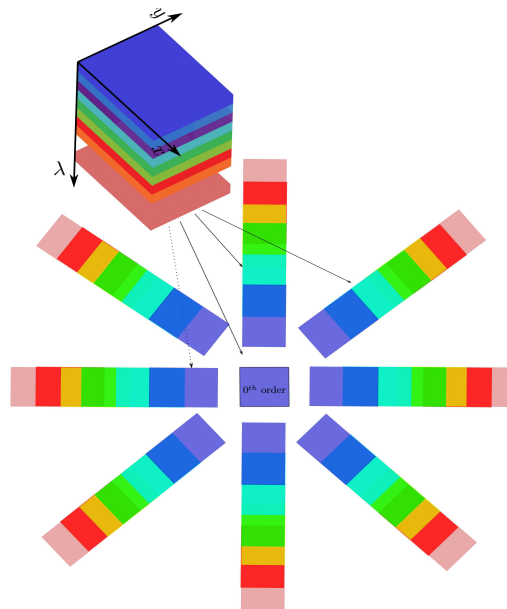
(d) Push-broom



(e) Spatio-spectral linescan



(f) MSFA



(g) CTIS

FIGURE 2.5: Multispectral image acquisition technologies. (b–e) Multishot devices, (f,g) Snapshot devices.

## 2.4 Multispectral radiance image acquisition by Snapscan camera

In this section, we first introduce the Snapscan camera, then we explain how radiance reflected by a given surface element of a scene is measured by the Snapscan camera. In Secs. 2.4.3 and 2.4.4, we present how the Snapscan acquires frames and assembles their stripes into a multispectral radiance image. Finally, in Sec. 2.4.5 we present our proposed multispectral image formation model for multispectral images acquired by multishot devices such as the Snapscan. This model shows that the spatio-spectral correlation assumption does not hold for radiance images acquired by such a device. This section is extracted from our published paper [7].

### 2.4.1 Snapscan camera

The Snapscan [141] is a multispectral camera manufactured by IMEC that embeds a single matrix sensor, covered by a series of narrow stripes of Fabry-Perot filters (see Fig. 2.6). It contains  $B = 192$  optical filters whose central wavelengths range from  $\lambda^{b=0} = 475.1$  nm to  $\lambda^{b=191} = 901.7$  nm with a variable center step (from 0.5 nm to 5 nm). Specifically, each filter is associated to 5 adjacent rows of 2048 pixels that form a filter stripe, and samples a band from the VIS or NIR spectral domain according to its SSF  $T^b(\lambda)$ ,  $b = 0, \dots, B - 1$ , with a full width at half maximum (FWHM) between 2 nm and 10 nm (see Fig. 2.7). We can see that the SSFs of some filters (those centered at 884.7 nm and 888.0 nm for instance) strongly overlap, which means that the spectral information associated to these filters may be redundant or highly correlated.

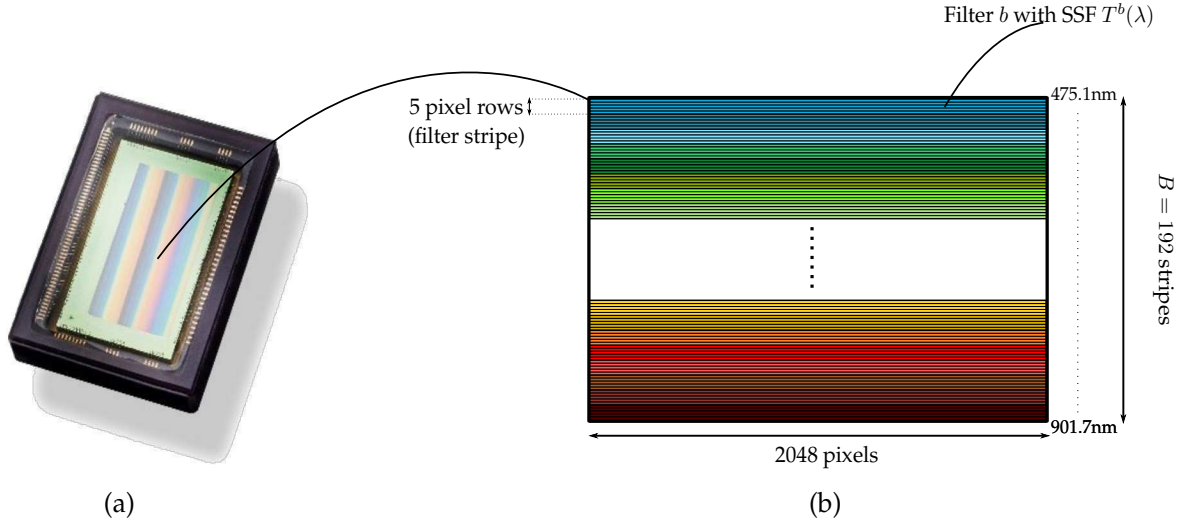
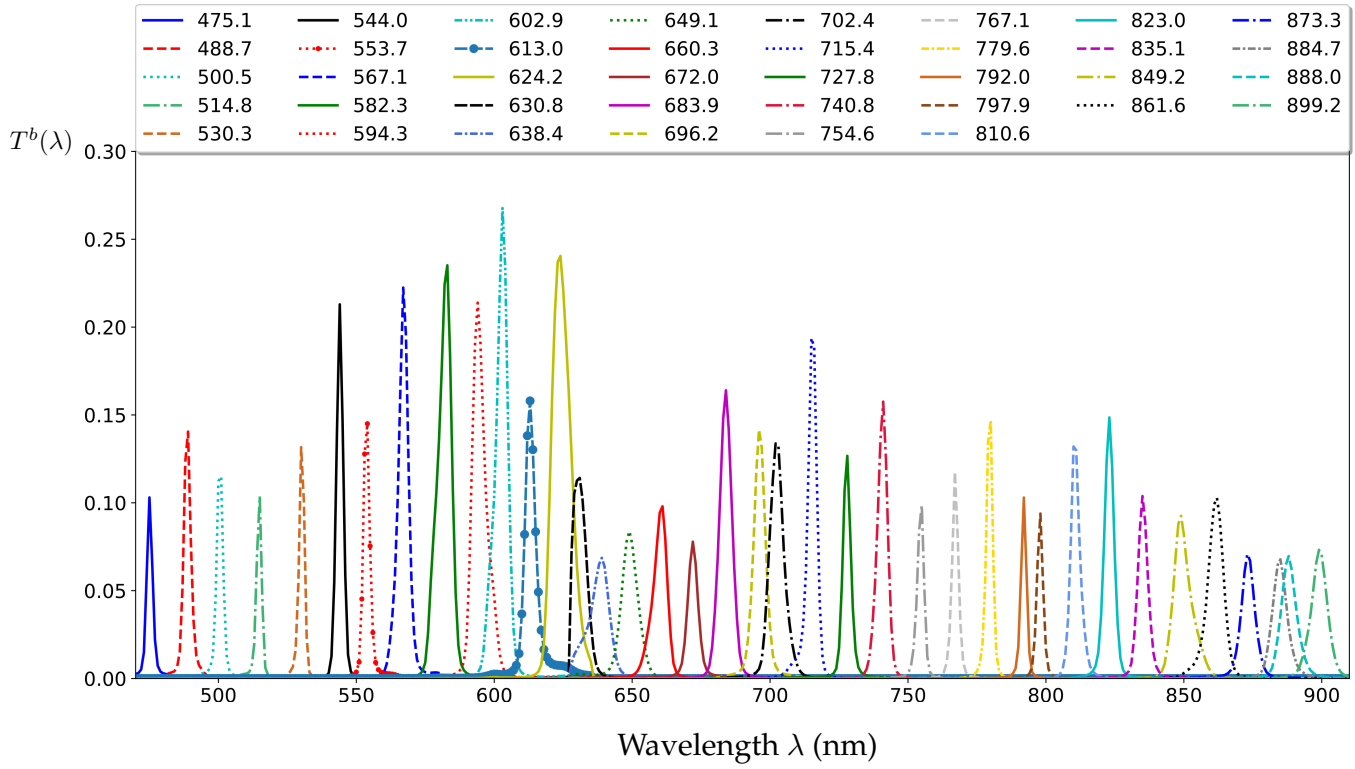


FIGURE 2.6: IMEC Snapscan: (a) sensor, (b) integrated filters.

FIGURE 2.7: SSF  $T^b(\lambda)$  of 38 filters (among the 192) embedded in the Snapscan camera.

### 2.4.2 Radiance measurement

Understanding how a multispectral camera measures radiance and forms images is necessary to develop adequate techniques to analyze them, and to accurately estimate reflectance. This also provides the ability to simulate the acquisition of multispectral radiance images according to the known camera characteristics.

The Snapscan camera acquires a sequence of frames to provide a multispectral image. During frame acquisitions, both the object and camera remain static while the sensor moves inside the camera. Therefore, the measurement of the radiance that is reflected by a given Lambertian surface element  $s$  of the scene varies according to the frame acquisition time  $t$ , although  $s$  is projected at a fixed point  $q$  of the image plane. During frame acquisitions, illumination may spectrally vary in outdoor conditions and we denote its RSPD at  $t$  as  $E_t(\lambda) \in [0, 1]$ , assuming that it homogeneously illuminates all of the surface elements of the scene. The radiance that is reflected by  $s$  and refracted by the camera lens projects onto the image plane at  $q$  as a stimulus  $\mathcal{L}_{t,q}(\lambda)$ :

$$\mathcal{L}_{t,q}(\lambda) = E_t(\lambda) \cdot R_q(\lambda) \cdot A_q(\lambda), \quad (2.2)$$

where  $R_q(\lambda) \in [0, 1]$  is the spectral reflectance of the surface element  $s$  that is observed by  $q$ , and  $A_q(\lambda) \in [0, 1]$  is the optical attenuation of the camera lens at  $q$ . All these functions depend on the wavelength  $\lambda$ . The sensor moves forward on the image plane according to the direction perpendicular to the filter stripes (see Fig. 2.8(a)). Between two successive frame acquisitions, it moves by a constant step  $v = 5$  (in pixels) that is equal to the number of rows in each stripe. Therefore, the radiance that is measured at  $q$  is filtered by a different Fabry–Perot filter of index  $b_{t,q}$  ( $0 \leq b_{t,q} < B$ ) at each acquisition time  $t$ . The radiance at  $q$  is fully sampled over  $N$  frame acquisitions, provided that each of them measures the radiance there, i.e.,  $N \geq B$ . Let the coordinates of point  $q$  be  $(x_q, y_q)_C$  in the camera 2D coordinate system  $(O, x, y)$  whose origin  $O$  corresponds to the intersection between the optical axis and image plane. The unit vectors of  $x$  and  $y$  are given by the photo-sensitive element size (i.e., axis units match with pixels), and  $y$  is oriented opposite to the

sensor movement. At a given point  $q$ , the filter index  $b_{t,q}$  can then be expressed as:

$$b_{t,q} = t + \left\lfloor \frac{y_q - y_0}{v} \right\rfloor, \quad (2.3)$$

where  $y_0$  is the coordinate along  $y$  of the first filter row at first acquisition time  $t = 0$ . Note that the light stimulus  $\mathcal{L}_{t,q}$  is only associated to a filter at a given point  $q$  when  $t_q^0 \leq t < t_q^B$ . The lower bound  $t_q^0 = \lfloor (y_0 - y_q)/v \rfloor$  is the acquisition time at which the first optical filter of the sensor observes  $\mathcal{L}_{t,q}$ . The upper bound  $t_q^B = \lfloor (y_0 - y_q)/v \rfloor + B$  is the time at which all of the sensor filters have observed  $\mathcal{L}_{t,q}$ .

Besides, at a given time  $t$ , the coordinate  $y_q$  of point  $q$  that is associated to a photo-sensitive element of the sensor satisfies:

$$y_0 - t \cdot v \leq y_q < y_0 + (B - t) \cdot v, \quad (2.4)$$

since  $0 \leq b_{t,q} < B$ . Given these restrictions, the radiance  $\mathcal{S}_{t,q}$  that is then measured at  $q$  by the sensor at acquisition time  $t$  is expressed as:

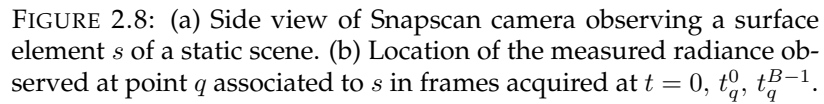
$$\mathcal{S}_{t,q} = Q \left( \tau \int_{\Omega} \mathcal{L}_{t,q}(\lambda) \cdot T^{b_{t,q}}(\lambda) d\lambda \right), \quad (2.5)$$

where  $Q$  is the quantization function according to the camera bit depth,  $\tau$  is the integration time of the frames, and  $\Omega$  is the working spectral domain. Note that  $\tau$  is set to the highest possible value that prevents saturation.

### 2.4.3 Frame acquisition

The radiance measured at  $q$  is stored by the camera as a pixel value  $f_{t,q} = \mathcal{S}_{t,q}$  in frame  $f_t$  (see Fig. 2.8(b)). We define the coordinate system  $(O', x', y')$  attached to the sensor, such that its origin  $O'$  is the first (top-left) photo-sensitive element location, axis  $y'$  corresponds to  $y$ , and  $x'$  is parallel to  $x$ , in order to compute the coordinates of  $q$  relative to the frame. In this frame system, the coordinates of  $q$  are  $(x'_q, y'_q)_F = (x_q, y_q - y_0 + t \cdot v)_F$ . Note that Eq. (2.4) allows us to check that  $0 \leq y'_q < B \cdot v$ . Conversely, any given pixel  $p(x'_p, y'_p)_F$  of a frame  $f_t$  is mapped to the coordinates in the camera coordinate system as:

$$(x_p, y_p)_C = (x'_p, y'_p + y_0 - t \cdot v)_C, \quad (2.6)$$


$$f_{t,p} = \mathcal{S}_{t,p} = Q \left( \tau \int_{\Omega} \mathcal{L}_{t,p}(\lambda) \cdot T^{b_p}(\lambda) d\lambda \right). \quad (2.7)$$

Before the frame acquisitions, the Snapscan uses its internal shutter to acquire a dark frame  $f_{dark}$  whose values are subtracted pixel-wise from the acquired frames. Therefore, we assume that the pixel value expressed by Eq. (2.7) is free from thermal noise. Let us also point out that, at two (e.g., successive) acquisition times  $t_1$  and  $t_2$ , the sensor is at different locations. Therefore it acquires the values  $f_{t_1,p}$  and  $f_{t_2,p}$  from the stimuli  $\mathcal{L}_{t_1,p}$  and  $\mathcal{L}_{t_2,p}$  of two different surface elements at a given pixel  $p$  whose coordinate  $y_p$  in the camera system is time-dependent (see Eq. (2.6)). Besides, the stimuli  $\mathcal{L}_{t_1,p}$  and  $\mathcal{L}_{t_2,p}$  are filtered by the same filter whose index only depends

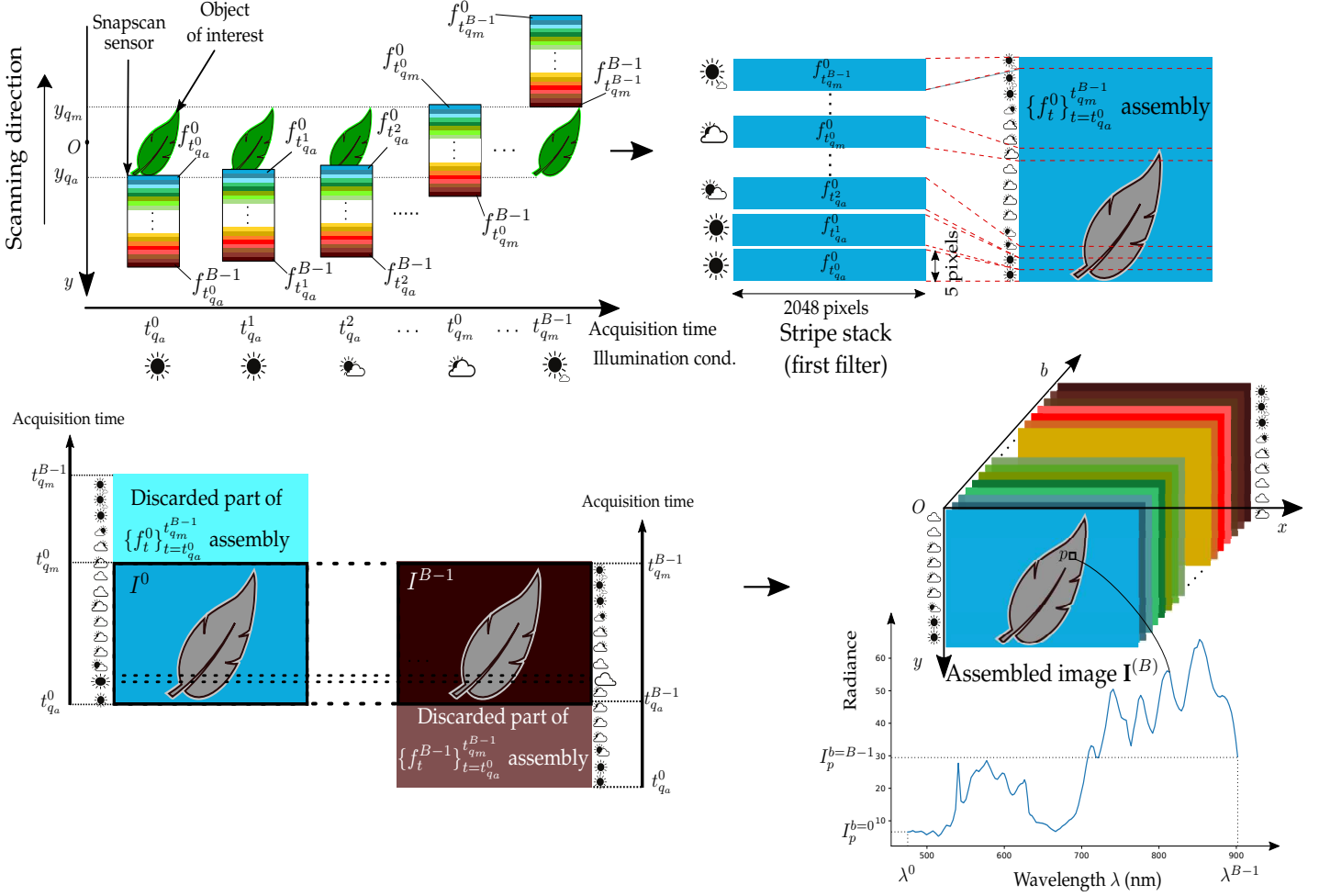


FIGURE 2.9: Frame acquisition and stripe assembly for channel  $I^0$  (top) and multispectral image with  $B$  spectral channels (bottom).

on the pixel coordinate  $y'_p$  in the frame system. Equations (2.5) and (2.7) model the radiance measured at a given point in the image plane and stored at a given pixel of a frame, respectively. Both equations take account of illumination variation during the frame sequence acquisition, but differently take the sensor movement into account. Indeed, the filter index changes at a given point of the image plane during the frame acquisition (see Eq. (2.5)), whereas the observed surface element changes at a given pixel in the successive frames (see Eq. (2.7)).

#### 2.4.4 Stripe assembly

We now determine the first and last acquisition times of the frame sequence that is required to capture an object of interest whose projection points on the image plane are bounded along the  $y$  axis by  $q_a(x_{q_a}, y_{q_a})_C$  and  $q_m(x_{q_m}, y_{q_m})_C$ , with  $y_{q_a} > y_{q_m}$ . Given the initial coordinate  $y_0$  of the sensor along  $y$ , we can compute the first and

last frame acquisition times  $t_{q_a}^0$  and  $t_{q_m}^{B-1}$ , so that the measured radiances at the points between  $q_a$  and  $q_m$  are consecutively filtered by the  $B$  sensor filters (see the top part of Fig. 2.9). The multispectral image composed from the frame sequence  $\{f_t\}_{t=t_{q_a}^0}^{t_{q_m}^{B-1}}$  takes account of the spatial and spectral organizations of each frame. A frame  $f_t$  is spatially organized as juxtaposed stripes of  $v$  adjacent pixel rows. A stripe  $f_t^b$ ,  $b = 0, \dots, B-1$ , of  $v$  adjacent pixel rows contains the spectral information of the scene radiance that is filtered according to the SSF  $T^b(\lambda)$  of filter  $b$  centered at wavelength  $\lambda^b$ . All of the stripes that are associated with filter  $b$  in the acquired frames are stacked by the assembly function  $\oplus$  to provide a stripe assembly defined as:

$$\{f_t^b\}_{t=t_{q_a}^0}^{t_{q_m}^{B-1}} \stackrel{\text{def}}{=} \oplus \left( \{f_t\}_{t=t_{q_a}^0}^{t_{q_m}^{B-1}}, b \right) = [f_{t_{q_m}^{B-1}}^b, \dots, f_{t_{q_a}^0}^b]^\top. \quad (2.8)$$

The size of each stripe assembly is 2048 pixels in width and  $N \cdot v$  pixels in height, where  $N = \lfloor (t_{q_m}^{B-1} - t_{q_a}^0) / \Delta \rfloor + 1$  is the number of acquired frames and  $\Delta$  is the frame acquisition period. To form the multispectral image  $\mathbf{I}^{(B)} = \{I^b\}_{b=0}^{B-1}$  of the object of interest, only the scene part that is common to all stripe assemblies is considered by the camera (see the bottom part of Fig. 2.9). Specifically, the retained stripes in the  $b$ -th assembly are acquired between  $t_{q_a}^b$  and  $t_{q_m}^b$  to form each channel  $I^b$ :

$$I^b = \{f_t^b\}_{t=t_{q_a}^b}^{t_{q_m}^b}. \quad (2.9)$$

The multispectral image  $\mathbf{I}^{(B)}$  has its own coordinate system. For convenience, in the sequel, we denote a pixel as  $p(x, y)$  in this system, since the camera and frame coordinate systems are not used any longer.

#### 2.4.5 Image formation model for Snapscan camera

We can now infer an image formation model for multishot linescan cameras, such as the Snapscan. At any pixel  $p$ , the radiance value  $I_p^b$  that is associated to a channel index  $b \in \llbracket 0, B-1 \rrbracket$  is acquired at  $t = t_p^b$ , with  $t_{q_a}^b \leq t_p^b \leq t_{q_m}^b$  (see Eq. (2.9)). It results from the light stimulus  $\mathcal{L}_{t_p^b, p}$  that is filtered according to  $T^b$  (whose index dependence upon  $p$  is dropped by stripe assembly step), and is therefore defined



from Eqs. (2.2) and (2.7), as:

$$I_p^b = Q \left( \tau \int_{\Omega} E_{t_p^b}(\lambda) \cdot R_p(\lambda) \cdot A_p(\lambda) \cdot T^b(\lambda) d\lambda \right). \quad (2.10)$$

The term  $E_{t_p^b}(\lambda)$  of Eq. (2.10) points out that illumination is associated to both a channel index and a pixel. These dependencies may weaken the spatio-spectral correlation assumptions of the measured scene radiance. *Spectral correlation* relies on the assumption that the SSFs that are associated to adjacent spectral channels strongly overlap. Thus, radiance measures at a given pixel in these channels should be very similar (or correlated). Let us consider the radiance values in two channels  $b_1$  and  $b_2$  at a given pixel  $p$ . Even if the SSFs  $T^{b_1}(\lambda)$  and  $T^{b_2}(\lambda)$  strongly overlap (and are equal in the extreme case), the illumination conditions at  $t_p^{b_1}$  and  $t_p^{b_2}$  are different, hence  $I_p^{b_1} \neq I_p^{b_2}$ . *Spatial correlation* relies on the assumption that reflectance across locally close surface elements of a scene does (almost) not change. Thus, under the same illumination, the radiance measures at their associated pixels within a channel are correlated. Let us consider two pixels,  $p_1(x_{p_1}, y_{p_1})$  and  $p_2(x_{p_2}, y_{p_2})$  that observe surface elements of a scene with the same reflectance  $R_{p_1}(\lambda) = R_{p_2}(\lambda)$  for all  $\lambda \in \Omega$ . If  $|y_{p_1} - y_{p_2}| \geq v$ , then the radiances at  $p_1$  and  $p_2$  are acquired at different times  $t_{p_1}^b$  and  $t_{p_2}^b$  associated to different illumination conditions  $E_{t_{p_1}^b}$  and  $E_{t_{p_2}^b}$ , hence  $I_{p_1}^b \neq I_{p_2}^b$ . Therefore, the spatio-spectral correlation assumption does not hold in the image formation model of the Snapscan camera when illumination varies.

## 2.5 Conclusion

In this chapter, we have first provided an overview about multispectral radiance image acquisition and presented the available technologies that can be used to acquire such an image. Specifically, we have introduced the camera used in this study (i.e., IMEC Snapscan camera), and detailed its working principles. We have shown how illumination variation during the multispectral image acquisition by this device impacts the measured radiance from a Lambertian surface element. This makes the classical image formation model not adapted to our case. Therefore, we have proposed an original multispectral image formation model suited to outdoor acquisition conditions. This model can be adapted to several multishot devices, such as

the HySpex VNIR-1800<sup>16</sup> or the V-EOS Bragg-grating camera [44]. In Chapter 3, we use it to propose a specific reflectance estimation method that takes illumination variation during frame acquisitions into account.

---

<sup>16</sup> <https://www.hypex.com/hypex-products/hypex-classic/hypex-vnir-1800/>



# Reflectance estimation

## Contents

---

|                                                              |           |
|--------------------------------------------------------------|-----------|
| <b>3.1 Introduction</b>                                      | <b>44</b> |
| <b>3.2 State-of-the-art about reflectance estimation</b>     | <b>44</b> |
| 3.2.1 Reflectance estimation under constant illumination     | 46        |
| 3.2.2 Reflectance estimation under varying illumination      | 48        |
| <b>3.3 Reference reflectance estimation</b>                  | <b>50</b> |
| 3.3.1 Assumption validation                                  | 51        |
| 3.3.2 Reflectance estimation                                 | 55        |
| 3.3.3 Spectral correction                                    | 57        |
| 3.3.4 Negative value removal                                 | 58        |
| <b>3.4 Reflectance estimation under varying illumination</b> | <b>59</b> |
| 3.4.1 White diffuser in the scene                            | 60        |
| 3.4.2 Vignetting correction                                  | 61        |
| 3.4.3 Row-wise (rw) based reflectance estimation             | 63        |
| 3.4.4 Optimized row-wise (orw) based reflectance estimation  | 66        |
| <b>3.5 Conclusion</b>                                        | <b>68</b> |

---

### 3.1 Introduction

THE Snapscan camera acquires a sequence of frames to build a multispectral image. During frame acquisitions, both object and camera remain static while the sensor moves. Since illumination may vary in the meanwhile, the measured radiance reflected by a surface element  $s$  of the object depends on the frame acquisition time  $t$ . To get illumination-invariant spectral signatures, we propose to estimate reflectance [115]. Besides, reflectance is strongly related to the biophysical and biochemical properties of a material (vegetation in our case) [156]. In order to get a description of these properties, reflectance needs to be estimated.

In this chapter, we address the problem of reflectance estimation from the multispectral radiance images acquired by Snapscan camera in outdoor conditions. The multispectral image formation model of Eq. (2.10) shows that radiance values within each channel are measured under different illumination conditions. Therefore, illumination variation must be considered at the frame level to accurately estimate reflectance.

In Sec. 3.2, we present state-of-the-art reflectance estimation methods, including their assumptions and limitations. In Sec. 3.3, we first validate our assumptions about filter SSFs and illumination used for reflectance computation. Then, we present the classical approach to estimate reflectance from radiance images acquired in constant illumination conditions, and the post-processing steps that such images undergo due to the sensor configuration. In Sec. 3.4, we introduce reflectance estimation in outdoor conditions and present two reflectance estimation methods that take illumination variations into account at the frame level.

### 3.2 State-of-the-art about reflectance estimation

In order to compute reflectance, the following assumptions are generally considered:

- (i) Each of the optical filters (Fabry-Perot for Snapscan) has an ideal SSF:

$$T^b(\lambda) = \delta(\lambda - \lambda^b) = \begin{cases} 1, & \text{if } \lambda = \lambda^b \\ 0, & \text{otherwise.} \end{cases}$$

- (ii) The illumination does not vary during the image acquisition, thus  $E_{t_p^b}(\lambda) = E(\lambda)$  for all  $b \in \llbracket 0, B-1 \rrbracket$  and  $p \in \mathbf{I}^{(B)}$ . Recall that the term  $E_{t_p^b}(\lambda)$  points out that illumination is associated to both a channel index and a pixel (see Eq. (2.10)).

- (iii) The illumination is spatially uniform.

The ideal SSF assumption (i) is usually retained in order to derive reflectance according to the image formation model (see Eq. (2.1)) [51, 148]. However, assumption (ii) fundamentally depends on the camera technology, and assumption (iii) on the acquisition environment. Since snapshot devices provide an image from a single shot, their acquisitions are ideally not affected by illumination variation. Hence, assumption (ii) actually holds and the accuracy of reflectance estimation only depends on the estimation method. Because multishot linescan devices provide multispectral images from several and successive frame acquisitions (see Sec. 2.3.3), the estimated reflectance accuracy both depends on the estimation method and on illumination variation. Assumption (iii) can be considered fulfilled in outdoor conditions where the main source of illumination is the sun. However, in indoor (or laboratory) conditions where scene illumination is generated by an artificial light source, the spatial uniformity assumption (iii) depends on the capacity of the light source to provide enough radiant energy in the considered spectral bands to illuminate all surface elements of the scene equally.

In Sec. 3.2.1, we present state-of-the-art reflectance estimation methods that assume constant illumination conditions during image acquisition. Section 3.2.1.1 focuses on device-based reflectance estimation methods, Sec. 3.2.1.2 on statistical-based ones, and Sec. 3.2.1.3 on methods based on a training procedure. Then in Sec. 3.2.2, we present some state-of-the-art methods that estimates reflectance from radiance images acquired in outdoor conditions using multishot devices.

### 3.2.1 Reflectance estimation under constant illumination

Considering the classical image formation model of Eq. (2.1) and the above assumptions, several reflectance computation methods have been proposed for color or multispectral imaging. Most of these methods rely either on illumination estimation [50, 85, 183], or on finding a linear transform that maps radiance measurements (RGB values or spectra, depending on the camera spectral resolution) into reflectance spectra thanks to prior knowledge about the SSFs of the sensor filters or reference reflectances of samples from the scene of interest [59, 104, 137, 187]. According to the approach followed to estimate reflectance, we categorize these methods into three main groups: device-based, statistics-based, and training-based methods.

#### 3.2.1.1 Device-based methods

When multispectral images are acquired by a snapshot camera [24, 88], they are not affected by illumination variation. So, device-based methods first estimate the illumination by including a reference panel (a white diffuser or a ColorChecker Chart) in the scene [24, 37, 49, 177]. Then, reflectance is estimated at each pixel  $p$  by channel-wise dividing the radiance image value at  $p$  by the pixel values of the white diffuser at  $p$  or the average value of the ColorChecker white patch.

A snapshot multispectral camera is embedded on a UAV to acquire outdoor multispectral radiance images in [88]. Incident illumination (irradiance) is measured thanks to the acquisition of a white diffuser image. To estimate scene reflectance, the authors assume that only the brightness of illumination varies. They split the measured illumination into brightness and spectral components. A correction factor is then estimated for each channel from the spectral component using a maximum likelihood approach. Reflectance is deduced by normalizing each radiance channel with respect to its associated estimated correction factor. Other methods use a spectroradiometer rather than reference devices to directly measure the downwelling irradiance. In [191], a multispectral camera is used in conjunction with a skyward pointing spectrometer to estimate reflectance from the acquired scene radiance.

### 3.2.1.2 Statistics-based methods

As an alternative to device-based methods, illumination can be estimated thanks to image statistics [52]. The max-RGB, gray-world, shades-of-gray, and gray-edge algorithms are commonly used in color imaging [180]. These algorithms operate channel-wise to estimate illumination. Khan et al. [85] propose an extension of these algorithms to the multispectral domain. The max-spectral algorithm uses the maximum pixel value within each channel to estimate illumination. The spectral gray-world algorithm uses the average values within each channel. The spectral shades of gray algorithm computes the mean  $p$ -norm of radiance values, and the spectral gray-edge computes the mean  $p$ -norm of radiance differences (deduced by Gaussian filtering). The authors use reflectance images (estimated by channel-wise dividing radiance images acquired by a multishot tunable filter-based camera with radiance values of a small neutral (Munsell) reference surface measured by a spectroradiometer after each image acquisition) from the Foster Dataset 2004 [53]. They simulate radiance images using different illuminants (D65, F11, mixture of both) and different filter SSFs (equi-Gaussian, Dirac delta, equi-energy). They evaluate the performance of illumination estimation using different numbers of channels. They report good performances for the max-spectral and the spectral gray-edge algorithms. Note that because radiance is simulated from reflectance images, assumptions (ii) and (iii) hold. However the first assumption (i) holds only when the filter SSFs used to simulate radiance images are Dirac delta.

Although statistics-based methods are simple to implement since they do not require additional devices included in the scene, they may be prone to errors because they directly depend on the physical properties of the imaged scenes.

### 3.2.1.3 Training-based methods

Reflectance may also be estimated using methods based on a training procedure. Linear inverse methods build a mapping matrix to transform the radiance values measured by a camera into reflectance ones [164]. To obtain this matrix, conventional methods are used such as Wiener estimation [160, 166], pseudo-inverse method [33], or partial least-squares [161]. In [87], a model based on Wiener estimation and spectral gray-edge method is proposed to estimate multispectral reflectance. The authors



show that incorporating a spectral adaptation transform [84] to the linear model can improve the quality of reflectance estimation, especially when the SSFs of the sensor filters overlap. To improve reflectance estimation, the mapping matrix is estimated in [75] from a training set of radiance and reflectance values using regression and Gaussian kernel functions.

A similar approach is proposed in [187] to estimate reflectance spectra of art painting images, using k-means clustering to extract relevant training samples. At the surface elements associated to these samples, reflectance spectra are measured using a spectroradiometer. The mapping matrix is deduced from these samples using a Gaussian kernel function and pseudo-inverse computation. Other methods assume that spectral reflectance can be approximated by a linear combination of a few basis vectors derived by principal component analysis [102] or singular value decomposition [92, 131] deduced from training reflectance spectra. In [116] and [82], a learning approach based on neural networks is proposed to estimate reflectance. The training set is composed of radiance and reflectance spectra of the GretagMacbeth™ ColorChecker chart patches. The trained neural networks behave like a mapping matrix: they take radiance values as inputs and give probabilistic outputs that represent an estimate of spectral reflectance. Other methods use CNNs to estimate illumination [19, 163]. They assume a strong relationship between the training samples (ColorChecker patches for example) used to build the mapping matrix and the remaining scene objects. The underlying assumption is that reflectance of the scene objects can be estimated from the mapping matrix deduced from the training samples. This generalization however depends on both the training set and the physical properties of scene objects. When their spectral reflectance is too different from that of the training samples used to build the mapping matrix, their estimated reflectances might be inaccurate.

### 3.2.2 Reflectance estimation under varying illumination

In outdoor conditions, the main source of illumination is the sun. Thus, the incident irradiance can be assumed to be spatially uniform over the scene but is subject to variations during frame acquisitions by the Snapscan camera. Figure 3.1 illustrates the variation of incident irradiance during 2 hours of a cloudy day at wavelengths

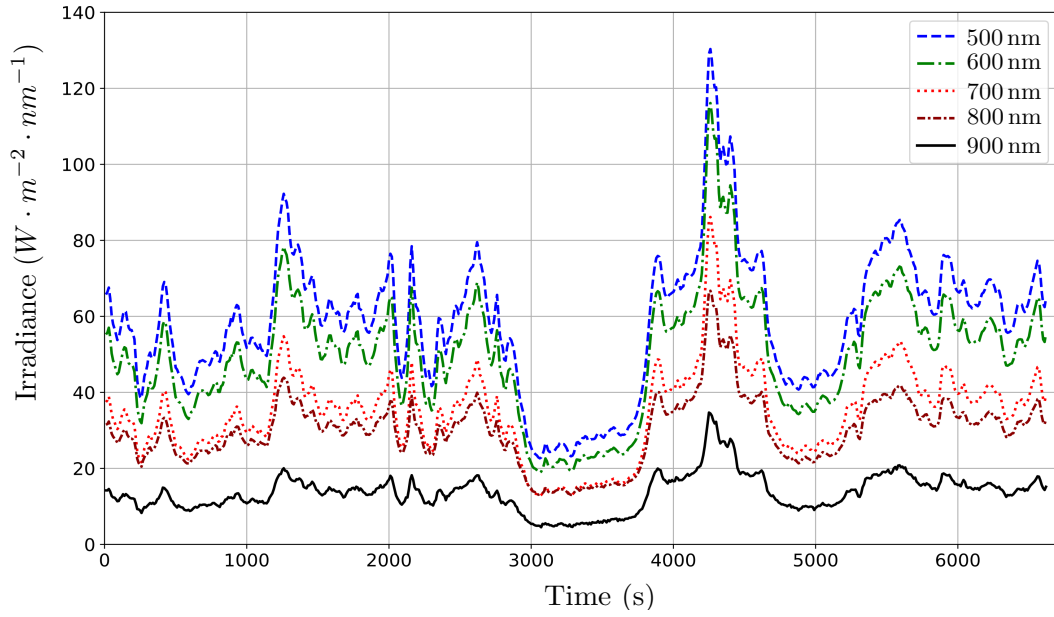


FIGURE 3.1: Illumination variation on a cloudy day (after noon) at wavelengths 500, 600, 700, 800, and 900 nm. The spectroradiometer was configured to perform an irradiance measurement each 10 s during 2 h.

500, 600, 700, 800, and 900 nm measured in crop field by the AvaSpec-ULS2048L spectrometer. We can see that the downwelling irradiance can vary significantly on cloudy days. These variations may be more significant and abrupt when sun light is obscured by small clouds.

Since the Snapscan camera builds a multispectral image from successive frame acquisitions that may last several seconds, radiance measurement is described by Eq. (2.10). With  $E_{t_p}(\lambda) \neq E(\lambda)$ , assumption (ii) (see Sec. 3.2) does not hold any longer. Since the Snapscan can be considered as a spatio-spectral linescan camera, we now review the literature about reflectance estimation from radiance images acquired by linescan cameras under varying illumination.

In [44], a Bragg-grating-based multispectral camera (the V-EOS) manufactured by Photon Etc. is used to provide outdoor multispectral radiance images by spatio-spectrally scanning the scene (see Sec. 2.3.3). The spatio-spectral scans provide a sequence of frames that contain columnly variable spectral information. The multispectral image is obtained by a resampling procedure of the acquired frames, which provides a so-called rectified multispectral image. To compute reflectance, the device acquires two images. The first one is a full-field white diffuser image. The second image characterizes the scene of interest in which two white diffusers are

included, such that one covers the bottom image border while the second is fully visible. To estimate scene reflectance, the scene image is first pixel-wise divided by the white diffuser image. Because illumination conditions vary among frame acquisitions, illumination is column-wise scaled using correction factors deduced from the white diffuser image and the white diffuser stripe at the scene image bottom. Finally, reflectance of the interest scene part (all image pixels but those of the two white diffusers) is deduced from the second white diffuser included in the image. Wendel and Underwood [183] use a UGV equipped with a linescan multispectral camera to gather high resolution multispectral images of vegetation under natural illumination. A process of illumination compensation is performed to extract the inherent reflectance properties of vegetation, despite variable illumination. This work adopts the subspace model approach proposed by Drew and Finlayson [42] to recover reflectance and illumination, and enhance its performance by learning a historical database of reflectance spectra and illuminations (measured by a white diffuser) associated to various natural lighting conditions. The methods presented above account for varying illumination conditions, but that of [44] is cumbersome since it needs two white diffusers and two image acquisitions to compute its reflectance, and that of [183] requires to learn historical illuminations and reflectances for better performances.

In Sec. 3.4, we propose a practical method to estimate reflectance from outdoor radiance images acquired under skylight. This method has to be robust against illumination variations during frame acquisitions.

### 3.3 Reference reflectance estimation

This section focuses on reflectance estimation from images acquired by the Snapscan camera in constant illumination conditions. This reflectance is considered as a reference that is useful to compare the quality of outdoor reflectance estimations. First, assumptions about our illumination setup and the Snapscan filter SSFs are validated in Sec. 3.3.1. Then, the classical reflectance estimation model is presented in Sec. 3.3.2. Finally, in Secs. 3.3.3 and 3.3.4, we present the post-processings applied to our estimated reflectance images.

### 3.3.1 Assumption validation

In order to compute reflectance according to the linear image formation model proposed in Eq. (2.10), the assumptions of Sec. 3.2 need to be verified. Let us consider to what extent these assumptions are satisfied by our experimental acquisition environment, specifically filter sensitivities and illumination.

#### 3.3.1.1 Filter sensitivities

Our first assumption is that each Fabry-Perot filter has an ideal response that can be modeled as a Dirac delta function. This assumption is commonly used to model the sensor responses for scene reflectance or illumination estimation [45, 50, 78, 118].

The Snapscan sensor incorporates narrow filters with a full width at half maximum (FWHM) between 2 and 10 nm, some of which are shown in Fig. 3.2.

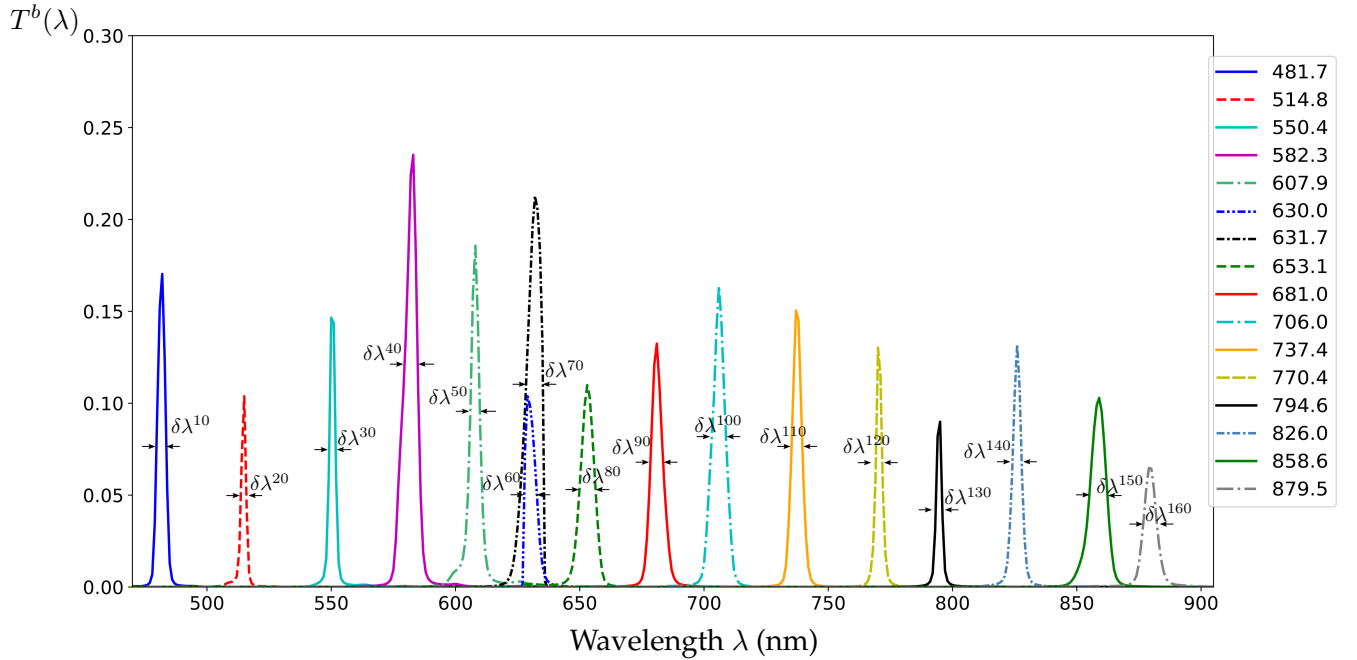


FIGURE 3.2: SSF  $T^b(\lambda)$  and FWHM  $\delta\lambda^b$  of 16 filters ( $(b = 10 : 10 : 160)$  among the 192 ones) embedded in the Snapscan. Caption shows the filter centers. Respective FWHM values (in nm) are: 2.74, 1.83, 2.40, 3.81, 3.64, 4.22, 4.88, 4.88, 4.39, 3.98, 3.48, 2.57, 2.16, 3.23, 5.46, 4.88.

An important property when using devices integrating narrow band filters is the spectral resolving power (or finesse). To quantify the finesse  $\mathcal{F}$  of devices such as Fabry-Perot interferometers, the Rayleigh's criterion, the free spectral range (FSR)  $\Delta\lambda$ , and the FWHM  $\delta\lambda$  are generally used [77, 81, 152]. To quantify the finesse of the Snapscan's integrated filters, we compare the FSR  $\Delta\lambda^{b,b+1} = \lambda^{b+1} - \lambda^b$  and the

FWHM  $\overline{\delta\lambda}^{b,b+1} = \frac{\delta\lambda^{b+1} + \delta\lambda^b}{2}$  for each pair of successive SSF centers. The criterion  $\Delta\lambda^{b,b+1} \geq \overline{\delta\lambda}^{b,b+1}$  simply stipulates that two successive SSF centers  $\lambda^b$  and  $\lambda^{b+1}$  must be distant from each other by at least the average of those SSFs (because FWHM is specific to each SSF). In other words, the ratio  $\frac{\Delta\lambda^{b,b+1}}{\overline{\delta\lambda}^{b,b+1}}$  must be at least equal to 1 for a satisfying finesse.

The overall average finesse is given by:

$$\mathcal{F} = \frac{1}{B-1} \sum_{b=0}^{B-2} \frac{\Delta\lambda^{b,b+1}}{\overline{\delta\lambda}^{b,b+1}}. \quad (3.1)$$

For the Snapscan,  $\mathcal{F}$  equals **0.87**. We also compute the average transmitted energy of the Fabry-Perot filters as the average area of their associated SSFs  $\{T^b(\lambda)\}_{b=0}^{B-1}$  over  $\Omega$ :

$$\overline{\mathbb{E}} = \frac{1}{B-1} \sum_{b=0}^{B-1} \sum_{\lambda \in \Omega} T^b(\lambda), \quad (3.2)$$

which is 1.62.

The SSFs of the Fabry-Perot filters integrated in the Snapscan's sensor are not exactly Dirac delta functions ( $\overline{\mathbb{E}} > 1$ ), and the ratio  $\frac{\Delta\lambda^{b,b+1}}{\overline{\delta\lambda}^{b,b+1}}$  is below 1 for most pairs of successive SSF centers. To the extent of our knowledge, there is no multispectral camera with such a fine resolving power allowing to fully validate assumption (i) (see Sec. 3.2).

However, the filters embedded in the Snapscan are very narrow, and because of the optical attenuation  $A_p(\lambda)$  (see Eq. (2.10)), the intensity of the incoming light supposed to reach each filter may be attenuated. As a consequence, the average transmitted energy of the Fabry-Perot filters may be also attenuated. The Snapscan's filters approximate the Dirac delta assumption of (i) at best for today's technology.

### 3.3.1.2 Illumination

Let us now focus on assumptions (ii) and (iii) about illumination. Since the Fabry-Perot filters sample the spectrum over different narrow spectral bands in the VIS and NIR domains, the illumination source must provide enough radiant energy in all of these bands. This would ensure that multispectral images acquired by the Snapscan contains no low-dynamics channel, which could impact reflectance computation. In addition, the multispectral image of either the white diffuser or the color rendition

chart [119] are used as references. Therefore, they must be acquired in laboratory conditions under a constant illumination that is as spatially and spectrally uniform as possible. Thus, we combine different light sources to illuminate the scene in the VIS and NIR domains. Specifically, we use 3 halogen lamps, a white lamp, a blue and a green lamp, that are powered by a stabilized supply (10V) and placed at particular locations above the observed scene (see Fig. 3.4).

To evaluate our illumination system, we use a white diffuser (Spectralon® Standards with 95% diffuse reflectance) and a spectroradiometer (Avantes AvaSpec-ULS2048L), that samples the VIS and NIR domains (from 426 nm to 971 nm) with a sampling step of 0.3 nm. For the measurements, we set the integration time to 2 ms.

Figure 3.3 shows the radiance spectra measured at the five locations of the white diffuser surface displayed in Fig. 3.4.

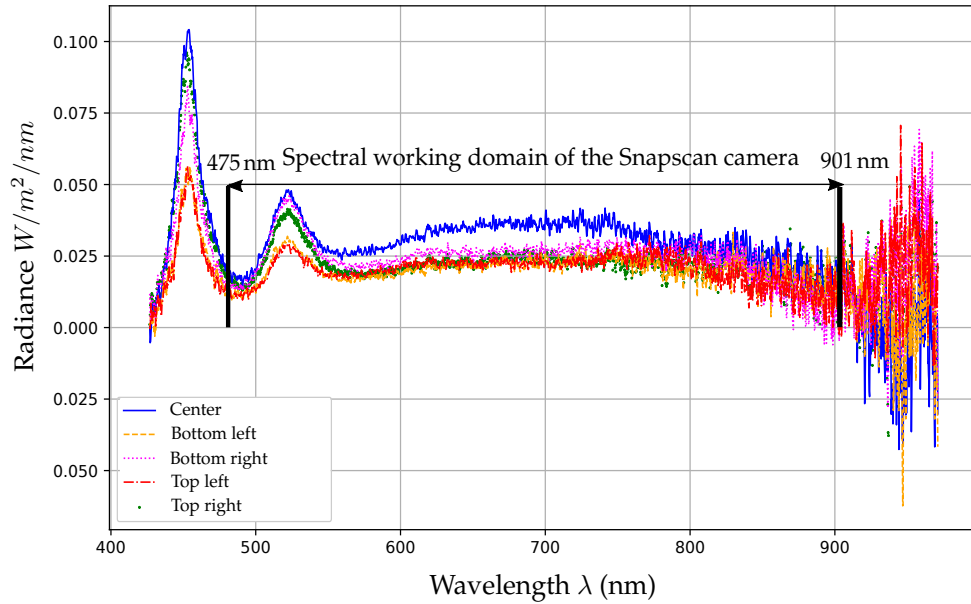


FIGURE 3.3: Illumination at five locations of the white diffuser (center and four corners).

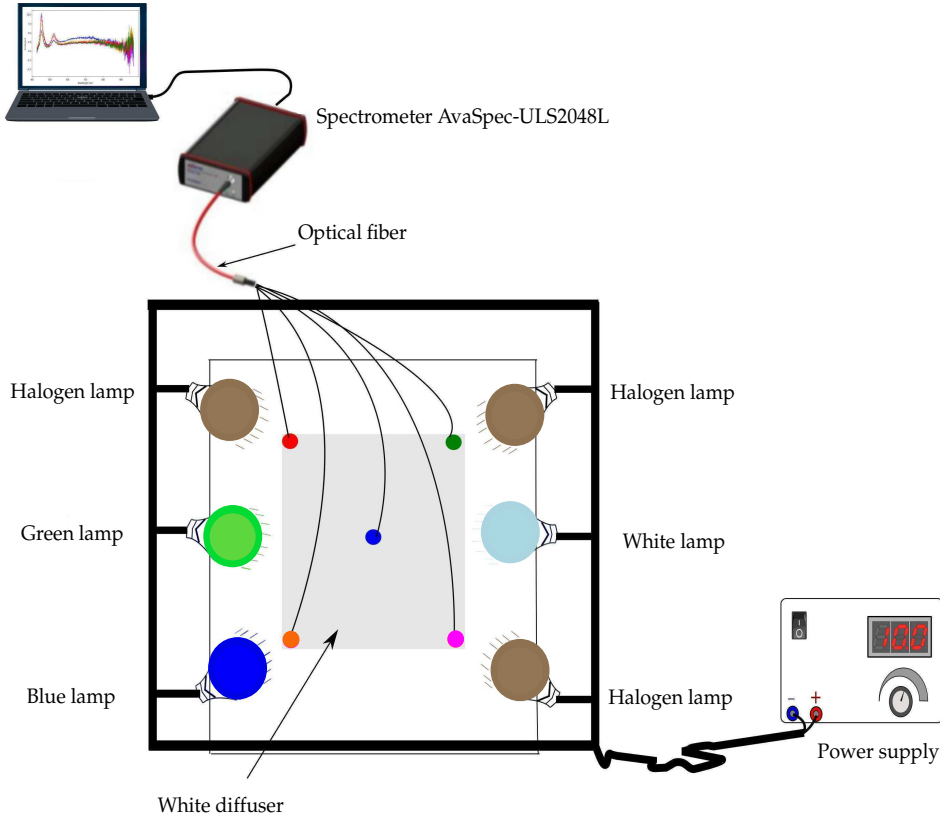


FIGURE 3.4: Illumination system. Colored dots show the white diffuser locations to which the optical fiber is pointed to measure the reflected radiance.

The five radiance spectra have similar shapes, but we can see that the white diffuser center (blue spectrum) receives a slightly larger amount of energy than its corners. Below 550 nm, illumination intensity slightly differs on the left and right sides of the white diffuser. Since the blue and green lamps point to the right, this part of the white diffuser receives more energy from these two lamps. We can also notice that radiance measurements become very noisy above 900 nm since the lamps emit almost no radiant energy in this range of the spectrum. This is not a problem for our image acquisitions since the spectral working domain of the Snapscan camera is [475 nm, 901 nm].

To evaluate our illumination system quantitatively, we assess its spatial uniformity by computing the mean and Kullback-Leibler divergence (KLD) of each spectrum considered as a distribution. KLD is a non-symmetric difference measure between two distributions for which a value of 0.0 indicates that these are identical. It is defined as:

$$D_{KL}(S_{ref}, S) = \sum_{l=0}^{B'-1} S_{ref}(\lambda^l) \cdot \log \frac{S_{ref}(\lambda^l)}{S(\lambda^l)}, \quad (3.3)$$

where  $S_{ref}(\lambda)$  is the radiance measured at the center of the white diffuser considered as the reference,  $S(\lambda)$  is the radiance measured at a corner of the white diffuser, and  $B'$  is the number of spectral bands sampled by the spectroradiometer. Note that after mean computation, all radiance distributions are normalized to compute KLD. The evaluation is restricted to the Snapscan working domain. Table 3.1 shows the illumination uniformity measurements.

| Radiance spectrum  | Mean ( $\times 10^2$ ) | KLD ( $\times 10^2$ ) |
|--------------------|------------------------|-----------------------|
| $S_{BR}(\lambda)$  | 2.377                  | 2.758                 |
| $S_{BL}(\lambda)$  | 2.008                  | 1.629                 |
| $S_{TR}(\lambda)$  | 2.210                  | 2.117                 |
| $S_{TL}(\lambda)$  | 2.011                  | 1.993                 |
| $S_{ref}(\lambda)$ | 3.004                  | -                     |

TABLE 3.1: Spatial uniformity evaluation of the illumination system used in laboratory. BR: bottom-right, BL: bottom-left, TR: top-right, TL: top-left corner.

KLD values show that the illumination distributions at the four corners of the white diffuser are very similar to that at the center. However, KLD reflects the similarity between distributions but not between values. For example, the radiance spectrum  $S_{BR}(\lambda)$  has the lowest similarity (0.027) to the reference distribution  $S_{ref}(\lambda)$ , but has the closest mean value. When comparing the means, we can see that illuminations at the corners of the white diffuser are similar, but have somewhat lower mean values than the illumination at the center. Illumination intensity is slightly higher at the center of the white diffuser. However, this slight difference vanishes away from the center. Thus, we consider that the system delivers an illumination which is nearly spatially uniform, which allows us to check the spatial uniformity assumption (*iii*) (see Sec. 3.2).

### 3.3.2 Reflectance estimation

#### 3.3.2.1 Model

We can now formulate a discrete version of the image formation model of Eq. (2.10) as:

$$I_p^b = \tau \cdot \sum_{l=0}^{B-1} E(\lambda^l) \cdot R_p(\lambda^l) \cdot A_p(\lambda^l) \cdot T^b(\lambda^l). \quad (3.4)$$

Note that the quantization function  $Q$  is omitted in (3.4) since the different terms are considered as being already quantized. Assuming that  $T^b(\lambda)$  has an ideal response



that can be modeled as a Dirac delta function (see Sec. 3.2), we can rewrite Eq. (3.4) as:

$$I_p^b = \tau \cdot E(\lambda^b) \cdot R_p(\lambda^b) \cdot A_p(\lambda^b). \quad (3.5)$$

We can now derive the reflectance for any surface element  $s$  observed by pixel  $p$  associated to spectral band centered at  $\lambda^b$  as:

$$R_p(\lambda^b) = \frac{I_p^b}{\tau \cdot E(\lambda^b) \cdot A_p(\lambda^b)}, \quad (3.6)$$

where the product  $E(\lambda^b) \cdot A_p(\lambda^b)$  characterizes the illumination in the sensor domain for channel  $I^b$ , and can be deduced by acquiring the image of a calibration tile, typically a white diffuser.

### 3.3.2.2 Estimation

In order to estimate reflectance from radiance images that were acquired under an illumination that is almost constant over time, one classically uses the image  $\mathbf{I}^{(B)}[\text{WD}]$  of a white diffuser acquired in full field beforehand. The white diffuser is assumed to be perfectly diffuse and reflect the incident light with a constant diffuse reflection factor  $\rho_{\text{wd}}$  that neither depends on the pixel  $p$  nor on the wavelength  $\lambda$  ( $\rho_{\text{wd}} = 95\%$  in our case). Using again assumption (i), we can write:

$$\rho_{\text{wd}} = \frac{I_p^b[\text{WD}]}{\tau_{\text{wd}} \cdot E(\lambda^b) \cdot A_p(\lambda^b)}, \quad (3.7)$$

where  $\tau_{\text{wd}}$  is the frame integration time of  $\mathbf{I}^{(B)}[\text{WD}]$ . Plugging Eq. (3.7) into (3.6) yields the reflectance image that is estimated from a  $B$ -channel radiance image  $\mathbf{I}^{(B)}$ :

$$\hat{R}_p^b = \rho_{\text{wd}} \cdot \frac{I_p^b}{I_p^b[\text{WD}]} \cdot \frac{\tau_{\text{wd}}}{\tau}. \quad (3.8)$$

This reflectance estimation model implicitly compensates the vignetting effect, since the white diffuser and object (scene of interest) occupy the same (full) field of view. Accordingly,  $I_p^b$  and  $I_p^b[\text{WD}]$  are affected by the same optical attenuation whose effect vanishes after division.

The estimated  $B$ -channel reflectance image  $\hat{\mathbf{R}}^{(B)}$  should then undergo two post-processing steps: spectral correction and negative value removal.

### 3.3.3 Spectral correction

Each of the Fabry-Perot integrated filters is designed to sample a specific spectral band from the spectrum according to its spectral sensitivity function  $T^b(\lambda)$ . However, due to both SSFs and optical properties of some filters (angular dependence [63], high-energy harmonics), several bands are redundant, which limits the accuracy of the spectral imaging system. Figure 3.5 shows two filter SSFs with harmonics and overlapping (finesse  $\mathcal{F}$  equals 0.80) central peaks. According to their central wavelengths, these filters should sample the spectrum at  $\lambda^{40} = 582.3 \text{ nm}$  and  $\lambda^{41} = 585.3 \text{ nm}$  in the green domain, but high-energy harmonics make them sensitive to incident wavelengths in the blue domain. This leads to redundancy in spectral bands and introduces spectral information bias. Therefore, the  $B$ -channel ( $B=192$ ) reflectance image  $\hat{\mathbf{R}}^{(B)}$  is spectrally corrected and only  $K = 141$  channels are kept in practice.

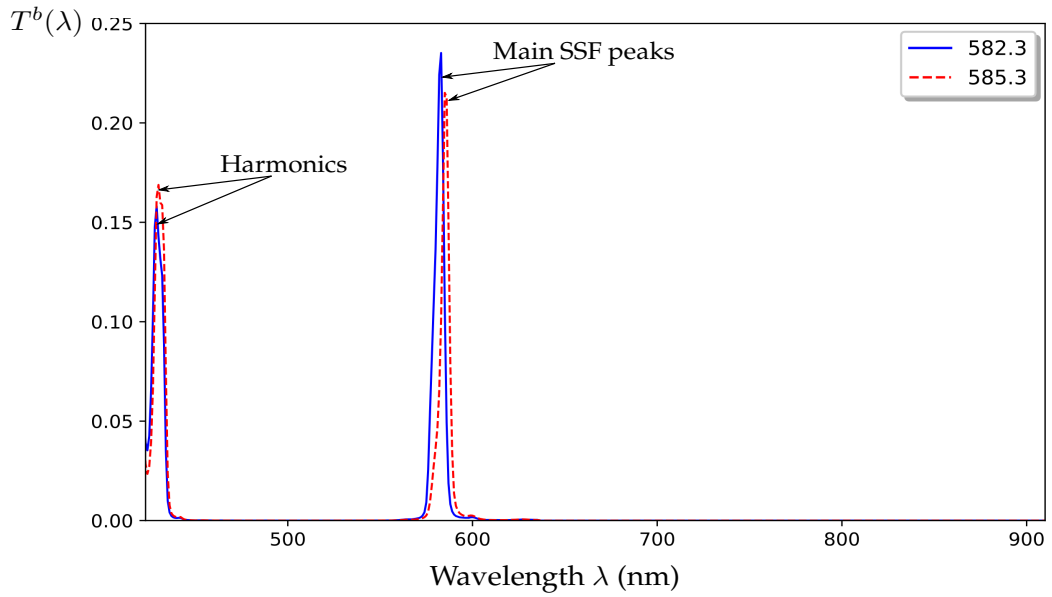


FIGURE 3.5: SSF harmonics of the filters centered at  $\lambda^{b=40} = 582.3 \text{ nm}$  and  $\lambda^{b=41} = 585.3 \text{ nm}$  (FWHMs:  $\delta\lambda^{b=40} = 3.81 \text{ nm}$  and  $\delta\lambda^{b=41} = 3.64 \text{ nm}$ ).

The spectral correction of  $\hat{\mathbf{R}}^{(B)}$  provides a spectrally corrected  $K$ -channel reflectance image  $\hat{\mathbf{R}}^{(K)}$  that is expressed at each pixel  $p$  as:

$$\hat{\mathbf{R}}_p^{(K)} = \mathbf{M} \cdot \hat{\mathbf{R}}_p^{(B)}, \quad (3.9)$$

where  $\mathbf{M}$  is the sparse  $K \times B$  correction matrix that is provided by the calibration file of our Snapscan camera. The linear combinations of the channel values of  $\hat{\mathbf{R}}^{(B)}$  according to Eq. (3.9) are designed by the manufacturer to remove the redundant channels and attenuate second-order harmonics. This spectral correction provides new centers  $\{\lambda^k\}_{k=0}^{K-1}$  for the bands (referred to as “virtual” bands by the manufacturer) that are associated to the image channels, but the spectral working domain  $\Omega = [475.1 \text{ nm}, 901.7 \text{ nm}]$  is unchanged.

### 3.3.4 Negative value removal

The acquired radiance image contains negative values due to dark frame subtraction, when the value of a dark frame pixel is higher than the measured radiance at this pixel. This generally occurs in low-dynamics channels, where the central wavelengths are in the range  $[475.1 \text{ nm}, 560.4 \text{ nm}]$  (before spectral correction). These negative values may lay on vegetation pixels and corrupt reflectance estimation at these pixels. Because we intend to classify vegetation pixels, this could lead to unexpected prediction errors. Negative values also occur—for even more pixels—in the spectrally-corrected reflectance image  $\hat{\mathbf{R}}^{(K)}$  (see Eq. (3.9)), because the correction matrix  $\mathbf{M}$  contains negative coefficients. Negative values have no physical meaning and they must be discarded. Because our images mostly contain smooth textures (vegetation, reference panels, soil), we consider that, unlike radiance, reflectance values are highly correlated over close surface elements. Thus, we propose correcting negative values in image  $\hat{\mathbf{R}}^{(K)}$  by conditionally using a  $3 \times 3$  median filter, as:

$$\hat{R}_{\text{ref},p}^k = \begin{cases} \text{median}_{3 \times 3}\{\hat{R}_p^k\} & \text{if } \hat{R}_p^k < 0, \\ \hat{R}_p^k & \text{otherwise,} \end{cases} \quad (3.10)$$

where  $\hat{R}_{\text{ref},p}^k$  is the final reflectance value at pixel  $p$  for channel  $k$ . Because we consider the reflectance that is estimated by this model (Eqs. (3.8)–(3.10)) as a reference, it is denoted as  $\hat{\mathbf{R}}_{\text{ref}}^{(K)}$ . Figure 3.6(b) shows the rate of negative values in the reflectance image of Fig. 3.6(a) before and after median filtering.

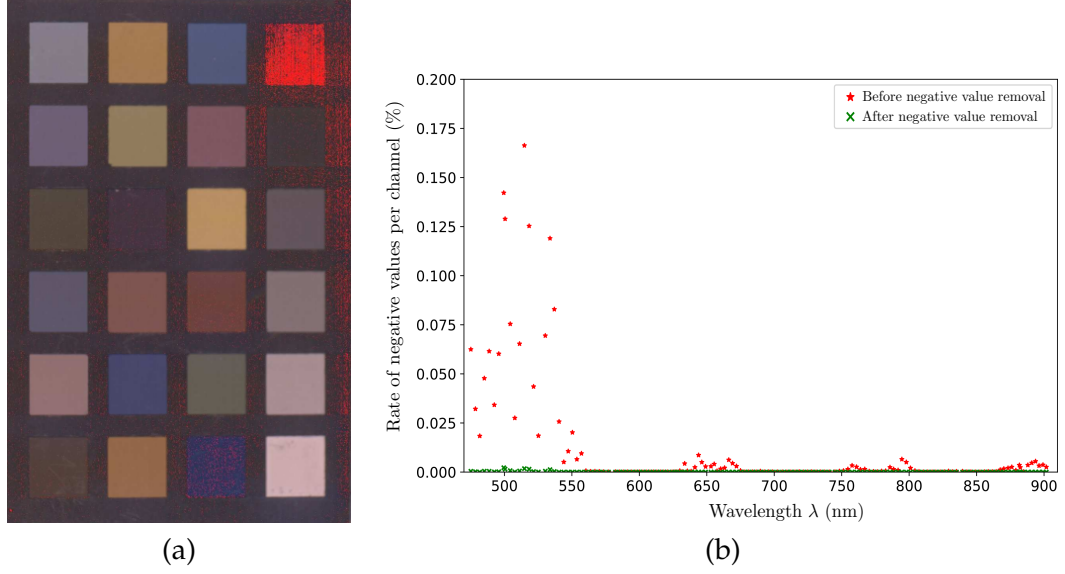


FIGURE 3.6: Negative values in a reflectance image  $\hat{\mathbf{R}}^{(K)}[\text{CC}]$  (a) shown in red on an RGB rendering of  $\hat{\mathbf{R}}^{(K)}[\text{CC}]$ . The rate of negative values per channel in  $\hat{\mathbf{R}}^{(K)}[\text{CC}]$  before and after removal is shown in (b).

### 3.4 Reflectance estimation under varying illumination

This section focuses on reflectance estimation when radiance images are acquired in varying illumination conditions. In the previous section, we show that a  $B$ -channel multispectral reflectance image  $\hat{\mathbf{R}}^{(B)} = \{\hat{R}^b\}_{b=0}^{B-1}$  can be estimated from a radiance one when the illumination of the acquired scene is known (see Eq. (3.6)). Because illumination varies between the acquisitions of the white diffuser image and that of the scene, the reflectance estimation method described by Eq. (3.8) is not adapted to our problem. Moreover, our image formation model (see Eq. (2.10)) shows that illumination variations during the frame acquisitions affect the radiance measurements. Since the multispectral radiance image is reconstructed from individual frames acquired at different times, the pixel rows containing the values of the scene radiance filtered by a given filter are associated to different illumination conditions (see Sec. 2.4.5). Thus, we propose to compute reflectance by analyzing each row of the radiance image. For this purpose, we need to estimate the illumination condition for each row of every channel.

From a given radiance channel  $I^b$ , we propose two methods to estimate the reflectance  $\hat{R}^b$ . The first one (see Sec. 3.4.3) is based on a row-wise estimation from the white diffuser, and denoted by the subscript *rw*. The second one (see Sec. 3.4.4)

is an optimization of the rw method based on a linear regression of the ColorChecker patch values, and denoted by the subscript  $orw$ .

### 3.4.1 White diffuser in the scene

We propose to mount the white diffuser on the acquisition system, so that the sensor vertically observes a portion of it. Therefore, the (about 10%) right border pixels of each row in any channel represent the white diffuser ( $\mathcal{WD}$  zone in Fig. 3.7(b)). Thanks to these pixels, we can estimate the illumination condition for the scene part associated to this row acquired at a specific time. To estimate the reflectance with the proposed method, we assume that the illumination is spatially uniform within each pixel row, such that at each row the white diffuser and the scene part pixels row-wise observe surface elements that are enlightened by the same illumination. Note that this assumption may be violated if local shadows occur during the frame acquisitions.

Since reflectance is no longer estimated according to Eq. (3.8), it would be affected by vignetting effect. To highlight this effect on reflectance estimation, let us consider the quantized discrete version of Eq. (2.10) at pixel  $p_1$  associated to a surface element of the scene of interest as:

$$I_{p_1}^b = \tau \cdot \sum_{l=0}^{B-1} E_{t_{p_1}^b}(\lambda^l) \cdot R_{p_1}(\lambda^l) \cdot A_{p_1}(\lambda^l) \cdot T^b(\lambda^l). \quad (3.11)$$

Under the Dirac delta assumption about  $T^b(\lambda)$ , Eq. (3.11) can be rewritten as:

$$I_{p_1}^b = \tau \cdot E_{t_{p_1}^b}(\lambda^b) \cdot R_{p_1}(\lambda^b) \cdot A_{p_1}(\lambda^b). \quad (3.12)$$

Thus, the reflectance at  $p_1$  is expressed as:

$$R_{p_1}(\lambda^b) = \frac{I_{p_1}^b}{\tau \cdot E_{t_{p_1}^b}(\lambda^b) \cdot A_{p_1}(\lambda^b)}. \quad (3.13)$$

The illumination  $E_{t_{p_1}^b}(\lambda^b)$  that is associated to  $p_1$  can be determined using the radiance measured at a white diffuser pixel  $p_{WD} \in \mathcal{WD}$ . We can rewrite Eq. (3.12) at  $p_{WD}$  as:

$$I_{p_{WD}}^b = \tau \cdot E_{t_{p_{WD}}^b}(\lambda^b) \cdot R_{p_{WD}}(\lambda^b) \cdot A_{p_{WD}}(\lambda^b), \quad (3.14)$$

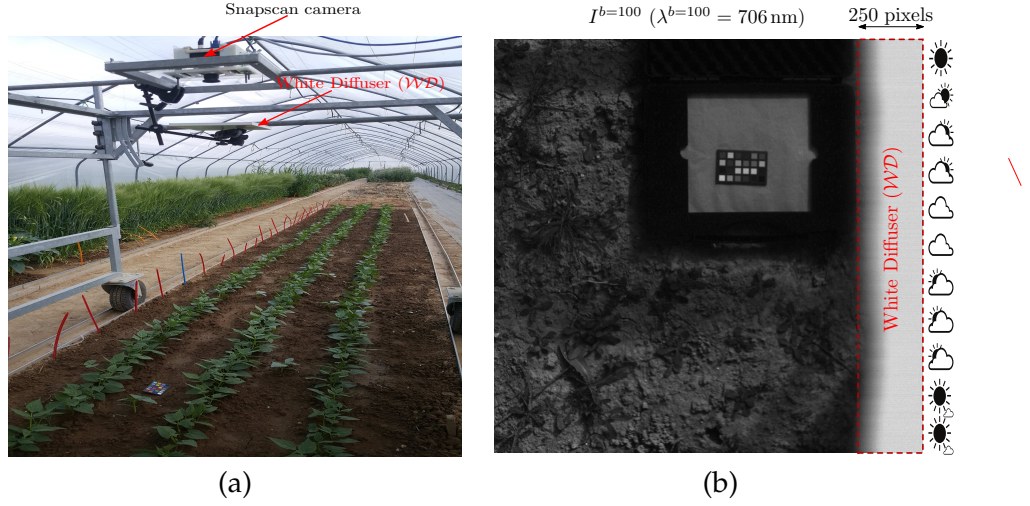


FIGURE 3.7: (a) Acquisition system. The Snapscan camera is mounted on the top of it and observes from nadir a portion of the white diffuser  $WD$  and the scene. (b) Channel of a radiance image with white diffuser along its right border.

where  $R_{p_{WD}}(\lambda^b) = \rho_{wd}$  is the diffuse reflection factor of the white diffuser. Since  $p_1$  and  $p_{WD}$  are located on the same row,  $t_{p_1}^b = t_{p_{WD}}^b$  and the spatial uniformity assumption stipulates that  $E_{t_{p_1}^b}(\lambda^b) = E_{t_{p_{WD}}^b}(\lambda^b)$ . Therefore Eq. (3.14) can be rewritten as:

$$E_{t_{p_1}^b}(\lambda^b) \cdot A_{p_{WD}}(\lambda^b) = \frac{I_{p_{WD}}^b}{\tau \cdot \rho_{wd}}, \quad (3.15)$$

where  $\frac{I_{p_{WD}}^b}{\tau \cdot \rho_{wd}}$  can be considered as an estimation of the illumination condition associated to pixel  $p_1$ . However, since the optical attenuation  $A_{p_1}(\lambda)$  at  $p_1$  is different from that of  $p_{WD}$ , we cannot replace  $A_{p_{WD}}(\lambda)$  by  $A_{p_1}(\lambda)$  in Eq. (3.15). Thus, to accurately estimate the reflectance, we propose to correct the vignetting effect at each pixel  $p$  of the radiance image  $\mathbf{I}^{(B)}$  so that optical attenuation is compensated.

### 3.4.2 Vignetting correction

Vignetting effect refers to a loss in intensity values from the image center to its borders due to the geometry of the sensor optics. This phenomenon can be well observed on the image  $\mathbf{I}^{(B)}[WD]$  of a uniformly illuminated white diffuser, as shown in Fig. 3.8.

Methods of vignetting correction can be divided into two groups: physically-based ones (with anti-vignetting optical filters for example) and soft methods (image-based approaches, modeling the optical pathway) [88, 93, 190]. Image-based methods are

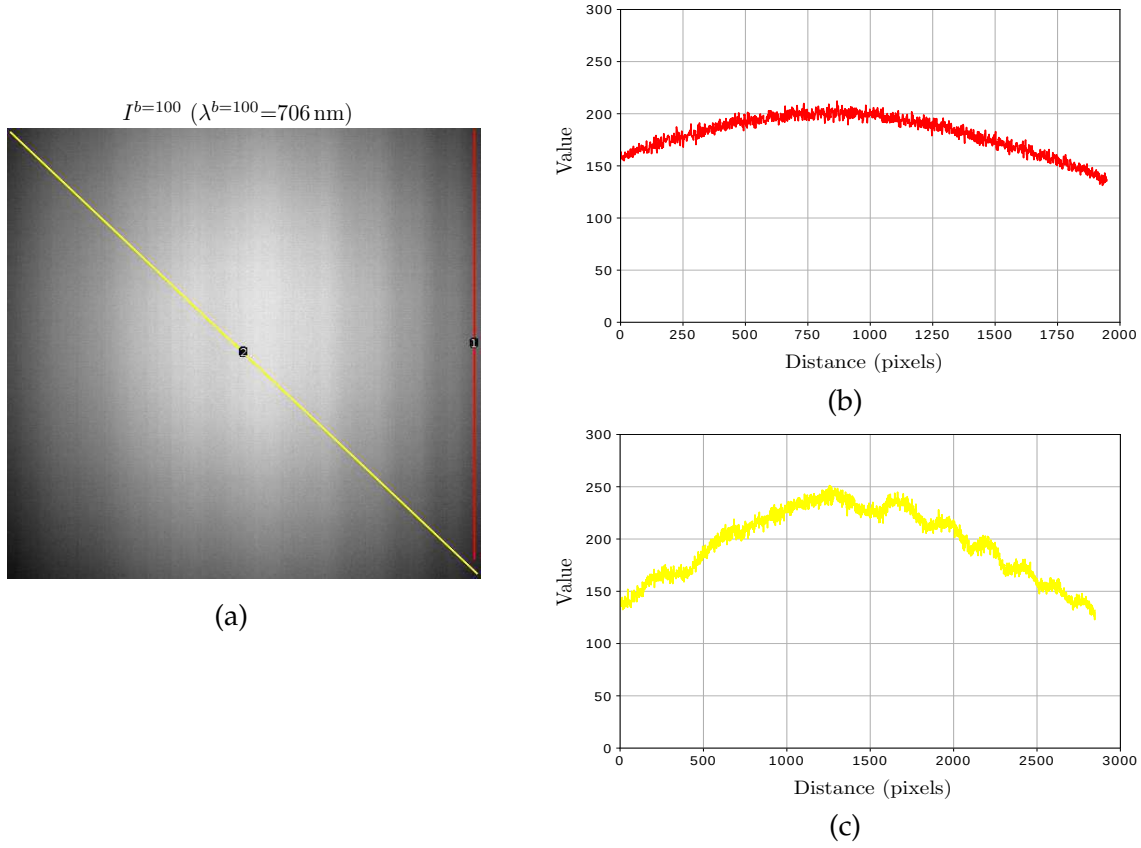


FIGURE 3.8: Vignetting effect. (a) Channel of white diffuser radiance image. (b), (c) Profile of pixel values over the red vertical and yellow diagonal lines on image (a), respectively.

easier to implement, and are generally performed by computing a correction factor for each pixel [190]. In order to compensate for the spatial variation of  $A_p(\lambda^b)$ , we choose to use this approach because it requires no prior knowledge about the optical system behavior. To generate correction factors, one first acquires a  $B$ -channel multispectral radiance image  $\mathbf{I}^{(B)}[\text{WD}]$  of a white diffuser under a spatially uniform illumination (like the one described in Sec. 3.3.1.2) and in a dark room to avoid ambient light interference. The correction factor is channel-wise and pixel-wise computed as:

$$C_p^b = \frac{I^b[\text{WD}]}{I_p^b[\text{WD}]}, \quad (3.16)$$

where  $I^b[\text{WD}]$  is the median value of the  $m$  pixels with highest values over  $I^b[\text{WD}]$  ( $m = 11$  in our experiments), which discards saturated or defective pixel values. The correction factors are stored in a  $B$ -channel multispectral image denoted as  $\mathbf{C}^{(B)}$ .

Because  $\mathbf{C}^{(B)}$  is deduced from a single white diffuser image, it would be corrupted

by noise (even after thermal noise removal during the frame acquisitions). One simple method is to acquire several white diffuser images to compute correction factors on the averaged image. However, this only considers the temporal noise related to the acquisition that affects each pixel, but neglects spatial noise that is more important (see red profile in Fig. 3.9). Thus, we propose to directly denoise  $\mathbf{C}^{(B)}$  by convolving each of its channels  $C^b$  with an  $11 \times 11$  averaging filter  $\mathcal{H}$ :

$$\tilde{C}^b = C^b * \mathcal{H}. \quad (3.17)$$

The vignetting effect in the  $B$ -channel radiance image  $\mathbf{I}^{(B)}$  is corrected pixel-wise and channel-wise using the smoothed correction factors:

$$\tilde{I}_p^b = I_p^b \cdot \tilde{C}_p^b, \quad (3.18)$$

where  $I_p^b$  and  $\tilde{I}_p^b$  are the intensity values before and after vignetting correction. This procedure should reduce noise while preserving image textures. Figure 3.9 illustrates a case of spatial correction. We can see that vignetting effect is compensated whether  $\mathbf{C}^{(B)}$  is smoothed or not. But, smoothed correction factors produce a spatially corrected image  $\tilde{\mathbf{I}}^{(B)}$  that is less affected by noise. Indeed, the standard deviation of corrected radiance values over the yellow line is 9.1 without smoothing but is decreased to 6.9 with smoothed factors. In Fig. 3.9(b), we can also observe that spatially corrected radiance image  $\tilde{\mathbf{I}}^{(B)}$  presents a slight shift of values along the green (or red) profile plot due to illumination variation during the frame acquisitions. However, we can assume that the attenuation is spatially uniform after vignetting correction (i.e.,  $A_p(\lambda^b) \cdot \tilde{C}_p^b = \alpha^b \in \mathbb{R}$  for any given channel index  $b$  and pixel  $p$ ), such that each value of the vignetting-free radiance image is expressed from Eq. (3.12) as:

$$\tilde{I}_p^b = \tau \cdot R_p(\lambda^b) \cdot E_{t_p^b}(\lambda^b) \cdot \alpha^b. \quad (3.19)$$

### 3.4.3 Row-wise (rw) based reflectance estimation

The underlying assumption is that illumination is spatially uniform over each row at both the white diffuser and scene pixels (that may be not verified in the case of shadows). Based on this row uniformity assumption for illumination, we estimate



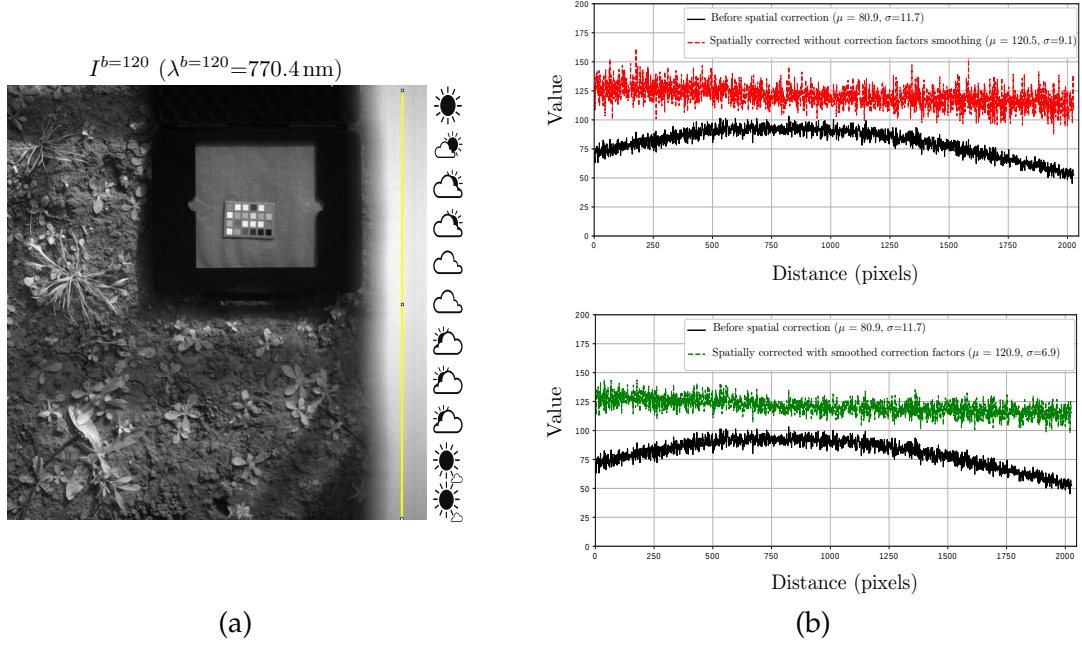


FIGURE 3.9: (a) Multispectral image channel with the observed scene and a portion of a white diffuser. (b) Intensity values of the pixels under the yellow line before (in black) and after spatial correction using correction factor smoothing (in green) or not (in red).

reflectance from  $\tilde{\mathbf{I}}^{(B)}$  in a row-wise manner, as follows. At pixel  $p$  with spatial coordinates  $x_p$  and  $y_p$ , Eq. (3.19) can be rewritten as:

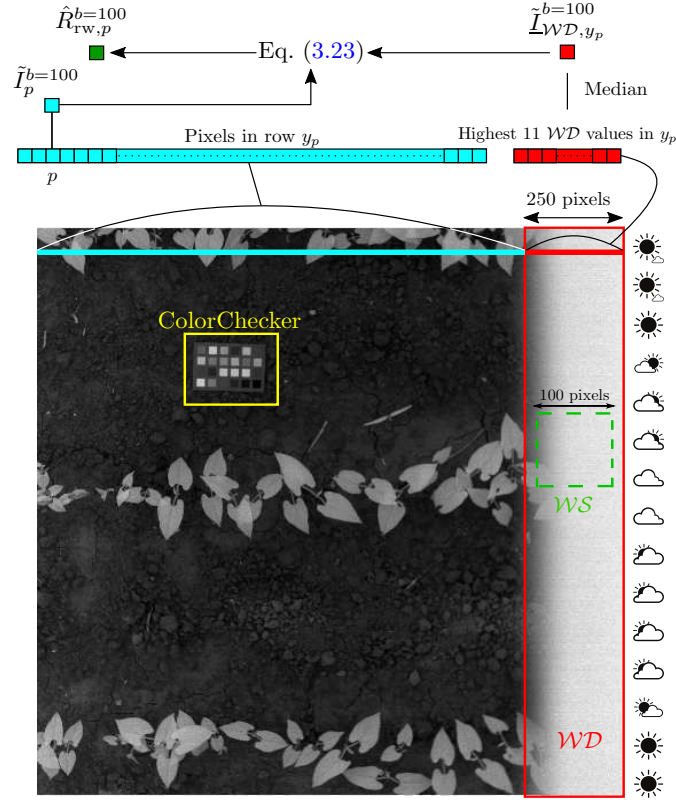
$$R_p(\lambda^b) = \frac{\tilde{I}_p^b}{\tau \cdot E_{t_{y_p}^b}(\lambda^b) \cdot \alpha^b}. \quad (3.20)$$

To determine the illumination  $E_{t_{y_p}^b}(\lambda^b)$  that is associated to the row of  $p$  for channel index  $b$ , we use a white diffuser pixel  $r_{WD} \in \mathcal{WD}$  located on the same row as  $p$ . At  $r_{WD}$ , the reflectance is equal to the white diffuser reflection factor  $\rho_{wd}$ , and Eq. (3.19) provides the vignetting-free radiance as:

$$\tilde{I}_{r_{WD}}^b = \tau \cdot \rho_{wd} \cdot E_{t_{r_{WD}}^b}(\lambda^b) \cdot \alpha^b. \quad (3.21)$$

Because  $p$  and  $r_{WD}$  are located on the same row,  $t_{y_p}^b = t_{r_{WD}}^b$  and  $E_{t_{y_p}^b}(\lambda^b) = E_{t_{r_{WD}}^b}(\lambda^b)$  according to the assumption regarding the spatial uniformity over each row. Therefore, Eq. (3.21) can be rewritten as:

$$E_{t_{y_p}^b}(\lambda^b) = \frac{\tilde{I}_{r_{WD}}^b}{\tau \cdot \rho_{wd} \cdot \alpha^b}, \quad (3.22)$$

FIGURE 3.10: rw-based reflectance estimation (from channel  $\tilde{I}^{b=100}$ ).

which can be considered to be an estimation of the illumination that is associated to pixel  $p$ . For robustness sake, we propose to compute it from the median value  $\tilde{I}_{WD,y_p}^b$  of the  $m$  highest pixel values<sup>17</sup> that represent the white diffuser subset  $WD$  in  $y_p$ , rather than from a single value  $\tilde{I}_{r_{WD}}^b$ . Plugging Eq. (3.22) into (3.20) yields our row-wise (rw) reflectance estimation at pixel  $p$  for channel index  $b$ :

$$\hat{R}_{rw,p}^b = \rho_{wd} \cdot \frac{\tilde{I}_p^b}{\tilde{I}_{WD,y_p}^b}. \quad (3.23)$$

As it was done to obtain the reference reflectance  $\hat{\mathbf{R}}_{\text{ref}}^{(B)}$  (see Eqs. (3.9) and (3.10)), the  $B$ -channel reflectance image  $\hat{\mathbf{R}}_{rw}^{(B)}$  is spectrally corrected to provide a  $K$ -channel reflectance image  $\hat{\mathbf{R}}_{rw}^{(K)}$  that is afterwards freed from negative values (see Eq. (3.10)).

<sup>17</sup> In practice, setting  $m = 11$  pixels is a good compromise for accurately estimating the illumination for each row and each channel.

### 3.4.4 Optimized row-wise (orw) based reflectance estimation

We propose to optimize the rw-based reflectance estimation thanks to the patches of the GretagMacbeth<sup>TM</sup> ColorChecker that is present in each of our images. The so-called orw method can be seen as a hybrid method that combines illumination and training-based principles. The  $K$ -channel rw-based reflectance image  $\hat{\mathbf{R}}_{\text{rw}}^{(K)}$  is first estimated by Eq. (3.23), then optimized according to correction factors computed by least square regression of learning patch reflectance.

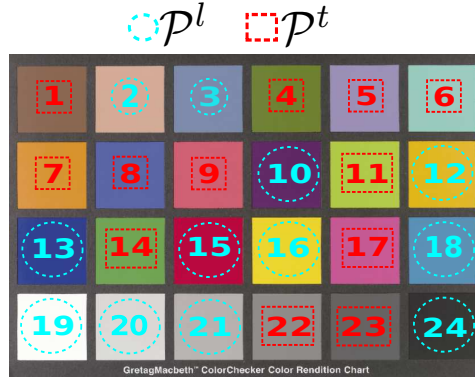
Assuming that reflectance across a patch of the ColorChecker is spatially uniform, each patch  $\mathcal{P}_j$  ( $j = 1..24$ ) is represented as a  $K$ -dimensional reflectance vector  $\hat{\mathbf{R}}_{\mathcal{P}_j}^{(K)}$ . Each coordinate of this vector is associated to a channel index  $k$  and is computed as the average value for this channel over all the patch pixels:

$$\hat{R}_{\mathcal{P}_j}^k = \frac{1}{|\mathcal{P}_j|} \sum_{p \in \mathcal{P}_j} \hat{R}_p^k, \quad (3.24)$$

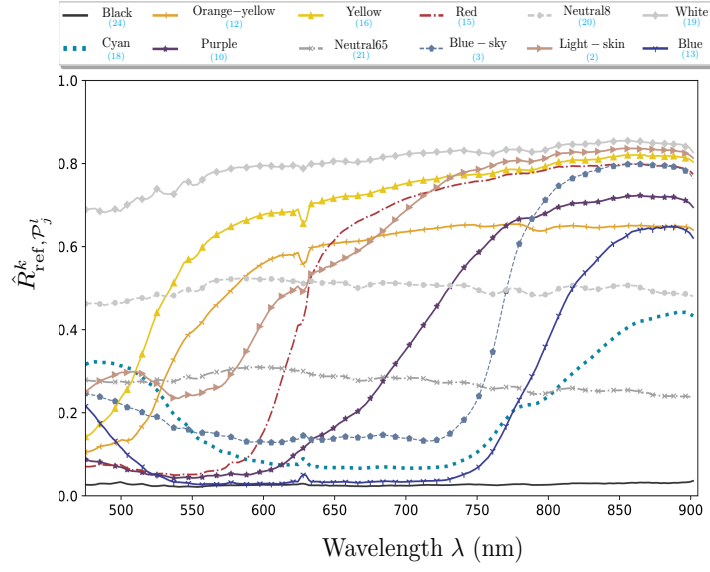
where  $|\mathcal{P}_j|$  is the number of pixels characterizing the considered patch, and  $\hat{\mathbf{R}}^{(K)}$  a given  $K$ -channel reflectance image.

Among the 24 color patches of the Macbeth ColorChecker chart, we use a set  $\mathcal{P}^l$  of 12 patches for the learning procedure and a set  $\mathcal{P}^t$  of 12 patches (used in the next chapter) for testing the estimated reflectance quality (see Fig. 3.11(a)). The patches of  $\mathcal{P}^l$  are selected using an exhaustive search. We have tested all the 2,704,156 combinations and retained the combination that provides the lowest reflectance estimation error.

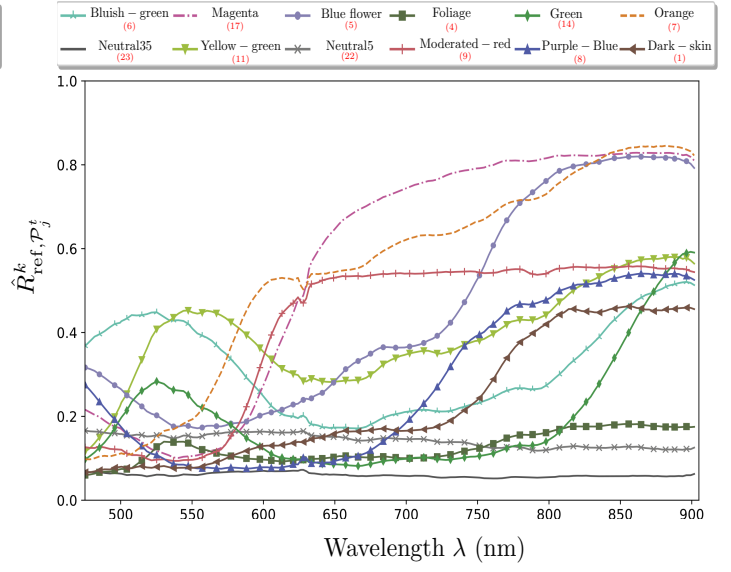
From the reference reflectance image  $\hat{\mathbf{R}}_{\text{ref}}^{(K)}[\text{CC}]$  (see Fig. 3.6(a)) estimated by Eqs. (3.8)–(3.10), we compute the reference reflectance value  $\hat{R}_{\text{ref}, \mathcal{P}_j^l}^k[\text{CC}]$  for each learning patch  $\mathcal{P}_j^l$  and channel  $k$  according to Eq. (3.24), that we consider as the target output. The reflectance  $\hat{R}_{\text{rw}, \mathcal{P}_j^l}^k$  of each learning patch  $\mathcal{P}_j^l$  within each channel  $k$  of the rw-based reflectance image is considered as the predicted output.



(a) ColorChecker with learning and test patch sets.



(b) Reflectance spectra of learning patches.



(c) Reflectance spectra of test patches.

FIGURE 3.11: (a) ColorChecker patch numbers and (b, c) reference reflectance spectra.

We can thus define the following relationship:

$$\begin{bmatrix} \hat{R}_{\text{ref}, \mathcal{P}_1^l}^k [\text{CC}] \\ \cdot \\ \cdot \\ \cdot \\ \hat{R}_{\text{ref}, \mathcal{P}_{12}^l}^k [\text{CC}] \end{bmatrix} = \hat{b}_0^k + \hat{b}_1^k \cdot \begin{bmatrix} \hat{R}_{\text{rw}, \mathcal{P}_1^l}^k \\ \cdot \\ \cdot \\ \cdot \\ \hat{R}_{\text{rw}, \mathcal{P}_{12}^l}^k \end{bmatrix}, \quad (3.25)$$

where  $\hat{b}_0^k$  is the bias coefficient and  $\hat{b}_1^k$  is the scale factor. Least square regression leads to the following values:

$$\hat{b}_1^k = \frac{\sum_{j=1}^{12} (\hat{R}_{\text{rw}, \mathcal{P}_j^l}^k - \mu_{\text{rw}}^k) (\hat{R}_{\text{ref}, \mathcal{P}_j^l}^k [\text{CC}] - \mu_{\text{ref}}^k)}{\sum_{j=1}^{12} (\hat{R}_{\text{rw}, \mathcal{P}_j^l}^k - \mu_{\text{rw}}^k)^2} \quad (3.26)$$

and

$$\hat{b}_0^k = \mu_{\text{ref}}^k - \hat{b}_1^k \cdot \mu_{\text{rw}}^k, \quad (3.27)$$

where  $\mu_{\text{ref}}^k$  and  $\mu_{\text{rw}}^k$  are the means of all reflectance values over the 12 training patches for channel index  $k$  of the reference and rw-based reflectance images, respectively.  $\hat{b}_0^k$  and  $\hat{b}_1^k$  are then used to optimize the reflectance estimated by rw method. Thus, the reflectance value  $\hat{R}_{\text{orw}, p}^k$  estimated for channel index  $k$  at pixel  $p$  is given by:

$$\hat{R}_{\text{orw}, p}^k = \hat{b}_0^k + \hat{b}_1^k \cdot \hat{R}_{\text{rw}, p}^k. \quad (3.28)$$

Finally, the reflectance image  $\hat{\mathbf{R}}_{\text{orw}}^{(K)}$  undergoes negative value removal (see Eq. (3.10)). Figure 3.12 outlines the process of reflectance estimation from a  $B$ -channel radiance image  $\mathbf{I}^{(B)}$  using the proposed rw and orw methods.

### 3.5 Conclusion

To compute features that are invariant to illumination conditions, we estimate reflectance in this chapter. Our proposed image formation model (see Eq. (2.10)) shows that varying illumination during the frame acquisitions can alter classical reflectance estimation. Indeed, reflectance estimation methods usually assume constant illumination conditions during the multispectral image acquisition. Therefore, they may be inappropriate to estimate reflectance from radiance images acquired by a multi-shot camera such as the Snapscan. We first highlight the effect of vignetting induced by the sensor optics on reflectance computation, and perform vignetting correction based on a correction factor computed for each pixel in each channel. Then, we propose a method that exploits a portion of a white diffuser to estimate reflectance in a row-wise manner, such that illumination variation is considered at the frame level. Furthermore, we propose a method that combines illumination and training-based principles to refine reflectance estimation.

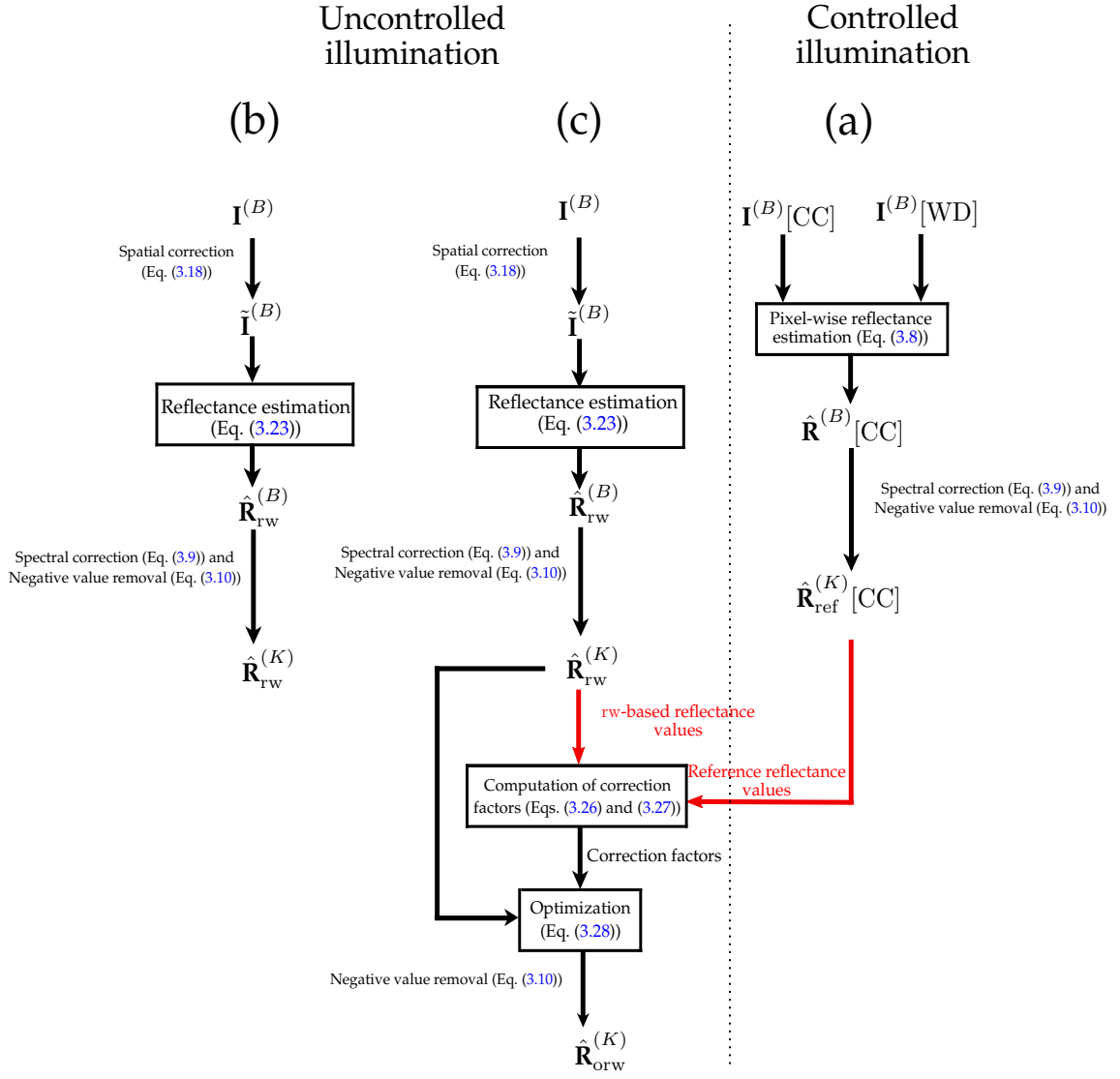


FIGURE 3.12: Flowchart of rw and orw methods. (a) Computation of the reference reflectance image  $\hat{R}_{ref}^{(K)}$  with controlled illumination condition. (b) and (c) rw and orw-based reflectance estimation in uncontrolled illumination conditions.

The next chapter focuses on the assessment of reflectance estimation methods. We compare the performance of our proposed methods against state-of-the-art ones. We also evaluate the contribution of the reflectance features provided by each method to the crop/weed detection and identification problems.



# Reflectance estimation assessment

## Contents

---

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| <b>4.1 Introduction</b>                                                    | 73 |
| <b>4.2 Multispectral vegetation databases</b>                              | 73 |
| <b>4.3 Multispectral dataset acquisition</b>                               | 74 |
| 4.3.1 Experimental setup                                                   | 74 |
| 4.3.2 Vegetation pixel extraction and labelling                            | 77 |
| <b>4.4 State-of-the-art methods adapted to the Snapscan</b>                | 77 |
| 4.4.1 Illumination-based reflectance estimation                            | 78 |
| 4.4.2 Training-based reflectance estimation                                | 80 |
| <b>4.5 Estimated reflectance fidelity</b>                                  | 83 |
| 4.5.1 Metrics                                                              | 83 |
| 4.5.2 Results                                                              | 84 |
| 4.5.3 Reflectance estimated at vegetation pixels                           | 90 |
| 4.5.4 Computation cost                                                     | 92 |
| <b>4.6 Reflectance features for beet/weed detection and identification</b> | 93 |
| 4.6.1 Learning and test pixel extraction                                   | 93 |
| 4.6.2 Supervised classification                                            | 94 |
| 4.6.3 Supervised classification by boosting approach                       | 97 |
| 4.6.4 Evaluation metrics                                                   | 98 |



|            |                                      |            |
|------------|--------------------------------------|------------|
| 4.6.5      | Classification experiments . . . . . | 99         |
| <b>4.7</b> | <b>Conclusion . . . . .</b>          | <b>105</b> |

---

## 4.1 Introduction

**S**PECIFICALLY, we need a database of images to assess reflectance-based crop/weed recognition systems. Preparing a campaign of vegetation image acquisition is time-consuming and tedious because crops and weeds must be first planted in the field, then maintained healthy throughout the entire acquisition period to prevent additional variations of their spectral signatures caused by diseases or a hydric stress. Furthermore, the acquisition itself may be challenging and the quality of acquired images be unsatisfying in outdoor conditions where several external variations (such as illumination, wind, etc.) are uncontrolled. In this chapter, we assess the performance reached by reflectance estimation methods in the agricultural context. The contribution of reflectance features for supervised crop/weed detection and identification is evaluated.

Section 4.2 presents an overview of some public multispectral vegetation databases dedicated to crop/weed control applications. Because there is no public database that contains high spectral resolution images of the plants of interest, we acquire our own. Details about the acquisition procedure are provided in Sec. 4.3. In Sec. 4.4, we present the considered state-of-art methods used to estimate reflectance. Then in Sec. 4.5, we compare their performance to that of our proposed methods.

## 4.2 Multispectral vegetation databases

Most of studies about weed detection use their own multispectral images due to the lack of available public datasets. Among the few available ones [112], the *Carrots 2017* and *Onions 2017* datasets<sup>18</sup> [23] of RGB-NIR images have been acquired in the fields of Lincolnshire, UK, in June 2017 using two cameras (Teledyne DALSA Genie Nano). The *WeedMap*<sup>19</sup> [154] and *weedNet*<sup>20</sup> [153] datasets are acquired by a UAV equipped with two sensors, one that acquires channels in the VIS domain and the

<sup>18</sup> <https://lcas.lincoln.ac.uk/wp/research/data-sets-software/crop-vs-weed-discrimination-dataset>

<sup>19</sup> <https://projects.asl.ethz.ch/datasets/doku.php?id=weedmap:remotesensing2018weedmap>

<sup>20</sup> <https://github.com/inkyusa/weedNet>

second one in the NIR domain. RGB-NIR images of the *Sugar Beets* dataset<sup>21</sup> [28] have been acquired in a sugar beet farm near Bonn, Germany, in spring 2016 using the autonomous field robot *BoniRob* [28] equipped with a prism-based 2-charge-coupled device (CCD) multispectral camera (JAI AD-130GE). The same devices have been used to acquire the *CWFID* dataset<sup>22</sup> [71] under controlled illumination in an organic carrot farm located in Northern Germany in 2013. The *Ladybird Cobbitty 2017 Brassica* dataset<sup>23</sup> [17] has been acquired using a multispectral linescan camera (Resonon Pika XC2) that samples 447 spectral bands in the range 400–1000 nm. The targeted cauliflower and broccoli plants have been planted at Lansdowne Farm, Cobbitty, a suburb 70 km southwest of Sydney, Australia.

These datasets mostly contain crop/weed images with RGB channels and an additional channel associated to a NIR band. The Ladybird Cobbitty 2017 Brassica dataset contains images with a high number of channels, but targets other plant species than those considered in this study. Because no publicly available dataset contains the crops (beet, wheat, and bean) and weeds (thistle, goosefoot, and datura) we aim to analyze, we acquire our own vegetation database.

### 4.3 Multispectral dataset acquisition

This section presents our multispectral vegetation database. In Sec. 4.3.1, we present the experimental setup used for outdoor image acquisition, then we explain how vegetation pixels are extracted and labelled in Sec. 4.3.2.

#### 4.3.1 Experimental setup

An image acquisition campaign has been conducted in a greenhouse under skylight (see Fig. 4.1(a)) by the Chambre d’Agriculture de la Somme in early April 2019. The targeted plants are beet, wheat, and bean for crops, and datura, thistle, and goosefoot for weeds. These plants have been planted in a greenhouse located near Amiens, France, that contains 80 micro-plots, 36 of which are used for implantations and 44 are reserved for future settlements. The images have been acquired at different dates of May and June 2019, and different day times (see Fig. 4.1(b)). The camera

<sup>21</sup> <http://www.ipb.uni-bonn.de/data/sugarbeets2016/>

<sup>22</sup> <https://github.com/cwfid/dataset>

<sup>23</sup> <https://researchdata.edu.au/ladybird-cobbitty-2017-brassica-dataset/1370660>

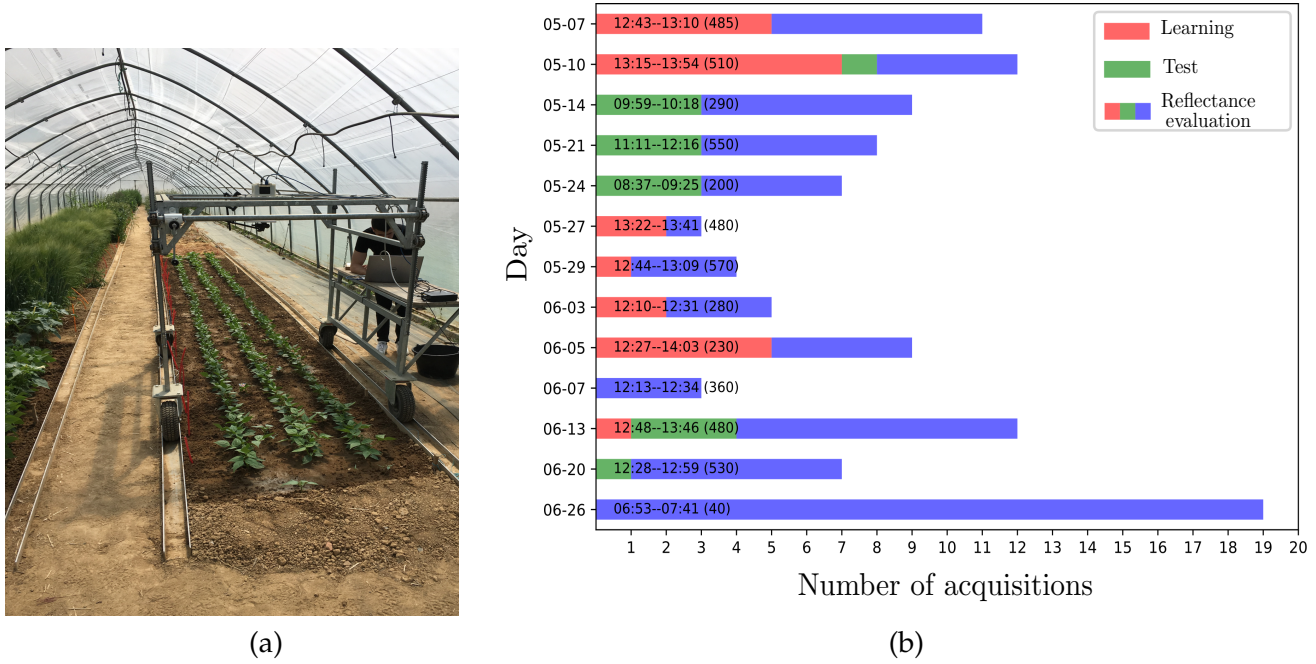


FIGURE 4.1: (a) Our experimental site and apparatus for vegetation image acquisitions. (b) Acquisition dates and times of the 109 images for the 2019 campaign. The text along each bar gives the acquisition time range and a coarse estimation of global solar irradiance ( $\text{W}\cdot\text{m}^{-2}$ ) at the median acquisition time in parentheses [1]. The images used to assess supervised beet and weed detection/identification (see Sec. 4.6) are shown in red and green, other images in blue. All images are used to assess the estimated reflectance quality. Series are stacked for readability and their order is not meaningful in regards to any acquisition time order.

is mounted on top of the acquisition system (see Fig. 3.7 (b)) at a distance of about 1.50 m from the ground. The white diffuser is mounted on the acquisition system so that the camera vertically observes a portion of it, as explained in Sec. 3.4. Among the acquired images, we distinguish “single-species” images that represent only one type of plant, and “mixed” images where several (at least two) plants are present. This campaign provided  $U = 109$  radiance images of  $2048 \times 2048$  pixels  $\times$  192 channels of 10 bit depth. The memory size of each image stored in ENVI format is 3.2 GB. These images represent the observed crop (beet, wheat, and bean) and weed (thistle, goosefoot, and datura) species at different growth stages (see Fig. 4.2). The phenological development of plants is generally described by the *BBCH-scale* [120] that categorizes plant development cycle into ten principal growth stages, starting from the germination stage to the full maturity stage. Analyzing each crop and weed species at each of these ten growth stages is not experimentally possible to us, neither it is our main concern. Our database contains images of several growth stages



FIGURE 4.2: Illustration of nine multispectral images from our database rendered as RGB under D65 illuminant, that show the considered plant species. Each image contains a white diffuser on its right border and a ColorChecker chart. Dots show weeds: thistle in blue (a, b, c, f), goosefoot in cyan (d, e), and datura in magenta (h, i). Other plants are crop: beet (a, b, c, d, e), wheat (f, g), and bean (h, i).

for each species (see Fig. 4.2(e, i)). Therefore, we categorize plant growth according to the size and number of leaves into two stages only: youth stage (development of 2 to 9 leaves to stem elongation) and mature stage where the plant has fully evolved. The database also contains images where crop and weed plants strongly overlap (see Fig. 4.2(f)) or are shaded (see Fig. 4.2(a)), which makes crop/weed recognition challenging.

All images contain a GretagMacbeth<sup>TM</sup> ColorChecker that is principally used to assess the quality of reflectance estimation. Figure 4.2 shows an RGB rendering of nine multispectral images under D65 illuminant from the database acquired by the Snapscan camera.



### 4.3.2 Vegetation pixel extraction and labelling

Only vegetation pixels are analyzed because we aim to detect/identify crops and weeds. They are distinguished from the background (white diffuser, ColorChecker, and soil pixels) using the normalized difference vegetation index (NDVI) [172]. We compute the NDVI values from the rw-based reflectance image  $\hat{\mathbf{R}}_{\text{rw}}^{(K)}$ , since the rw method considers illumination variation, but the images provided by any other reflectance estimation method should yield similar vegetation pixel detection results. We consider  $p$  to be a vegetation pixel if its NDVI value is greater than a threshold  $\gamma$ :

$$\frac{\hat{R}_{\text{rw},p}^{139} - \hat{R}_{\text{rw},p}^{67}}{\hat{R}_{\text{rw},p}^{139} + \hat{R}_{\text{rw},p}^{67}} \geq \gamma, \quad (4.1)$$

with the Snapscan “virtual” band centers  $\lambda^{67} = 678.2 \text{ nm}$  and  $\lambda^{139} = 899.2 \text{ nm}$ . Setting  $\gamma = 0.45$  experimentally provides a good trade-off between under- and over-segmentation of vegetation pixels. Noisy vegetation pixels are filtered out as much as possible by morphological opening. The vegetation pixels are then manually labelled by an expert in agronomy to build the segmentation ground truth for each multispectral image. Table 4.1 displays the number of pixels for each considered class.

TABLE 4.1: Number of pixels per class.

| Class        | Crop/weed label                                                                     | Class-wise label                                                                    | #Pixels            |
|--------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------|
| Beet         |  |  | 22,503,437         |
| Wheat        |                                                                                     |  | 16,497,666         |
| Bean         |                                                                                     |  | 23,481,321         |
| Thistle      |                                                                                     |  | 19,251,028         |
| Goosefoot    |  |  | 32,577,057         |
| Datura       |                                                                                     |  | 10,877,044         |
| <b>Total</b> | -                                                                                   | -                                                                                   | <b>125,187,553</b> |

## 4.4 State-of-the-art methods adapted to the Snapscan

In this section, we present the state-of-the-art reflectance estimation methods that we adapt to the images acquired by the Snapscan camera. We consider four methods: three ones based on illumination estimation (see Sec. 4.4.1), and one based on training from the ColorChecker patches (see Sec. 4.4.2).

#### 4.4.1 Illumination-based reflectance estimation

The classical white-average (wa) method [49, 86] uses the surface of a white diffuser fully included in the scene to estimate the illumination for each channel. We adopt the wa implementation by considering a subset  $\mathcal{WS}$  of  $\mathcal{WD}$  (see dashed green square in Fig. 4.3). Reflectance is estimated at each image pixel by the wa method as:

$$\hat{R}_{\text{wa},p}^b = \rho_{\text{wd}} \cdot \frac{\tilde{I}_p^b}{\frac{1}{|\mathcal{WS}|} \sum_{s \in \mathcal{WS}} \tilde{I}_s^b}, \quad (4.2)$$

where  $|\cdot|$  is the set cardinal.

Similarly, the max-spectral (ms) method [85] assumes that the pixel with maximum value within each channel can be considered to be a white diffuser pixel for estimating the illumination. While ignoring the diffuse reflection factor, reflectance is estimated at each pixel in each channel by the ms method, as:

$$\hat{R}_{\text{ms},p}^b = \frac{\tilde{I}_p^b}{\max_{s \in X} \tilde{I}_s^b}, \quad (4.3)$$

where  $X$  contains all of the image pixels, except  $\mathcal{WD}$ , and those of the ColorChecker. The wa and ms-based  $B$ -channel reflectance images undergo spectral correction and negative value removal (see Eqs. (3.9) and (3.10)) to provide the final  $K$ -channel reflectance images  $\hat{\mathbf{R}}_{\text{wa}}^{(K)}$  and  $\hat{\mathbf{R}}_{\text{ms}}^{(K)}$ .

The double white diffuser (dwd) method described in [44] uses two images to both eliminate vignetting effect and estimate reflectance in a pixel-wise manner as in Eq. (3.8). The first image is the full-field white diffuser image  $\mathbf{I}^{(B)}[\text{WD}]$  and the second one characterizes the scene of interest and contains two white diffusers (one covers the bottom image border while the second is fully visible).

Our images contain a single white diffuser on the right image border that is used by rw and orw methods to estimate reflectance. The dwd method requires a second white diffuser fully included in the scene to be more robust against spatially non uniform illumination. The ColorChecker white patch ( $\mathcal{WP}$ ) (see Fig. 4.3) is present in each of our images and exhibits optical properties that are very similar to that of a white diffuser. We therefore consider it as the second white diffuser to adapt the dwd method to our images. The dwd method estimates reflectance thanks to three

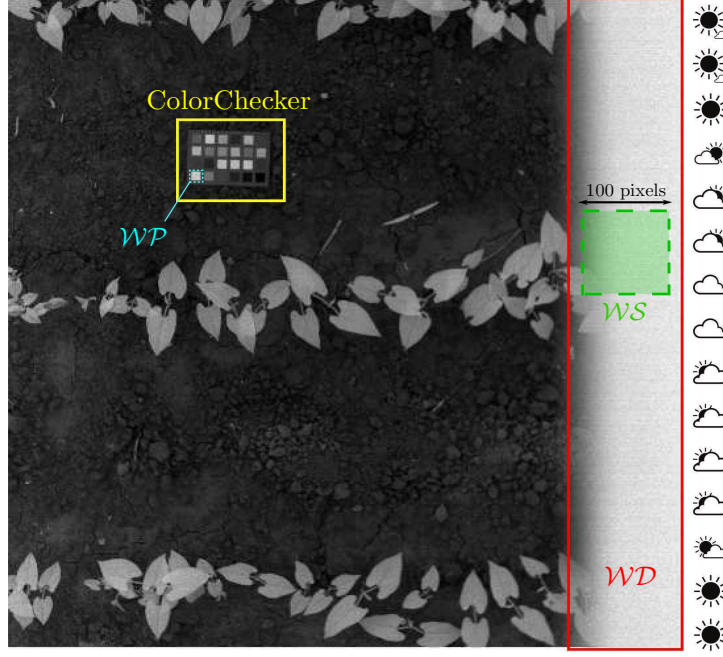


FIGURE 4.3: White diffuser subset ( $\mathcal{WS}$ ) and the white patch subset ( $\mathcal{WP}$ ) used respectively by wa and dwd methods for reflectance estimation.

successive steps, that are adapted to our image contents as follows:

- Following Eq. (3.8) but neglecting integration times and diffuse reflection factor, a coarse reflectance estimation is first computed as:

$$\tilde{R}_p^b = \frac{I_p^b}{I_p^b[\text{WD}]}. \quad (4.4)$$

Like in Section 3.3.2, this step aims to compensate the vignetting effect by a pixel-wise division of values  $I_p^b[\text{WD}]$  associated to the full-field white diffuser and  $I_p^b$  associated to the scene. Note that a full-field white diffuser image  $\mathbf{I}^{(B)}[\text{WD}]$  is acquired before each image acquisition.

- Because the illumination associated to  $\mathbf{I}^{(B)}[\text{WD}]$  is different from that of the scene image  $\mathbf{I}^{(B)}$ ,  $\tilde{\mathbf{R}}^{(B)}$  is rescaled row-wise at each pixel  $p$  as:

$$\tilde{R}_p^{tb} = \tilde{R}_p^b \cdot \alpha_{y_p}^b, \quad (4.5)$$



where the illumination scaling factor  $\alpha_{y_p}^b$  is computed at the row  $y_p$  of  $p$  as:

$$\alpha_{y_p}^b = \frac{\bar{I}_{\mathcal{WD}, y_p}^b[\text{WD}]}{\bar{I}_{\mathcal{WD}, y_p}^b}. \quad (4.6)$$

Each term in this equation is the average value over the row of  $p$  within the white diffuser subset  $\mathcal{WD}$  in channel  $I^b$  of either the full-field white diffuser image or the scene image.

- Finally, the values of  $\tilde{\mathbf{R}}^{(B)}$  are normalized channel-wise to provide the dwd reflectance estimation as:

$$\hat{R}_{\text{dwd}, p}^b = \tilde{R}_p^b \cdot \frac{\rho_{\mathcal{WP}}^b}{\beta_{\mathcal{WP}}^b}, \quad (4.7)$$

where  $\beta_{\mathcal{WP}}^b$  is the average value over the white patch subset  $\mathcal{WP}$  in channel  $\tilde{R}^b$ , and  $\rho_{\mathcal{WP}}^b$  is the diffuse reflection factor of the white patch for the spectral band centered at  $\lambda^b$  measured by a spectroradiometer in laboratory.

The  $B$ -channel reflectance image  $\hat{\mathbf{R}}_{\text{dwd}}^{(B)}$  undergoes spectral correction and negative value removal (see Eqs (3.9) and (3.10)) to provide the final  $K$ -channel reflectance image  $\hat{\mathbf{R}}_{\text{dwd}}^{(K)}$ .

Like rw method, dwd accounts for illumination variation during the frame acquisition. It attempts to use the properties of a full-field white diffuser image acquired in outdoor to both estimate reflectance and eliminate vignetting effect pixel-wise. Illumination scaling and reflectance normalization are required because the illuminations associated to the object and full-field white diffuser images are different. In contrast, rw method is more straightforward. It uses the same correction factors (stored in the  $B$ -channel image  $\tilde{\mathbf{C}}^{(B)}$ , see Sec. 3.4.2) to spatially correct vignetting effect in all acquired radiance images before estimating reflectance in a row-wise approach.

#### 4.4.2 Training-based reflectance estimation

We also apply the linear Wiener (wn) estimation technique [166] to estimate reflectance. It is based on matrix  $\mathbf{G}$  that transforms radiance spectra into reflectance. From any radiance image  $\mathbf{I}^{(B)}$  in the database, we compute the spectrally-corrected

vignetting-free radiance image  $\tilde{\mathbf{I}}^{(K)}$  while using Eqs. (3.9) and (3.18), and then estimate the  $K$ -channel reflectance image as:

$$\hat{\mathbf{R}}_{\text{wn},p}^{(K)} = \mathbf{G} \cdot \tilde{\mathbf{I}}_p^{(K)}. \quad (4.8)$$

To compute  $\mathbf{G}$ , we use the spectra of the ColorChecker learning patches ( $\mathcal{P}^l$  subset, see Fig. 3.11(a)) that are represented in each of our images. The estimation matrix  $\mathbf{G}$  that is associated to each input radiance image is determined as:

$$\mathbf{G} = \mathbf{T}_{\text{ref}} \cdot \mathbf{T}_{\text{rad}}^{\top} \left( \mathbf{T}_{\text{rad}} \cdot \mathbf{T}_{\text{rad}}^{\top} \right)^{-1}, \quad (4.9)$$

where  $\mathbf{T}_{\text{ref}}$  and  $\mathbf{T}_{\text{rad}}$  are the  $K \times 12$  matrices that are formed by horizontally stacking the centered and transposed reference reflectance vectors (from  $\hat{\mathbf{R}}_{\text{ref}}^{(K)}[\text{CC}]$ ) and radiance vectors (from the current image  $\tilde{\mathbf{I}}^{(K)}$ ) of the learning patches, and  $^{\top}$  denotes the transpose. Figure 4.4 outlines the process of reflectance estimation of a  $B$ -channel radiance image  $\mathbf{I}^{(B)}$  using ms, wa, wn, and dw methods.

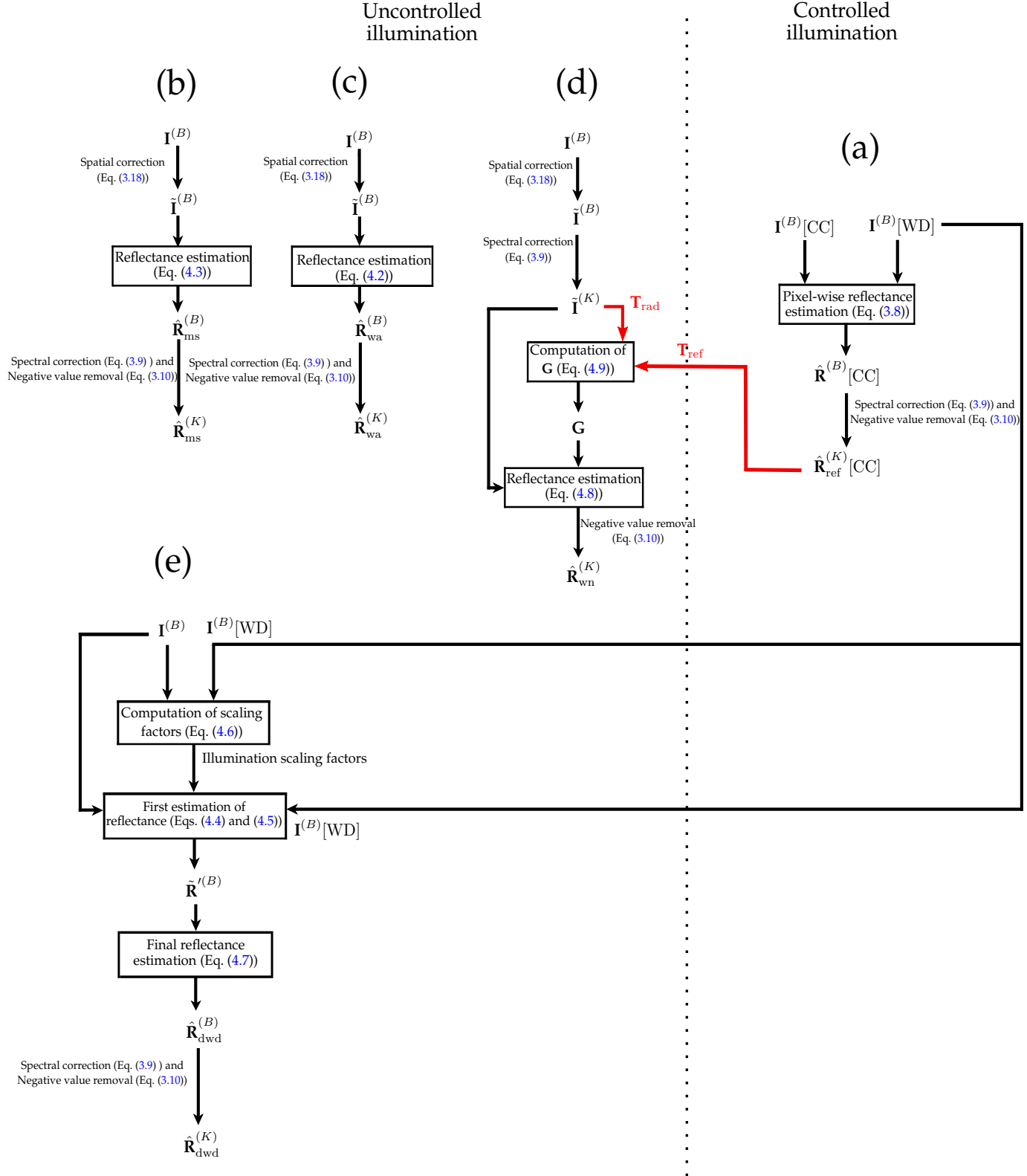


FIGURE 4.4: Flowchart of ms, wa, wn, and dwd methods. (a) Computation of the reference reflectance image acquired under controlled illumination. (b), (c), (d), and (e) ms, wa, wn, and dwd-based reflectance estimation from  $I^{(B)}$  acquired under uncontrolled illumination.

## 4.5 Estimated reflectance fidelity

This section focuses on reflectance quality assessment. We compare the quality of reflectance estimated by the proposed rw and orw methods, with the results obtained by the considered state-of-art methods. In Sec. 4.5.1, we present the metrics used in the objective evaluation. In Sec. 4.5.2, we focus on qualitative results. In Sec. 4.5.3, we provide a subjective assessment of the estimated reflectance spectra.

### 4.5.1 Metrics

To evaluate the accuracy of estimated reflectance, we use the set of test patches  $\mathcal{P}^t$  of the ColorChecker (see Fig. 3.11(a)). Let  $\hat{\mathbf{R}}_*^{(K)}$ ,  $*$   $\in$  {rw, orw, wa, ms, wn, dwd}, denote the reflectance image estimated by either the proposed methods (see Eqs. (3.23) and (3.28)) or the four implemented state-of-the-art methods (see Eqs. (4.2)–(4.8)). All surface elements of a test patch  $\mathcal{P}_j^t$  are assumed to have the same spectral response. Each test patch  $\mathcal{P}_j^t$ ,  $j = 1..12$  is then represented by a  $K$ -dimensional reflectance vector  $\hat{\mathbf{R}}_{*,\mathcal{P}_j^t}^{(K)}$  whose  $k$ -th component is computed as the average value in channel  $k$  over all the pixels characterizing the patch (see Eq. (3.24)). This vector is compared to the reference reflectance  $\hat{\mathbf{R}}_{\text{ref},\mathcal{P}_j^t}^{(K)}[\text{CC}]$  of the same patch estimated according to Eqs. (3.8)–(3.10) from an image acquired in laboratory under controlled illumination. When outdoor reflectance is well estimated, the spectra of the considered color chart patches should be similar, and ideally superposed to their laboratory counterparts.

Note that the ColorChecker is placed at various spatial coordinates in the images of our database. In case of high vegetation density, it is placed on the top of a wooden block to prevent patch occlusions by leaves. Thus we use a  $14 \times 14$  pixel window around the automatically detected center of each patch to ensure all these pixels belong to the same patch whatever the image.

We objectively assess each estimated reflectance image thanks to the mean absolute error (MAE) and angular error  $\Delta\theta$  of each test patch  $\mathcal{P}_j^t \in \mathcal{P}^t$  given by:

$$\text{MAE}(\hat{\mathbf{R}}_{\text{ref},\mathcal{P}_j^t}^{(K)}[\text{CC}], \hat{\mathbf{R}}_{*,\mathcal{P}_j^t}^{(K)}) = \frac{1}{K} \sum_{k=1}^K \left| \hat{R}_{\text{ref},\mathcal{P}_j^t}^k[\text{CC}] - \hat{R}_{*,\mathcal{P}_j^t}^k \right|, \quad (4.10)$$

and:

$$\Delta\theta(\hat{\mathbf{R}}_{\text{ref}, \mathcal{P}_j^t}^{(K)}[\text{CC}], \hat{\mathbf{R}}_{*, \mathcal{P}_j^t}^{(K)}) = \arccos \left( \frac{\langle \hat{R}_{\text{ref}, \mathcal{P}_j^t}^k[\text{CC}], \hat{R}_{*, \mathcal{P}_j^t}^k \rangle}{\|\hat{\mathbf{R}}_{\text{ref}, \mathcal{P}_j^t}^{(K)}[\text{CC}]\|_2 \cdot \|\hat{\mathbf{R}}_{*, \mathcal{P}_j^t}^{(K)}\|_2} \right), \quad (4.11)$$

where  $\|\cdot\|_2$  is the Euclidean norm. When  $\Delta\theta$  between two vectors (spectra in our case) is equal to zero, it means that these two vectors are collinear. We compute the average mean absolute error  $\overline{\text{MAE}}_*$  and angular error  $\overline{\Delta\theta}_*$  over all test patches of all estimated reflectance images.

We also compute the average channel-wise mean absolute error  $\overline{\text{MAE}}_*^k$  over all test patches of all reflectance images  $\{\hat{\mathbf{R}}_*^{(K)}\}_{u=1}^U$  provided by method  $*$ , defined as:

$$\overline{\text{MAE}}_*^k = \frac{1}{U} \frac{1}{J} \sum_{u=1}^U \sum_{j=1}^J |\hat{R}_{\text{ref}, \mathcal{P}_j^t}^k[\text{CC}] - \hat{R}_{*, \mathcal{P}_j^t}^k|, \quad (4.12)$$

where  $J = 12$  is the number of test patches, and  $U = 109$  is the number of images.

Moreover, we evaluate the computational cost of the examined reflectance estimation methods, and perform a visual observation of estimated reflectance at vegetation pixels to ensure that their integrity is preserved. At last, we assess the contribution of each reflectance estimation method for beet/weed detection and beet/thistle/goosefoot identification tasks.

## 4.5.2 Results

### 4.5.2.1 Mean absolute and angular errors

Table 4.2 shows the average mean absolute and angular errors over all images in the database for the six tested methods.

TABLE 4.2: Estimated reflectance errors. Bold shows the best result and italics the second best one.

|                                         | Method             |              |        |       |                |              |
|-----------------------------------------|--------------------|--------------|--------|-------|----------------|--------------|
|                                         | Illumination-based |              |        |       | Training-based | Hybrid       |
|                                         | rw                 | wa           | ms     | dwd   | wn             | orw          |
| $\overline{\text{MAE}}_*(\%)$           | 4.315              | 5.883        | 14.670 | 3.236 | 3.628          | <b>1.426</b> |
| $\overline{\Delta\theta}_*(\text{rad})$ | <i>0.046</i>       | <i>0.046</i> | 0.309  | 0.063 | 0.063          | <b>0.036</b> |

We can see that in terms of estimated reflectance error, the hybrid orw method provides the lowest average mean absolute ( $\overline{\text{MAE}}$ ) and angular ( $\overline{\Delta\theta}$ ) errors. The dwd and wn-based methods provide the second and third lowest  $\overline{\text{MAE}}$ , respectively. The rw-based method provides lower estimation error in terms of  $\overline{\text{MAE}}$  than wa and ms

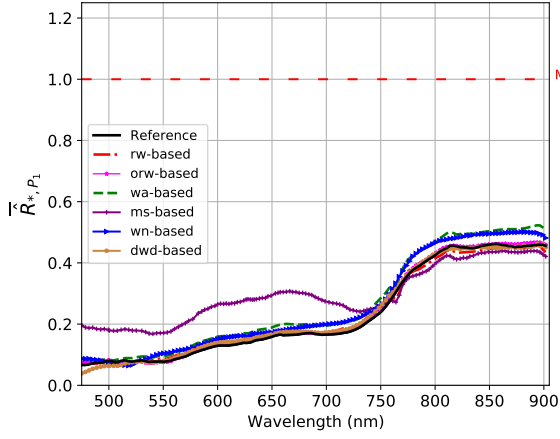
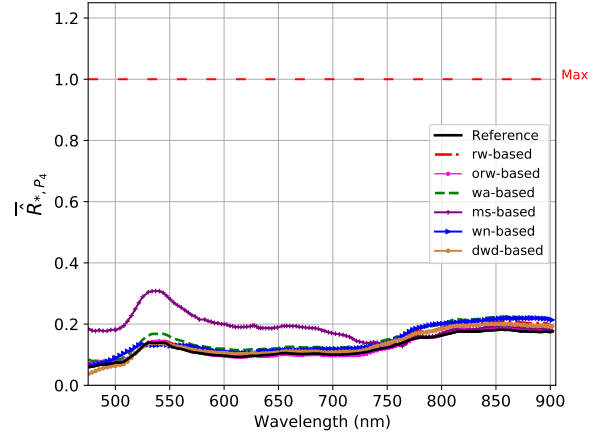
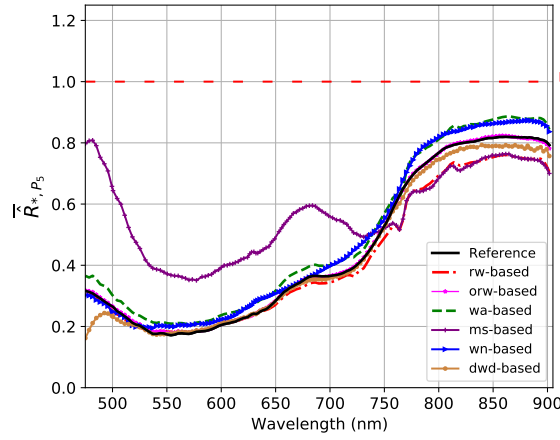
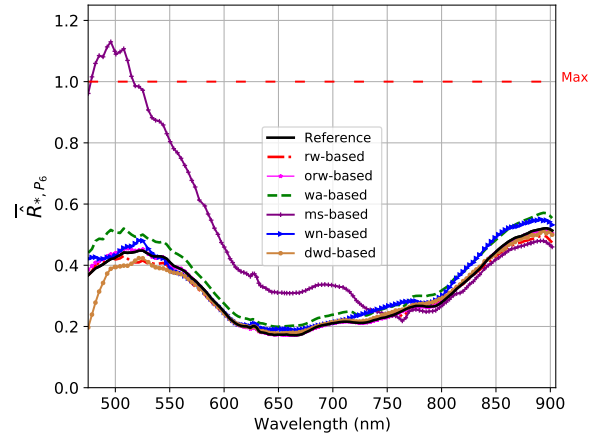
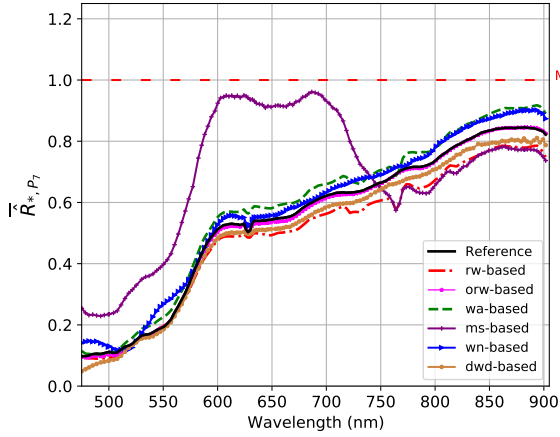
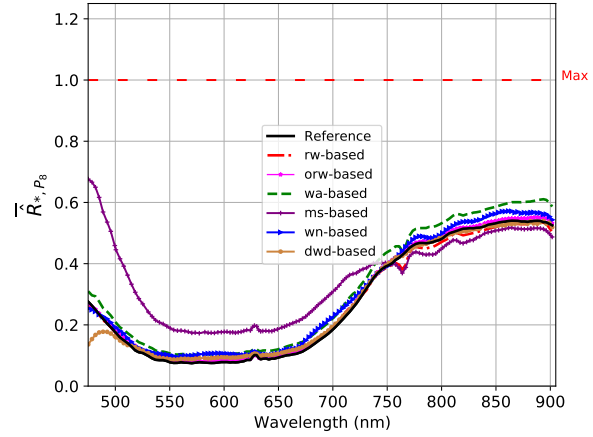
and the second lowest angular error along with wa method.

The MAE and  $\Delta\theta$  are complementary metrics and they, respectively, highlight two important properties: the scale and shape of the estimated spectra. Indeed, while MAE is mainly sensitive to the scale of the estimated spectra,  $\Delta\theta$  especially focuses on the shape of the spectra, because it is a scale-insensitive measure. Consequently, there might be no correlation between the results that were obtained by the MAE measure and those obtained by  $\Delta\theta$ . An optimal reflectance estimation method should satisfy both constraints (as does orw). As we can see from Tab. 4.2, wn and dwd provide better results than rw and wa in terms of  $\overline{\text{MAE}}$ . However, in terms of  $\overline{\Delta\theta}$ , rw and wa provide better results. In our case, more importance should be granted to the shape of the estimated spectra, and then the  $\Delta\theta$  metric is more relevant than the MAE. Reflectance error related to the scale between the reference and estimated reflectance spectra can be compensated by a normalization procedure [126, p. 54]. However, the shape of estimated spectra cannot be modified.

The ms method provides the worst results because it only analyzes pixels of background and vegetation that strongly absorb the incident light in the VIS domain. Hence, the biased illumination estimation in this domain affects the performance of ms method. It is worthwhile to mention that the wn method performance might also be biased, since it uses some of the ColorChecker patches as training references (to build the estimation matrix  $\mathbf{G}$ ), while the other patches of the same chart are used to evaluate the reflectance estimation quality. The best results are obtained by our orw method ; however, its performance might also be biased because it uses some of the ColorChecker patches to refine rw estimation.

Among illumination-based methods that analyze a single reference device, rw provides similar results to wa in terms of  $\Delta\theta$ , as well as better MAE results. This shows that taking account of the illumination variation during the frame acquisitions improves the reflectance estimation quality.

Figure 4.5(a)–(l) show the reference reflectance spectrum  $\hat{\mathbf{R}}_{\text{ref}, \mathcal{P}_j^t}[\text{CC}]$  (in black) and the average reflectance spectrum  $\overline{\mathbf{R}}_{*, \mathcal{P}_j^t}$  of each test patch computed over all reflectance images.

(a) Dark-skin patch ( $\mathcal{P}_1^t$  in Fig. 3.11)(b) Foliage patch ( $\mathcal{P}_4^t$  in Fig. 3.11)(c) Blue flower patch ( $\mathcal{P}_5^t$  in Fig. 3.11)(d) Bluish-green patch ( $\mathcal{P}_6^t$  in Fig. 3.11)(e) Orange patch ( $\mathcal{P}_7^t$  in Fig. 3.11)(f) Purple-blue patch ( $\mathcal{P}_8^t$  in Fig. 3.11)

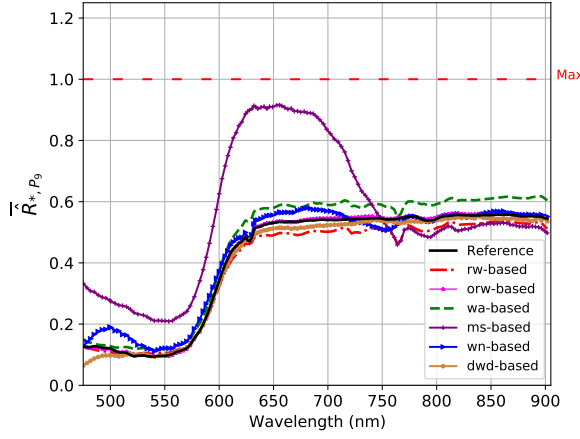
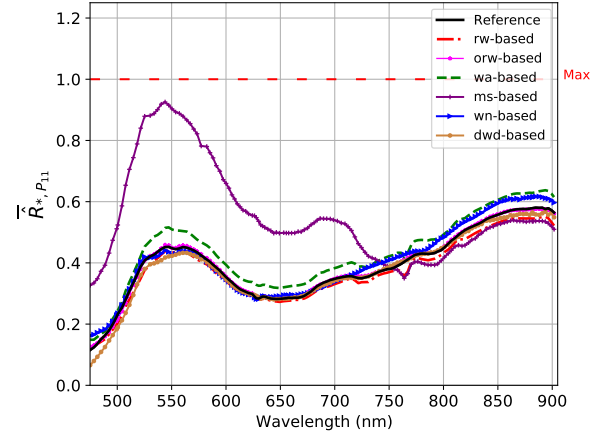
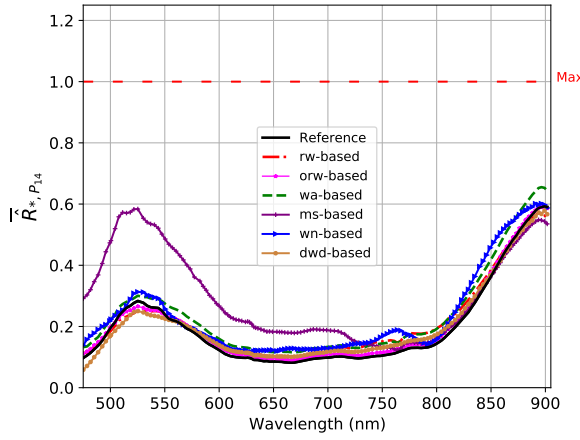
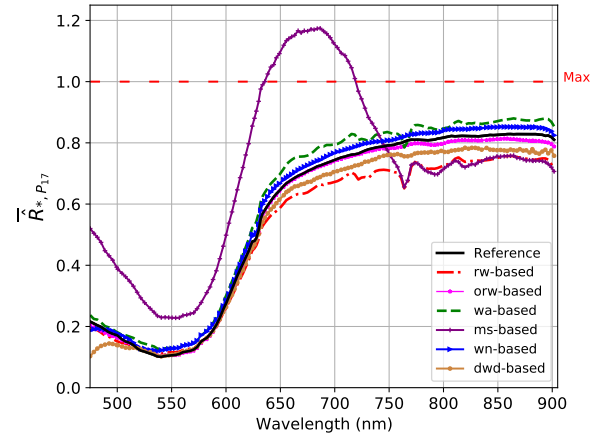
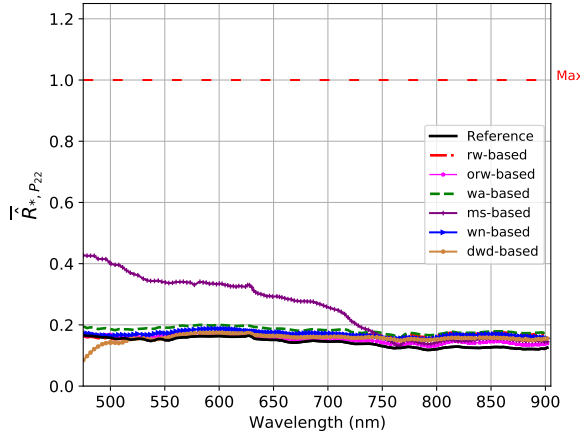
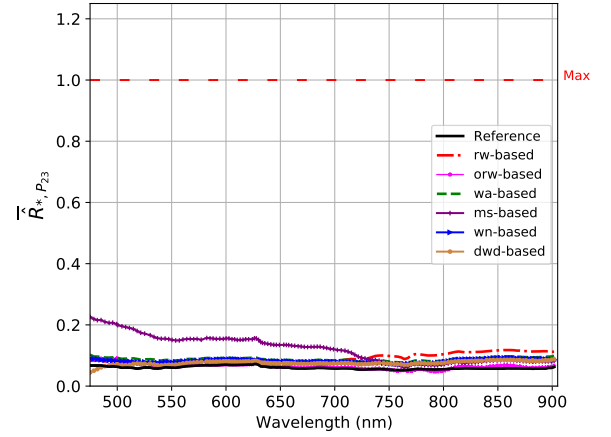
(g) Moderated-red patch ( $\mathcal{P}_9^t$  in Fig. 3.11)(h) Yellow-green patch ( $\mathcal{P}_{11}^t$  in Fig. 3.11)(i) Green patch ( $\mathcal{P}_{14}^t$  in Fig. 3.11)(j) Magenta patch ( $\mathcal{P}_{17}^t$  in Fig. 3.11)(k) Neutral 5 patch ( $\mathcal{P}_{22}^t$  in Fig. 3.11)(l) Neutral 35 patch ( $\mathcal{P}_{23}^t$  in Fig. 3.11)

FIGURE 4.5: Estimated reflectance spectra of the ColorChecker test patches.

We can see that the shape of the reference reflectance spectra of the ColorChecker test patches and their rw and orw-based estimations are very similar and totally overlap



for some patches (e.g., purple-blue and bluish-green patches). This also can be deduced from the low angular error values (see Tab. 4.2) that indicate to what extent the estimated spectra are similar (in a collinearity sense) to the reference spectra.

The *wn*-based reflectance spectra are very similar to the reference reflectance spectra except for certain patches (such as orange or moderated-red) where their shapes slightly differ. The *dwd* method provides also good estimation results. For some patches (such as blue flower, bluish-green, and purple-blue), the estimated reflectance values in the range [475 nm, 500 nm] are different from the reference reflectance values. The *wa* method provides acceptable estimations in the range [475 nm, 901 nm]. In the range [475 nm, 745 nm], the *ms* method provides the worst estimation results. However, in the range [750 nm, 901 nm], *ms* provides similar reflectance estimation results to those of *rw* method.

#### 4.5.2.2 Channel-wise estimated reflectance error

Figure 4.6 shows the average channel-wise MAE of estimated reflectance computed according to Eq. (4.12), and Fig. 4.7 shows the channel-wise average energy that is computed as the average reflectance value over all images for each channel  $\hat{R}_*^k$ .

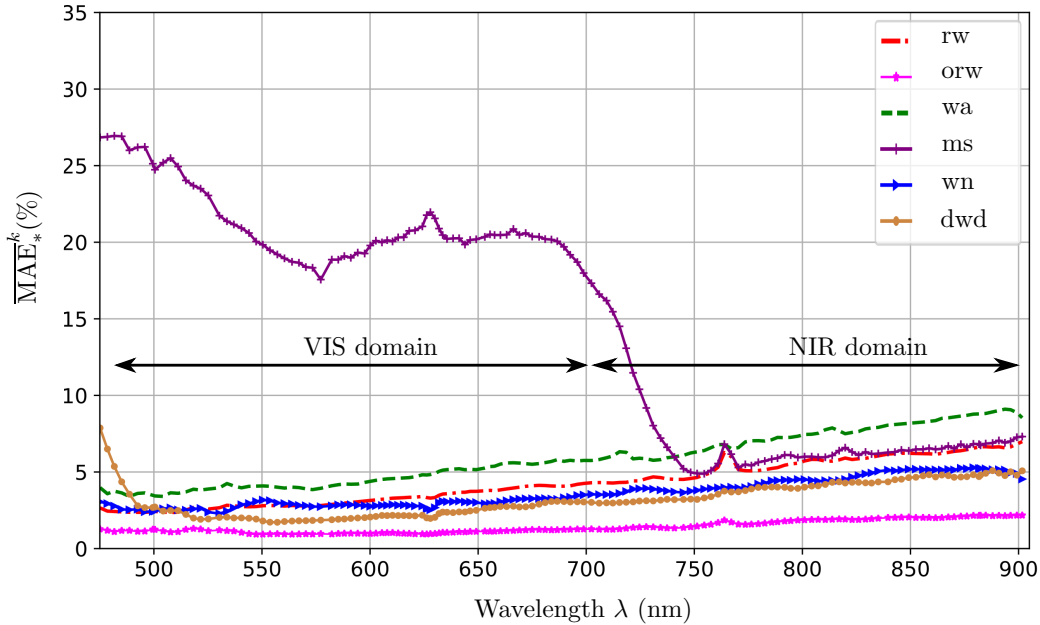


FIGURE 4.6: Channel-wise MAE over all reflectance images estimated by each method.

We can see that the estimated reflectance error for each channel is correlated with the channel-wise energy. Indeed, higher errors of estimated reflectance are more

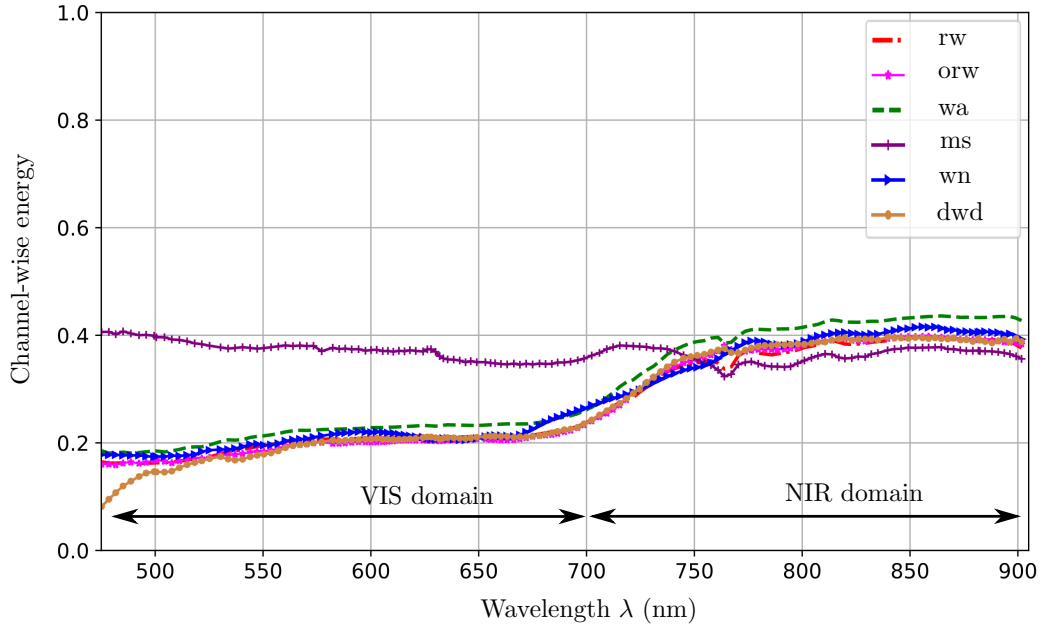


FIGURE 4.7: Channel-wise energy over all reflectance images estimated by each method.

susceptible to occur in channels associated to the NIR domain since energy within these channels is higher than within those associated to the VIS domain.

Considering the channel-wise error (see Fig. 4.6), the highest error rates in the range [475 nm, 750 nm] are obtained by *ms* because it only analyzes pixels of background and vegetation that strongly absorb the incident light. Hence, illumination estimation is biased using *ms* in this domain. In the range [750 nm, 901 nm] (NIR domain), the light reflection factor of vegetation is high (it appears brighter in Fig. 4.3), and the *ms* method gives very similar results as *rw*. The *rw* method provides lower channel-wise error than the *wa* method. The *dwd* and *wn* provide the second and third lowest error rates respectively along the working domain [475 nm, 901 nm] of the Snapscan camera, while *orw* method provides the lowest error rates.

In the following, we assess a visual observation of some vegetation spectra. Indeed, the quality of reflectance estimation evaluated at pixels of test patches of the ColorChecker chart may not be generalized over all image pixels, especially for reflectance estimation methods such as *wn*. Reflectance estimated at pixels of test patches might be accurate while being inaccurate or biased at vegetation or soil pixels.

### 4.5.3 Reflectance estimated at vegetation pixels

To ensure that there is a correlation between low estimated reflectance errors at ColorChecker chart patches and the integrity of the estimated vegetation reflectance, we assess reflectance spectra estimated at vegetation pixels by the six considered methods thanks to a visual observation. We expect that vegetation spectra estimated by methods that provide low reflectance estimation errors (computed on chart patches) should be similar. For this purpose, we estimate the reflectance using the six tested methods from two radiance images that contain beet plants and a mixture of beet, thistle and goosefoot. The vegetation signature of each species is computed by averaging the spectra at 100 pixels defined by a small region of interest (colored spots in Figs. 4.8(a) and 4.9(a)). From Fig. 4.8(b), we can see that the rw, orw, wa, ms, and

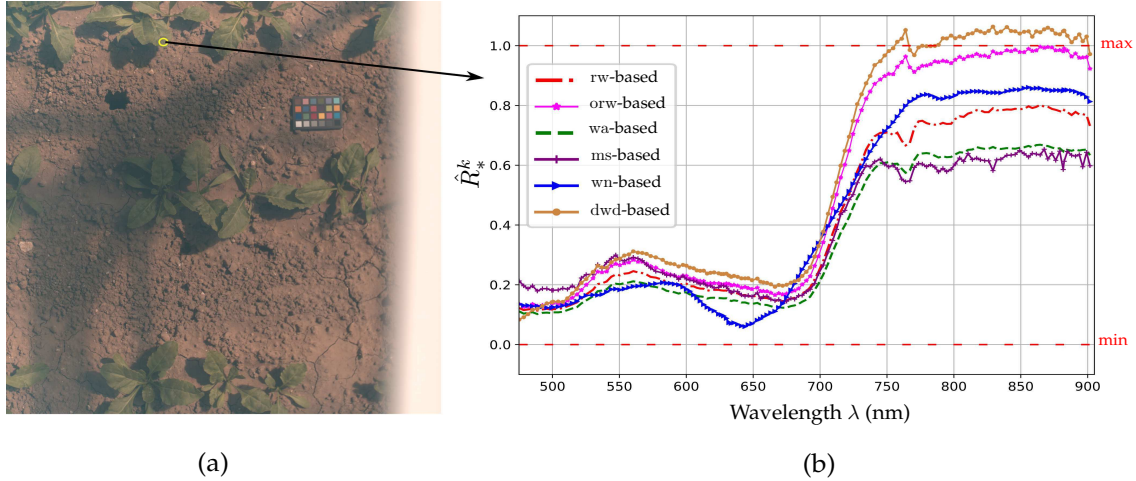


FIGURE 4.8: (a) RGB rendering of a multispectral image of beet. (b) Estimation of beet reflectance spectrum at the yellow spot.

dwd-based estimations provide a similar spectrum with values in the range  $[0, 1]$  except for dwd reflectance spectrum whose values in the range  $[760 \text{ nm}, 901 \text{ nm}]$  are above 1 (saturated). Thus, spectral information held by several spectral bands in this range might be lost. The shape of the averaged spectrum deduced from spectra that are estimated using wn method is different from those estimated by other methods in the range  $[475 \text{ nm}, 680 \text{ nm}]$ . The same phenomenon is observed in Fig. 4.9 where the shape of the wn-based reflectance spectrum is also different. Moreover, in the range  $[620 \text{ nm}, 680 \text{ nm}]$ , it contains negative values. This analysis confirms our first intuition about estimation methods based on a transform matrix (see Sec. 3.2.1.3). Such

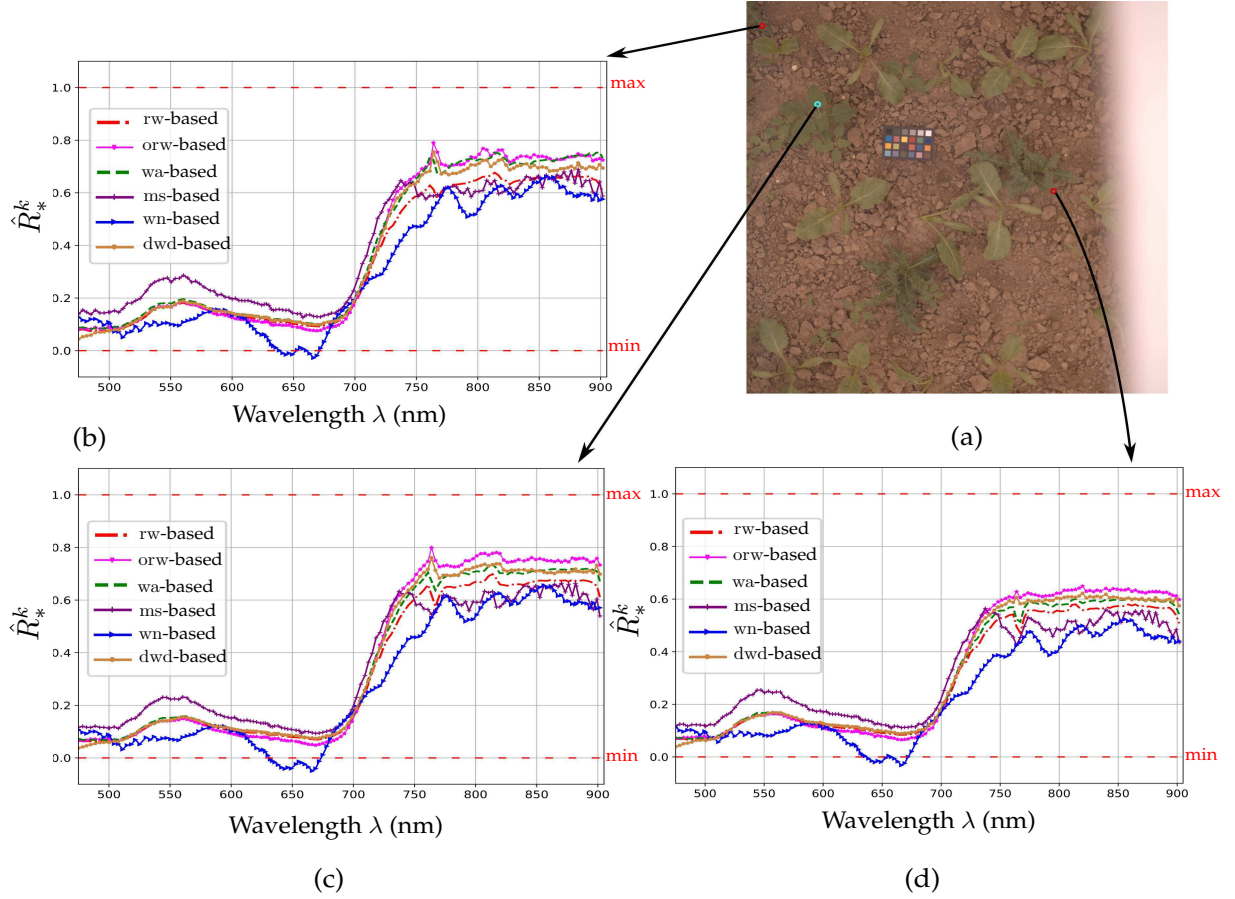


FIGURE 4.9: (a) RGB rendering of a multispectral image of a mixture of beet, thistle, and goosefoot. (b)–(d) Estimation of thistle and goosefoot reflectance spectra at the red and cyan spots respectively.

methods strongly depend on the training set of radiance and reflectance spectra. Because they generally use GretagMacbeth<sup>TM</sup> ColorChecker patches to build the matrix  $\mathbf{G}$  (see Eq. (4.9)), these methods might be able to accurately estimate the reflectance of the other patches, but are not adapted to estimate the reflectance of remaining scene objects especially when (as vegetation spectra) their spectral signatures are much different from those of the chart patches. Besides, such methods do not consider the illumination variation during the frame acquisitions that is highlighted by our image formation model (see Eq. (2.10)), which further weakens their performances in reflectance estimation. The dwd method provides generally accurate estimations of vegetation spectra (see Fig. 4.9(b)–(d)). However, in some cases, dwd provides spectra with saturated reflectance values in the range [760 nm, 901 nm] (see Fig. 4.8(b)). This is due to the channel-wise normalization of the first estimated reflectance by the white patch reflectance. Indeed, since vegetation strongly reflects light in the NIR domain, reflectance values estimated at vegetation pixels in this spectral range

might be higher than those of the white patch, which provides saturated values after normalization according to Eq. (4.7). Therefore, spectral information that might be crucial for pixel classification is lost in this spectral range.

#### 4.5.4 Computation cost

Table 4.3 shows the average execution time per image (evaluated over 10 images) required by each considered reflectance estimation method. Note that steps of spatial and spectral corrections, and negative value removal are not considered in Tab. 4.3 because they are performed in all methods. Thus, only reflectance estimation computation cost is evaluated because it is specific to each method. The tests were con-

TABLE 4.3: Average execution time of each reflectance estimation method.

|                            | Method             |       |       |                |              |                       |
|----------------------------|--------------------|-------|-------|----------------|--------------|-----------------------|
|                            | Illumination-based |       |       | Training-based |              | Hybrid                |
|                            | rw                 | wa    | ms    | dwd            | wn           | orw                   |
| Eqs.                       | (3.23)             | (4.2) | (4.3) | (4.4)–(4.7)    | (4.8), (4.9) | (3.23), (3.26)–(3.28) |
| Average execution time (s) | 28                 | 20    | 15    | 42             | 81           | 41                    |

ducted on a PC running under Linux Mint 19.3 (Tricia) operating system, featuring a 6-core Intel CPU at 2.6 GHz with 32 GB of RAM, and using the python kernel 3.7.9. The *ms* method is the fastest among the six tested methods and takes only 15 s to estimate the reflectance image. It is followed by the *wa* method with 20 s because illumination is only estimated  $B$  times. The reflectance at all pixels associated to channel index  $b$  is estimated by dividing the radiance values at these pixels by the estimated illumination associated to channel index  $b$ . The *wn* method is the most greedy because of the vectorial procedure of reflectance estimation. The  $K \times K$  matrix  $\mathbf{G}$  is applied successively at each pixel  $p$ . The number of elementary operations to estimate the reflectance spectrum at  $p$  using  $\mathbf{G}$  is  $(2K - 1) \times K$  operations. It takes 81 s to process all the  $X \times Y$  pixels. The *dwd* operates in three steps and takes 42 s on average to estimate the reflectance image. The radiance image  $\mathbf{I}^{(B)}$  is first divided by the white diffuser image  $\mathbf{I}^{(B)}[\text{WD}]$ , then each resulting value in each channel at each pixel is multiplied by an illumination scaling factor to provide the first reflectance image  $\tilde{\mathbf{R}}'^{(B)}$ . Finally,  $\tilde{\mathbf{R}}'^{(B)}$  is normalized in each channel by the reflectance of the white patch to provide the final reflectance image  $\hat{\mathbf{R}}_{\text{dwd}}^{(B)}$ . Our *rw* method operates row-wise and estimates illumination  $Y \times B$  times. For each channel, reflectance at

all pixels in a row is estimated by dividing the radiance values at these pixels by the estimated illumination in the same pixel row. The average running time of *rw* method is 28 s with our implementation. The *orw* method takes 41 s (including 13 s for the optimization step) on average to process all pixels and requires  $2 \times K$  extra operations per pixel to optimize the *rw* reflectance image. Our *rw* method provides a good trade-off between computation cost and robustness against illumination variation during acquisitions.

## 4.6 Reflectance features for beet/weed detection and identification

In this section, we evaluate the contribution of our proposed *rw* and *orw*-based reflectance estimation methods for supervised beet/weed detection and identification. For this experiment, we focus on beet (crop) that must be distinguished from thistle and goosefoot (weeds). For this purpose, we compare the performance of a pixel classifier that analyzes these features. We use a data set composed of 37 radiance (13 single-species and 24 mixed) images acquired at different days (on May and June 2019) that we split into a learning set and a test set, denoted as  $\mathcal{S}^{learn}$  (23 images) and  $\mathcal{S}^{test}$  (14 images) (see Tab. 4.4). Illumination conditions are various in the two sets and  $\mathcal{S}^{test}$  mostly includes images that are acquired on different days from those of  $\mathcal{S}^{learn}$  (see Fig. 4.1(b)). Note that, as a consequence, vegetation in the learning and test image sets may not be exactly at the same growth stages. We first compare the discrimination power of reflectance features provided by our *rw* and *orw* methods against radiance features to assess each reflectance estimation method for crop/weed identification and detection. Subsequently, we compare them with reflectance features that are estimated using each of the four considered state-of-the-art methods (*wa*, *ms*, *wn*, and *dwd*) (Secs. 4.6.4–4.7).

### 4.6.1 Learning and test pixel extraction

From the learning set  $\mathcal{S}^{learn}$ , we randomly extract  $\mathcal{N}^l$  learning pixels per class. For a given class  $\mathcal{C}^i$ ,  $i \in \llbracket 1, N_c \rrbracket$ , the number of extracted learning pixels per image depends on the number of images where class  $\mathcal{C}^i$  is represented in  $\mathcal{S}^{learn}$  (occurrences).

Among the 23 learning images, the beet (crop) class appears in 17 images, thistle in nine images, and goosefoot in 12 images. In the test set  $\mathcal{S}^{test}$ , beet, thistle, and goosefoot are represented, respectively, in 12, 10, and four images. For the weed detection task, we extract  $2\mathcal{N}^l$  learning pixels, half for crop and half for weed class. Because we merge thistle and goosefoot prototype pixels to build a single weed class, we extract  $\mathcal{N}^l/2$  learning pixels for thistle and  $\mathcal{N}^l/2$  for goosefoot. Each pixel is characterized by a  $K$ -dimensional ( $K = 141$ ) feature vector of reflectance (or radiance) values. The reflectance/radiance images are averaged channel-wise over a  $5 \times 5$  pixel window to reduce noise and within-class variability. Table 4.4 shows the number of learning and test pixels per class for beet/weed detection and beet/thistle/goosefoot identification. All of the available pixels in  $\mathcal{S}^{test}$  are used to assess the generalization power of a supervised classifier.

TABLE 4.4: Number of learning and test pixels for beet/weed detection (left sub-column) and beet/thistle/goosefoot identification (right sub-column) ( $\mathcal{N}^l = 400,000$  pixels in this experiment).

| Class $\mathcal{C}^i$ |           | $\mathcal{S}^{learn}$ (23 images) |                                 |                    | $\mathcal{S}^{test}$ (14 images) |                        |
|-----------------------|-----------|-----------------------------------|---------------------------------|--------------------|----------------------------------|------------------------|
|                       |           | #Occurrences                      | #Learning pixels per occurrence |                    | #Occurrences                     | #Test pixels per class |
| Crop                  | Beet      | 17                                | $\mathcal{N}^l/17$              |                    | 12                               | 5,714,326              |
| Weed                  | Thistle   | 9                                 | $(\mathcal{N}^l/2)/9$           | $\mathcal{N}^l/9$  | 10                               | 5,461,013              |
|                       | Goosefoot | 12                                | $(\mathcal{N}^l/2)/12$          | $\mathcal{N}^l/12$ | 4                                | 1,283,620              |

#### 4.6.2 Supervised classification

Choosing a machine learning classifier to tackle a classification problem is strongly related to the type of dataset. An ideal classifier should be fast during training and inference procedures, capable of identifying non linearly separable classes in high dimensional feature spaces, and provide good classification results. Classifiers that are fast during the learning and inference procedures are highly appreciated.

From a machine learning standpoint, a classification problem is called linear if the set of samples characterizing each class (cluster) can be separated by a hyperplane in the feature space (low within-class dispersion and high between-class dispersion). Figure 4.10 shows a 2D projection of rw-based estimated reflectance spectra at 15,000 (5,000 per class) vegetation pixels of beet, thistle, and goosefoot. These spectra are projected into a 2-dimensional space computed with the linear supervised partial least squares discriminant analysis (PLS-DA) technique [162] in Fig. 4.10 (a), and by



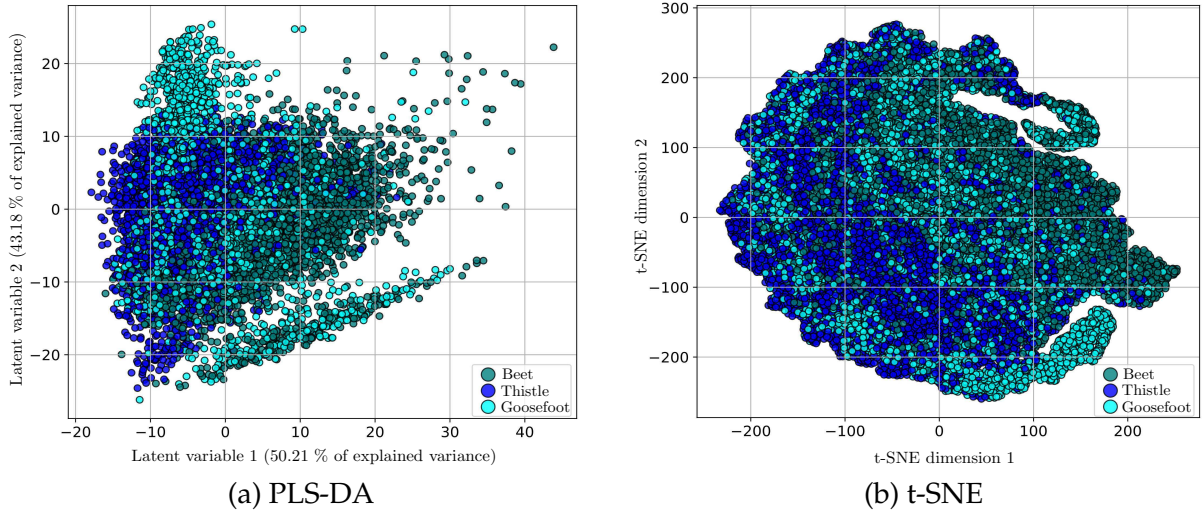


FIGURE 4.10: Visualization of rw-based estimated reflectance spectra associated to beet, thistle, and goosefoot pixels using PLS-DA and t-SNE dimension reduction techniques.

the non-linear non-supervised t-SNE technique [181] in Fig. 4.10(b). We can see that the clusters strongly overlap. So, we can deduce that the classes are not linearly separable. In this case, classifiers that model non-linear rules should be applied in order to separate these classes.

Quadratic discriminant analysis (QDA) is a well-known supervised method used to predict class membership. Like linear discriminant analysis (LDA), QDA assumes Gaussian distributions for all classes. However, in contrast to LDA that computes a common covariance matrix for all classes, QDA determines a class-specific covariance matrix, and provides a quadratic boundary decision. In order to take local distributions into account, support vector machines (SVMs) can be an alternative to decision rules based on the distance to the gravity center of each cluster. The SVM classifier was originally introduced by Cortes and Vapnik [34] and has been extensively used in machine learning community for various tasks. An SVM is inherently a two-class classifier as it aims to find the optimal hyperplane that maximizes the margin between two classes. In the case of linearly separable classes, this optimal hyperplane has the largest distance to the closest learning samples of each class (so-called “support vectors”). To handle multiclass problems ( $N_C > 2$ ), the algorithm proceeds in a “one-vs-all” way and  $N_C$  classifiers are inferred, where each one specializes in a specific class. Because the decision rule of an SVM is based on



a hyperplane, it is inherently adapted to linearly separable data. In order to generalize to non-linear problems, the “kernel trick” [70, p. 423] is used to provide the so-called kernel support vector machines (K-SVMs). A K-SVM classifier uses a kernel function that maps features into a higher dimensional space in which classes are assumed to be linearly separable. Then, linear SVM is applied in the transformed feature space in order to obtain the hyperplane that maximizes the distance between the classes. Combining SVMs with the kernel trick provides a powerful classifier that is able to handle non-linear problems. SVMs have been successfully used to classify multispectral data [5, 47]. However, the learning step of SVM (computation of the support vectors) is computationally expensive when dealing with high-dimensional and large data sets.

The method based on  $k$ -nearest neighbors ( $k$ -NN) is a popular non-parametric learning-free one (no prior statistical assumption about the class distributions) that has been successfully used in classifying multispectral data [57]. The classification of a given test (or query) sample is achieved thanks to a similarity-based or distance measure (such as the Euclidean distance) between the test sample and each learning sample. The test sample is assigned to the majority class of its  $k$  nearest learning neighbors. This classifier is simple yet efficient since it does not rely on any assumption about the class distributions. Unfortunately, predictions with  $k$ -NN tend to be computationally expensive since all learning samples need to be scanned in order to predict the class of each test sample. In our case, both learning and test data sets are large (see Tab. 4.4) and lay in a high-dimensional feature space (141 spectral features). Therefore,  $k$ -NN might not be adapted to this problem. Moreover, accuracy of  $k$ -NN strongly depends on the used metric [2, 3].

The Random Forest (RF) is a “bagging” algorithm that combines multiple weak randomly created learners (or decision trees) to predict the final output as the majority class among all classes predicted by each model. The weak learner models are independent since each one is trained with a different subset of samples/features generated using “bootstrap aggregation” method. RF has been successfully adopted by several applications such as weed detection [72, 108] because it is able to discriminate non-linear classes. However, the learning step with large datasets might still be slow (though faster than K-SVM).

### 4.6.3 Supervised classification by boosting approach

Alternatively to the bagging approach, the “boosting” approach attempts to build a strong classifier by using weak learners in series. One of the first successful boosting algorithms is AdaBoost [54]. In contrast to RF, AdaBoost generally uses “decision stumps” (trees with one node and two leaves) as weak learners rather than fully grown trees. It sequentially fits these stumps, and new ones are added thanks to a sample reweighting procedure to concentrate on samples where previous learners performed poorly [54]. The final predicted output is deduced by the weighted majority voting rule since each stump has a different weight according to its performance.

Another kind of non-linear classifier is a gradient boosting machine (GBM). A GBM also sequentially fits several weak learners in order to build a strong learner. While AdaBoost updates the sample weights to reduce classification errors, a GBM follows gradient descent optimization using a differentiable loss function to sequentially improve the performance of a weak learner. Considering a learning dataset with  $N_C$  classes,  $N_C$  ensemble models are induced in order to build the final learner denoted as  $\hat{\Theta} = \{\hat{\theta}^i\}_{i=1}^{N_C} : \mathbb{R}^K \rightarrow \mathbb{R}^{N_C}$ . The classification score associated to each model  $\hat{\theta}^i$  is used to determine the class membership of a given sample [55]. The predicted label  $\hat{y}_s$  for a test sample  $\mathbf{x}_s \in \mathbb{R}^K$  is obtained thanks to Bayes’ rule:

$$\hat{y}_s = \arg \max_{i=1, \dots, N_C} \hat{\mathcal{P}}^i(\mathbf{x}_s), \quad (4.13)$$

where  $\hat{\mathcal{P}}^i(\mathbf{x}_s) = \frac{\exp(\hat{\theta}^i(\mathbf{x}_s))}{\sum_{l=1}^{N_C} \exp(\hat{\theta}^l(\mathbf{x}_s))}$ . The classification procedure using a GBM for multi-class classification is detailed in Appendix A.1.

GBMs can be very efficient but conventional implementations face several challenges. Indeed, for every feature, the algorithm scans all the samples to estimate the information gain of all possible split points (thresholds) in order to find the optimal one. Because it also scans all the samples in order to compute the gradients, the algorithm can be quite slow to analyze large and high-dimensional datasets. Furthermore, because there are several hyper-parameters to adjust, the resulting classifier can be also over-fitted.

Recent GBM implementations such as XGBoost [29], CatBoost [146], or LightGBM [83]

bypass these problems and provide efficient classifiers. The LightGBM implementation for instance is known to be highly efficient in both learning and inference steps and to reach high classification performances. In our case, only few seconds are required to analyze all learning samples, and all test pixels of a multispectral image are classified within few seconds (less than 3 s for 500,000 test pixels). Such a classifier of course requires to adjust specific parameters such as the number of leaves, the depth of trees, and the learning rate, and needs to be fine-tuned in order to achieve good performances and avoid over-fitting.

#### 4.6.4 Evaluation metrics

Since our application is dedicated to process real-world data (crop/weed monitoring), we do not control the number of test samples (pixels) that have to be regrouped into each class (crop and weeds). Table 4.4 shows that the number  $N^i$  of test pixels associated to each class is highly skewed. Therefore, the classical accuracy score can be a misleading measure to evaluate a classifier performance [73, p. 114]. A classification model that predicts the majority class for all test pixels reaches a high classification accuracy. However, this model can also be considered as weak when misclassifying pixels of the minority classes is worse than missing pixels from the majority classes. This phenomenon is known as the “accuracy paradox”.

To overcome the accuracy paradox, the performance of a classification model for imbalanced datasets should be assessed with appropriate metrics such as balanced accuracy and precision/recall curve [73, pp. 53–56, 114]. Although some of the metrics may be more meaningful and easy to interpret than others, there is no consensus in the literature for choosing a single optimal metric. In our case, we want to correctly detect weed pixels without sub-detection and to correctly detect crop pixels without over-detection, to avoid both missing weeds and spraying crops. Therefore, the performance of our classification model on both crop and weed detection should be comparable. For this purpose, we compute the per-class accuracy score and the weighted overall accuracy score. We also compute the F1-score that combines the precision and recall measures. These three measures should well summarize the classification performance of imbalanced sets of test pixels.

Let us denote the set of true test pixel labels as  $y$  and the set of predicted labels as  $\hat{y}$ .

The per-class accuracy score for class  $\mathcal{C}^i$ ,  $i \in \llbracket 1, N_C \rrbracket$ , is:

$$\text{Accuracy}_{\mathcal{C}^i} = \frac{1}{\mathcal{N}^{t_i}} \sum_{j=1}^{\mathcal{N}^i} \mathbb{1}(\hat{y}_j^i = y_j^i), \quad (4.14)$$

where  $y_j^i$  and  $\hat{y}_j^i$  are the true and predicted labels for the  $j$ -th test pixel of class  $\mathcal{C}^i$ , and  $\mathcal{N}^{t_i}$  is the number of test pixels characterizing the class  $\mathcal{C}^i$ .

The weighted overall accuracy is defined as:

$$\overline{\text{Accuracy}} = \frac{\sum_{i=1}^{N_C} \text{Accuracy}_{\mathcal{C}^i} \cdot \omega_{\mathcal{C}^i}}{\sum_{i=1}^{N_C} \omega_{\mathcal{C}^i}}, \quad (4.15)$$

where  $\omega_{\mathcal{C}^i} = \frac{1}{\mathcal{N}^{t_i}}$  is the weight associated to class  $\mathcal{C}^i$  computed as the inverse of its size, so as to handle imbalanced classes. Because the F1-score (see Eqs. (4.17)–(4.19)) privileges the classification of true positives pixels (weed pixels in our case), we compute the overall  $\overline{\text{F1}}$ -score as the population-weighted F1-score so that the performances over all classes are considered:

$$\overline{\text{F1}} = \frac{\sum_{i=1}^{N_C} \text{F1}_{\mathcal{C}^i} \cdot \omega_{\mathcal{C}^i}}{\sum_{i=1}^{N_C} \omega_{\mathcal{C}^i}}. \quad (4.16)$$

The F1-score of class  $\mathcal{C}^i$  is computed as:

$$\text{F1}_{\mathcal{C}^i} = \frac{2 \cdot \text{Precision}_{\mathcal{C}^i} \cdot \text{Recall}_{\mathcal{C}^i}}{\text{Precision}_{\mathcal{C}^i} + \text{Recall}_{\mathcal{C}^i}}, \quad (4.17)$$

where

$$\text{Precision}_{\mathcal{C}^i} = \frac{\sum_{j=1}^{\mathcal{N}^i} \mathbb{1}(\hat{y}_j^i = y_j^i)}{|\hat{y}^i|}, \quad (4.18)$$

and

$$\text{Recall}_{\mathcal{C}^i} = \frac{\sum_{j=1}^{\mathcal{N}^{t_i}} \mathbb{1}(\hat{y}_j^i = y_j^i)}{\mathcal{N}^{t_i}}, \quad (4.19)$$

where  $|\hat{y}^i|$  is the number of predicted pixels for class  $\mathcal{C}^i$ .

## 4.6.5 Classification experiments

### 4.6.5.1 Retained classifier

The non-parametric QDA and parametric LightGBM (LGBM) classifiers are applied for supervised weed detection and identification problems. The choice of these two

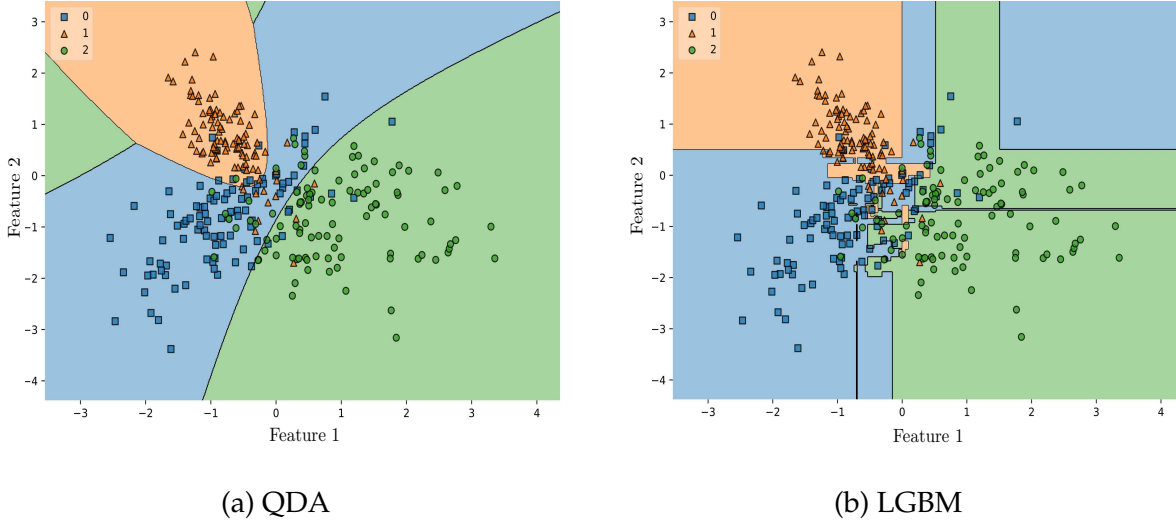


FIGURE 4.11: Non-linear decision boundaries learned by (a) QDA and (b) LGBM in the case of three synthetic classes. Samples have been generated using the *sklearn*<sup>24</sup> Python module.

non-linear classifiers is motivated by their processing time during the learning and prediction procedures.

Furthermore, QDA is a simple non-parametric classifier that is based on Bayes' theorem to perform predictions, whereas LGBM is a parametric tree-based classifier that requires a learning procedure to model a complex classification rule. It uses a histogram-based algorithm to bucket the features into discrete bins, which drastically reduces the memory and time consumption. Figure 4.11 illustrates the difference between their decision boundary in the case of three non-linearly separable classes projected in a 2D feature space. The parameter-free QDA computes statistics of the features that characterize each class in order to perform predictions. A sample  $\mathbf{x}_s \in \mathbb{R}^K$  is assigned to the class that maximizes the quadratic discriminant function  $\Psi$ :

$$\hat{y}_s = \arg \max_{i=1, \dots, N_C} \Psi^i(\mathbf{x}_s), \quad (4.20)$$

where

$$\Psi^i(\mathbf{x}_s) = -\frac{1}{2} \log |\Sigma_{C^i}| - \frac{1}{2} (\mathbf{x}_s - \mu_{C^i})^\top \Sigma_{C^i}^{-1} (\mathbf{x}_s - \mu_{C^i}) + \log(\pi_{C^i}), \quad (4.21)$$

where  $\Sigma_{C^i}$  is the covariance matrix of class  $C^i$ ,  $\mu_{C^i}$  is the mean feature vector of class  $C^i$ , and  $\pi_{C^i}$  is the prior probability of class  $C^i$ .

<sup>24</sup> <https://scikit-learn.org>

LGBM is a histogram-based algorithm that uses gradient boosted decision trees (GB-DTs) in a way that solves the high-dimensional learning problem thanks to two techniques named *Exclusive Feature Bundling* (EFB) and *Gradient-based One-Side Sampling* (GOSS) [83]. Details about EFB and GOSS techniques are given in Appendix A.2.

For the learning procedure of LGBM (see Appendix A.1), we set the learning rate  $\epsilon$  to 0.05. This value is chosen experimentally as it allows the classifier to learn from the learning samples while preventing over-fitting. Training with low learning rate values is also a common practice when using deep convolutional neural networks for classification [165]. The sampling ratios  $a$  and  $b$  (see Appendix A.2.2) are set to 0.2 and 0.1 respectively (default values). The number of leaves and boosting operations are set to 150 and 100 respectively so that the underlying structure of our data is learned. As learning evaluation metrics, we use the log loss function since it is one of the major metrics used in classification that can handle binary (weed detection) and multiclass (weed identification) classification tasks. The number of bins used for histogram building is set to 255 (default value).

#### 4.6.5.2 Classification results

Tables 4.5a and 4.5b show the classification results obtained by QDA and LGBM classifiers for weed detection and identification, respectively. Figures 4.12 and 4.13 show the color-coded vegetation pixel classification of two test images using the LGBM classifier in weed detection and identification tasks, respectively.

Let us first compare the classification performance of reflectance against radiance features for the weed detection task. From the results given in Tab. 4.5a, we can see that reflectance features estimated by illumination-based methods (rw, wa, ms, and dwd) provide better classification results than radiance features in terms of average F1 and accuracy scores, whatever the classifier. The worst classification results are obtained by reflectance features estimated using the wn method. Training-based methods, such as wn, can provide an accurate reflectance estimation of scene objects when their optical properties are close to those of the training samples used to build the matrix  $\mathbf{G}$  (see Eq. (4.9)). In our case, the optical properties of vegetation are very different from that of the training ColorChecker patches. Thus, wn provides

TABLE 4.5: Beet/weed detection (4.5a) and beet/thistle/goosefoot identification (4.5b) results with QDA and LGBM classifiers. Bold shows best result and italics second best one.

|                        |      | Classifier | Radiance feature | Reflectance features |      |      |      |                |        |
|------------------------|------|------------|------------------|----------------------|------|------|------|----------------|--------|
|                        |      |            |                  | Illumination-based   |      |      |      | Training-based | Hybrid |
|                        |      |            |                  | rw                   | wa   | ms   | dwd  | wn             | orw    |
| Per-class Accuracy (%) | Beet | QDA        | 90.5             | 73.6                 | 73.3 | 86.8 | 78.3 | 36.5           | 76.6   |
|                        |      | LGBM       | 81.0             | 85.8                 | 81.7 | 82.1 | 84.1 | 66.4           | 69.6   |
|                        | Weed | QDA        | 49.3             | 72.9                 | 73.4 | 70.2 | 75.3 | 61.8           | 74.7   |
|                        |      | LGBM       | 72.3             | 83.4                 | 82.8 | 82.0 | 82.5 | 76.5           | 78.5   |
| Accuracy (%)           |      | QDA        | 71.6             | 76.0                 | 73.3 | 79.5 | 76.9 | 48.1           | 75.7   |
|                        |      | LGBM       | 77.0             | 86.1                 | 84.7 | 79.2 | 83.4 | 71.0           | 73.4   |
| F1 (%)                 |      | QDA        | 67.9             | 76.0                 | 73.0 | 78.2 | 76.5 | 48.0           | 75.4   |
|                        |      | LGBM       | 76.2             | 85.4                 | 84.4 | 77.8 | 83.1 | 71.3           | 73.9   |

(a) Beet/Weed detection.

|                        |           | Classifier | Radiance feature | Reflectance features |      |      |      |                |        |
|------------------------|-----------|------------|------------------|----------------------|------|------|------|----------------|--------|
|                        |           |            |                  | Illumination-based   |      |      |      | Training-based | Hybrid |
|                        |           |            |                  | rw                   | wa   | ms   | dwd  | wn             | orw    |
| Per-class accuracy (%) | Beet      | QDA        | 88.5             | 73.9                 | 73.5 | 86.0 | 77.7 | 18.8           | 77.5   |
|                        |           | LGBM       | 78.8             | 87.0                 | 82.1 | 83.6 | 82.5 | 65.7           | 68.0   |
|                        | Thistle   | QDA        | 47.8             | 70.9                 | 68.1 | 75.2 | 55.0 | 55.3           | 62.7   |
|                        |           | LGBM       | 68.3             | 73.4                 | 71.8 | 65.6 | 72.8 | 63.1           | 75.8   |
|                        | Goosefoot | QDA        | 19.1             | 32.2                 | 28.5 | 1.0  | 24.1 | 52.6           | 26.2   |
|                        |           | LGBM       | 23.9             | 35.7                 | 38.1 | 33.7 | 31.5 | 20.0           | 19.9   |
| Accuracy (%)           |           | QDA        | 34.4             | 44.9                 | 41.8 | 26.0 | 37.3 | 47.8           | 40.0   |
|                        |           | LGBM       | 39.5             | 49.7                 | 50.3 | 46.5 | 46.0 | 34.0           | 36.3   |
| F1 (%)                 |           | QDA        | 34.3             | 40.8                 | 40.1 | 25.4 | 33.7 | 27.0           | 37.3   |
|                        |           | LGBM       | 39.7             | 47.1                 | 44.4 | 41.4 | 42.5 | 32.9           | 35.1   |

(b) Beet/thistle/goosefoot identification.

inaccurate reflectance estimations at vegetation pixels, which affects its classification performance.

Let us now compare the classification performances of the reflectance features. The overall classification performance using reflectance features estimated by the proposed hybrid method orw does not outperform those obtained by rw method despite that orw achieves the lowest reflectance estimation errors (see Tab. 4.2 and Fig. 4.6). It provides comparable results with rw method using QDA classifier for weed detection. However its classification performances are degraded with LGBM, where wa and dwd methods perform globally better. The best overall classification results are obtained by our proposed rw method that performs well with both classifiers and reaches the highest average F1 and accuracy scores with LGBM for weed detection (85.4% and 86.1%, respectively). The wa method provides good classification results, often better than those obtained by dwd, although the latter accounts for illumination variations during the frame acquisitions.

Figure 4.12 illustrates the satisfying  $\overline{\text{Accuracy}}$  and  $\overline{\text{F1}}$  scores obtained thanks to the analysis of illumination-based (including the hybrid orw) reflectance features by



LGBM. The figure shows that weed is globally well detected by these methods. For weed identification, the classification performances of all features are degraded, because they provide weak performances on the goosefoot class (see Fig. 4.13). This lack of generalization may be caused by the high within-class dispersion (since we consider vegetation at different growth stages) and/or the physiological vegetation changes.

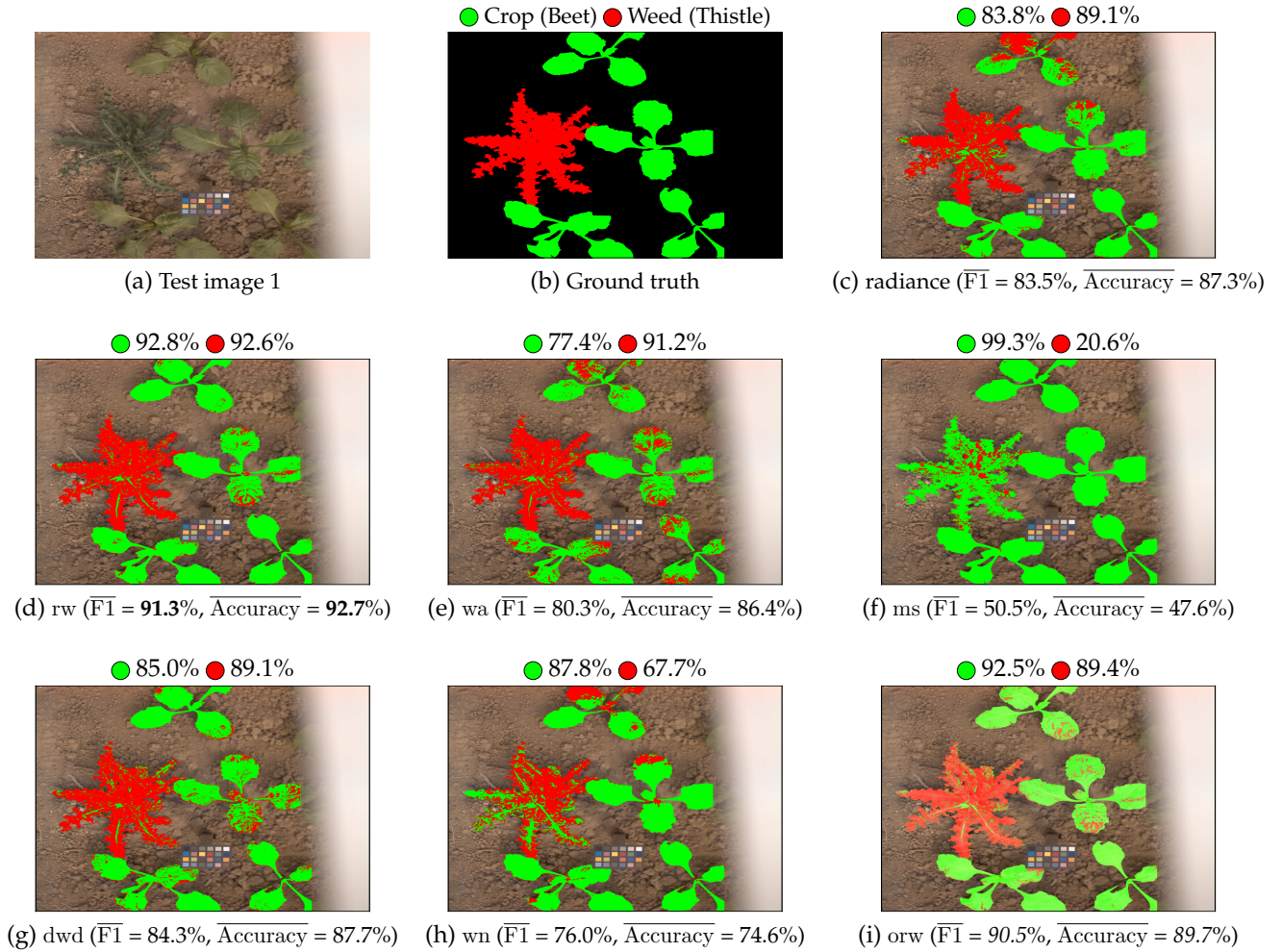


FIGURE 4.12: Crop/weed detection. Beet is displayed as green and weed as red. The per-class accuracy score is displayed near each colored circle (class label) for each considered feature. Bold values show the best results among all methods, italics show the second best ones.

To summarize evaluation results, Tabs. 4.6a (for weed detection) and 4.6b (for weed identification) show the rank  $\text{Rank}_{\diamond,*}$  obtained by each reflectance estimation method \* according to each evaluation criterion  $\diamond$  used in Tabs. 4.2, 4.5a, and 4.5b. The method with the lowest total rank is considered to be the best one, since it satisfies several criteria.



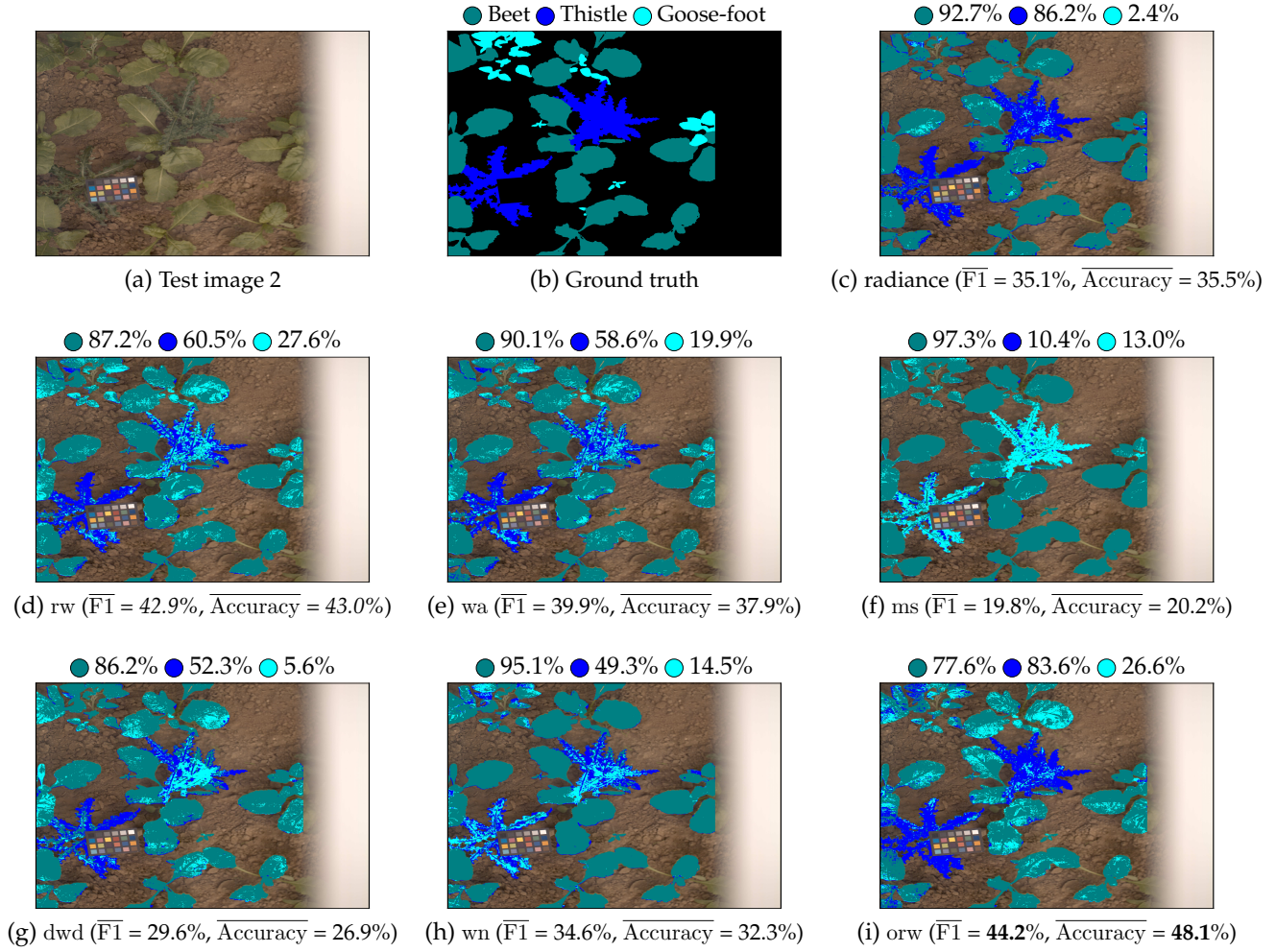


FIGURE 4.13: Beet/thistle/goosefoot identification. Beet is displayed as teal, thistle as blue, and goosefoot as cyan. The per-class accuracy score is displayed near each colored circle (class label) for each considered feature. Bold values show the best results among all methods, italics show the second best ones.

The total ranks of ms and wn methods are the highest ones, because they provide the worst results for either reflectance estimation quality (ms) or classification performance (wn). On the one hand, the dwd method that uses two reference devices to cope with illumination variation provides the second best total rank for weed detection (see Tab. 4.6a). On the other hand, the wa method that uses one reference device, but assumes constant illumination, gives the second best total rank for weed identification (see Tab. 4.6b). Our rw method, which row-wise analyzes one single reference device in order to take account of illumination variations, reaches the best total ranks for both weed detection and identification problems and outperforms orw method that provides the lowest estimated reflectance errors. This suggests that

TABLE 4.6: The ranking of reflectance estimation methods for beet/weed detection (4.6a) and beet/thistle/goosefoot identification (4.6b). Bold shows best result and italics second best one.

|                                                |                                |      | Method |    |    |     |    |     |
|------------------------------------------------|--------------------------------|------|--------|----|----|-----|----|-----|
| Evaluation criterion                           |                                |      | rw     | wa | ms | dwd | wn | orw |
| $\overline{\text{MAE}}_*$ (%)                  |                                |      | 4      | 5  | 6  | 2   | 3  | 1   |
| $\overline{\Delta\theta}_*$ (rad)              |                                |      | 2      | 2  | 6  | 4   | 4  | 1   |
| Beet/weed detection                            | $\overline{\text{Accuracy}}_*$ | QDA  | 3      | 5  | 1  | 2   | 6  | 4   |
|                                                |                                | LGBM | 1      | 2  | 4  | 3   | 6  | 5   |
|                                                | $\overline{\text{F1}}_*$       | QDA  | 3      | 5  | 1  | 2   | 6  | 4   |
|                                                |                                | LGBM | 1      | 2  | 4  | 3   | 6  | 5   |
| $\sum_{\diamond=1}^6 \text{Rank}_{\diamond,*}$ |                                |      | 14     | 21 | 22 | 16  | 31 | 20  |

(a) Beet/weed detection.

| Evaluation criterion                     |                                |                                                | Method |    |    |     |    |     |
|------------------------------------------|--------------------------------|------------------------------------------------|--------|----|----|-----|----|-----|
|                                          |                                |                                                | rw     | wa | ms | dwd | wn | orw |
| $\overline{\text{MAE}}_*$ (%)            |                                |                                                | 4      | 5  | 6  | 2   | 3  | 1   |
| $\Delta\theta_*$ (rad)                   |                                |                                                | 2      | 2  | 6  | 4   | 4  | 1   |
| Beet/thistle/goosefoot<br>identification | $\overline{\text{Accuracy}}_*$ | QDA                                            | 2      | 3  | 6  | 5   | 1  | 4   |
|                                          |                                | LGBM                                           | 2      | 1  | 3  | 4   | 6  | 5   |
|                                          | $\overline{\text{F1}}_*$       | QDA                                            | 1      | 2  | 6  | 4   | 5  | 3   |
|                                          |                                | LGBM                                           | 1      | 2  | 4  | 3   | 6  | 5   |
|                                          |                                | $\sum_{\diamond=1}^6 \text{Rank}_{\diamond,*}$ |        | 12 | 15 | 31  | 22 | 25  |

(b) Beet/thistle/goosefoot identification.

rw-based reflectance features are relevant for crop/weed recognition under variable illumination conditions.

## 4.7 Conclusion

The experiments on our outdoor image database allow us to compare the performances of different reflectance features according to the estimation quality and pixel classification, and to draw the following conclusions:

- (i) Reflectance features are illumination-invariant, therefore more adapted to outdoor classification tasks than radiance features (see Tabs. 4.5a and 4.5b).
- (ii) Illumination-based reflectance estimation methods provide more accurate scene reflectance estimation and achieve better classification performances than the training-based reflectance estimation method wn. Moreover, accounting for illumination variation during the frame acquisition reduces estimated reflectance

errors and improves classification performances. Indeed, as *rw* considers the illumination variation at the frame level, it provides better performances than *wa* method.

- (iii) Estimated reflectance quality and discrimination power are two distinct properties. Better estimation results does not necessarily imply better classification results. Indeed, the hybrid *orw* method provides the lowest reflectance estimation errors, but *rw* method performs better for classification.

In the next chapter, we use *rw*-based reflectance images to further assess the problem of crop/weed recognition. Specifically, we focus on the problem of dimension reduction and band selection.

# Dimension reduction

## Contents

---

|            |                                                                         |            |
|------------|-------------------------------------------------------------------------|------------|
| <b>5.1</b> | <b>Introduction</b>                                                     | <b>108</b> |
| <b>5.2</b> | <b>Feature transform</b>                                                | <b>108</b> |
| 5.2.1      | Data-driven methods                                                     | 109        |
| 5.2.2      | SSF-based method                                                        | 110        |
| <b>5.3</b> | <b><math>K</math>-dimensional vs. RGB-NIR reflectance feature space</b> | <b>111</b> |
| 5.3.1      | From $K$ -dimensional to RGB-NIR feature space                          | 112        |
| 5.3.2      | Learning and test pixel extraction                                      | 113        |
| 5.3.3      | Classification protocol                                                 | 113        |
| 5.3.4      | Results                                                                 | 115        |
| 5.3.5      | NIR feature contribution                                                | 117        |
| <b>5.4</b> | <b>Feature selection</b>                                                | <b>118</b> |
| 5.4.1      | State of the art                                                        | 119        |
| 5.4.2      | Band selection                                                          | 121        |
| 5.4.3      | Global band selection                                                   | 124        |
| 5.4.4      | MSFA design                                                             | 125        |
| <b>5.5</b> | <b>Conclusion</b>                                                       | <b>127</b> |

---

## 5.1 Introduction

CHAPTER 4 shows that crops and weeds can be well detected by the analysis of reflectance features extracted from  $K$ -channel ( $K=141$ ) images. However, as the number of channels is high, the analysis (feature extraction and segmentation) of these images is time- and memory-consuming. Therefore, we propose to reduce the spectral dimension of the analyzed 141-channel reflectance images.

To reduce feature space dimension, two main strategies have been widely followed in the computer vision community: *feature transform* and *feature selection* [35]. In Sec. 5.2, we first present state-of-the-art feature transform techniques that are based on the analysis of data distribution. Then we introduce the SSF-based feature transform method. In Sec. 5.3, we evaluate the performance of the transformed reflectance features for crop/weed recognition. In Sec. 5.4, we first present state-of-the-art feature selection methods, then the procedure that we consider. As each feature is associated to a specific spectral band, feature selection can be considered as band selection in our case. The selected bands are finally used to design a specific multispectral camera for outdoor crop/weed recognition.

## 5.2 Feature transform

The goal is to transform the original  $K$ -dimensional feature space into a new  $D$ -dimensional feature space, where  $D \ll K$ . The assumption is that the  $D$  transformed features carry an amount of *information* that is comparable with the original  $K$ -dimensional feature vectors. In a classification context, the discrimination power of the original features could be carried by a relatively small set of transformed features. In Sec. 5.2.1, we first present state-of-the-art data-driven feature transform techniques. Then, in Sec. 5.2.2 we present the spectral sensitivity function (SSF) feature transform method that computes  $D$  features from the  $K$  reflectance ones.

### 5.2.1 Data-driven methods

Several unsupervised and supervised feature transform methods have been proposed in the literature. All of them share a common goal, but the assumptions regarding the information hold by their transformed features differ. One of the most known and used techniques in the field of computer vision is probably the unsupervised linear principal component analysis (PCA) technique [173, pp. 331–332] and its non-linear variant (K-PCA) [173, pp. 351–353]. These two methods assume that most of the *variance* of the data is carried by only few (usually two or three) transformed features laying in a space spanned by uncorrelated orthogonal axes (known as the principal components). K-PCA follows exactly the same procedure as PCA, except that the data points are first projected onto a higher dimensional space using a kernel function such as the radial basis function (RBF), then PCA is applied to reduce the dimension of these feature vectors. Another efficient unsupervised technique that became very popular in the machine learning community is the t-Distributed Stochastic Neighbor Embedding (t-SNE) technique [181]. This method converts Euclidean distances between high dimensional data points into similarities expressed as conditional probabilities. It aims to minimize the Kullback-Leibler divergence between the conditional probabilities computed in the high-dimensional feature space.

The supervised linear discriminant analysis (LDA) [171] is a powerful technique used for both classification and dimension reduction. LDA projects the features onto a lower dimensional space that maximizes the between-class variance  $S_B$  while reducing the within-class variance  $S_W$ . Data points are projected onto the eigenvectors of  $S_W^{-1}S_B$  associated with the largest eigenvalues. To solve non-linear problems by LDA, the kernel trick is used before applying LDA in the same way as for K-PCA. Note that in case of collinear features, assumption about feature independence is violated, and LDA might not be adapted. In this case, the partial least square discriminant analysis (PLS-DA) [162] should be used. PLS-DA is derived from PLS regression methods used to analyze multivariate features (the predictors) when the response vector (the predicted variables) contains categorical values (pixels labels). PLS-DA handles the feature collinearity problem thanks to a specific decomposition of the features and response vector.

Feature transform methods have been successfully applied to analyze multispectral data [39, 89]. However, because the transformed features are obtained by a linear or non-linear combination of the original features, the ability to interpret the data distribution in the original feature space from the transformed feature space is not straightforward and is often lost. In our case, keeping the interpretation ability of the feature space is very important. Furthermore, we seek to find the set of relevant features that provides a good trade-off between computation cost and classification performances.

### 5.2.2 SSF-based method

Rather than analyzing the data distribution, spectral features can be computed from the SSF of optical filters that are embedded in the camera. The Snapscan camera for instance provides a  $K$ -dimensional ( $K = 141$ ) feature vector at each pixel, that can be reduced to a 4-dimensional vector by simulating its acquisition thanks to an RGB-NIR camera. The feature vectors produced in this way can be then interpreted since each of their components is associated to a specific spectral band. We then propose to transform our  $K$ -channel rw-based reflectance images to 4-channel ones thanks to a spectral integration with broad-band SSFs of two different RGB-NIR cameras. The first one is designed by Monno et al. [128] and the second one is manufactured by IMEC<sup>25</sup>.

Because the SSFs are not ideal, spectral crosstalk occurs between SSFs of the RGB filters and that of the NIR filter (see Figs. 5.1(a) and 5.2(a)). To avoid the undesirable effects (such as mixing spectral information from different domains) of crosstalk, Monno et al. [128] add an optical notch (cutoff) filter (see Fig. 5.1(b)) in front of the lens to discard crosstalk effect in the range [660 nm, 800 nm]. Then, they compute a correction matrix to further correct crosstalk at pixel level. To compute features from the SSFs of the RGB-NIR camera of Monno et al. [128], we follow the same procedure as the authors but without any pixel value correction. We only use the notch filter response to discard crosstalk in the range [660 nm, 800 nm]. The remaining responses of the RGB filters in the range [800 nm, 1000 nm] and that of the NIR filter in the range [420 nm, 660 nm] are simply ignored (see Fig. 5.1(d)). Because IMEC RGB-NIR

<sup>25</sup> <https://www.imechyperspectral.com/en/cameras/ximea-snapshot-rgb-nir#specs>

SSFs are less affected by crosstalk (see Fig. 5.2(a)), there is no need to use the notch filter. To compute features, we simply ignore the responses of the RGB filters in the range [680 nm, 1000 nm] and that of the NIR filter in the range [400 nm, 680 nm] (see Fig. 5.2(b)).

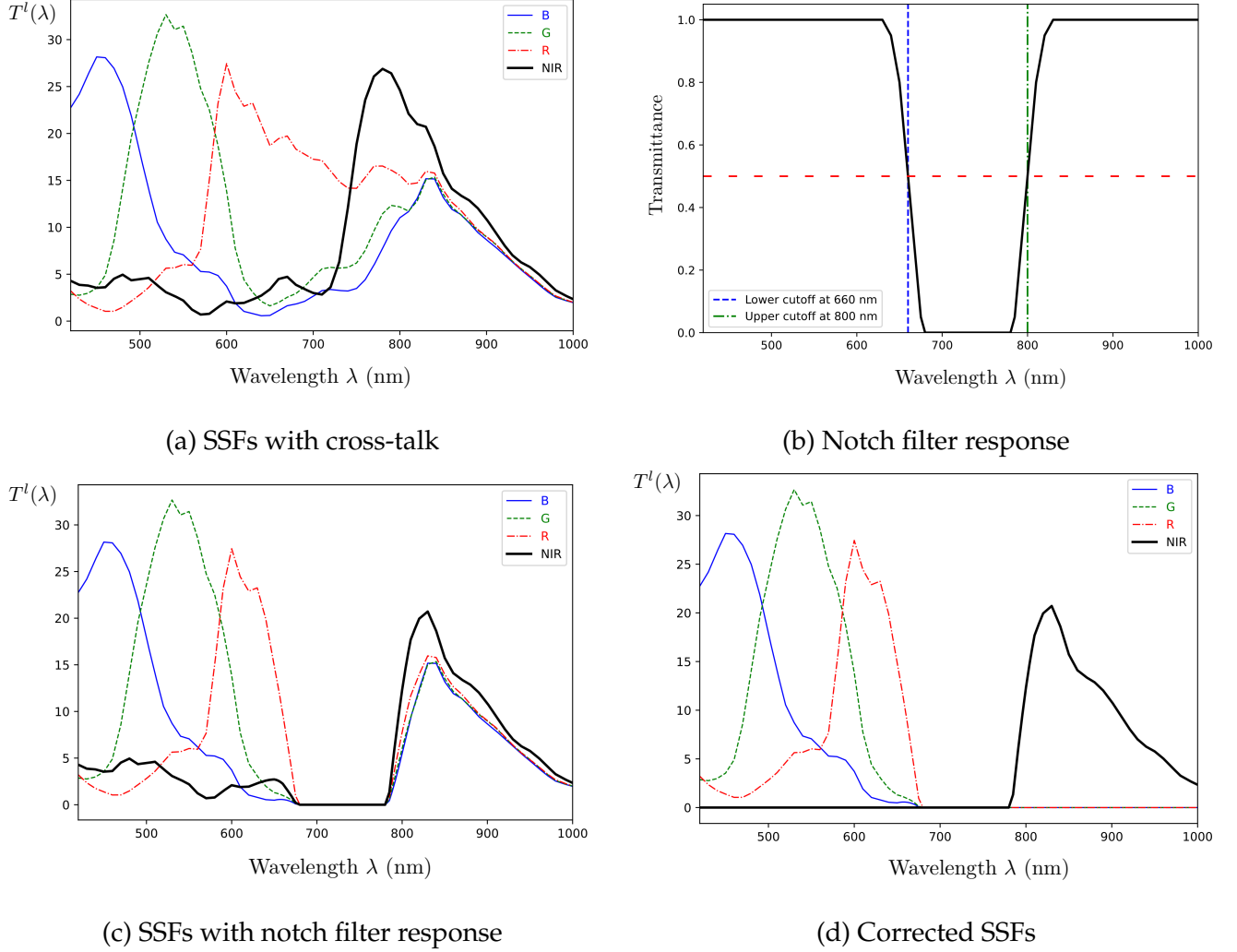


FIGURE 5.1: Filters SSFs and crosstalk correction of Monno et al. [128] RGB-NIR camera.

### 5.3 $K$ -dimensional vs. RGB-NIR reflectance feature space

The transformed features should provide at least as good results as the original features. Below, we propose to compare the classification performance reached by broadband RGB-NIR reflectance feature vectors with those obtained by the original  $K$ -dimensional narrowband feature vectors. In Sec. 5.3.1, we first present our



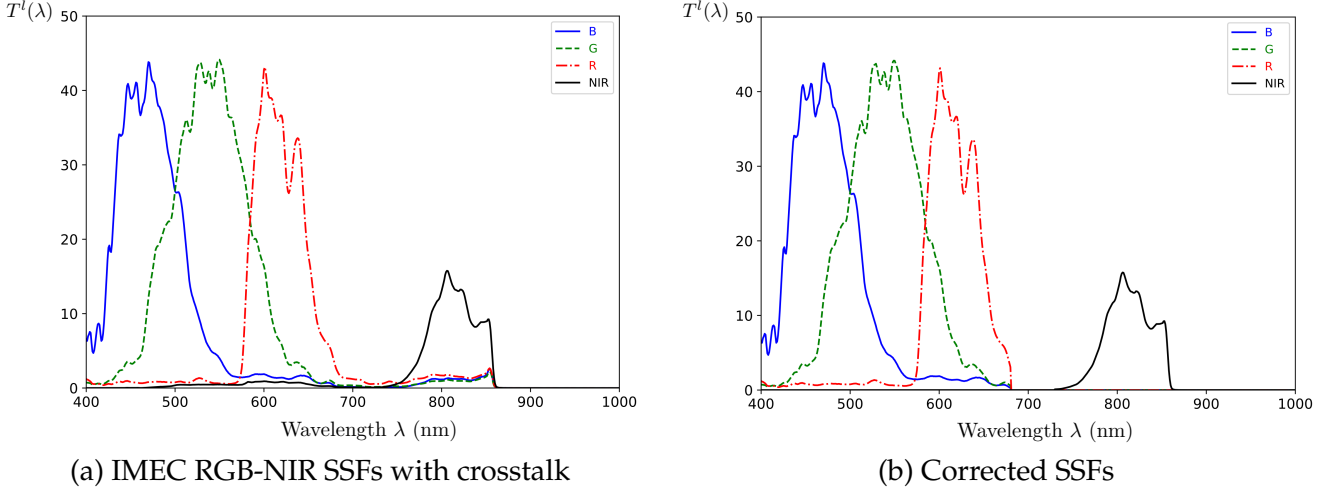


FIGURE 5.2: Filters SSFs and crosstalk correction of IMEC's RGB-NIR camera.

protocol to simulate RGB-NIR reflectance feature vectors from the  $K$ -dimensional ones. Then in Secs. 5.3.2 and 5.3.3, we present the learning and test pixel extraction steps and the classification protocol, respectively. Finally, in Sec. 5.3.4 we present the performance of the crop/weed recognition and identification obtained by each feature.

### 5.3.1 From $K$ -dimensional to RGB-NIR feature space

The outdoor multispectral radiance images provided by our Snapscan camera are built from successive frames acquired under different illumination conditions. Therefore, we cannot deduce RGB-NIR radiance thanks to a spectral integration of radiance spectra contained in  $\mathbf{I}^{(K)}$ . We thus simulate the formation of a 4-channel reflectance image  $\mathbf{R}_{\text{rw}}^{(4)}$  from an estimated  $K$ -channel reflectance image  $\hat{\mathbf{R}}_{\text{rw}}^{(K)}$  (see Eq. (3.23)). A  $K$ -channel rw-based reflectance image is converted to a 4-channel one by:

$$R_{\text{rw},p}^l = \sum_{k=0}^{K-1} R_{\text{rw},p}^k \cdot T^l(\lambda^k), \quad (5.1)$$

where  $T^l(\lambda^k)$  is the sensitivity value for spectral band  $k$  of the RGB-NIR camera filter  $l \in \llbracket 0, 3 \rrbracket$  centered at  $\lambda^l$ , normalized as:

$$T^l(\lambda^k) = \frac{T^l(\lambda^k)}{\sum_{k=0}^{K-1} T^l(\lambda^k)}. \quad (5.2)$$

### 5.3.2 Learning and test pixel extraction

For classification experiments, we target three crop (beet, wheat, and bean) and three weed species (thistle, goosefoot, and datura). We consider three detection problems (beet vs. weeds, wheat vs. weeds, and bean vs. weeds) and three identification problems (crop/thistle/goosefoot/datura for either of the three crop species).

To assess each classification problem, specific learning ( $\mathcal{S}^{learn}$ ) and test ( $\mathcal{S}^{test}$ ) sets are formed so that  $\mathcal{S}^{learn}$  contains learning images from which learning vegetation pixels are extracted while all vegetation pixels of  $\mathcal{S}^{test}$  are used as test pixels. For each detection problem and for a given class  $\mathcal{C}^i$ , the number of learning pixels per image depends on the number of images (occurrences) where  $\mathcal{C}^i$  is represented in  $\mathcal{S}^{learn}$ . For crop/weed detection problems, we extract  $2 \cdot \mathcal{N}^l$  learning pixels,  $\mathcal{N}^l$  for crop class, and  $\mathcal{N}^l$  for weed class ( $\mathcal{N}^l = 300,000$  pixels in these experiments). As we merge thistle, goosefoot, and datura prototype pixels to build a single weed class, we extract  $\mathcal{N}^l/3$  learning pixels for each of thistle, goosefoot, and datura classes.

Note that since we do not compare the robustness of reflectance estimation methods against illumination like in Sec. 4.6, we only focus on detection/identification performances here. Therefore,  $\mathcal{S}^{learn}$  and  $\mathcal{S}^{test}$  contain images that may be acquired on the same day.

Tables 5.1a–5.1c show the number of learning and test images, and the number of extracted learning and test pixels per class for each crop/weed detection and identification problem. All the available pixels in  $\mathcal{S}^{test}$  are used to assess the generalization power of a supervised classifier.

### 5.3.3 Classification protocol

Row-wise reflectance estimation is based on assumption about illumination spatial uniformity in each pixel row (see Sec. 3.4.3). However, illumination may vary along each pixel row acquired by the Snapscan. To overcome illumination non-uniformity

TABLE 5.1: Number of learning and test pixels for crop/weeds detection (left sub-columns) and crop/thistle/goosefoot/datura identification (right sub-columns).

|                       |           | $\mathcal{S}^{learn}$ (47 images) |                                 |                    | $\mathcal{S}^{test}$ (25 images) |                        |           |
|-----------------------|-----------|-----------------------------------|---------------------------------|--------------------|----------------------------------|------------------------|-----------|
| Class $\mathcal{C}^i$ |           | #Occurrences                      | #Learning pixels per occurrence |                    | #Occurrences                     | #Test pixels per class |           |
| Crop                  | Beet      | 22                                | $\mathcal{N}^l/22$              |                    | 16                               | 9,016,280              |           |
| Weed                  | Thistle   | 11                                | $(\mathcal{N}^l/3)/11$          | $\mathcal{N}^l/11$ | 10                               | 12,659,599             | 6,371,760 |
|                       | Goosefoot | 15                                | $(\mathcal{N}^l/3)/15$          | $\mathcal{N}^l/15$ | 6                                |                        | 3,856,704 |
|                       | Datura    | 17                                | $(\mathcal{N}^l/3)/17$          | $\mathcal{N}^l/17$ | 6                                |                        | 2,431,135 |

(a) Beet crop

|                       |           | $\mathcal{S}^{learn}$ (49 images) |                                 |                    | $\mathcal{S}^{test}$ (18 images) |                        |           |
|-----------------------|-----------|-----------------------------------|---------------------------------|--------------------|----------------------------------|------------------------|-----------|
| Class $\mathcal{C}^i$ |           | #Occurrences                      | #Learning pixels per occurrence |                    | #Occurrences                     | #Test pixels per class |           |
| Crop                  | Wheat     | 5                                 | $\mathcal{N}^l/5$               |                    | 4                                | 8,210,142              |           |
| Weed                  | Thistle   | 13                                | $(\mathcal{N}^l/3)/13$          | $\mathcal{N}^l/13$ | 6                                | 7,838,817              | 3,962,203 |
|                       | Goosefoot | 17                                | $(\mathcal{N}^l/3)/17$          | $\mathcal{N}^l/17$ | 4                                |                        | 1,445,479 |
|                       | Datura    | 17                                | $(\mathcal{N}^l/3)/17$          | $\mathcal{N}^l/17$ | 6                                |                        | 2,431,135 |

(b) Wheat crop

|                       |           | $\mathcal{S}^{learn}$ (54 images) |                                 |                    | $\mathcal{S}^{test}$ (18 images) |                        |           |
|-----------------------|-----------|-----------------------------------|---------------------------------|--------------------|----------------------------------|------------------------|-----------|
| Class $\mathcal{C}^i$ |           | #Occurrences                      | #Learning pixels per occurrence |                    | #Occurrences                     | #Test pixels per class |           |
| Crop                  | Bean      | 23                                | $\mathcal{N}^l/5$               |                    | 5                                | 2,856,367              |           |
| Weed                  | Thistle   | 11                                | $(\mathcal{N}^l/3)/11$          | $\mathcal{N}^l/11$ | 9                                | 8,858,699              | 4,982,085 |
|                       | Goosefoot | 17                                | $(\mathcal{N}^l/3)/17$          | $\mathcal{N}^l/17$ | 4                                |                        | 1,445,479 |
|                       | Datura    | 17                                | $(\mathcal{N}^l/3)/17$          | $\mathcal{N}^l/17$ | 6                                |                        | 2,431,135 |

(c) Bean crop

issues and improve classification performances, several authors propose to normalize reflectance signatures to make them robust against shading and specular reflection [79, 168]. We hence normalize estimated/simulated ( $K$ -channel/4-channel) reflectance images at each pixel so that reflectance energy sums up to 1:

$$\bar{R}_{rw,p}^k = \frac{R_{rw,p}^k}{\sum_{i=0}^{K-1} R_{rw,p}^i}, \quad (5.3)$$

and

$$\bar{R}_{rw,p}^l = \frac{R_{rw,p}^l}{\sum_{i=0}^3 R_{rw,p}^i}. \quad (5.4)$$

Additionally, to reduce sparse pixel misclassification, we assume that reflectance across locally close surface elements of a scene does (almost) not change. Thus, these elements most likely belong to the same material, hence to the same class. Each prediction associated to a test pixel is then filtered using a majority voting rule. The final class label of a test vegetation pixel is the most frequent label over its  $7\text{px} \times 7\text{px}$  neighborhood. We compare the classification performances reached by

analyzing the  $K$ -channel narrowband reflectance ( $\hat{\mathbf{R}}_{\text{rw}}^{(K)}$  and  $\bar{\mathbf{R}}_{\text{rw}}^{(K)}$ ) against RGB-NIR broad-band reflectance ( $\mathbf{R}'_{\text{rw}}^{(4)}$  and  $\bar{\mathbf{R}}'_{\text{rw}}^{(4)}$ ) with LGBM classifier.

### 5.3.4 Results

Table 5.2 shows the classification results obtained for each feature, and Fig. 5.3 shows the color-coded pixel classification results of three different test images.

TABLE 5.2: 141-dimensional (narrowband) reflectance vs. RGB-NIR (broadband) reflectance feature vectors based on Monno et al. [128] (left sub-column) and IMEC (right sub-column) RGB-NIR SSFs. Bold and italicized values show the best and second best performance for each problem and metric.

|       |                            | Weed detection                         |                                        |                                 |      |                                       |      | Weed identification                    |                                        |                                 |      |                                       |      |
|-------|----------------------------|----------------------------------------|----------------------------------------|---------------------------------|------|---------------------------------------|------|----------------------------------------|----------------------------------------|---------------------------------|------|---------------------------------------|------|
|       |                            | Multispectral                          |                                        | RGB-NIR                         |      |                                       |      | Multispectral                          |                                        | RGB-NIR                         |      |                                       |      |
| Crop  | Metric                     | $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\mathbf{R}'_{\text{rw}}^{(4)}$ |      | $\bar{\mathbf{R}}'_{\text{rw}}^{(4)}$ |      | $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\mathbf{R}'_{\text{rw}}^{(4)}$ |      | $\bar{\mathbf{R}}'_{\text{rw}}^{(4)}$ |      |
| Beet  | Accuracy(%)                | 86.3                                   | <b>89.2</b>                            | 75.1                            | 75.4 | 75.6                                  | 75.5 | 59.3                                   | <b>68.6</b>                            | 53.7                            | 54.0 | 52.5                                  | 53.2 |
|       | $\overline{\text{F1}}$ (%) | 87.0                                   | <b>89.7</b>                            | 77.5                            | 77.7 | 78.2                                  | 78.1 | 49.9                                   | <b>64.0</b>                            | 40.8                            | 41.0 | 45.2                                  | 45.7 |
| Wheat | Accuracy(%)                | 90.2                                   | <b>94.1</b>                            | 78.6                            | 78.9 | 82.9                                  | 83.2 | 52.9                                   | <b>58.3</b>                            | 42.9                            | 43.6 | 45.3                                  | 45.8 |
|       | $\overline{\text{F1}}$ (%) | 90.4                                   | <b>94.2</b>                            | 78.4                            | 78.6 | 82.9                                  | 83.3 | 52.1                                   | <b>55.9</b>                            | 42.4                            | 43.2 | 41.6                                  | 42.2 |
| Bean  | Accuracy(%)                | 73.3                                   | <b>73.8</b>                            | 68.4                            | 68.3 | 67.1                                  | 67.3 | 48.9                                   | <b>54.4</b>                            | 43.4                            | 44.0 | 43.9                                  | 44.1 |
|       | $\overline{\text{F1}}$ (%) | 64.0                                   | <b>67.4</b>                            | 56.8                            | 56.8 | 57.5                                  | 56.9 | 47.3                                   | <b>51.0</b>                            | 42.2                            | 42.7 | 42.1                                  | 42.4 |

Let us first compare the classification results (weighted accuracy and F1 scores) obtained by narrow-band (141-dimensional) reflectance feature vectors ( $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ ) and RGB-NIR broad-band (4-dimensional) reflectance feature vectors ( $\mathbf{R}'_{\text{rw}}^{(4)}$ ). We can see that  $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ -based features provide better results than those obtained by RGB-NIR reflectance features for both problems of weed detection and identification. Normalizing reflectance features improves all results except for some RGB-NIR-based crop/weed identification (whatever the crop) and bean/weed detection cases. The normalized 141-dimensional reflectance feature vectors  $\bar{\mathbf{R}}_{\text{rw}}^{(141)}$  provide the best results because they are robust against variation and spatial non-uniformity of illumination. Note that beet/weed identification results obtained with both unnormalized and normalized  $K$ -dimensional reflectance vectors are improved in this experiment, with regard to that of Sec. 4.6.5.2 (see Tab. 4.5b). This is probably because the classifier here learns spectral signatures from more images, which improves its generalization capabilities.

Reducing the dimension of multispectral reflectance images from the SSFs of an RGB-NIR camera provides interpretable features since each feature is associated to a

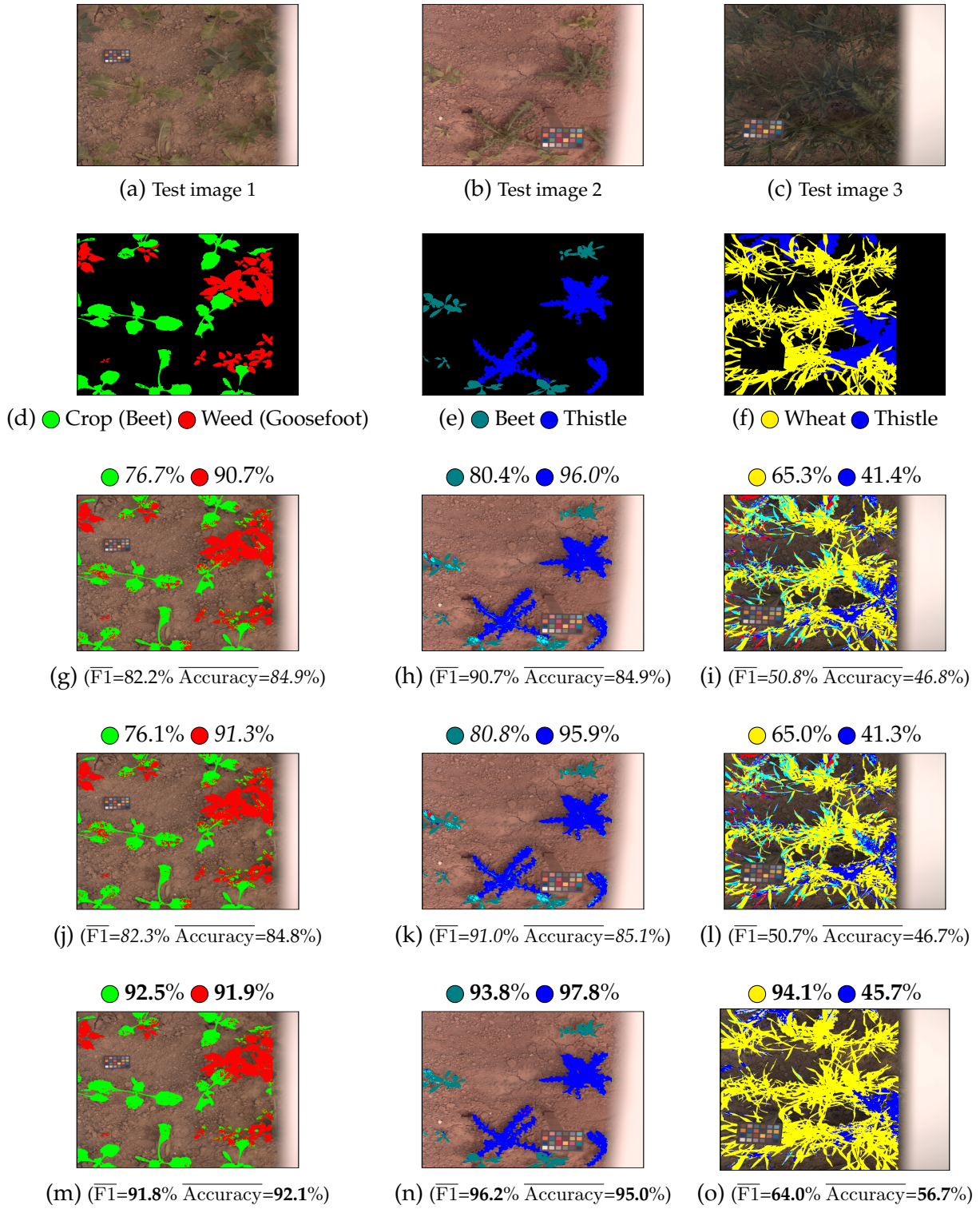


FIGURE 5.3: Segmentation results obtained by LightGBM classifier trained with  $\mathbf{R}_{rw}^{(4)}$  (based on Monno et al. [128] (third row) and IMEC (fourth row) RGB-NIR SSFs) and with  $\mathbf{R}_{rw}^{(K)}$  (fifth row) reflectance features. (a)–(c): RGB renderings of three multispectral test images, (d)–(f): ground truths, (g, j, m): weed detection results, (h, i, k, l, n, o): weed identification results. The per-class accuracy score is displayed near each colored circle (class label) for each considered feature. Bold values show best results and italics second best one for each problem and metric.

specific spectral band. However, as for several state-of-the-art dimension reduction techniques, these features are generated by linearly combining (or merging) the original features according to the RGB-NIR SSFs. Therefore, some discriminant spectral information may be lost in this merging process. In order to reduce the spectral dimension of our images in a more effective way, it is worthwhile to investigate feature selection techniques. In the following, we first propose to assess the contribution of NIR features to classification performances.

### 5.3.5 NIR feature contribution

In Sec. 5.3, we show that  $K$ -dimensional narrowband reflectance feature vectors outperform RGB-NIR broadband ones. However, because NIR spectral bands are highly correlated (see Fig. 5.4), several NIR features may be redundant and thus not useful to discriminate classes. Therefore, we assess the contribution of NIR features to the classification performance and filter out those that do not improve classification accuracy. For this purpose, we consider the crop/weed identification problem since it is more complex than weed detection. We assume that features that are relevant for weed identification will be also relevant for weed detection. For each considered identification problem (beet, wheat, or bean vs. thistle/goosefoot/datura), we use a learning subset to select features and a test subset to assess the classification performance. For computation efficiency, we randomly extract 40,000 learning pixels and 40,000 validation pixels (10,000 per class) from the available learning pixels (see Tabs. 5.1a–5.1c).

We follow a sequential procedure to assess the contribution of each feature. At each iteration step, the classifier is trained with a feature subset whose size is the iteration index  $d$ . At the first iteration, the classifier is trained with the reflectance feature associated to the first spectral band (centered at  $\lambda^{k=0}$ ). At the second iteration, the classifier is trained with reflectance features associated to the first and second spectral bands (centered at  $\lambda^{k=0}$  and  $\lambda^{k=1}$ ), and so on till iteration  $K = 140$  is reached. This first assessment allows us to determine the number of features that are necessary to reach classification performances similar to those obtained using all the available  $K$  features. The features that do not improve the classification performances over the



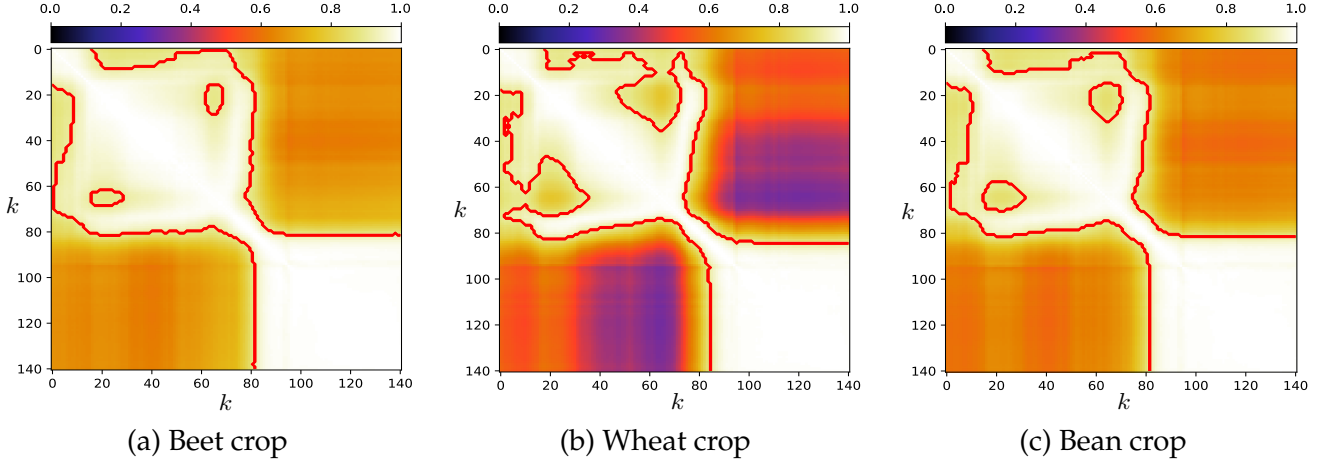


FIGURE 5.4: Color-coded *Pearson* correlation values between reflectance features associated to spectral bands centered at  $\lambda^k$  and estimated at 40,000 (10,000 per class) randomly selected learning pixels used for crop/thistle/goosefoot/datura identification problems. The value 1.0 (white color) indicates that two features are totally correlated. The red contours delimit regions where correlation is above 0.9. Note that channel index  $k = 75$  delimits the VIS and NIR domains.

successive iterations are discarded to provide a first reduced  $K'$ -dimensional feature space.

Figure 5.5 shows the classification results of the validation pixels obtained by LGBM classifier trained with rw-based reflectance features. We can see that, for the three considered problems, once the classifier analyzes 100 features, classification accuracy no longer (or barely) increases using the remaining NIR spectral features because they are highly correlated (see Fig. 5.4). Therefore, spectral bands whose centers are in the range  $[\lambda^{k=100} = 782.6 \text{ nm}, \lambda^{k=140} = 901.6 \text{ nm}]$  can be dropped and only those whose centers are in the range  $[\lambda^{k=0} = 475.1 \text{ nm}, \lambda^{k=99} = 779.6 \text{ nm}]$  should be kept. We denote this first feature subset as  $\mathcal{B}^{K'}$ . In the following, we further reduce the dimension of the feature space to  $D \ll K'$ .

## 5.4 Feature selection

In contrast to feature transform methods, feature selection ones do not transform the original features. They provide a subset of relevant features that are selected from the original feature set according to a performance criterion. This section focuses on the problem of feature selection for crop/weed recognition. As each reflectance feature is associated to a specific spectral band, feature selection can be also formulated

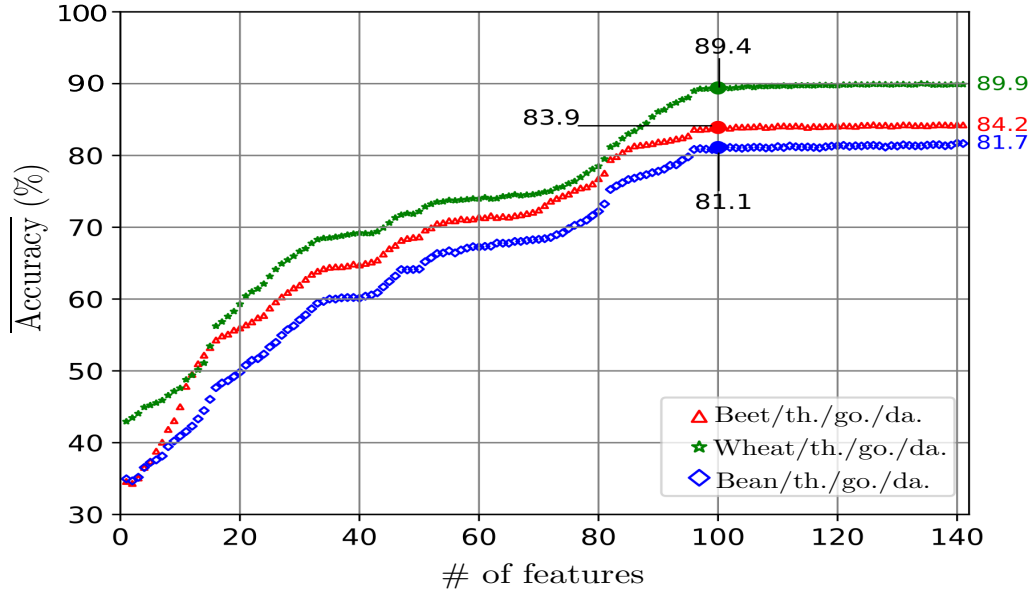


FIGURE 5.5: Weighted classification accuracy obtained by LGBM classifier sequentially trained with larger and larger feature subsets on (40,000) validation pixels.

as band selection. The spectral bands associated to the selected features will be used to design a camera dedicated to our crop/weed recognition problem. In Sec. 5.4.1, we present state-of-the-art feature selection methods. Then in Sec. 5.4.2.1, we introduce the method that we apply for feature selection. In Sec. 5.4.2.2, we first select the best (problem-specific) spectral bands for each crop/thistle/goosefoot/datura identification problems. Then in Sec. 5.4.3, we use the problem-specific bands to select the best global bands that are adapted to the three identification problems. Finally in Sec. 5.4.4, we use the selected global bands to design two multispectral filter arrays for our crop/weed detection/identification problems.

### 5.4.1 State of the art

Feature selection methods can be categorized into filter, wrapper, and embedded methods. Filter methods rely only on data distribution in the feature space and can run in unsupervised, semi-supervised, or fully supervised learning context. The features are ranked according to a score given by a specific evaluation criterion. Some of these criteria are correlation [189], independence [117], or feature ability to discriminate the targeted classes [65]. Features with the highest ranks are selected.

Wrapper methods rely on a supervised classifier performance to select a relevant set



of features. A given wrapper method first generates a subset of features and evaluates the classification performances reached by analyzing the selected features. This workflow is repeated until a stopping criterion is reached. The feature subset that provides the highest classification performance is selected. Despite the relevance of the features provided by wrapper methods, they depend on the classifier. Moreover, when the desired number of features is large, the exhaustive search of the optimal feature subset becomes impractical.

Embedded methods combine both filter and wrapper strategies by including filter selection steps into learning model [98]. These methods inherit the merits of both filter and wrapper methods. They require far less computational efforts since the search strategy is no more exhaustive. They usually find the optimal subset of features.

Our feature selection problem can be referred to as *spectral band selection* or even *channel selection* because the features are spectral information associated to specific narrow bands in the VIS-NIR spectrum. Let  $\mathcal{B}^{(K')}$  be the set of bands sampled by  $K'$  optical filters. Band selection consists in finding a relevant subset  $\mathcal{B}^{(D)} \subset \mathcal{B}^{(K')}$ , with  $D \ll K'$ , that maximizes an objective function. In the spirit of LDA for supervised feature transform, feature selection based on Fisher score aims to find a feature space where the between-class separation is maximized while the within-class variance is minimized. The Fisher score is computed for each feature subset and the  $D$  features with the highest scores are selected.

The supervised minimum-redundancy maximal-relevance (mRMR) criterion [139] estimates relevance and redundancy between features. First, mRMR finds the feature that maximizes the mutual information with the class prototypes. Then, it selects the feature that maximizes the mutual information with the class prototypes and minimizes the mutual information with the already selected feature(s) (minimal correlation). This process is repeated until  $D$  features are selected. Note that one also may use different criteria such as the F-test and Pearson correlation to quantify relevance and redundancy.

To select features in a supervised manner, one also can use tree-based models. A feature is given an importance degree based on the number of times that this feature is

used to split the data (split count) across all trees or on the training loss reduction (information gain) obtained when using this feature. As the importance attributed for each feature may be inconsistent if the model is re-trained with additional samples, more advanced techniques such as SHAP (SHapley Additive exPlanations) [113] can be applied to provide a more accurate and consistent explanation of tree-based classification models.

### 5.4.2 Band selection

In our case, the problem of feature selection can be formalized according to two different ways that are strongly related. From a machine learning standpoint, the selected  $D$  features must provide at least as good performance as the available  $K'$  features for the considered classification problems. From a camera-based perspective, it can be seen as a band (or filter) selection problem. Because the analyzed features are associated to specific spectral bands, the selected features correspond to the  $D$  optical filters of an MSFA. The snapshot camera equipped with this MSFA samples the desired  $D$  spectral bands and provides a  $D$ -channel multispectral image after demosaicing. As the size of an MSFA basic square pattern can be  $3 \times 3$ ,  $4 \times 4$ , or  $5 \times 5$  (larger basic patterns are less common due to demosaicing complexity and/or technological constraints), we propose to select  $D \in \{9, 16, 25\}$  features among the available  $K'$  ones.

In Sec. 5.4.2.1, we present the Sequential Forward Selection (SFS) procedure. In Sec. 5.4.2.2, we first analyze each identification problem (beet vs. weeds, wheat vs. weeds, and bean vs. weeds) separately to select  $D \in \{9, 16, 25\}$  problem-specific features. Then in Sec. 5.4.3, we analyze the problem-specific features in order to find global features that are adapted to the three problems. The performance of the classifier obtained with the selected global features over all test pixels in our database is demonstrated.

#### 5.4.2.1 Sequential Forward Selection (SFS)

SFS is a supervised greedy search algorithm that selects features from the accuracy reached by a classifier. It takes all the features as input and provides the best feature subset at each iteration  $d$ . We set the maximum number of iterations to 100 so that

all dimensions are evaluated. The objective function to maximize is the weighted accuracy score. At the first iteration, the feature that provides the highest accuracy score on the validation subset (of 40,000 pixels) with LGBM classifier is retained. At the next iterations, the feature that improves the classification accuracy score at most when it is combined with the already selected feature(s) is retained, until the last iteration. Although SFS does not consider correlation between features in its selection procedure, it can provide highly discriminant features, especially when the optimal subset search is not large.

The main steps of SFS procedure are:

- **Input:**  $\mathcal{B}^{(K')} = \{f_1, \dots, f_{K'}\}$ , the available set of  $K'$  features.

1 Initialization:  $\mathcal{B}^{(0)} = \emptyset$ , and  $d = 0$ .

2 Inclusion:

$$f_{k'}^+ = \arg \max_{f_{k'} \in \mathcal{B}^{(K')} - \mathcal{B}^{(d)}} (\overline{\text{Accuracy}}(\mathcal{B}^{(d)} + f_{k'})).$$

$$\mathcal{B}^{(d+1)} = \mathcal{B}^{(d)} + f_{k'}^+.$$

$$d = d + 1.$$

3 Repeat step 2 until  $D$  features are selected.

- **Output:** The selected feature subset  $\mathcal{B}^{(D)}$ .

#### 5.4.2.2 Problem-specific band selection

We apply the SFS procedure described above to select the most discriminant features for each identification problem. LGBM classifier analyzes the same learning (40,000 pixels, 10,000/class) and validation pixels (40,000 pixels, 10,000/class) as in Sec. 5.3.5.

Figure 5.6 shows the classification accuracy of the test pixels for each identification problem obtained by LGBM classifier trained with each selected  $d$ -dimensional feature space. We can see that classification performances barely improves beyond  $d = 40$  selected features. We can also notice that with 9 selected features, the classifier provides good classification results but does not reach its highest performances. The classification performances are improved using 16 selected features (82.4%, 88.7%, and 79.2% for beet, wheat, and bean vs. weeds, respectively), and still by about 1%

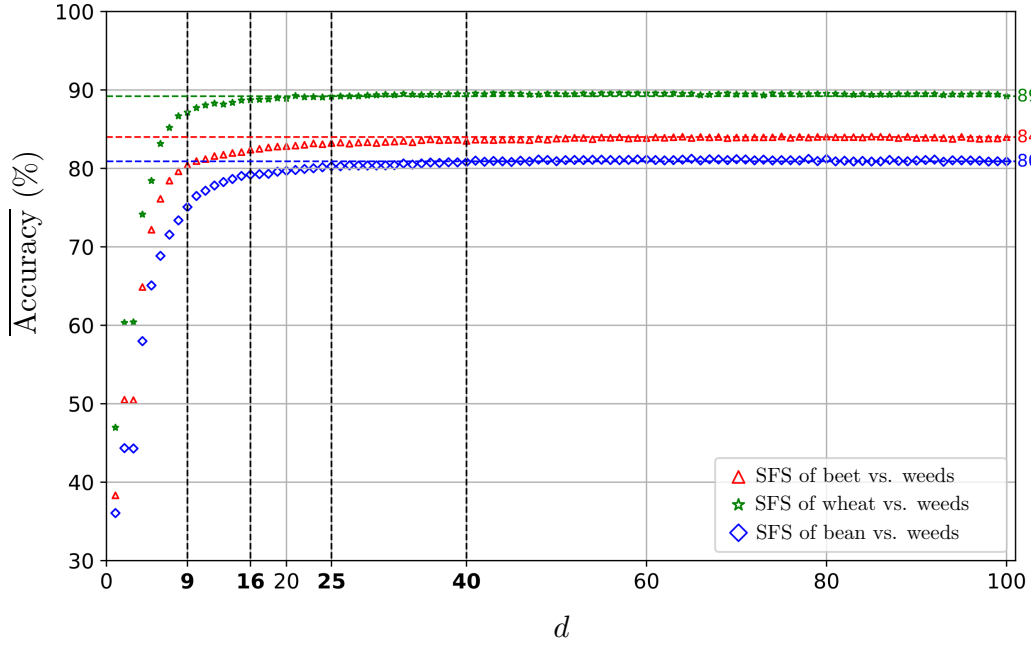


FIGURE 5.6: Classification accuracy of the (40,000, 10,000/class) validation pixels for each identification case obtained by LGBM classifier trained with the  $d$  features selected by SFS, at each iteration, for the three identification problems.

for each problem with 25 selected features, for which LGBM reaches similar performances to those obtained using all features (see colored scores in Fig. 5.6). This experiment shows that when the multispectral camera is designed to sample relevant spectral bands for crop/weed recognition, there is no need to analyze more than 40 spectral bands. In fact  $D = 16$ , or 25 relevant spectral bands are enough to achieve similar classification results to those obtained by the  $K$ -band Snapscan camera. The index  $k$  of the best 25 bands for each identification problem is:

- Beet/thistle/goosefoot/datura:
  - In ascending order of  $d^{26}$ : 23, 93, 98, 2, 44, 79, 95, 84, 62, 15, 5, 25, 77, 69, 46, 85, 94, 53, 78, 39, 28, 18, 99, 32, 74
  - In ascending order of  $k$ : 2, 5, 15, 18, 23, 25, 28, 32, 39, 44, 46, 53, 62, 69, 74, 77, 78, 79, 84, 85, 93, 94, 95, 98, 99
- Wheat/thistle/goosefoot/datura:
  - In ascending order of  $d$ : 79, 93, 99, 85, 2, 46, 95, 23, 65, 84, 76, 16, 96, 74, 44, 25, 64, 75, 94, 73, 80, 86, 90, 51, 15

<sup>26</sup> Snapscan band whose center is  $\lambda^{23}$  is the first one selected by SFS for beet/thistle/goosefoot/datura identification problem.

- In ascending order of  $k$ : 2, 15, 16, 23, 25, 44, 46, 51, 64, 65, 73, 74, 75, 76, 79, 80, 84, 85, 86, 90, 93, 94, 95, 96, 99
- Bean/thistle/goosefoot/datura:
  - In ascending order of  $d$ : 3, 33, 40, 94, 81, 95, 23, 93, 85, 66, 78, 48, 99, 15, 73, 10, 2, 84, 77, 17, 86, 43, 24, 64, 13
  - In ascending order of  $k$ : 2, 3, 10, 13, 15, 17, 23, 24, 33, 40, 43, 48, 64, 66, 73, 77, 78, 81, 84, 85, 86, 93, 94, 95, 99

### 5.4.3 Global band selection

Our aim in this section is to select the bands that are globally the best from the three problem-specific band sets (see Sec. 5.4.2.2). For this purpose, we apply the SFS method and seek for the best features incrementally. For instance, to select the first global band, we consider the first best problem-specific ones ( $k = 23$  for beet crop,  $k = 79$  for wheat, and  $k = 3$  for bean) selected at  $d = 1$  as candidate bands. Then, we train the LGBM classifier with the features associated to these bands and analyze each classification problem individually. The selected global band among  $k = 23$  (retained),  $k = 79$ , and  $k = 3$ , maximizes the average weighted accuracy score over the three classification problems. To find the second global band, the new candidate features are the second best problem-specific features ( $k \in \{33, 93\}$ ) plus the remaining ones that have not been selected at the previous iteration ( $k \in \{3, 79\}$ ). The second best global band maximizes the average weighted accuracy score. This step is repeated until the desired number of features  $D$  is reached.

Figure 5.7 shows the index  $k$  and selection order  $d$  of each of the selected global bands. We can see that the 25 selected bands cover the VIS ( $\lambda^k \in [481.7 \text{ nm}, 698.8 \text{ nm}]$ ) and NIR domains ( $\lambda^k \in [712.3 \text{ nm}, 779.6 \text{ nm}]$ ) of the spectrum. This shows that both VIS and NIR spectral information are important to discriminate the considered species. To validate the discrimination power and relevance of reflectance features associated to the selected global bands for crop/weed detection and identification, we use the same learning and test pixel sets as in Sec. 5.3 (see Tabs. 5.1a–5.1c), and we train the LGBM classifier with the best  $D \in \{9, 16, 25\}$  global features to classify all test pixels. Tables 5.3a and 5.3b show that LGBM classification performances with the selected global features are comparable to those obtained using the  $K = 141$

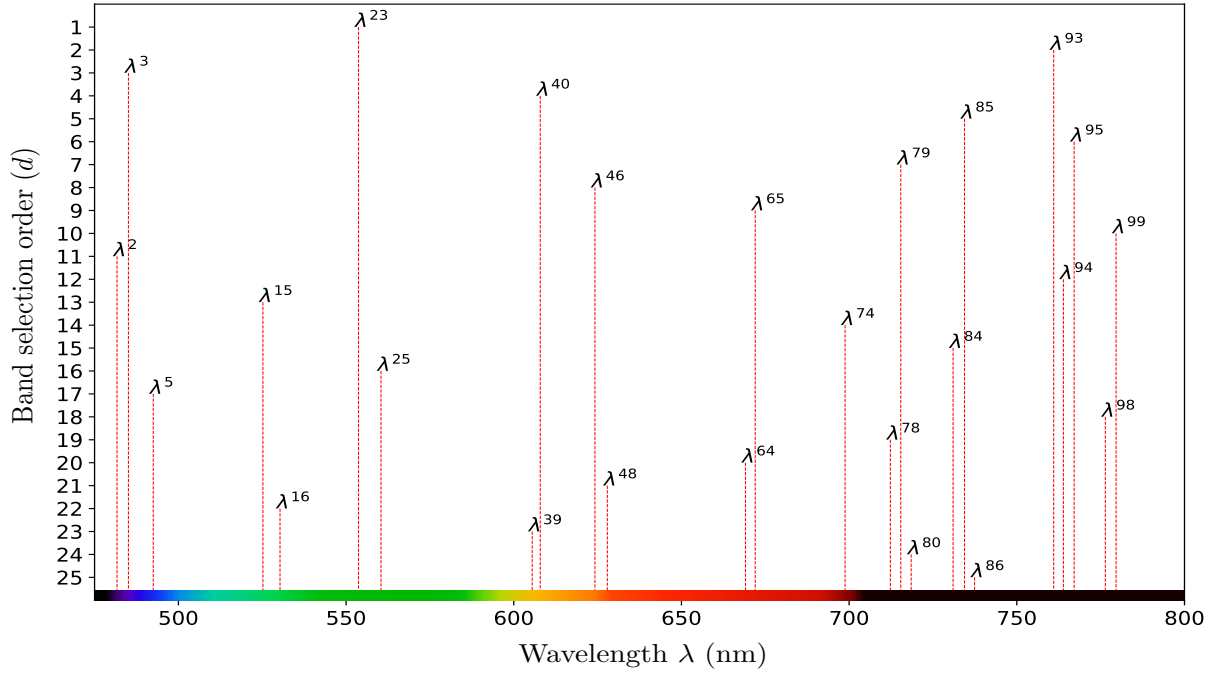


FIGURE 5.7: 25 best selected global bands. Their indices  $k$  (in ascending order of  $d$ ) are: 23, 93, 3, 40, 85, 95, 79, 46, 65, 99, 2, 94, 15, 74, 84, 25, 5, 98, 78, 64, 48, 16, 39, 80, 86.

original features, especially when analyzing 16 and 25 global features. Performances are slightly degraded for bean/weed detection and identification (about 4% loss), and for wheat/thistle/goosefoot/datura identification (about 2% loss). However, performances are similar to those obtained using  $K$  features for beet/weed and wheat/weed detection, and beet/thistle/goosefoot/datura identification, and are even slightly better in some cases with normalized reflectance features. This experiment confirms the results shown in Fig. 5.6 and validates the global features selected by SFS method. This experiment also suggests that spectral signatures extracted from images acquired by a camera that samples  $D = 16$  or  $D = 25$  selected bands can provide similar crop/weed recognition performances to those obtained with  $K$ -band linescan devices.

#### 5.4.4 MSFA design

As mentioned before, our feature selection problem is not only related to dimensionality reduction but also to band selection and camera filter specification.

Indeed, since the selected features are associated to specific narrowband filters, they can be used to design linescan or snapshot cameras dedicated to our crop/weed recognition problems. In this section, we focus on snapshot imaging for crop/weed

TABLE 5.3: Crop/weed detection (5.3a) and identification (5.3b) results obtained with LGBM classifier trained with 25, 16, and 9 best global features. Bold and italicized values show the best and second best performance respectively.

|       |             | Weed detection                         |                                        |                                       |                                       |                                      |                                       |                                       |                                      |
|-------|-------------|----------------------------------------|----------------------------------------|---------------------------------------|---------------------------------------|--------------------------------------|---------------------------------------|---------------------------------------|--------------------------------------|
|       |             | Original features                      |                                        | Selected features                     |                                       |                                      |                                       |                                       |                                      |
| Crop  | Metric      | $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(25)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(16)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(9)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(25)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(16)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(9)}$ |
| Beet  | Accuracy(%) | 86.3                                   | 89.2                                   | 85.8                                  | 85.6                                  | 83.6                                 | <b>89.3</b>                           | 88.9                                  | 86.9                                 |
|       | F1 (%)      | 87.0                                   | <b>89.7</b>                            | 86.6                                  | 86.4                                  | 84.6                                 | 89.5                                  | 89.2                                  | 87.5                                 |
| Wheat | Accuracy(%) | 90.2                                   | <b>94.1</b>                            | 89.3                                  | 89.2                                  | 88.9                                 | <b>94.1</b>                           | 93.9                                  | 93.4                                 |
|       | F1 (%)      | 90.4                                   | <b>94.2</b>                            | 89.4                                  | 89.4                                  | 89.0                                 | <b>94.2</b>                           | 94.0                                  | 93.5                                 |
| Bean  | Accuracy(%) | 73.3                                   | <b>73.8</b>                            | 71.7                                  | 70.5                                  | 69.0                                 | 69.7                                  | 68.0                                  | 65.8                                 |
|       | F1 (%)      | 64.0                                   | <b>67.4</b>                            | 61.8                                  | 61.2                                  | 59.2                                 | 63.1                                  | 61.5                                  | 58.5                                 |

(a) Detection.

|       |             | Weed identification                    |                                        |                                       |                                       |                                      |                                       |                                       |                                      |
|-------|-------------|----------------------------------------|----------------------------------------|---------------------------------------|---------------------------------------|--------------------------------------|---------------------------------------|---------------------------------------|--------------------------------------|
|       |             | Original features                      |                                        | Selected features                     |                                       |                                      |                                       |                                       |                                      |
| Crop  | Metric      | $\hat{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(141)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(25)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(16)}$ | $\hat{\mathbf{R}}_{\text{rw}}^{(9)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(25)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(16)}$ | $\bar{\mathbf{R}}_{\text{rw}}^{(9)}$ |
| Beet  | Accuracy(%) | 59.3                                   | <b>68.6</b>                            | 59.3                                  | 58.7                                  | 57.5                                 | 67.1                                  | 66.5                                  | 64.8                                 |
|       | F1 (%)      | 49.9                                   | 64.0                                   | 50.2                                  | 49.4                                  | 47.7                                 | <b>64.3</b>                           | 63.8                                  | 61.0                                 |
| Wheat | Accuracy(%) | 52.9                                   | <b>58.3</b>                            | 52.0                                  | 52.4                                  | 51.0                                 | 55.8                                  | 56.0                                  | 53.5                                 |
|       | F1 (%)      | 52.1                                   | <b>55.9</b>                            | 51.4                                  | 51.5                                  | 49.9                                 | 53.8                                  | 53.7                                  | 50.7                                 |
| Bean  | Accuracy(%) | 48.9                                   | <b>54.4</b>                            | 48.1                                  | 47.7                                  | 47.0                                 | 51.8                                  | 50.8                                  | 48.7                                 |
|       | F1 (%)      | 47.3                                   | <b>51.0</b>                            | 46.4                                  | 45.9                                  | 45.1                                 | 49.4                                  | 47.9                                  | 45.3                                 |

(b) Identification.

recognition. Most of snapshot cameras use an MSFA (see Fig. 2.4) laid over the sensor to sample the scene radiance. Their design provides the ability to acquire a multispectral radiance image in a single shot. Hence, they are not affected by illumination that may vary during the acquisition of a multispectral image. As the selected  $D = 16$  or  $D = 25$  global bands provide similar classification performances to those obtained with the original  $K = 141$  bands, we propose to use them to design two MSFAs (CA  $4 \times 4$  and CA  $5 \times 5$ ) that are defined by repetition of a  $\sqrt{D} \times \sqrt{D}$  basic pattern that samples  $D = 16$  or  $D = 25$  different bands in the VIS and NIR domains (see Fig. 5.8(c, d)). We follow the MSFA arrangements of two snapshot cameras manufactured by IMEC [60] (see Fig. 5.8(a, b)). In chapter 6, we use these MSFAs to simulate raw images that would be acquired by a snapshot camera equipped with CA  $4 \times 4$  or CA  $5 \times 5$  MSFA. We then propose to analyze raw images for crop/weed recognition.

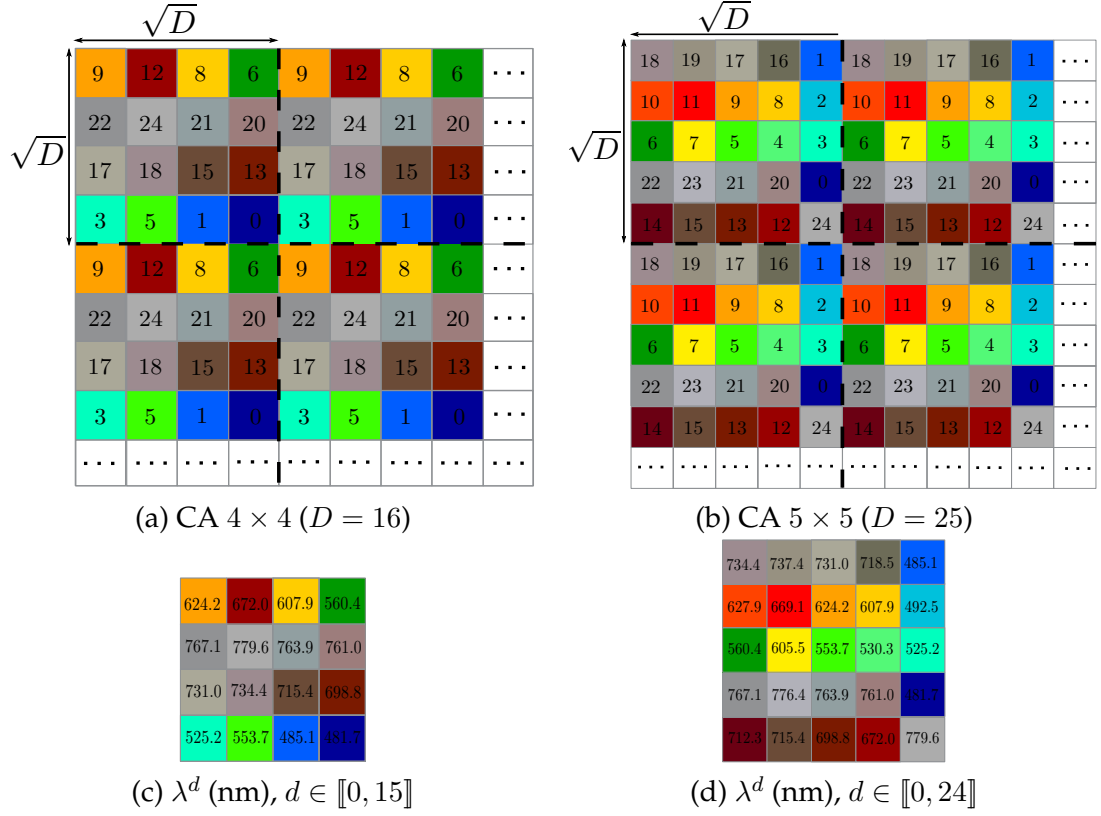


FIGURE 5.8: Considered MSFAs with square basic patterns and no redundant band: (a) CA  $4 \times 4$  and (b) CA  $5 \times 5$ . (c) and (d) show the spectral bands  $\lambda^d$  sampled by each MSFA.

## 5.5 Conclusion

In this chapter, we address the problem of feature space dimension reduction based on feature transform and feature selection approaches. We first present the feature transform approach, its assumptions and principles. We apply a feature transform method based on the SSFs of two RGB-NIR cameras to reduce the dimension of the  $K$ -dimensional reflectance feature space. We show that alternatively to other state-of-the-art methods, features provided by this method are physically interpretable because they are associated to specific spectral bands. However, we also show that the features it provides alter the performance of LGBM classifier for both weed detection and identification problems. As we aim to reduce the dimension of the feature space while keeping a reduced number of physically interpretable features, we focus on a feature selection approach. We therefore analyze reflectance features by applying the SFS method. First, we select problem-specific features that are relevant to discriminate each crop/thistle/goosefoot/datura identification problem. Then, among



these features, we select those that are relevant to all three problems. We show that features associated to the selected global bands are discriminant since LGBM classifier rapidly converges to its highest performance. The best 16 and 25 global bands are finally used to design two MSFAs (CA  $4 \times 4$  and CA  $5 \times 5$ ) for crop/weed recognition.

In chapter 6, we simulate multispectral (raw) images that would be acquired by a snapshot camera equipped with the proposed MSFAs. From these images, we propose to extract texture features to improve the crop/weed recognition performances.

## 6

## CHAPTER

# Raw texture features for crop/weed recognition

## Contents

---

|                                                         |     |
|---------------------------------------------------------|-----|
| <b>6.1 Introduction</b>                                 | 130 |
| <b>6.2 Multispectral texture features</b>               | 130 |
| 6.2.1 Texture features from multispectral images        | 131 |
| 6.2.2 Handcrafted features                              | 132 |
| 6.2.3 Deep learning-based features                      | 133 |
| 6.2.4 MSFA texture features                             | 133 |
| 6.2.5 CNN-based raw texture features                    | 134 |
| <b>6.3 Texture feature for crop/weed classification</b> | 134 |
| 6.3.1 Patch extraction                                  | 135 |
| 6.3.2 Compared texture features                         | 136 |
| 6.3.3 CNN Training                                      | 139 |
| <b>6.4 Classification results and discussion</b>        | 139 |
| 6.4.1 CA $4 \times 4$                                   | 140 |
| 6.4.2 CA $5 \times 5$                                   | 140 |
| 6.4.3 Discussion                                        | 142 |
| <b>6.5 Conclusion</b>                                   | 144 |

---

## 6.1 Introduction

**S**INGLE-sensor multispectral cameras embed an MSFA laid over the sensor, like the widely-used Bayer color filter array in color imaging, to spatio-spectrally sample the incoming radiance according to the photosensor locations. The MSFAs used in this chapter (CA  $4 \times 4$  and CA  $5 \times 5$ , see Sec. 5.4.4) are defined by a repetition of a  $\sqrt{D} \times \sqrt{D}$  basic pattern that samples  $D$  bands. Each filter of the MSFA is sensitive to a specific narrow spectral band, so that each pixel of the acquired raw image represents the radiance sampled around a single band center. Texture features are generally extracted from a fully-defined image that is estimated from the raw image by demosaicing [10, 122]. This step may generate spatio-spectral estimation artifacts, and texture feature extraction becomes computationally inefficient and yields to a high-dimensional feature space as the number of bands increases. Some authors propose to directly process raw images for reflectance estimation [91] or feature extraction [123, 194]. In the same spirit, we here propose to extract texture features for crop/weed recognition directly from raw images. We exploit deep learning advantages and design a convolutional neural network (CNN) that acts as a texture feature descriptor from raw images. Experiments on crop/weed recognition show the relevance of the proposed approach.

In Sec. 6.2, we present the state-of-the-art approaches for multispectral texture feature extraction, then we describe ours that extracts texture features directly from raw images (see Sec. 6.2.5). In Sec. 6.3, we assess the contribution of texture features extracted from raw images to outdoor crop/weed recognition. Experimental results show the relevance of our proposed approach for outdoor crop/weed recognition in Sec. 6.4.

## 6.2 Multispectral texture features

This section focuses on texture features computed from multispectral images. In Sec. 6.2.2, we first present state-of-the-art multispectral texture features based on the handcrafted approach. Then, we present those based on deep learning in Sec. 6.2.3,

and finally the MSFA-based features (in Sec. 6.2.4). In Sec. 6.2.5, we describe our CNN architecture for texture feature computation from raw images.

### 6.2.1 Texture features from multispectral images

To classify texture images provided by a single-sensor snapshot camera that samples  $D$  bands through an MSFA, one usually estimates fully-defined ( $D$ -channel) images from raw images by demosaicing, then applies a specific descriptor to compute texture features. The simplest demosaicing scheme uses a weighted bilinear (WB) interpolation filter to estimate the  $D - 1$  values that miss at each pixel from those available at its neighbors for the same band [25]. Each neighbor is associated with a weight that depends on its spatial distance to the considered pixel. This method only exploits intra-channel spatial correlation to estimate missing values. To improve the estimation, demosaicing should use inter-channel correlation or, if the latter is low, the correlation between each channel and the pseudo-panchromatic image (PPI) [122]. The PPI is first estimated from the raw image thanks to an averaging filter. Then its sharpness is improved using local directional variations of raw values. Finally, the PPI is analyzed by an iterative procedure similar to that of [127] to provide the PPI difference (PPID) demosaicing method. Rathi and Goyal [147] design specific WB interpolation filters based on band occurrence probability in the MSFA. They first estimate an initial fully-defined image using the proposed WB interpolation filters, then use the spectral difference (SD) method to compute sparse band differences between the initial estimation and the raw image. Finally, each sparse channel difference is interpolated using WB method to estimate the final fully-defined image. Antonucci et al. [10] consider demosaicing as a minimization problem and propose to use non-convex matrix completion to estimate the fully-defined image. Specifically, they apply conjugate gradient iterative hard thresholding and alternating steepest descent methods on initial estimations provided either by intensity difference (ID) or SD methods to estimate the fully-defined image. Recently, other studies [13] exploit deep learning advantages to improve demosaicing quality.

### 6.2.2 Handcrafted features

Among texture features, those based on the LBP operator and its variants have been widely used for their robustness against illumination, rotation, and scale [143]. They have also been extended to the multispectral domain using vector approaches [36]. Considering spectral correlation between all bands provides state-of-the-art image classification performance, but is computationally greedy and yields high-dimensional LBP feature vectors that are neither memory efficient nor easily interpretable [123]. The relative spectral difference occurrence matrix (RSDOM) [31] performs spectral differences using the Kullback-Leibler pseudo-divergence measure to extract low-dimensional texture feature vectors from multispectral images as a multi-dimensional probability density function. Zhang et al. [192] propose to compute the vector of locally aggregated descriptors (VLAD) at each pixel by considering its  $3 \times 3$  neighborhood. The reflectance values of neighbors are channel-wise reordered to form different 9-dimensional feature vectors. These feature vectors are then concatenated to form one high ( $9 \times D$ ) dimensional feature vector for each pixel. The dimension of these feature vectors is then reduced to a  $d$ -dimensional one using the random projection technique. Finally, the  $d \ll D$  feature vectors associated to each class are encoded by the VLAD descriptor to build the final texture features, that are  $L_2$ -normalized and PCA-transformed before being fed into a supervised (K-SVM) classifier.

Brown and Ssstrunk [26] propose to extract features from RGB-NIR images using the scale-invariant feature transform (SIFT) descriptor. First, the channels of the input RGB-NIR image are decorrelated using PCA technique. Then, SIFT is applied to each channel separately, and the extracted features are concatenated to form a 512-dimensional feature vector. Its dimension is finally reduced by PCA, which provides the final 128-dimensional feature vector that is used for classification.

Such handcrafted texture features have recently been overshadowed by deep learning techniques based on CNNs [165]. For instance, the SegNet model has been extensively and successfully applied to segment scene images [14], and to analyze multispectral (RGB-NIR) images in the context of weed detection [153].

### 6.2.3 Deep learning-based features

CNNs are commonly used as feature extractors and classifiers. Hidden layers perform feature extraction, and the output one (usually a softmax function in the multi-class case) turns features from the last hidden layer into probabilities for prediction. Some authors propose to use CNNs only as feature extractors. Donahue et al. [41] show that features provided by deep hidden layers, especially the last two ones, are highly discriminant and provide astonishing classification performances when combined with a supervised classifier. Zhou et al. [193] use deep features of the last hidden (fully-connected) layer to train a linear SVM classifier for scene classification. Razavian et al. [149] conduct different recognition tasks (e.g., object detection, visual instance, and fine-grained recognition) using the Overfeat CNN model. In each experiment, features of the first fully-connected layer are  $L_2$ -normalized and used to train an SVM classifier to perform predictions. The 2D-CNN called S-CNN is used to extract features from multispectral images [159, 170]. These features are used by an SVM classifier to perform multispectral band selection and face recognition. This deep scheme outperforms state-of-the-art handcrafted descriptors such as HOG, LBP, and SIFT [159]. In spite of their performances, deep learning-based approaches are greedy in computation time and memory, and using them with high spectral resolution images may be intractable.

### 6.2.4 MSFA texture features

To consider spatio-spectral correlation, some studies directly process raw images [105, 123, 194]. When a descriptor suitably analyzes a raw image, it can achieve similar or even better classification performances than from a demosaiced one because demosaicing may introduce artifacts that alter texture representation. In [123], texture features are directly computed from raw images, which avoids the demosaicing step and provides discriminant features. Specifically, the method analyzes a raw image with respect to the MSFA basic pattern and its band arrangement to build an LBP-based texture descriptor. In the same spirit, we propose a CNN architecture that is adapted to raw images.

### 6.2.5 CNN-based raw texture features

Our CNN architecture called MSFA-Net directly extracts texture features from raw square patches of size  $X \times X$  pixels, where  $X = m \cdot \sqrt{D}$  is a multiple of the MSFA basic pattern width. MSFA-Net is composed of three convolutional blocks, followed by an average pooling layer and two fully-connected layers (see Fig. 6.1). The first convolutional layer is of utmost importance because it guides the feature extraction according to the MSFA basic pattern. It uses 128 convolutional kernels  $\{H_n\}_{n=0}^{127}$  of size  $\sqrt{D} \times \sqrt{D}$  and depth 1 without padding. A specific stride of  $\sqrt{D}$  pixels along both spatial dimensions ensures that each kernel coefficient is always associated to the same MSFA band for all convolutions. This first layer learns spatio-spectral interactions among channel values in each raw patch that matches the basic MSFA pattern. The convolution between a raw patch  $P^{\text{raw}}$  and a kernel  $H_n$ ,  $n \in \llbracket 0, 127 \rrbracket$ , is defined at each pixel  $(x, y) \in \llbracket 0, m-1 \rrbracket^2$  as:

$$O_n(x, y) = \sum_{i=0}^{\sqrt{D}-1} \sum_{j=0}^{\sqrt{D}-1} H_n(i, j) \cdot P^{\text{raw}}(\sqrt{D} \cdot x + i, \sqrt{D} \cdot y + j). \quad (6.1)$$

The resulting 128 feature maps  $\{O_n\}_{n=0}^{127}$  of size  $m \times m$  are fed into the second convolutional block that uses 256 kernels of size  $3 \times 3$  with both a stride and zero-padding of one pixel, such that its input and output feature maps have the same size. The last convolutional block uses 384 kernels of size  $3 \times 3$  with one pixel stride and no padding. Feature maps of the last convolutional layer are usually vectorized using a flattening layer before being fed into fully-connected layers. Following [193], we introduce a global pooling layer to average feature maps channel-wise so that the resulting 384-dimensional feature vector is more robust against noise and spatial translations. To introduce non-linearity and reduce the feature size, a fully-connected layer finally provides the 128-dimensional texture feature vector that is fed into the softmax layer.

## 6.3 Texture feature for crop/weed classification

To assess the contribution of features extracted from raw images for outdoor crop/weed recognition, we use CA  $4 \times 4$  and CA  $5 \times 5$  MSFAs (see Sec. 5.4.4) to simulate raw images from the estimated fully-defined  $D$ -channel reflectance images. In Sec. 6.3.1, we

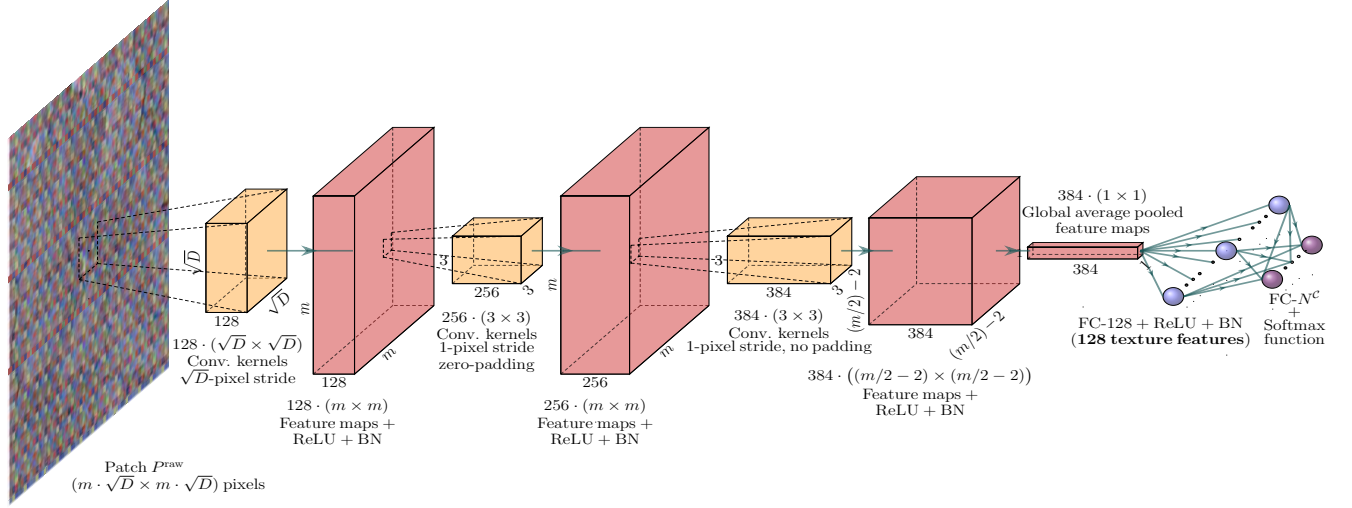


FIGURE 6.1: MSFA-Net architecture. ReLU: rectified linear unit activation, BN: batch normalization, FC: fully-connected layer.

describe how raw patches are obtained and introduce our segmentation approach. In Sec. 6.3.2, we present the considered state-of-the-art descriptors used for feature extraction from either demosaiced or raw patches. In Sec. 6.3.3, we describe the training procedure of the considered deep learning-based descriptors. In Sec. 6.4, we present the experimental results of crop/weed recognition.

### 6.3.1 Patch extraction

We consider the same detection (beet vs. weeds, wheat vs. weeds, and bean vs. weeds) and identification (crop/thistle/goosefoot/datura for either of the three crop species) problems, and use the same learning and test image sets as in Sec. 5.3. Each  $K$ -channel rw-based reflectance image is first transformed into a  $D$ -channel one ( $D = 16$  or  $D = 25$ ) by selecting the channels associated to the selected spectral bands (see Sec. 5.4.3). To obtain raw patches for feature extraction, we simulate raw images that would be acquired by snapshot cameras equipped with CA  $4 \times 4$  and CA  $5 \times 5$  MSFAs, by spatio-spectrally sub-sampling the fully-defined  $D$ -channel images according to CA  $4 \times 4$  or CA  $5 \times 5$  MSFA. To compare with the classical strategy, we also demosaic these MSFA images using PPID as one of the state-of-the-art multispectral demosaicing methods [122]. We denote rw-based reflectance image demosaiced by PPID as  $\hat{\mathbf{R}}_{\text{rw}}^{(D)}$ . Note that reflectance estimation from raw patches/images [91] is another problem that is not addressed in this study.

Our segmentation approach is a supervised pixel classification. At each vegetation



pixel, we consider a centered neighborhood as a patch whose size is a small multiple of  $\sqrt{D}$  and in which at least 85% of the pixels represent vegetation. For each detection case (beet vs. weeds, wheat vs. weeds, and bean vs. weeds), a specific number of learning and test patches are extracted. We extract patches of size  $24 \times 24$  (for  $\sqrt{D} = 4$ ),  $25 \times 25$  (for  $\sqrt{D} = 5$ ), or  $20 \times 20$  pixels for the case wheat vs. weeds (for  $\sqrt{D} = 4$  or  $\sqrt{D} = 5$ ) to have enough samples to characterize thin leaves of wheat. Raw patches are simulated from the original multispectral patches using the CA  $4 \times 4$  or  $5 \times 5$  MSFA. For crop/weed detection, we randomly extract  $\mathcal{N}_1^l \approx 180 \cdot 10^3$  patches from learning images, half for crop and half for weed class. As we merge thistle, goosefoot, and datura patches to build a single weed class, we extract  $(\mathcal{N}_1^l/2)/3$  training patches for each of them. For crop/thistle/goosefoot/datura identification, we extract  $\mathcal{N}_2^l \approx 50 \cdot 10^3$  training patches per class. The number of test patches for both problems are displayed in Tab. 6.1. We extract texture features from these patches, then the central pixel of each test patch is classified as crop or weed (detection problem), or as one of the four (one crop and three weeds) vegetation classes (identification problem).

TABLE 6.1: Number of CA  $4 \times 4$ /CA  $5 \times 5$  test patches for crop/weed detection and identification problems, for each of the three crop species (beet, wheat, and bean).

| Patch size (pixels)                | 24 × 24/25 × 25 |         |         |         | 20 × 20 |         |        |         | 24 × 24/25 × 25 |         |        |         |
|------------------------------------|-----------------|---------|---------|---------|---------|---------|--------|---------|-----------------|---------|--------|---------|
| Class                              | Beet            | Th.     | Go.     | Da.     | Wheat   | Th.     | Go.    | Da.     | Bean            | Th.     | Go.    | Da.     |
| Test patches (identification case) | 694,201         | 509,547 | 259,130 | 203,273 | 584,599 | 342,925 | 79,169 | 215,540 | 166,761         | 381,717 | 63,260 | 203,767 |
| Test patches (detection case)      |                 | 971,950 |         |         |         | 637,634 |        |         |                 | 648,744 |        |         |

### 6.3.2 Compared texture features

We consider several state-of-the-art descriptors, either deep learning-based or hand-crafted ones, to compare their performance to ours. We use a shallower version of SegNet [14] model, called SegNet-Basic, that we adapt to image classification by considering the sole encoder part with an extra flattening layer to vectorize feature maps. All convolutional kernels are of size  $3 \times 3$  instead of  $7 \times 7$  to capture small details (and reduce the number of hyperparameters to learn), and we add two fully-connected layers for non-linearity and dimension reduction, which provides a 512-dimensional feature vector. To make it possible to learn from very small patches, we

TABLE 6.2: CNN architectures used for feature extraction from fully-defined images. Filter depth (e.g.,  $D$  for first layer) is not shown for sake of clarity. See caption and colors of Fig. 6.1. LRN: local response normalization.

| SegNet-Basic encoder                                         | Blob-Net                                                     | cNet                                                        |
|--------------------------------------------------------------|--------------------------------------------------------------|-------------------------------------------------------------|
| 64 · (3 × 3) Conv. kernels<br>1-pixel stride + zero-padding  | 192 · (5 × 5) Conv. kernels<br>1-pixel stride + zero-padding | 64 · (5 × 5) Conv. kernels<br>1-pixel stride + zero-padding |
| BN + ReLU                                                    | ReLU                                                         | ReLU                                                        |
| Maxpool 2 × 2                                                | Maxpool 2 × 2                                                | Maxpool 2 × 2 + LRN                                         |
| 128 · (3 × 3) Conv. kernels<br>1-pixel stride + zero-padding | 256 · (5 × 5) Conv. kernels<br>1-pixel stride + zero-padding | 64 · (5 × 5) Conv. kernels<br>1-pixel stride + zero-padding |
| BN + ReLU                                                    | ReLU                                                         | ReLU                                                        |
| 256 · (3 × 3) Conv. kernels<br>1-pixel stride + zero-padding | Maxpool 2 × 2                                                | Maxpool 2 × 2 + LRN                                         |
| BN + ReLU                                                    | 256 · (3 × 3) Conv. kernels<br>1-pixel stride + zero-padding | Flatten                                                     |
| 512 · (3 × 3) Conv. kernels<br>1-pixel stride + zero-padding | ReLU                                                         | FC-384 + ReLU                                               |
| BN + ReLU                                                    | Flatten                                                      | FC-192 + ReLU                                               |
| Maxpool 2 × 2                                                | FC-512 + ReLU                                                | -                                                           |
| Flatten                                                      | FC-512 + ReLU                                                | -                                                           |
| FC-1024 + ReLU                                               | -                                                            | -                                                           |
| FC-512 + ReLU                                                | -                                                            | -                                                           |
| FC- $N^C$ + Softmax                                          | FC- $N^C$ + Softmax                                          | FC- $N^C$ + Softmax                                         |

omit the second and third maxpooling layers. We also consider two additional models proposed for crop/weed detection using RGB-NIR images. Milioto et al. [125] propose a CNN (that we denote here as *Blob-Net*) composed of three convolutional layers (two of  $5 \times 5$  kernels and one of  $3 \times 3$  kernels), followed by two  $2 \times 2$  maxpooling layers, and two fully-connected layers that provide a 512-dimensional feature vector. In [144], a CNN model called *cNet* that also exploits RGB-NIR images is proposed for crop/weed detection. It is composed of two convolutional layers (of  $5 \times 5$  kernels), two  $2 \times 2$  maxpooling layers, and two fully-connected layers that provide a 192-dimensional feature vector. In all models (see details in Tab. 6.2), the texture feature vector is fed into a fully-connected layer that uses the softmax function to provide the  $N^C$ -dimensional probability vector. Note that oppositely to MSFA-Net (see Eq. (6.1)), the first convolutional layer of SegNet-Basic, Blob-Net, and cNet applies (with one-pixel stride) kernels  $\mathbf{H}_n^{(D)}$  of size  $w \times w$  and depth  $D$  to the input  $D$ -channel patch  $\mathbf{P}^{(D)}$  at each pixel  $(x, y) \in \llbracket 0, (m \cdot \sqrt{D}) - 1 \rrbracket^2$  in a classical way:

$$O_n(x, y) = \sum_{d=0}^{D-1} \sum_{i=0}^{w-1} \sum_{j=0}^{w-1} H_n^d(i, j) \cdot P^d(x+i, y+j). \quad (6.2)$$

TABLE 6.3: Feature size provided by each descriptor with CA  $4 \times 4$  and CA  $5 \times 5$  MSFAs.

| Input                   | MSFA     |       | Demosaiced           |          |      |                                                                           |
|-------------------------|----------|-------|----------------------|----------|------|---------------------------------------------------------------------------|
| Descriptor              | MSFA-Net | M-LBP | SegNet-Basic encoder | Blob-Net | cNet | $\hat{\mathbf{R}}_{\text{rw}}^{(D)} / \bar{\mathbf{R}}_{\text{rw}}^{(D)}$ |
| CA $4 \times 4$ patches | 128      | 4096  | 512                  | 512      | 192  | 16                                                                        |
| CA $5 \times 5$ patches | 128      | 6400  | 512                  | 512      | 192  | 25                                                                        |

As handcrafted features, we compute histograms of LBP operators that have proven to be powerful for texture extraction. Specifically, we use M-LBP descriptor that extracts features from raw patches [123]. We consider the simple case of a square neighborhood around any given pixel in a raw patch at a spatial distance of 1 pixel (8 neighbors with no interpolation), which provides a  $256 \cdot D$ -dimensional feature vector.

We also consider reflectance spectra as features. Each demosaiced learning or test patch is represented by a  $D$ -dimensional reflectance vector  $\hat{\mathbf{R}}_{\text{rw},p}^{(D)}$ , whose  $d$ -th component is the average reflectance value over a small square window (of 25 pixels) centered at the center  $p$  of the considered patch to reduce noise influence. Furthermore, to make reflectance signatures robust against shading and specular reflection, we normalize each reflectance vector as  $\bar{\mathbf{R}}_{\text{rw},p}^{(D)}$  so that its energy sums up to 1 (see Eq. (5.3)). As  $D = 16$  or  $25$  here,  $\hat{\mathbf{R}}_{\text{rw},p}^{(16)} / \hat{\mathbf{R}}_{\text{rw},p}^{(25)}$  and  $\bar{\mathbf{R}}_{\text{rw},p}^{(16)} / \bar{\mathbf{R}}_{\text{rw},p}^{(25)}$  represent the 16-/25-dimensional unnormalized and normalized reflectance feature vectors extracted from demosaiced patches. Table 6.3 shows the texture feature size provided by each descriptor with each MSFA.

To reduce sparse pixel misclassification after prediction, we assume that reflectance does (almost) not change across locally close surface elements of a scene. Plausibly, these elements belong to the same material, hence to the same class. Each prediction associated to a test vegetation pixel is then filtered using a majority voting rule, and its final class label is the most frequent one over its  $9 \times 9$  neighborhood. Furthermore, to get a dense visualization of the predictions, we interpolate the filtered predictions to obtain the final (dense) classification map. We consider an interpolation window centered at a given unpredicted vegetation pixel and check if the number of predicted vegetation pixels inside this window is higher than 2 pixels. Otherwise, the size of the interpolation window is repeatedly increased by half of its size until the condition is satisfied.

### 6.3.3 CNN Training

To train and validate CNN-based features, 95% of the training patches are used for learning and the remainder for validation. The CNNs are then trained using the learning patches as follows.

**Initialization:** As we aim to train MSFA-Net from scratch, we must choose an appropriate way to initialize the weights of its convolutional and fully-connected layers. Initializing the weights with too small values may slow the learning process and lead to the vanishing gradient problem, and larger values may lead to exploding gradient. Since all compared CNNs use the ReLU as activation function, we follow He et al. [74] technique to initialize their weights. Specifically, we use the He-uniform variant that draws the weights from a uniform distribution instead of a normal one because it slightly provides better results in our case.

**Training:** We use the stochastic gradient descent (SGD) weight optimizer for all models. The loss function to be minimized is the multi-class log-loss, and the optimization is performed for 40 epochs because no performance increase is observed afterwards. The batch size is set to 128. We train MSFA-Net with a fixed learning rate  $\epsilon = 0.005$  because it provides better convergence, and weight decay and momentum values of  $5 \cdot 10^{-4}$  and 0.9 to regularize the learning process. For SegNet-Basic, we set  $\epsilon = 10^{-3}$  and momentum to 0.9 [14]. For Blob-Net and cNet, we also use the SGD optimizer as in [124, 144] and the same learning rate and momentum values as for SegNet-Basic.

## 6.4 Classification results and discussion

We now assess the contribution of texture features for crop/weed recognition extracted from raw patches by our proposed MSFA-Net with respect to those extracted by other descriptors either from raw or demosaiced patches. The crop/weed recognition performances obtained with LGBM classifier trained with features extracted from CA  $4 \times 4$  and CA  $5 \times 5$  patches are reported in Table 6.4.

### 6.4.1 CA $4 \times 4$

Let us first study the classification performances obtained by LGBM classifier with each tested feature in the case of raw and demosaiced patches associated to CA  $4 \times 4$  MSFA (see Tabs. 6.4a and 6.4b). For both crop/weed detection and identification problems, M-LBP descriptor provides bad performances because there are not enough pixels to efficiently capture the spatio-spectral band interactions. Reflectance features (that neither take these interactions into account) even perform better than M-LBP in most cases. cNet-based features provide the best beet/weed detection results in terms of precision, overall weighted accuracy, and F1-score, followed by BlobNet, then SegNet-basic, and MSFA-Net provides comparable performances to SegNet-Basic.

For wheat/weed and bean/weed detection problems, MSFA-Net-based features extracted from raw patches outperform all the other ones and provide the best results according to all metrics, despite their low dimension. This is probably because the demosaicing procedure reduces the quality of texture description from 16-channel patches. In the case of crop/weed identification, the classification performances of all descriptors is lower than for detection. Crop species and datura are mostly well recognized, while thistle and especially goosefoot plants are not. The low classification performances obtained on these two classes severely impact the overall weighted accuracy and F1 scores. Textures of goosefoot leaves are very similar to beet and datura ones, and because texture of thistle is smooth, it can be confused with that of wheat plant that also has a similar darkish green, especially at advanced growth stages.

### 6.4.2 CA $5 \times 5$

Let us now study the classification performances obtained using features extracted from raw and demosaiced patches associated to CA  $5 \times 5$  MSFA (see Tabs. 6.4c and 6.4d). We can see that performances of the descriptors are either higher or lower than with CA  $4 \times 4$ , depending on the problem. Indeed, as CA  $5 \times 5$  MSFA has more bands than CA  $4 \times 4$  in its basic pattern, the spectral sampling of vegetation textures increases while the spatial one decreases. Thus, M-LBP fails to capture enough spatial information to well characterize crops and weeds. In order to perform well, a

TABLE 6.4: Crop/weed detection (a, c) results (Pr(ecision), Re(call), weighted overall accuracy, and F1-score) and identification (b, d) results (per-class accuracy, weighted overall accuracy, and F1 score) obtained with LGBM classifier trained with features extracted from raw ( $\dagger$ ) and demosaiced ( $\star$ ) patches simulated with CA  $4 \times 4$  (a, b,  $D = 16$ ) and CA  $5 \times 5$  (c, d,  $D = 25$ ) MSFAs. Bold shows best result and italics second best one for each tested feature.

| Feature                                       | Beet vs. weed |             |             |             | Wheat vs. weed |             |             |             | Bean vs. weed |             |             |             |
|-----------------------------------------------|---------------|-------------|-------------|-------------|----------------|-------------|-------------|-------------|---------------|-------------|-------------|-------------|
|                                               | Pr            | Re          | Acc.        | F1          | Pr             | Re          | Acc.        | F1          | Pr            | Re          | Acc.        | F1          |
| MSFA-Net $\dagger$                            | 88.5          | <b>96.4</b> | 88.2        | 89.8        | <b>99.6</b>    | <b>94.0</b> | <b>96.9</b> | <b>96.6</b> | <b>97.2</b>   | <b>93.4</b> | <b>90.5</b> | <b>85.8</b> |
| M-LBP $\dagger$                               | 87.2          | 92.0        | 85.6        | 86.5        | 97.4           | 78.4        | 88.5        | 87.6        | 91.7          | 80.4        | 73.5        | 63.5        |
| SegNet-Basic $\star$                          | 90.1          | 95.1        | 89.4        | 90.3        | 99.4           | 92.7        | 96.2        | 95.9        | 96.1          | 89.3        | 86.7        | 79.1        |
| BlobNet $\star$                               | 92.7          | 93.6        | 91.3        | 91.5        | 99.5           | 90.6        | 95.2        | 94.9        | 95.4          | 90.4        | 84.7        | 79.1        |
| cNet $\star$                                  | <b>93.6</b>   | 93.0        | <b>91.9</b> | <b>91.8</b> | 99.3           | 90.9        | 95.2        | 95.1        | 95.0          | 87.7        | 83.3        | 75.5        |
| $\hat{\mathbf{R}}_{\text{rw}}^{(16)} \star$   | 85.1          | 92.1        | 83.4        | 84.7        | 98.7           | 88.3        | 93.8        | 93.3        | 91.6          | 77.2        | 73.4        | 61.2        |
| $\tilde{\mathbf{R}}_{\text{rw}}^{(16)} \star$ | 89.3          | 92.1        | 87.7        | 88.1        | 98.7           | 90.8        | 94.9        | 94.6        | 90.6          | 80.7        | 70.2        | 61.7        |

(a) Detection results: CA  $4 \times 4$  MSFA

| Feature                                       | Beet        | Th          | Go          | Da          | Acc.        | F1          | Wheat       | Th          | Go          | Da          | Acc.        | F1          | Bean        | Th          | Go          | Da          | Acc.        | F1          |
|-----------------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| MSFA-Net $\dagger$                            | 82.6        | <b>76.9</b> | 61.4        | 77.3        | 72.5        | 66.3        | <b>99.6</b> | <b>57.3</b> | <b>58.4</b> | 82.3        | <b>66.5</b> | <b>55.8</b> | <b>89.4</b> | <b>51.8</b> | 48.7        | <b>81.9</b> | 62.8        | <b>48.8</b> |
| M-LBP $\dagger$                               | 73.5        | 71.3        | 44.5        | 72.2        | 63.3        | 55.8        | 95.8        | 36.9        | 38.9        | 76.0        | 51.0        | 44.3        | 62.5        | 50.2        | 35.6        | 69.5        | 48.1        | 38.9        |
| SegNet-Basic $\star$                          | 83.5        | 69.4        | 66.0        | <b>81.5</b> | <b>74.8</b> | 67.3        | 99.2        | 47.1        | 53.0        | <b>84.5</b> | 62.5        | 50.8        | 84.9        | 43.4        | 48.9        | 81.0        | 61.1        | 46.2        |
| BlobNet $\star$                               | <b>86.3</b> | 70.4        | <b>66.7</b> | 78.7        | 74.5        | <b>68.1</b> | 99.3        | 48.9        | 57.1        | 81.6        | 64.5        | 51.8        | 83.3        | 47.0        | <b>53.7</b> | 80.4        | <b>63.6</b> | 47.6        |
| cNet $\star$                                  | 86.2        | 69.7        | 61.5        | 77.7        | 72.2        | 65.8        | 99.3        | 46.3        | 57.7        | 82.1        | 65.0        | 53.8        | 78.9        | 42.7        | 53.2        | 79.0        | 61.8        | 45.5        |
| $\hat{\mathbf{R}}_{\text{rw}}^{(16)} \star$   | 68.7        | 55.6        | 50.7        | 75.1        | 63.3        | 53.3        | 97.8        | 40.4        | 52.4        | 74.6        | 59.4        | 49.2        | 60.1        | 34.6        | 45.4        | 71.6        | 51.8        | 37.9        |
| $\tilde{\mathbf{R}}_{\text{rw}}^{(16)} \star$ | 77.7        | 61.5        | 61.3        | 71.6        | 67.3        | 60.5        | 97.7        | 43.8        | 47.0        | 73.3        | 56.6        | 48.1        | 57.5        | 46.2        | 41.3        | 69.4        | 49.7        | 39.0        |

(b) Identification results: CA  $4 \times 4$  MSFA

| Feature                                       | Beet vs. weed |             |             |             | Wheat vs. weed |             |             |             | Bean vs. weed |             |             |             |
|-----------------------------------------------|---------------|-------------|-------------|-------------|----------------|-------------|-------------|-------------|---------------|-------------|-------------|-------------|
|                                               | Pr            | Re          | Acc.        | F1          | Pr             | Re          | Acc.        | F1          | Pr            | Re          | Acc.        | F1          |
| MSFA-Net $\dagger$                            | 89.7          | <b>97.2</b> | 89.7        | 91.2        | 99.4           | <b>92.6</b> | <b>96.1</b> | <b>95.8</b> | <b>96.8</b>   | <b>92.6</b> | <b>89.1</b> | <b>84.0</b> |
| M-LBP $\dagger$                               | 83.0          | 90.3        | 81.1        | 82.2        | 93.5           | 73.1        | 84.4        | 83.0        | 93.8          | 77.4        | 79.4        | 64.9        |
| SegNet-Basic $\star$                          | 92.3          | 96.3        | 91.2        | <b>92.7</b> | <b>99.5</b>    | 91.0        | 95.4        | 95.1        | <b>96.8</b>   | 90.2        | 82.6        | 77.5        |
| BlobNet $\star$                               | 93.1          | 95.1        | <b>92.2</b> | 92.6        | <b>99.5</b>    | 91.2        | 95.5        | 95.2        | 94.3          | 88.0        | 81.0        | 74.5        |
| cNet $\star$                                  | <b>93.2</b>   | 94.8        | <b>92.2</b> | 92.5        | <b>99.5</b>    | 90.1        | 95.0        | 94.7        | 94.7          | 84.7        | 82.3        | 72.0        |
| $\hat{\mathbf{R}}_{\text{rw}}^{(25)} \star$   | 86.1          | 92.8        | 84.7        | 85.9        | 98.7           | 88.0        | 93.6        | 93.1        | 91.7          | 76.2        | 73.7        | 60.7        |
| $\tilde{\mathbf{R}}_{\text{rw}}^{(25)} \star$ | 90.2          | 92.9        | 88.7        | 89.3        | 98.6           | 90.8        | 94.5        | 94.5        | 90.6          | 81.1        | 70.0        | 61.8        |

(c) Detection results: CA  $5 \times 5$  MSFA

| Feature                                       | Beet        | Th          | Go          | Da          | Acc.        | F1          | Wheat       | Th          | Go          | Da          | Acc.        | F1          | Bean        | Th          | Go          | Da          | Acc.        | F1          |
|-----------------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| MSFA-Net $\dagger$                            | 82.6        | 75.5        | 70.2        | 82.5        | <b>77.4</b> | <b>70.4</b> | 99.2        | <b>58.1</b> | 51.6        | 85.7        | 63.4        | <b>55.2</b> | <b>87.7</b> | 53.0        | 48.9        | 79.8        | 62.4        | <b>48.5</b> |
| M-LBP $\dagger$                               | 62.5        | 70.1        | 39.0        | 60.9        | 55.3        | 48.2        | 90.3        | 36.7        | 34.2        | 65.9        | 45.2        | 39.1        | 59.8        | <b>53.4</b> | 30.8        | 54.4        | 42.3        | 34.1        |
| SegNet-Basic $\star$                          | 83.2        | 69.2        | <b>70.6</b> | 83.9        | 77.2        | 69.1        | <b>99.4</b> | 50.6        | 52.8        | <b>86.0</b> | 63.2        | 52.4        | 81.1        | 47.7        | 47.6        | <b>80.2</b> | 59.9        | 46.2        |
| BlobNet $\star$                               | 90.4        | <b>81.7</b> | 29.0        | 82.9        | 66.2        | 61.0        | 99.3        | 50.0        | 57.2        | 83.6        | 65.1        | 52.9        | 83.5        | 49.2        | 52.1        | 79.3        | <b>62.8</b> | 47.7        |
| cNet $\star$                                  | <b>90.9</b> | 80.9        | 25.4        | <b>84.4</b> | 65.6        | 59.9        | <b>99.4</b> | 46.8        | <b>58.6</b> | 81.9        | <b>65.2</b> | 52.2        | 80.7        | 45.8        | <b>52.4</b> | 79.4        | 62.1        | 46.4        |
| $\hat{\mathbf{R}}_{\text{rw}}^{(25)} \star$   | 71.3        | 56.1        | 51.3        | 76.1        | 64.3        | 54.7        | 98.3        | 40.2        | 49.5        | 77.0        | 57.9        | 47.1        | 61.3        | 34.2        | 43.8        | 71.9        | 51.2        | 37.9        |
| $\tilde{\mathbf{R}}_{\text{rw}}^{(25)} \star$ | 80.5        | 63.9        | 60.0        | 75.6        | 69.2        | 62.9        | 98.2        | 43.9        | 44.7        | 77.0        | 55.6        | 46.3        | 58.6        | 47.3        | 41.0        | 73.3        | 50.5        | 40.3        |

(d) Identification results: CA  $5 \times 5$  MSFA

trade-off between spatial and spectral samplings may be required for M-LBP. Similarly for MSFA-Net, its wheat/weeds and bean/weeds detection and identification performances are slightly decreased. However, they are improved in beet/weeds detection and beet/thistle/goosefoot/datura identification problems. We observe a similar behaviour in performance with the other CNN models. Performances are either degraded or improved depending on the detection/identification problem. Table 6.4c shows that deep features outperform handcrafted ones for crop/weed detection. Moreover, our MSFA-Net provides the best recall for the three detection problems and the best precision for bean/weed detection. It also provides a precision (99.4%) that is comparable to that of the other deep features (99.5%) for wheat/weed detection. Even if CA  $5 \times 5$  MSFA potentially allows for deep learning-based descriptors extracted by SegNet-Basic, BlobNet, and cNet to learn more spectral characteristics from the 25-channel demosaiced patches, it does not necessarily mean a gain in crop/weeds recognition performance. Indeed, these CNNs may overfit to the learning vegetation patches which impacts their performances. Moreover, since demosaicing is more complicated with CA  $5 \times 5$  raw images than with CA  $4 \times 4$  ones, the fully-defined estimated 25-channel images are subject to more estimation artifacts that may alter texture representation. For reflectance features, we observe the opposite behaviour: the results obtained by  $\hat{\mathbf{R}}_{\text{rw}}^{(25)} / \bar{\mathbf{R}}_{\text{rw}}^{(25)}$  reflectance features are globally better (except in wheat/thistle/goosefoot/datura identification problem) than those of  $\hat{\mathbf{R}}_{\text{rw}}^{(16)} / \bar{\mathbf{R}}_{\text{rw}}^{(16)}$ .

### 6.4.3 Discussion

For beet/weed detection, BlobNet and cNet provide the best performances. However, for beet/thistle/goosefoot/datura identification problem, MSFA-Net provides the best performances followed by SegNet-Basic, BlobNet, and cNet. For wheat/weed and bean/weed detection, MSFA-Net also provides the best performance. Moreover, for wheat/thistle/goosefoot/datura and bean/thistle/goosefoot/datura identification problems, MSFA-Net and cNet provide similar performances, followed by BlobNet, then SegNet-encoder. Figure 6.2(a) shows a t-SNE projection [181] of raw patches characterized by MSFA-Net. We can see that MSFA-Net is able to learn a non-linear feature space where the classes do not overlap. However, because texture



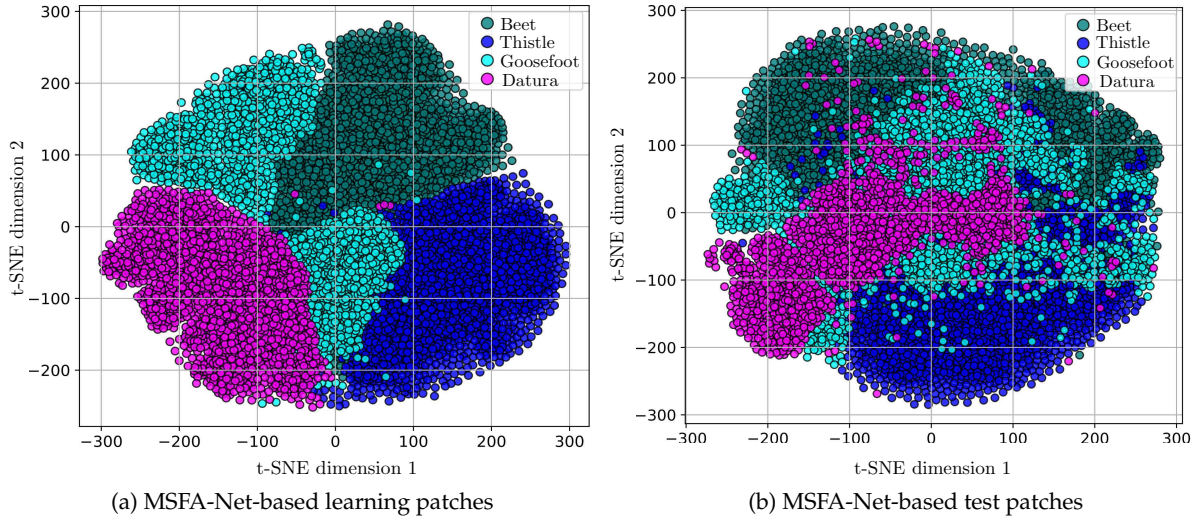


FIGURE 6.2: t-SNE visualization of beet, thistle, goosefoot, and datura learning (a) and test (b) patches (2000/class), characterized by MSFA-Net.

feature and spectral signatures of the test patches are subject to a severe domain shift (due to varying growth stage, leaf orientation, shadows, etc.), the clusters strongly overlap. In order to improve outdoor crop/weed identification, *domain adaptation* techniques [90, 186] may be applied to overcome the target domain shift problem. Our proposed approach globally outperforms all the others in crop/weed detection and provides at least comparable crop/thistle/goosefoot/datura identification performances. This confirms that features provided by MSFA-Net are more discriminant despite their small size. This also confirms that feature extraction from raw patches is relevant for the crop/weed recognition problem.

MSFA-Net requires to learn 17 times fewer hyperparameters than the considered SegNet-Basic variant, and 6 times fewer than Blob-Net, with patches of size  $25 \times 25$  pixels for weed detection. cNet requires to learn slightly ( $\approx 10\%$ ) fewer hyperparameters than MSFA-Net because it is shallower. All in all, MSFA-Net provides better or comparable performances than other approaches at much reduced computation costs.

Figure 6.3 shows the cost of feature extraction from raw and demosaiced patches associated to test image 1 (see Fig. 6.4(a)). We can see that MSFA-Net provides a good trade-off between performance and computation time of feature extraction.

Figure 6.4 displays the segmentation results obtained by MSFA-Net, SegNet-Basic, Blob-Net, and cNet approaches on three test images for beet/weed detection,



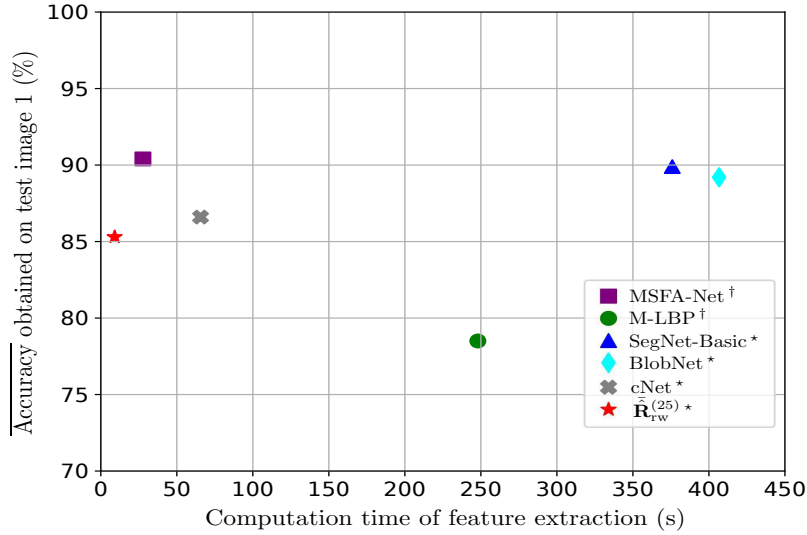


FIGURE 6.3: Classification accuracy obtained by LGBM classifier vs. computation time of feature extraction (parallelized over 12 CPU cores) from the (249,895)  $25 \times 25$  patches of test image 1 (see Fig. 6.4(a)). Nor demosaicing nor CNN training computation times are considered. <sup>†</sup> and <sup>\*</sup> indicate feature extraction from raw and demosaiced patches, respectively.

beet/goosefoot and bean/datura identification problems. It shows comparable detection performances between the three tested CNNs and MSFA-Net. However, for beet/weed and bean/datura identification, MSFA-Net has more success in recognizing beet/goosefoot and bean/datura leaves.

## 6.5 Conclusion

In this chapter, we address crop/weed recognition using texture features. For this purpose, we use the two MSFAs proposed in Chapter 5 to simulate raw and fully-defined patches. We propose multispectral texture features extraction based on a CNN architecture called MSFA-Net. It directly analyzes raw patches thanks to its first layer that learns spatio-spectral interactions among channel values that match the basic MSFA pattern. This approach avoids the demosaicing step that can be greedy in computation requirements and may alter the texture representations. It requires learning much fewer hyperparameters than most of state-of-the-art CNNs. Experimental results show the relevance of MSFA-based features for outdoor crop/weed recognition. Combining MSFA-based snapshot imaging with a CNN provides low-size and highly discriminant features that are suitable for outdoor crop/weed recognition.



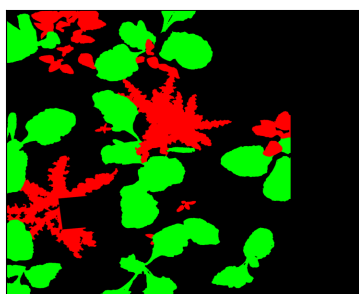
(a) Test image 1



(b) Test image 2



(c) Test image 3



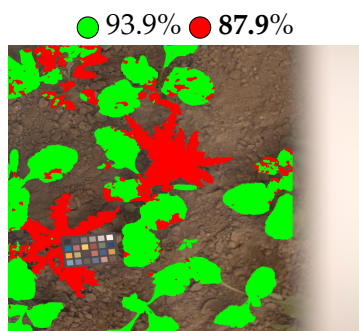
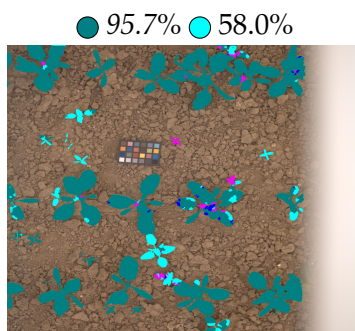
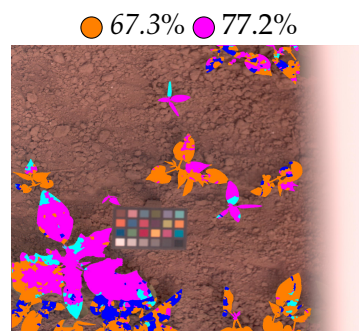
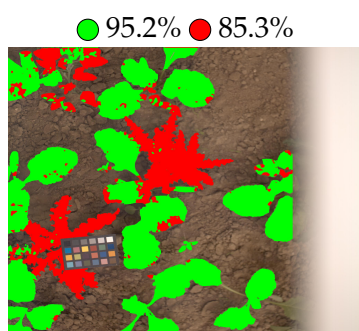
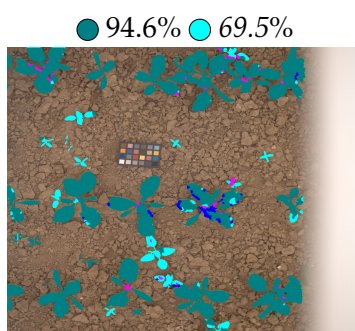
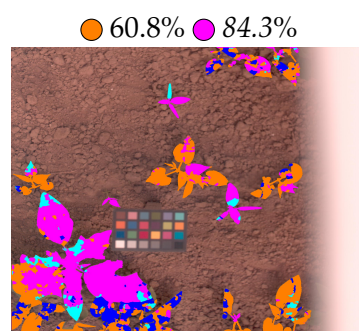
(d) ● Crop (Beet) ● Weed (Goosefoot)



(e) ● Beet ● Thistle



(f) ● Bean ● Datura

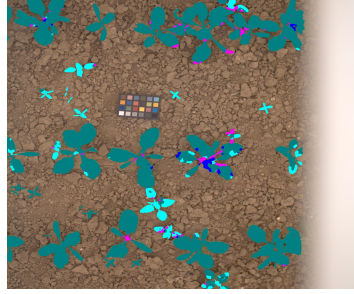
**SegNet-Basic (g-i)**(g) ( $\overline{F1}$ =89.8%  $\overline{Accuracy}$ =89.9%)(h) ( $\overline{F1}$ =69.1%  $\overline{Accuracy}$ =62.5%)(i) ( $\overline{F1}$ =78.8%  $\overline{Accuracy}$ =73.0%)**BlobNet (j-l)**(j) ( $\overline{F1}$ =89.6%  $\overline{Accuracy}$ =88.5%)(k) ( $\overline{F1}$ =77.7%  $\overline{Accuracy}$ =72.5%)(l) ( $\overline{F1}$ =79.8%  $\overline{Accuracy}$ =74.2%)

**cNet (m-o)**

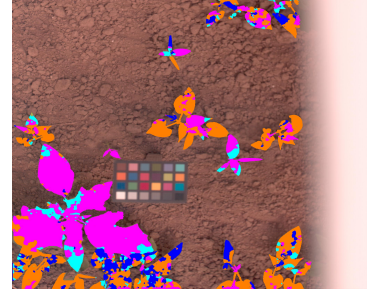
● 96.5% ● 81.7%

(m) ( $\overline{F1}$ =88.9%  $\overline{Accuracy}$ =86.6%)

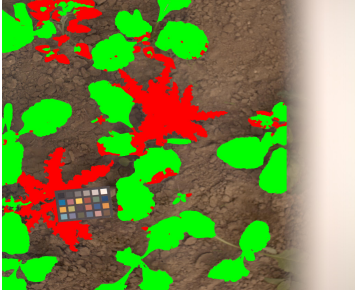
● 94.5% ● 60.9%

(n) ( $\overline{F1}$ = 72.4%  $\overline{Accuracy}$ =64.9%)

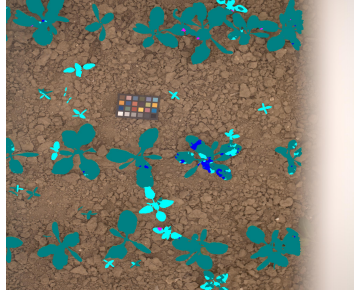
● 61.2% ● 82.1%

(o) ( $\overline{F1}$ =78.2%  $\overline{Accuracy}$ =73.0%)**MSFA-Net (p-r)**

● 96.7% ● 87.4%

(p) ( $\overline{F1}$ =**91.8%**  $\overline{Accuracy}$ =**90.4%**)

● 96.7% ● 75.2%

(q) ( $\overline{F1}$ =**82.8%**  $\overline{Accuracy}$ =77.7%)

● 77.0% ● 89.9%

(r) ( $\overline{F1}$ =**86.5%**  $\overline{Accuracy}$ =**84.4%**)

FIGURE 6.4: Segmentation results obtained by LightGBM classifier trained with features extracted from demosaiced (third to fifth row) and raw (sixth row) patches by SegNet-Basic (g-i), BlobNet (j-l), cNet (m-o), and MSFA-Net (p-r). (a)-(c): RGB renderings of three multi-spectral test images, (d)-(f): ground truths, (g, j, m, p): weed detection results, (h, i, k, l, n, o, q, r): weed identification results. The per-class accuracy score is displayed near each colored circle (class label) for each considered feature. Bold values show best results and italics second best one for each problem and metric.

# Conclusion and perspectives

## Conclusion

The main objective of this work is to develop an automatic recognition system of crop and weed plants in field conditions based on multispectral imaging. This thesis can be summarized into four main contributions:

### New multispectral vegetation database

The targeted species are (sugar) beet (*Beta vulgaris*), wheat (*Triticum* sp.), and (green) bean (*Phaseolus vulgaris*) for crops, and thistle (*Cirsium arvense*), goose-foot (*Chenopodium* sp.), and datura (*Datura stramonium*) for weeds. To analyze spectral signatures of the considered plants, we build our own multispectral image database of vegetation. A specific acquisition device is designed to acquire outdoor multispectral images of vegetation using a multishot (linescan) camera, called the Snapscan, that observes vegetation from nadir. A white diffuser is mounted on the acquisition device so that it is contained in the acquired images. A ColorChecker chart is also included in the scene for estimated reflectance assessment. The dataset built in this study contains 109 radiance images of  $2048 \times 2048$  pixels  $\times$  192 channels of 10 bit depth. Each channel is associated to a specific narrow spectral band centered at wavelength  $\lambda^b$ ,  $b \in \llbracket 0, 191 \rrbracket$ , that ranges from 475.1 nm to 901.7 nm.

### Original multispectral image formation model

Because outdoor illumination varies during the multispectral image acquisition by the Snapscan, the classical Lambertian image formation model is not adapted in our case. Indeed, the Snapscan camera acquires a sequence of frames to provide a multispectral image. Each frame may be acquired under a specific illumination, hence, the

measurement of the radiance that is reflected by a given Lambertian surface element of the scene varies according to the frame acquisition time. Therefore, we propose an original multispectral image formation model that accounts for illumination variation at the frame level. This model can be adapted to model the image acquisition of other multishot multispectral cameras.

### **Illumination-variation-robust reflectance estimation**

To improve outdoor crop and weed recognition, we compute illumination-invariant features by estimating the reflectance from the acquired radiance images. Our image formation model shows that state-of-the-art reflectance estimation methods may be inappropriate to estimate reflectance when illumination varies during image acquisition. We propose a method that exploits a white diffuser to estimate reflectance in a row-wise approach, such that illumination variation is considered at the frame level. The contribution of reflectance features estimated by our proposed approach to outdoor crop/weed recognition is experimentally demonstrated.

### **CNN-based texture feature from raw image**

As multishot cameras are sensitive to illumination variation, they are not adapted to outdoor crop/weed recognition. Therefore, we propose to use a single-sensor (snapshot) camera. To specify a snapshot camera based on the MSFA technology, we propose to select the most informative spectral bands. These bands define filters that compose the basic pattern of the MSFA. For experiments, we simulate raw patch acquisitions by this snapshot camera.

Finally, we propose an original multispectral texture feature extraction based on a CNN. It directly analyzes raw patches by respecting the basic pattern arrangement in the MSFA, and learns spatio-spectral band interactions at reduced computation costs. This approach avoids the demosaicing step that is greedy in computation requirements and is prone to alter the texture representations. We show that combining MSFA-based snapshot imaging with deep learning principles provides discriminant features for outdoor crop/weed recognition.



## Perspectives

Although this thesis provides several contributions, it also arises several interesting problems regarding reflectance estimation, raw image analysis, and crop/weed recognition.

### Reflectance estimation from multispectral images

Our two proposed reflectance estimation methods (see Secs. 3.4.3 and 3.4.4) rely on reference devices to estimate reflectance under varying illuminations. Specifically, they rely on a white diffuser to estimate the illumination for each pixel row. However, the white diffuser must be fixed on the acquisition system so that it is constantly observed by the camera. Therefore, it would be interesting to focus on reflectance estimation without any reference device. The next challenge is then to explore a deep learning approach for reflectance estimation.

### Reflectance estimation from raw images

We experimentally demonstrate that MSFA-based snapshot multispectral imaging is relevant for outdoor crop/weed recognition. The raw reflectance patches have been simulated from fully-defined reflectance patches that have been estimated from acquired radiance images. In practice however, a snapshot camera provides a raw radiance image, and not a raw reflectance one. To the best of our knowledge, the literature reports almost no work (only one paper [91]) regarding reflectance estimation from raw radiance images provided by a snapshot camera. Ongoing work should focus on reflectance estimation from raw radiance images acquired by a snapshot camera. Depending on the goal, one could then either directly provide the estimated raw reflectance images, or recover the fully-defined multispectral reflectance images.

### Handcrafted and deep learning-based feature fusion

To detect and identify weeds from crops, we either rely on reflectance spectra (handcrafted) or on deep texture features extracted from reflectance patches. However, it is also interesting to assess the feasibility of combining reflectance spectra with deep

texture features to improve crop/weed identification. As characteristics and dimensions of both features are different, they cannot directly be concatenated. A specific fusion strategy would be studied together with the feasibility of combining texture features with spectra into the CNN.

### MSFA-Net

Our proposed MSFA-Net directly analyzes raw patches thanks to its first convolutional layer. Future works will first study the impact of band arrangement in the basic pattern on MSFA-Net performance. They will also focus on the improvement of MSFA-Net using residual learning to increase the spatio-spectral information in the feature maps. Moreover, since the first convolutional layer of MSFA-Net only captures the spatio-spectral band interactions of a specific basic pattern, setting different starting convolutional positions can learn several basic patterns with different band arrangements. Then, similarly to the approach proposed in [100], we can build a global MSFA-Net composed of micro MSFA-Nets, whose each one is specialized in a specific pattern (see Fig. 6.5). The texture feature vector provided by each micro MSFA-Net can be concatenated and fed into a fully-connected layer for dimension reduction to provide the final texture feature vector.

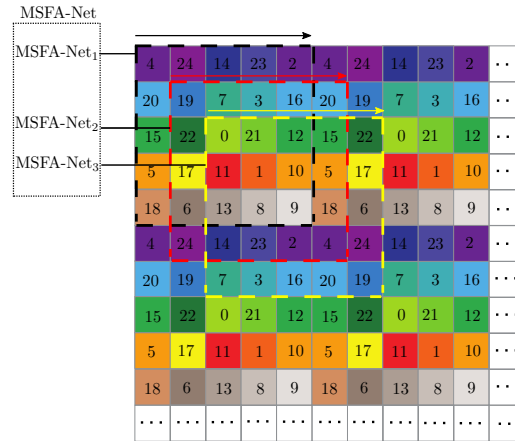


FIGURE 6.5: MSFA-Net learning from different band arrangements.

### Domain adaptation for crop/weed recognition

Spectral signatures and textures depend on plant growth stages and plant observation conditions, e.g., leaf orientation, or shadows. As a consequence, the performance of classical supervised learning approaches is weakened because the feature

distribution of test samples may be different to that of the learning ones. This is known as *target domain shift* or *out-of-domain* problem. To overcome this problem and adapt the classifier to those variations, one can either try to adapt test feature distributions to those of learning features thanks to unsupervised *domain adaptation* [103, 157, 167]. Several of these techniques (e.g., based on adversarial learning) try to align the distribution shift of test samples with that of the learning ones thanks to a repetitive learning procedure that can be greedy in computation time, especially with multispectral images. As we propose to learn raw images, adversarial domain adaptation with raw multispectral texture features becomes feasible.

### 3D analysis

Multispectral cameras provide flat views of vegetation. Therefore, the geometrical structure of crops and weeds cannot be retrieved, especially under high overlap among plants. Because this information may be relevant to distinguish crops and weeds, it should be interesting to investigate the potential of 3D imaging combined with 2D texture analysis in crop/weed recognition.





## Appendix A

# GBM steps and LightGBM features

### A.1 GBM steps for multiclass problem

Let us consider a learning dataset  $\mathcal{S}^{learn} = \{\mathbf{x}_j, \mathbf{y}_j\}_{j=1}^N$ , with  $N_C$  classes, where  $\{\mathbf{x}_j\}_{j=1}^N$  are the learning samples described by  $K$  features, and  $\mathbf{y}_j$  is the label of a given sample  $\mathbf{x}_j$  encoded as a one-hot vector of size  $N_C$  (with only one non-zero element, which is 1) such as:

$$y_j^i = \mathbb{1}(\mathbf{x}_j \in \mathcal{C}^i) = \begin{cases} 1 & \text{if } \mathbf{x}_j \in \mathcal{C}^i, \\ 0 & \text{otherwise,} \end{cases} \quad (\text{A.1})$$

where  $\mathcal{C}^i$  is the  $i$ -th class. Let us also consider a decision function  $\theta(\cdot)$  (provided by the weak learner  $\theta$ ), and a differentiable loss function  $L$  for vectorial inputs and its scalar version  $\phi$ .

The classification procedure using a GBM can be summarized by the following steps:

- Step 1: Initial prediction:

The  $N_C$  ensemble models are initialized with a per class value  $\gamma^i \in [0, 1]$  such that  $\theta_0^i(\{\mathbf{x}_j\}_{j=1}^N) = \gamma^i$ . Note that  $\gamma^i$  can be deduced by seeking the value that minimizes the loss function over all samples. However, to speed up the training phase, it is more common to initialize each model to the average class probability, or to zero [55].

- Step 2: Residual computation, model fitting and update:

For  $m = 1$  to  $M$  boosting iterations:

- Set  $\theta^i = \theta_{m-1}^i$  for  $i = 1, 2, \dots, N_C$ , thus,  $\Theta = \Theta_{m-1}$ .

For  $n = 1$  to  $N_C$  class iterations:

- Do steps 2.(a)–2.(d):

2.(a) Compute the pseudo-residuals  $r_{j,m}^n$  for each sample  $\mathbf{x}_j$  as:

$$r_{j,m}^n = - \left[ \frac{\partial L(\mathbf{y}_j, \boldsymbol{\Theta}(\mathbf{x}_j))}{\partial \theta^n(\mathbf{x}_j)} \right]. \quad (\text{A.2})$$

The minus sign in Eq. (A.2) reflects that it is a minimization challenge.

To solve this optimization problem in a multiclass classification context, the multinomial log-loss (or cross-entropy loss) is a commonly used function [70, p. 348]. It can be defined for a single sample  $\mathbf{x}_j$  and its label vector  $\mathbf{y}_j$  as :

$$L(\mathbf{y}_j, \boldsymbol{\Theta}(\mathbf{x}_j)) = - \sum_{i=1}^{N_C} y_j^i \cdot \log(\mathcal{P}^i(\mathbf{x}_j)). \quad (\text{A.3})$$

$\mathcal{P}^i(\mathbf{x}_j)$  is the posterior probability that sample  $\mathbf{x}_j$  belongs to class  $\mathcal{C}^i$  given the model  $\theta^i = \theta_{m-1}^i$  computed at the previous iteration. Note that in multiclass classification, the probability vector  $\mathcal{P}(\mathbf{x}_j) \in \mathbb{R}^{N_C}$  is usually obtained by applying the softmax function over the classification logits (raw scores)  $\{\theta^i(\mathbf{x}_j)\}_{i=1}^{N_C}$  provided by the multiclass learner  $\boldsymbol{\Theta} = \{\theta^i\}_{i=1}^{N_C}$ . The reason of using the softmax is to squash the scores into the range  $[0, 1]$  and to ensure that they sum up to 1, thereby fulfilling the constraints of a probability density [70, p. 348]. Thus, we can express each probability  $\mathcal{P}^i(\mathbf{x}_j)$  as:

$$\mathcal{P}^i(\mathbf{x}_j) = \frac{\exp(\theta^i(\mathbf{x}_j))}{\sum_{l=1}^{N_C} \exp(\theta^l(\mathbf{x}_j))}. \quad (\text{A.4})$$

We can then compute the pseudo-residual for sample  $\mathbf{x}_j$  by deriving Eq. (A.3) with regard to the score  $\theta^n(\mathbf{x}_j)$  of the current class  $\mathcal{C}^n$  as:

$$\frac{\partial L(\mathbf{y}_j, \boldsymbol{\Theta}(\mathbf{x}_j))}{\partial \theta^n(\mathbf{x}_j)} = - \sum_{i=1}^{N_C} \frac{y_j^i}{\mathcal{P}^i(\mathbf{x}_j)} \cdot \frac{\partial \mathcal{P}^i(\mathbf{x}_j)}{\partial \theta^n(\mathbf{x}_j)}, \quad (\text{A.5})$$

where

$$\frac{\partial \mathcal{P}^i(\mathbf{x}_j)}{\partial \theta^n(\mathbf{x}_j)} = \begin{cases} \frac{\exp(\theta^n(\mathbf{x}_j))}{\sum_{l=1}^{N_C} \exp(\theta^l(\mathbf{x}_j))} \cdot \left( \frac{\sum_{l=1}^{N_C} \exp(\theta^l(\mathbf{x}_j)) - \exp(\theta^n(\mathbf{x}_j))}{\sum_{l=1}^{N_C} \exp(\theta^l(\mathbf{x}_j))} \right) & \text{if } i = n, \\ -\frac{\theta^n(\mathbf{x}_j) \cdot \exp(\theta^i(\mathbf{x}_j))}{\left( \sum_{l=1}^{N_C} \exp(\theta^l(\mathbf{x}_j)) \right)^2} & \text{if } i \neq n. \end{cases} \quad (\text{A.6})$$

Using Eq. (A.4), we can rewrite Eq. (A.6) as:

$$\frac{\partial \mathcal{P}^i(\mathbf{x}_j)}{\partial \theta^n(\mathbf{x}_j)} = \begin{cases} \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j)) & \text{if } i = n, \\ -\mathcal{P}^n(\mathbf{x}_j) \cdot \mathcal{P}^i(\mathbf{x}_j) & \text{if } i \neq n. \end{cases} \quad (\text{A.7})$$

Plugging Eq. (A.7) into Eq. (A.5) yields:

$$\begin{aligned} \frac{\partial L(\mathbf{y}_j, \boldsymbol{\Theta}(\mathbf{x}_j))}{\partial \theta^n(\mathbf{x}_j)} &= -\frac{y_j^n}{\mathcal{P}^n(\mathbf{x}_j)} \cdot \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j)) + \sum_{i \neq n} \frac{y_j^i}{\mathcal{P}^i(\mathbf{x}_j)} \cdot \mathcal{P}^n(\mathbf{x}_j) \cdot \mathcal{P}^i(\mathbf{x}_j) \\ &= -y_j^n(1 - \mathcal{P}^n(\mathbf{x}_j)) + \sum_{i \neq n} y_j^i \mathcal{P}^n(\mathbf{x}_j) \\ &= -y_j^n + \mathcal{P}^n(\mathbf{x}_j) \left( y_j^n + \sum_{i \neq n} y_j^i \right) \\ &= \mathcal{P}^n(\mathbf{x}_j) - y_j^n, \end{aligned} \quad (\text{A.8})$$

where  $y_j^n + \sum_{i \neq n} y_j^i = 1$  since  $\mathbf{y}_j$  is a one-hot encoded vector. Thus, the residual  $r_{j,m}^n$  for sample  $\mathbf{x}_j$  is:

$$\begin{aligned} r_{j,m}^n &= -\left( \mathcal{P}^n(\mathbf{x}_j) - y_j^n \right) \\ &= y_j^n - \mathcal{P}^n(\mathbf{x}_j) \begin{cases} > 0 & \text{if } y_j^n = 1, \\ < 0 & \text{if } y_j^n = 0. \end{cases} \end{aligned} \quad (\text{A.9})$$

From Eq. (A.9), we can see that when  $y_j^n = 1$ ,  $\mathcal{P}^n(\mathbf{x}_j)$  should be as close to 1 as possible to minimize the loss, whereas when  $y_j^n = 0$ ,  $\mathcal{P}^n(\mathbf{x}_j)$  should be as close as possible to 0. In both cases, we seek to minimize the gradient whatever the class.

2.(b) Fit a regression tree to  $\{(\mathbf{x}_j, r_{j,m}^n)\}_{j=1}^N$ , then create terminal nodes  $\{\mathcal{R}_{t,m}^n\}_{t=1}^T$

such that  $\cap_{t=1}^T \mathcal{R}_{t,m}^n = \emptyset$  and  $\cup_{t=1}^T \mathcal{R}_{t,m}^n = \mathcal{S}^{learn}$ .

Note that the number of leaves  $T$  determines the size and complexity of the tree. There is no rule of thumb to find the optimal value of  $T$ . It is a tuning parameter that depends on the size and dimensionality of the dataset and can be deduced experimentally according to model accuracy and learning/inference speed.

A regression tree fits to continuous dependent variables. It is built thanks to a recursive splitting procedure. The root (or parent) node that regroups all the learning samples is split into two child nodes. Then, the child (or leaf) nodes are further split and become parents of their child nodes. In order to perform splitting, all features are scanned and for each feature, several thresholds (split points) are tested. The pair (feature, threshold) that minimizes the *impurity* within the resulting subsets is retained for splitting. To measure subset impurity, we usually rely on dispersion measured thanks to an objective function such as the mean squared error (MSE) or the residual sum of squares (RSS) [70, p. 307]. The splitting procedure of the resulting subsets is repeated until reaching some criterion such as the minimum number of samples in a leaf. Finally, for each terminal leaf, an output value is computed, usually as the average of the dependent variables associated to the learning samples regrouped in that leaf. The fitted regression tree can be then regarded as a set of decision rules (or *if* conditions) sequentially applied to the features characterizing the test samples. To perform a prediction, a test sample is passed through the fitted tree and assigned to a specific terminal leaf according to the learned rules. The predicted value for this test sample could basically be the leaf output value computed as the average value of the dependent variables associated to that leaf. Alternatively, a Gradient-boosted Decision Tree (GBDT) seeks to find an optimal output leaf value that minimizes the loss over all the samples belonging to that leaf thanks to the following optimization procedure.

2.(c) Compute the output value in each terminal node, for  $t = 1$  to  $T$ :

$$\gamma_{t,m}^n = \arg \min_{\gamma_m^n} \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j) + \gamma_m^n), \quad (\text{A.10})$$

where  $\phi(y_j^i, \theta^i(\mathbf{x}_j)) = -y_j^i \cdot \log(\mathcal{P}^i(\mathbf{x}_j))$ . Note that in this particular case, rewriting the loss  $L$  using the summation of the loss  $\phi$  over all the classes allows us to introduce the constant  $\gamma_m^n$  in the loss, which could not be possible by using  $L$  directly since  $\Theta(\mathbf{x}_j)$  is a vector.

Because deducing  $\gamma_{t,m}^n$  from Eq. (A.10) is quite complex, second-order approximations of the loss function are considered [55, 56]. For instance, recent state-of-the-art implementations of tree-based GBMs [29, 155] use a second-order Taylor expansion of  $\phi(y_j^i, \theta_{m-1}^i(\mathbf{x}_j) + \gamma_m^n)$  for small values of  $\gamma_m^n$ :

$$\begin{aligned} \gamma_{t,m}^n = \arg \min_{\gamma_m^n} & \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j)) + \frac{\partial \sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j))}{\partial \theta^n(\mathbf{x}_j)} \gamma_m^n \\ & + \frac{1}{2} \cdot \frac{\partial^2 \sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j))}{\partial^2 \theta^n(\mathbf{x}_j)} \gamma_m^n^2. \end{aligned} \quad (\text{A.11})$$

Deriving the function to be minimized in Eq. (A.11) with regard to  $\gamma_m^n$  and setting it equal to zero (in order to find  $\gamma_{t,m}^n$  according to Eq. (A.11)) yields:

$$\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \frac{\partial \sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j))}{\partial \theta^n(\mathbf{x}_j)} + \frac{\partial^2 \sum_{i=1}^{N_c} \phi(y_j^i, \theta_{m-1}^i(\mathbf{x}_j))}{\partial^2 \theta^n(\mathbf{x}_j)} \gamma_{t,m}^n = 0. \quad (\text{A.12})$$

Knowing that  $\sum_{i=1}^{N_c} \phi(y_j^i, \theta^i(\mathbf{x}_j)) = L(\mathbf{y}_j, \Theta(\mathbf{x}_j))$ , and using the result of Eq. (A.8), we can rewrite Eq. (A.12) as:

$$\begin{aligned} \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) - y_j^n + \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \frac{\partial^2 (\mathcal{P}^n(\mathbf{x}_j) - y_j^n)}{\partial^2 \theta^n(\mathbf{x}_j)} \gamma_{t,m}^n &= 0 \\ \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) - y_j^n + \sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j)) \gamma_{t,m}^n &= 0 \end{aligned} \quad (\text{A.13})$$

Finally, solving Eq. (A.13) for  $\gamma_{t,m}^n$  yields:

$$\begin{aligned}\gamma_{t,m}^n &= \frac{-\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) - y_j^n}{\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j))} \\ &= \frac{\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} y_j^n - \mathcal{P}^n(\mathbf{x}_j)}{\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j))}. \quad (\text{A.14}) \\ &= \frac{\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} r_{j,m}^n}{\sum_{\mathbf{x}_j \in \mathcal{R}_{t,m}^n} \mathcal{P}^n(\mathbf{x}_j) \cdot (1 - \mathcal{P}^n(\mathbf{x}_j))}\end{aligned}$$

where  $\mathcal{P}^n(\mathbf{x}_j)$  is the posterior probability that sample  $\mathbf{x}_j$  belongs to class  $\mathcal{C}^n$ , given  $\theta^n(\mathbf{x}_j)$  associated to the previous iteration.

2.(d) Update learner  $\theta_m^n$  for each sample  $\mathbf{x}_j$  as:

$$\theta_m^n(\mathbf{x}_j) = \theta_{m-1}^n(\mathbf{x}_j) + \epsilon \cdot \sum_{t=1}^T \gamma_{t,m}^n \cdot \mathbb{1}(\mathbf{x}_j \in \mathcal{R}_{t,m}^n) \quad \text{for } j = 1, \dots, \mathcal{N}, \quad (\text{A.15})$$

where  $\epsilon \in [0, 1]$  is the learning rate.

End for  $n$ .

End for  $m$ .

→ Final decision model  $\hat{\Theta} = \{\hat{\theta}^i\}_{i=1}^{N_c}$ , where  $\hat{\theta}^i(\cdot) = \epsilon \cdot \sum_{m=1}^M \theta_m^i(\cdot)$ .

The predicted label for a test sample  $\mathbf{x}_s \in \mathbb{R}^K$  is provided by Eq. (4.13).

## A.2 LightGBM features

LightGBM (LGBM) learns from  $\mathcal{S}^{learn}$  with reduced processing time in contrast to other GBM approaches, and bypasses the high dimensionality of its data thanks to *Exclusive Feature Bundling* (EFB) and *Gradient-based One-Side Sampling* (GOSS) techniques.

### A.2.1 Exclusive Feature Bundling (EFB)

EFB focuses on how to effectively reduce the feature space size. Specifically, the authors of [83] assume that a high dimensional feature space is usually sparse when several features can be mutually exclusive (i.e., do not take non-zero values simultaneously). Based on this assumption, they propose to bundle exclusive features into

a single feature called *exclusive feature bundle*. The sparsity assumption may not be always verified because several features may take simultaneously non-zero values. Thus, the algorithm tolerates a small fraction of conflicts between features. A conflict occurs when two features take simultaneously non-zero values. In other words, features that are almost exclusive can be also put in the same bundle if their conflicts is below the conflict ratio (defined as  $|S^{learn}|/10000$ ). To find candidate features to bundle, conflicts between the features are computed. The features are then sorted in descending order according to their total conflicts. Finally, each feature is analyzed in descending order according to its conflict degree (sum of its conflicts with all the remaining features) and the feasibility of combining (*bundling*) it with other ones depends on their conflict. If it is below the conflict ratio, then these features can be merged to form a feature bundle. Note that in practice, the authors propose to directly consider the count of non-zero values as a measure of conflict to avoid building the conflict matrix. They assume that the more a feature has non-zero values, the higher it might be in conflict with other features. EFB can be summarized by the following steps:

- 1 Sort features in descending order according to their number of non-zero values.
- 2 Analyze the features in descending order and find candidate ones to merge.
- 3 Merge the candidate features (see algorithm 4 of [83]) to obtain bundled features (new features), where  $\#bundles \ll K$ .

Table A.1 shows a simple example of conflict computation and feature sorting for a dataset composed of 10 samples and 5 features, using 5 conflicts as the conflict ratio. We start by feature  $f^1$  since it has the highest conflict degree and seek in descending order for a feature that can be merged with it. We can see that  $f^1$  has a conflict with features  $f^3$ ,  $f^5$ , and  $f^2$  superior to the conflict ratio, whereas it has a conflict with  $f^4$  inferior than the conflict ratio. Thus  $f^1$  and  $f^4$  can be merged to form *featureBundle*<sup>1,4</sup>. Then we consider feature  $f^3$  and seek a feature to be merged with it. We can see that the conflict between  $f^3$  and  $f^5$  is superior to the conflict ratio whereas it is inferior with  $f^2$ . Thus,  $f^3$  and  $f^5$  will be merged to form *featureBundle*<sup>3,2</sup>. The remaining feature  $f^5$  is put on its own bundle *featureBundle*<sup>5</sup>.



|          | $f^1$ | $f^2$ | $f^3$ | $f^4$ | $f^5$ |
|----------|-------|-------|-------|-------|-------|
| $x_1$    | 2     | 3     | 0     | 2     | 0     |
| $x_2$    | 3     | 0     | 1     | 0     | 4     |
| $x_3$    | 0     | 0     | 2     | 0     | 7     |
| $x_4$    | 10    | 8     | 13    | 10    | 0     |
| $x_5$    | 5     | 6     | 15    | 0     | 3     |
| $x_6$    | 15    | 12    | 2     | 0     | 12    |
| $x_7$    | 12    | 0     | 5     | 6     | 4     |
| $x_8$    | 15    | 1     | 0     | 9     | 15    |
| $x_9$    | 0     | 0     | 3     | 7     | 0     |
| $x_{10}$ | 1     | 1     | 9     | 0     | 2     |

(a) Data set. Conflicts between  $f^1$  and  $f^2$  are highlighted with yellow color.

|       | $f^1$ | $f^2$ | $f^3$ | $f^4$ | $f^5$ |
|-------|-------|-------|-------|-------|-------|
| $f_1$ | -     | 6     | 6     | 4     | 6     |
| $f_2$ | 6     | -     | 4     | 3     | 3     |
| $f_3$ | 6     | 4     | -     | 3     | 6     |
| $f_4$ | 4     | 3     | 3     | -     | 2     |
| $f_5$ | 6     | 3     | 6     | 2     | -     |

(b) Conflict matrix of the features deduced from (a).

|                 | $f^1$ | $f^3$ | $f^5$ | $f^2$ | $f^4$ |
|-----------------|-------|-------|-------|-------|-------|
| Conflict degree | 22    | 19    | 17    | 16    | 12    |

(c) Features ordered by their total conflicts deduced from (b).

TABLE A.1: Conflict computation and feature ordering.

EFB can be seen as a preprocessing step that can considerably improve computational efficiency (see Tab. 2 of [83]). After applying EFB, the feature space dimension is reduced from  $K$  to  $\#bundles$ . Note that in case of highly dense features, the improvement of computational efficiency obtained using EFB is not always guaranteed since there might be no or very few features to bundle (see also Tab. 2 of [83]).

### A.2.2 Gradient-based One-Side Sampling (GOSS)

GOSS focuses on effectively reducing the number of learning samples while keeping as much information as possible about the entire dataset. In order to effectively reduce the number of learning samples, GOSS uses a sampling approach based on the gradients. Each sample in  $\mathcal{S}^{learn}$  is now represented by a  $\#bundles$ -dimensional feature vector. GOSS first sorts the learning samples in descending order according to the absolute values of their gradients. Then, the top  $a \cdot |\mathcal{S}^{learn}|$  ( $a$  is a sampling ratio) of learning samples with highest gradients are kept in order to obtain a subset denoted as  $\mathcal{A}$ . The remaining learning samples  $(1 - a) \cdot |\mathcal{S}^{learn}|$  with lower gradients are randomly sampled according to a sampling ratio  $b$  to obtain a subset  $\mathcal{B}$ . Finally, the subset  $\mathcal{A} \cup \mathcal{B}$  is used for further computations (information gain computation,

split point). The assumption behind this approach is that samples with low gradients would have smaller training errors. Thus, their number might be reduced. Note that in order to have a data distribution in the set  $\mathcal{A} \cup \mathcal{B}$  similar to that of the original set  $\mathcal{S}^{learn}$ , a constant multiplier  $(1-a)/b$  is introduced for the selected lower gradient samples in order to normalize their sum back to the size of the set  $(1-a) \cdot |\mathcal{S}^{learn}|$ . In the end, by combining EFB and GOSS, the complexity of histogram building reduces from  $\mathcal{O}(|\mathcal{S}^{learn}| \times K)$  to  $\mathcal{O}(|\mathcal{A} \cup \mathcal{B}| \times \#bundles)$ , where  $|\mathcal{A} \cup \mathcal{B}| \ll |\mathcal{S}^{learn}|$  and  $\#bundles \ll K$ . This drastically reduces memory and time consumption. The final prediction of LGBM is then made according to Eq. (4.13).



# Bibliography

- [1] Three-hourly global solar irradiance data in France, by region. [Online]. Available: <https://www.data.gouv.fr/fr/datasets/rayonnement-solaire-global-et-vitesse-du-vent-a-100-metres-tri-horaires-regionaux-depuis-janvier-2016/>
- [2] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Proceedings of the 8th International Conference on Database Theory (ICDT'01)*, London, UK, October 2001, pp. 420–434. doi:[10.1007/3-540-44503-X\\_27](https://doi.org/10.1007/3-540-44503-X_27)
- [3] H. A. A. Alfeilat, A. B. Hassanat, O. Lasassmeh, A. S. Tarawneh, M. B. Alhasanat, H. S. E. Salman, and V. S. Prasath, "Effects of distance measure choice on k-nearest neighbor classifier performance: A review," *Big Data*, vol. 7, no. 4, pp. 221–248, December 2019. doi:[10.1089/big.2018.0175](https://doi.org/10.1089/big.2018.0175)
- [4] K. Alibabaei, P. D. Gaspar, T. M. Lima, R. M. Campos, I. Girão, J. Monteiro, and C. M. Lopes, "A review of the challenges of using deep learning algorithms to support decision-making in agricultural activities," *Remote Sensing*, vol. 14, no. 3, January 2022. doi:[10.3390/rs14030638](https://doi.org/10.3390/rs14030638)
- [5] A. Alsuwaidi, B. Grieve, and H. Yin, "Combining spectral and texture features in hyperspectral image analysis for plant monitoring," *Measurement Science and Technology*, vol. 29, no. 10, p. 104001, August 2018. doi:[10.1088/1361-6501/aad642](https://doi.org/10.1088/1361-6501/aad642)
- [6] A. Amziane, O. Losson, B. Mathon, A. Duménil, and L. Macaire, "Frame-based reflectance estimation from multispectral images for weed identification in varying illumination conditions," in *Proceedings of the 2020 Tenth International*

- Conference on Image Processing Theory, Tools and Applications (IPTA'20)*, Paris, France, November 2020, pp. 1–7. doi:[10.1109/IPTA50016.2020.9286692](https://doi.org/10.1109/IPTA50016.2020.9286692)
- [7] A. Amziane, O. Losson, B. Mathon, A. Duménil, and L. Macaire, “Reflectance estimation from multispectral linescan acquisitions under varying illumination—Application to outdoor weed identification,” *Sensors*, vol. 21, no. 11, p. 3601, May 2021. doi:[10.3390/s21113601](https://doi.org/10.3390/s21113601)
- [8] A. Amziane, O. Losson, B. Mathon, L. Macaire, and A. Duménil, “Weed detection by analysis of multispectral images acquired under uncontrolled illumination conditions,” in *Proceedings of the 15th International Conference on Quality Control by Artificial Vision (QCAV'21)*, vol. 11794, Tokushima, Japan (virtual), May 2021, pp. 286–293. doi:[10.1117/12.2586823](https://doi.org/10.1117/12.2586823)
- [9] G. L. Anderson, J. H. Everitt, A. J. Richardson, and D. E. Escobar, “Using satellite data to map false broomweed (*Ericameria austrotexana*) infestations on south Texas rangelands,” *Weed Technology*, vol. 7, no. 4, pp. 865–871, 1993. doi:[10.1017/S0890037X00037908](https://doi.org/10.1017/S0890037X00037908)
- [10] G. A. Antonucci, S. Vary, D. Humphreys, R. A. Lamb, J. Piper, and J. Tanner, “Multispectral snapshot demosaicing via non-convex matrix completion,” in *Proceedings of the 2019 IEEE Data Science Workshop (DSW)*, Minneapolis, MN, USA, June 2019, pp. 227–231. doi:[10.1109/DSW.2019.8755561](https://doi.org/10.1109/DSW.2019.8755561)
- [11] M. Á. C. Apaza, H. M. B. Monzón, and R. P. A. Garrido, “Semantic segmentation of weeds and crops in multispectral images by using a convolutional neural networks based on U-Net,” in *Proceedings of the 1st International Conference on Applied Technologies (ICAT'19)*, Quito, Ecuador, December 2019, pp. 473–485. doi:[10.1007/978-3-030-42520-3\\_38](https://doi.org/10.1007/978-3-030-42520-3_38)
- [12] C. Aquilani, A. Confessore, R. Bozzi, F. Sirtori, and C. Pugliese, “Review: Precision livestock farming technologies in pasture-based livestock systems,” *Animal*, vol. 16, no. 1, p. 100429, January 2022. doi:[10.1016/j.animal.2021.100429](https://doi.org/10.1016/j.animal.2021.100429)
- [13] B. Arad, R. Timofte, R. Yahel, N. Morag, A. Bernat, Y. Wu, X. Wu, Z. Fan, C. Xia, F. Zhang, S. Liu, Y. Li, C. Feng, L. Lei, M. Zhang, K. Feng *et al.*, “Ntire 2022 spectral demosaicing challenge and data set,” in *Proceedings of*

- the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'22)*, New Orleans, Louisiana, USA, June 2022, pp. 881–895. doi:[10.1109/CVPRW56347.2022.00103](https://doi.org/10.1109/CVPRW56347.2022.00103)
- [14] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017. doi:[10.1109/TPAMI.2016.2644615](https://doi.org/10.1109/TPAMI.2016.2644615)
- [15] M. D. Bah, A. Hafiane, and R. Canals, “Deep learning with unsupervised data labeling for weed detection in line crops in UAV images,” *Remote Sensing*, vol. 10, no. 11, 2018. doi:[10.3390/rs10111690](https://doi.org/10.3390/rs10111690)
- [16] O. Bawden, J. Kulk, R. Russell, C. McCool, A. English, F. Dayoub, C. Lehnert, and T. Perez, “Robot for weed species plant-specific management,” *Journal of Field Robotics*, vol. 34, no. 6, pp. 1179–1199, 2017. doi:[10.1002/rob.21727](https://doi.org/10.1002/rob.21727)
- [17] A. Bender, B. Whelan, and S. Sukkarieh, “A high-resolution, multimodal data set for agricultural robotics: A Ladybird’s-eye view of Brassica,” *Journal of Field Robotics*, vol. 37, no. 1, pp. 73–96, January 2020. doi:[10.1002/rob.21877](https://doi.org/10.1002/rob.21877). [Online]. Available: <https://researchdata.edu.au/ladybird-cobbitty-2017-brassica-dataset/1370660>
- [18] G. Bianco, F. Bruno, and M. Muzzupappa, “Multispectral data cube acquisition of aligned images for document analysis by means of a filter-wheel camera provided with focus control,” *Journal of Cultural Heritage*, vol. 14, no. 3, pp. 190–200, 2013. doi:[10.1016/j.culher.2012.07.002](https://doi.org/10.1016/j.culher.2012.07.002)
- [19] S. Bianco, C. Cusano, and R. Schettini, “Single and multiple illuminant estimation using convolutional neural networks,” *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4347–4362, September 2017. doi:[10.1109/TIP.2017.2713044](https://doi.org/10.1109/TIP.2017.2713044)
- [20] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, “Hyperspectral unmixing overview: Geometrical, statistical,

- and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, 2012. doi:[10.1109/JSTARS.2012.2194696](https://doi.org/10.1109/JSTARS.2012.2194696)
- [21] B. Bonicelli, O. Naud, S. Rousset, C. Sinfort, V. Rudnicki, J.-M. Lescot, B. Ruelle, L. Scheyer, and E. Cotteux, "The challenge for precision spraying," in *AgEng 2010: International Conference on Agricultural Engineering*, Clermont-Ferrand, France, September 2010, p. 11. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-00650005/document>
- [22] P. Bosilj, T. Duckett, and G. Cielniak, "Analysis of morphology-based features for classification of crop and weeds in precision agriculture," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2950–2956, October 2018. doi:[10.1109/LRA.2018.2848305](https://doi.org/10.1109/LRA.2018.2848305)
- [23] P. Bosilj, E. Aptoula, T. Duckett, and G. Cielniak, "Transfer learning between crop types for semantic segmentation of crops versus weeds in precision agriculture," *Journal of Field Robotics*, vol. 37, no. 1, pp. 7–19, March 2020. doi:[10.1002/rob.21869](https://doi.org/10.1002/rob.21869). [Online]. Available: <https://lcas.lincoln.ac.uk/wp/research/data-sets-software/crop-vs-weed-discrimination-dataset>
- [24] M.-A. Bourgeon, J.-N. Paoli, G. Jones, S. Villette, and C. Gée, "Field radiometric calibration of a multispectral on-the-go sensor dedicated to the characterization of vineyard foliage," *Computers and Electronics in Agriculture*, vol. 123, pp. 184–194, April 2016. doi:[10.1016/j.compag.2016.02.019](https://doi.org/10.1016/j.compag.2016.02.019)
- [25] J. Brauers and T. Aach, "A color filter array based multispectral camera," in *12. Workshop Farbbildverarbeitung*, Illmenau, Germany, October 2006, pp. 55–64.
- [26] M. Brown and S. Süsstrunk, "Multi-spectral sift for scene category recognition," in *Proceedings of the 24th IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11)*, Colorado Springs, CO, USA, June 2011, pp. 177–184. doi:[10.1109/CVPR.2011.5995637](https://doi.org/10.1109/CVPR.2011.5995637)

- [27] J. Champ, A. Mora-Fallas, H. Goëau, E. Mata-Montero, P. Bonnet, and A. Joly, "Instance segmentation for the fine detection of crop and weed plants by precision agricultural robots," *Applications in Plant Sciences*, vol. 8, no. 7, p. e11373, July 2020. doi:[10.1002/aps3.11373](https://doi.org/10.1002/aps3.11373)
- [28] N. Chebrolu, P. Lottes, A. Schaefer, W. Winterhalter, W. Burgard, and C. Stachniss, "Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1045–1052, July 2017. doi:[10.1177/0278364917720510](https://doi.org/10.1177/0278364917720510). [Online]. Available: <http://www.ipb.uni-bonn.de/data/sugarbeets2016/>
- [29] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, August 2016, pp. 785–794. doi:[10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785)
- [30] J.-H. Cheng, H. Jin, Z. Xu, and F. Zheng, "NIR hyperspectral imaging with multivariate analysis for measurement of oil and protein content in peanut varieties," *Analytical Methods*, vol. 9, October 2017. doi:[10.1039/C7AY02115A](https://doi.org/10.1039/C7AY02115A)
- [31] R. J. Chu, N. Richard, H. Chatoux, C. Fernandez-Maloigne, and J. Y. Hardeberg, "Hyperspectral texture metrology based on joint probability of spectral and spatial distribution," *IEEE Transactions on Image Processing*, vol. 30, pp. 4341–4356, April 2021. doi:[10.1109/TIP.2021.3071557](https://doi.org/10.1109/TIP.2021.3071557)
- [32] I. Cisternas, I. Velásquez, A. Caro, and A. Rodríguez, "Systematic literature review of implementations of precision agriculture," *Computers and Electronics in Agriculture*, vol. 176, p. 105626, September 2020. doi:[10.1016/j.compag.2020.105626](https://doi.org/10.1016/j.compag.2020.105626)
- [33] A. R. Cortes, "Multispectral analysis and spectral reflectance reconstruction of art paintings," Ph.D. dissertation, Télécom ParisTech, December 2003.
- [34] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, September 1995. doi:[10.1007/BF00994018](https://doi.org/10.1007/BF00994018)



- [35] P. Cunningham, "Dimension Reduction," in *Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval*. Springer, Berlin, Heidelberg, 2008, pp. 91–112. doi:[10.1007/978-3-540-75171-7\\_4](https://doi.org/10.1007/978-3-540-75171-7_4)
- [36] C. Cusano, P. Napoletano, and R. Schettini, "Local angular patterns for color texture classification," in *New Trends in Image Analysis and Processing – ICIAP 2015 Workshops*, vol. 9281, Genoa, Italy, September 2015, pp. 111–118. doi:[10.1007/978-3-319-23222-5\\_14](https://doi.org/10.1007/978-3-319-23222-5_14)
- [37] S. Del Pozo, P. Rodríguez-Gonzálvez, D. Hernández-López, and B. Felipe-García, "Vicarious radiometric calibration of multispectral camera on board unmanned aerial system," *Remote Sensing*, vol. 6, pp. 1918–1937, February 2014. doi:[10.3390/rs6031918](https://doi.org/10.3390/rs6031918)
- [38] M. Descour and E. Dereniak, "Computed-tomography imaging spectrometer: experimental calibration and reconstruction results," *Applied Optics*, vol. 34, no. 22, pp. 4817–4826, August 1995. doi:[10.1364/AO.34.004817](https://doi.org/10.1364/AO.34.004817)
- [39] B. M. Devassy, S. George, and P. Nussbaum, "Unsupervised clustering of hyperspectral paper data using t-SNE," *Journal of Imaging*, vol. 6, no. 5, May 2020. doi:[10.3390/jimaging6050029](https://doi.org/10.3390/jimaging6050029)
- [40] C. W. Dirk, M. F. Delgado, M. Olguin, and J. Druzik, "A prism–grating–prism spectral imaging approach," *Studies in Conservation*, vol. 54, no. 2, pp. 77–89, 2009. doi:[10.1179/sic.2009.54.2.77](https://doi.org/10.1179/sic.2009.54.2.77)
- [41] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning (ICML'14)*, vol. 32, no. 1, Beijing, China, June 2014, pp. 647–655. [Online]. Available: <https://proceedings.mlr.press/v32/donahue14.html>
- [42] M. S. Drew and G. D. Finlayson, "Analytic solution for separating spectra into illumination and surface reflectance components," *Journal of the Optical Society of America. A*, vol. 24, pp. 294–303, March 2007. doi:[10.1364/JOSAA.24.000294](https://doi.org/10.1364/JOSAA.24.000294)
- [43] M. Dyrmann, A. K. Mortensen, H. S. Midtiby, and R. N. Jørgensen, "Pixel-wise classification of weeds and crop in images by using a fully convolutional

- neural network," in *Proceedings of the 4th CIGR International Conference of Agricultural Engineering (CIGR-AgEng 2016)*, Aarhus, Denmark, June 2016.
- [44] J. Eckhard, T. Eckhard, E. M. Valero, J. L. Nieves, and E. G. Contreras, "Outdoor scene reflectance measurements using a Bragg-grating-based hyperspectral imager," *Applied Optics*, vol. 54, no. 13, pp. D15–D24, May 2015. doi:[10.1364/AO.54.000D15](https://doi.org/10.1364/AO.54.000D15)
- [45] E. Eibenberger and E. Angelopoulou, "The narrow-band assumption in log-chromaticity space," in *Proceedings of the 11th European Conference on Computer Vision (ECCV'10)*, Heraklion, Crete, Greece, September 2010, pp. 76–89. doi:[10.1007/978-3-642-35740-4\\_7](https://doi.org/10.1007/978-3-642-35740-4_7)
- [46] G. Elmasry, M. Kamruzzaman, D.-W. Sun, and P. Allen, "Principles and applications of hyperspectral imaging in quality evaluation of agro-food products: A review," *Critical Reviews in Food Science and Nutrition*, vol. 52, pp. 999–1023, 2012. doi:[10.1080/10408398.2010.543495](https://doi.org/10.1080/10408398.2010.543495)
- [47] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 652–675, March 2013. doi:[10.1109/JPROC.2012.2197589](https://doi.org/10.1109/JPROC.2012.2197589)
- [48] M. Fawakherji, A. Youssef, D. D. Bloisi, A. Pretto, and D. Nardi, "Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation," in *Proceedings of the 3rd IEEE International Conference on Robotic Computing (IRC'19)*, Naples, Italy, February 2019, pp. 146–152. doi:[10.1109/IRC.2019.00029](https://doi.org/10.1109/IRC.2019.00029)
- [49] F. Feyaerts and L. Van Gool, "Multi-spectral vision system for weed detection," *Pattern Recognition Letters*, vol. 22, no. 6, pp. 667–674, 2001. doi:[10.1016/S0167-8655\(01\)00006-X](https://doi.org/10.1016/S0167-8655(01)00006-X)
- [50] G. D. Finlayson, "Colour and illumination in computer vision," *Interface Focus*, vol. 8, no. 4, 2018. doi:[10.1098/rsfs.2018.0008](https://doi.org/10.1098/rsfs.2018.0008)

- [51] G. D. Finlayson and S. D. Hordley, "Color constancy at a pixel," *Journal of the Optical Society of America A*, vol. 18, no. 2, p. 253, February 2001. doi:[10.1364/JOSAA.18.000253](https://doi.org/10.1364/JOSAA.18.000253)
- [52] D. H. Foster, "Color constancy," *Vision Research*, vol. 51, no. 7, pp. 674–700, April 2011. doi:[10.1016/j.visres.2010.09.006](https://doi.org/10.1016/j.visres.2010.09.006)
- [53] D. H. Foster, K. Amano, S. M. C. Nascimento, and M. J. Foster, "Frequency of metamerism in natural scenes," *Journal of the Optical Society of America. A*, vol. 23, pp. 2359–72, November 2006. doi:[10.1364/JOSAA.23.002359](https://doi.org/10.1364/JOSAA.23.002359)
- [54] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proceedings of the 13th International Conference on Machine Learning (ICML 1996)*, Bari, Italy, July 1996, pp. 148–156. [Online]. Available: <https://cseweb.ucsd.edu/~yfreund/papers/boostingexperiments.pdf>
- [55] J. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, October 2001. doi:[10.1214/aos/1013203451](https://doi.org/10.1214/aos/1013203451)
- [56] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors)," *The Annals of Statistics*, vol. 28, no. 2, pp. 337–407, April 2000. doi:[10.1214/aos/1016218223](https://doi.org/10.1214/aos/1016218223)
- [57] P. B. Garcia-Allende, O. M. Conde, M. Amado, A. Quintela, and J. M. Lopez-Higuera, "Hyperspectral data processing algorithm combining principal component analysis and K nearest neighbours," in *Proceedings of the SPIE Defense and Security Symposium: Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XIV*, vol. 6966, Orlando, FL, USA, April 2008, pp. 127–135. doi:[10.1117/12.770298](https://doi.org/10.1117/12.770298)
- [58] N. Gat, "Imaging spectroscopy using tunable filters: a review," in *Proceedings of the SPIE AeroSense Conference: Wavelet Applications VII*, vol. 4056, Orlando, FL, USA, April 2000, pp. 50–64. doi:[10.1117/12.381686](https://doi.org/10.1117/12.381686)

- [59] A. Gebejes, I. Tomić, I. Karlovits, and I. Jurič, "Evaluation of the algorithms for recovering reflectance from virtual digital camera response," *Journal of Graphic Engineering and Design*, vol. 3, pp. 23–30, October 2012.
- [60] B. Geelen, N. Tack, and A. Lambrechts, "A compact snapshot multispectral imager with a monolithically integrated per-pixel filter mosaic," in *Proceedings of the SPIE: Advanced Fabrication Technologies for Micro/Nano Optics and Photonics VII*, vol. 8974, San Francisco, CA, USA, February 2014, pp. 89 740L–89 740L–8. doi:[10.1117/12.2037607](https://doi.org/10.1117/12.2037607)
- [61] R. Gerhards and S. Christensen, "Real-time weed detection, decision making and patch spraying in maize, sugarbeet, winter wheat and winter barley," *Weed Research*, vol. 43, no. 6, pp. 385–392, 2003. doi:[10.1046/j.1365-3180.2003.00349.x](https://doi.org/10.1046/j.1365-3180.2003.00349.x)
- [62] T. M. Goodman, "18 - International standards for colour," in *Colour Design (Second Edition)*, second edition ed., J. Best, Ed. Woodhead Publishing, 2012, pp. 417–452. doi:[10.1016/B978-0-08-101270-3.00018-7](https://doi.org/10.1016/B978-0-08-101270-3.00018-7)
- [63] T. Goossens, B. Geelen, J. Pichette, A. Lambrechts, and C. Van Hoof, "Finite aperture correction for spectral cameras with integrated thin film Fabry-Perot filters," *Applied Optics*, vol. 57, no. 26, pp. 7539–7549, September 2018. doi:[10.1364/AO.57.007539](https://doi.org/10.1364/AO.57.007539)
- [64] L. Grimstad and P. J. From, "The Thorvald II agricultural robotic system," *Robotics*, vol. 6, no. 4, September 2017. doi:[10.3390/robotics6040024](https://doi.org/10.3390/robotics6040024)
- [65] Q. Gu, Z. Li, and J. Han, "Generalized Fisher score for feature selection," in *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI'11)*, Barcelona, Spain, July 2011, pp. 266–273. [Online]. Available: <https://faculty.ist.psu.edu/jessieli/Publications/uai11.pdf>
- [66] R. Gupta and R. I. Hartley, "Linear pushbroom cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 9, pp. 963–975, September 1997. doi:[10.1109/34.615446](https://doi.org/10.1109/34.615446)

- [67] N. Hagen and M. W. Kudenov, "Review of snapshot spectral imaging technologies," *Optical Engineering*, vol. 52, no. 9, p. 090901, September 2013. doi:[10.1117/1.OE.52.9.090901](https://doi.org/10.1117/1.OE.52.9.090901)
- [68] N. Hagen, E. L. Dereniak, and D. T. Sass, "Fourier methods of improving reconstruction speed for CTIS imaging spectrometers," in *Proceedings of SPIE Imaging Spectrometry XII*, vol. 6661, San Diego, CA, USA, 2007, pp. 15–25. doi:[10.1117/12.732669](https://doi.org/10.1117/12.732669)
- [69] B. Hartzler, "The cost of convenience: The impact of weeds on crop yields," in *Proceedings of the 21st Annual Integrated Crop Management Conference*, Ames, IA, USA, December 2009. doi:[10.31274/icm-180809-11](https://doi.org/10.31274/icm-180809-11)
- [70] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York, NY, USA: Springer, 2009.
- [71] S. Haug and J. Ostermann, "A crop/weed field image dataset for the evaluation of computer vision based precision agriculture tasks," in *Proceedings of the 2014 European Conference on Computer Vision (ECCV'14)*, Zurich, Switzerland, September 2014, pp. 105–116. doi:[10.1007/978-3-319-16220-1\\_8](https://doi.org/10.1007/978-3-319-16220-1_8). [Online]. Available: <https://github.com/cwfid/dataset>
- [72] S. Haug, A. Michaels, P. Biber, and J. Ostermann, "Plant classification system for crop/weed discrimination without segmentation," in *Proceedings of the 2014 IEEE Winter Conference on Applications of Computer Vision*, Steamboat Springs, CO, USA, March 2014, pp. 1142–1149. doi:[10.1109/WACV.2014.6835733](https://doi.org/10.1109/WACV.2014.6835733)
- [73] H. He and Y. Ma, *Imbalanced Learning: Foundations, Algorithms, and Applications*, 1st ed. Wiley-IEEE Press, 2013.
- [74] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV'15)*, Santiago, Chile, December 2015, pp. 1026–1034. doi:[10.1109/ICCV.2015.123](https://doi.org/10.1109/ICCV.2015.123)

- [75] V. Heikkinen, "Spectral reflectance estimation using Gaussian processes and combination kernels," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3358–3373, July 2018. doi:[10.1109/TIP.2018.2820839](https://doi.org/10.1109/TIP.2018.2820839)
- [76] J. Hemming and T. Rath, "Computer-vision-based weed identification under field conditions using controlled lighting," *Journal of Agricultural Engineering Research*, vol. 78, pp. 233–243, March 2001. doi:[10.1006/jaer.2000.0639](https://doi.org/10.1006/jaer.2000.0639)
- [77] M. Hercher, "The spherical mirror Fabry-Perot interferometer," *Applied Optics*, vol. 7, no. 5, May 1968. doi:[10.1364/AO.7.000951](https://doi.org/10.1364/AO.7.000951)
- [78] A. Hurlbert, "Formal connections between lightness algorithms." *Journal of the Optical Society of America A*, vol. 3, no. 10, pp. 1684–93, October 1986. doi:[10.1364/JOSAA.3.001684](https://doi.org/10.1364/JOSAA.3.001684)
- [79] A. Ibrahim, S. Tominaga, and T. Horiuchi, "Invariant representation for spectral reflectance images and its application," *EURASIP Journal on Image and Video Processing*, no. 2, pp. 1–12, June 2011. doi:[10.1186/1687-5281-2011-2](https://doi.org/10.1186/1687-5281-2011-2)
- [80] N. Jain, S. S. Ray, J. P. Singh, and S. Panigrahy, "Use of hyperspectral data to assess the effects of different nitrogen applications on a potato crop," *Precision Agriculture*, vol. 8, no. 4-5, pp. 225–239, October 2007. doi:[10.1007/s11119-007-9042-0](https://doi.org/10.1007/s11119-007-9042-0)
- [81] J. R. Johnson, "A high resolution scanning confocal interferometer," *Applied Optics*, vol. 7, no. 6, pp. 1061–1072, June 1968. doi:[10.1364/AO.7.001061](https://doi.org/10.1364/AO.7.001061)
- [82] R. Jolivot, P. Vabres, and F. Marzani, "Reconstruction of hyperspectral cutaneous data from an artificial neural network-based multispectral imaging system," *Computerized Medical Imaging and Graphics*, vol. 35, no. 2, pp. 85–88, March 2011. doi:[10.1016/j.compmedimag.2010.07.001](https://doi.org/10.1016/j.compmedimag.2010.07.001)
- [83] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS'17)*, vol. 30, Long Beach, CA, USA, December 2017, pp. 3146–3154. [Online]. Available: <http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf>

- [84] H. A. Khan, J.-B. Thomas, and J. Y. Hardeberg, "Multispectral constancy based on spectral adaptation transform," in *Proceedings of the 20th Scandinavian Conference on Image Analysis (SCIA 2017)*, vol. 10270, Tromsø, Norway, June 2017, pp. 459–470. doi:[10.1007/978-3-319-59129-2\\_39](https://doi.org/10.1007/978-3-319-59129-2_39)
- [85] H. A. Khan, J.-B. Thomas, J. Y. Hardeberg, and O. Laligant, "Illuminant estimation in multispectral imaging," *Journal of the Optical Society of America A*, vol. 34, no. 7, pp. 1085–1098, July 2017. doi:[10.1364/JOSAA.34.001085](https://doi.org/10.1364/JOSAA.34.001085)
- [86] H. A. Khan, S. Mihoubi, B. Mathon, J.-B. Thomas, and J. Y. Hardeberg, "HyTex-iLa: High resolution visible and near infrared hyperspectral texture images," *Sensors*, vol. 18, no. 7, p. 2045, June 2018. doi:[10.3390/s18072045](https://doi.org/10.3390/s18072045)
- [87] H. A. Khan, J.-B. Thomas, J. Y. Hardeberg, and O. Laligant, "Multispectral camera as spatio-spectrophotometer under uncontrolled illumination," *Optics Express*, vol. 27, no. 2, pp. 1051–1070, January 2019. doi:[10.1364/OE.27.001051](https://doi.org/10.1364/OE.27.001051)
- [88] R. Khanna, I. Sa, J. Nieto, and R. Siegwart, "On field radiometric calibration for multispectral cameras," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA'17)*, Singapore, May 2017, pp. 6503–6509. doi:[10.1109/ICRA.2017.7989768](https://doi.org/10.1109/ICRA.2017.7989768)
- [89] J. Khodr and R. Younes, "Dimensionality reduction on hyperspectral images: A comparative review based on artificial datas," in *Proceedings of the 4th International Congress on Image and Signal Processing (CISP'11)*, vol. 5, Shanghai, China, October 2011. doi:[10.1109/CISP.2011.6100531](https://doi.org/10.1109/CISP.2011.6100531)
- [90] M. Kim and H. Byun, "Learning texture invariant representation for domain adaptation of semantic segmentation," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, Seattle, WA, USA, June 2020, pp. 12 972–12 981. doi:[10.1109/CVPR42600.2020.01299](https://doi.org/10.1109/CVPR42600.2020.01299)
- [91] V. Kitanovski, J.-B. Thomas, and J. Y. Hardeberg, "Reflectance estimation from snapshot multispectral images captured under unknown illumination," in *Proceedings of the 29th Color and Imaging Conference*, vol. 29. Virtual event: Society for Imaging Science and Technology, November 2021, pp. 264–269. doi:[10.2352/issn.2169-2629.2021.29.264](https://doi.org/10.2352/issn.2169-2629.2021.29.264)



- [92] J. Klein, J. Brauers, and T. Aach, "Methods for spectral characterization of multispectral cameras," in *Proceedings of the SPIE: Digital Photography VII*, vol. 7876, San Francisco, CA, USA, January 2011, pp. 102–112. doi:[10.1117/12.872432](https://doi.org/10.1117/12.872432)
- [93] A. Kordecki, H. Palus, and A. Bal, "Practical vignetting correction method for digital camera with measurement of surface luminance distribution," *Signal, Image and Video Processing*, vol. 10, no. 8, pp. 1417–1424, November 2016. doi:[10.1007/s11760-016-0941-2](https://doi.org/10.1007/s11760-016-0941-2)
- [94] D. W. Lamb and R. B. Brown, "Remote-sensing and mapping of weeds in crops," *Journal of Agricultural Engineering Research*, vol. 78, no. 2, pp. 117–125, 2001. doi:<https://doi.org/10.1006/jaer.2000.0630>
- [95] P.-J. Lapray, X. Wang, J.-B. Thomas, and P. Gouton, "Multispectral filter arrays: Recent advances and practical implementation," *Sensors*, vol. 14, no. 11, pp. 21 626–21 659, November 2014. doi:[10.3390/s141121626](https://doi.org/10.3390/s141121626)
- [96] L. W. Lass, T. S. Prather, N. F. Glenn, K. T. Weber, J. T. Mundt, and J. Pettingill, "A review of remote sensing of invasive weeds and example of the early detection of spotted knapweed (*Centaurea maculosa*) and babysbreath (*Gypsophila paniculata*) with a hyperspectral sensor," *Weed Science*, vol. 53, no. 2, p. 242–251, 2005. doi:[10.1614/WS-04-044R2](https://doi.org/10.1614/WS-04-044R2)
- [97] V. W. S. Lee, D. C. Slaughter, and D. K. Giles, "Robotic weed control system for tomatoes," *Precision Agriculture*, vol. 1, pp. 95–113, January 1999. doi:[10.1023/A:1009977903204](https://doi.org/10.1023/A:1009977903204)
- [98] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, November 2017. doi:[10.1145/3136625](https://doi.org/10.1145/3136625)
- [99] F. Lin, D. Zhang, Y. Huang, X. Wang, and X. Chen, "Detection of corn and weed species by the combination of spectral, shape and textural features," *Sustainability*, vol. 9, no. 8, 2017. doi:[10.3390/su9081335](https://doi.org/10.3390/su9081335)
- [100] M. Lin, Q. Chen, and S. Yan, "Network In Network," in *Proceedings of the 2014 International Conference on Learning Representations (ICLR'14)*, Banff, Canada, April 2014. doi:[10.48550/ARXIV.1312.4400](https://doi.org/10.48550/ARXIV.1312.4400)



- [101] J. Liu, J. Xiang, Y. Jin, R. Liu, J. Yan, and L. Wang, "Boost precision agriculture with unmanned aerial vehicle remote sensing and edge intelligence: A survey," *Remote Sensing*, vol. 13, no. 21, October 2021. doi:[10.3390/rs13214387](https://doi.org/10.3390/rs13214387)
- [102] L. Liu, B. Song, S. Zhang, and X. Liu, "A novel principal component analysis method for the reconstruction of leaf reflectance spectra and retrieval of leaf biochemical contents," *Remote Sensing*, vol. 9, p. 1113, October 2017. doi:[10.3390/rs9111113](https://doi.org/10.3390/rs9111113)
- [103] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 3071–3085, December 2019. doi:[10.1109/TPAMI.2018.2868685](https://doi.org/10.1109/TPAMI.2018.2868685)
- [104] M. López-Álvarez, J. Hernández-Andrés, E. M. Valero, and J. Romero, "Selecting algorithms, sensors, and linear bases for optimum spectral recovery of skylight," *Journal of the Optical Society of America. A*, vol. 24, no. 4, pp. 942–956, 2007. doi:[10.1364/JOSAA.24.000942](https://doi.org/10.1364/JOSAA.24.000942)
- [105] O. Losson and L. Macaire, "CFA local binary patterns for fast illuminant-invariant color texture classification," *Journal of Real-Time Image Processing*, vol. 10, pp. 387–401, June 2012. doi:[10.1007/s11554-012-0302-5](https://doi.org/10.1007/s11554-012-0302-5)
- [106] P. Lottes and C. Stachniss, "Semi-supervised online visual crop and weed classification in precision farming exploiting plant arrangement," in *Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, September 2017, pp. 5155–5161. doi:[10.1109/IROS.2017.8206403](https://doi.org/10.1109/IROS.2017.8206403)
- [107] P. Lottes, M. Hoferlin, S. Sander, M. Müter, P. S. Lammers, and C. Stachniss, "An effective classification system for separating sugar beets and weeds for precision farming applications," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'16)*, Stockholm, Sweden, May 2016, pp. 5157–5163. doi:[10.1109/ICRA.2016.7487720](https://doi.org/10.1109/ICRA.2016.7487720)
- [108] P. Lottes, M. Hörferlin, S. Sander, and C. Stachniss, "Effective vision-based classification for separating sugar beets and weeds for precision farming,"

- Journal of Field Robotics*, vol. 34, no. 6, pp. 1160–1178, September 2016. doi:[10.1002/rob.21675](https://doi.org/10.1002/rob.21675)
- [109] P. Lottes, R. Khanna, P. Johannes, R. Siegwart, and C. Stachniss, “UAV-based crop and weed classification for smart farming,” in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA 2017)*, Singapore, May 2017, pp. 3024–3031. doi:[10.1109/ICRA.2017.7989347](https://doi.org/10.1109/ICRA.2017.7989347)
- [110] P. Lottes, J. Behley, A. Milioto, and C. Stachniss, “Fully convolutional networks with sequential information for robust crop and weed detection in precision farming,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2870–2877, October 2018. doi:[10.1109/LRA.2018.2846289](https://doi.org/10.1109/LRA.2018.2846289)
- [111] M. Louargant, G. Jones, R. Faroux, J.-N. Paoli, T. Maillot, C. Gée, and S. Vilette, “Unsupervised classification algorithm for early weed detection in row-crops by combining spatial and spectral information,” *Remote Sensing*, vol. 10, p. 761, May 2018. doi:[10.3390/rs10050761](https://doi.org/10.3390/rs10050761)
- [112] Y. Lu and S. Young, “A survey of public datasets for computer vision tasks in precision agriculture,” *Computers and Electronics in Agriculture*, vol. 178, p. 105760, 2020. doi:[10.1016/j.compag.2020.105760](https://doi.org/10.1016/j.compag.2020.105760)
- [113] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS’17)*, vol. 30, Long Beach, CA, USA, December 2017, pp. 4765–4774. [Online]. Available: <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>
- [114] A. Mahlein, M. Kusk, S. Thomas, D. Bohnenkamp, E. Alisaac, J. Behmann, M. Wahabzada, and K. Kersting, “Plant disease detection by hyperspectral imaging: from the lab to the field,” *Advances in Animal Biosciences*, vol. 8, no. 2, pp. 238–243, 2017. doi:[10.1017/S2040470017001248](https://doi.org/10.1017/S2040470017001248)
- [115] L. T. Maloney and B. A. Wandell, “Color constancy: a method for recovering surface spectral reflectance,” *Journal of the Optical Society of America A*, vol. 3, no. 1, pp. 29–33, January 1986. doi:[10.1364/JOSAA.3.000029](https://doi.org/10.1364/JOSAA.3.000029)

- [116] A. Mansouri, F. Marzani, and P. Gouton, "Neural networks in two cascade algorithms for spectral reflectance reconstruction," in *IEEE International Conference on Image Processing (ICIP'05)*, Genova, Italy, October 2005, pp. II-718. doi:[10.1109/ICIP.2005.1530156](https://doi.org/10.1109/ICIP.2005.1530156)
- [117] A. Martínez-Usó, F. Pla, J. M. Sotoca, and P. García-Sevilla, "Clustering-based hyperspectral band selection using information measures," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 12, pp. 4158–4171, December 2007. doi:[10.1109/TGRS.2007.904951](https://doi.org/10.1109/TGRS.2007.904951)
- [118] B. Mazin, J. Delon, and Y. Gousseau, "Estimation of illuminants from projections on the planckian locus," *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1944–1955, June 2015. doi:[10.1109/TIP.2015.2405414](https://doi.org/10.1109/TIP.2015.2405414)
- [119] C. McCamy, H. Marcus, and J. Davidson, "A color-rendition chart," *Journal of Applied Photographic Engineering*, vol. 2, no. 3, pp. 95–99, June 1976.
- [120] U. Meier, *Growth stages of mono-and dicotyledonous plants. BBCH monograph*. Blackwell Wissenschafts-Verlag, 1997.
- [121] S. Mihoubi, "Snapshot multispectral image demosaicing and classification," PhD thesis, University of Lille, November 2018.
- [122] S. Mihoubi, O. Losson, B. Mathon, and L. Macaire, "Multispectral demosaicing using pseudo-panchromatic image," *IEEE Transactions on Computational Imaging*, vol. 3, no. 4, pp. 982–995, December 2017. doi:[10.1109/TCI.2017.2691553](https://doi.org/10.1109/TCI.2017.2691553)
- [123] S. Mihoubi, O. Losson, B. Mathon, and L. Macaire, "Spatio-spectral binary patterns based on multispectral filter arrays for texture classification," *Journal of the Optical Society of America A*, vol. 35, no. 9, pp. 1532–1542, September 2018. doi:[10.1364/JOSAA.35.001532](https://doi.org/10.1364/JOSAA.35.001532)
- [124] A. Milioto, P. Lottes, and C. Stachniss, "Real-time blob-wise sugar beets vs. weeds classification for monitoring fields using convolutional neural networks," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-2/W3, pp. 41–48, August 2017. doi:[10.5194/isprs-annals-IV-2-W3-41-2017](https://doi.org/10.5194/isprs-annals-IV-2-W3-41-2017)

- [125] A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs," in *Proceedings of 2018 IEEE International Conference on Robotics and Automation (ICRA'18)*, Brisbane, QLD, Australia, May 2018, pp. 2229–2235. doi:[10.1109/ICRA.2018.8460962](https://doi.org/10.1109/ICRA.2018.8460962)
- [126] J. Minet, "Imagerie multispectrale, vers une conception adaptée à la détection de cibles," PhD thesis, Université Paris Sud - Paris XI, December 2011. [Online]. Available: <https://pastel.archives-ouvertes.fr/pastel-00714207>
- [127] J. Mizutani, S. Ogawa, K. Shinoda, M. Hasegawa, and S. Kato, "Multispectral demosaicking algorithm based on inter-channel correlation," in *Proceedings of the 2014 IEEE Visual Communications and Image Processing Conference (VCIP'14)*, Valletta, Malta, December 2014, pp. 474–477. doi:[10.1109/VCIP.2014.7051609](https://doi.org/10.1109/VCIP.2014.7051609)
- [128] Y. Monno, H. Teranaka, K. Yoshizaki, M. Tanaka, and M. Okutomi, "Single-sensor RGB-NIR imaging: High-quality system design and prototype implementation," *IEEE Sensors Journal*, vol. 19, pp. 497–507, January 2019. doi:[10.1109/JSEN.2018.2876774](https://doi.org/10.1109/JSEN.2018.2876774)
- [129] M. S. Moran, Y. Inoue, and E. M. Barnes, "Opportunities and limitations for image-based remote sensing in precision crop management," *Remote Sensing of Environment*, vol. 61, no. 3, pp. 319–346, 1997. doi:[10.1016/S0034-4257\(97\)00045-X](https://doi.org/10.1016/S0034-4257(97)00045-X)
- [130] D. J. Mulla, "Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps," *Biosystems Engineering*, vol. 114, pp. 358–371, April 2013. doi:[10.1016/j.biosystemseng.2012.08.009](https://doi.org/10.1016/j.biosystemseng.2012.08.009)
- [131] Y. Murakami, T. Obi, M. Yamaguchi, N. Ohyama, and Y. Komiya, "Spectral reflectance estimation from multi-band image using color chart," *Optics Communications*, vol. 188, pp. 47–54, February 2001. doi:[10.1016/S0030-4018\(00\)01131-7](https://doi.org/10.1016/S0030-4018(00)01131-7)
- [132] R. Oberti, M. Marchi, P. Tirelli, A. Calcante, M. Iriti, E. Tona, M. Hočevár, J. Baur, J. Pfaff, C. Schütz, and H. Ulbrich, "Selective spraying of grapevines

- for disease control using a modular agricultural robot," *Biosystems Engineering*, vol. 146, pp. 203–215, 2016. doi:[10.1016/j.biosystemseng.2015.12.004](https://doi.org/10.1016/j.biosystemseng.2015.12.004)
- [133] OCDE/FAO, *OECD-FAO Agricultural Outlook 2012-2021*, 2012. doi:[10.1787/19991142](https://doi.org/10.1787/19991142)
- [134] M. A. Oliver, T. F. A. Bishop, and B. P. Marchant, *Precision Agriculture for Sustainability and Environmental Protection*, 1st ed. Routledge, 2013. doi:[10.4324/9780203128329](https://doi.org/10.4324/9780203128329)
- [135] S. Ortega, H. Fabelo, D. Iakovidis, A. Koulaouzidis, and G. M. Callico, "Use of hyperspectral/multispectral imaging in gastroenterology. Shedding some-different-light into the dark," *Journal of Clinical Medicine*, vol. 8, p. 36, January 2019. doi:[10.3390/jcm8010036](https://doi.org/10.3390/jcm8010036)
- [136] J.-I. Park, M.-H. Lee, M. D. Grossberg, and S. K. Nayar, "Multispectral imaging using multiplexed illumination," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV'07)*, Rio de Janeiro, Brazil, October 2007, pp. 1–8. doi:[10.1109/ICCV.2007.4409090](https://doi.org/10.1109/ICCV.2007.4409090)
- [137] J. P. S. Parkkinen, J. Hallikainen, and T. Jaaskelainen, "Characteristic spectra of Munsell colors," *Journal of the Optical Society of America A*, vol. 6, no. 2, pp. 318–322, 1989. doi:[10.1364/JOSAA.6.000318](https://doi.org/10.1364/JOSAA.6.000318)
- [138] M. Parmar, S. Lansel, and J. Farrell, "An LED-based lighting system for acquiring multispectral scenes," in *Proceedings of the IS&T/SPIE Electronic Imaging Conference: Digital Photography VIII*, vol. 8299, Burlingame, CA, USA, January 2012, pp. 211–218. doi:[10.1117/12.912513](https://doi.org/10.1117/12.912513)
- [139] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005. doi:[10.1109/TPAMI.2005.159](https://doi.org/10.1109/TPAMI.2005.159)
- [140] A. J. Peters, B. C. Reed, M. D. Eve, and K. C. McDaniel, "Remote sensing of broom snakeweed (*Gutierrezia sarothrae*) with NOAA-10 spectral image processing," *Weed Technology*, vol. 6, no. 4, pp. 1015–1020, December 1992. doi:[10.1017/S0890037X00036642](https://doi.org/10.1017/S0890037X00036642)

- [141] J. Pichette, W. Charle, and A. Lambrechts, "Fast and compact internal scanning CMOS-based hyperspectral camera: the Snapscan," in *Proceedings of the SPIE Electronic Imaging Annual Symposium: Photonic Instrumentation Engineering IV*, vol. 10110, San Francisco, CA, USA, 2017, pp. 1–10. doi:[10.1117/12.2253614](https://doi.org/10.1117/12.2253614)
- [142] F. J. Pierce and P. Nowak, "Aspects of precision agriculture," in *Advances in Agronomy*, ser. Advances in Agronomy. Academic Press, 1999, vol. 67, pp. 1–85. doi:[10.1016/S0065-2113\(08\)60513-1](https://doi.org/10.1016/S0065-2113(08)60513-1)
- [143] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen, *Computer Vision Using Local Binary Patterns*. Springer London, 2011, vol. 40. doi:[10.1007/978-0-85729-748-8\\_14](https://doi.org/10.1007/978-0-85729-748-8_14)
- [144] C. Potena, D. Nardi, and A. Pretto, "Fast and accurate crop and weed identification with summarized train sets for precision agriculture," in *Proceedings of the 14th International Conference on Intelligent Autonomous Systems*, vol. 531, Shanghai, China, July 2016, pp. 105–121. doi:[10.1007/978-3-319-48036-7\\_9](https://doi.org/10.1007/978-3-319-48036-7_9)
- [145] A. Pretto, S. Aravecchia, W. Burgard, N. Chebrolu, C. Dornhege, T. Falck, F. Fleckenstein, A. Fontenla, M. Imperoli, R. Khanna, F. Liebisch, P. Lottes, Andres, D. Nardi, S. Nardi, J. Pfeifer, M. Popović, C. Potena, C. Pradalier, E. Rothacker-Feder, I. Sa, A. Schaefer, R. Siegwart, C. Stachniss, A. Walter, W. Winterhalter, X. Wu, and J. Nieto, "Building an aerial-ground robotics system for precision farming: An adaptable solution," *IEEE Robotics and Automation Magazine*, vol. 28, no. 3, pp. 29–49, September 2020. doi:[10.1109/MRA.2020.3012492](https://doi.org/10.1109/MRA.2020.3012492)
- [146] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features," in *Proceedings of the 32nd Conference on Neural Information Processing Systems (NIPS'18)*, vol. 31, Montréal, Canada, December 2018, pp. 6638–6648. [Online]. Available: <https://papers.nips.cc/paper/2018/file/14491b756b3a51daac41c24863285549-Paper.pdf>
- [147] V. Rathi and P. Goyal, "Convolution filter based efficient multispectral image demosaicking for compact MSFAs," in *Proceedings of the 16th International Joint*

- Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISIGRAPP'21, Volume 4: VISAPP*, Virtual event, February 2021, pp. 112–121. doi:[10.5220/0010249601120121](https://doi.org/10.5220/0010249601120121)
- [148] S. Ratnasingam and J. Hernández-Andrés, “Illuminant spectrum estimation at a pixel,” *Journal of the Optical Society of America A*, vol. 28, no. 4, p. 696, April 2011. doi:[10.1364/JOSAA.28.000696](https://doi.org/10.1364/JOSAA.28.000696)
- [149] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: An astounding baseline for recognition,” in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'14)*, Columbus, OH, USA, June 2014, pp. 512–519. doi:[10.1109/CVPRW.2014.131](https://doi.org/10.1109/CVPRW.2014.131)
- [150] P. Ricci, M. Barzman, F. Bigler, P. M. Boonekamp, I. Denholm, B. Hommel, J. Kiss, P. Kudsk, A. Messean, S. Jean Louis, M. Sattin, and V. Troillard, “Integrated pest management in Europe,” Endure Network, Tech. Rep., November 2010.
- [151] B. R. Ritz, A. D. Manthripragada, S. Costello, S. J. Lincoln, M. J. Farrer, M. Cockburn, and J. Bronstein, “Dopamine transporter genetic variants and pesticides in Parkinson’s disease,” *Environmental Health Perspectives*, vol. 117, no. 6, pp. 964–969, 2009. doi:[10.1289/ehp.0800277](https://doi.org/10.1289/ehp.0800277)
- [152] J. G. Robertson, “Quantifying resolving power in astronomical spectra,” *Publications of the Astronomical Society of Australia*, vol. 30, p. e048, September 2013. doi:[10.1017/pasa.2013.26](https://doi.org/10.1017/pasa.2013.26)
- [153] I. Sa, Z. Chen, M. Popović, R. Khanna, F. Liebisch, J. I. Nieto, and R. Siegwart, “weedNet: Dense semantic weed classification using multispectral images and MAV for smart farming,” *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 588–595, January 2018. doi:[10.1109/LRA.2017.2774979](https://doi.org/10.1109/LRA.2017.2774979). [Online]. Available: <https://github.com/inkyusa/weedNet>
- [154] I. Sa, M. Popović, R. Khanna, Z. Chen, P. Lottes, F. Liebisch, J. Nieto, C. Stachniss, A. Walter, and R. Siegwart, “WeedMap: A large-scale semantic weed mapping framework using aerial multispectral imaging and deep



- neural network for precision farming," *Remote Sensing*, vol. 10, no. 9, 2018. doi:10.3390/rs10091423. [Online]. Available: <https://projects.asl.ethz.ch/datasets/doku.php?id=weedmap:remotesensing2018weedmap>
- [155] M. J. Saberian, H. Masnadi-Shirazi, and N. Vasconcelos, "TaylorBoost: First and second-order boosting algorithms with explicit margin control," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11)*, Colorado Springs, CO, USA, June 2011, pp. 2929–2934. doi:10.1109/CVPR.2011.5995605
- [156] R. N. Sahoo, S. S. Ray, and K. Manjunath, "Hyperspectral remote sensing of agriculture," *Current Science*, vol. 108, pp. 848–859, March 2015. [Online]. Available: <https://ncfc.gov.in/publications/p2.pdf>
- [157] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'18)*, Salt Lake City, UT, USA, June 2018, pp. 3723–3732. doi:10.1109/CVPR.2018.00392
- [158] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, July 1964. doi:10.1021/ac60214a047
- [159] V. Sharma, A. Diba, T. Tuytelaars, and L. Van Gool, "Hyperspectral CNN for image classification & band selection, with application to face recognition," *Technical report KUL/ESAT/PSI/1604*, KU Leuven, ESAT, Leuven, Belgium, December 2016. [Online]. Available: <https://www.scinapse.io/papers/2593119498>
- [160] H.-L. Shen, P.-Q. Cai, S.-J. Shao, and J. H. Xin, "Reflectance reconstruction for multispectral imaging by adaptive Wiener estimation," *Optics Express*, vol. 15, no. 23, pp. 15 545–15 554, November 2007. doi:10.1364/OE.15.015545
- [161] H.-L. Shen, H.-J. Wan, and Z.-C. Zhang, "Estimating reflectance from multispectral camera responses based on partial least-squares regression," *Journal of Electronic Imaging*, vol. 19, p. 3, April 2010. doi:10.1117/1.3385782



- [162] N. Shetty, M. H. Olesen, R. Gislum, L. C. Deleuran, and B. Boelt, "Use of partial least squares discriminant analysis on visible-near infrared multispectral image data to examine germination ability and germ length in spinach seeds," *Journal of Chemometrics*, vol. 26, no. 8-9, pp. 462–466, March 2012. doi:[10.1002/cem.1415](https://doi.org/10.1002/cem.1415)
- [163] W. Shi, C. C. Loy, and X. Tang, "Deep specialized network for illuminant estimation," in *Proceedings of the 2016 European Conference on Computer Vision (ECCV'16)*, vol. 9908, Amsterdam, The Netherlands, October 2016, pp. 371–387. doi:[10.1007/978-3-319-46493-0\\_23](https://doi.org/10.1007/978-3-319-46493-0_23)
- [164] N. Shimano, K. Terai, and M. Hironaga, "Recovery of spectral reflectances of objects being imaged by multispectral cameras," *Journal of the Optical Society of America A*, vol. 24, no. 10, pp. 3211–3219, October 2007. doi:[10.1364/JOSAA.24.003211](https://doi.org/10.1364/JOSAA.24.003211)
- [165] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR'15)*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, May 2015. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [166] P. Stigell, K. Miyata, and M. Hauta-Kasari, "Wiener estimation method in estimating of spectral reflectance from RGB images," *Pattern Recognition and Image Analysis*, vol. 17, no. 2, pp. 233–242, June 2007. doi:[10.1134/S1054661807020101](https://doi.org/10.1134/S1054661807020101)
- [167] B. Sun and K. Saenko, "Deep CORAL: Correlation alignment for deep domain adaptation," in *Proceedings of the 2016 European Conference on Computer Vision (ECCV'16)*, Amsterdam, The Netherlands, October 2016, pp. 443–450. doi:[10.1007/978-3-319-49409-8\\_35](https://doi.org/10.1007/978-3-319-49409-8_35)
- [168] Y. Suzuki, H. Okamoto, and T. Kataoka, "Image segmentation between crop and weed using hyperspectral imaging for weed detection in soybean field," *Environmental Control in Biology*, vol. 46, no. 3, pp. 163–173, 2008. doi:[10.2525/ecb.46.163](https://doi.org/10.2525/ecb.46.163)

- [169] H. T. Sogaard and I. Lund, "Application accuracy of a machine vision-controlled robotic micro-dosing system," *Biosystems Engineering*, vol. 96, no. 3, pp. 315–322, 2007. doi:[10.1016/j.biosystemseng.2006.11.009](https://doi.org/10.1016/j.biosystemseng.2006.11.009)
- [170] H. Tang, Y. Li, X. Han, Q. Huang, , and W. Xie, "A spatial-spectral prototypical network for hyperspectral remote sensing image," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 1, pp. 167–171, May 2019. doi:[10.1109/LGRS.2019.2916083](https://doi.org/10.1109/LGRS.2019.2916083)
- [171] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," *AI Communications*, vol. 30, no. 2, pp. 169–190, May 2017. doi:[10.3233/AIC-170729](https://doi.org/10.3233/AIC-170729)
- [172] P. S. Thenkabail, R. B. Smith, and E. De Pauw, "Evaluation of narrowband and broadband vegetation indices for determining optimal hyperspectral wavebands for agricultural crop characterization," *Photogrammetric Engineering & Remote Sensing*, vol. 68, no. 6, pp. 607–621, June 2002. [Online]. Available: <http://www.gisresources.com/wp-content/uploads/2014/03/Evaluation-of-Narrowband-and-Broadband-Vegetation-Indices.pdf>
- [173] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*. Academic Press, October 2008.
- [174] C. Thiour-Mauprivez, F. Martin-Laurent, C. Calvayrac, and L. Barthelmebs, "Effects of herbicide on non-target microorganisms: Towards a new class of biomarkers?" *Science of The Total Environment*, vol. 684, pp. 314–325, 2019. doi:[10.1016/j.scitotenv.2019.05.230](https://doi.org/10.1016/j.scitotenv.2019.05.230)
- [175] J.-B. Thomas, P.-J. Lapray, P. Gouton, and C. Clerc, "Spectral characterization of a prototype SFA camera for joint visible and NIR acquisition," *Sensors*, vol. 16, no. 7, June 2016. doi:[10.3390/s16070993](https://doi.org/10.3390/s16070993)
- [176] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation—A review," *Information Processing in Agriculture*, vol. 7, no. 1, pp. 1–19, March 2020. doi:[10.1016/j.inpa.2019.09.006](https://doi.org/10.1016/j.inpa.2019.09.006)

- [177] K. Uto, H. Seki, G. Saito, and Y. Kosugi, "Characterization of rice paddies by a UAV-mounted miniature hyperspectral sensor system," *IEEE J-STARS*, vol. 6, no. 2, pp. 851–860, April 2013. doi:[10.1109/JSTARS.2013.2250921](https://doi.org/10.1109/JSTARS.2013.2250921)
- [178] K. Uto, H. Seki, G. Saito, Y. Kosugi, and T. Komatsu, "Development of a low-cost hyperspectral whiskbroom imager using an optical fiber bundle, a swing mirror, and compact spectrometers," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 9, pp. 3909–3925, September 2016. doi:[10.1109/JSTARS.2016.2592987](https://doi.org/10.1109/JSTARS.2016.2592987)
- [179] T. Utstumo, F. Urdal, A. Brevik, J. Dørum, J. Netland, Ø. Overskeid, T. Berge, and J. T. Gravdahl, "Robotic in-row weed control in vegetables," *Computers and Electronics in Agriculture*, vol. 154, pp. 36–45, Nov. 2018. doi:[10.1016/j.compag.2018.08.043](https://doi.org/10.1016/j.compag.2018.08.043)
- [180] J. van de Weijer and T. Gevers, "Color constancy based on the Grey-edge hypothesis," in *IEEE International Conference on Image Processing 2005*, Genova, Italy, October 2005, pp. II–722. doi:[10.1109/ICIP.2005.1530157](https://doi.org/10.1109/ICIP.2005.1530157)
- [181] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008. [Online]. Available: <http://jmlr.org/papers/v9/vandermaaten08a.html>
- [182] M. Weisa and R. Gerhards, "Feature extraction for the identification of weed species in digital images for the purpose of site-specific weed control," in *Proceedings of the 6th European Conference on Precision Agriculture (ECPA'07)*, Skiathos, Greece, June 2007, pp. 537–544.
- [183] A. Wendel and J. Underwood, "Illumination compensation in ground based hyperspectral imaging," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 129, pp. 162–178, July 2017. doi:[10.1016/j.isprsjprs.2017.04.010](https://doi.org/10.1016/j.isprsjprs.2017.04.010)
- [184] A. Wendel and J. Underwood, "Self-supervised weed detection in vegetable crops using ground based hyperspectral imaging," in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA'16)*, Stockholm, Sweden, May 2016, pp. 5128–5135. doi:[10.1109/ICRA.2016.7487717](https://doi.org/10.1109/ICRA.2016.7487717)

- [185] Z. Wu, Y. Chen, B. Zhao, X. Kang, and Y. Ding, "Review of weed detection methods based on computer vision," *Sensors*, vol. 21, no. 11, p. 3647, May 2021. doi:[10.3390/s21113647](https://doi.org/10.3390/s21113647)
- [186] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, Seattle, WA, USA, June 2020, pp. 10 687–10 698. doi:[10.1109/CVPR42600.2020.01070](https://doi.org/10.1109/CVPR42600.2020.01070)
- [187] P. Xu, H. Xu, C. Diao, and Z. Ye, "Self-training-based spectral image reconstruction for art paintings with multispectral imaging," *Applied Optics*, vol. 56, no. 30, p. 8461, October 2017. doi:[10.1364/AO.56.008461](https://doi.org/10.1364/AO.56.008461)
- [188] S. C. Yoon, B. Park, K. C. Lawrence, W. R. Windham, and G. W. Heitschmidt, "Line-scan hyperspectral imaging system for real-time inspection of poultry carcasses with fecal material and ingesta," *Computers and Electronics in Agriculture*, vol. 79, no. 2, pp. 159–168, September 2011. doi:[10.1016/j.compag.2011.09.008](https://doi.org/10.1016/j.compag.2011.09.008)
- [189] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of the 12th International Conference on Machine Learning (ICML'03)*, vol. 2, Washington, DC, USA, August 2003, pp. 856–863. [Online]. Available: <https://www.aaai.org/Papers/ICML/2003/ICML03-111.pdf>
- [190] W. Yu, "Practical anti-vignetting methods for digital cameras," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 4, pp. 975–983, November 2004. doi:[10.1109/TCE.2004.1362487](https://doi.org/10.1109/TCE.2004.1362487)
- [191] C. Zeng, D. J. King, M. Richardson, and B. Shan, "Fusion of multispectral imagery and spectrometer data in UAV remote sensing," *Remote Sensing*, vol. 9, p. 696, July 2017. doi:[10.3390/rs9070696](https://doi.org/10.3390/rs9070696)
- [192] Y. Zhang, C. P. Huynh, N. Habili, and K. N. Ngan, "Material segmentation in hyperspectral images with a spatio-spectral texture descriptor," in *Proceedings of the IEEE International Conference on Acoustics, Speech and*

- Signal Processing (ICASSP'19)*, Brighton, UK, May 2019, pp. 2397–2401. doi:[10.1109/ICASSP.2019.8683637](https://doi.org/10.1109/ICASSP.2019.8683637)
- [193] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, “Learning deep features for discriminative localization,” in *Proceedings of the 2016 Conference on Computer Vision and Pattern Recognition (CVPR'16)*, vol. 6661, Las Vegas, NV, USA, June 2016, pp. 2921–2929. doi:[10.1109/CVPR.2016.319](https://doi.org/10.1109/CVPR.2016.319)
- [194] W. Zhou, S. Gao, L. Zhang, and X. Lou, “Histogram of oriented gradients feature extraction from raw Bayer pattern images,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 5, pp. 946–950, March 2020. doi:[10.1109/TCSII.2020.2980557](https://doi.org/10.1109/TCSII.2020.2980557)
- [195] M. Zude-Sasse, S. Fountas, T. A. Gemtos, and N. Abu-Khalaf, “Applications of precision agriculture in horticultural crops,” *European Journal of Horticultural Science*, vol. 81, pp. 78–90, April 2016. doi:[10.17660/eJHS.2016/81.2.2](https://doi.org/10.17660/eJHS.2016/81.2.2)