DESIGN AND OPTIMIZATION OF A COMPRESSIVE SPECTRAL VIDEO SENSING SYSTEM

KARETH MARCELA LEÓN LÓPEZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2022

DESIGN AND OPTIMIZATION OF A COMPRESSIVE SPECTRAL VIDEO SENSING SYSTEM

KARETH MARCELA LEÓN LÓPEZ

Trabajo de Grado para optar al título de
Doctora en Ciencias de la Computación

Director
Ph.D. HENRY ARGUELLO FUENTES

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2022

AGRADECIMIENTOS

Agradezco a mi familia por apoyarme tanto durante el desarrollo de mi carrera.

Agradezco especialmente a mi madre, Maria Eugenia, que ha sido mi apoyo incondicional durante todo este tiempo. Gracias por tanto cariño y comprensión durante los momentos más difíciles.

Agradezco a Edson por su amor, fuerza y compañía durante estos últimos años.

También quiero agradecer a mis colegas y compañeros del grupo HDSP por enseñarme tanto. Especialmente, agradezco a mi director, Henry, que prestó todo su apoyo para hacerme una doctora colombiana, y a Laura, que ha sido un gran apoyo y gran referente profesional y personal durante mi carrera. Estaré infinitamente agradecida con ellos por darme la oportunidad ser una mejor versión de mi cada día.
I thank to Jean-Yves and Corinne for receiving me in their lab, TéSA, and to allow me to meet and work with them. Je suis très ravie de vous rencontrer.

Agradezco a Dios por poner tantas bellas personas en mi camino.

Finalmente, me agradezco a mi, por no rendirme en cada larga jornada, en cada caída, y por continuar hasta llegar a la meta.

RESUMEN

TÍTULO: DISEÑO Y OPTIMIZACIÓN DE UN SISTEMA COMPRESIVO PARA LA ADQUISICIÓN DE VIDEO ESPECTRAL [*]

AUTOR: Kareth Marcela León López [**]

PALABRAS CLAVE: Muestreo compresivo, video espectral, aperturas codificadas, diseño de sistema de adquisición.

DESCRIPCIÓN: Los videos espectrales contienen información espacial y espectral de una escena en el tiempo, implicando un conjunto de cubos de datos tridimensionales. Los sistemas de adquisición de video espectral compresivo (CSVS) adquieren de manera comprimida los videos mediante la codificación y proyección de cada cuadro espectral en un sensor bidimensional, resultando en un conjunto de cuadros espectrales comprimidos. El video es reconstruido a partir de estas medidas comprimidas usando un algoritmo de recuperación, asumiendo que la señal tiene una representación escasa en una base de transformación. La calidad del video espectral reconstruido depende de la base de transformación, la apertura codificada (CA) usada en el sistema CSVS y el método de reconstrucción. Hasta la fecha, se han realizado diferentes esfuerzos para incrementar la calidad de reconstrucción de estos videos tal como agregar una cámara extra para adquirir información adicional. Sin embargo, éstas soluciones son costosas o ineficientes en aplicaciones prácticas. Según la literatura, es posible obtener un alto rendimiento diseñando conjuntamente la base, la CA y el procedimiento de recuperación. Sin embargo, hasta donde se tiene conocimiento, no existe trabajos previos sobre el diseño conjunto de éstas etapas en sistemas CSVS, donde la información espectral es valiosa. Esta tesis estudia diferentes estrategias para diseñar y optimizar un sistema CSVS para mejorar la calidad de los cuadros espectrales reconstruidos. Una primera estrategia implica el diseño conjunto de la base de transformación y del método de recuperación usando una representación tensorial de orden superior. Y una segunda estrategia implica la optimización del sistema usando redes neuronales convolucionales, aprovechando la creciente cantidad de datos disponibles en la comunidad científica. Los experimentos numéricos sobre diferentes bases de datos a partir de

---

[*] Tesis doctoral

[**] Escuela de ingeniería de sistemas e informática. Director: Ph.D. Henry Arguello Fuentes.

las metodologías propuestas muestran calidades de reconstrucción superiores en comparación con técnicas de la literatura.

ABSTRACT

TITLE: DESIGN AND OPTIMIZATION OF A COMPRESSIVE SPECTRAL VIDEO SENSING SYSTEM [*]

AUTHOR: Kareth Marcela León López [**]

KEYWORDS: Compressive sensing, spectral video, coded aperture, acquisition system design.

DESCRIPTION: Spectral videos contain spatial and spectral information of a scene across time, entailing a set of three-dimensional data cubes. Compressive spectral video sensing (CSVS) systems compressively acquire 4D spectral videos by encoding and projecting each spectral frame onto a two-dimensional sensor, resulting in a set of compressed spectral frames. Then, a recovery algorithm is employed to obtain the spectral video from the compressed measurements, where it is assumed that the signal is sparse on some transformation basis. Particularly, the quality of the recovered spectral video mainly depends on the representation basis, the coded aperture (CA) used in the CSVS system and the applied method for the recovering. Up to date, different efforts have been made for increasing the reconstruction quality of the videos such as adding an extra camera to acquire side information. However, these solutions are costly or inefficient for practical applications. According to the literature, state-of-the-art performance can be obtained by jointly designing the basis, the CA and the recovery procedure. Nonetheless, up to our knowledge, there is no prior work concerning the joint design of these stages in CSVS systems, where the spectral information is valuable. This dissertation studies different strategies for designing and optimizing a CSVS system to obtain an image quality improvement on the reconstructed spectral frames. A first strategy entails the jointly sparse basis representation and recovery method design based on a higher-order tensor representation. And a second strategy involves the optimization of the CSVS system based on convolutional neural networks taking advantage of the increasing amount of data available in the scientific community. Extensive numerical simulations on different datasets evaluate the performance of the reconstructed videos from the proposed methodologies showing superior accuracy scores against state-of-the-art techniques.

---

[*] Doctoral Thesis

[**] Department of Systems Engineering and Informatics. Director: Ph.D. Henry Arguello Fuentes.

Contents

List of Figures

pág.

List of Tables

pág.

Table 13 Confusion matrix for the SVM-RBF classifier for realization # 10 (Pc: 0.83, OA: 70.1%, kappa: 0.57) 168

LIST OF ANNEXES

pág.

EXTENDED ABSTRACT

Spectral videos contain spatial and spectral information of a scene across time, entailing a set of three-dimensional data cubes. Such four-dimensional (4D) spectral videos have gained relevance in applications such as disease detection on crops, surveillance, early cancer detection, among others. Compressive spectral video sensing (CSVS) systems compressively acquire 4D spectral videos by encoding and projecting each spectral frame onto a two-dimensional sensor, resulting in a set of compressed spectral frames. Then, a recovery algorithm is employed to obtain the spectral video from the compressed measurements, assuming that the signal is sparse on some transformation basis. Particularly, the quality of the recovered spectral video mainly depends on the representation basis, the coded aperture (CA) used in the CSVS system and the applied method for the recovery. Up to date, different efforts have been made for increasing the reconstruction quality of the videos such as adding an extra camera to acquire side information or increasing the number of shots per frame to better conditioning the system, yielding a less ill-posed inverse problem. However, these solutions are costly or inefficient for practical applications. Recent works have exploited deep learning methods for recovering spectral videos, nonetheless, the CA is set to random structure patterns. According to the literature, state-of-the-art performance can be obtained by jointly designing the basis, the CA and the recovery procedure. Nonetheless, as far as we know, there is no prior work concerning the joint design of these stages in CSVS systems, where the spectral information is valuable.

This dissertation aims to address different strategies for designing and optimizing a CSVS system to obtain an image quality improvement on the reconstructed spectral frames. A first strategy entails the jointly sparse basis representation and recovery method design based on a higher-order tensor representation. The higher-order

structure in 4D spectral videos is crucial for exploiting the inherent redundancy of the information. Given that conventional CS-based-acquisition modeling of optical systems and, in turn, the sparse representation models rely on the data representation in vector/matrix form (leading to a high computational burden), the proposed design aims to exploit the video tensor-based representation to simultaneously learn and reconstruct the sparse transform and the spectral video under a CSVS framework. Then, contrary to traditional offline dictionary learning methods, the proposed tensor sparsifying transform is updated online from the data, while being reconstructed.

A second strategy involves the optimization of the CSVS system based on convolutional neural networks taking advantage of the increasing amount of data available in the scientific community. In this scenario, the CA of the CSVS system and the recovery method are jointly designed in an end-to-end (E2E) network, in which the sparse representation is substituted by convolution layers that learn complex features from the data. The E2E cost function is set to find the best CA set where the distance between the original and the recovered video is minimized. Extensive numerical simulations on two spectral video datasets evaluate the accuracy of the reconstructed videos from the proposed methodologies showing superior accuracy scores against state-of-the-art techniques.

Additionally to the main objectives of this dissertation, a practical application of spectral videos (with low temporal resolution) from satellite remote sensing data for crop monitoring in agriculture is presented. To date, detecting anomalies in time series of multi-temporal spectral remote sensing images for crop monitoring is generally performed using a large sample of historical data at a pixel level. Conversely, the proposed framework involves an anomaly detection, localization, and classification methodology that exploits the temporal information contained in a given season at a parcel level to detect and localize outliers using hidden Markov models (HMM).

1. Introduction

Spectral video acquisition (which includes multispectral and hyperspectral videos) involves the sensing of spatial information across several wavelengths at different instants of time [1][2][3][4]. The spectral-spatial information at different time lapses is relevant in practical and scientific applications such as object tracking [5][6], background subtraction [7][8], endoscopy-based early cancer detection [9], and monitoring and char-

---

[1] Kareth León-López, Laura Galvis, and Henry Arguello. "Temporal Colored Coded Aperture Design in Compressive Spectral Video Sensing". In: *IEEE Transactions on Image Processing* 28.1 (2018), pp. 253–264.

[2] Lizhi Wang et al. "High-speed hyperspectral video acquisition by combining Nyquist and compressive sampling". In: *IEEE transactions on pattern analysis and machine intelligence* 41.4 (2019), pp. 857–870.

[3] Xuemei Hu et al. "Multispectral video acquisition using spectral sweep camera". In: *Optics express* 27.19 (2019), pp. 27088–27102.

[4] Kareth M León-López and Henry Arguello Fuentes. "Online Tensor Sparsifying Transform Based on Temporal Superpixels From Compressive Spectral Video Measurements". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 5953–5963.

[5] Fengchao Xiong, Jun Zhou, and Yuntao Qian. "Material based object tracking in hyperspectral videos". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3719–3733.

[6] Lulu Chen et al. "Object Tracking in Hyperspectral-Oriented Video with Fast Spatial-Spectral Features". In: *Remote Sensing* 13.10 (2021), p. 1922.

[7] Yannick Benezeth, Désiré Sidibé, and Jean-Baptiste Thomas. "Background subtraction with multispectral video sequences". In: *IEEE International Conference on Robotics and Automation workshop on Non-classical Cameras, Camera Networks and Omnidirectional Vision (OMNIVIS)*. Hong Kong, China, 2014, p. 6.

[8] Andrews Sobral et al. "Online stochastic tensor decomposition for background subtraction in multispectral video sequences". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2015, pp. 106–113.

[9] Raimund Leitner et al. "Multi-spectral video endoscopy system for the detection of cancerous tissue". In: *Pattern Recognition Letters* 34.1 (2013), pp. 85–93.

acterization of crops behavior [10][11]. The acquisition of these four-dimensional (4D) data (2D spatial, 1D spectral, and 1D temporal dimensions) using spectral imaging sensors is generally expensive and requires high storage and elevated processing load due to the high-dimensionality of the data [53]. On the other hand, spectral video acquisition via compressive spectral imaging, termed compressive spectral video sensing (CSVS), has shown promising results and arises as an alternative for reducing dimensionality, processing and sensor costs [12][13][21][4][15].

For CSVS, snapshot compressive spectral imaging systems have been extended to acquire spectral frames of dynamic scenes by multiplexing the spatio-spectral information [16][13][11][7]. Moreover, even though the temporal information is not multiplexed

---

[10] Qiang Zhang et al. "Missing data reconstruction in remote sensing image with a unified spatial–temporal–spectral deep convolutional neural network". In: *IEEE Transactions on Geoscience and Remote Sensing* 56.8 (2018), pp. 4274–4288.

[11] Kareth M. León-López et al. "Anomaly Detection and Classification in Multispectral Time Series based on Hidden Markov Models". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–11. DOI: 10.1109/TGRS.2021.3101127.

[12] Ashwin A Wagadarikar et al. "Video rate spectral imaging using a coded aperture snapshot spectral imager". In: *Optics express* 17.8 (2009), pp. 6368–6388.

[13] Xun Cao et al. "Computational snapshot multispectral cameras: toward dynamic capture of the spectral world". In: *IEEE Signal Processing Magazine* 33.5 (2016), pp. 95–108.

[14] Samuel Pinilla et al. "Salient Motion Detection for Spectral Video on the Compressive Domain". In: *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE. 2019, pp. 106–110.

[15] Ziyi Meng, Jiawei Ma, and Xin Yuan. "End-to-end low cost compressive spectral imaging with spatial-spectral self-attention". In: *European Conference on Computer Vision*. Springer. 2020, pp. 187–204.

[16] Claudia Correa, Henry Arguello, and Gonzalo Arce. "Spatiotemporal blue noise coded aperture design for multi-shot compressive spectral imaging". In: *JOSA A* 33.12 (2016), pp. 2312–2322.

[17] Kareth León-López et al. "Higher-Order Tensor Sparse Representation for Video-Rate Coded Aperture Snapshot Spectral Image Reconstruction". In: *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE. 2019, pp. 704–

or compressed under the CSVS framework, the temporal correlations joined to the spectral-spatial redundancies are exploited through a sparsifying transform either in the encoding or the decoding steps to yield suitable sensing or reconstruction protocols for high quality frames [12174]. Typically, the encoding step in the CSVS framework encompasses the dispersion and codification of the input scene in the optical path before the light is impinged at the sensor, where the dispersion and codification are usually performed by using elements such as a prism and a coded aperture (CA), respectively. Then, a reconstruction algorithm is employed to estimate a version of the underlying scene from the compressed frames [12161]. Different works in the literature have presented strategies to improve the recovered image quality by either designing the CA [116] or by customizing the recovery algorithm [4218] in a data-independent manner.

Recently, data-driven deep learning approaches have shown state-of-the-art performance in terms of image quality when the CA and the recovery algorithm are jointly designed in video compressing sensing by exploiting tones of data available nowadays [192021]. However, these approaches disregard the spectral information of the dynamic scene, and the multiplexing is conducted across the temporal dimension

---

708.

[18] Crisostomo Alberto Barajas-Solano, Juan-Marcos Ramirez, and Henry Arguello. “Spectral Video Compression Using Convolutional Sparse Coding”. In: *2020 Data Compression Conference (DCC)*. IEEE. 2020, pp. 253–262.

[19] Michael Iliadis, Leonidas Spinoulas, and Aggelos K Katsaggelos. “Deep fully-connected networks for video compressive sensing”. In: *Digital Signal Processing* 72 (2018), pp. 9–18.

[20] Yuqi Li et al. “End-to-end video compressive sensing using anderson-accelerated unrolled networks”. In: *2020 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2020, pp. 1–12.

[21] Mu Qiao et al. “Deep learning for video compressive sensing”. In: *APL Photonics* 5.3 (2020), p. 030801.

aimed at compressing a sequence of video frames into a single 2D measurement. Other works such as [15] have exploited self-attention in deep neural networks for learning an end-to-end approach on snapshot compressive spectral imaging at video rates for spectral images and video reconstruction, however, the CA is fixed and not optimized across time.

1.1. Motivation

The increasing interest in multispectral and hyperspectral image acquisition, mostly used for remote sensing and collected by sensors mounted on satellites, has boosted the development of several spectral imaging systems. Recent progress on snapshot compressive imagers has made possible the acquisition of spectral images at video rate, showing promising approaches for real applications [15][22]. Despite recent advances, these methods disregard the joint design of the encoding element and the recovery procedure by fixing the coded aperture element, leading to the same codification across the different spectral frames. Additionally, a sparse representation transform for such 4D data has not been totally exploited to achieve a better representation and compression.

This dissertation investigates the joint design of the sparsifying transform and the recovery method via higher-order tensors to properly exploit the high-dimensionality of spectral videos. Moreover, considering that the encoding pattern is crucial for having high image quality frames, this dissertation explores coding design strategies tied to the recovery procedure for compressive spectral video sensing, taking advantage of the increasing amount of data available for data-driven approaches.

---

[22] Xin Yuan, David J Brady, and Aggelos K Katsaggelos. “Snapshot compressive imaging: Theory, algorithms, and applications”. In: *IEEE Signal Processing Magazine* 38.2 (2021), pp. 65–88.

1.2. Dissertation Objectives and Organization

The specific objectives of this dissertation are defined as follows:

- To perform a sparsity analysis of spectral videos to determine the multidimensional transformation that better preserves the relevant data.
- To design a high-dimensional-representation-based algorithm for the spectral video recovery from the compressed measurements such that the computation time is reduced and the image quality of the reconstruction is improved.
- To design the system sensing matrix to minimize the number of projections required for the system considering the spatio-temporal correlation of spectral videos.
- To acquire a dataset of spectral videos in a laboratory prototype.
- To verify the performance of the transformation basis, the system sensing matrix and the developed algorithms by numerical experiments using the synthetic and the acquired real spectral videos.

This dissertation is organized as follows: Chapter 2 establishes the basic concepts involved in the dissertation. The mathematical modeling of the state-of-the-art CSVS acquisition systems is presented. The traditional methods for recovering the compressed scene are described.
Chapter 3 presents a sparsity transformation based on tensor decomposition which updates the coefficients online and recovers the compressed spectral video. In compressive spectral video acquisition, tackling dictionary learning is time-consuming since it increases the computational complexity and presents drawbacks for real-time processing, where offline learning is required. Then, this Chapter introduces a

tensor-decomposition learning (TenDL) framework for simultaneous online sparsifying and recovering the spatial-spectral-temporal information of a spectral video performed on several temporal superpixels (TSP-TenDL) for time processing reduction. The framework is composed of two main stages: preprocessing and joint estimation. The preprocessing stage includes a strategy for a grayscale approximation of the video to provide a suitable initialization of the sparsifying basis to be learned. To fully exploit the high signal correlation, a set of temporal superpixels is estimated from the grayscale approximation, reducing the reconstruction time of the large-scale data. Then, the outcome of the first stage is used to estimate the basis and the signal coefficients, where an optimization problem is solved to learn and reconstruct the basis and the signal, respectively, following a block-descent coordinate strategy. The proposed approach is compared from simulations with an offline-learned based method, traditional matrix-based recovery algorithms and the tensor-based recovery, the two latter using a fixed basis, where TSP-TenDL exhibits higher image quality results and lower computation time. The proposed methodology is presented for reconstructing videos from measurements obtained using the video 3D-CASSI system in Chapter 3 and measurements obtained using the video-CASSI system in Chapter 4, where the formulation of the methodology is adapted to the spatio-spectral shifted measurements of the video-CASSI system.

Chapter 5 introduces an end-to-end (E2E) deep learning approach to jointly design the coded aperture and the reconstruction method for improving the reconstruction quality of spectral video frames under the CSVS framework. The proposed formulation takes advantage of state-of-the-art denoising networks to provide a two stage learning for exploiting the spatio-spectral and the spatio-temporal correlations. Simulations on a set of real sequences acquired in the Optics lab of the High Dimensional Signal Processing (HDSP) research group and a set of multispectral videos of the literature show the significant advantages of designing the E2E network, leading to

1dB and 5dB improvements in PSNR compared to deep-learning and traditionally iterative based approaches, respectively. Notice that Chapters 3 and 4 introduce a design methodology for the sparse representation and the recovery method considering only the scene under observation (letting fixed the CA) whereas, Chapter 5 introduces a data-driven framework for jointly designing the recovery and the CA using a collection of databases.

Annex 4 extends the scope of this dissertation to provide a solution for a practical application of spectral video data in agriculture monitoring. Specifically, this Chapter presents a parcel-level anomaly detection and classification framework using features extracted from spectral videos at low-temporal resolution (multitemporal-multispectral images acquired from satellites) via hidden Markov models. The proposed method in this extension Chapter exploits the temporal information contained in remote sensing time series of a given season using hidden Markov models (HMM). The anomaly detection part is based on the learning of HMM parameters associated with unlabeled normal data, that are used in a second step to detect abnormal crop parcels referred to as anomalies. The learned HMM can also be used in time segments to temporally localize the anomalies affecting the crop parcels. The detected and localized anomalies are finally classified using a supervised classifier, e.g., based on support vector machines. The proposed method was studied in french crops.

1.3. Research Contribution

The contents of this dissertation have been published in the following journals and conferences:

Journal Papers:

- **K. León-López**, and H. Arguello, “End-to-End Spatio-Temporal Binary Coded Aperture Design and Recovery in Compressive Spectral Video Sensing”, In

preparation to be submitted in a Journal (2021).

- **K. León-López**, F. Mouret, H. Arguello, and J-Y. Tourneret, “Anomaly detection and classification in multispectral time series based on hidden Markov models”, IEEE Transaction on Geoscience and Remote Sensing, (2021).

- **K. León-López**, and H. Arguello, “Online tensor sparsifying transform based on temporal superpixels from compressive spectral video measurements”, IEEE Transaction on Image Processing, Vol. 29, pp. 5953-5963, Apr. (2020).

- **K. León-López**, L. Galvis, and H. Arguello, “Temporal colored coded aperture design in compressive spectral video sensing”, IEEE Transaction on Image Processing, Vol. 28, No. 1, pp. 253 - 264, Jan. (2019).

Conference Papers:

- **K. León-López**, E. Vargas, F. Rojas Morales, and H. Arguello, “Higher-order tensor sparse representation for video-rate coded aperture snapshot spectral image reconstruction”, IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), Le gosier, Guadeloupe, December 2019.

Other contributions:

Journal Papers:

- O. Villarreal, **K. León-López**, D. Espinosa, W. Agudelo, and H. Arguello. “Seismic source reconstruction in an orthogonal geometry based on local and non-local information in the time slice domain”. Journal of Applied Geophysics, 170, 103846, Nov. (2019).

Conference Papers:

- **K. León-López**, J. M. Ramírez, W. Agudelo and H. Arguello, “Regular Multi-Shot Subsampling and Reconstruction on 3D Orthogonal Symmetric Seismic Grids via Compressive Sensing”, 2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA), Apr. 2019, pp. 1-5.

- J. Monsegny, J. Monsalve, **K. León-López**, M. Becerra, W. Agudelo, and H. Arguello. “Fast marching method in seismic ray tracing on parallel GPU devices”. In Latin American High Performance Computing Conference (CARLA), Bucaramanga, Colombia, September 2018.

- S. Pinilla, **K. León-López**, D. Molina, A. Camacho, A., and H. Arguello. “Subsampling Schemes for the 2D Nuclear Magnetic Resonance Spectroscopy”. In Computational Optical Sensing and Imaging (pp. CTu5D-3). Optical Society of America, June 2018.

2. COMPRESSIVE SPECTRAL VIDEO SENSING

This chapter overviews some background topics related to this dissertation. First, the compressive-sensing-based sampling scheme for video rate spectral imaging is presented. Then, some compressive spectral video sensing architectures are described. Later, the recovery problem for compressive spectral video measurements is presented.

2.1. Spectral Video Acquisition via Compressive Sensing

Different from the Nyquist-based sampling or full-sampling theory, compressive sampling (CS) establishes that a signal can be recovered from a number of samples significantly smaller than those required by the Nyquist criterion, if the signal has a sparse or compressible representation in some known transform basis or dictionary [23][24]. In particular, a signal is sparse if most of its coefficients are zero, and a signal is compressible if its coefficients decay quickly when sorted in decreasing order of magnitude [25].

Based on CS, several compressive spectral video sensing (CSVS) architectures have been developed and implemented, enabling the acquisition of dynamic spectral scenes onto compressed observations [26][12]. Particularly, the coded aperture snap-

---

[23] Marco F Duarte and Richard G Baraniuk. "Kronecker compressive sensing". In: *IEEE Transactions on Image Processing* 21.2 (2012), pp. 494–504.

[24] Emmanuel J Candès and Michael B Wakin. "An introduction to compressive sampling". In: *IEEE signal processing magazine* 25.2 (2008), pp. 21–30.

[25] Gabriel Cristóbal, Peter Schelkens, and Hugo Thienpont. *Optical and digital image processing: fundamentals and applications*. John Wiley & Sons, 2013.

[26] Ashwin Wagadarikar et al. "Single disperser design for coded aperture snapshot spectral imag-

shot spectral imager (CASSI) is a remarkable architecture that has demonstrated to be suitable for the acquisition of spectral video since the spectral information is measured from a single exposure or snapshot on the camera sensor [1312271]. Other related methods for compressive spectral video acquisition, such as the hybrid spectral video imaging system (HVIS) [2829] and the high-speed hyperspectral (HSHS) imager [2], add an extra-camera into the optical path to acquire high spatial and spectral resolution but require more complex optics configuration and calibration processes.

2.1.1. Matrix-form CSVS Modeling A CSVS system is principally composed of four optical elements: a set of lenses, a coded aperture (CA) or mask, a dispersive element, and a focal plane array (FPA) detector. Into the system, each 3D spectral frame is encoded and dispersed to be integrated onto the 2D detector, where the order of the encoding and dispersion procedures is set according to the optical configuration of the specific system [261213]. Mathematically, let $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ be a discrete spectral video where $I_1 \times I_2$ represents the spatial size, I_3 the spectral bands, and I_4 the number of video frames. Let $\mathbf{f} = [\mathbf{f}_0^T, \mathbf{f}_1^T, \ldots, \mathbf{f}_{I_4-1}^T]^T$ be the column-wise vector form of $\mathcal{F}$, where $\mathbf{f} \in \mathbb{R}^n$ and $n = I_1 I_2 I_3 I_4$. Specifically, the column-wise vectorization is given by $\mathbf{f} = \mathrm{vec}(\mathcal{F}) = [\mathbf{f}_0^T, \mathbf{f}_1^T, \ldots, \mathbf{f}_{I_4-1}^T]^T$, where each frame $\mathbf{f}_{i_4}$ is written as $\mathbf{f}_{i_4} = [\mathbf{f}_0^{i_4}, \ldots, \mathbf{f}_{I_3-1}^{i_4}]$, and whose entries can be expressed as $(f_{i_3}^{i_4})_r = \mathcal{F}_{(r-\ell I_1),\ell,i_3,i_4}$,

---

ing". In: *Applied optics* 47.10 (2008), B44–B51.

[27] Claudia V Correa-Pugliese, Diana F Galvis-Carreño, and Henry Arguello-Fuentes. "Sparse representations of dynamic scenes for compressive spectral video sensing". In: *Dyna* 83.195 (2016), pp. 42–51.

[28] Chenguang Ma et al. "Acquisition of high spatial and spectral resolution video with a hybrid camera system". In: *International journal of computer vision* 110.2 (2014), pp. 141–155.

[29] Xun Cao et al. "High resolution multispectral video capture with a hybrid camera system". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 297–304.

where $\ell = \lfloor r/I_1 \rfloor$, $r = 0, ..., I_1 I_2 - 1$, $i_3 = 0, ..., I_3 - 1$, and $i_4 = 0, ..., I_4 - 1$. Then, the acquisition procedure in a CSVS architecture can be modeled as

$$\underbrace{\begin{bmatrix} \mathbf{y}_0 \\ \vdots \\ \mathbf{y}_{i_4} \\ \vdots \\ \mathbf{y}_{I_4-1} \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \mathbf{H}_0 & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ & \ddots & & & \\ \vdots & & \mathbf{H}_{i_4} & & \vdots \\ & & & \ddots & \\ \mathbf{0} & \mathbf{0} & \cdots & & \mathbf{H}_{I_4-1} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} \mathbf{f}_0 \\ \vdots \\ \mathbf{f}_{i_4} \\ \vdots \\ \mathbf{f}_{I_4-1} \end{bmatrix}}_{\mathbf{f}} + \boldsymbol{\omega}, \tag{1}$$

for $i_4 = 0, ..., I_4 - 1$, where $\mathbf{y}_{i_4} = \mathbf{H}_{i_4}\mathbf{f}_{i_4}$ denotes the compressive projection for the i_4-th frame, $\mathbf{f}_{i_4}$ is the vector-form of the i_4-th spectral video frame, $\mathbf{H}_{i_4}$ is the i_4-th CSVS measurement matrix, and $\boldsymbol{\omega}$ denotes the noise of the sensing system. In particular, the structure of the measurement matrix $\mathbf{H}_{i_4}$ depends on the physical phenomenon produced by the system, and its entries are the column vectorization of the coded aperture (CA). In the next section, some state-of-the-art CAs are presented.
Succinctly, Eq. (1) can be expressed as

$$\mathbf{y} = \mathbf{H}\mathbf{f} + \boldsymbol{\omega}, \tag{2}$$

where $\mathbf{H} \in \mathbb{R}^{m \times n}$ accounts for the encoding and dispersion processes for the I_4 frames of the full video and $\mathbf{y} \in \mathbb{R}^m$ represents the compressed measurement vector, with $m \ll n$. Additionally, exploiting the fact that spectral videos can be highly sparse or compressible in some basis, i.e. $\mathbf{f} = \boldsymbol{\Psi}\boldsymbol{\theta}$, the set of CSVS outputs can be rewritten as

$$\mathbf{y} = \mathbf{H}\boldsymbol{\Psi}\boldsymbol{\theta} + \boldsymbol{\omega} = \mathbf{A}\boldsymbol{\theta} + \boldsymbol{\omega}, \tag{3}$$

where $\mathbf{A} = \mathbf{H}\boldsymbol{\Psi}$ is the sensing matrix of the system, $\boldsymbol{\theta} \in \mathbb{R}^n$ represents the sparse coefficients of the signal and if $\boldsymbol{\theta}$ has at most K non-zero entries, it is known as

K-sparse, i.e., $||\boldsymbol{\theta}||_0 \leq K$, with $K \ll n$.

2.1.2. Sparse Transform The sparse representation bases used to exploit the inherent redundancy of high-dimensional videos include synthesis dictionaries [30][31], analytical transforms such as Wavelets or Cosines [12], and analytical transforms based on the Kronecker product of one-dimensional bases for each signal dimension [27]. Particularly, the sparse representation based on the Kronecker product for a 4D spectral video can be expressed as

$$\mathbf{f} = \boldsymbol{\Psi}_{4D}\boldsymbol{\theta} = \boldsymbol{\Psi}_1 \otimes \boldsymbol{\Psi}_2 \otimes \boldsymbol{\Psi}_3 \otimes \boldsymbol{\Psi}_4\boldsymbol{\theta}, \tag{4}$$

where $\{\boldsymbol{\Psi}_r\}_{r=1}^4$ is a set of one-dimensional sparsifying transforms. Figure 1 illustrates the sparse representation coefficients for four different transforms from a spectral video with 128×128 spatial pixels, 8 spectral bands and 8 frames, where its RGB representation and sorted coefficients are shown in Figure 1(a) and (b), respectively. Figure 1(c) shows the coefficients using a 1D Wavelet transform, this is $\mathbf{f} = \boldsymbol{\Psi}_{1D}\boldsymbol{\theta}$; Figure 1(d) shows the coefficients from a 2D Wavelet, where the sparsification is applied on the spatial axis; Figure 1(e) shows the coefficients from the Kronecker product between a 2D Wavelet, for the spatial dimension, and a 1D DCT, for the spectral dimension; and Figure 1(f) shows the coefficients from the 4D Kronecker: a Kronecker product between a 2D Wavelet, for the spatial dimension, a 1D DCT, for the spectral dimensions, and a 1D DCT, for the temporal axis (this configuration was

---

[30] Ajmal Mian and Richard Hartley. "Hyperspectral video restoration using optical flow and sparse coding". In: *Optics express* 20.10 (2012), pp. 10658–10673.

[31] Lizhi Wang et al. "High-speed hyperspectral video acquisition with a dual-camera architecture". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 4942–4950.

selected based on the sparse transform analysis performed in [32] and in the experiments reproduced in Annex 1). It can be noticed that the 4D Kronecker produces the sparsest representation, since most of the information is concentrated in fewer coefficients in comparison with the other bases, where the sparsest representation is that in which a large number of projected coefficients are small enough to be ignored [33] (see Annex 1 for more details). Based on literature and Figure 1, the 2D Wavelet transform provides the sparsest representation for the spatial information [34][27], however, in practice, the signal must be of a dyadic length for a fast and a compact representation.

The main limitations of the state-of-the-art bases for spectral video representation are: First, learning synthesis dictionaries can be a time-consuming task and can introduce drawbacks for real-time processing since it is necessary to learn the basis offline from the acquired data to recover it later, such as in [2], where the learning and reconstruction procedures take up to three hours for one spectral video scene. Second, some analytical transforms such as Wavelets are image-size restrictive, e.g., for having a fast and compact transformation the signal must be dyadic, and independent of the data. Given these points, it is crucial for spectral video representations to have more flexible bases that can be adapted to the time-varying information that impinges on the sensor, and that can fully exploit the highly redundant information and correlated structure of spectral videos without incurring in time-consuming learning tasks.

---

[32] Kareth Marcela León-López. "DISEÑO DE APERTURAS DE CODIFICACIÓN PARA LA ADQUISICIÓN COMPRESIVA DE IMÁGENES ESPECTRALES DINÁMICAS [recurso electronico]". M.S. Thesis. Bucaramanga, Colombia: Universidad Industrial de Santader (UIS), 2017.

[33] Saad Qaisar et al. "Compressive sensing: From theory to applications, a survey". In: *Journal of Communications and networks* 15.5 (2013), pp. 443–456.

[34] Gonzalo Arce et al. "Compressive coded aperture spectral imaging: An introduction". In: *IEEE Signal Processing Magazine* 31.1 (2014), pp. 105–115.

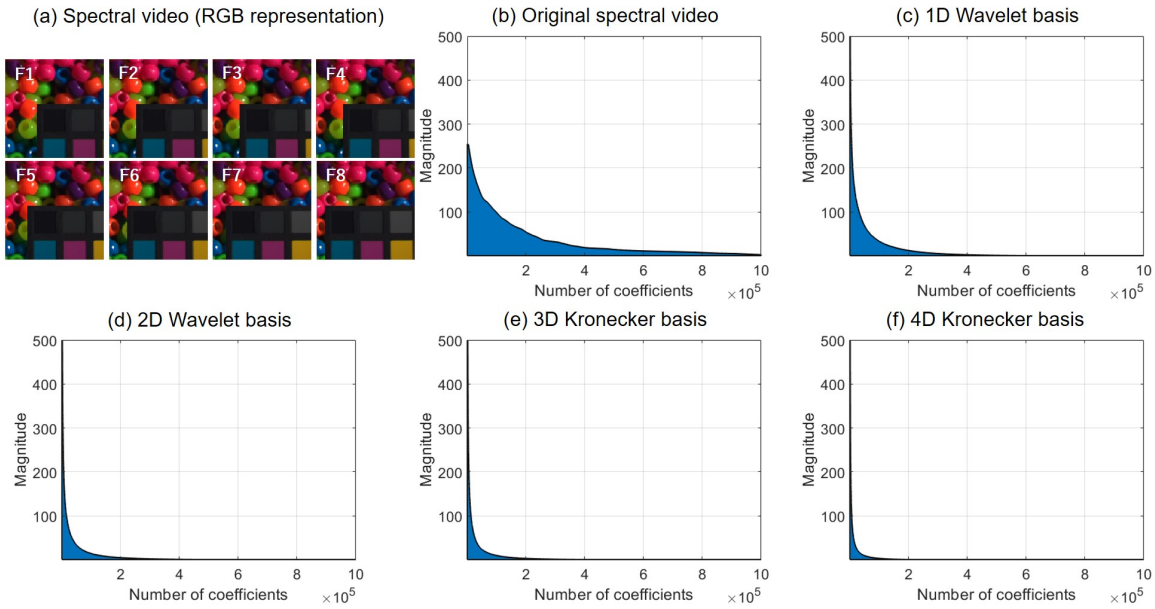


Figure 1. Sparse representation comparison of (a) a spectral video with 128×128 spatial pixels, 8 spectral bands and 8 frames between the (b) original spectral video coefficients and its representation on the (c) 1D Wavelet, (d) 2D Wavelet, (e) 3D Kronecker (2D Wavelet-DCT), and (f) 4D Kronecker (2D Wavelet-DCT-DCT) transforms.

A suitable approach for high-dimensional spectral video is the tensor representation, where the inherent structure and the local correlation inside multidimensional signals are considered [35]. A remarkable tensor representation is the Tucker decomposition which decomposes a higher-order array into a core tensor multiplied by an orthogonal matrix along each mode. Figure 2 illustrates the Tucker decomposition of a spectral video of size $I_1 \times I_2 \times I_3 \times I_4$, where the symbol $\times_z$ denotes the mode-z product, for $z = 1, ..., 4$ [35][36]. Observe in Figure 2 that $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3 \times R_4}$, with

[35] Tamara G Kolda and Brett W Bader. "Tensor decompositions and applications". In: *SIAM review* 51.3 (2009), pp. 455–500.

[36] Andrzej Cichocki et al. "Tensor decompositions for signal processing applications: From two-way to multiway component analysis". In: *IEEE Signal Processing Magazine* 32.2 (2015), pp. 145–163.

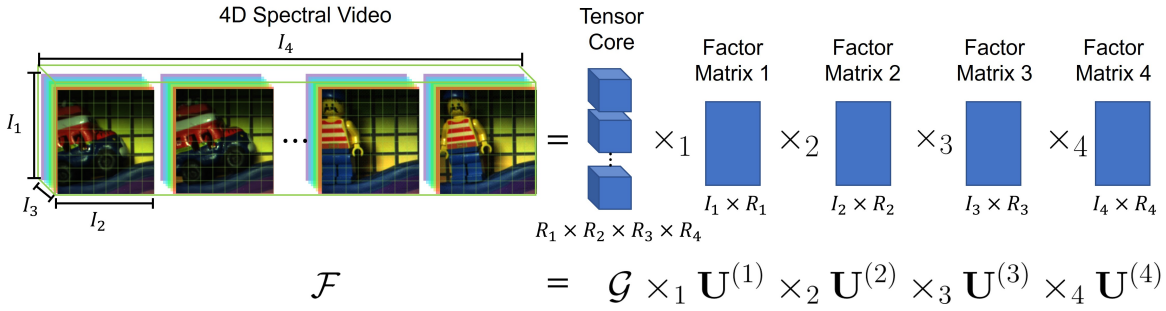


Figure 2. Tensor representation of a spectral video via Tucker decomposition.

$R_1 \leq I_1, R_2 \leq I_2, R_3 \leq I_3, R_4 \leq I_4$, denotes the core tensor, which it is assumed sparse, and the matrices $\mathbf{U}^{(z)}$, $z = 1, ..., 4$, are the unitary matrices. It can be noticed that each matrix $\mathbf{U}^{(z)}$ accounts for one dimension of the higher-order array.

2.2. CSVS Architectures and Coded Aperture Design

In this section, two CSVS architectures are mathematically described: the video colored coded aperture snapshot spectral imager (video-CASSI) and the ideal spatial-spectral coded compressive spectral imager (3D-CASSI), where the main difference between them is the shifting performed on the spectral information. Then, basic properties used for designing the sensing matrix entries and recent approaches for coded aperture design based on deep learning are presented.

2.2.1. Video Colored Coded Aperture Snapshot Spectral Imager Let $f_0(x, y, \lambda, t)$ be a dynamic spectral source incoming into the video-CASSI system. Then, each frame from the source is first encoded by a time-varying colored coded aperture $T(x, y, \lambda, t)$, where (x, y) represents the spatial coordinates, λ the wavelength component, and t for the temporal dimension. Later, the resulting encoded source is spectrally dispersed, to finally be integrated onto the sensor $Y_{i_4}(x, y, t)$. Each pixel in the sensor is a discretized measurement. In addition, during the integration time, the time-varying coded aperture remains fixed for each frame, that is, each frame

is modulated by a different pattern of the coded aperture. It is important to mention that a time-varying colored coded aperture is composed of a set of 2D arrays whose spatial points are optical filters, and where each array changes the filters position across time. Specifically, the coded aperture pixels are different optical filters with a specific spectral response which let certain frequency components of the source pixel to pass and reject the remaining ones. That is, the pixels can operate on the spectral axis as frequency-selective filters, i.e. as low pass, band pass, or high pass optical filters. For illustration purposes, Figure 3(a) shows the optical elements in the Video C-CASSI architecture [1], whose encoding element is a time-varying colored coded aperture and their pixels spectral responses are shown in Figure 3(b).

As a matter of fact, the colored coded apertures based on optical filters in the CASSI system are a modification of the traditional CASSI system, where the encoding element is a block-unblock coded aperture, i.e. wavelength-independent patterns that misuse the richness of the redundancy in the spectral information [26].

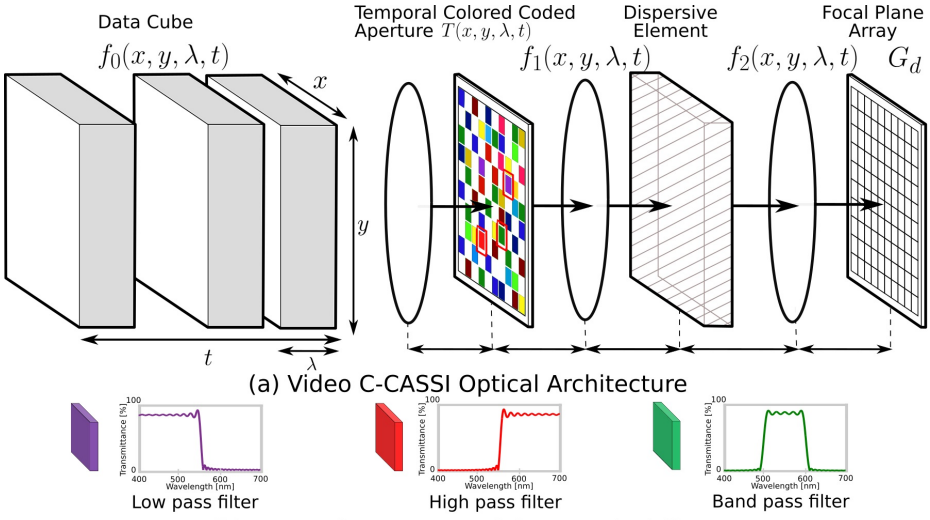


Figure 3. (a) Illustration of the video-CASSI system, where the encoding element is a time-varying colored coded aperture whose pixels (b) correspond to a specific spectral response.

In discrete form, the source $f_0(x, y, \lambda, t)$ can be written as $(\mathcal{F}_{i_3}^{i_4})_{i_1,i_2}$. Then, the discrete

output on the sensor for the i_4-th frame $\mathcal{F}^{i_4}$ can be written as

$$(\mathcal{Y}^{i_4})_{i_1,i_2} = \sum_{i_3=0}^{I_3-1} (\mathcal{F}_{i_3}^{i_4})_{i_1,(i_2-i_3)} (\mathcal{T}_{i_3}^{i_4})_{i_1,(i_2-i_3)} + (\Omega^{i_4})_{i_1,i_2}, \tag{5}$$

for $i_4 = 0, ..., I_4 - 1$, where $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$, $(\mathcal{Y}^{i_4})_{i_1,i_2}$ denotes the measurement at the (i_1, i_2)-th position on the detector at time i_4, $(\mathcal{T}_{i_3}^{i_4})_{i_1,i_2} \in \{0, 1\}$ represents the discrete form of the time-varying colored coded aperture, where the optical filters responses can be represented by a set of block-unblock elements; and $(\Omega^{i_4})_{i_1,i_2}$ represents the sensor noise [1]. Observe that the sub-index $(i_2 - i_3)$ represents the horizontal dispersion induced by the dispersive element in the discrete model.

In matrix form, the acquisition model is equivalent to that in Eq. (2), where the structure of the video C-CASSI measurement matrix $\mathbf{H} \in \mathbb{R}^{I_1(I_2+I_3-1)I_4 \times I_1 I_2 I_3 I_4}$ is shown in Figure 4 for $I_1 = 3, I_2 = 3, I_3 = 3$ and $I_4 = 4$ frames. Notice in Figure 4 that white elements on the diagonal represent the transmissive elements in the time-varying colored coded aperture.

2.2.2. Spatial-spectral Coded Compressive Spectral Imager A theoretical architecture that allows spatial-spectral encoding of the information and achieves a high performance is the 3D-CASSI [13]. In the 3D-CASSI extended to video acquisition, so-called video 3D-CASSI, the compressed measurements $\mathcal{Y}^{i_4}$ for the i_4-th frame can be modeled as

$$(\mathcal{Y}^{i_4})_{i_1,i_2} = \sum_{i_3=0}^{I_3-1} (\mathcal{F}_{i_3}^{i_4})_{i_1,i_2} (\mathcal{T}_{i_3}^{i_4})_{i_1,i_2} + (\Omega^{i_4})_{i_1,i_2}, \ i_4 = 0, ..., I_4 - 1, \tag{6}$$

where the spectral video and the coded aperture are not spectrally sheared unlike the previous architecture [13]. In matrix form, the measurement set is modeled as in Eq. (2), where the structure of the video 3D-CASSI measurement matrix $\mathbf{H} \in \mathbb{R}^{I_1 I_2 I_4 \times I_1 I_2 I_3 I_4}$ is presented in Figure 5 for $I_1 = 3, I_2 = 3, I_3 = 3$ and $I_4 = 4$ frames.

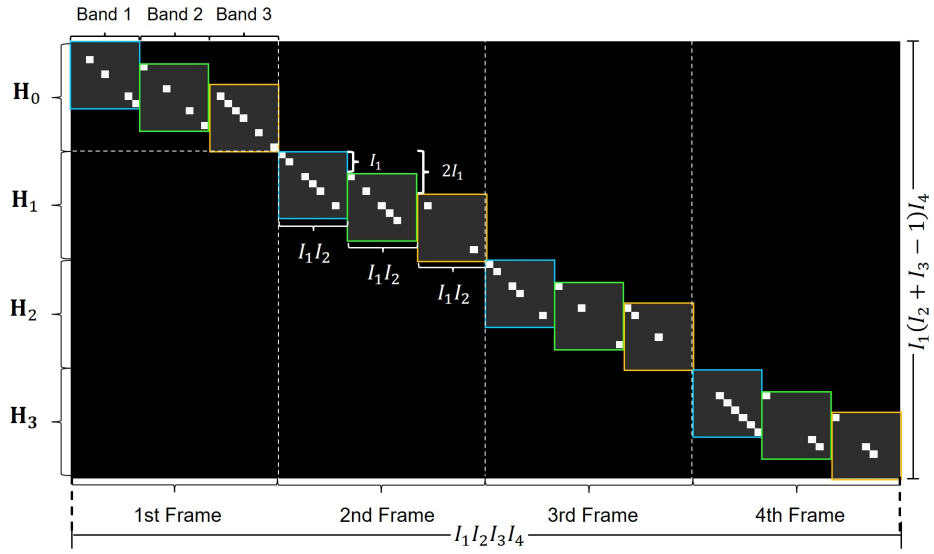


Figure 4. Video-CASSI measurement matrix $\mathbf{H} \in \mathbb{R}^{I_1(I_2+I_3-1)I_4 \times I_1I_2I_3I_4}$ for $I_1 = 3, I_2 = 3, I_3 = 3$ and $I_4 = 4$ frames. White squares on the diagonal depict transmissive elements (unblock elements) in the random time-varying colored coded aperture.

Notice that, compared with the video C-CASSI in Figure 4, there is no shifting on the spectral bands, hence, spatial information preserves its structure.

In general, it can be noticed from the two described architectures that although spectral frames are individually sampled, the sensing matrix and representation basis can be designed to exploit temporal correlations from the underlying scene since some information on the scene remains static over a time-lapse [137]

2.2.3. Sensing Matrix Design The structure of the sensing matrix of a CSVS system is given by both the specific configuration of the optical architecture and the entries of the encoding element, i.e. the coded aperture pattern. Traditionally, the entries of the coded aperture can be generated by following a random Gaussian

[37] Henry Arguello and Gonzalo Arce. "Colored coded aperture design by concentration of measure in compressive spectral imaging". In: *IEEE Transactions on Image Processing* 23.4 (2014), pp. 1896–1908.

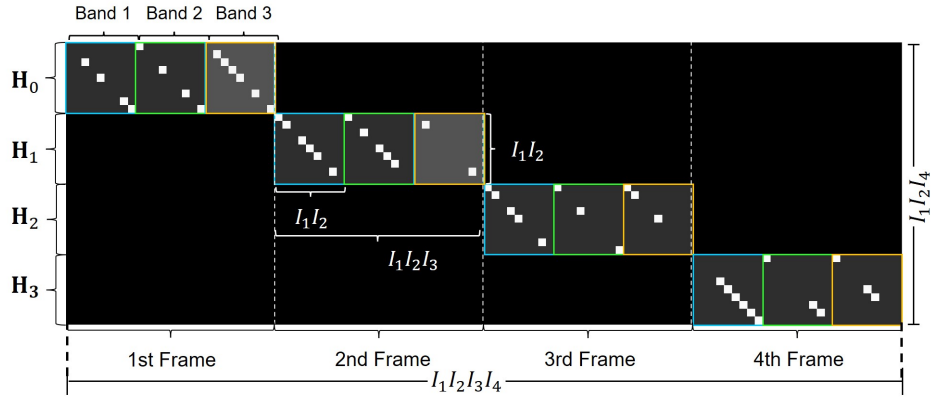


Figure 5. 3D-CASSI measurement matrix $\mathbf{H} \in \mathbb{R}^{I_1 I_2 I_4 \times I_1 I_2 I_3 I_4}$ for $I_1 = 3, I_2 = 3, I_3 = 3$ and $I_4 = 4$ frames.

or Bernoulli spatial distribution [2712], or by following a given structure such as the Boolean-coded apertures, which provide a spatially random distribution while exploiting temporal correlation [16]. However, different works have demonstrated that a specific spatio-spectral and temporal design improves the reconstruction results [137]. In the literature, some works have proposed the sensing matrix design in compressive sensing based on theoretical constraints such as in [38394041]. However, in most approaches the sensing matrix is a unitary dense Gaussian random matrix without a specific structure, whose entries are drawn from zero mean Gaussian random vari-

---

[38] Michael Elad. "Optimized projections for compressed sensing". In: *IEEE Transactions on Signal Processing* 55.12 (2007), pp. 5695–5702.

[39] Gang Li et al. "On projection matrix optimization for compressive sensing systems". In: *IEEE Transactions on Signal Processing* 61.11 (2013), pp. 2887–2898.

[40] Gang Li et al. "Designing robust sensing matrix for image compression". In: *IEEE Transactions on Image Processing* 24.12 (2015), pp. 5389–5400.

[41] Tao Hong and Zhihui Zhu. "An efficient method for robust projection matrix design". In: *Signal Processing* 143 (2018), pp. 200–210.

ables [42]. Different from these designs, sensing matrices that model implementable systems are highly structured, sparse and with binary entries due to the physical device that encodes the information. In general, there are two main properties to measure the recovery capabilities of sensing matrices: the restricted isometry property (RIP) and the coherence [43].

Restricted Isometry Property: This property provides guidelines to determine the sufficient number of projections in the system for signal reconstruction [143]. More specifically, for a given sensing matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, it is said that $\mathbf{A}$ satisfies the RIP of order $K \geq 1$ if for the smallest *restricted isometry constant* $\delta_K \geq 0$, the inequality $(1 - \delta_K)||\boldsymbol{\theta}||_2^2 \leq ||\mathbf{A}\boldsymbol{\theta}||_2^2 \leq (1 + \delta_K)||\boldsymbol{\theta}||_2^2$ holds for all $\boldsymbol{\theta} \in \mathbb{R}^n$ with $|\text{supp}(\boldsymbol{\theta}) \leq K|$. The RIP constant δ_K is then given by

$$\delta_K(\mathbf{A}) := \max_{|\mathcal{S}| \leq K} ||\mathbf{A}_{\mathcal{S}}^T \mathbf{A}_{\mathcal{S}} - \mathbf{I}_{\mathbb{R}^{\mathcal{S}}}||_2^2, \tag{7}$$

where the maximum is over all subsets $\mathcal{S}$ with $|\mathcal{S}| \leq K$, and $|\mathcal{S}| = \text{card}(\mathcal{S})$, where $\text{card}(\cdot)$ is the cardinality of the set, and $\mathbf{A}_{\mathcal{S}} \in \mathbb{R}^{m \times \mathcal{S}}$ is a sub-matrix of $\mathbf{A}$ whose columns are $\mathcal{S}$ columns of $\mathbf{A}$ [43]. A properly way to design $\mathbf{A}$ is by minimizing the constant δ_K to better satisfy the RIP [137].

Coherence: Similar to the RIP, the coherence property supplies directions to evaluate the fundamental conditions of $\mathbf{A}$ for efficient reconstruction from the compressed projections, where a low coherence is desired. Specifically, the coherence of the sensing matrix $\mathbf{A} = \mathbf{H}\boldsymbol{\Psi}$ can be defined as the maximum absolute value of the inner product between any two normalized columns of $\mathbf{A}$. In other words, coherence mea-

---

[42] Yuri Mejia and Henry Arguello. "Binary Codification Design for Compressive Imaging by Uniform Sensing". In: *IEEE Transactions on Image Processing* 27.12 (2018), pp. 5775–5786.

[43] Simon Foucart and Holger Rauhut. *A mathematical introduction to compressive sensing*. Vol. 1. 3. Birkhäuser Basel, 2013.

sures the correlation between $\mathbf{H}$ and $\mathbf{\Psi}$, where a low correlation guarantees uniqueness on the solution [1643]. Formally, coherence of $\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_n]$, with $||\mathbf{a}_i||_2 = 1$, $\forall i \in [n]$, is defined as [43]

$$\mu(\mathbf{A}) := \max_{1 \leq i \neq j \leq n} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|. \tag{8}$$

Works such as [44] have addressed the design and optimization of compressive spectral imagers based on the analysis of the coherence of the sensing matrix, where the sensing matrix structure is optimized in the sense that an upper bound of the coherence is minimized.

2.3. Reconstruction Problem

The inverse problem to recover $\mathcal{F}$ from the measurements $\mathbf{y} = \mathbf{Hf}$ entails the seeking of the sparse solution of the underlying scene. For this, an unconstrained optimization problem that consists in minimizing an objective function composed of a quadratic error term and a sparsity-promoting term has been posed. Formally, the optimization problem is written as

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ \frac{1}{2} ||\mathbf{y} - \mathbf{A}\boldsymbol{\theta}||_2^2 + \phi(\boldsymbol{\theta}) \right\}, \\ \hat{\mathbf{f}} &= \mathbf{\Psi}\hat{\boldsymbol{\theta}}, \end{aligned} \tag{9}$$

where $\mathbf{A}$ is the sensing matrix of the system and $\phi(\boldsymbol{\theta})$ is a regularization function for the sparsity-promoting solution [24271].

Different recent approaches have proposed to recover the underlying spectral signal

---

[44] Alejandro Parada-Mayorga and Gonzalo R Arce. "Colored Coded Aperture Design in Compressive Spectral Imaging via Minimum Coherence". In: *IEEE Transactions on Computational Imaging* 3.2 (2017), pp. 202–216.

by exploiting deep neural networks [45][46]. Thus, the weights of the network are trained to recover a version of the signal from the compressed measurements. Even though the training of the deep learning model is intensive, the recovery task is performed in seconds, providing *real-time* reconstructions. Works such as [15] have recovered spectral images at video-rates by using real coded apertures from the CASSI system, where the coded aperture is set as random.

Other works in video compressive sensing have demonstrated that the training of the weights of both the recovery and the coded aperture improves the reconstructions results [21][20][19]. Thus, this dissertation explores the jointly training of the recovery and coded aperture designs.

---

[45] Daniel Gedalin, Yaniv Oiknine, and Adrian Stern. "DeepCubeNet: reconstruction of spectrally compressive sensed hyperspectral images with deep neural networks". In: *Optics express* 27.24 (2019), pp. 35811–35822.

[46] Xin Miao et al. "l-net: Reconstruct hyperspectral images from a snapshot measurement". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 4059–4069.

3. ONLINE TENSOR SPARSIFYING TRANSFORM FROM COMPRESSIVE SPECTRAL VIDEO MEASUREMENTS

3.1. Introduction

Taking advantage of the fact that spectral videos can be highly sparse or compressible in some bases, i.e. $\boldsymbol{f} = \boldsymbol{\Psi\theta}$, the measured projections can be rewritten as $\mathbf{y} = \mathbf{H\Psi\theta}$, where $\boldsymbol{\theta} \in \mathbb{R}^n$ represents the sparse coefficients of the signal over the spatial, spectral and temporal axes [27]. Thus, the set of CSVS outputs $\mathbf{y} = [\mathbf{y}_0^\intercal, \mathbf{y}_1^\intercal, \ldots, \mathbf{y}_{I_4-1}^\intercal]^\intercal$ can be rewritten as $\mathbf{y} = \mathbf{H\Psi\theta} = \mathbf{A\theta}$, where the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ is named the CSVS sensing matrix. It is important to highlight that the sensing matrix structure and its entries depend on the used optical configuration. The underlying spectral video is then recovered from $\hat{\mathbf{f}} = \boldsymbol{\Psi}\big(\underset{\boldsymbol{\theta}}{\operatorname{argmin}}||\mathbf{y} - \mathbf{H\Psi\theta}||_2^2 + \rho||\boldsymbol{\theta}||_1\big)$, where ρ is a regularization constant.

To obtain $\mathbf{f}$ from the measurements $\mathbf{y}$, CS theory establishes two principles: sparsity, which is related to the signal under observation projected on a basis $\boldsymbol{\Psi}$, and incoherence, which is related to the sensing matrix $\mathbf{A}$ [24]. Particularly, the coherence of $\mathbf{A}$ measures the largest correlation between any two columns of $\mathbf{H}$ and $\boldsymbol{\Psi}$, where a low coherence is desired. In this manner, the sparse representation basis $\boldsymbol{\Psi}$ relates the two principles for the accurate signal recovery playing an important role in the CS-based reconstruction protocol. To date, transformation bases such as analytical transforms, e.g. Wavelets or Cosines [12], and learned dictionaries [302] have been used to exploit the inherent redundancy of high-dimensional videos. Moreover, sparse transforms based on the Kronecker product of one-dimensional bases for each signal dimension have also been used [27]. However, in the CSVS framework, learning dictionaries can be a time-consuming task and can introduce drawbacks for real-time processing since it is necessary to offline learn the basis from the acquired

data and later, to recover the signal using the learned basis, such as in [2]. Additionally, some of the other bases are image-size restrictive and independent of the data, such as the Wavelets. Therefore, it is crucial for CSVS reconstruction to have more flexible bases that can be adapted according to the time-varying information.

In addition, it is well known in the literature that most of the CS-based-acquisition modeling of optical systems and, in turn, the sparse representation models, rely on the data representation in vector/matrix form [13]. In consequence, high-dimensional signals, as it is the case of spectral videos, measured by huge sensing matrices are converted into very long vectors leading to a high computational burden [47]. In contrast, tensor data analysis offers the opportunity to model the data in its natural representation, e.g. a spectral video can be modeled as a four-dimensional (4D) array [48]. Tensors are the high-order generalizations of vectors and matrices, where the tensor decomposition allows the interaction between the multiple data dimensions and, also, a compact representation of high-dimensional signals [35].

3.1.1. Tensor Sparsifying Transform and High-dimensional data Recent works have modeled the CS problem based on tensors taking full advantage of the high-

---

[47] Shmuel Friedland, Qun Li, and Dan Schonfeld. "Compressive sensing of sparse tensors." In: *IEEE Trans. Image Processing* 23.10 (2014), pp. 4438–4447.

[48] Wenfei Cao et al. "Total variation regularized tensor RPCA for background subtraction from compressive measurements". In: *IEEE Transactions on Image Processing* 25.9 (2016), pp. 4075–4090.

order structure of the signals and better representing the data [49][50][51]. Thereby, based on the Kronecker CS formulation proposed in [23], tensor CS modeling makes preserving the structure of the high-dimensional signal possible and separates the sensing process along signal dimensions, easing the data storage and reducing computational complexity [47][50]. Moreover, several works have investigated sparsifying transforms based on tensors for hyperspectral images, focusing on denoising and redundancy reduction algorithms [52][53]. Other works have used tensor-based sparsifying transforms by applying tensor factorization, such as the Tucker decomposition, to the CS framework[54]. However, to date, the state-of-the-art has focused just on magnetic resonance imaging (MRI) and dynamic MRI (dMRI) signals, and they report extensive computation time in the numerical experiments making the method impractical for real applications. In [48], a tensor-based sparse model using several similar groups of patches for background subtraction was proposed, where the patch-based processing greatly reduces the computational costs. However, this approach is focused

---

[49] Zhixi Feng et al. “Superpixel Tensor Sparse Coding for Structural Hyperspectral Image Classification”. In: *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens* 10.4 (2017), pp. 1632–1639.

[50] Xin Ding, Wei Chen, and Ian J Wassell. “Joint sensing matrix and sparsifying dictionary optimization for tensor compressive sensing”. In: *IEEE Transactions on Signal Processing* 65.14 (2017), pp. 3632–3646.

[51] Miguel Marquez, Hoover Rueda-Chacon, and Henry Arguello. “Compressive Spectral Light Field Image Reconstruction via Online Tensor Representation”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3558–3568.

[52] Xuefeng Liu, Salah Bourennane, and Caroline Fossati. “Denoising of hyperspectral images using the PARAFAC model and statistical performance analysis”. In: *IEEE Transactions on Geoscience and Remote Sensing* 50.10 (2012), pp. 3717–3724.

[53] Lefei Zhang et al. “Compression of hyperspectral remote sensing images by tensor approach”. In: *Neurocomputing* 147 (2015), pp. 358–363.

[54] Yeyang Yu et al. “Multidimensional compressed sensing MRI using tensor decomposition-based sparsifying transform”. In: *PloS one* 9.6 (2014), e98441.

on background subtraction over grayscale videos where the spectral information is discarded.

On the other hand, different from full-video processing, a promising field to better use correlation and sparsity of high-dimensional signals is the employment of local coherent patches on compressive video processing and reconstruction [55], usually estimated from block matching[55] or k-nearest neighbor[48] techniques. In fact, the idea of using patches based on superpixels for better handling large-scale datasets has gained research interest in recent years [56]. For video applications, the concept of temporal superpixels (TSP) has been introduced to group local and coherent patches of spatial information across time[56][57]. However, this approach has not been exploited in CSVS reconstruction, where the acquisition of compressed dynamic spectral information is considered.

This chapter presents a tensor-based model to simultaneously learn and reconstruct the sparse transform and the spectral video under a CSVS framework. The proposed tensor-decomposition-based learning model (TenDL) can be performed either on full-images or onto patches obtained from temporal superpixels (TSP-TenDL), where the TSP-TenDL model exhibits less computational complexity. Specifically, contrary to offline dictionary learning, the proposed tensor sparsifying transform is updated online from the data, while data are reconstructed.

To fully exploit the structure, as well as the high signal correlation within a 4D spectral

---

[55] Bihan Wen, Saiprasad Ravishankar, and Yoram Bresler. "VIDOSAT: High-Dimensional Sparsifying Transform Learning for Online Video Denoising". In: *IEEE Transactions on Image Processing* 28.4 (2019), pp. 1691–1704.

[56] Radhakrishna Achanta et al. "SLIC superpixels compared to state-of-the-art superpixel methods". In: *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), pp. 2274–2282.

[57] Jason Chang, Donglai Wei, and John W Fisher. "A video representation using temporal superpixels". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 2051–2058.

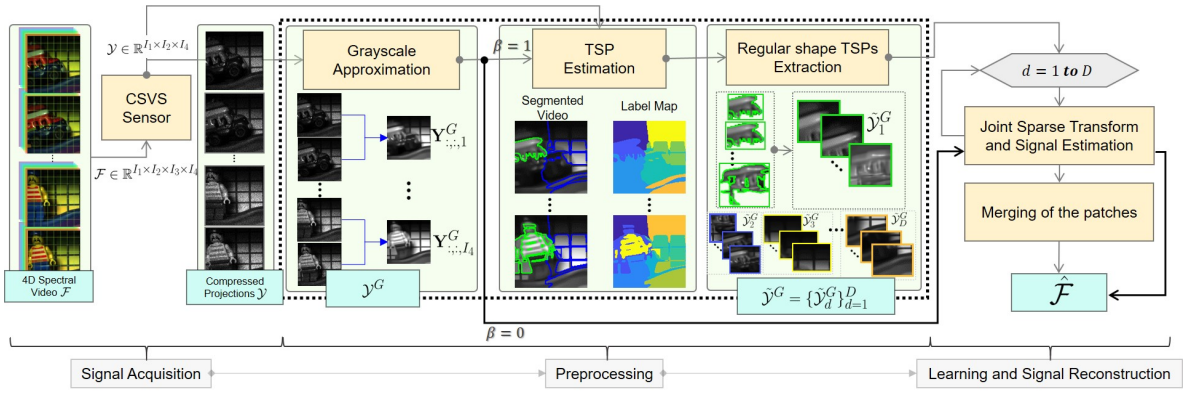


Figure 6. Flowchart of the general steps of the proposed framework for the simultaneously sparse transform learning and signal reconstruction. The dotted line square marks off the developed strategy to estimate both a grayscale approximation and the temporal superpixels patches from the compressed measurements.

video, the problem formulation is performed on tensor form with or without several patches, since each patch contains highly-correlated information. The flowchart of the proposed approach is pictured in Figure 6, where three stages are highlighted: signal acquisition, preprocessing (dotted line square), and joint learning and reconstruction. In the preprocessing stage, the parameter β establishes if the video is processed either fully ($\beta = 0$) or by patches ($\beta = 1$). Then, after obtaining the measurements from the CSVS sensor, the compressed video is used to first estimate a grayscale approximation and second to compute the TSP, where the patches can be efficiently computed from a TSP segmentation algorithm, such as [5657]. Later, in the reconstruction stage, the grayscale estimation is used to initialize the sparse transform to be learned. An algorithm based on the block-coordinate descent method for the simultaneous sparse transform learning and reconstruction optimization problem is proposed. In the algorithm, if the process is performed on the patches stage, each patch is independently reconstructed and then merged to obtain the spectral video. Numerical experiments show an improvement by the TenDL and TSP-TenDL approaches compared to analytical and offline-learned bases in terms of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) index on the reconstruction

results. Gains of up to 7 dB of PSNR and 0.1 of SSIM are achieved with respect to the state-of-the-art recovery and the tensor-based recovery with the fixed basis. The impact of the number of TSP on the reconstruction when the data is processed on patches and the reconstruction time are also analyzed.

Notation and Multilinear Algebra

Tensors are denoted by Euler script letters, e.g. $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times ... \times I_N}$ is an N-th order tensor. Matrices are denoted by boldface capital letters, e.g. $\mathbf{X}$, and vectors are denoted by boldface lowercase letters, e.g. $\mathbf{x}$. The n-th matrix in a set is denoted by adding a superscript in parenthesis as $\mathbf{A}^{(n)}$. Subtensors can be created from the original when a subset of indices is fixed. Slices are obtained by fixing all but two indices, e.g. $\mathbf{X}_{:,:,i_3,...,i_N}$ is a frontal slice of $\mathcal{X}$ [35], and fibers are vector-valued subtensors defined by fixing every index except one. The mode-n matrix representation of a tensor $\mathcal{X}$ is denoted as $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times (I_1 ... \times I_{n-1} \times I_{n+1} ... \times I_N)}$, and it is obtained by arranging the mode-n fibers as the column of the resulting matrix. The mode-n product between a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times ... \times I_N}$ and a matrix $\mathbf{A} \in \mathbb{R}^{J \times I_n}$ is denoted as $\mathcal{Z} = \mathcal{X} \times_n \mathbf{A}$, with $\mathcal{Z} \in \mathbb{R}^{I_1 ... \times I_{n-1} \times J \times I_{n+1} ... \times I_N}$, and in matrix representation is written as $\mathbf{Z}_{(n)} = \mathbf{A}\mathbf{X}_{(n)}$. The Tucker decomposition is defined as a higher-order analogue of the singular value decomposition, where the goal is to decompose a tensor $\mathcal{X}$ into a core tensor $\mathcal{B}$ multiplied by a set of matrices $\mathbf{A}^{(n)} \in \mathbb{R}^{J_n \times I_n}$, for $n = 1, ..., N$, along each mode [35]:

$$\mathcal{X} = \mathcal{B} \times_1 \mathbf{A}^{(1)} ... \times_N \mathbf{A}^{(N)} = [\![\mathcal{B}; \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}]\!]. \tag{10}$$

Equivalently, the Tucker decomposition of Eq. (10) can be expressed as $\mathbf{X}_{(n)} = \mathbf{A}^{(n)}\mathbf{X}_{(n)}\left(\mathbf{A}_{(N)} \otimes ... \mathbf{A}_{(n+1)} \otimes \mathbf{A}_{(n-1)} \otimes ... \mathbf{A}_{(1)}\right)^T$, where $\otimes$ denotes the Kronecker product and $(\cdot)^T$ is the transpose.

The column-wise vectorization of a 4-th order spectral video tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$

is given by $\mathbf{x} = \mathrm{vec}(\mathcal{X}) = [\mathbf{x}_0^\intercal, \mathbf{x}_1^\intercal, \ldots, \mathbf{x}_{I_4-1}^\intercal]^\intercal$, where each frame $\mathbf{x}_{i_4}$ is written as $\mathbf{x}_{i_4} = [\mathbf{x}_0^{i_4}, \ldots, \mathbf{x}_{I_3-1}^{i_4}]$, and whose entries can be expressed as $(x_{i_3}^{i_4})_r = \mathcal{X}_{(r-kI_1),k,i_3,i_4}$, where $k = \lfloor r/I_1 \rfloor$, $r = 0, ..., I_1I_2 - 1$, $i_3 = 0, ..., I_3 - 1$, and $i_4 = 0, ..., I_4 - 1$. The inverse operator of $\mathrm{vec}(\cdot)$ that rearranges the vectorized signal to the original tensor shape is denoted by $\mathrm{vec}^{-1}(\mathcal{X})$. Table 1 summarizes the above-mentioned notations and operations. Note that notation in Table 1 is considered only for this Chapter.

Table 1. Notation summary

Notation	**Description**
$\mathcal{X}, \mathbf{X}, \mathbf{x}, x$	Tensor, matrix, vector, scalar
$\mathbf{X}_{:,:,i_3,\ldots,i_N}$	Frontal slice of tensor $\mathcal{X}$
$\mathbf{x}_{:,i_2,i_3,\ldots,i_N}$	Vector fiber of tensor $\mathcal{X}$
$\mathbf{X}_{(n)}$	Mode-n matrix representation or unfolding of tensor $\mathcal{X}$
$\mathbf{A}^{(n)}$	n-th matrix in a sequence
$\mathcal{Z} = \mathcal{X} \times_n \mathbf{A}$	mode-n product between a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ and a matrix $\mathbf{A} \in \mathbb{R}^{J \times I_n}$
$\mathcal{X} = [\![\mathcal{B}; \mathbf{A}^{(1)}, ..., \mathbf{A}^{(N)}]\!]$	Tucker decomposition of $\mathcal{X}$, $\mathcal{X} = \mathcal{B} \times_1 \mathbf{A}^{(1)} ... \times_N \mathbf{A}^{(N)}$
$\mathrm{vec}(\mathcal{X})$	Column-wise vectorization of the tensor $\mathcal{X}$
$\mathrm{vec}^{-1}(\mathcal{X})$	Rearrange $\mathrm{vec}(\mathcal{X})$ in its original shape

3.2. Tensor-based Compressive Spectral Video Sensing (CSVS) Modeling

Snapshot compressive spectral cameras for capturing dynamic spectral images rely on the modulation of the incoming light towards the camera sensor [13]. Some compressive spectral video sensing architectures include the video colored coded aperture snapshot spectral imager (video C-CASSI) [1], the hybrid spectral video imaging system (HVIS), the high-speed hyperspectral (HSHS) and the spatial-spectral coded compressive spectral imager (3D-CASSI) for video-rate [13]. In particular, the 3D-CASSI provides a spatio-spectral modulation of the data cube using a 3D coded aperture, also known as a colored coded aperture [37]. In the video 3D-CASSI, the

compressed measurements y can be modeled as

$$y(x', y', t) = \int f(x', y', \lambda, t) T(x', y', \lambda, t) d\lambda, \tag{11}$$

where $f(x', y', \lambda, t)$ denotes the dynamic spectral source, $T(x', y', \lambda, t)$ represents the time-varying colored coded aperture[1], (x', y') represent the spatial coordinates, λ denotes the wavelength component, and t accounts for the time axis.

A discretized spectral video scene can be represented as a fourth-order tensor $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$, where $I_1 \times I_2$ represents the spatial size, I_3 the spectral bands, and I_4 the number of video frames. In general form, the acquisition procedure in a CSVS architecture can be expressed as

$$\mathcal{Y} = \mathcal{H}(\mathcal{F}) + \mathcal{W}, \tag{12}$$

where $\mathcal{H}(\mathcal{F}) : \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4} \rightarrow \mathbb{R}^{I_1 \times I_2 \times I_4}$ represents the CSVS operator and establishes the modulation and compression of the incoming signal. In particular, for the video 3D-CASSI measurements $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_4}$, the i_4-th frame can be modeled as

$$\mathcal{Y}_{:,:,i_4} = \sum_{i_3=1}^{I_3} \mathcal{F}_{:,:,i_3,i_4} \circ \mathcal{T}_{:,:,i_3,i_4} + \mathcal{W}_{:,:,i_4}, \tag{13}$$

for $i_4 = 1, ..., I_4$, where $\circ$ denotes the Hadamard product (element-wise product), $\mathcal{F}_{:,:,i_3,i_4}$ is a frontal slice of $\mathcal{F}$, $\mathcal{W} \in \mathbb{R}^{I_1 \times I_2 \times I_4}$ denotes the noise in the system, and $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ represents the tensor form of the time-varying colored coded aperture (T-CCA). In particular, the entries of the T-CCA can be generated by following a specific design, as in [371], or by following a structure such as the Boolean coded apertures, which provide a spatially random distribution and exploit the temporal correlation[16].

Following the tensor notation, the spectral video $\mathcal{F}$ can be decomposed as a multilinear transformation of a dictionary basis $\{\mathbf{\Psi}^{(n)}\}_{n=1}^{4} \in \mathbb{R}^{I_n \times I_n}$, along each mode-$n$,

and a core tensor $\mathcal{G} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ as follows

$$\mathcal{F} = \mathcal{G} \times_1 \mathbf{\Psi}^{(1)} \times_2 \mathbf{\Psi}^{(2)} \times_3 \mathbf{\Psi}^{(3)} \times_4 \mathbf{\Psi}^{(4)}, \tag{14}$$

where the core tensor $\mathcal{G}$ corresponds to the coefficients of $\mathcal{F}$ on each dictionary, assumed to be sparse. Thus, Eq. (12) can be rewritten as

$$\mathcal{Y} = \mathcal{H}\big(\mathcal{G} \times_1 \mathbf{\Psi}^{(1)} \times_2 \mathbf{\Psi}^{(2)} \times_3 \mathbf{\Psi}^{(3)} \times_4 \mathbf{\Psi}^{(4)}\big) + \mathcal{W}, \tag{15}$$

which is the general tensor-based CS acquisition model for a CSVS architecture.

3.3. TSP from Compressed Measurements for Online Learning and Recovery Estimation

In this section, the concept of temporal superpixels (TSP) is introduced, and the proposed strategy to compute the grayscale scene version that leads to an accurate TSP estimation from the measurements is described. The problem to jointly learn the sparse basis and recover the signal is formulated.

3.3.1. Temporal superpixel subtensors Spectral video tensors contain highly redundant information such that several pixels share similar features in the spatial, spectral, and temporal axes. Thus, these tensors can be partitioned into several four-dimensional (4D) patches to speed up processing tasks or to alleviate storage load. A 4D patch tensor $\mathcal{F}_d$ can be defined as a subtensor of $\mathcal{F}$, where $d = 1, ..., D$ indicates the d-th 4D patch. A rough way to obtain the 4D patches consists in splitting up the 4D information following a regular grid over the spatial dimension across time; however, these regular shapes entail the grouping of non-smooth regions. In contrast, superpixels provide a suitable partition of the data, assigning each pixel to a coherent spatial local region [56], and these local regions connected in successive

frames are known as temporal superpixels [57].

Thus, a 4D patch obtained from a temporal superpixel method can be defined by using the resulting segmentation label map $\mathcal{L} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$. For this, the 4D TSP patch is written as a function of the set of indices that belong to a determined region of the map, where the map has D labeled regions. Mathematically, the d-th 4D TSP subtensor $\mathcal{F}_d \in \mathbb{R}^{I_1^d \times I_2^d \times I_3^d \times I_4^d}$ of $\mathcal{F}$ can be expressed as

$$\mathcal{F}_d = f^d_{i_1^d i_2^d i_3^d i_4^d} = \{f^d_{i_j^d}\}_{j=1}^4, \tag{16}$$

where the subindices $i_j^d \in I_j^d$ belong to the d-th patch, where the patch has $I_1^d \times I_2^d$ spatial pixels, $I_3^d = I_3$ spectral bands, and a set of I_4^d temporal frames. Notice that TSP-based grouping yields an assembly of irregular shapes to model the underlying scene. In particular, the TSP estimation can be conducted by grouping pixels, for instance, based on the Euclidean distance of pixels [56] or following probabilistic models as in [57].

On the other hand, given that in a CSVS system the signal is unknown, several TSP subtensors can be obtained from the compressed projections $\mathcal{Y}$ as $\mathcal{Y}_d = y^d_{i_1^d i_2^d i_4^d}$, where the spectral dimension has been compressed. However, since the spatio-spectral information has been encoded, yielding a non-smooth signal, the TSP estimation from the measurements $\mathcal{Y}$ is unsuitable. This can result in inaccurate estimations of the TSPs given that the TSP estimation is based on the pixel intensity [5657]. Thus, in the next subsection, a strategy to overcome this limitation is presented. Figure 7 illustrates the concept of TSP over a video, where two objects appear across the temporal axis and, for illustration purposes, Figure 7(a) shows the trajectory of the objects along time, and in Figure 7(b) the objects are segmented and labeled for assembling the TSPs.

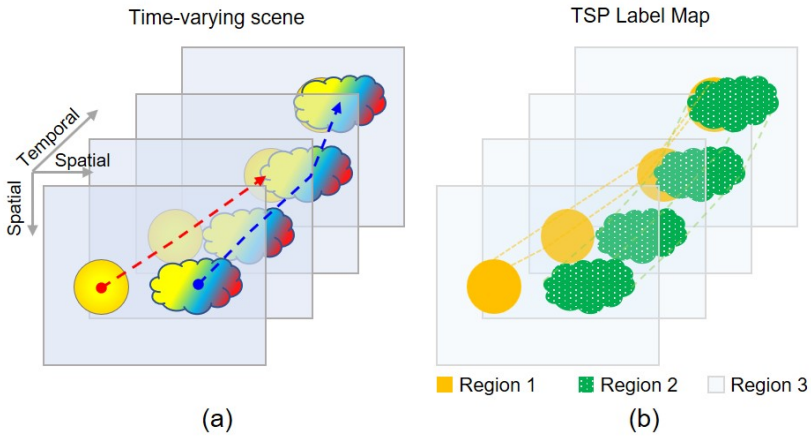


Figure 7. Illustration of the information across different instants of time. (a) The trajectory of the objects is drawn along the temporal axis. (b) The objects across time are segmented and labeled to assemble the temporal superpixels, where three regions are identified on the scene.

3.3.2. Grayscale Representation from the Compressed Video Aiming to exploit the spatio-temporal acquired measurements, a grayscale version of the scene can be formed to estimate the TSP and provide a suitable initialization of the sparse representation basis. In general for any CSVS architecture, the grayscale version of the video can be attained from the compressed measurements in Eq. (12) by estimating a preview of the scene from a coarse reconstruction, such as in [1], and then projecting all the spectral bands of the frame to form a grayscale frame. Specifically, from [1], the signal preview can be estimated by solving $\hat{\mathbf{f}}_{Low} = \mathbf{\Psi}\big(\text{argmin}_{\boldsymbol{\theta}_{Low}} ||\mathbf{y} - \mathbf{H}\mathbf{S}^T\mathbf{S}(\mathbf{\Psi}\boldsymbol{\theta})||_2^2 + \rho||\boldsymbol{\theta}||_1\big)$, where $\mathbf{S} \in \mathbb{R}^{(n/\kappa)\times n}$ is a spatial downsampling operator, with a dimensional reduction factor κ; the subscript $(\cdot)_{Low}$ denotes the low spatial resolution version, and then, the preview tensor is estimated as $\hat{\mathcal{F}} = \text{vec}^{-1}(\mathbf{S}^T\hat{\mathbf{f}}_{Low})$. Then, given the preview $\hat{\mathcal{F}}$ of the spectral video, the grayscale version of the i_4-th frame can be estimated as

$$\mathbf{Y}^G_{:,:,i_4} = \sum_{i_3=1}^{I_3} \hat{\mathcal{F}}_{:,:,i_3,i_4}, \tag{17}$$

for $i_4 = 1, ..., I_4$. Following this, the set of the I_4 frames containing the grayscale approximation of the scene can be denoted as $\mathcal{Y}^G = \{\mathbf{Y}^G_{:,:,1}, ..., \mathbf{Y}^G_{:,:,I_4}\}$, with $\mathcal{Y}^G \in \mathbb{R}^{I_1 \times I_2 \times I_4}$. Considering that the 3D-CASSI sensing model projects the scene along the spectral axis without shearing [13], a rapid grayscale approximation can be obtained under a specific encoding by adding two consecutive measurement frames. With this in mind, assume that the measurements $\mathcal{Y}$ are encoded by using time-varying boolean colored coded apertures [16], thus the rapid spatial approximation for the t-th frame can be obtained as

$$\mathbf{Y}^G_{:,:,t} = \mathcal{Y}_{:,:,t} + \mathcal{Y}_{:,:,t+1}, \tag{18}$$

for $t = 1, ..., I_4 - 1$, where for the (I_4)-th frame, the $(I_4 - 1)$-th spatial approximation is assigned, i.e. $\mathbf{Y}^G_{:,:,I_4} := \mathbf{Y}^G_{:,:,I_4-1}$. Therefore, when the video 3D-CASSI scheme is employed, the rapid approximation can be estimated from Eq. (18) rather than Eq. (17), avoiding the preview reconstruction. Subsequently, the obtained grayscale tensor $\mathcal{Y}^G$ is then employed to estimate the TSP such that the subindices of the segmented data can be used to segment the measurement tensor $\mathcal{Y}$.

3.3.3. Joint Dictionary and Recovery Problem Formulation The inverse problem to recover $\mathcal{F}$ from the measurements $\mathcal{Y}$ entails seeking a sparse solution of the underlying scene. The recovery problem for a fixed basis $\{\mathbf{\Psi}^{(z)}\}_{z=1}^4$ can be written as

$$\begin{aligned} &\underset{\mathcal{G} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}}{\text{minimize}} \left\| \mathcal{Y} - \mathcal{H}\big(\mathcal{G} \times_1 \mathbf{\Psi}^{(1)} \times_2 \mathbf{\Psi}^{(2)} \times_3 \mathbf{\Psi}^{(3)} \times_4 \mathbf{\Psi}^{(4)}\big) \right\|_F^2 \\ &\text{subject to } \; ||\text{vec}(\mathcal{G})||_1 \leq S, \end{aligned} \tag{19}$$

where the constant S denotes the sparsity level of the core tensor.

Let $\mathbf{U}^{(z)} \in \mathbb{R}^{I_z \times I_z}$, for $z = 1, ..., 4$, be the factor matrices that sparsify the core tensor

$\mathcal{G}$ [58], then the joint sparse transform and reconstruction estimation can be expressed as

$$\begin{aligned}\{\hat{\mathbf{U}}^{(z)}, \hat{\mathcal{G}}\} \in \underset{\substack{\{\mathbf{U}^{(z)}\}_{z=1}^{4}, \\ \mathcal{G}}}{\operatorname{argmin}} & \left\| \mathcal{Y} - \mathcal{H}\Big(\mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}\Big) \right\|_F^2 \\ \text{subject to } & ||\text{vec}(\mathcal{G})||_1 \leq S, \\ & \mathbf{U}^{(z)^T}\mathbf{U}^{(z)} = \mathbf{I}^{(z)},\ z = 1, .., 4,\end{aligned} \tag{20}$$

where $\mathcal{G} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ is the core tensor, and $\mathbf{I}^{(z)}$ is an identity matrix.

Thus, according to the proposed TSP-based reconstruction methodology, the measurements tensor $\mathcal{Y}$ is partitioned onto several 3D temporal superpixel subtensors. Then, the objective function in Eq. (20) can be rewritten as follows

$$\begin{aligned}\left\{\hat{\mathcal{G}}_d, \hat{\mathbf{U}}_d^{(z)}, \hat{\mathbf{U}}^{(3)}\right\} \in \underset{\substack{\mathcal{G}_d, \mathbf{U}^{(3)} \\ \{\mathbf{U}_d^{(z)}\}_{z=1}^{2,4}}}{\operatorname{argmin}} & \left\| \mathcal{Y}_d - \mathcal{H}_d\Big(\mathcal{G}_d \times_1 \mathbf{U}_d^{(1)} \times_2 ... \times_4 \mathbf{U}_d^{(4)}\Big) \right\|_F^2 \\ \text{subject to } & ||\text{vec}(\mathcal{G}_d)||_1 \leq S, \\ & \{\mathbf{U}_d^{(z)^T}\mathbf{U}_d^{(z)} = \mathbf{I}^{(z)}\}_{z=1,2,4}, \\ & \mathbf{U}^{(3)^T}\mathbf{U}^{(3)} = \mathbf{I}^{(3)},\end{aligned} \tag{21}$$

where $\mathcal{Y}_d = y_{i_1 i_2 i_4}^d$ is a temporal superpixel patch computed from the measurements, $\mathcal{H}_d$ is the CSVS sub-operator selected for the indices $i_1 i_2 i_3 i_4$ from the TSP, and the set of the unitary matrices $\mathbf{U}_d^{(z)}$ for $z = 1, 2, 4$ is computed for each patch (with the restriction that they must be orthogonal). Notice that for the third dimension of the data, the unitary matrix $\mathbf{U}^{(3)}$ is estimated without using the index d, denoting that the spectral information is not partitioned along that axis.

---

[58] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. “A multilinear singular value decomposition”. In: *SIAM journal on Matrix Analysis and Applications* 21.4 (2000), pp. 1253–1278.

3.4. Optimization Algorithm for the Basis Estimation and Signal Recovering

In this section, two algorithms are presented for the joint sparse-transform estimation and signal recovery processes: Algorithm 1, that summarizes the general steps of the proposed framework, and Algorithm 2, which presents the steps of the proposed algorithm based on block-coordinate descent (BCD) method to solve the problem in Eq. (20). The proposed BCD-based method is explained in general form, and then, it is used for each temporal patch from the main algorithm.

3.4.1. General Algorithm As discussed in subsection 3.3.3, even though the proposed framework is focused on the TSP-based transform, it is possible to solve the problem without partitioning the data, however, this implies much more processing time as it will be shown in Section 3.5. Algorithm 1 receives the parameter $\beta \in \{0, 1\}$ as input that selects the TSP-based processing ($\beta = 1$) or the full-data processing ($\beta = 0$), the measurements $\mathcal{Y}$, an initial 1D transform $\mathbf{\Psi}_0$ for the third dimension, and the approximated number of temporal superpixels D. Then, for the TSP-based processing (TSP-TenDL) when $\beta = 1$, the label map is estimated from the grayscale tensor $\mathcal{Y}^G$, taking into account the number of desired TSPs. In line 3, the $\mathbf{\Omega}(\cdot)$ operator represents the TSP estimation operation, where the set of irregular TSPs $\hat{\mathcal{Y}}^G$ and the label map $\mathcal{L}$ are obtained. Then, it is necessary to extract the TSP in a regular form to be processed, since TSP-based grouping generates irregular shapes. That operation is represented by $\mathbf{\Delta}(\cdot)$ in line 4 and 5, where each TSP patch is taken in a rectangular form for its processing, whose rectangular shape is given by the largest spatial size of a patch in that specific TSP (see Figure 6 'Regular shape TSPs Extraction' box for an intuitive illustration). As a result, the sets of regular-shape TSPs $\tilde{\mathcal{Y}} = \{\tilde{\mathcal{Y}}_d\}_{d=1}^D$ and $\tilde{\mathcal{Y}}^G = \{\tilde{\mathcal{Y}}_d^G\}_{d=1}^D$ are obtained, where $\tilde{\mathcal{Y}}_d = \{\mathcal{Y}_{i_1 i_2 i_4} | (i_1, i_2, i_4) \in \mathcal{L}\}$. After that, the JOINTESTIMATION algorithm, which is explained in detail in the next section, is run for each TSP patch.

When all the patches are recovered, these are merged using the MERGING function in line 9, which takes the initial indices from the labeled tensor and assigns the recovered patch to the specific indices for each patch. On the other hand, for the full-data processing (TenDL) when $\beta = 0$, the JOINTESTIMATION algorithm is run over the high-resolution compressed video.

3.4.2. BCD-based Formulation In general form, the problem in Eq. (20) can be efficiently solved using an alternating minimization approach. More precisely, setting $\mathcal{F} = [\![\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}]\!]$[59], Eq. (20) can be reformulated as

$$\begin{aligned} \underset{\substack{\{\mathbf{U}^{(z)}\}_{z=1}^{4}, \\ \mathcal{G}, \mathcal{F}}}{\text{minimize}} & \left\|\mathcal{Y} - \mathcal{H}(\mathcal{F})\right\|_F^2 + \lambda ||\text{vec}(\mathcal{G})||_1, \\ \text{subject to } & \mathcal{F} = [\![\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}]\!] \\ & {\mathbf{U}^{(z)}}^T \mathbf{U}^{(z)} = \mathbf{I}^{(z)}, \ z = 1, .., 4, \end{aligned} \tag{22}$$

where the variable $\mathcal{F}$ is introduced and $\lambda > 0$. Thus, the augmented Lagrangian of Eq. (22) can be written as

$$\begin{aligned} \mathcal{L}_A(\mathcal{G}, \mathcal{F}, \{\mathbf{U}^{(z)}\}_{z=1}^{4}, \mathcal{Q}) &= \left\|\mathcal{Y} - \mathcal{H}(\mathcal{F})\right\|_F^2 + \lambda ||\text{vec}(\mathcal{G})||_1 \\ + (\lambda/2) \left\|\mathcal{F} - [\![\mathcal{G}; \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}]\!] + \mathcal{Q}\right\|_F^2 &+ \sum_{z=1}^{4} \mathcal{I}_{\mathcal{U}}(\mathbf{U}^{(z)}), \end{aligned} \tag{23}$$

where $\mathcal{Q}$ is the Lagrange multiplier and $\mathcal{I}_{\mathcal{U}}(\mathbf{U}^{(z)})$ is an indicator function defined as

$$\mathcal{I}_{\mathcal{U}}(\mathbf{U}^{(z)}) = \begin{cases} 1, & \text{if } \mathbf{U}^{(z)} \in \mathcal{U} \\ 0, & otherwise \end{cases}, \tag{24}$$

---

[59] Alternative notation for the tensor decomposition from Eq. (10).

where $\mathcal{U} = \left\{\mathbf{U} \in \mathbb{R}^{I_z \times I_z} | \mathbf{U}^T\mathbf{U} = \mathbf{I}\right\}$, $z = 1, ..., 4$.

Equation (40) can be iteratively solved by the following three steps, where each variable is updated while the others are fixed:

$\tilde{\mathcal{F}}^{k+1}$ sub-problem:

$$\tilde{\mathcal{F}}^{k+1} \in \underset{\mathcal{F}}{\operatorname{argmin}} \frac{\lambda}{2} \left\| \mathcal{F}_k - [\![\mathcal{G}_k; \mathbf{U}_k^{(1)}, \mathbf{U}_k^{(2)}, \mathbf{U}_k^{(3)}, \mathbf{U}_k^{(4)}]\!] + \mathcal{Q}_k \right\|_F^2 + \frac{1}{2} \left\| \mathcal{Y} - \mathcal{H}(\mathcal{F}_k) \right\|_F^2. \quad (25)$$

The subproblem in Eq. (25) can be solved as the linear problem given by

$$\begin{aligned} \tilde{\mathbf{f}} =& \lambda \mathrm{vec}([\![\mathcal{G}_k; \mathbf{U}_k^{(1)}, \mathbf{U}_k^{(2)}, \mathbf{U}_k^{(3)}, \mathbf{U}_k^{(4)}]\!]) + \mathbf{H}^T(\mathrm{vec}(\mathcal{Y})) \\ =& \lambda \mathbf{f} + \mathbf{H}^T(\mathbf{H}\mathbf{f}), \end{aligned} \quad (26)$$

where $\mathbf{f}$ is zero-initialized, $\mathbf{H}$ is the sensing matrix that encloses the projection operation performed by the camera, $\mathbf{H}^T$ denotes the transpose operation for $\mathbf{H}$, and $\tilde{\mathbf{f}}$ can be found from the conjugate gradient (CG) method reported in [60] (Section 2.3.1, Fig. (2.5)). Note that the preconditioner in the CG algorithm is fixed to an identity matrix.

$\tilde{\mathcal{G}}^{k+1}$ sub-problem:

$$\tilde{\mathcal{G}}^{k+1} \in \underset{\mathcal{G}}{\operatorname{argmin}} \frac{\lambda}{2} \left\| \mathcal{F}_{k+1} - [\![\mathcal{G}_k; \mathbf{U}_k^{(1)}, \mathbf{U}_k^{(2)}, \mathbf{U}_k^{(3)}, \mathbf{U}_k^{(4)}]\!] + \mathcal{Q}_k \right\|_F^2 + \tau ||\mathrm{vec}(\mathcal{G}_k)||_1, \quad (27)$$

where this subproblem-update is a proximal operator evaluation, whose closed-form

---

[60] Richard Barrett et al. *Templates for the solution of linear systems: building blocks for iterative methods*. Vol. 43. Siam, 1994.

solution can be obtained from the well-known soft shrinkage operator given by

$$\tilde{\mathcal{G}}^{k+1} = \text{vec}^{-1}\{\mathcal{S}_{\lambda/\tau}\big(\text{vec}(\mathcal{F}^{k+1} + \mathcal{G}^{k}), \lambda/\tau\big)\}, \tag{28}$$

with $\mathcal{S}_{\lambda/\tau}(\mathbf{x}, \beta) := \text{sgn}(\mathbf{x})\max(|\mathbf{x}| - \beta, 0)$ as the soft thresholding operator, and $\lambda, \tau > 0$ are regularization parameters.

Algorithm 1: TSP-TenDL / TenDL Main Algorithm

Input: $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_4}$;
$\mathbf{\Psi}_0$: Initial 1D transform (e.g. DCT);
D : desired number of TSP (default $D = 10$);
$\beta \in \{0, 1\}$: 1 for TSP-TenDL processing,
0 for TenDL processing;

Initialize: $\mathbf{U}^{(3)} \leftarrow \mathbf{\Psi}_0$; Compute $\mathcal{Y}^G$ via Eq. (18)
if $\beta = 1$ **then** ▷ TSP-TenDL approach
  $\{\hat{\mathcal{Y}}^G, \mathcal{L}(i_1, i_2, i_4)\} \leftarrow \mathbf{\Omega}(\mathcal{Y}^G, D)$ ▷ TSP guess
  $\tilde{\mathcal{Y}} \leftarrow \mathbf{\Delta}(\mathcal{Y}, \mathcal{L}(i_1, i_2, i_4))$ ▷ Patch extraction
  $\tilde{\mathcal{Y}}^G \leftarrow \mathbf{\Delta}(\hat{\mathcal{Y}}^G, \mathcal{L}(i_1, i_2, i_4))$
  for $d = 1$ **to** D **do**
    $\tilde{\mathcal{F}}_d \leftarrow \text{JOINTESTIMATION}\big(\tilde{\mathcal{Y}}_d, \tilde{\mathcal{Y}}_d^G, \mathbf{U}^{(3)}\big)$
  $\hat{\mathcal{F}} \leftarrow \text{MERGING}\big(\{\tilde{\mathcal{F}}_d\}_{d=1}^{D}, \mathcal{L}(i_1, i_2, i_4)\big)$
else ▷ TenDL approach (full-image)
  $\hat{\mathcal{F}} \leftarrow \text{JOINTESTIMATION}\big(\mathcal{Y}, \mathcal{Y}^G, \mathbf{U}^{(3)}\big)$

Output: Recovered Spectral Video $\hat{\mathcal{F}} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$

Online Transform Refinement $\tilde{\mathbf{U}}_{k+1}^{(z)}$ for $z = 1, ..., 4$ After estimating $\tilde{\mathcal{F}}^{k+1}$ and $\tilde{\mathcal{G}}^{k+1}$, the sparse transform is refined for each dimension as

$$\tilde{\mathbf{U}}_{k+1}^{(z)} \in \underset{\{\mathbf{U}^{(z)}\}_{z=1}^{4}}{\text{argmin}} \frac{\lambda}{2} \left\| \mathcal{F}_{k+1} - [\![\mathcal{G}_{k+1}; \mathbf{U}_k^{(1)}, \mathbf{U}_k^{(2)}, \mathbf{U}_k^{(3)}, \mathbf{U}_k^{(4)}]\!] + \mathcal{Q}_k \right\|_F^2 + \mathcal{I}_z\big(\mathbf{U}^{(z)}\big). \tag{29}$$

Note that this subproblem can be alternatively written in a general form for the mode-z as

$$\tilde{\mathbf{U}}_{k+1}^{(z)} \in \underset{\mathbf{U}^{(z)}}{\operatorname{argmin}} \frac{\lambda}{2}||\mathbf{F}_{(z)} - \mathbf{U}^{(z)}\mathbf{G}_{(z)}(\mathbf{U}^{(Z)} \otimes ...\mathbf{U}^{(z-1)} \otimes \mathbf{U}^{(z+1)} \otimes \mathbf{U}^{(1)})^T + \mathbf{Q}_{(z)}||_F^2 + \mathcal{I}_z\big(\mathbf{U}^{(z)}\big), \tag{30}$$

for $z = 1, ..., Z$, where $Z = 4$, are the modes of the tensor, such that each matrix $\mathbf{U}^{(z)}$ can be refined by using the mode-z of the Eq. (29). Problem in Eq. (30) is known as the Orthogonal Procrustes problem [61], whose closed-form solution is given by

$$\tilde{\mathbf{U}}_{k+1}^{(z)} = \mathbf{S}\mathbf{V}^T, \tag{31}$$

where $\mathbf{S}$ and $\mathbf{V}^T$ are obtained from the matrix-based singular value decomposition (SVD) of the factor $\big(\mathbf{F}_{(z)} + \mathbf{Q}_{(z)}\big)\big(\mathbf{G}_{(z)}\big(\mathbf{U}^{(Z)} \otimes ...\mathbf{U}^{(z-1)} \otimes \mathbf{U}^{(z+1)}... \otimes \mathbf{U}^{(1)}\big)^T\big)^T$, i.e.

$$\begin{aligned}\mathbf{S}\mathbf{\Sigma}\mathbf{V}^T = &\operatorname{SVD}\big((\mathbf{F}_{(z)} + \mathbf{Q}_{(z)})\big(\mathbf{G}_{(z)}(\mathbf{U}^{(Z)} \otimes ... \\ &\otimes\mathbf{U}^{(z-1)} \otimes \mathbf{U}^{(z+1)}... \otimes \mathbf{U}^{(1)})^T\big)^T\big).\end{aligned} \tag{32}$$

Thus, the sparse transform update is reduced to the computation of Eq. (31) for $z = 1, 2, 3, 4$. And finally, the multiplier is updated as

$$\tilde{\mathcal{Q}}_{k+1} = \mathcal{Q}_k + \tilde{\mathcal{F}}_{k+1} - [\![\tilde{\mathcal{G}}_{k+1}; \tilde{\mathbf{U}}_{k+1}^{(1)}, \tilde{\mathbf{U}}_{k+1}^{(2)}, \tilde{\mathbf{U}}_{k+1}^{(3)}, \tilde{\mathbf{U}}_{k+1}^{(4)}]\!]. \tag{33}$$

The main steps of the BCD-based optimization are summarized in Algorithm 2, where the inputs are the measurements tensor $\mathcal{Y}$, the grayscale approximation $\mathcal{Y}^G$ and the initial guess for $\mathbf{U}^{(3)}$. Algorithm 2 starts with zero-initialization of the tensors

---

[61] Hui Zou, Trevor Hastie, and Robert Tibshirani. “Sparse principal component analysis”. In: *Journal of computational and graphical statistics* 15.2 (2006), pp. 265–286.

$\mathcal{F}_k, \mathcal{G}_k, \mathcal{Q}_k$ according to the size of the input $\mathcal{Y}$ and the number of spectral bands I_3. In line 4, a suitable initialization of the sparse transform is performed by using the grayscale approximation from the multilinear SVD operation (MLSVD), since the MLSVD decomposition asserts that the obtained factor matrices $\mathbf{U}^{(z)} \in \mathbb{R}^{I_z \times I_z}$ are orthogonal [58]. Then, while some stopping criterion is not satisfied, such as the number of iterations or the tolerance error, the BCD steps are computed. Finally, the recovered tensor $\hat{\mathcal{F}}$ is obtained from the estimated core tensor and the learned basis.

Algorithm 2: Joint Sparse Basis and Signal Estimation

1 **Function** JOINTESTIMATION$\big(\mathcal{Y}, \mathcal{Y}^G, \tilde{\mathbf{U}}_k^{(3)}\big)$
2 **Initialize:** $\{J_1, J_2, J_3\} \leftarrow$ size of $\mathcal{Y}$
3 $\{\mathcal{F}_k, \mathcal{G}_k, \mathcal{Q}_k\} = \mathbf{0} \in \mathbb{R}^{J_1 \times J_2 \times I_3 \times J_4}$
4 $[\![\tilde{\mathbf{U}}_k^{(1)}, \tilde{\mathbf{U}}_k^{(2)}, \tilde{\mathbf{U}}_k^{(4)}]\!] \leftarrow$ MLSVD$\big(\mathcal{Y}^G\big)$, $k = 0$;
5 **while** *some stop criterion is not satisfied* **do**
6 Update $\tilde{\mathcal{F}}^{k+1}$ by solving Eq. (25)
7 Update $\tilde{\mathcal{G}}^{k+1}$ from Eq. (28)
8 Update $\tilde{\mathbf{U}}_{k+1}^{(1)}$, $\tilde{\mathbf{U}}_{k+1}^{(2)}$, $\tilde{\mathbf{U}}_{k+1}^{(3)}$ and $\tilde{\mathbf{U}}_{k+1}^{(4)}$ via Eq. (31)
9 Update the multiplier $\tilde{\mathcal{Q}}^k$ from Eq. (33)
10 $k = k + 1$
11 $\check{\mathcal{F}} = [\![\tilde{\mathcal{G}}_k; \tilde{\mathbf{U}}_k^{(1)}, \tilde{\mathbf{U}}_k^{(2)}, \tilde{\mathbf{U}}_k^{(3)}, \tilde{\mathbf{U}}_k^{(4)}]\!]$
12 **return** $\check{\mathcal{F}}$

3.4.3. Complexity Analysis In previous works on CSVS, vector-based algorithms such as the gradient projection for sparse reconstruction (GPSR)[62] or the sparse reconstruction by separable approximation (SpaRSA) [63] have been used

[62] Mário AT Figueiredo, Robert D Nowak, and Stephen J Wright. "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems". In: *IEEE Journal of selected topics in signal processing* 1.4 (2007), pp. 586–597.

[63] Stephen J Wright, Robert D Nowak, and Mário AT Figueiredo. "Sparse reconstruction by separable approximation". In: *IEEE Transactions on Signal Processing* 57.7 (2009), pp. 2479–2493.

to recover the spectral video [127]. In particular, to reconstruct the 4D tensor $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$, a vector-based algorithm leads to a computational complexity $\mathcal{O}\big((I_1 I_2 I_3 I_4)^{\alpha}\big)$ per iteration, where the exponent $\alpha > 1$ is empirically estimated: for SpaRSA $\alpha = 1.05$ and GSPR $\alpha = 1.06$[63]. It can be noticed that if the size of the video increases, the complexity of the problem also increases.

On the other hand, the computational complexity of the proposed approach is mainly given by the updates of the tensors $\tilde{\mathcal{F}}^{k+1}$ and $\tilde{\mathcal{G}}^{k+1}$ from the MLSVD estimation and basic tensor operations. These processes demand a computational cost of $\mathcal{O}(J_1 J_2 I_3 J_4) + \mathcal{O}(\min(J_n \prod_{n \neq m} J_m^2, J_n^2 \prod_{n \neq m} J_m))$, where, if the process is performed by TSPs, $J_n \ll I_n$, with J_n denoting the size of the largest temporal patch from the patches set, for $n, m = \{1, 2, 3, 4\}$. But, if the procedure is performed by full-video processing, then $J_n = I_n$, which entails the increasing of computation costs since MLSVD computation cost increases as much as the scale of the tensor.

Hence, the resulting cost for the JOINTESTIMATION step in Line 6-7 of Algorithm 1 is $\mathcal{O}(\frac{D}{C}(J_1 J_2 I_3 J_4 + \min(J_n \prod_{n \neq m} J_m^2, J_n^2 \prod_{n \neq m} J_m)))$, for D TSPs, where C is the number of available cores for parallel processing. For the case of full-video processing, Line 10 of Algorithm 1, the cost is $\mathcal{O}(I_1 I_2 I_3 I_4 + \min(I_n \prod_{n \neq m} I_m^2, I_n^2 \prod_{n \neq m} I_m))$.

3.5. Simulations and Results

Numerical experiments over three spectral videos were carried out by simulating the set of compressive measurements of Eq. (13) to analyze the performance of the proposed sparse transform learning and recovery approach. The first and second datasets are cropped sections of the scene in [30], named in this Chapter as Video 1 and 2, respectively. The third dataset is a real sequence[64] of spectral images acquired in the Optics Lab of the High Dimensional Signal Processing (HDSP) re-

[64] The video sequence dataset can be made available upon email request to: henarfu@uis.edu.co

search group at Universidad Industrial de Santander, named Video 3. Table 2 shows the dimensions of all spectral videos.

Table 2. Size of the spectral videos used for simulations

	Spatial pixels		**Spectral Bands**	**Number of frames**
Size	I_1	I_2	I_3	I_4
Video 1	128	128	8	8
Video 2	256	256	8	32
Video 3	128	128	24	16

The CSVS system employed for the simulations was the video 3D-CASSI. A set of time-varying random boolean colored coded apertures was used for all the experiments, in particular, the entries of these patterns are realizations of a Bernoulli random variable with parameter $p = 0.5$. Each boolean pattern is generated for two consecutive frames such that the sum of the patterns in the ensemble along each spatial coordinate is equal to a constant c. For the grayscale tensor estimation, the strategy presented in Eq. (18) was adopted.

On the other hand, the simple linear iterative clustering algorithm (SLIC)[65] was used to generate the temporal superpixels, which generates the TSPs based on k-means clustering and the Euclidean distance [56]. This algorithm was chosen for its simplicity and speedy performance, however, other implementations can be employed for the patch segmentation and extraction such as that proposed in [57], since the TSP algorithm is just used for the grouping of coherent pixels along spatio-temporal axes. The proposed TSP-TenDL and TenDL approaches are compared with a fixed analysis basis on the tensor-form problem in Eq. (20), an offline dictionary-learning-based approach, and the traditional vector-form inverse problem for CSVS. More specifically, by denoting $\mathbf{\Psi}^W$ as an 1D Wavelet-basis and $\mathbf{\Psi}^D$ as an 1D-Discrete Cosine

[65] Implementation available online at `https://www.epfl.ch/labs/ivrl/research/slic-superpixels/`

basis (DCT), the fixed analysis basis in Eq. (20) is set as follows: $\mathbf{U}^{(1)} = \mathbf{\Psi}^W$ and $\mathbf{U}^{(2)} = \mathbf{\Psi}^W$ for the spatial dimensions, $\mathbf{U}^{(3)} = \mathbf{\Psi}^D$ and $\mathbf{U}^{(4)} = \mathbf{\Psi}^D$ for the spectral and the temporal dimensions [271], named as 'WWDD-TenD', given the initial letters of the used bases the sparsity analysis between the WWDD transform and the proposed transform is presented in the Annex 1. For the case of the traditional vector-form inverse problem in CSVS, the basis $\mathbf{\Psi}^{4D}$ is fixed as the Kronecker product given by $\mathbf{\Psi}^{4D} = \mathbf{\Psi}^W \otimes \mathbf{\Psi}^W \otimes \mathbf{\Psi}^D \otimes \mathbf{\Psi}^D$, here referred to 'WWDD-Vec', since the vector-form recovery is used. On the other hand, for the offline dictionary-learning approach, the simultaneous spectral sparse (3S) model proposed in [2] was employed with an acceleration-rate parameter $K = 1$, for a fair comparison with the proposed method (see [2] Eq.(1)-(3)). To get the side information, which is used for the dictionary learning step in the 3S model, were considered two scenarios: an additional camera (Panchromatic camera) and the grayscale approximation presented in section 3.3.2, named as 3SDL-Vec and 3SDLg-Vec, respectively. The patches for the dictionary learning in the 3SDL-Vec and 3SDLg-Vec methods are set based on the obtained results in [2] as $6 \times 6 \times K$, where MN patches for the KSVD-based learning were extracted. Also, since there are three regularization parameters in the model, it was performed an exhaustive search of these parameters to get the higher PSNR in the signal recovery.

In summary, the methods to be compared are as follows: the proposed TSP-TenDL and TenDL models, the proposed tensor-based modeling with the fixed basis: WWDD-TenD, the vector-form recovery with the fixed basis: WWDD-Vec, the dictionary-learning-based method with simultaneous sparse model using a PanChromatic camera: 3SDL-Vec, and using the grayscale approximation: 3SDLg-Vec. The peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) metrics are used to evaluate the image quality of the reconstructions, and for the spectral reconstruction assessment, the root mean squared error (RMSE) is employed. The PSNR,

given in decibels (dB), is related to the mean squared error (MSE) as $10\log_{10}(MAX^2/MSE)$, where MAX is the maximum possible value of an image pixel. Meanwhile, the SSIM measures the similarity between two images, taking values from 0 to 1, 1 being the value obtained when two identical images are compared. For the RMSE metric, the smaller the RMSE values, the better the reconstruction results.

3.5.1. Comparison of the Recovery Results Several experiments were performed to evaluate the accuracy of the reconstructions with the proposed recovery approach[66]. Subsection 3.5.3 analyzes the impact of the number of TSPs in the reconstruction results and in the computation time for the TSP-TenDL method. In this subsection, just for comparison purposes, the number of TSP in the TSP-TenDL method is fixed to $D = 10$, which is the default value in Algorithm 1. Specifically, this value was chosen as default since it provides a trade-off between high PSNR reconstruction and short processing time. Figure 8 shows an RGB representation of the original and reconstructions of frames 1, 5 and 10 of each video from each method, where the quality of reconstruction in terms of the average PSNR is also shown. Note that the obtained reconstructions from the TSP-TenDL and TenDL outperform the other approaches up to 7 dB, as can be seen in the reconstructed video 1. Annex 2 depicts the spectral bands from each reconstruction method and each spectral video.

To illustrate the spectral accuracy of the proposed approach, the spectral signature (i.e. a spatial point along the spectral bands) of two spatial points in the frames 10, 20, and 30 of the spectral video 2 are pictured for: a static zone in Figure 9, and a dynamic zone in Figure 10. The RMSE of each profile is provided in the legend of each picture. For the point P1 in the static zone, despite there is no motion along

[66] All simulations were performed in a desktop architecture with an Intel(R) Xeon(R) CPU E5-1603 v3 @ 2.80 GHz processor, 128 GB RAM.

Figure 8. RGB profile of the originals (1st column) and the reconstructed frames 1, 5 and 10 of each video by using the WWDD-Vec (2nd column), the WWDD-TenD (3rd column), the 3SDL-Vec (4th column), the 3SDLg-Vec (5th column), the TenDL (6th column) and the TSP-TenDL (7th column) methods. PSNR is shown for each selected frame.

the temporal dimension, the proposed bases TenDL and TSP-TenDL produce more accurate reconstruction than the other approaches. On the other hand, for the point P2 in the dynamic zone, the reconstructed spectral signatures using the TSP-TenDL and TenDL methods (discontinued line -$\diamond$ and -$\star$) are closer to the original spectrum in comparison to the profiles obtained with the other approaches.

Table 3 summarizes the obtained results in terms of average PSNR, SSIM and RMSE. In particular, the RMSE metric is estimated on the spectral axis to evaluate the accuracy in the spectral reconstruction, i.e. the values are obtained by computing the averaged error of each spectral signature. Table 3 and Figures 8-10 demonstrate that reconstruction results from the TSP-TenDL and TenDL approaches exhibit lower RMSE values and higher PSNR-SSIM values resulting in higher accuracy in comparison with the other methods, where the best (highest and lowest) values are in bold and underlined. Note that, in general, the best performance is achieved by the TSP-

TenDL method. Regarding the performance of the 3SDL-Vec approach compared to the 3SDLg-Vec, it can be noticed that by using the proposed grayscale approximation a slightly similar accurate reconstruction can be achieved, where the additional camera can be replaced by a grayscale version from measurements. Considering this, it is possible to disregard the additional camera and compute the grayscale from measurements to obtain the initial dictionary for the offline-learning model, bearing in mind that a slightly lower recovery performance is obtained when the grayscale approximation is used. In addition, Table 4 provides the computation time from the different approaches when $D = 10$ on the TSP-TenDL method. In these results, the TSP-TenDL exhibits a lower computation time in comparison with the fixed-basis-based approaches, where speedups of up to 1.6$\times$ with respect to the WWDD-Vec and 6.6$\times$ with respect to the WWDD-TenD are achieved. On the other hand, even though the 3SDL-Vec method takes less time for the video 3 recovery, the reconstruction accuracy is lower than the proposed method.

Table 3. Mean of PSNR, SSIM and RMSE (on the spectral axis) of the Reconstructed Videos using the Different Approaches. The standard deviation is shown into the brackets.

PSNR (dB) - **SSIM**						
	Video 1		**Video 2**		**Video 3**	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
WWDD-Vec	31.26 (0.70)	0.933 (0.004)	30.31 (0.49)	0.923 (0.006)	30.70 (2.8)	0.851 (0.017)
WWDD-TenD	30.31 (0.66)	0.931 (0.005)	30.56 (0.40)	0.930 (0.005)	32.08 (2.6)	0.843 (0.010)
3SDL-Vec	29.84 (0.73)	0.915 (0.008)	27.47 (0.30)	0.854 (0.007)	30.59 (2.5)	0.832 (0.022)
3SDLg-Vec	29.30 (0.73)	0.907 (0.009)	26.64 (0.30)	0.835 (0.008)	30.35 (2.5)	0.823 (0.023)
TenDL	**35.61 (0.66)**	**0.978 (0.003)**	**33.97** (0.16)	**0.962** (0.002)	**34.62** (1.97)	**0.907 (0.021)**
TSP-TenDL	**37.17 (0.67)**	**0.980 (0.004)**	**33.44 (0.23)**	**0.960** (0.00)	**34.77** (1.8)	**0.915 (0.021)**
RMSE						
Method	**Video 1**		**Video 2**		**Video 3**	
WWDD-Vec	0.0206 (0.0026)		0.0299 (0.0019)		0.0221 (0.0036)	
WWDD-TenD	0.0228 (0.0028)		0.0241 (0.0016)		0.0196 (0.0024)	
3SDL-Vec	0.0252 (0.0038)		0.0371 (0.0019)		0.0229 (0.0034)	
3SDLg-Vec	0.0269 (0.0040)		0.0411 (0.0020)		0.0236 (0.0035)	
TenDL	**0.0136 (0.0014)**		**0.0175** (0.0006)		**0.0137 (0.0022)**	
TSP-TenDL	**0.0110 (0.0012)**		**0.0176** (0.0007)		**0.0130 (0.0023)**	

Table 4. Computation time from the different approaches

	Time (seconds)					
	WWDD-Vec	WWDD-TenD	3SDL-Vec	3SDLg-Vec	TenDL	TSP-TenDL
Video 1	100.1	405	111.3	101.6	326.8	**61.7**
Video 2	1275.2	5955	959.4	1288.8	6137.5	**954.1**
Video 3	828.8	3411.6	**350.5**	373.1	3393.8	519.4

3.5.2. Convergence of the Proposed Algorithm To show the convergence of the proposed recovery algorithm, each iteration of the objective function in Eq. (22) is evaluated using the test videos. In addition, the PSNR from the estimated signal is computed on each iteration. Figure 11 shows the convergence of the algorithm for 300 iterations. Note that the curves converge to a minimum point after some iterations, and PSNR values become constant after the iteration number 200.

3.5.3. Impact of the number of TSPs in the Reconstruction Results Video 3 was reconstructed from the TSP-TenDL approach by selecting different number of TSPs. Particularly, video 3 was selected for this analysis since both its spectral and temporal resolution are higher than the other videos, attributes that show the usefulness of the proposed approach. Figure 12 illustrates the performance of the reconstruction results when the number of the desired TSPs goes from 5 up to 220, where the zero-position in the plot is referred to the reconstruction from the TenDL method.

It can be noticed in Figure 12(a) that increasing the number of TSP leads to an improvement in the PSNR and SSIM values. However, increasing the number of TSP can entail an increase in the complexity of the algorithm since the complexity also depends on the number of TSPs. In spite of this, computation time is still much lower than that of the TenDL method. Figure 12(b) shows the computation time of reconstructions in seconds by using the proposed approaches when it is assumed that the same number of cores in the CPU as the number of TSPs ($-\circ$ line) are

available, and when there are 28 cores in the CPU working in parallel ($-\triangleright$ line). Notice in the zoomed section on Figure 12(b) that the computing time is reduced when the video is processed by patches instead of the full-data, where for the TenDL method (zero-position on the plot) the time reaches around 3400 seconds (≈ 57 min) to reconstruct the video 3, meanwhile the TSP-TenDL in parallel takes less than 520 seconds (≈ 9 min) to obtain the reconstruction result from just 10 TSPs. That is, a speedup of 6.3$\times$ for the TSP-TenDL approach. As can be seen from the obtained numerical results, the TSP-TenDL approach not only exhibits higher accuracy in the reconstruction results but less computation time.

3.6. Conclusions

In this Chapter, a framework for online sparse transform learning and reconstruction procedures that exploits the high-order structure and the compressed measurements of spectral video tensors was introduced. The framework is based on the tensor decomposition and the temporal superpixel processing to fully exploit the high signal correlation. In particular, the video-rate spatial-spectral coded compressive spectral imager (video 3D-CASSI) was considered for the study. Numerical experiments over different spectral videos show that the proposed approach improves the spatial, spectral and temporal accuracy of the reconstructions when compared to analytical and offline-learned bases. In particular, gains of up to 7 dB of PSNR and 0.1 of SSIM are obtained with respect to the state-of-the-art recovery methods and the tensor-based recovery with the fixed basis. In addition, a speedup from 1.6$\times$ up to 6.6$\times$ is achieved compared with state-of-the-art counterparts. It is important to highlight that the obtained results were run on a CPU architecture, where the acceleration and performance are limited in comparison with GPU or specialized architectures for parallel programming. Thus, more sophisticated devices can be used to obtain more accelerated reconstruction results offering a promising future for the study of

real-time applications on compressive spectral videos.

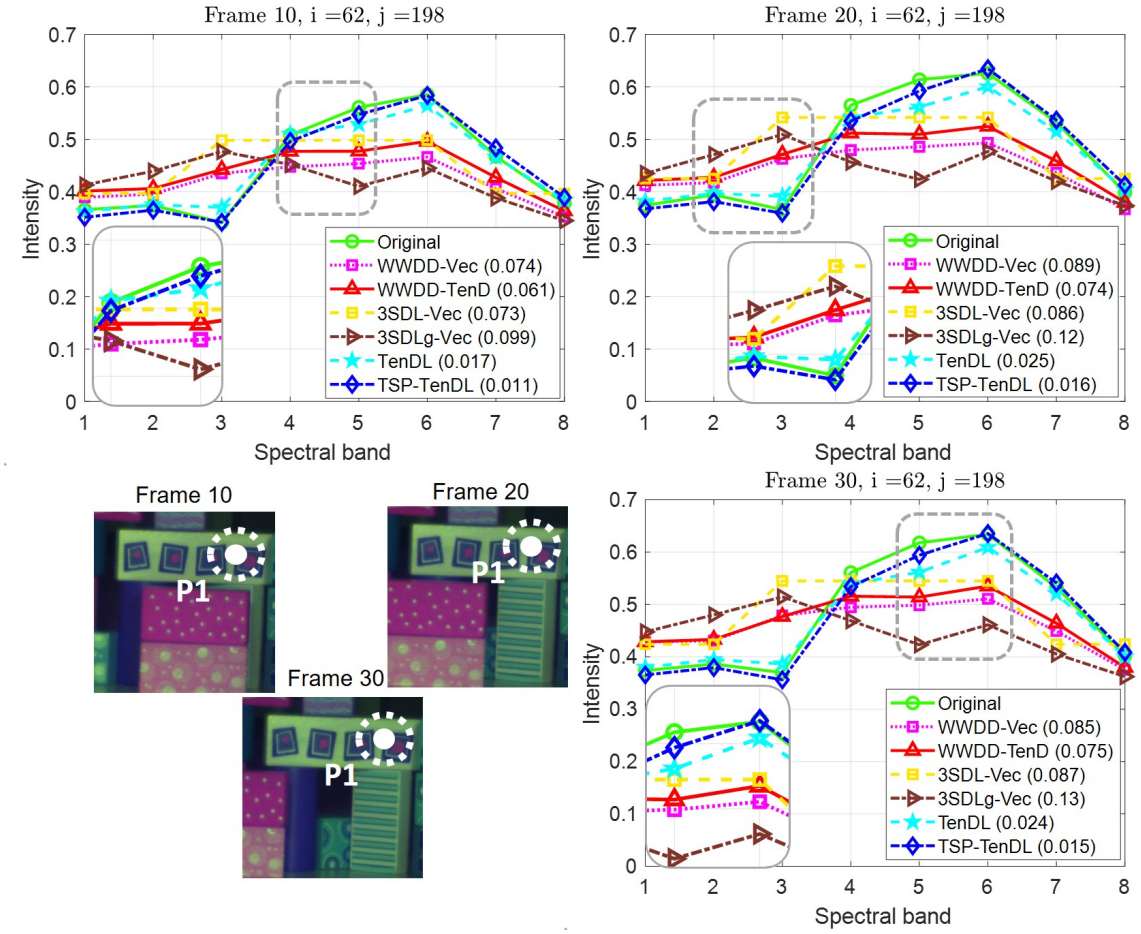


Figure 9. Spectral signature comparison for the different approaches in the point P1 on a *static zone* of the video 2 across the frames 10, 20 and 30, where the RMSE of each profile is shown in the legend. The zoomed section shows that the TenDL and TSP-TenDL methods provide a closer spectral response to the original than the other methods.

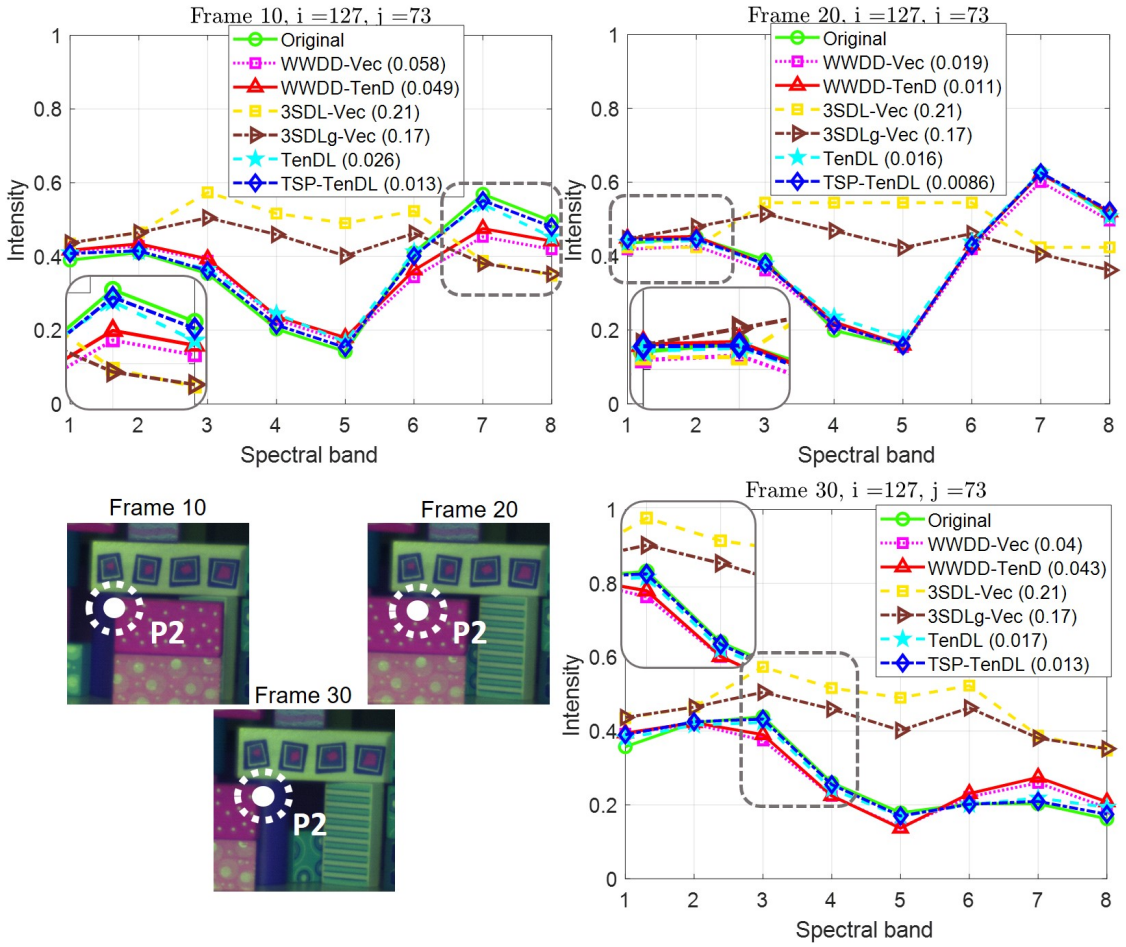


Figure 10. Spectral signature comparison for the different approaches in the point P2 on a *dynamic zone* of the video 2 across the frames 10, 20 and 30, where the RMSE of each profile is shown in the legend.

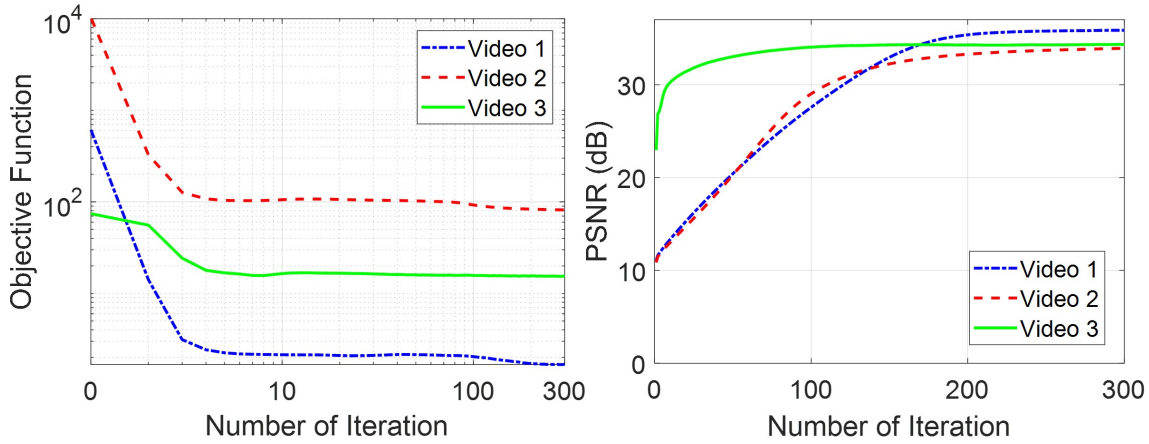


Figure 11. Verification of the convergence of the proposed method for each video from the objective function evaluation (plotted in logarithmic scale) and the progressive PSNR reconstruction for 300 iterations.

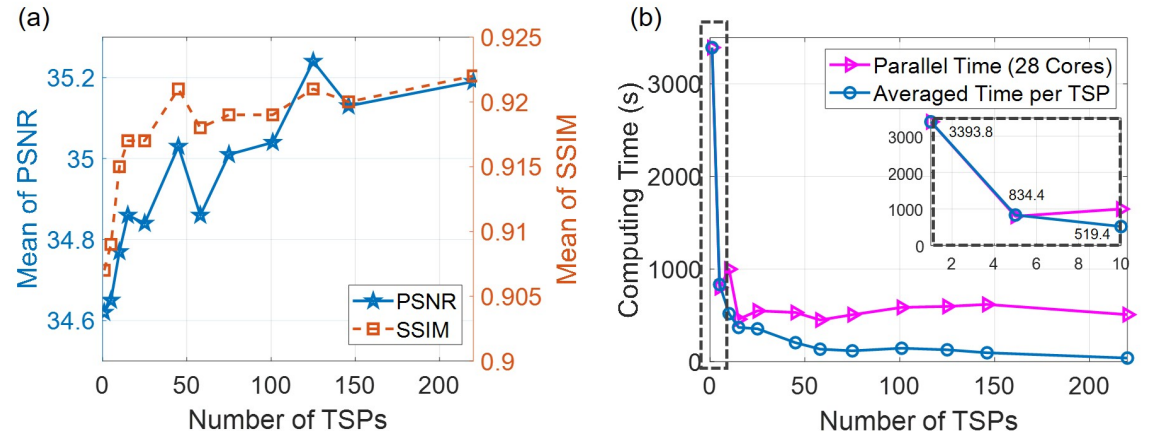


Figure 12. Impact of the number of TSPs in the reconstruction process and computing time using the video 3. Zero-position on the plot refers to the result from the TenDL method. (a) PSNR (left axis) and SSIM (right axis) when the number of TSP grows up to 220. (b) Computing time when the number of cores in CPU is the same as the number of TSP ($-\circ$ line) and when 28 cores are used working in parallel ($-\triangleright$ line).

4. HIGHER-ORDER TENSOR SPARSE REPRESENTATION FOR VIDEO-CASSI RECONSTRUCTION

In this Chapter, the methodology previously presented in Chapter 3 is extended and adapted to the video-CASSI system in the full-data processing mode (i.e., without TSP patches), considering that the spectral-spatial information is shifted and mixed at the encoding step. The methodology is evaluated on measurements with different levels of noise[67]

4.1. Video-rate CASSI Model

Let $\mathcal{T}$ be the four-dimensional (4D) time-varying colored coded aperture in discrete form, $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ the fourth-order tensor representation of the discrete spectral video, where $I_1 \times I_2$ represents the spatial size, I_3 the spectral bands, and I_4 the number of video frames. Then, the video C-CASSI acquisition procedure of the i_4-th frame can be expressed as

$$(\mathcal{Y}^{i_4})_{i_1,i_2} = \sum_{i_3=0}^{I_3-1} (\mathcal{F}_{i_3}^{i_4})_{i_1,i_2-i_3} \circ (\mathcal{T}_{i_3}^{i_4})_{i_1,i_2-i_3} + (\mathcal{W}^{i_4})_{i_1,i_2} \tag{34}$$

for $i_4 = 0, ..., I_4 - 1$, $i_1 = 0, ..., I_1 - 1$, $i_2 = 0, ..., I_2 - 1$, where $\circ$ denotes the Hadamard product, $(\mathcal{Y}^{i_4})_{i_1,i_2}$ is the acquired projection at the (i_1, i_2) position at time i_4, $W \in \mathbb{R}^{I_1 \times J_2 \times I_4}$ denotes the noise in the system, with $J_2 = I_2 + I_3 - 1$, and $(\mathcal{T}_{i_3}^{i_4})_{i_1,i_2}$ and $(\mathcal{F}_{i_3}^{i_4})_{i_1,i_2}$ are the elements in the (i_1, i_2, i_3, i_4) position of the arrays $\mathcal{T}$ and $\mathcal{F}$, respectively.

---

[67] Note that, since tensors of order three or higher are called higher-order tensors, *higher-order tensor sparse representation* it is equivalently referred to the tensor sparsifying transform performed on the high-dimensionality of the spectral videos.

Alternatively, the spectral video $\mathcal{F}$ can be written in vector form as $\mathbf{f} \in \mathbb{R}^n$, with $n = I_1 I_2 I_3 I_4$. Thus, the video C-CASSI acquisition model can be expressed in matrix form as the projection of $\mathbf{f}$ into the video-CASSI sensing matrix $\mathbf{H} \in \mathbb{R}^{m \times n}$ as $\mathbf{y} = \mathbf{H}\mathbf{f}$, where $\mathbf{H}$ accounts for the encoding and dispersion processes and $\mathbf{y} \in \mathbb{R}^m$ represents the compressed measurement vector, with $m = J_2 I_4$ and $m \ll n$. Moreover, given the fact that spectral videos can be highly sparse or compressible in some representation basis, i.e., $\mathbf{f} = \mathbf{D}\boldsymbol{\theta}$, the acquired projections can be rewritten as $\mathbf{y} = \mathbf{H}\boldsymbol{\Psi}\boldsymbol{\theta}$, where $\boldsymbol{\theta} \in \mathbb{R}^n$ is a K-sparse representation of the signal over the spatial, spectral and temporal axes, with $K \ll n$, and $\boldsymbol{\Psi}$ can be selected as a Wavelet or a Cosine basis [1]. In addition, the $\boldsymbol{\Psi}$ basis can be set as the Kronecker product between different basis, e.g., for a spectral video the basis can be expressed as $\boldsymbol{\Psi} = \boldsymbol{\Psi}^1 \otimes \boldsymbol{\Psi}^2 \otimes \boldsymbol{\Psi}^3 \otimes \boldsymbol{\Psi}^4$, where $\boldsymbol{\Psi}^n$ is the transformation that sparsifies the $n-$th dimension of the data and $\boldsymbol{\Psi} \in \mathbb{R}^{n \times n}$. Finally, the set of the video C-CASSI outputs $\mathbf{y} = [\mathbf{y}_0^T, \mathbf{y}_1^T, \ldots, \mathbf{y}_{I_4-1}^T]^T$ can be rewritten as

$$\mathbf{y} = \mathbf{H}\mathbf{D}\boldsymbol{\theta} + \boldsymbol{\omega} = \mathbf{A}\boldsymbol{\theta} + \boldsymbol{\omega}, \tag{35}$$

where the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ represents the video-CASSI sensing matrix and $\boldsymbol{\omega}$ is the noise of the system.

Then, the compressed measurements vector is used to recover the signal, where the inverse problem entails the reconstruction of the spatial, spectral and temporal information of the underlying scene. The recovery problem can be written as

$$\hat{\mathbf{f}} = \boldsymbol{\Psi}^T \left\{ \underset{\boldsymbol{\theta}}{\operatorname{argmin}} ||\mathbf{y} - \mathbf{H}\boldsymbol{\Psi}\boldsymbol{\theta}||_2^2 + \rho ||\boldsymbol{\theta}||_1 \right\} \tag{36}$$

where ρ is a regularization constant.

4.2. Signal Recovery based on Higher-Order Tensor Transform

Unlike the above-mentioned sparsifying transformation, the spectral video $\mathcal{F}$ can be decomposed as a multilinear transformation of a dictionary basis $\{\mathbf{U}^{(z)}\}_{z=1}^{4} \in \mathbb{R}^{R_z \times R_z}$, along each z-mode, and a core tensor $\mathcal{G} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ as follows:

$$\mathcal{F} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}, \tag{37}$$

where the core tensor $\mathcal{G}$ corresponds to the coefficients of $\mathcal{F}$ on each dictionary basis, with $R_1 \leq I_1$, $R_2 \leq I_2$, $R_3 \leq I_3$, and $R_4 \leq I_4$. Figure 13 illustrates the higher-order decomposition of a spectral video, i.e., a four-dimensional tensor.

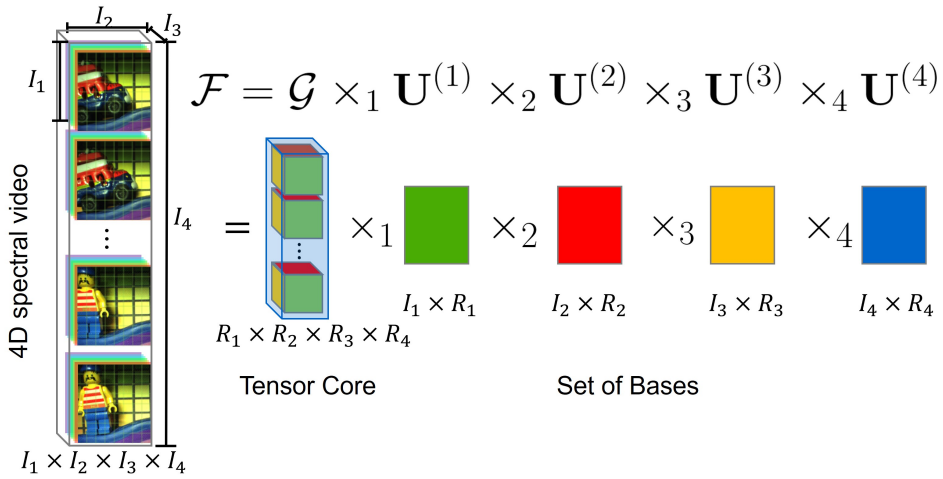


Figure 13. Illustration of the higher-order decomposition of a spectral video scene.

Considering the higher-order sparse representation of Eq. (37), the recovery problem can be reformulated as

$$\begin{aligned} \min_{\mathbf{U}^{(z)}, \mathcal{G}} \quad & \left\| \mathbf{y} - \mathbf{H}\, \mathrm{vec}(\mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}) \right\|_2^2 \\ \text{subject to} \quad & \|\mathrm{vec}(\mathcal{G})\|_1 \leq K, \\ & \mathbf{U}^{(z)T} \mathbf{U}^{(z)} = \mathbf{I}^{(z)}, \ z = 1, ..., 4, \end{aligned} \tag{38}$$

where $\mathcal{G} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$, $\mathbf{U}^{(z)}$ is a dictionary for each dimension $z = 1, ..., 4$, $\mathrm{vec}(\cdot) \rightarrow \{\mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4} :\rightarrow \mathbb{R}^{I_1 I_2 I_3 I_4}\}$ is an operator that arranges a tensor into a column-wise vector. Observe that due to the $\mathrm{vec}(\cdot)$ operation over the tensor representation of the signal, the acquisition model of Eq. (35) is still used in the recovery problem. Notice that Eq. (38) can be also written as

$$\begin{aligned} \underset{\mathbf{U}^{(z)}, \mathbf{x}, \mathcal{G}}{\operatorname{argmin}} \quad & \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 \\ \text{subject to} \quad & \mathbf{x} = \mathrm{vec}(\mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}), \\ & \|\mathrm{vec}(\mathcal{G})\|_1 \leq K, \\ & \mathbf{U}^{(z)T}\mathbf{U}^{(z)} = \mathbf{I}^{(z)}, \ z = 1, ..., 4, \end{aligned} \tag{39}$$

where $\mathbf{x}$ is an auxiliary variable. An alternating direction method can be used to solve Eq. (39) [68]. The augmented Lagrangian function of problem in Eq. (39) is

$$\begin{aligned} \mathbb{L}_\rho\big(\mathcal{G}, \mathbf{x}, \mathbf{U}^{(z)}, \mathcal{B}\big) = & \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda ||\mathrm{vec}(\mathcal{G})||_1 \\ & + \frac{\lambda}{2}\|\mathbf{x} - \mathrm{vec}(\mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}) + \mathrm{vec}(\mathcal{B})\|_2^2 \\ & + \sum_{z=1}^{4} \mathcal{I}_{\mathcal{U}}\big(\mathbf{U}^{(z)}\big), \ z = 1, ..., 4, \end{aligned} \tag{40}$$

where $\lambda > 0$, $\mathcal{B}$ is the Lagrange multiplier, and the indicator function $\mathcal{I}_{\mathcal{U}}\big(\mathbf{U}^{(z)}\big)$ is given by $\mathcal{I}_{\mathcal{U}}\big(\mathbf{U}^{(z)}\big) = \big\{\mathbf{U}^{(z)} \in \mathcal{U} \rightarrow 1\big\}$, where $\mathcal{U} = \{\mathbf{U} \in \mathbb{R}^{J_z \times J_z} : \mathbf{U}^{(z)T}\mathbf{U}^{(z)} = \mathbf{I}^{(z)}\}$. The problem in Eq. (40) can split into three main subproblems related to the variables $\mathcal{G}, \mathbf{x}$, and $\mathbf{U}^{(z)}$, where each subproblem is iteratively updated while the others are

---

[68] Stephen Boyd et al. "Distributed optimization and statistical learning via the alternating direction method of multipliers". In: *Foundations and Trends® in Machine learning* 3.1 (2011), pp. 1–122.

fixed. Specifically, the $\mathbf{x}$ subproblem is formulated as

$$\mathbf{x}^{p+1} = \underset{\mathbf{x}}{\text{argmin}} \ \|\mathbf{y} - \mathbf{H}\mathbf{x}^{p}\|_2^2 + \frac{\lambda}{2}\|\mathbf{x} - \text{vec}(\mathcal{X}^{p}) + \text{vec}(\mathcal{G}^{p})\|_2^2, \tag{41}$$

where $\mathcal{X}^{p} = \mathcal{G}^{p} \times_1 \mathbf{U}_p^{(1)} \times_2 \mathbf{U}_p^{(2)} \times_3 \mathbf{U}_p^{(3)} \times_4 \mathbf{U}_p^{(4)}$, and the solution can be found using a conjugate gradient. On the other hand, the $\mathcal{G}$ subproblem is given by

$$\mathcal{G}^{p+1} = \underset{\mathcal{G}}{\text{argmin}}\, \tau \left\|\text{vec}(\mathcal{G}^{p})\right\|_1 + \frac{\lambda}{2}\|\mathbf{x}^{p+1} - \text{vec}(\mathcal{X}^{p}) + \text{vec}(\mathcal{B}^{p})\|_2^2, \tag{42}$$

where τ is a regularization parameter. The solution for the $\mathcal{G}$ subproblem can be obtained from the well-known soft-thresholding operation. Then, the $\mathbf{U}^{(z)}$ subproblem is written as

$$\mathbf{U}_{p+1}^{(z)} = \underset{\mathbf{U}^{(z)}}{\text{argmin}}\, \mathcal{I}_{\mathcal{U}}\big(\mathbf{U}^{(z)}\big) + \frac{\lambda}{2}\|\mathbf{x}^{p+1} - \text{vec}(\mathcal{G}^{p+1} \times_1 \mathbf{U}_p^{(1)} \times_2 \mathbf{U}_p^{(2)} \times_3 \mathbf{U}_p^{(3)} \times_4 \mathbf{U}_p^{(4)}) + \text{vec}(\mathcal{B}^{p})\|_2^2, \tag{43}$$

for $z = 1, ..., 4$, whose solution can be found by using the Higher-Order Orthogonal Iteration algorithm reported in [69] [Section 4.2, Algorithm 4.2, step 2], where, for each z-th dimension, the $\mathbf{U}^{(z)}$ matrix is refined while the other matrices are kept constant. Finally, the Lagrange multiplier is updated as

$$\mathcal{B}^{p+1} = \mathcal{B}^{p} - \mathcal{G}^{p+1} \times_1 \mathbf{U}_{p+1}^{(1)} \times_2 \mathbf{U}_{p+1}^{(2)} \times_3 \mathbf{U}_{p+1}^{(3)} \times_4 \mathbf{U}_{p+1}^{(4)} + \text{vec}^{-1}(\mathbf{x}^{p+1}), \tag{44}$$

where $\mathcal{B}^0$ is zero-initialized. Algorithm 3 summarizes the steps of the alternating direction scheme, where the stopping criterion can be the number of iterations. Finally, the recovered spectral video is obtained as $\hat{\mathcal{F}} = \hat{\mathcal{G}} \times_1 \hat{\mathbf{U}}^{(1)} \times_2 \hat{\mathbf{U}}^{(2)} \times_3 \hat{\mathbf{U}}^{(3)} \times_4 \hat{\mathbf{U}}^{(4)}$.

---

[69] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. "On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors". In: *SIAM journal on Matrix Analysis and Applications* 21.4 (2000), pp. 1324–1342.

Algorithm 3: Higher-order Representation-based Recovery Algorithm

Input : $\mathbf{y}$, $\mathbf{H}$, $\lambda > 0, \tau > 0$

$\mathcal{B}^0 = \mathbf{0}$, $\mathbf{x}^0 = 0$, $p = 0$, $\mathbf{U}_0^{(n)}$: set an initial 1D transform for each dimension n (e.g. a Cosine)

while *Some stopping criterion is not satisfied* **do**

  Update $\mathbf{x}^{p+1}$ by solving Eq. (41)

  Update $\mathcal{G}^{p+1}$ by solving Eq. (42)

  for $n = 1, ..., 4$ **do**

    Update $\mathbf{U}_{p+1}^{(n)}$ by solving Eq. (42)

  Update the multiplier $\mathcal{B}^{p+1}$ by Eq. (44)

  $p = p + 1$

Output: $\hat{\mathcal{G}}$, $\hat{\mathbf{U}}^{(n)}$

4.3. Simulations and Results

In order to evaluate the proposed algorithm for compressive spectral video recovery, three multi-spectral videos were sensed using the model in Eq. (35). The first dataset is a cropped section of the scene taken from [30] called 'Boxes'. The second is a synthetic video of a moving window over a static spectral scene taken from [70] called 'Windows', and the third dataset is a real scene of a surveillance camera taken from [7] called 'Cars'. The 'Boxes' and 'Windows' datasets exhibit a resolution of $128(I_1) \times 128(I_2)$ spatial pixels, $I_3 = 8$ spectral bands, and $I_4 = 8$ frames, and the 'Cars' dataset exhibits $128(I_1) \times 128(I_2)$ pixels of spatial resolution, $I_3 = 7$ spectral bands, and $I_4 = 8$ frames. For the different experiments, realizations of temporal colored coded apertures with a transmittance of 0.25 were employed [1], where the transmittance is defined as the portion of light intensity passing through the aperture code with

[70] Fumihito Yasuma et al. *CAVE Projects: Multispectral Image Database*. 2008. URL: `http://www.cs.columbia.edu/CAVE/databases/multispectral/`.

respect to the overall intensity [71]. The peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) metrics were used to quantify the reconstruction quality.

The set of compressed videos were recovered with the proposed method based on sparse tensor representation and the traditional recovery method using the GPSR algorithm [62], so-called 'Proposed' and 'Traditional', respectively. The representation basis for the traditional reconstruction problem is selected as follows. Denoting $\mathbf{\Psi}_{DWT}$ and $\mathbf{\Psi}_{DCT}$ as a 1D-Discrete Wavelet Transform (DWT) and a 1D-Discrete Cosine Transform (DCT), respectively, then, the sparse representation is the Kronecker product given by $\mathbf{\Psi} = \mathbf{\Psi}_{DWT} \otimes \mathbf{\Psi}_{DWT} \otimes \mathbf{\Psi}_{DCT} \otimes \mathbf{\Psi}_{DCT}$ [271]. For Algorithm 1, the dictionaries were initialized using a 1D DCT for each dimension. And, even though there are no convergence guarantees to find a global minimum given the non-convexity of the problem, in practice, the algorithm reaches PSNR values higher than its counterparts after 100 iterations. The regularization parameters (λ, τ, ρ) were tuned via cross-validation.

Figure 14 shows an RGB representation of the original frames 1, 4 and 7 and the reconstructions obtained from the traditional and the proposed recovery methods, where the quality of reconstruction of the frame in terms of the average PSNR over the spectral frames is also shown. Notice that the reconstructions obtained from the proposed recovery outperform the traditional from 1dB up to 5dB.

To evaluate the performance of the proposed basis with respect to the additive noise produced in the acquisition process, the set of measurements from spectral videos were simulated by adding levels of Gaussian noise of 15, 20, 25, 30, and 50 decibels. Table 5 summarizes the obtained results in terms of average PSNR and SSIM. Observe that the lowest PSNR values are obtained using the real dataset 'Cars', never-

---

[71] Laura Galvis et al. "Coded aperture design in compressive spectral imaging based on side information". In: *Applied optics* 56.22 (2017), pp. 6332–6340.

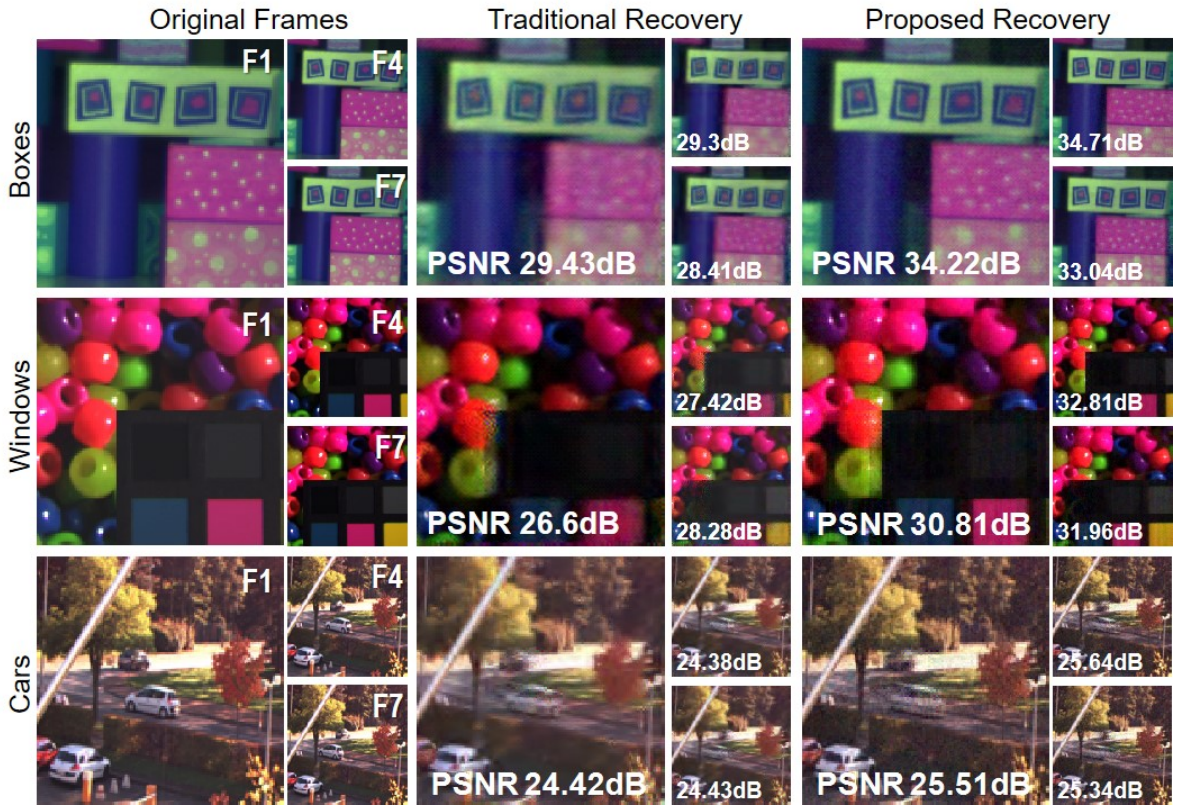


Figure 14. RGB representation of the original frames 1, 4 and 7 (1rst column) of the test videos and the recovery results from the traditional method (2nd column) and the proposed recovery (3rd column). The PSNR of each frame is also shown.

theless, the performance of the proposed approach exceeds the traditional recovery. In general, the reconstruction results obtained from the proposed approach exhibit higher PSNR and SSIM values than those obtained with the traditional method.

Figures 15, 16, and 17 illustrate the comparison of spectral bands from the reconstruction with 25 dB of noise on the measurements. Observe that in general the proposed method entails better reconstructions in comparison with the traditional method.

Table 5. Average reconstruction PSNR and SSIM from different levels of noise and the three spectral video datasets.

PSNR						
	SNR Noise Level [dB]	15	20	25	30	50
Boxes	Traditional	21.52	25.56	28.17	28.92	28.94
	Proposed	**23.61**	**27.03**	**28.76**	**31.21**	**33.95**
Windows	Traditional	25.78	26.94	27.09	27.18	27.52
	Proposed	**26.84**	**28.65**	**30.09**	**31.49**	**32.13**
Cars	Traditional	20.05	23.05	23.67	24.26	24.40
	Proposed	**21.57**	**24.02**	**25.20**	**26.07**	**27.00**
SSIM						
	SNR Noise Level [dB]	15	20	25	30	50
Boxes	Traditional	0.563	0.805	0.870	0.905	0.908
	Proposed	**0.673**	**0.825**	**0.871**	**0.928**	**0.966**
Windows	Traditional	0.702	0.710	0.829	0.864	0.872
	Proposed	**0.737**	**0.824**	**0.863**	**0.905**	**0.932**
Cars	Traditional	0.644	0.838	0.868	0.893	0.897
	Proposed	**0.724**	**0.842**	**0.877**	**0.906**	**0.921**

4.4. Conclusions

In this Chapter, a higher-order sparse representation-based recovery algorithm that exploits the high-order structure spectral videos in the coded aperture snapshot imaging architecture has been presented. The spectral video is modeled by using a higher-order decomposition for taking advantage of the structure and for fully exploiting the inherent redundancy of the data. The recovery algorithm refines the representation basis while the reconstruction proceeds. Numerical simulations demonstrated that considering the higher-order representation of the high-dimensional signal in the recovery problem leads to an improvement in the reconstruction accuracy, even in the presence of Gaussian noise in the measurements.

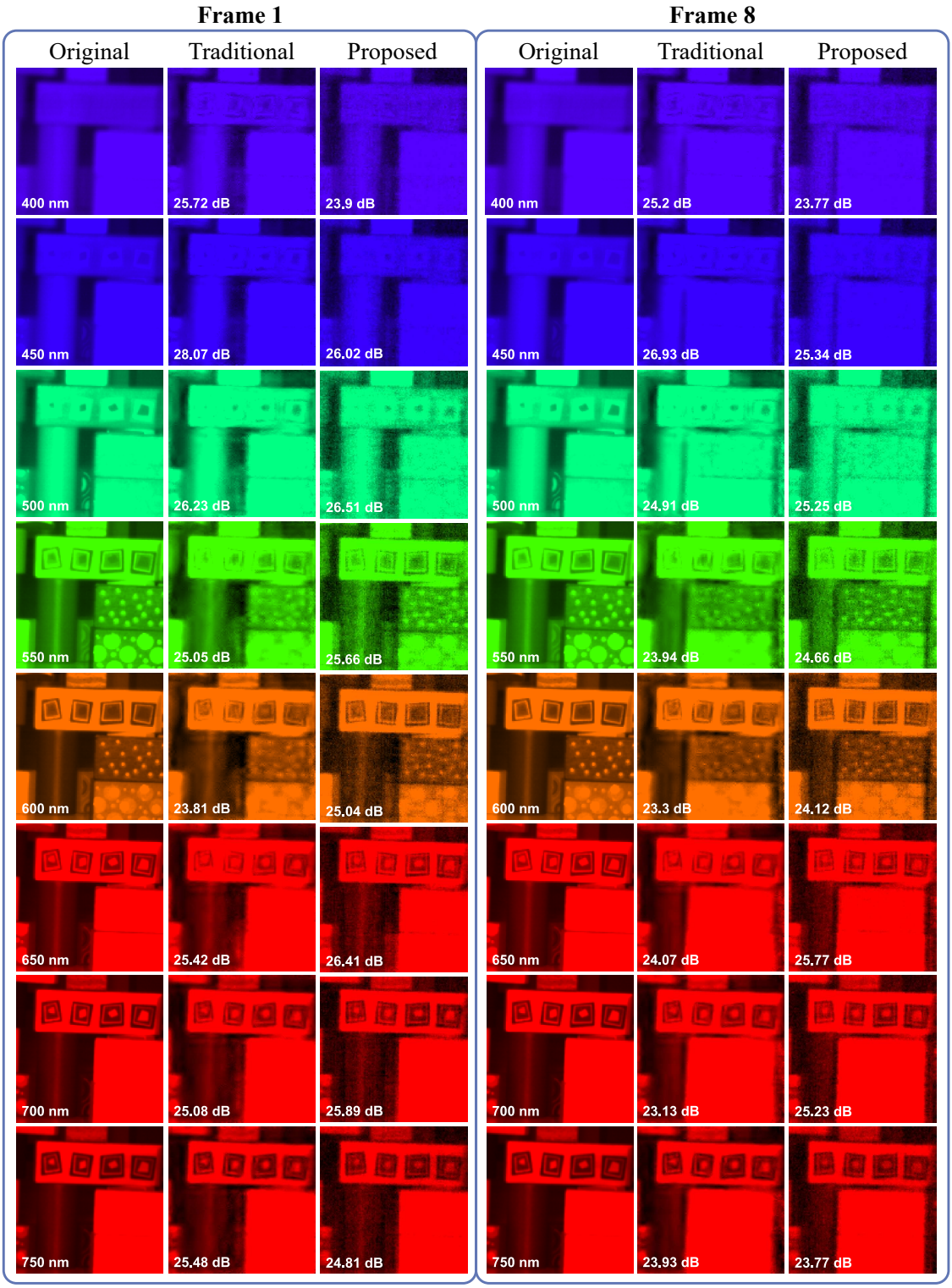


Figure 15. Spectral bands of the original and reconstructions from the frames 1 and 8 of the Boxes video 1 with 25dB of level of noise.

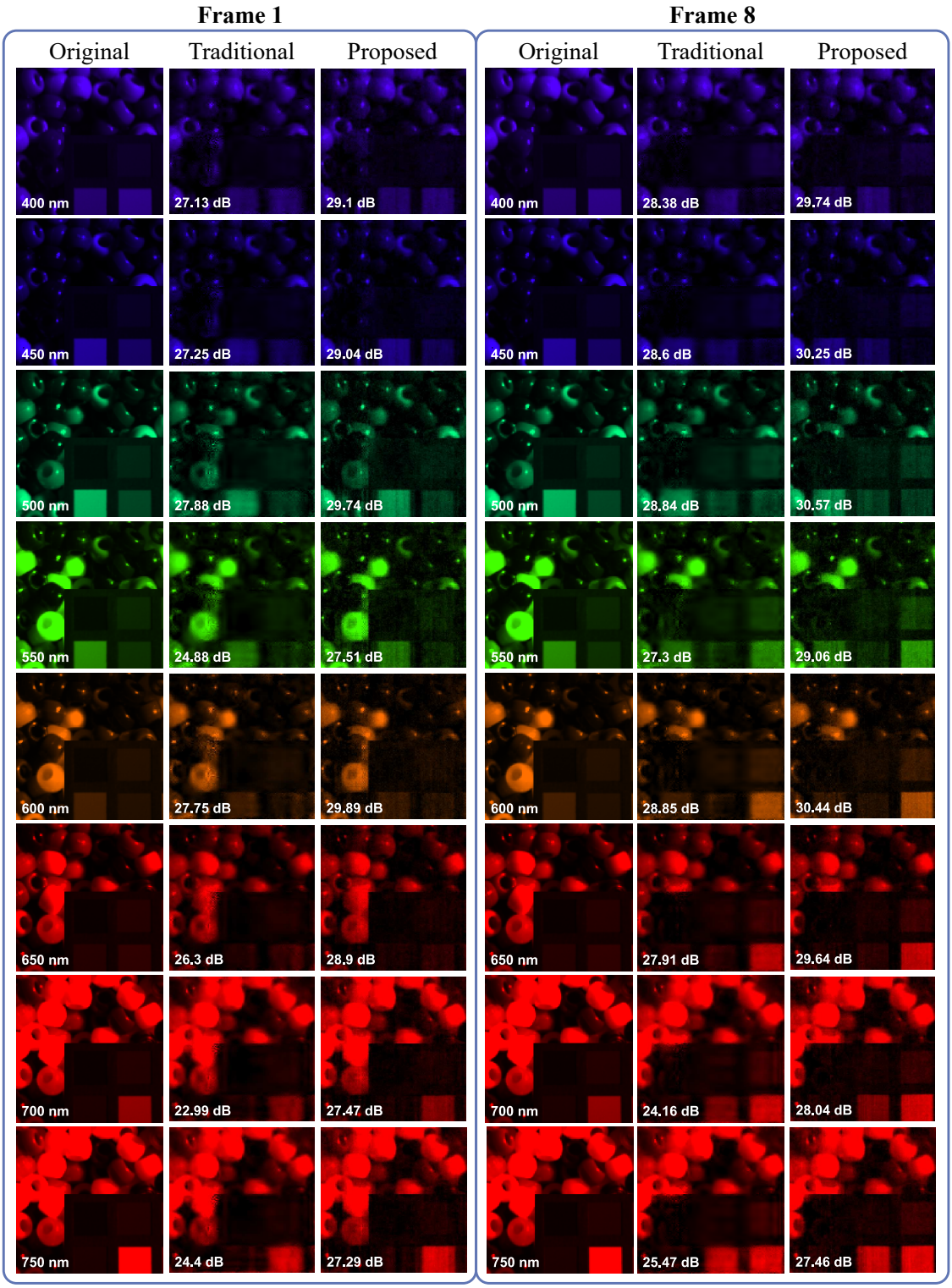


Figure 16. Spectral bands of the original and reconstructions from the frames 1 and 8 of the Windows video 1 with 25dB of level of noise.

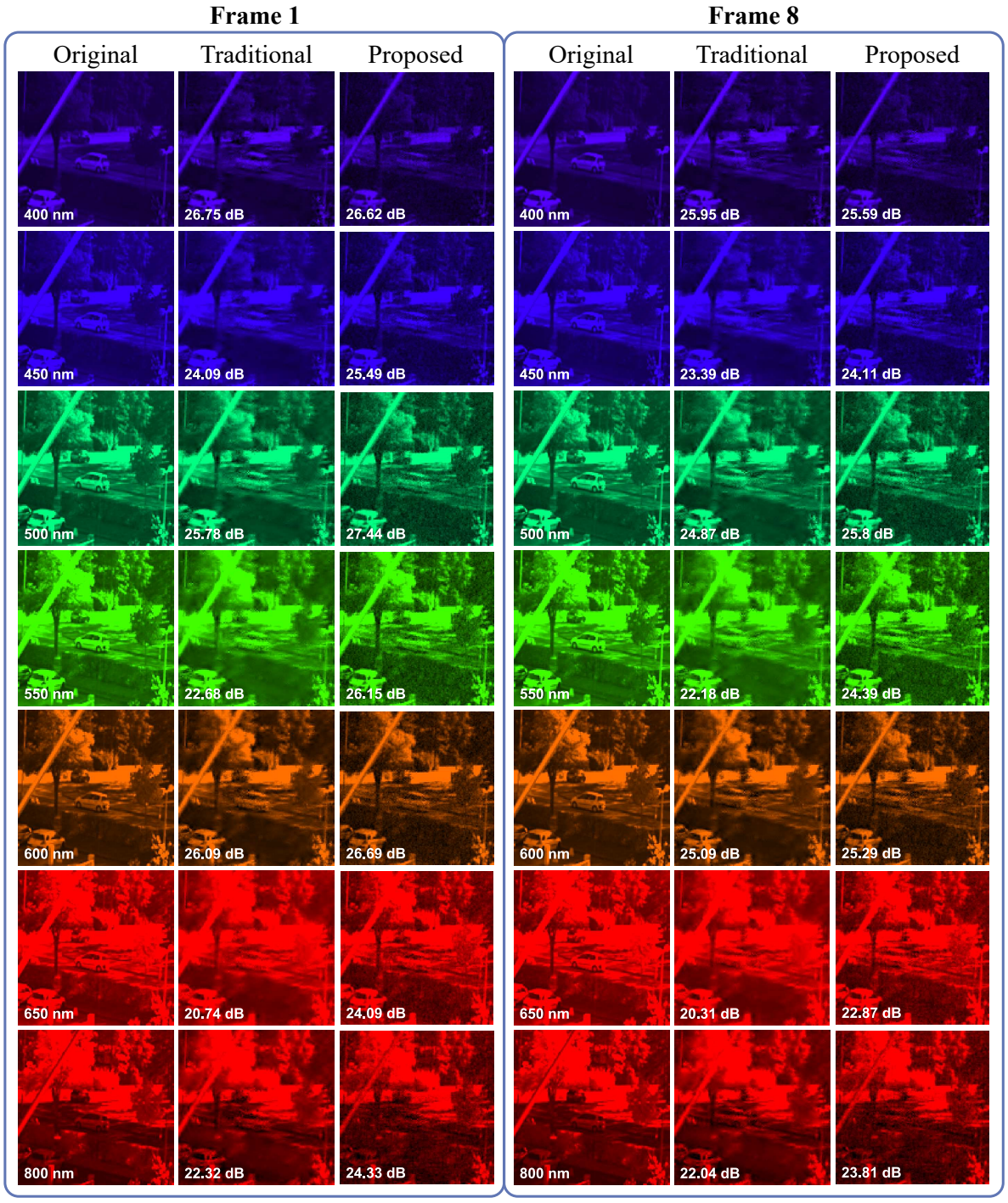


Figure 17. Spectral bands of the original and reconstructions from the frames 1 and 8 of the Cars video 1 with 25dB of level of noise.

5. END-TO-END SPATIO-TEMPORAL BINARY CODED APERTURE DESIGN AND RECOVERY IN COMPRESSIVE SPECTRAL VIDEO SENSING

5.1. Introduction

Spectral video acquisition via compressive spectral imaging, named compressive spectral video sensing (CSVS), has shown promising results and arisen as an alternative for dimensionality, processing, and sensor costs reduction [12][13][21][4][15]. To this end, snapshot compressive imaging (SCI) systems have been extended to acquire spectral image frames from dynamic scenes by multiplexing the spatio-spectral information [16][13][11][7]. Some SCI architectures employed for spectral video acquisition include the coded-aperture snapshot spectral imager (CASSI) [27][11][4] and the spatial-spectral coded compressive spectral imager (3D-CASSI) [4]. Although the temporal information under the CSVS framework is not multiplexed or compressed, the temporal correlations joined to the spatial and spectral redundancies are exploited in the encoding and decoding steps to yield suitable sensing and reconstruction protocols for the scene under observation [12][17][4]. The encoding step in the CSVS encompasses the dispersion and codification of the input scene in the optical path before the light is recorded at the sensor. Typically, the dispersion and codification are obtained using optical elements such as a prism and a coded aperture (CA) or mask, respectively. Then, a reconstruction algorithm is employed to estimate a version of the underlying scene from the compressed frames [12][16][1]. Different works in the literature have proposed strategies to improve the recovered image quality by either designing the CA pattern [1][16] or by customizing the recovery algorithm [4][2][18] independently of the data under observation. The CA can be composed of binary pixels [27][16], which entails the block or unblock encoding of the scene, or *colored* pixels, which indeed are optical filters that modulate the scene with a specific wavelength for a richness encoding of

the spectral information [37][44]. Random distributions of the CA elements are typically used for compressive image acquisition. Nonetheless, several works have proposed to design these patterns for better sensing and to exploit the scene under observation, providing better image quality reconstructions. The interest of the CA design problem has become a research study area in the last decade. In particular, state-of-the-art in CA design for the acquisition of spectro-temporal scenes includes both binary CAs, such as the blue noise (BN) patterns [16]; and colored CAs, such as the temporal colored coded aperture (TCCA) [72][21]. And, although the colored CAs provide a richer sensing than binary CAs, the binary CAs are much easier to implement on real experiments [15].

Recently, data-driven deep learning (DL) approaches have shown outstanding performances in terms of image quality when the CA and the recovery algorithm are jointly designed by exploiting tons of current available data in video compressing sensing [19][73][20][21]. However, these approaches disregard the spectral information of the dynamic scene, and the multiplexing is lead across the temporal dimension aimed at compressing a sequence of video frames into a single 2D measurement. Other works such as [15] have exploited self-attention mechanisms in DL on SCI at video rates for spectral images and video reconstruction, however, the CA is fixed, i.e. it is not learned from the training data. Finally, authors in [74] have demonstrated the potential of designing the CA and the recovery problem by considering either physical constraints in the SCI system like the binarization of the mask or the opti-

---

[72] Kareth León-López, Laura Galvis, and Henry Arguello Fuentes. “Spatio-spectro-temporal coded aperture design for multiresolution compressive spectral video sensing”. In: *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE. 2017, pp. 728–732.

[73] Jiawei Ma et al. “Deep tensor admm-net for snapshot compressive imaging”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 10223–10232.

[74] Jorge Bacca, Tatiana Gelvez, and Henry Arguello. “Deep Coded Aperture Design: An End-to-End Approach for Computational Imaging Tasks”. In: *arXiv preprint arXiv:2105.03390* (2021).

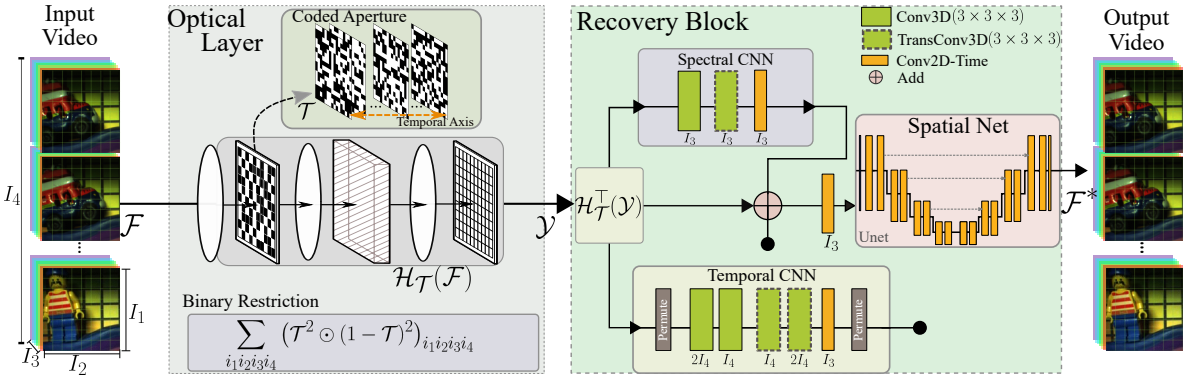


Figure 18. Proposed E2E architecture composed of the optical (CSVS) layer, which is a layer that emulates the video acquisition while learns the coded aperture pattern; and the recovery block (so-called STNET), which learns the weights for recovering the videos. A set of I_4 frames of a spectral video go through the optical layer. Then, the recovery block takes as input the I_4 video measurements and outputs the recovered version of the video and the resulting CA from the training. Spectral, temporal, and spatial convolutional layers are applied for recovering the video, where the *Permute* operation swaps the spectral and the temporal dimensions to operate the convolutions across the time axis.

mal number of multiple snapshots, where the methodology is presented for different applications and signals such as hyperspectral and depth images but not for spectral dynamic scenes, where the signal is changing across time.

This Chapter presents an end-to-end (E2E) deep learning approach to jointly design a set of binary CAs and the reconstruction method for sensing and recovery spectral videos from CASSI compressed measurements. The proposed E2E network is composed of the optical (CSVS) layer, which encodes the inputs while learns the binary CA from the training data; and the recovery block (so-called STNET since it is based on spatio-spectro-temporal convolutions), which applies convolutions across the different dimensions of the spectral video to minimize the error between the measurements and the projection of the network output into the system. The weights of the optical layer are particularly restricted in the loss function to be binary for obtaining a set of designed binary CAs, which are easier to implement in the CASSI system. The loss function of the proposed E2E network then attempts to learn the

weights for obtaining the best CA pattern meanwhile learns the weights for recovering the best version of the training dataset from the measurements. Figure 18 illustrates the proposed E2E architecture for the spectral video CA design and reconstruction. Observe that the input to the recovery block is a transposed version of the measurements. Then, the underlying signal goes through three sub-networks: the spectral, the temporal, and the spatial modules, for exploiting the different dimensions of the signal. The proposed E2E approach gains of up to 1dB compared with a state-of-the-art E2E architecture and 5dB compared with a traditional recovery method using BN patterns. It is important to highlight that the resulting binary CAs are designed to acquire a single snapshot per frame for doing a more realistic scenario that can be implemented using optical devices. Additionally, note that the CSVS architecture employed for showing the profit of the E2E methodology is the CASSI system, nonetheless, the approach can be extended to other compressive spectral-based architectures such as the 3D-CASSI by adjusting the optical layer sensing.

The main contribution of the method of this Chapter relies on the E2E approach for CA designing and recovering spectral videos from compressive snapshot imaging systems using a single snapshot per frame by exploiting the spectral, spatial, and temporal correlations of the scenes.

5.2. Video CASSI System Modeling

Let $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4}$ be a discretized spectral video scene represented as a fourth-order tensor, where $I_1 \times I_2$ denotes the spatial size, I_3 the spectral bands, and I_4 the number of video frames. Then, in general form, the acquisition procedure in a CSVS architecture can be expressed as

$$\mathcal{Y} = \mathcal{H}_{\mathcal{T}}(\mathcal{F}) + \mathcal{W}, \tag{45}$$

where $\mathcal{H}_{\mathcal{T}}(\mathcal{F}) : \mathbb{R}^{I_1 \times I_2 \times I_3 \times I_4} \rightarrow \mathbb{R}^{I_1 \times J_2 \times 1 \times I_4}$ represents the CSVS operator whose operation depends on both the CA $\mathcal{T}$ and the optical configuration of the system, and J_2 denotes the number of rows of the resulting measurements.

For the video CASSI system, the measurements $\mathcal{Y} \in \mathbb{R}^{I_1 \times J_2 \times I_4}$ in the i_4-th frame can be modeled as

$$\mathcal{Y}_{i_1,i_2,i_4} = \sum_{i_3=1}^{I_3} \mathcal{F}_{i_1,(i_2-i_3),i_3,i_4} \odot \mathcal{T}_{i_1,(i_2-i_3),i_4} + \mathcal{W}_{i_1,i_2,i_4}, \tag{46}$$

for $i_4 = 1, ..., I_4$, where $\odot$ denotes the element-wise multiplication, $\mathcal{W} \in \mathbb{R}^{I_1 \times J_2 \times I_4}$ denotes the noise in the system with $J_2 = (I_2 + I_3 - 1)$, and $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times I_4}$ represents the tensor form of the binary CA. In general, the entries of the $\mathcal{T}$ can be generated either by following a specific design, as in [16], or by following random structures such as Bernoulli or Gaussian random distributions [2715].

The sensing process of the system can be compactly formulated in matrix form as

$$\mathbf{y} = \mathbf{H}_{\mathcal{T}}\mathbf{f} + \mathbf{w}, \tag{47}$$

where $\mathbf{y}$ and $\mathbf{f}$ are the column-vectorized versions of $\mathcal{Y}$ and $\mathcal{F}$, and $\mathbf{H}_{\mathcal{T}} \in \mathbb{R}^{m \times I_1 I_2 I_3 I_4}$ is the CSVS sensing matrix that models the CA and shifting effects of the system, and m is set as $m = I_1(I_2 + I_3 - 1)I_4$ for the video CASSI system given the shifting produced by the dispersion element.

5.3. End-to-End (E2E) Learning Approach

5.3.1. Loss Function and Regularization Let $\boldsymbol{N}_{\boldsymbol{\theta}}\{\cdot\}$ denotes the spatial-spectral-temporal (STNET) convolutional neural network (CNN) to be trained with weights $\boldsymbol{\theta}$. From a set of L training spectral videos, the cost function of the E2E approach is

given by

$$(\mathcal{F}^*, \mathcal{T}^*) \in \underset{\mathcal{T},\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{L} \sum_{\ell=1}^{L} \left\| \mathcal{Y}^\ell - \boldsymbol{N}_{\boldsymbol{\theta}} \{ \mathcal{H}_{\mathcal{T}}(\mathcal{F}^\ell) \} \right\|_2^2 + \tau \boldsymbol{R}(\mathcal{T}), \tag{48}$$

where τ is a regularization constant, and $\boldsymbol{R}(\mathcal{T})$ is a regularization function to induce the weights of $\mathcal{T}$ being designed following specific properties such as binary entries, or efficient number of snapshots [75][74]. Typically, in compressive video systems, the acquisition of multiple snapshots of a given frame is not contemplated, due to the scene is rapidly changing across time. On the other hand, the binarization constraint provides a suitable CA that can be implemented in real acquisition systems using digital micromirror devices (DMD) [37]. Mathematically, the binarization constraint is written as

$$\boldsymbol{R}(\mathcal{T}) = \sum_{i_1 i_2 i_3 i_4} \left(\mathcal{T}^2 \odot (1 - \mathcal{T})^2 \right)_{i_1 i_2 i_3 i_4}, \tag{49}$$

where $\odot$ denotes the element-wise product [74]. Then, Eq. (49) is minimized into the cost function (48) when the elements in $\mathcal{T}$ are either (0) or (1). In this way, the binary CA and recovery network weights are jointly trained in the E2E network.

5.3.2. Network Architecture As shown in Figure 18, the proposed E2E network is mainly composed of the optical layer and the recovery block. The optical layer initially generates a CA at random with $I_1 \times I_2$ spatial resolution and I_4 temporal frames. Then, the operation in Eq. (46) is performed for each video of the training set. After obtaining the measurements $\mathcal{Y}$, an operator of the form $\mathcal{H}_{\mathcal{T}}^{\top}(\mathcal{Y})$ is applied, where the operator can be conducted by a repeat copy operation of $\mathcal{Y}$ I_3 times, considering the shifting of the CASSI, and an element-wise multiplication with $\mathcal{T}$.

---

[75] Catherine F Higham et al. “Deep learning for real-time single-pixel video”. In: *Scientific reports* 8.1 (2018), pp. 1–9.

These two operations are performed efficiently in tensor form to avoid extra steps in the vectorization and to reduce the storage of huge matrices. Finally, the weights of the CA are binarized using Eq. 49 during the training process.

On the other hand, the recovery block is composed of three sub-networks: the spectral, temporal, and spatial networks, where the spectral and temporal networks work independently and the spatial network employs as backbone the well-known Unet architecture [76][74]. In particular, the spectral network is composed of a 3D convolutional layer followed by its corresponding transpose operator and then a 2D convolution operation performed across time. The temporal network starts the process by swapping the spectral and temporal dimensions of the input, then, two 3D convolutions and their transposes are applied, followed by a 2D convolution. The outputs of the spectral and temporal networks are summed to the input of the recovery layers. The resulting summed tensor passes through a set of 2D convolution layers that input to the spatial network. For the spatial network based on the Unet, the 2D convolutions operations are estimated across the spectral video frames using a time distributed wrapper, where the spectral dimension correspond to the dimensionality of the output space of the convolution. It is important to mention that all layers in the proposed architecture use ReLU as their activation function.

5.4. Simulations and Results

5.4.1. Spectral Video Datasets The numerical experiments were conducted on a multispectral video dataset provided by a hyperspectral object tracking challenge[77][75]. A total of 39 videos, 29 for training and 10 for testing, were chosen from

---

[76] O. Ronneberger, P.Fischer, and T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. vol. 9351. LNCS. Springer, 2015, pp. 234–241.

[77] Dataset link: `https://www.hsitracking.com/contest/`

the dataset. To prepare the dataset for the training, a preprocessing and augmentation procedures are applied to each spectral video sequence. Figure 19 illustrates the flowchart of the procedures to get (a) a cleaned and (b) augmented subset of training and validation data. In the procedure, the input spectral video with D spectral frames is temporally segmented according to the a given time window length I_4, resulting in $\lfloor D/I_4 \rfloor$ subsequences. Then, each subsequence is visually inspected across the spectral bands to detect bands with errors. If it is found a band with an errors such as missed information, the subsequence is discarded. If there are not errors, the subsequence is spatially resized to a given size and the normalized with a min-max normalization. After this preprocessing, each subsequence is randomly stored either in the training or validation data folder. For augmentation, operations such as random rotation, random scale, and vertical and horizontal displacements are applied three times per sequence segment, obtaining, in this way, 3 augmented variations of each subsequence per subset. Regarding the testing dataset, this set is preprocessed using the procedure illustrated in Fig. 19 (a), omitting the augmentation procedure, and only the first subsequence of each video is selected for the testing database. For analysis purposes, two testing datasets are used, the Testing Dataset 1 are videos that are not related to the training neither to the validation data; these videos are used to evaluate the generalization of the network to natural scenes. And the Testing Dataset 2 are 10 subsequences videos extracted from the validation set (but not used in the validation) used to show the performance of the network on videos with similar information of the background but different information on the foreground, since the scene has changed.

After preprocessing and data augmentation, the total number of spectral videos for the training stage was 526 with a spatial resolution of 128×128 spatial pixels, 16 spectral bands in the wavelength from 470nm to 620nm with a step of 10nm, and

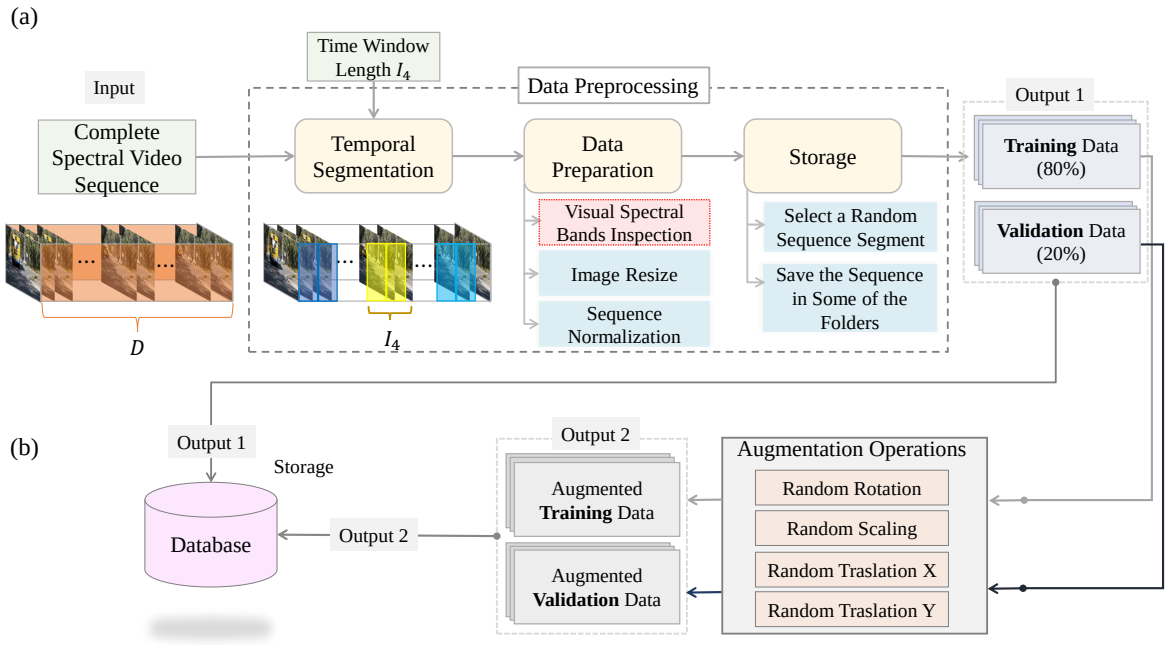


Figure 19. Illustration of the dataset (a) preprocessing and (b) augmentation procedures of one spectral video sequence with an initial temporal resolution D. In the *Visual Spectral Bands Inspection* step, if the sequence segment has errors across the spectral bands, the sequence is discarded.

$I_4 = 8$ frames per second (FPS)[78]. And the resulting number of spectral videos for validation was 146. All the spectral videos of this dataset are videos with associated challenging factors, including illumination variations, occlusions, deformations, motion blur, low resolution, among others [78]. These factors allow the network to learn more realistic features from the real world than when using a synthetic controlled dataset. The set of spectral videos used for the testing part are shown in Figures 20 and 21, where each row shows a given second of the video and each column shows the different scenes in an RGB false colour representation.

[78] Note that the original videos in (Fengchao Xiong, Jun Zhou, and Yuntao Qian. "Material based object tracking in hyperspectral videos". In: *IEEE Transactions on Image Processing* 29 [2020], pp. 3719–3733) have 25 FPS, however, for evaluation purposes in this work, the videos were cropped to 8 frames, leading to 8 FPS.

On the other hand, five real sequences of spectral images acquired in the Optics Lab of the High Dimensional Signal Processing (HDSP) research group at Universidad Industrial de Santander were used to assess the designed CAs. All the sequences were acquired with a CCD camera, and the wavelengths were selected in the range of 470nm to 620nm with a step of 10nm, to keep the uniformity with the previously described multispectral video dataset. The real sequences, named as *Campesina, Toy Car, Hat, Lego,* and *Chiva*, contain different kinds of controlled movements such as vertical/horizontal displacement of an object and circular motion in the background. For the vertical and circular movements of the objects, two Thorlabs devices were used: a single-axis translation stage with standard micrometer device, and a high-precision rotation mount for 25.4 mm device, respectively. For more details of the dataset refers to the Annex 3. The videos were resized to $128 \times 128 \times 16 \times 8$. Figure 22 shows 3 frames of three sequences (named *Campesina, toy car,* and *hat*) in (a) an RGB representation and (b) the given wavelength for each last frame in (a).

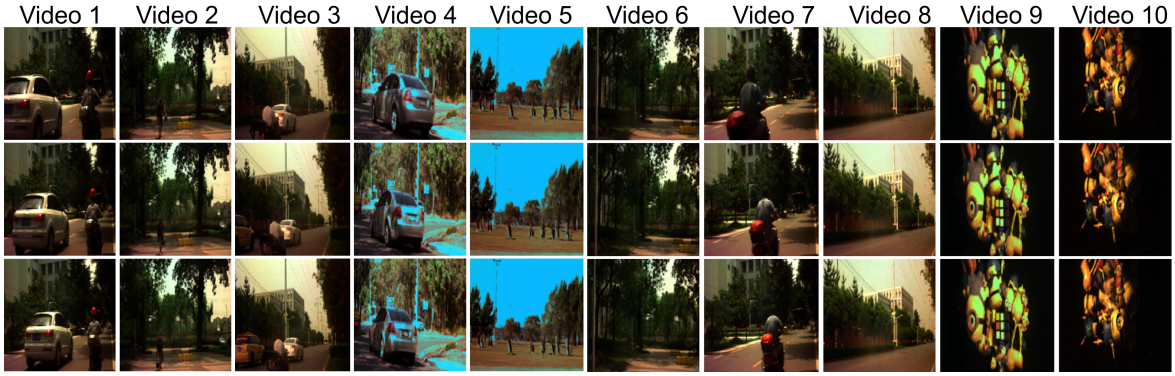


Figure 20. Testing Dataset 1. RGB false colour representation of the 10 spectral videos used for testing. Each row shows the image frame in the seconds 0, 0.37, and 1 (or the frames 1, 5, and 8) of each video.

5.4.2. Compared Methods and Performance Metrics For comparison purposes, the alternating direction method of multipliers algorithm (ADMM) for minimizing the

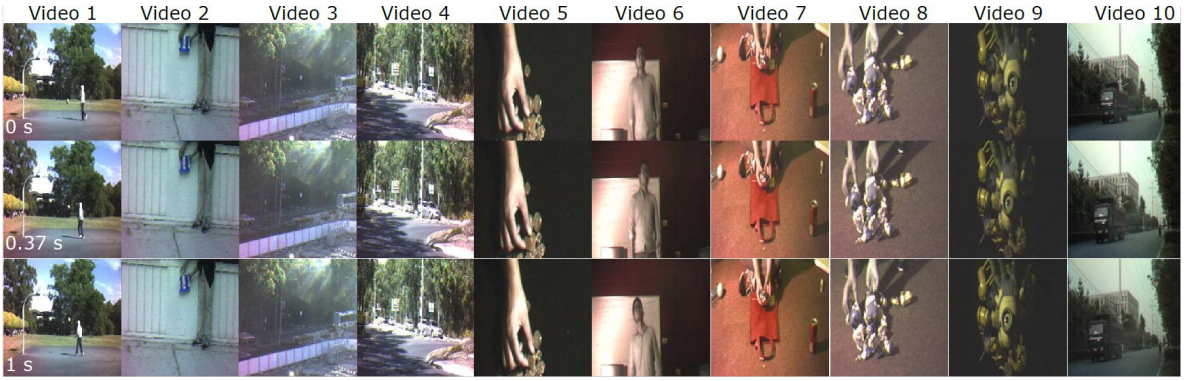


Figure 21. Testing Dataset 2. RGB false colour representation of the 10 spectral videos used for testing. Each row shows the image frame in the seconds 0, 0.37, and 1 (or the frames 1, 5, and 8) of each video.

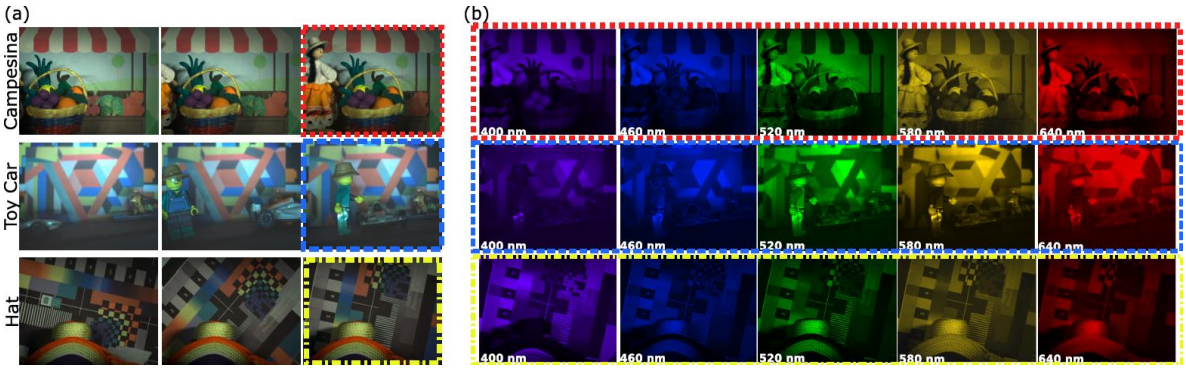
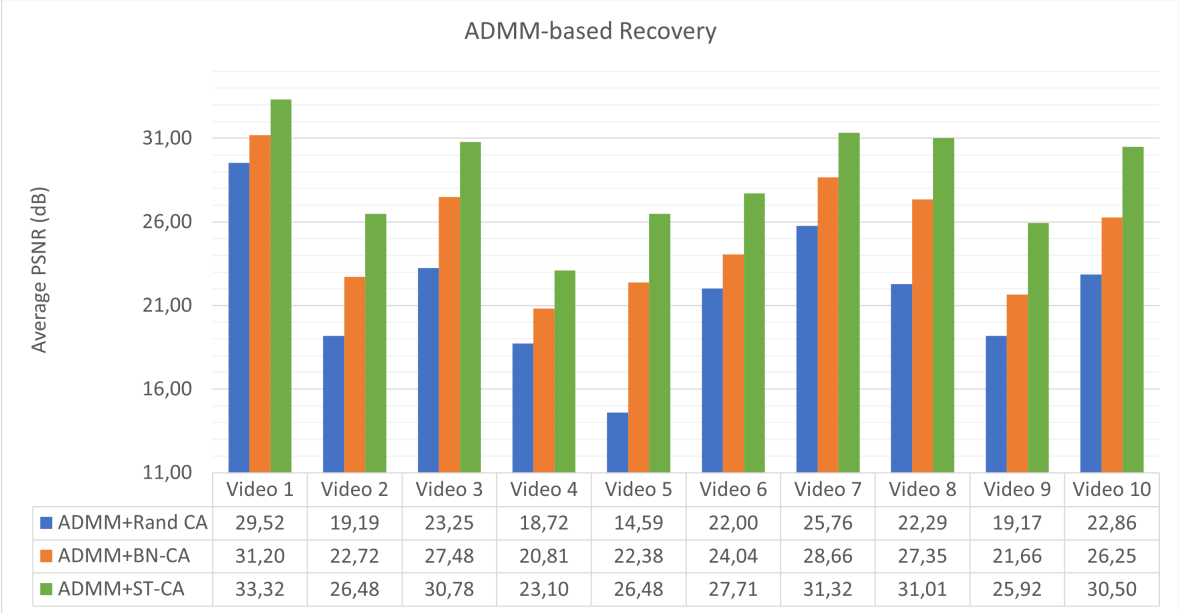


Figure 22. Illustration of three real sequences (i.e., *Campesina, toy car, and hat*) acquired in the Optics Lab of the HDSP research group. (a) RGB representation of three frames. (b) Subset of spectral bands from the last frame in (a).

$\ell_2 - \ell_1$ problem is employed with a 4D fixed sparse basis composed of the Kronecker product between a 2D Wavelet, a 1D discrete Cosine, and a 1D Wavelet basis [14]. The ADMM algorithm was run with a realization of random CA (denoted as ADMM+Rand CA) and the blue noise (BN) patterns (denoted as ADMM+BN CA) [16]. The window size parameter for the BN realization was set as 3×3 and $K = 2$ snapshots to assure a similar transmittance per frame respect to the other coded apertures of the comparison. Further, the BN CA is repeated and concatenated 4 times to cover the temporal window of 8 frames of the spectral video. Additionally, the learned CAs from the proposed E2E approach, named ST-CA, were used for running the ADMM to evaluate the different codification in the recovery algorithm (denoted as ADMM+ST-CA). The ADMM algorithm works with 1000 iterations and the parameters are set such that the best performance is obtained for each video and each CA. Finally, the proposed approach STNET is compared in simulations against the spatial Unet sub-network (Proposed E2E Unet) [76], where the CA and recovery are jointly trained as in the proposed approachbut the spectro-temporal CNN are removed. All the experiments were performed on noise-free measurements. The proposed STNET and the Unet were implemented on Tensorflow by using the Adam optimizer. The weights were initialized using a Gaussian distribution with standard deviation 0.05 and the batch-size was set as 24. The number of epochs is set to 1000 and the learning rate is varied between 10^{-3} and 10^{-4}. The reconstruction performance is evaluated in terms of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) index.

5.4.3. Evaluation on a Testing Dataset 1 Figure 23 shows the performance of the different methods and CAs compared to the proposed approaches. Note that the STNET network outperforms the iterative ADMM in around 5dB and the E2E Unet network in up to 1dB in terms of PSNR. Figure 24 illustrates the performance of the ADMM recovery by using the Rand-CA, BN-CA, and the ST-CAs, where can be

observed that the sensing based on the resulting ST-CAs provides a better sensing and thus a better reconstruction of the spectral videos.



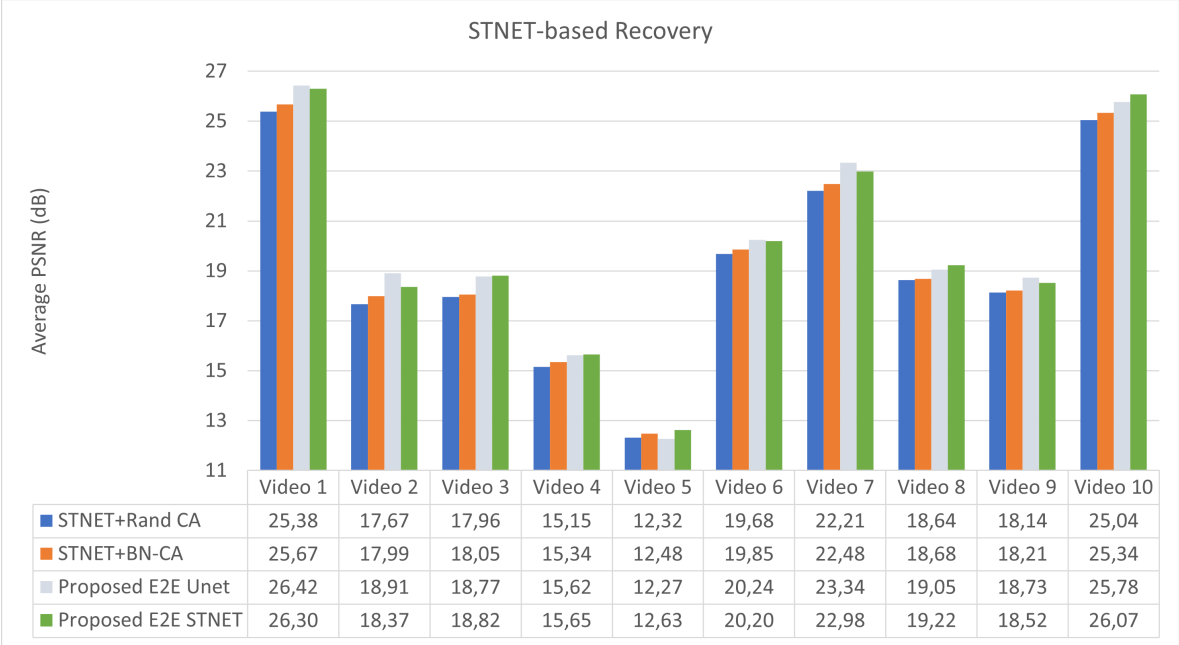
	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6	Video 7	Video 8	Video 9	Video 10
ADMM+Rand CA	29,52	19,19	23,25	18,72	14,59	22,00	25,76	22,29	19,17	22,86
ADMM+BN-CA	31,20	22,72	27,48	20,81	22,38	24,04	28,66	27,35	21,66	26,25
ADMM+ST-CA	33,32	26,48	30,78	23,10	26,48	27,71	31,32	31,01	25,92	30,50

Figure 23. Comparison results in terms of PSNR and SSIM by using different methods and CAs on 10 spectral videos.

In Figure 27 is illustrated an RGB profile of the fifth frame of each testing video (columns) and its corresponding reconstruction from the different methods (rows). As can be seen, the proposed approach overall outperforms the compared methods. Furthermore, observe that the spatial artifacts in the reconstruction are reduced by using the proposed approach.

5.4.4. Evaluation on a Testing Dataset 2 Table 6 shows the performance of the different methods and CAs compared to the proposed approaches. Note that the STNET network outperforms the iterative ADMM in around 5dB and the E2E Unet network in up to 1dB in terms of PSNR.

To illustrate the spectral accuracy of the proposed network, a spatial point along the spectral bands, aka spectral signature, is pictured in Figure 29 from three con-



	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6	Video 7	Video 8	Video 9	Video 10
STNET+Rand CA	25,38	17,67	17,96	15,15	12,32	19,68	22,21	18,64	18,14	25,04
STNET+BN-CA	25,67	17,99	18,05	15,34	12,48	19,85	22,48	18,68	18,21	25,34
Proposed E2E Unet	26,42	18,91	18,77	15,62	12,27	20,24	23,34	19,05	18,73	25,78
Proposed E2E STNET	26,30	18,37	18,82	15,65	12,63	20,20	22,98	19,22	18,52	26,07

Figure 24. ADMM recovery performance using the different coded apertures in terms of PSNR on the 10 testing spectral videos.

secutive frames of Video 1. Notice that the spectral signature resulting from the proposed method obtains higher SSIM values respect to the other methods at each point frame, demonstrating a better recovery of the data in the spectral axis.

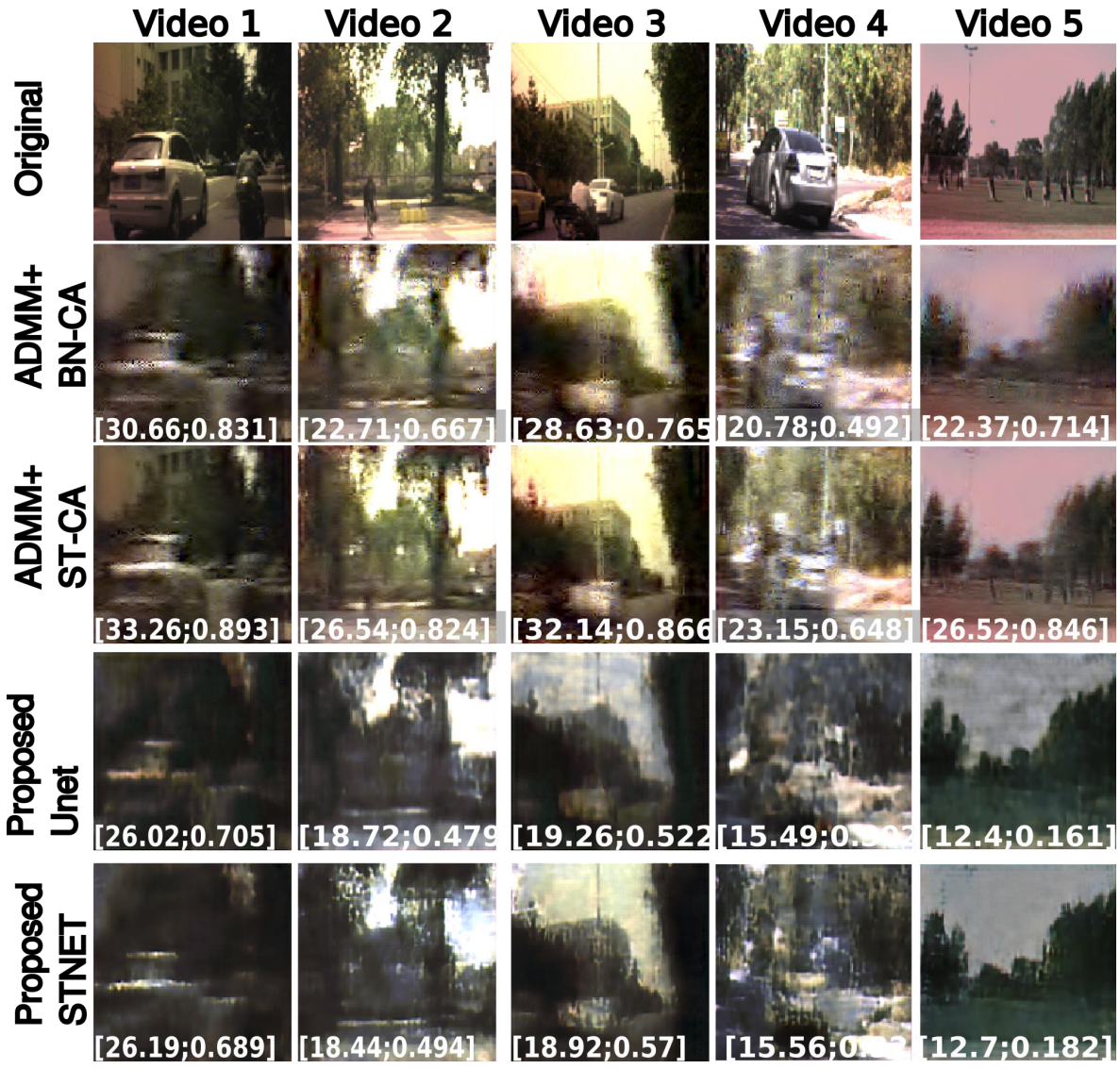


Figure 25. RGB profile of the original frame 5 (1st row) and the reconstructed frame of each testing video by using the different methods. The PSNR and SSIM values are shown for each given spectral frame.

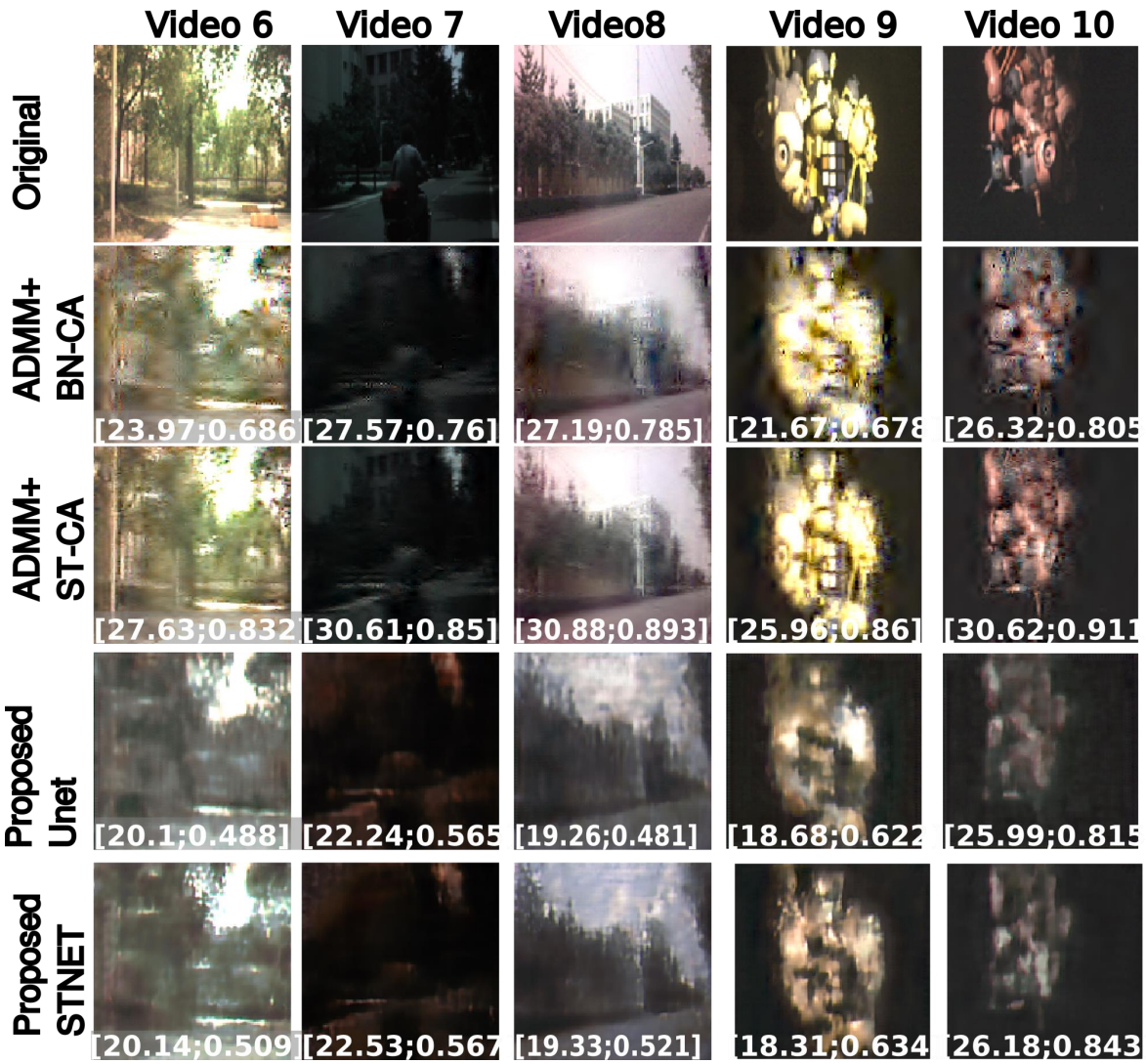


Figure 26. Continuation-Fig. 25.

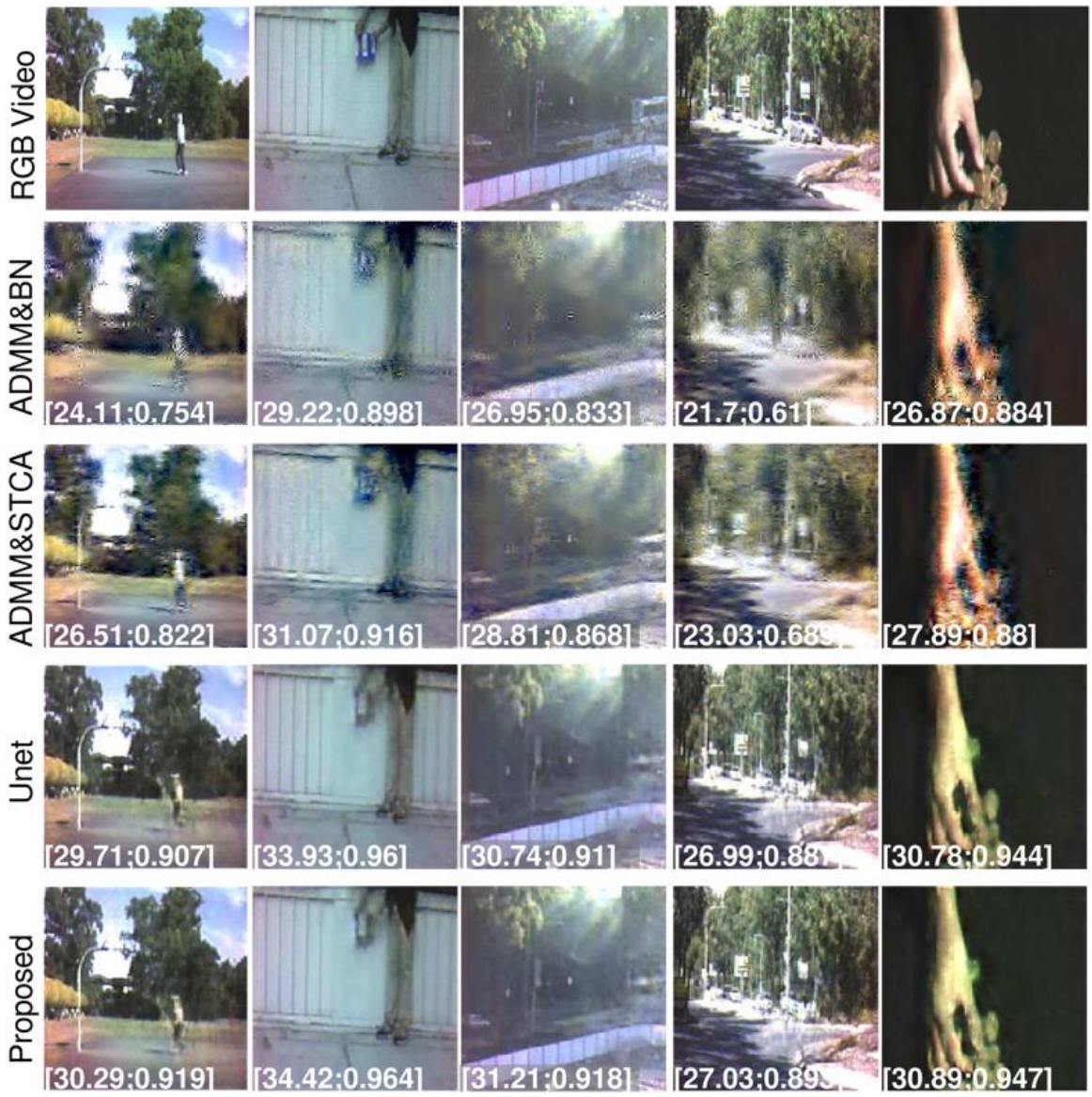


Figure 27. RGB profile of the original frame 5 (1st row) and the reconstructed frame of each testing video by using the different methods. The PSNR and SSIM values are shown for each given spectral frame.

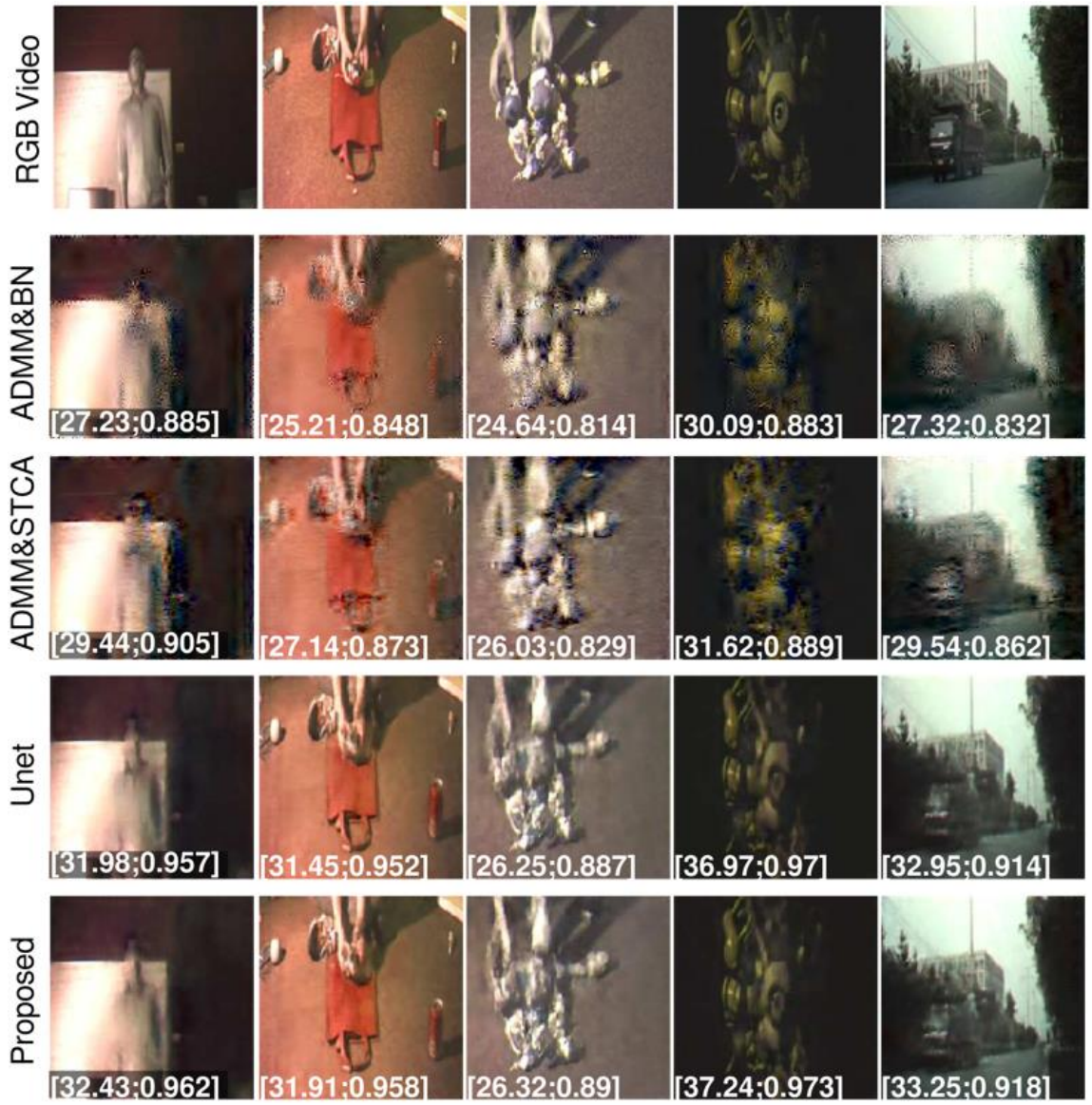


Figure 28. Continuation - Fig. 27.

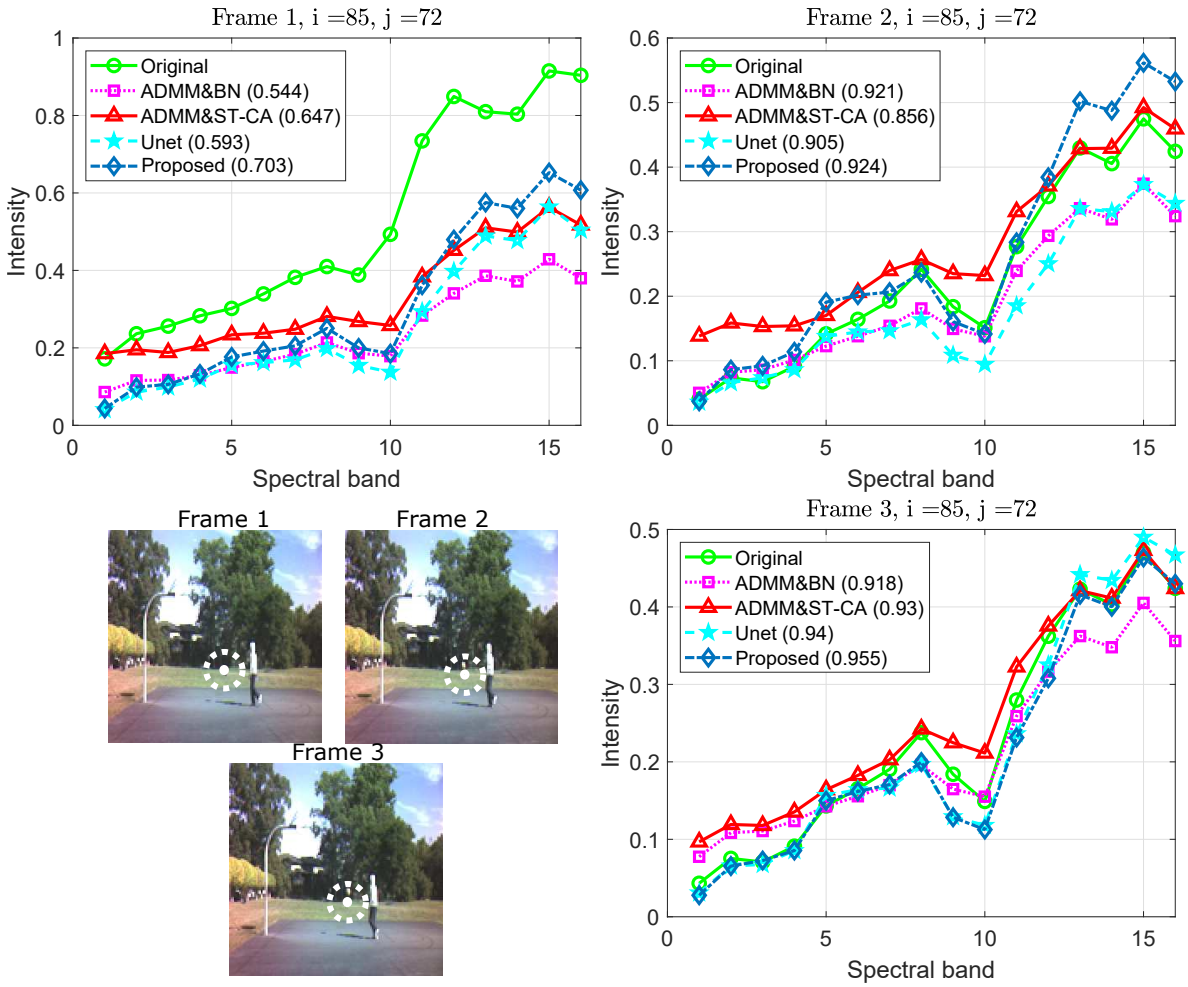


Figure 29. Spectral signature comparison for the different approaches across three consecutive frames in the Video 1. Observe that the selected point for showing the spectral signature is drawn in the given frame and points-out the basketball ball in the first frame.

Table 6. Comparison results in terms of PSNR and SSIM by using different methods and CAs on the Testing Dataset 2, spectral videos correlated to the Training Dataset.

Method	ADMM&BN		ADMM&ST-CA		Proposed E2E-Unet		Proposed STNET	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Video 1	23,80	0,749	26,16	0,818	29,41	0,906	**30,04**	**0,917**
Video 2	28,90	0,897	30,72	0,915	33,56	0,959	**34,17**	**0,964**
Video 3	27,12	0,834	28,94	0,869	30,87	0,911	**31,29**	**0,919**
Video 4	21,77	0,610	23,08	0,689	27,12	0,888	**27,14**	**0,894**
Video 5	26,49	0,879	27,54	0,877	30,73	0,943	**30,86**	**0,947**
Video 6	27,01	0,881	29,21	0,902	32,14	0,958	**32,45**	**0,963**
Video 7	25,09	0,844	27,09	0,871	31,49	0,954	**31,93**	**0,958**
Video 8	24,48	0,810	25,88	0,824	26,26	0,887	**26,28**	**0,890**
Video 9	30,08	0,881	31,64	0,887	36,90	0,969	**37,14**	**0,971**
Video 10	27,75	0,834	29,95	0,865	33,13	0,918	**33,48**	**0,923**

5.4.5. Evaluation on the Real Sequences Furthermore, for evaluating the CA design against the state-of-the-art BN, the dataset of Figure 22 was reconstructed using the ADMM algorithm with both the BN and the ST-CA patterns. Table 7 shows the obtained results from the ADMM using the different real sequences in terms of PSNR and SSIM. Observe that the designed ST-CA from spectral videos obtained better performances than the BN patterns, with gains of up to 4dB.

Table 7. CA evaluation of the blue noise CA and the ST-CA designs using the ADMM recovery procedure on a set of real sequences.

ADMM + CA	Campesina		Toy Car		Hat		Lego		Chiva	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
BN-CA	25,76	0,700	26,65	0,760	23,78	0,548	23,87	0,716	25,02	0,671
ST-CA	**27,52**	**0,779**	**30,66**	**0,849**	**25,67**	**0,655**	**25,98**	**0,797**	**27,48**	**0,774**

5.5. Conclusions

Inspired by existing works in DL for compressive image recovering, this Chapter introduces an E2E framework in compressive spectral video sensing to jointly learn

the CA pattern and the recovery method from the data. The proposed architecture relies on two main layers, the CSVS layer and the recovery layer, to train the weights for obtaining the best CA and reconstructed video. Numerical experiments using multispectral videos show that the proposed architecture outperforms the ADMM algorithm and the Unet network in up to 5dB and 1dB in terms of PSNR, respectively. Additionally, the designed CA set from the proposed E2E network is compared in the ADMM algorithm against the BN to reconstruct the set of real sequences, where a gain of up to 4dB was achieved. It can be noticed that the resulting CA is designed for a given time window period specified by the number of frames of the training database. This implies that it is used one measurement per frame for recovering the set of spectral data cubes, while temporal correlations are exploited in the recovery process. In this way, considering that state-of-the-art CA designs on compressive spectral imaging require multiple shots of the same scene to get high quality reconstructions, the proposed approach minimize the required number of measurements to obtain high quality results by doing a design based on the acquisition of a single shot per spectral frame.

As future work, it would be interesting to conduct experiments on real hardware, considering that the designed CAs are implementable on DMD devices. Moreover, further investigation should be conducted to evaluate the interest of the proposed approach for the joint CA design and reconstruction of spectral videos in others DL architectures such as unrolling networks. Another interesting future work includes the extension of the proposed E2E approach for designing colored CA (i.e., optical filters based CA) in the context of compressive spectral video sensing.

6. DISCUSSION AND CONCLUSION

In this thesis, we have presented a couple of approaches to jointly design the optical system and the recovery procedure for compressive spectral video acquisition and reconstruction. For the design, three key elements of the compressive spectral video pipeline were identified: the sparse representation, the coded aperture, and the recovery method. The results indicated that high quality reconstructions are obtained considering even only two of the three key elements such as in the tensor-based approach where is learned the sparse basis and the recovery. Nonetheless, by simultaneously designing the CA and the recovery, the learned pattern and the inference consider correlations across the different dimensions of the data. Indeed, the presented approaches demonstrated that, by exploiting the correlations across the multiple dimensions of spectral videos and into the higher-order array representation, outstanding reconstructions can be obtained from the compressed measurements by using only one shot per measurement.

The implementation of snapshot compressive imaging (SCI) systems to acquire spectral video (or CSVS) using a single snapshot is a concept that would allow unprecedented benefits in real and practical applications. While SCI-based research has focused on designing protocols for multiple snapshots, these results demonstrate that, into a time window, the sensing design across the temporal axis can be exploited for better sensing the scene under observation. These CSVS architectures are still a developing technology requiring high-dimensional encoding and recovering. While we have started to explore their optimal design, there still exists several issues to address, including the real implementation of these theoretically optimized systems. Even so, the advances and approaches presented in this thesis showed the potential of exploiting these kinds of signals for efficient acquisition and recovery, where high-quality recovered videos and low-time processing can be achieved.

Finally, the most important factor to advance technology beyond the limits of traditional approaches to spectral video acquisition is jointly designing the optical systems and recovery algorithms, which has been pursued in computational photography for several years. The approaches described in this thesis are key steps toward the next-generation of computational video cameras. Nonetheless, further research is needed to pave the way between the theoretical insights and the real system implementation.

Bibliography

Achanta, Radhakrishna et al. “SLIC superpixels compared to state-of-the-art superpixel methods”. In: *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012), pp. 2274–2282 (cit. on pp. 47, 48, 52, 53, 64).

Arce, Gonzalo et al. “Compressive coded aperture spectral imaging: An introduction”. In: *IEEE Signal Processing Magazine* 31.1 (2014), pp. 105–115 (cit. on p. 34).

Arguello, Henry and Gonzalo Arce. “Colored coded aperture design by concentration of measure in compressive spectral imaging”. In: *IEEE Transactions on Image Processing* 23.4 (2014), pp. 1896–1908 (cit. on pp. 39–41, 50, 51, 88, 92).

Atzberger, Clement. “Advances in remote sensing of agriculture: Context description, existing operational monitoring systems and major information needs”. In: *Remote sensing* 5.2 (2013), pp. 949–981 (cit. on p. 145).

Bacca, Jorge, Tatiana Gelvez, and Henry Arguello. “Deep Coded Aperture Design: An End-to-End Approach for Computational Imaging Tasks”. In: *arXiv preprint arXiv:2105.03390* (2021) (cit. on pp. 88, 92, 93).

Bannari, A et al. “A review of vegetation indices”. In: *Remote sensing reviews* 13.1-2 (1995), pp. 95–120 (cit. on p. 150).

Barajas-Solano, Crisostomo Alberto, Juan-Marcos Ramirez, and Henry Arguello. “Spectral Video Compression Using Convolutional Sparse Coding”. In: *2020 Data Compression Conference (DCC)*. IEEE. 2020, pp. 253–262 (cit. on pp. 23, 87).

Baret, Fred and Gerard Guyot. “Potentials and limits of vegetation indices for LAI and APAR assessment”. In: *Remote Sensing of Environment* 35.2 (1991), pp. 161–173 (cit. on p. 143).

Barrett, Richard et al. *Templates for the solution of linear systems: building blocks for iterative methods*. Vol. 43. Siam, 1994 (cit. on p. 59).

Benezeth, Yannick, Désiré Sidibé, and Jean-Baptiste Thomas. “Background subtraction with multispectral video sequences”. In: *IEEE International Conference on Robotics and Automation workshop on Non-classical Cameras, Camera Networks and Omnidirectional Vision (OMNIVIS)*. Hong Kong, China, 2014, p. 6 (cit. on pp. 21, 80).

Boyd, Stephen et al. “Distributed optimization and statistical learning via the alternating direction method of multipliers”. In: *Foundations and Trends® in Machine learning* 3.1 (2011), pp. 1–122 (cit. on p. 78).

Breunig, Markus M et al. “LOF: identifying density-based local outliers”. In: *Proc. ACM SIGMOD Int. Conf. on Management of Data*. Dallas, TX, USA, 2000, pp. 93–104 (cit. on p. 146).

Candès, Emmanuel J and Michael B Wakin. “An introduction to compressive sampling”. In: *IEEE signal processing magazine* 25.2 (2008), pp. 21–30 (cit. on pp. 30, 42, 44, 127).

Cao, Wenfei et al. “Total variation regularized tensor RPCA for background subtraction from compressive measurements”. In: *IEEE Transactions on Image Processing* 25.9 (2016), pp. 4075–4090 (cit. on pp. 45–47).

Cao, Xun et al. “Computational snapshot multispectral cameras: toward dynamic capture of the spectral world”. In: *IEEE Signal Processing Magazine* 33.5 (2016), pp. 95–108 (cit. on pp. 22, 31, 38, 45, 50, 55, 87).

Cao, Xun et al. “High resolution multispectral video capture with a hybrid camera system”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 297–304 (cit. on p. 31).

Chalapathy, Raghavendra and Sanjay Chawla. “Deep learning for anomaly detection: A survey”. In: *arXiv preprint arXiv:1901.03407* (2019) (cit. on pp. 145, 146).

Chandola, Varun, Arindam Banerjee, and Vipin Kumar. “Anomaly detection: A survey”. In: *ACM Computing Surveys (CSUR)* 41.3 (2009), pp. 1–58 (cit. on pp. 145, 154).

Chang, Chein-I and Shao-Shan Chiang. “Anomaly detection and classification for hyperspectral imagery”. In: *IEEE transactions on geoscience and remote sensing* 40.6 (2002), pp. 1314–1325 (cit. on p. 147).

Chang, Jason, Donglai Wei, and John W Fisher. “A video representation using temporal superpixels”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013, pp. 2051–2058 (cit. on pp. 47, 48, 53, 64).

Chawla, Nitesh V. et al. “SMOTE: Synthetic Minority over-Sampling Technique”. In: *J. Artif. Int. Res.* 16.1 (2002), 321–357 (cit. on p. 166).

Chen, Lulu et al. “Object Tracking in Hyperspectral-Oriented Video with Fast Spatial-Spectral Features”. In: *Remote Sensing* 13.10 (2021), p. 1922 (cit. on p. 21).

Cichocki, Andrzej et al. “Tensor decompositions for signal processing applications: From two-way to multiway component analysis”. In: *IEEE Signal Processing Magazine* 32.2 (2015), pp. 145–163 (cit. on p. 35).

Correa, Claudia, Henry Arguello, and Gonzalo Arce. “Spatiotemporal blue noise coded aperture design for multi-shot compressive spectral imaging”. In: *JOSA A* 33.12 (2016), pp. 2312–2322 (cit. on pp. 22, 23, 40, 42, 51, 55, 87, 88, 91, 98).

Correa-Pugliese, Claudia V, Diana F Galvis-Carreño, and Henry Arguello-Fuentes. “Sparse representations of dynamic scenes for compressive spectral video sensing”. In: *Dyna* 83.195 (2016), pp. 42–51 (cit. on pp. 31, 33, 34, 40, 42, 44, 63, 65, 81, 87, 91, 127).

Cristóbal, Gabriel, Peter Schelkens, and Hugo Thienpont. *Optical and digital image processing: fundamentals and applications*. John Wiley & Sons, 2013 (cit. on p. 30).

Daughtry, C. S. T. et al. “Estimating corn leaf chlorophyll concentration from leaf and canopy reflectance”. In: *Remote sensing of Environment* 74.2 (2000), pp. 229–239 (cit. on p. 150).

De Lathauwer, Lieven, Bart De Moor, and Joos Vandewalle. “A multilinear singular value decomposition”. In: *SIAM journal on Matrix Analysis and Applications* 21.4 (2000), pp. 1253–1278 (cit. on pp. 56, 62).

— “On the best rank-1 and rank-(r 1, r 2,..., rn) approximation of higher-order tensors”. In: *SIAM journal on Matrix Analysis and Applications* 21.4 (2000), pp. 1324–1342 (cit. on p. 79).

Defourny, Pierre et al. “Near real-time agriculture monitoring at national scale at parcel resolution: Performance assessment of the Sen2-Agri automated system in various cropping systems around the world”. In: *Remote sensing of environment* 221 (2019), pp. 551–568 (cit. on p. 146).

Ding, Xin, Wei Chen, and Ian J Wassell. “Joint sensing matrix and sparsifying dictionary optimization for tensor compressive sensing”. In: *IEEE Transactions on Signal Processing* 65.14 (2017), pp. 3632–3646 (cit. on p. 46).

Duarte, Marco F and Richard G Baraniuk. “Kronecker compressive sensing”. In: *IEEE Transactions on Image Processing* 21.2 (2012), pp. 494–504 (cit. on pp. 30, 46).

Elad, Michael. “Optimized projections for compressed sensing”. In: *IEEE Transactions on Signal Processing* 55.12 (2007), pp. 5695–5702 (cit. on p. 40).

Feng, Zhixi et al. “Superpixel Tensor Sparse Coding for Structural Hyperspectral Image Classification”. In: *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens* 10.4 (2017), pp. 1632–1639 (cit. on p. 46).

Figueiredo, Mário AT, Robert D Nowak, and Stephen J Wright. “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems”. In: *IEEE Journal of selected topics in signal processing* 1.4 (2007), pp. 586–597 (cit. on pp. 62, 81).

Foucart, Simon and Holger Rauhut. *A mathematical introduction to compressive sensing*. Vol. 1. 3. Birkhäuser Basel, 2013 (cit. on pp. 41, 42).

Friedland, Shmuel, Qun Li, and Dan Schonfeld. “Compressive sensing of sparse tensors.” In: *IEEE Trans. Image Processing* 23.10 (2014), pp. 4438–4447 (cit. on pp. 45, 46).

Galvis, Laura et al. “Coded aperture design in compressive spectral imaging based on side information”. In: *Applied optics* 56.22 (2017), pp. 6332–6340 (cit. on p. 81).

Gao, Bo-Cai. “NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space”. In: *Remote sensing of environment* 58.3 (1996), pp. 257–266 (cit. on p. 150).

García, Miguel A et al. “Using Hidden Markov Models for Land Surface Phenology: An Evaluation Across a Range of Land Cover Types in Southeast Spain”. In: *Remote Sensing* 11.5 (2019), p. 507 (cit. on p. 143).

Gedalin, Daniel, Yaniv Oiknine, and Adrian Stern. “DeepCubeNet: reconstruction of spectrally compressive sensed hyperspectral images with deep neural networks”. In: *Optics express* 27.24 (2019), pp. 35811–35822 (cit. on p. 43).

Gómez, Cristina, Joanne C White, and Michael A Wulder. “Optical remotely sensed time series data for land cover classification:A review”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 116 (2016), pp. 55–72 (cit. on p. 143).

Görnitz, Nico, Mikio Braun, and Marius Kloft. “Hidden Markov anomaly detection”. In: *International Conference on Machine Learning*. Lille, France, Jan. 2015, pp. 1833–1842 (cit. on p. 147).

Hagolle, Olivier et al. “A multi-temporal and multi-spectral method to estimate aerosol optical thickness over land, for the atmospheric correction of FormoSat-2, Land-

Sat, VENμS and Sentinel-2 images". In: *Remote Sensing* 7.3 (2015), pp. 2668–2691 (cit. on p. 159).

Higham, Catherine F et al. "Deep learning for real-time single-pixel video". In: *Scientific reports* 8.1 (2018), pp. 1–9 (cit. on p. 92).

Hong, Tao and Zhihui Zhu. "An efficient method for robust projection matrix design". In: *Signal Processing* 143 (2018), pp. 200–210 (cit. on p. 40).

Hu, Xuemei et al. "Multispectral video acquisition using spectral sweep camera". In: *Optics express* 27.19 (2019), pp. 27088–27102 (cit. on pp. 21, 22).

Iliadis, Michael, Leonidas Spinoulas, and Aggelos K Katsaggelos. "Deep fully-connected networks for video compressive sensing". In: *Digital Signal Processing* 72 (2018), pp. 9–18 (cit. on pp. 23, 43, 88).

Jaakkola, Tommi, Mark Diekhans, and David Haussler. "Using the Fisher kernel method to detect remote protein homologies." In: *Proc. Int. Conf. on Intelligent Systems for Molecular Biology*. Heidelberg, Germany, Aug. 1999, pp. 149–158 (cit. on p. 164).

Kay, Steven M. *Fundamentals of statistical signal processing*. Prentice Hall signal processing series. Upper Saddle River, NJ: Prentice Hall PTR, 1993 (cit. on p. 156).

Kocur-Bera, Katarzyna. "Understanding information about agricultural land. An evaluation of the extent of data modification in the Land Parcel Identification System for the needs of area-based payments–a case study". In: *Land Use Policy* 94 (2020), p. 104527 (cit. on p. 150).

Kolda, Tamara G and Brett W Bader. “Tensor decompositions and applications”. In: *SIAM review* 51.3 (2009), pp. 455–500 (cit. on pp. 35, 45, 49).

Kramer, Mark A. “Nonlinear principal component analysis using autoassociative neural networks”. In: *AIChE journal* 37.2 (1991), pp. 233–243 (cit. on p. 146).

Kriegel, Hans-Peter et al. “LoOP: local outlier probabilities”. In: *Proc 18th ACM Conf. Inform. Knowl. Manage. (CIKM ’09)*. Hong Kong, China, 2009, 1649–1652 (cit. on p. 146).

Leite Cerqueira, Paula Beatriz et al. “Hidden Markov Models for crop recognition in remote sensing image sequences”. In: *Pattern Recognition Letters* 32.1 (2011), pp. 19–26 (cit. on pp. 144, 157).

Leitner, Raimund et al. “Multi-spectral video endoscopy system for the detection of cancerous tissue”. In: *Pattern Recognition Letters* 34.1 (2013), pp. 85–93 (cit. on p. 21).

León-López, Kareth, Laura Galvis, and Henry Arguello. “Temporal Colored Coded Aperture Design in Compressive Spectral Video Sensing”. In: *IEEE Transactions on Image Processing* 28.1 (2018), pp. 253–264 (cit. on pp. 21–23, 31, 37–42, 50, 51, 54, 63, 65, 76, 80, 81, 87, 88, 98).

León-López, Kareth, Laura Galvis, and Henry Arguello Fuentes. “Spatio-spectro-temporal coded aperture design for multiresolution compressive spectral video sensing”. In: *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE. 2017, pp. 728–732 (cit. on p. 88).

León-López, Kareth et al. “Higher-Order Tensor Sparse Representation for Video-Rate Coded Aperture Snapshot Spectral Image Reconstruction”. In: *2019 IEEE*

8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP). IEEE. 2019, pp. 704–708 (cit. on pp. 22, 23, 87).

León-López, Kareth M and Henry Arguello Fuentes. “Online Tensor Sparsifying Transform Based on Temporal Superpixels From Compressive Spectral Video Measurements”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 5953–5963 (cit. on pp. 21, 23, 87, 98).

León-López, Kareth M. et al. “Anomaly Detection and Classification in Multispectral Time Series based on Hidden Markov Models”. In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–11. DOI: 10.1109/TGRS.2021.3101127 (cit. on pp. 22, 158).

León-López, Kareth Marcela. “DISEÑO DE APERTURAS DE CODIFICACIÓN PARA LA ADQUISICIÓN COMPRESIVA DE IMÁGENES ESPECTRALES DINÁMICAS [recurso electronico]”. M.S. Thesis. Bucaramanga, Colombia: Universidad Industrial de Santader (UIS), 2017 (cit. on pp. 34, 127).

Li, Gang et al. “Designing robust sensing matrix for image compression”. In: *IEEE Transactions on Image Processing* 24.12 (2015), pp. 5389–5400 (cit. on p. 40).

Li, Gang et al. “On projection matrix optimization for compressive sensing systems”. In: *IEEE Transactions on Signal Processing* 61.11 (2013), pp. 2887–2898 (cit. on p. 40).

Li, Jinbo, Witold Pedrycz, and Iqbal Jamal. “Multivariate time series anomaly detection: A framework of Hidden Markov Models”. In: *Applied Soft Computing* 60 (2017), pp. 229–240 (cit. on p. 147).

Li, Yuqi et al. “End-to-end video compressive sensing using anderson-accelerated unrolled networks”. In: *2020 IEEE International Conference on Computational Photography (ICCP)*. IEEE. 2020, pp. 1–12 (cit. on pp. 23, 43, 88).

Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation forest”. In: *Proc. Int. Conf. on Data Mining*. Pisa, Italy, Dec. 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17 (cit. on p. 147).

Liu, Xuefeng, Salah Bourennane, and Caroline Fossati. “Denoising of hyperspectral images using the PARAFAC model and statistical performance analysis”. In: *IEEE Transactions on Geoscience and Remote Sensing* 50.10 (2012), pp. 3717–3724 (cit. on p. 46).

Ma, Chenguang et al. “Acquisition of high spatial and spectral resolution video with a hybrid camera system”. In: *International journal of computer vision* 110.2 (2014), pp. 141–155 (cit. on p. 31).

Ma, Jiawei et al. “Deep tensor admm-net for snapshot compressive imaging”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 10223–10232 (cit. on p. 88).

Markou, Markos and Sameer Singh. “Novelty detection: a review—part 1: statistical approaches”. In: *Signal Processing* 83.12 (2003), pp. 2481–2497 (cit. on pp. 145, 146).

Marquez, Miguel, Hoover Rueda-Chacon, and Henry Arguello. “Compressive Spectral Light Field Image Reconstruction via Online Tensor Representation”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3558–3568 (cit. on p. 46).

McFeeters, Stuart K. “The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features”. In: *International journal of remote sensing* 17.7 (1996), pp. 1425–1432 (cit. on p. 150).

Mejia, Yuri and Henry Arguello. “Binary Codification Design for Compressive Imaging by Uniform Sensing”. In: *IEEE Transactions on Image Processing* 27.12 (2018), pp. 5775–5786 (cit. on p. 41).

Melgani, F. and L. Bruzzone. “Classification of hyperspectral remote sensing images with support vector machines”. In: *IEEE Transactions on Geoscience and Remote Sensing* 42.8 (2004), pp. 1778–1790 (cit. on p. 166).

Meng, Ziyi, Jiawei Ma, and Xin Yuan. “End-to-end low cost compressive spectral imaging with spatial-spectral self-attention”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 187–204 (cit. on pp. 22, 24, 43, 87, 88, 91).

Meroni, Michele et al. “Near real-time vegetation anomaly detection with MODIS NDVI: Timeliness vs. accuracy and effect of anomaly computation options”. In: *Remote Sensing of Environment* 221 (2019), pp. 508–521 (cit. on pp. 144, 147, 148).

Mian, Ajmal and Richard Hartley. “Hyperspectral video restoration using optical flow and sparse coding”. In: *Optics express* 20.10 (2012), pp. 10658–10673 (cit. on pp. 33, 44, 63, 80).

Miao, Xin et al. “l-net: Reconstruct hyperspectral images from a snapshot measurement”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 4059–4069 (cit. on p. 43).

Motohka, Takeshi et al. “Applicability of green-red vegetation index for remote sensing of vegetation phenology”. In: *Remote Sensing* 2.10 (2010), pp. 2369–2387 (cit. on p. 150).

Mouret, Florian et al. “Outlier Detection at the Parcel-Level in Wheat and Rapeseed Crops Using Multispectral and SAR Time Series”. In: *Remote Sensing* 13.5 (2021). DOI: 10.3390/rs13050956 (cit. on pp. 146, 148, 151, 160, 161).

Parada-Mayorga, Alejandro and Gonzalo R Arce. “Colored Coded Aperture Design in Compressive Spectral Imaging via Minimum Coherence”. In: *IEEE Transactions on Computational Imaging* 3.2 (2017), pp. 202–216 (cit. on pp. 42, 88).

Peters, Albert J et al. “Drought monitoring with NDVI-based standardized vegetation index”. In: *Photogrammetric Engineering and Remote sensing* 68.1 (2002), pp. 71–75 (cit. on pp. 143, 144).

Pimentel, Marco AF et al. “A review of novelty detection”. In: *Signal Processing* 99 (2014), pp. 215–249 (cit. on pp. 145, 146).

Pinilla, Samuel et al. “Salient Motion Detection for Spectral Video on the Compressive Domain”. In: *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*. IEEE. 2019, pp. 106–110 (cit. on pp. 22, 87).

Prendes, Jorge et al. “A Bayesian nonparametric model coupled with a Markov random field for change detection in heterogeneous remote sensing images”. In: *SIAM Journal on Imaging Sciences* 9.4 (2016), pp. 1889–1921 (cit. on p. 143).

Qaisar, Saad et al. “Compressive sensing: From theory to applications, a survey”. In: *Journal of Communications and networks* 15.5 (2013), pp. 443–456 (cit. on pp. 34, 127).

Qiao, Mu et al. “Deep learning for video compressive sensing”. In: *APL Photonics* 5.3 (2020), p. 030801 (cit. on pp. 23, 43, 88).

Rabiner, Lawrence R. “A tutorial on hidden Markov models and selected applications in speech recognition”. In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286 (cit. on pp. 144, 152–156).

Ramaswamy, Sridhar, Rajeev Rastogi, and Kyuseok Shim. “Efficient algorithms for mining outliers from large data sets”. In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 427–438 (cit. on p. 147).

Ramirez, J. M. and H. Arguello. “Spectral Image Classification From Multi-Sensor Compressive Measurements”. In: *IEEE Transactions on Geoscience and Remote Sensing* 58.1 (2020), pp. 626–636. DOI: 10.1109/TGRS.2019.2938724 (cit. on p. 169).

Rembold, Felix et al. “ASAP: A new global early warning system to detect anomaly hot spots of agricultural production for food security analysis”. In: *Agricultural systems* 168 (2019), pp. 247–257 (cit. on p. 147).

Ronneberger, O., P.Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Vol. 9351. LNCS. Springer, 2015, pp. 234–241 (cit. on pp. 93, 98).

Rouse, JW et al. “Monitoring vegetation systems in the Great Plains with ERTS”. In: *NASA special publication* 351.1974 (1974), p. 309 (cit. on p. 150).

Rueda, Hoover, Henry Arguello, and Gonzalo Arce. “DMD-based implementation of patterned optical filter arrays for compressive spectral imaging”. In: *JOSA A* 32.1 (2015), pp. 80–89 (cit. on p. 137).

Schölkopf, Bernhard et al. “Estimating the support of a high-dimensional distribution”. In: *Neural Computation* 13.7 (2001), pp. 1443–1471 (cit. on p. 147).

Shen, Yonglin et al. “Hidden Markov models for real-time estimation of corn progress stages using MODIS and meteorological data”. In: *Remote Sensing* 5.4 (2013), pp. 1734–1753 (cit. on p. 144).

Siachalou, Sofia, Giorgos Mallinis, and Maria Tsakiri-Strati. “A Hidden Markov Models Approach for Crop Classification: Linking Crop Phenology to Time Series of Multi-Sensor Remote Sensing Data”. In: *Remote Sensing Letters* 7.4 (2015), pp. 3633–3650 (cit. on pp. 144, 152).

— “Analysis of Time-Series spectral index data to enhance crop identification over a Mediterranean rural landscape”. In: *IEEE Geoscience and Remote Sensing Letters* 14.9 (2017), pp. 1508–1512. DOI: 10.1109/LGRS.2017.2719124 (cit. on pp. 144, 148, 152).

Sobral, Andrews et al. “Online stochastic tensor decomposition for background subtraction in multispectral video sequences”. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2015, pp. 106–113 (cit. on p. 21).

Vargas, H. and H. Arguello. “A Low-Rank Model for Compressive Spectral Image Classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 57.12 (2019), pp. 9888–9899 (cit. on p. 169).

Vargas, Héctor, Ariolfo Camacho, and Henry Arguello. “Spectral unmixing approach in hyperspectral remote sensing: a tool for oil palm mapping”. In: *TecnoLógicas* 22.45 (2019), pp. 131–145 (cit. on p. 145).

Veloso, Amanda et al. “Understanding the temporal behavior of crops using Sentinel-1 and Sentinel-2-like data for agricultural applications”. In: *Remote Sensing of Environment* 199 (2017), pp. 415–426 (cit. on pp. 143, 145, 161).

Venteris, Erik R et al. “Detection of anomalous crop condition and soil variability mapping using a 26 year Landsat record and the Palmer crop moisture index”. In: *International journal of applied earth observation and geoinformation* 39 (2015), pp. 160–170 (cit. on pp. 147, 148).

Villa-Pérez, Miryam Elizabeth et al. “Semi-supervised anomaly detection algorithms: A comparative summary and future research directions”. In: *Knowledge-Based Systems* (2021), p. 106878 (cit. on pp. 145, 146).

Viovy, Nicolas and Gilbert Saint. “Hidden Markov models applied to vegetation dynamics analysis using satellite remote sensing”. In: *IEEE Transactions on Geoscience and Remote Sensing* 32.4 (1994), pp. 906–917 (cit. on pp. 144, 152, 157).

Wagadarikar, Ashwin et al. “Single disperser design for coded aperture snapshot spectral imaging”. In: *Applied optics* 47.10 (2008), B44–B51 (cit. on pp. 30, 31, 37).

Wagadarikar, Ashwin A et al. “Video rate spectral imaging using a coded aperture snapshot spectral imager”. In: *Optics express* 17.8 (2009), pp. 6368–6388 (cit. on pp. 22, 23, 30, 31, 33, 40, 44, 87).

Wang, Lizhi et al. “High-speed hyperspectral video acquisition by combining Nyquist and compressive sampling”. In: *IEEE transactions on pattern analysis and machine intelligence* 41.4 (2019), pp. 857–870 (cit. on pp. 21–23, 31, 34, 44, 45, 65, 87).

Wang, Lizhi et al. “High-speed hyperspectral video acquisition with a dual-camera architecture”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 4942–4950 (cit. on p. 33).

Wen, Bihan, Saiprasad Ravishankar, and Yoram Bresler. “VIDOSAT: High-Dimensional Sparsifying Transform Learning for Online Video Denoising”. In: *IEEE Transactions on Image Processing* 28.4 (2019), pp. 1691–1704 (cit. on p. 47).

Wright, Stephen J, Robert D Nowak, and Mário AT Figueiredo. “Sparse reconstruction by separable approximation”. In: *IEEE Transactions on Signal Processing* 57.7 (2009), pp. 2479–2493 (cit. on pp. 62, 63).

Xiong, Fengchao, Jun Zhou, and Yuntao Qian. “Material based object tracking in hyperspectral videos”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3719–3733 (cit. on pp. 21, 22, 93, 95).

Yasuma, Fumihito et al. *CAVE Projects: Multispectral Image Database*. 2008. URL: http://www.cs.columbia.edu/CAVE/databases/multispectral/ (cit. on p. 80).

Yu, Yeyang et al. “Multidimensional compressed sensing MRI using tensor decomposition-based sparsifying transform”. In: *PloS one* 9.6 (2014), e98441 (cit. on p. 46).

Yuan, Xin, David J Brady, and Aggelos K Katsaggelos. “Snapshot compressive imaging: Theory, algorithms, and applications”. In: *IEEE Signal Processing Magazine* 38.2 (2021), pp. 65–88 (cit. on p. 24).

Yuan, Yuan et al. “Continuous change detection and classification using hidden Markov model: A case study for monitoring urban encroachment onto farmland in Beijing”. In: *Remote Sensing* 7.11 (2015), pp. 15318–15339 (cit. on p. 144).

Zeng, Linglin et al. “A review of vegetation phenological metrics extraction using time-series, multispectral satellite data”. In: *Remote Sensing of Environment* 237 (2020), p. 111511 (cit. on pp. 143, 169).

Zhang, Lefei et al. “Compression of hyperspectral remote sensing images by tensor approach”. In: *Neurocomputing* 147 (2015), pp. 358–363 (cit. on p. 46).

Zhang, Qiang et al. “Missing data reconstruction in remote sensing image with a unified spatial–temporal–spectral deep convolutional neural network”. In: *IEEE Transactions on Geoscience and Remote Sensing* 56.8 (2018), pp. 4274–4288 (cit. on p. 22).

Zou, Hui, Trevor Hastie, and Robert Tibshirani. “Sparse principal component analysis”. In: *Journal of computational and graphical statistics* 15.2 (2006), pp. 265–286 (cit. on p. 61).

ANNEXES

Annex A. SPARSITY ANALYSIS

State-of-the-Art Sparse Transform Bases

The results of the sparsity analysis performed on [32] are reproduced in this Annex to show the performance of sparse transforms on the spectral videos under study.
Based on the CS literature, a signal is said to be S-sparse if it has only S non-zero coefficients. Otherwise, if a large number of coefficients are small enough to be ignored, the signal is said to be *compressible* [3324]. Given that spectral videos are compressible in some sparse transforms, one way to evaluate the compression capabilities of the sparse transforms is by considering only few coefficients on the basis to represent the data. Specifically, this procedure consists on keeping a given percentage of the largest absolute coefficients in the basis and using them in the inverse basis to estimate an approximation of the original signal [27].
The videos shown in Figure 30 are used to evaluate the compression capabilities of a set of bases from the state-of-the-art which are combinations of Wavelet (W) and discrete Cosine transforms (D) transforms. For the comparison, the following combinations were used for the spatial, spectral, and temporal dimensions, respectively: WWDD, WWWW, WWWD, WWDW. Figures 31 and 32 shows the PSNR and average SSIM performances for the different bases. It can be noticed that in Fig.31 even though the WWDW obtains a comparable performance against the WWDD in terms of PSNR for low percentage values, the WWDD sparsification leads better SSIM values. On the other hand, from the Windows video sparsification, it can be seen that the best performance is obtained by using WWDD.

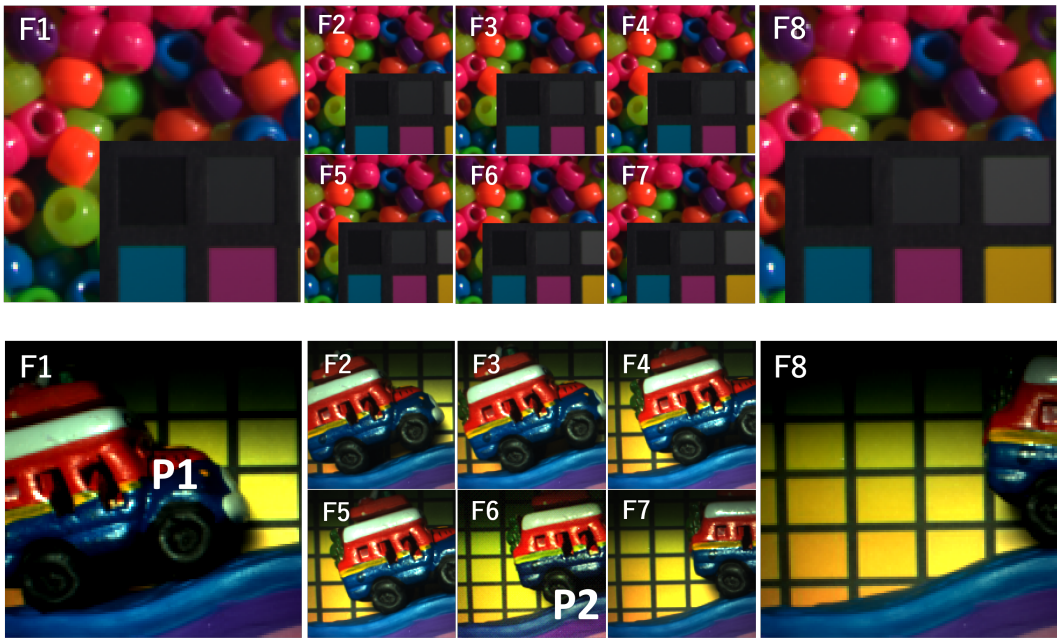


Figure 30. RGB representation of the spectral videos used for the sparsity analysis with 128×128 spatial pixel, 8 frames and 8 spectral bands, so called Windows (top) and Chiva (bottom) videos.

Complementary Comparison of the Proposed Tensor Transform

A sparsity analysis in terms of quality of reconstruction given the percentage of coefficients is presented in Figure 33 by using the 'Chiva' spectral video (or Video 3 from Chapter 3). For comparison purposes, the proposed tensor representation (proposed in Chapter 3) is employed in a full-data way (TenDL), in TSP patches (TSP-TenDL) and in regular patches (RegP-TenDL), i.e., the spatial dimension is divided in a regular grid of n_r sub tensors, for the example, it was used $n_r = 4$. Additionally, the regular patches are used for estimating the WWDD Kronecker transformation (RegP-WWDD) and all the mentioned bases are compared against the Kronecker from the full-data (WWDD). As can be noticed, the tensor-based transforms obtain higher PSNR and SSIM even when the coefficients are only the 0.1% of the total amount of data. Additionally, as presented in Chapter 3, the TSP-TenDL transform

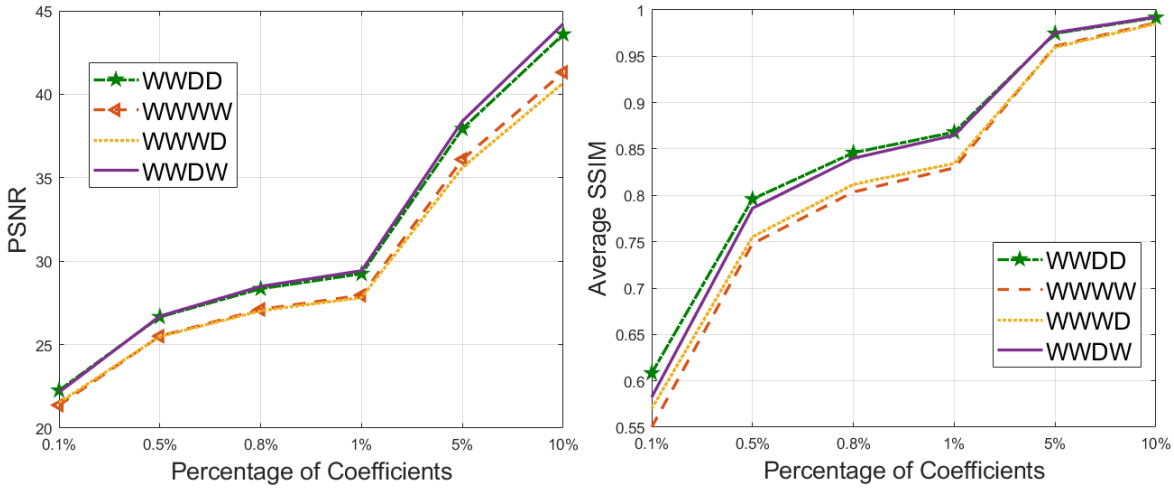


Figure 31. Evaluation of the compression capabilities of state-of-the-art bases for the ‘Windows’ spectral video

outperforms the other sparse representation bases.

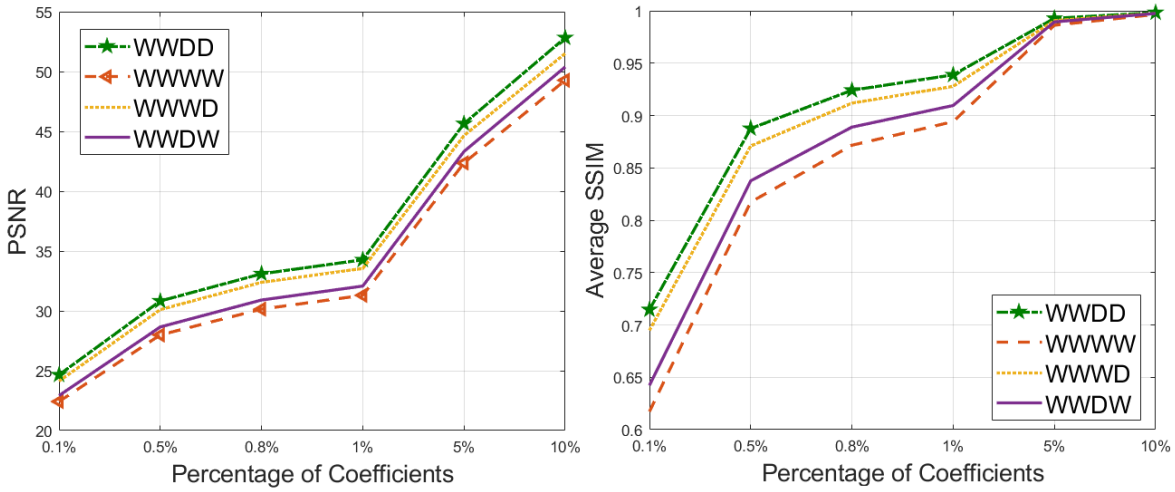


Figure 32. Evaluation of the compression capabilities of state-of-the-art bases for the ‘Chiva’ spectral video.

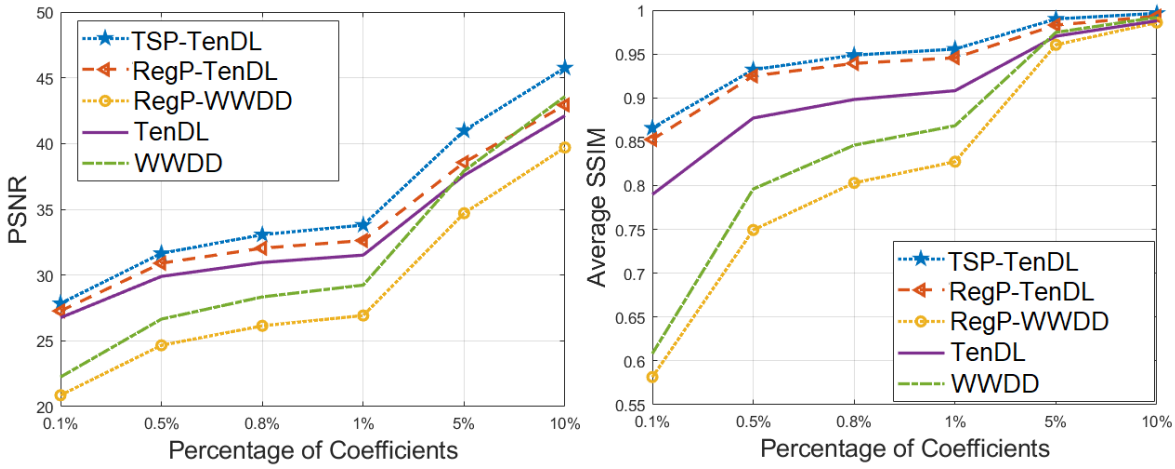


Figure 33. Evaluation of the compression capabilities for the different sparse representations in terms of PSNR and SSIM respect to the percentage of coefficients used for estimating the ‘Chiva’ spectral video.

Annex B. COMPLEMENTARY RESULTS: CHAPTER 3

Additional results for showing the spectral information of the reconstructed videos are presented as follows.

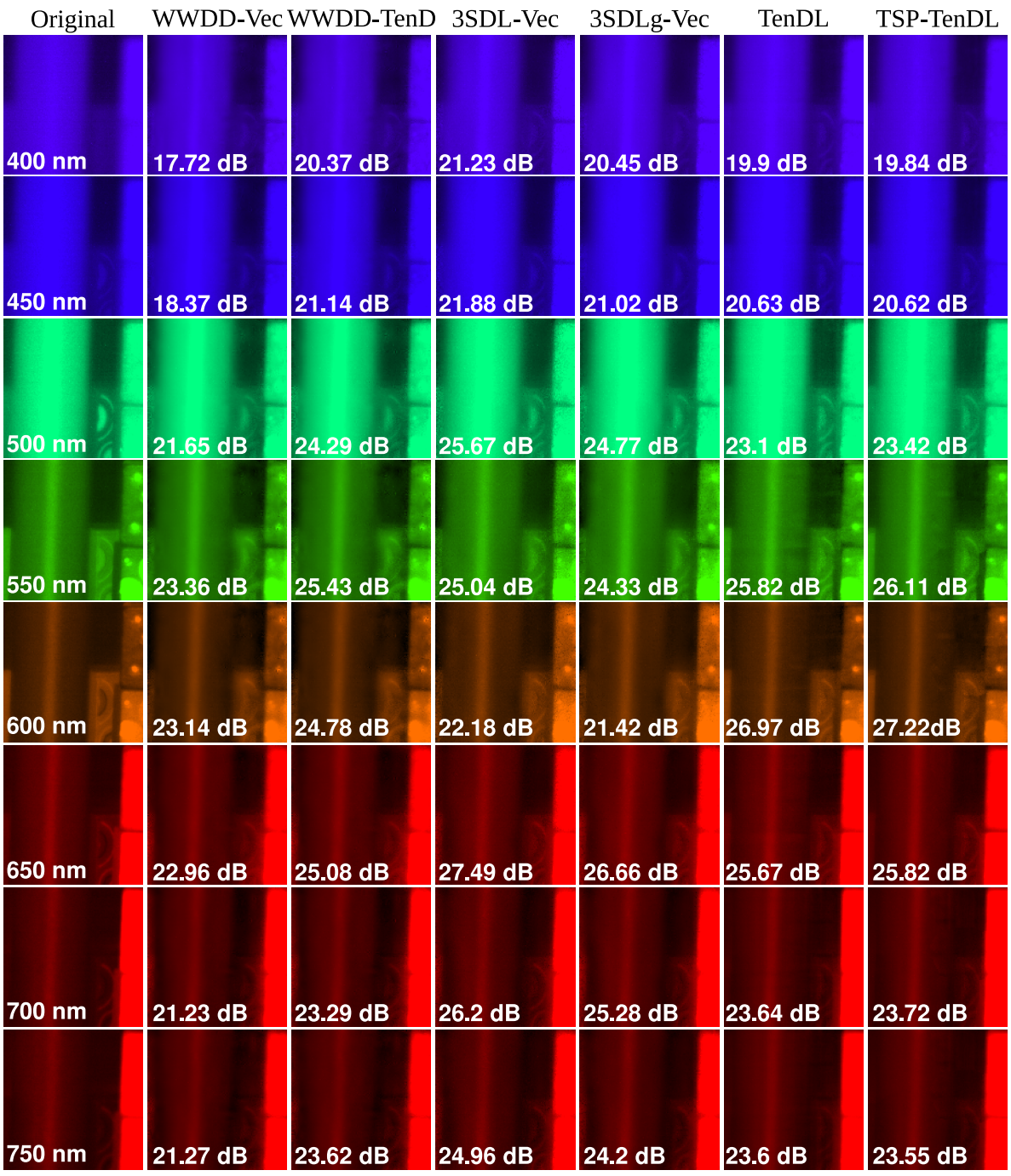


Figure 34. Spectral bands of the frame 1 from the original and reconstructions of the spectral video 1.

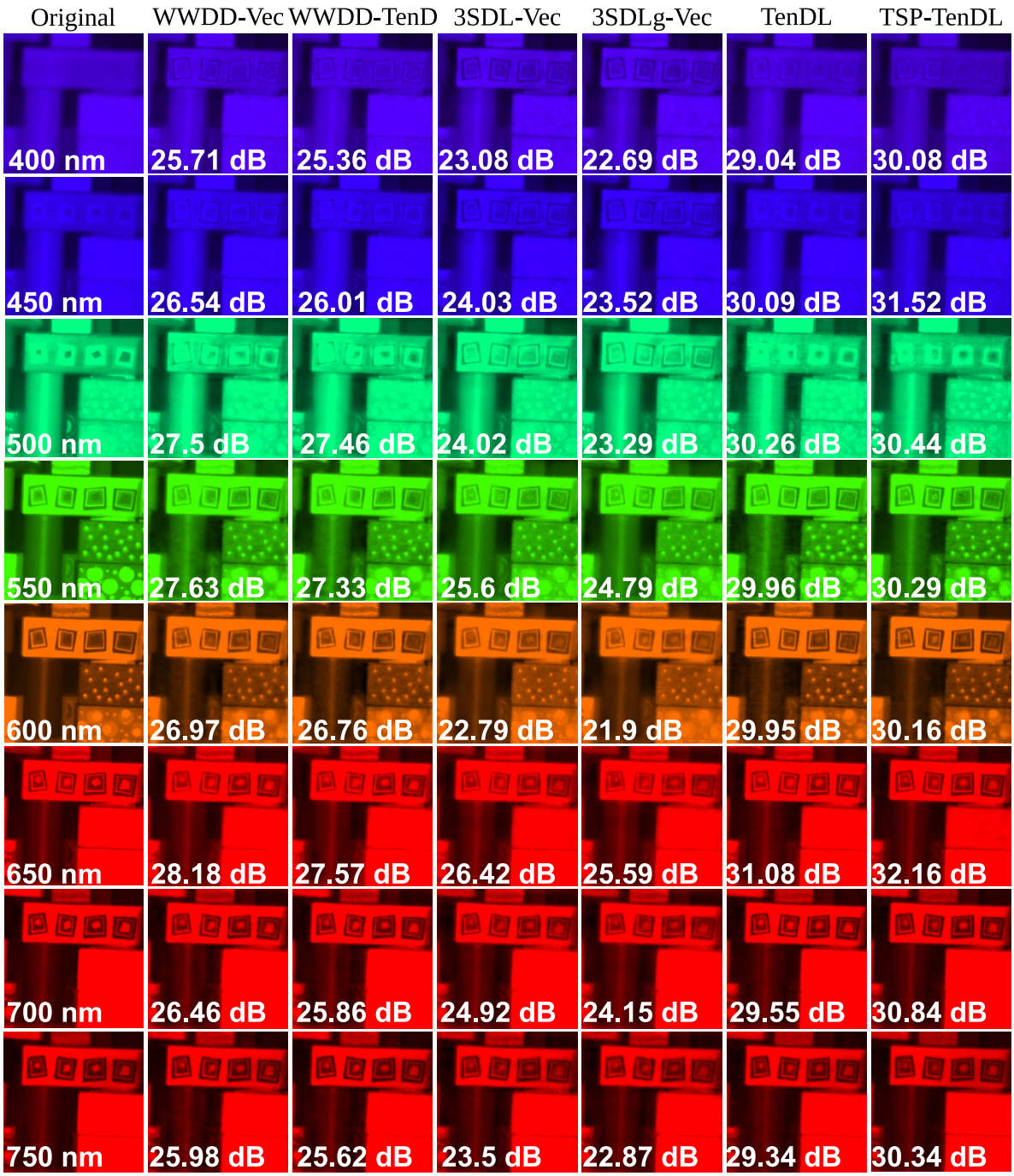


Figure 35. Spectral bands of the frame 1 from the original and reconstructions of the spectral video 2.

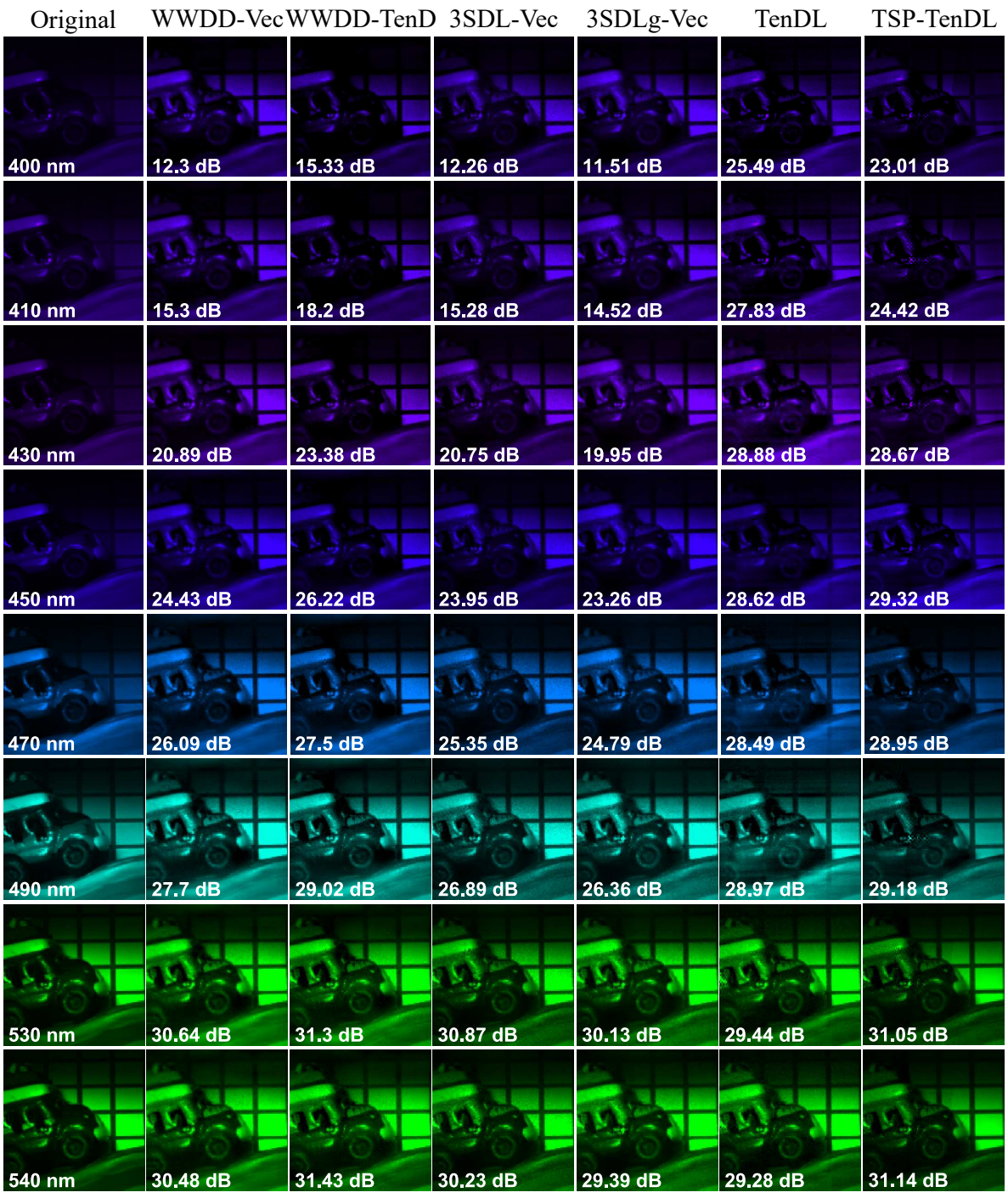


Figure 36. Spectral bands of the frame 1 from reconstructions of the spectral video 3 with L=24 - continuation

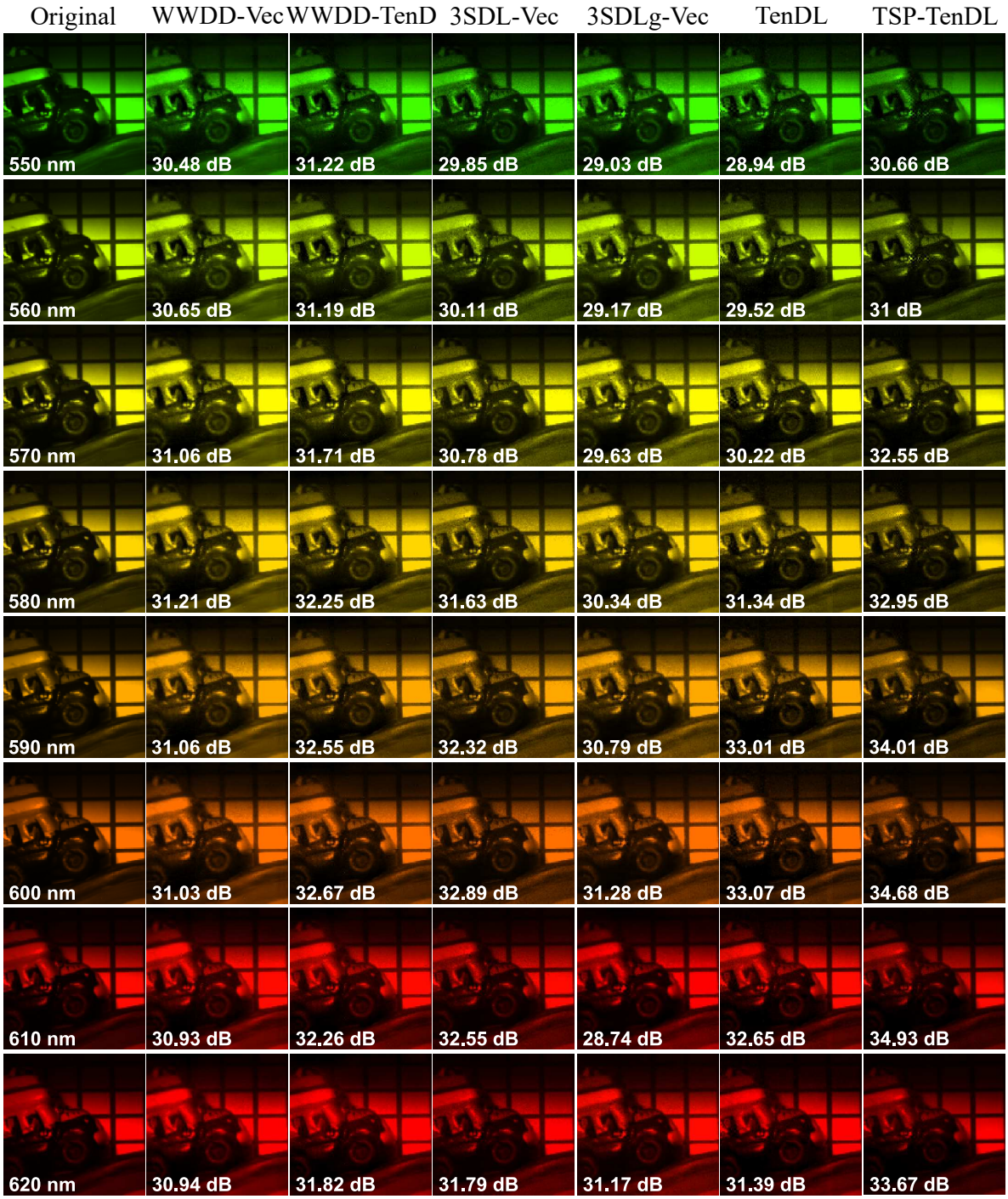


Figure 37. Spectral bands of the frame 1 from reconstructions of the spectral video 3 with L=24 - continuation

Figure 38. Spectral bands of the frame 1 from reconstructions of the spectral video 3 with L=24 - continuation

Annex C. LABORATORY PROTOTYPE

Taking advantage of the Optics laboratory of the HDSP research group, a lab prototype camera to acquire a spectral video dataset was mounted with the help of the laboratory team by using the following optical elements[79]: (1) An objective lens, (2) a monochromatic FPA detector, (3) a monochromatic light source [80]. And, for the vertical and circular movements of the objects, two Thorlabs devices were used: a single-axis translation stage with standard micrometer device, and a high-precision rotation mount for 25.4 mm device, respectively. Figure 39 shows the prototype built on the optics laboratory of the HDSP research group.

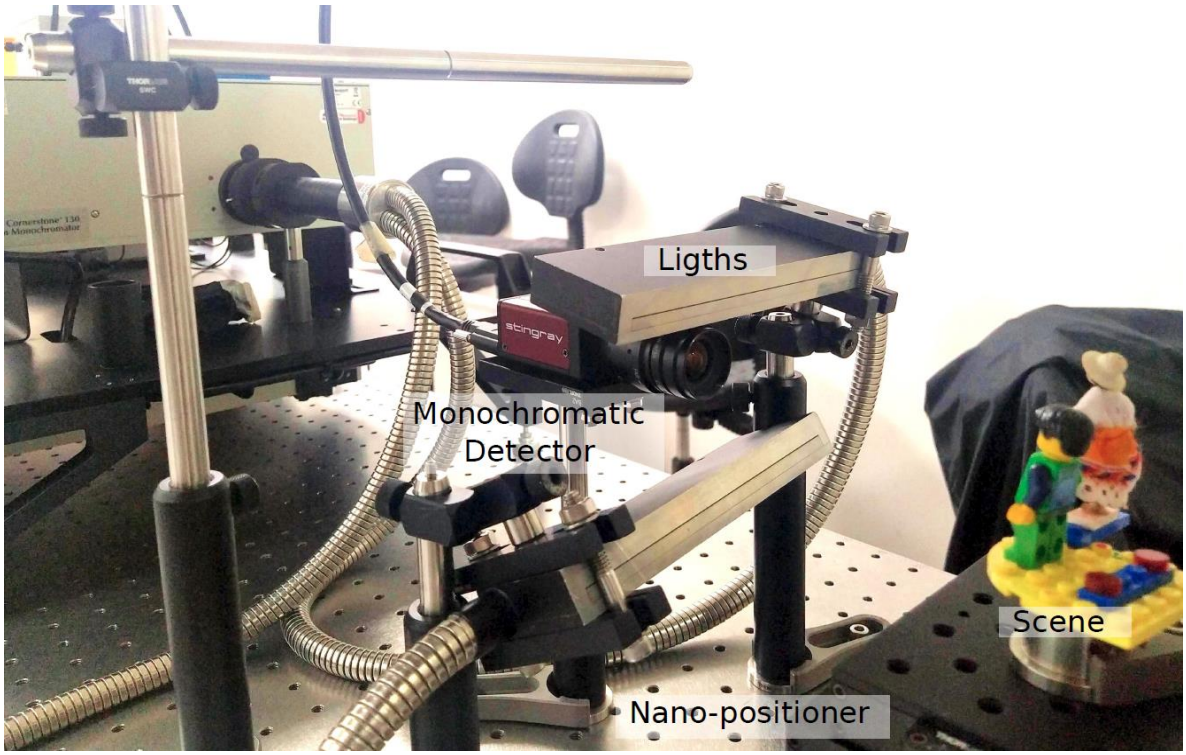


Figure 39. Spectral video camera laboratory prototype

[79] Hardware provided by the HDSP lab.

[80] Hoover Rueda, Henry Arguello, and Gonzalo Arce. "DMD-based implementation of patterned optical filter arrays for compressive spectral imaging". In: *JOSA A* 32.1 (2015), pp. 80–89.

The different databases acquired in the laboratory are shown in Figures 40, 41, 42 and 43, where some spectral bands of the video are pictured. Table 8 shows the resolution of each database.

Table 8. Size of the acquired spectral videos

Size	**Spatial pixels**		**Spectral Bands**	**Number of frames**
Video (type of motion)	I_1	I_2	I_3	I_4
Vertical Movements (Fig. 40)	776	1032	31	46
Angular (Fig. 41)	776	1032	31	14
Vertical inclined (Fig. 42)	776	1032	31	41
Vertical (Lego) (Fig. 43)	776	1032	31	46

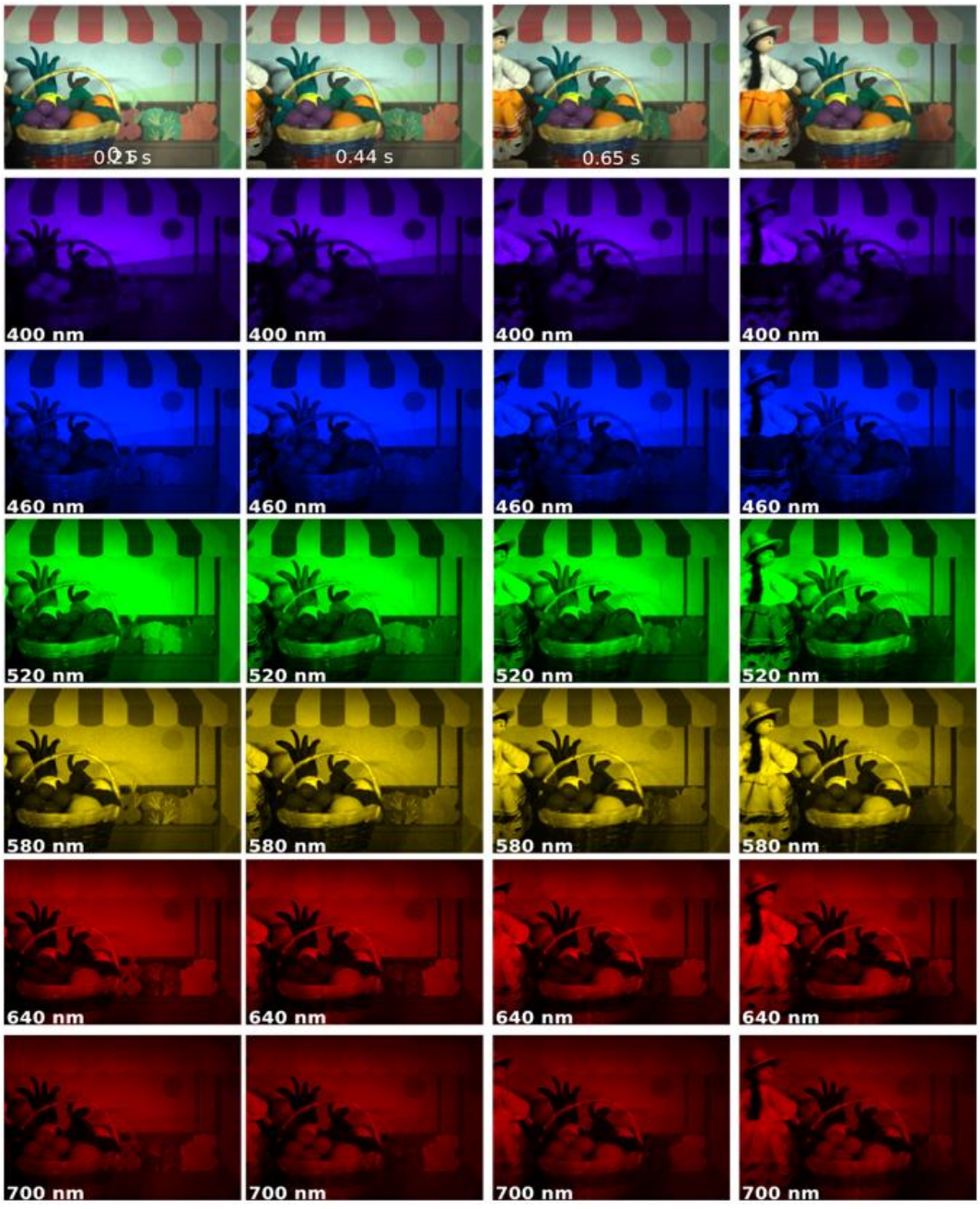


Figure 40. Spectral video scene: Vertical movements

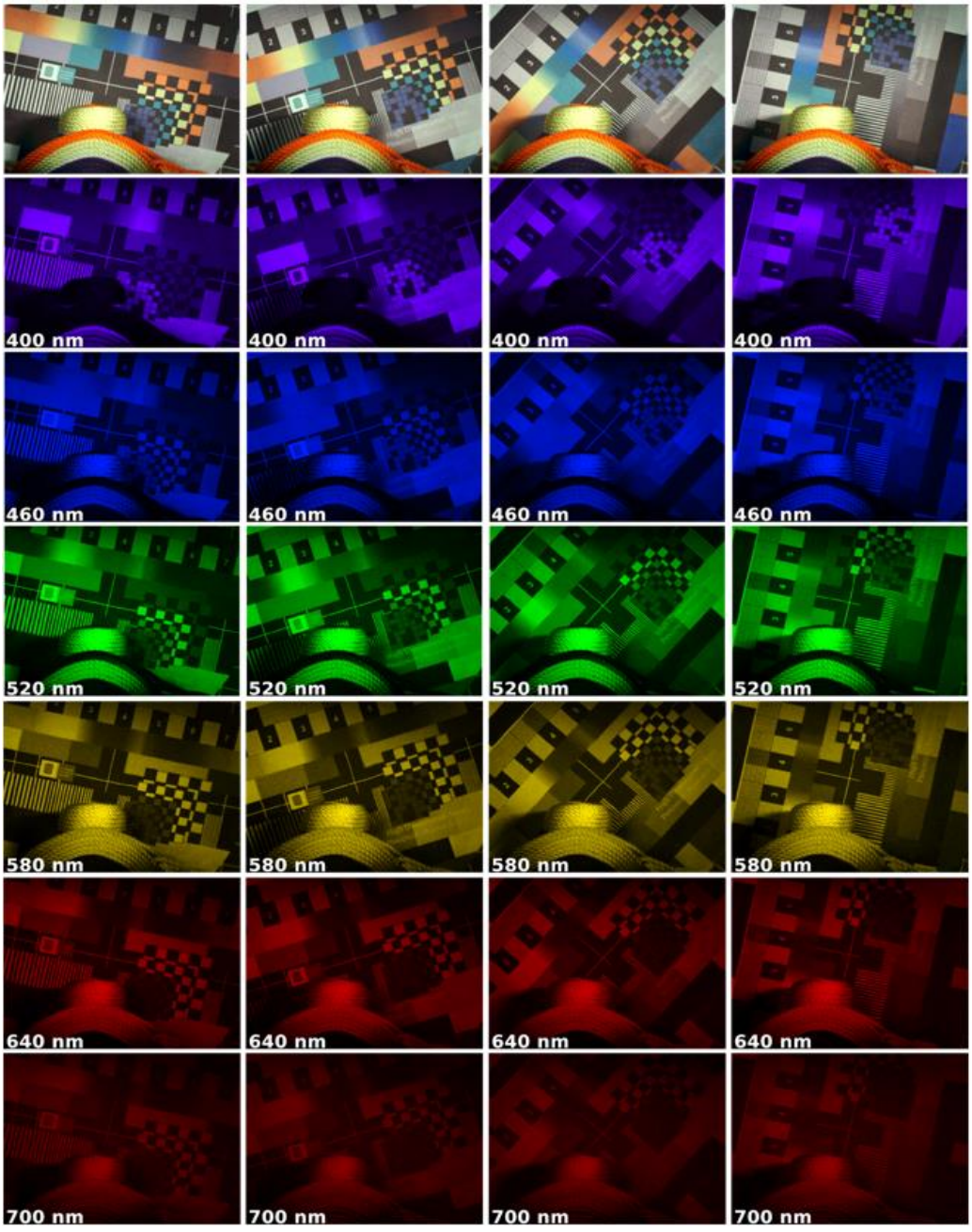


Figure 41. Spectral video scene: Circular movements

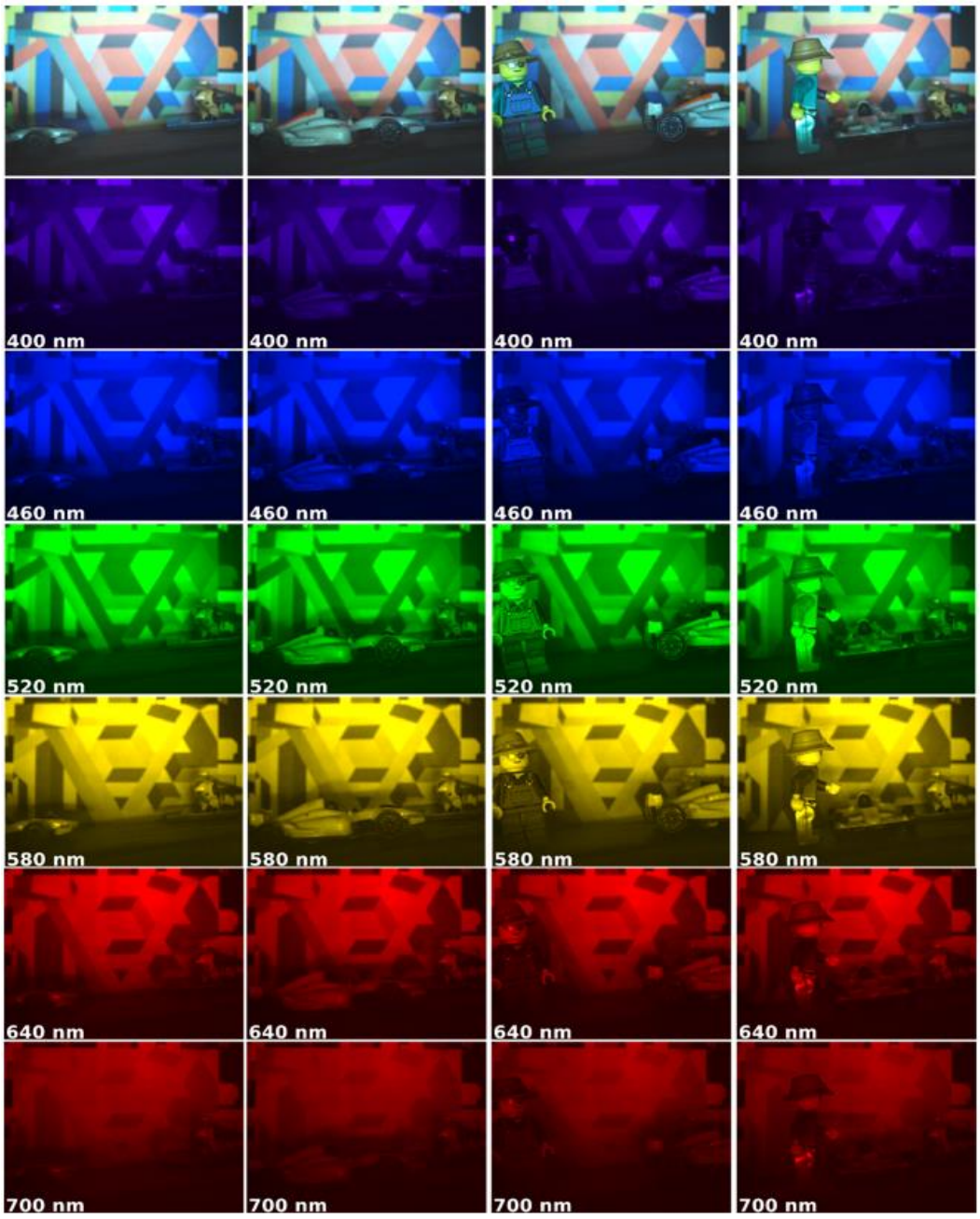


Figure 42. Spectral video scene: Vertical and horizontal movements

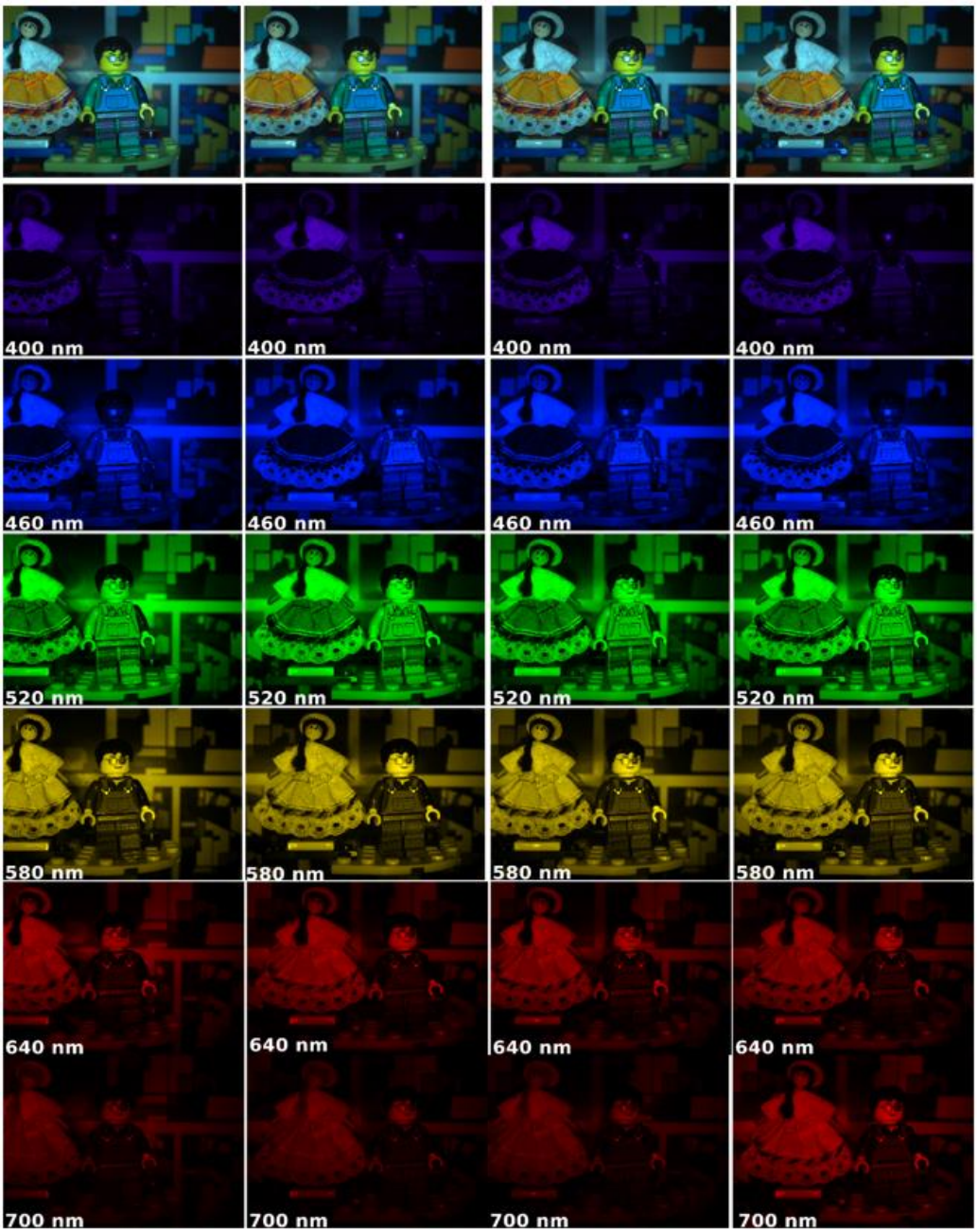


Figure 43. Spectral video scene: Vertical movements

Annex D. ANOMALY DETECTION AND CLASSIFICATION IN MULTISPECTRAL TIME SERIES BASED ON HIDDEN MARKOV MODELS

Introduction

Multispectral images have been widely used in many studies to explore the vegetation properties of plants through the extraction of vegetation indices [81][82][83][84]. In the last decade, researchers have proposed to use multi-temporal images for several applications including change detection [85] and landcover classification [86][87], where the challenge is mainly to exploit the redundancy and correlation across the spatial, spectral, and temporal dimensions of the images.

Hidden Markov models (HMM) are classical tools to analyze time series, allowing temporal correlations to be extracted with the introduction of latent variables inter-

---

[81] Fred Baret and Gerard Guyot. "Potentials and limits of vegetation indices for LAI and APAR assessment". In: *Remote Sensing of Environment* 35.2 (1991), pp. 161–173.

[82] Amanda Veloso et al. "Understanding the temporal behavior of crops using Sentinel-1 and Sentinel-2-like data for agricultural applications". In: *Remote Sensing of Environment* 199 (2017), pp. 415–426.

[83] Albert J Peters et al. "Drought monitoring with NDVI-based standardized vegetation index". In: *Photogrammetric Engineering and Remote sensing* 68.1 (2002), pp. 71–75.

[84] Linglin Zeng et al. "A review of vegetation phenological metrics extraction using time-series, multispectral satellite data". In: *Remote Sensing of Environment* 237 (2020), p. 111511.

[85] Jorge Prendes et al. "A Bayesian nonparametric model coupled with a Markov random field for change detection in heterogeneous remote sensing images". In: *SIAM Journal on Imaging Sciences* 9.4 (2016), pp. 1889–1921.

[86] Cristina Gómez, Joanne C White, and Michael A Wulder. "Optical remotely sensed time series data for land cover classification:A review". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 116 (2016), pp. 55–72.

[87] Miguel A García et al. "Using Hidden Markov Models for Land Surface Phenology: An Evaluation Across a Range of Land Cover Types in Southeast Spain". In: *Remote Sensing* 11.5 (2019), p. 507.

acting with the data [88][89][90]. Different works have shown that HMM are valuable tools for modeling the dynamic behavior of crops across time, where the dynamics of vegetation is related to the phenology, chemical nutrients, climatic conditions, or water stress of crops [91][92]. Some specific tasks for crop analysis based on HMM include crop recognition [93], crop classification [88][94], and time evolution featuring [92]. In addition, an analysis of the normalized difference vegetation index (NDVI) using the HMM framework is proposed in [91], where the NDVI changes are used to characterize the dynamics of the vegetation during a temporal window.

An important task in crop monitoring is the detection of anomalies that can represent risks for the harvest [83][95]. Detecting nutrient stresses or drought helps to better under-

---

[88] Sofia Siachalou, Giorgos Mallinis, and Maria Tsakiri-Strati. "A Hidden Markov Models Approach for Crop Classification: Linking Crop Phenology to Time Series of Multi-Sensor Remote Sensing Data". In: *Remote Sensing Letters* 7.4 (2015), pp. 3633–3650.

[89] Lawrence R Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286.

[90] Sofia Siachalou, Giorgos Mallinis, and Maria Tsakiri-Strati. "Analysis of Time-Series spectral index data to enhance crop identification over a Mediterranean rural landscape". In: *IEEE Geoscience and Remote Sensing Letters* 14.9 (2017), pp. 1508–1512. DOI: 10.1109/LGRS.2017.2719124.

[91] Nicolas Viovy and Gilbert Saint. "Hidden Markov models applied to vegetation dynamics analysis using satellite remote sensing". In: *IEEE Transactions on Geoscience and Remote Sensing* 32.4 (1994), pp. 906–917.

[92] Yonglin Shen et al. "Hidden Markov models for real-time estimation of corn progress stages using MODIS and meteorological data". In: *Remote Sensing* 5.4 (2013), pp. 1734–1753.

[93] Paula Beatriz Leite Cerqueira et al. "Hidden Markov Models for crop recognition in remote sensing image sequences". In: *Pattern Recognition Letters* 32.1 (2011), pp. 19–26.

[94] Yuan Yuan et al. "Continuous change detection and classification using hidden Markov model: A case study for monitoring urban encroachment onto farmland in Beijing". In: *Remote Sensing* 7.11 (2015), pp. 15318–15339.

[95] Michele Meroni et al. "Near real-time vegetation anomaly detection with MODIS NDVI: Timeliness vs. accuracy and effect of anomaly computation options". In: *Remote Sensing of Environment*

stand the management of nutrients and, in turn, leads to reduce cultivation costs and increases crop efficiency [968297]. Thus, depending on the kind of detected anomalies, the farmers can take action to reduce the adverse effects of the phenomenon that produces the anomaly response.

Anomaly detection (AD) (which includes outlier and novelty detection) is a widely studied problem that relies on the identification of patterns or events that differ from the expected normal behavior of the majority of the data [98]. The existing AD methods can be grouped into several categories based on classification/learning, nearest neighbours, clustering, statistical, and deep learning techniques [9899] (see [1009899101] for comprehensive reviews). In particular, AD as a learning task can be supervised, semi-supervised, or unsupervised [98102101], where the major challenge is the lack of labeled training instances [101].

In practical applications, it is generally easier to get access to labeled instances for

---

221 (2019), pp. 508–521.

[96] Clement Atzberger. "Advances in remote sensing of agriculture: Context description, existing operational monitoring systems and major information needs". In: *Remote sensing* 5.2 (2013), pp. 949–981.

[97] Héctor Vargas, Ariolfo Camacho, and Henry Arguello. "Spectral unmixing approach in hyperspectral remote sensing: a tool for oil palm mapping". In: *TecnoLógicas* 22.45 (2019), pp. 131–145.

[98] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM Computing Surveys (CSUR)* 41.3 (2009), pp. 1–58.

[99] Raghavendra Chalapathy and Sanjay Chawla. "Deep learning for anomaly detection: A survey". In: *arXiv preprint arXiv:1901.03407* (2019).

[100] Markos Markou and Sameer Singh. "Novelty detection: a review—part 1: statistical approaches". In: *Signal Processing* 83.12 (2003), pp. 2481–2497.

[101] Miryam Elizabeth Villa-Pérez et al. "Semi-supervised anomaly detection algorithms: A comparative summary and future research directions". In: *Knowledge-Based Systems* (2021), p. 106878.

[102] Marco AF Pimentel et al. "A review of novelty detection". In: *Signal Processing* 99 (2014), pp. 215–249.

normal data than getting access to anomalies, which leads to novelty detection [101]. Novelty detection aims at learning a model of normality from a set of data considered as normal to detect unobserved events, or *novelties*, in a semi-supervised mode [100][102]. On the other hand, outlier detection is generally defined as the detection of some data points, referred to as *outliers*, that seem to be inconsistent with the rest of the training set [102]. The main difference between *novelties* and *outliers* is that the detected novelties do not always correspond to anomalies [99][100]. This distinction is interesting in the context of crop monitoring at a parcel level since subtle deviations in data can be the result of external factors such as cloud or forest shadows and not due to anomalies damaging the crops [103][104]. Several AD algorithms have been investigated in the literature. Some of the most relevant and well-established techniques include the autoencoders (AE) [105], the local outlier factor [106] or its probabilistic version the local outlier probability [107], the one-class support vector machine

---

[103] Pierre Defourny et al. "Near real-time agriculture monitoring at national scale at parcel resolution: Performance assessment of the Sen2-Agri automated system in various cropping systems around the world". In: *Remote sensing of environment* 221 (2019), pp. 551–568.

[104] Florian Mouret et al. "Outlier Detection at the Parcel-Level in Wheat and Rapeseed Crops Using Multispectral and SAR Time Series". In: *Remote Sensing* 13.5 (2021). DOI: 10.3390/rs13050956.

[105] Mark A Kramer. "Nonlinear principal component analysis using autoassociative neural networks". In: *AIChE journal* 37.2 (1991), pp. 233–243.

[106] Markus M Breunig et al. "LOF: identifying density-based local outliers". In: *Proc. ACM SIGMOD Int. Conf. on Management of Data*. Dallas, TX, USA, 2000, pp. 93–104.

[107] Hans-Peter Kriegel et al. "LoOP: local outlier probabilities". In: *Proc 18th ACM Conf. Inform. Knowl. Manage. (CIKM '09)*. Hong Kong, China, 2009, 1649–1652.

(OC-SVM) [108], the isolation forest [109], and the k-nearest neighbour (kNN) [110].

Some recent works have proposed to solve the AD problem using HMM [111][112]. In particular, an HMM-based kernel was included in the traditional OC-SVM method in [111]. However, this kernel was defined assuming some specific kind of anomaly, e.g., resulting from mean-value jumps, which is a too strong assumption for crop monitoring. An interesting framework for AD in multivariate time series was proposed in [112], where a set of transformations was used to unify the time series and estimate appropriate features. However, the resulting AD algorithm was trained in a supervised mode, using normal and abnormal labels for the training samples, which are difficult to obtain in most crop monitoring applications.

In satellite remote sensing images, AD has been conducted either directly on the spectral image pixels [113] or on vegetation indices [114][95][115] computed from the com-

---

[108] Bernhard Schölkopf et al. "Estimating the support of a high-dimensional distribution". In: *Neural Computation* 13.7 (2001), pp. 1443–1471.

[109] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest". In: *Proc. Int. Conf. on Data Mining*. Pisa, Italy, Dec. 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17.

[110] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. "Efficient algorithms for mining outliers from large data sets". In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 427–438.

[111] Nico Görnitz, Mikio Braun, and Marius Kloft. "Hidden Markov anomaly detection". In: *International Conference on Machine Learning*. Lille, France, Jan. 2015, pp. 1833–1842.

[112] Jinbo Li, Witold Pedrycz, and Iqbal Jamal. "Multivariate time series anomaly detection: A framework of Hidden Markov Models". In: *Applied Soft Computing* 60 (2017), pp. 229–240.

[113] Chein-I Chang and Shao-Shan Chiang. "Anomaly detection and classification for hyperspectral imagery". In: *IEEE transactions on geoscience and remote sensing* 40.6 (2002), pp. 1314–1325.

[114] Erik R Venteris et al. "Detection of anomalous crop condition and soil variability mapping using a 26 year Landsat record and the Palmer crop moisture index". In: *International journal of applied earth observation and geoinformation* 39 (2015), pp. 160–170.

[115] Felix Rembold et al. "ASAP: A new global early warning system to detect anomaly hot spots of agricultural production for food security analysis". In: *Agricultural systems* 168 (2019), pp. 247–

bination of several spectral bands. Existing studies such as [95][114] exploit historical NDVI data to detect anomalies by comparing new observations against a full set of past observations at the global level. However, analyzing historical data at a parcel level is difficult in the proposed framework because of crop rotation and of missing data (due to the presence of clouds that cover some parcels).

Finally, it is interesting to mention some other works such as [90][104], which have demonstrated that crops analyzed at a parcel level from multi-temporal vegetation indices and vegetation phenology provide suitable knowledge of crops across time. Nevertheless, these studies have not addressed the problem of classifying and identifying possible factors that are damaging the harvest.

This Chapter introduces a framework for AD and classification of remote sensing time series at a parcel level based on HMM, allowing the detection, temporal localization and classification of anomalies. The proposed method referred to as AD-HMM learns the normal dynamic behavior of crops in a given season using several HMM whose parameters are estimated from normal data (i.e., data without any anomaly). Abnormal time series are then detected as those being unlikely to have been generated by these HMM. One advantage of AD-HMM is that the learned HMM can be used for specific time segments of the tested time series, allowing anomalies to be localized during specific time intervals. In a second step, the proposed AD algorithm is complemented by standard classifiers such as SVMs in order to determine the type of detected anomalies.

Up to our knowledge, this is the first approach providing an AD and classification framework to analyze remote sensing time series at the parcel level for detection, localization, and classification of anomalies, facing limitations such as low-temporal resolution. Consequently, the proposed method cannot be globally compared to a

---

257.

state-of-the-art reference. In order to evaluate the performance of the AD step, we considered some very popular methods such as OC-SVM, IF and HMAD in order to appreciate the interest of using HMMs. Numerical experiments conducted using synthetic and real data show the interest of the proposed strategy, allowing abnormal parcels to be detected, localized, and classified.

Proposed Method

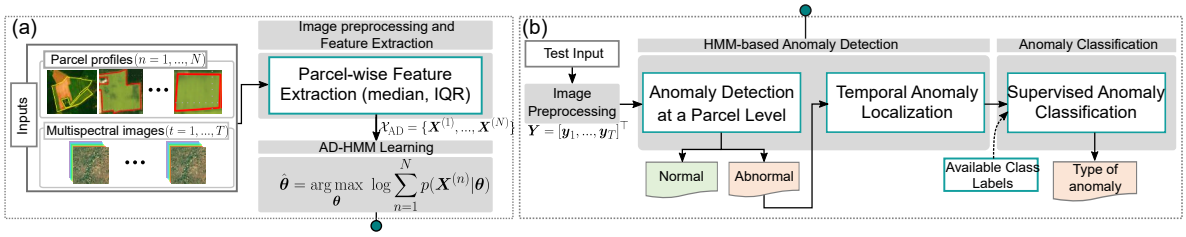


Figure 44. Flowchart illustrating the main steps and outputs of the proposed approach: (a) Learning step, where multi-temporal/multispectral images and parcel profiles are used to extract features of time series for a given parcel, and (b) Test step. Gray shaded squares indicate the different tasks, namely image preprocessing, AD-HMM learning, AD, anomaly localization, and anomaly classification.

This section presents the proposed AD and classification approach, which is summarized in the detailed flowchart depicted in Figure 44. Note that the gray shaded squares highlight the main steps of the method: (1) image preprocessing yielding features at the parcel level, (2) learning HMM associated with normal parcels referred to as AD-HMM learning, (3) AD at the parcel level (point AD), which includes the localization of anomalies, and (4) anomaly classification. Next subsections describe each procedure following the flowchart of Figure 44.

Image Preprocessing and Feature Extraction The image preprocessing step requires multi-temporal and multispectral images, and the corresponding parcel boundaries (e.g., resulting from a parcellation database such as the land parcel identifica-

tion system (LPIS[116])[117]). A set of temporal vegetation indices (VIs) is extracted from these images. For this study, five vegetation indices derived from the visible, near-infrared (NIR), and short-wave infrared (SWIR) were estimated based on images acquired by the Sentinel-2 sensor[118]. These VIs are summarized in Table 9 and the corresponding spectral bands are provided in Table 10.

Table 9. Vegetation indices estimated from multispectral images, where NIR, R, G, SWIR, and Re denote the near-infrared, red, green, short-wave infrared, and red-edge bands.

Vegetation Index (VI)	Formula
Normalized difference VI [119][120]	NDVI $= \dfrac{\mathrm{NIR}-\mathrm{R}}{\mathrm{NIR}+\mathrm{R}}$
Green-Red VI [121]	GRVI $= \dfrac{\mathrm{G}-\mathrm{R}}{\mathrm{G}+\mathrm{R}}$
Normalized difference water Index (SWIR) [122]	NDWI$_{SWIR} = \dfrac{\mathrm{NIR}-\mathrm{SWIR}}{\mathrm{NIR}+\mathrm{SWIR}}$
Normalized difference water Index (Green) [123]	NDWI$_G$ = $\dfrac{\mathrm{G}-\mathrm{NIR}}{\mathrm{G}+\mathrm{NIR}}$
Modified Chlorophyll Absorption Ratio Index using the Optimized Soil Adjusted VI [124]	$\mathrm{MCARI/OSAVI} = \dfrac{(\mathrm{Re}-\mathrm{IR})-0.2(\mathrm{Re}-\mathrm{R})}{(1+0.16)\dfrac{\mathrm{NIR}-\mathrm{R}}{\mathrm{NIR}+\mathrm{R}+0.16}}$

Two statistical indicators, namely the median and interquartile range (IQR), were computed for each temporal VI, where the IQR is defined by the difference between the 75th and 25th percentiles of the indicator. The motivation for employing statistical indicators for the temporal VIs is that they encompass the mean and dispersion of the VIs with a reduced computational load in the data processing. The preprocessing

[116] LPIS is a system based on images of agricultural parcels used in European countries as a tool to check the eligibility of agricultural land for subsidiary payments.

[117] Katarzyna Kocur-Bera. “Understanding information about agricultural land. An evaluation of the extent of data modification in the Land Parcel Identification System for the needs of area-based payments–a case study”. In: *Land Use Policy* 94 (2020), p. 104527.

[118] Sentinel-2 (S2A & S2B) level 2A images were downloaded with a spatial resolution of 10-60 m and a spectral resolution of 13 bands. The theoretical revisit time is 5 days. Bands with a lower spatial resolution were resampled to obtain a spatial resolution of 10×10 meters, and images with a cloud coverage greater than 20% were removed from the database.

Table 10. Spectral bands of the Sentinel-2A multispectral images employed in the VIs Estimation

Spectral band	Band 3 Green	Band 4 Red	Band 5 Red-Edge	Band 8 NIR	Band 11 SWIR
Wavelength (μm)	0,560	0,665	0,705	0,842	1,610
Resolution (m)	10	10	20	10	20

step provides K features for each parcel of the multi-temporal image and for each time instant, i.e., $N \times K \times T$ features, where N is the total number of available parcels and T denotes the number of time instants.These features are the median and the interquartile range of 5 vegetation indices reported in Table 9, leading to $K = 2 \times 5 = 10$ features. To define the extracted time series, let $\boldsymbol{X}^{(n)} = [\boldsymbol{x}_1^{(n)}, ..., \boldsymbol{x}_T^{(n)}]^\top$ in $\mathbb{R}^{T \times K}$ denotes all the time series computed for the n-th parcel (with $n = 1, ..., N$), where $\boldsymbol{x}_t^{(n)} \in \mathbb{R}^K$ is the feature vector at time $t \in \{1, ..., T\}$, and K is the number of features. Finally, $\mathcal{X}_{\text{AD}} = \{\boldsymbol{X}^{(1)}, ..., \boldsymbol{X}^{(N)}\}$ is the set of N time series extracted from the normal multispectral images and included in the learning set for AD.

The construction of the set $\mathcal{X}_{\text{AD}}$ is referred to as *parcel-wise feature extraction*, since it extracts statistics from the temporal VIs to build a set of features from the multi-temporal images and their parcel boundaries. The obtained time series are then validated by experts to make sure that they correspond to a normal behavior for AD-HMM learning. Note that the anomalies identified by the expert are excluded from the database and saved (with an anomaly label) in another set denoted as $\mathcal{X}_{\text{AC}}$, which will be used in the anomaly classification procedure (see [104] for further details about the database construction).

AD based on HMM Learning

HMM for Temporal Vegetation Indices Hidden Markov models (HMM) are doubly stochastic processes defined using an unobservable (hidden) state process,

which can be observed via another set of stochastic processes produced by a sequence of observations [89]. HMM allow the characterization of dynamic systems via a set of hidden states $\boldsymbol{s} = \{s_1, ..., s_D\}$ which are inferred from the observations of the system, where D is the number of states in the model. Concisely, an HMM can be formally described by the unknown parameters $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{B}\}$, where $\boldsymbol{\pi} \in \mathbb{R}^D$ is the initial probability vector, which defines the initial probabilities of the system to be in the different states, $\boldsymbol{A} \in \mathbb{R}^{D\times D}$ is the transition probability matrix, which relates the state changes of the hidden latent variable, and $\boldsymbol{B}$ is the emission probability matrix, which is the probability of observing a given value in state s.

In particular, given the n-th time series of temporal VIs $\boldsymbol{X}^{(n)}$, the hidden state sequence that reveals a possible state $z_t^{(n)} \in \{s_1, ..., s_D\}$ of $\boldsymbol{x}_t^{(n)}$ across time is denoted as $\boldsymbol{Z}^{(n)} = [\boldsymbol{z}_1^{(n)}, ..., \boldsymbol{z}_T^{(n)}]^\top$. On the other hand, the entries of the transition probability matrix $\boldsymbol{A}$ are given by $a_{i,j} = P(\boldsymbol{z}_t^{(n)} = s_i | \boldsymbol{z}_{t-1}^{(n)} = s_j)$, which is the probability of transition from a state s_j to the state s_i, for $i, j \in \{1, ..., D\}$. Finally, the entries of the emission probability matrix $\boldsymbol{B}$ are given by $b_{i,t} = P(\boldsymbol{x}_t^{(n)} | \boldsymbol{z}_t^{(n)} = s_i)$, which defines the probability density function of the time-sample $\boldsymbol{x}_t^{(n)}$ at time t given that $\boldsymbol{x}_t^{(n)}$ is in the state s_i. More precisely, in the proposed analysis, the emission probability distribution $\boldsymbol{B}$ is assumed to be a mixture of Gaussian distributions, with M multivariate normal densities. Note that the set of states $\boldsymbol{s}$ is typically related to phenological stages that describe the life cycle of vegetation [88,90,91].

The detection of anomalies in the AD-based HMM is made using two hypotheses defined as follows

$$H_0 : \text{Absence of anomaly}$$
$$H_1 : \text{Presence of anomaly},$$

where under hypothesis H_1 a given parcel $\boldsymbol{X}^{(n)}$ is supposed to be abnormal whereas it corresponds to a normal behavior under hypothesis H_0. The likelihood of a given

parcel is defined as

$$
\begin{aligned}
P(\boldsymbol{X}^{(n)}|\boldsymbol{\theta}) &= \sum_{\text{all } \boldsymbol{Z}^{(n)}} P(\boldsymbol{X}^{(n)}|\boldsymbol{Z}^{(n)},\boldsymbol{\theta})P(\boldsymbol{Z}^{(n)},\boldsymbol{\theta}) \\
&= \sum_{\boldsymbol{z}_1^{(n)},\ldots,\boldsymbol{z}_T^{(n)}} \pi_{\boldsymbol{z}_1^{(n)}} b_{\boldsymbol{z}_1^{(n)},1} a_{\boldsymbol{z}_1^{(n)},\boldsymbol{z}_2^{(n)}} \ldots a_{\boldsymbol{z}_{T-1}^{(n)},\boldsymbol{z}_T^{(n)}} b_{\boldsymbol{z}_T^{(n)},T}.
\end{aligned}
\tag{50}
$$

To correctly model and learn the temporal structure of the underlying data for AD, the HMM model parameter vector $\boldsymbol{\theta}$ is estimated by maximizing the log-likelihood, i.e.,

$$
\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \ \log \sum_{n=1}^{N} P(\boldsymbol{X}^{(n)}|\boldsymbol{\theta}),
\tag{51}
$$

where $\hat{\boldsymbol{\theta}}$ is the parameter vector that better explains $\mathcal{X}_{\text{AD}}$. A local optimal solution of Problem (51) can be found via the Baum-Welch algorithm [89].

Generating Different HMM-models The estimator $\hat{\boldsymbol{\theta}}$ defined in (51) is associated with all the parcels contained in the training set $\mathcal{X}_{\text{AD}}$. In order to account for different possible structures in the underlying data, it is proposed to build several HMM associated with subsets of N_s samples chosen in $\mathcal{X}_{\text{AD}}$, with $N_s \ll N$. These subsets are built using blocks of time series chosen randomly in $\mathcal{X}_{\text{AD}}$, which leads to L HMM models denoted as $\hat{\boldsymbol{\Theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_L\}$, with $\hat{\boldsymbol{\theta}}_\ell = \{\boldsymbol{\pi}^{(\ell)}, \boldsymbol{A}^{(\ell)}, \boldsymbol{B}^{(\ell)}\}$ for $\ell = 1, ..., L$. These subsets of time series will be denoted as $\{\mathcal{X}^\ell\}_{\ell=1}^{L}$ with $\mathcal{X}^\ell \in \mathbb{R}^{N_s \times K \times T}$ and $\mathcal{X}_{\text{AD}} = \bigcup_{\ell=1}^{L} \mathcal{X}^\ell$. The choice of the parameter L will be discussed in Section 5.

AD-HMM Learning Algorithm 4 summarizes the main steps of the proposed AD-HMM learning, corresponding to the second gray box in Figure 44(a). This algorithm receives the set of time series $\mathcal{X}_{\text{AD}}$, the number N_s of subsets used to learn a single model, the number of HMM states D, and the number of models L to be learned. Default values resulting from simulations conducted on real images are provided

for each parameter. In the initialization step (Line 1), the number of components M used in the Gaussian mixture model used for the emission probability distribution introduced in 3 was computed directly from the input data. A strategy to estimate this number of Gaussians is to use the number of local maxima in the data histogram and to reduce this number until the algorithm performance decreases significantly. In the next step, a random index selection of N_s time series stacked in $\mathcal{X}^\ell$ is performed, indicating the parcels to be selected from the set $\mathcal{X}_{\text{AD}}$, as detailed in the previous subsection. The HMM model parameters $\boldsymbol{\theta}$ are then randomly initialized [89]. Finally, the HMM model parameters are estimated using the Baum-Welch procedure and stacked into the set $\hat{\boldsymbol{\Theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_L\}$, where $\hat{\boldsymbol{\theta}}_\ell = \{\boldsymbol{\pi}^{(\ell)}, \mathbf{A}^{(\ell)}, \mathbf{B}^{(\ell)}\}$.

Algorithm 4: AD-HMM Learning Procedure

Input: $\mathcal{X}_{\text{AD}}$: Set of time series associated with parcels;
N_s: # of images per model (default $N_s = 100$);
L: # of models to be learned (default $L = 10$);
D : # of states (default $D = 18$);

1 **Initialize:** M: Estimate the number of Gaussian mixtures;
2 **for** $\ell = 1$ **to** L **do**
3 Built $\mathcal{X}^\ell$ by randomly selecting N_s parcels in $\mathcal{X}_{\text{AD}}$;
4 Initialize $\boldsymbol{\theta}_\ell = \{\boldsymbol{\pi}^{(\ell)}, \boldsymbol{A}^{(\ell)}, \boldsymbol{B}^{(\ell)}\}$;
5 $\hat{\boldsymbol{\theta}}_\ell \leftarrow$ BAUM-WELCH$(\mathcal{X}^\ell, \boldsymbol{\theta}_\ell, M, D)$;

Output: L HMM models $\hat{\boldsymbol{\Theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_L\}$ and their subsets of time series $\mathcal{X}^\ell$.

HMM-based AD For the testing part (see Figure 44(b)), AD is first performed at the parcel level to detect abnormal parcels. The detected anomalies are then localized in time as explained in Section 5. The proposed strategy is composed of 1) a *point AD* step detecting abnormal parcels, and 2) a *contextual AD* [98] step allowing the starting time of the anomaly to be estimated.

AD at the Parcel Level The probability that a time series $\boldsymbol{Y} = [\boldsymbol{y}_1, ..., \boldsymbol{y}_T]^\top$ has been generated by the ℓ-th learned model is written as

$$w_\ell = \log P(\boldsymbol{Y}|\hat{\boldsymbol{\theta}}_\ell), \tag{52}$$

for $\ell = 1, ..., L$, where $\boldsymbol{w} = [w_1, ..., w_L]^\top \in \mathbb{R}^L$ is a vector containing the log-probabilities of the test signal with respect to the L HMM models learned using Algorithm 1. Note that the log-probability in (52) is determined following the procedure in Table III, i.e., using the forward-algorithm [89], where $\alpha_{i,t}$ is the probability of partial observation of the time series at time t and state i (so-called forward variable), $a_{i,j}^{(\ell)}$ and $b_{j,t}^{(\ell)}$ are the elements of the matrices $\boldsymbol{A}^{(\ell)}$ and $\boldsymbol{B}^{(\ell)}$ for the ℓ-th model with parameter vector $\hat{\boldsymbol{\theta}}_\ell$. Consequently, (52) is the sum of the forward variables $\alpha_{i,t}$ across t providing a unique probability w_ℓ for a given parcel.

Table 11. Estimation of Log-Probabilities for a Test Signal.

Forward-procedure for the $\ell = 1, ..., L$ models	
1) $\alpha_{i,1} = \pi_i^{(\ell)} b_{i,t}^{(\ell)}$	(Initialization)
2) $\alpha_{i,t+1} = \left[\sum_{j=1}^{D} \alpha_t \; a_{i,j}^{(\ell)}\right] b_{j,t}^{(\ell)}$	(Induction)
3) $\log P(\boldsymbol{Y}|\hat{\boldsymbol{\theta}}_\ell) = \sum_{j=1}^{D} \alpha_{i,T}$	(Ending)

The final AD rule (at the parcel level) is defined as:

$$w^* = \max_{\ell=1,...,L} w_\ell \underset{H_0}{\overset{H_1}{\lessgtr}} \tau, \tag{53}$$

where w^* is the probability associated with the most likely model class (with the highest probability) and τ is a threshold related to the probability of false alarm (PFA) and the probability of detection (PD) of the test. More precisely, the value of τ was determined as the point of the ROC curve (expressing the PD as a function of the PFA)

located the closest to the ideal point $(\text{PFA}, \text{PD}) = (0, 1)$ [125]. Looking carefully at the proposed AD rule (53), a tested time series is declared as normal if the highest probability w^* exceeds the threshold τ, i.e., if a least one of the L models associated with the normal HMM models is likely to correspond to the observations. This detection rule is motivated by the fact that it is assumed that L different HMM models capture the possible temporal structures or signal dynamics of normal time series.

Anomaly Localization via Segmentation When a tested time series $\boldsymbol{Y}$ has been declared as abnormal in (53), it goes into the second step devoted to anomaly localization. In this step, the HMM models $\hat{\boldsymbol{\Theta}} = \{\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_L\}$ determined using Algorithm 1 are used on time segments $\left[t_{\rho_1}, t_{\rho_2}\right] = \{t \,|\, t_{\rho_1} \leq t \leq t_{\rho_2}\}$ (instead of analyzing the complete time series) to determine the starting point of the anomaly in the time series. Consider the forward variable $\alpha_{i,t-1}$ at time $t-1$ in its scaled version defined as $\tilde{\alpha}_{i,t-1} = \alpha_{i,t-1} / \sum_{i=1}^{D} \alpha_{i,t-1}$, where $i = 1, ..., D$, and D is the number of HMM states. The probability of having $\boldsymbol{Y}$ generated by the model $\hat{\boldsymbol{\theta}}$ at time t can be written in terms of $\alpha_{i,t-1}$ as follows

$$u_t = 1/\big(\sum_{i=1}^{D} \tilde{\alpha}_{i,t-1} a_{i,j} b_{i,t-1}\big), \tag{54}$$

where u_t depends on the scaled forward variable $\tilde{\alpha}_{i,t-1}$, the transition probability $a_{i,j}$, and the emission probability at time $t-1$. Note that this expression for u_t results from the first-order Markov chain rule, which assumes that the current state (at time t) depends only on its predecessor state (at time $t-1$) [89]. Note also that the forward variable $\alpha_{i,t-1}$ is used in its scaled version $\tilde{\alpha}_{i,t-1}$ to avoid overflow. Indeed, this variable relies on the sum of a large number of terms, as shown in the induction step

---

[125] Steven M Kay. *Fundamentals of statistical signal processing*. Prentice Hall signal processing series. Upper Saddle River, NJ: Prentice Hall PTR, 1993.

of the forward procedure. The log-likelihood of the time series in the time segment $[t_{\rho_1}, t_{\rho_2}]$ is defined as

$$\log P(\boldsymbol{y}_{t_{\rho_1}}, ..., \boldsymbol{y}_{t_{\rho_2}} | \boldsymbol{\pi}, \boldsymbol{A}, b_{i,[t_{\rho_1}-1, t_{\rho_2}-1]}) = -\sum_{t=t_{\rho_1}}^{t_{\rho_2}} u_t, \tag{55}$$

where u_t has been defined in (54), and $b_{i,[t_{\rho_1}-1,t_{\rho_2}-1]}$ is the emission probability for the time interval $\left[t_{\rho_1}, t_{\rho_2}\right]$, for $i = 1, ..., D$. As a result, the probability that a time series $[\boldsymbol{y}_{t_{\rho_1}}, ..., \boldsymbol{y}_{t_{\rho_2}}]$ has been generated by the ℓ-th learned model on the segment $[t_{\rho_1}, t_{\rho_2}]$ can be computed as

$$\boldsymbol{w}_{\ell,[t_{\rho_1}, t_{\rho_2}]} = \log P\Big(\boldsymbol{y}_{t_{\rho_1}}, ..., \boldsymbol{y}_{t_{\rho_2}} | \boldsymbol{\pi}^{(\ell)}, \boldsymbol{A}^{(\ell)}, b^{(\ell)}_{i,[t_{\rho_1}-1, t_{\rho_2}-1]}\Big), \tag{56}$$

where $\boldsymbol{w}_{\ell,[t_{\rho_1}, t_{\rho_2}]}$ is the vector containing the log-probabilities of the different models in the time interval $[t_{\rho_1}, t_{\rho_2}]$.

The anomaly localization studied in this Chapter considers four time intervals associated with different phenological stages of crop growth to identify potential anomalies. These intervals are referred to as *Growing*, *Flowering*, *Adult-phase*, and *Senescence* (displayed in Figure 8 (a)), as in [9193]. The hypothesis test to localize the anomaly in a given interval is defined as

$$w_S = \max_{\ell=1,...,L} w_{\ell,[t_{\rho_i}, t_{\rho_j}]} \underset{H_0}{\overset{H_1}{\lessgtr}} \tau_S, \tag{57}$$

where $S \in \{$*Growing, Flowering, Adult-phase, Senescence*$\}$ denotes the growth stage under analysis in the hypothesis test, τ_S is a threshold depending on the probability of false alarm and the probability of detection of the test sample for a given stage S, w_S contains the probabilities the phenological stage S given the tested time series, t_{ρ_i} denotes the beginning of the given stage, and t_{ρ_j} represents the end of the stage. Note that (53) defines the parcel detection rule whereas (57) localizes the

anomaly in one of the pre-defined phenological stages.

Anomaly Classification A final step can be included in the analysis to identify the anomaly that has affected the parcel $\boldsymbol{Y}$ using a supervised classifier. After an anomaly has been localized using the steps displayed in Figure 44(b), we propose to classify the detected anomaly into one of the C classes defined by the user and corresponding to the possible types of anomalies affecting the analyzed crop.
The set of features used for the classification is composed of the abnormal time series that has been detected in (53), whose different feature vectors are introduced as columns in the input matrix. More precisely, the feature matrix for training the classifier is of the form $\boldsymbol{X}_{\mathsf{AC}} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_R]$, where $\boldsymbol{x}_r = [x_{r,1}, ..., x_{r,KT}]^\top$ is a vector containing the KT features extracted from the r-th time series at all time instants. It is important to highlight that each time series $\boldsymbol{x}_r$ selected for training the classifier contains an abnormal time series from $\mathcal{X}_{\mathsf{AC}}$ with the corresponding label, denoted by $\boldsymbol{v}_r \in \{1, ..., C\}$, where C is the total number of classes. In the testing part, the classifier generates for each time series $\boldsymbol{y} = [y_1, ..., y_{KT}]^\top$ a label $v_{\boldsymbol{y}} \in \{1, ..., C\}$ indicating the class of the anomaly $\boldsymbol{y}$.

Simulation Results

The performance of the proposed methodology[126] is evaluated on real data[127]. AD is evaluated in terms of precision, recall, and area under the precision-vs-recall curve (AUC), whereas the overall accuracy (OA), the kappa coefficient, and the probabil-

---

[126] For the AD-HMM implementation, we used the HMM toolbox available online at `https://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html`.

[127] An complementary analysis conducted on synthetic data is provided in the journal paper (Kareth M. León-López et al. "Anomaly Detection and Classification in Multispectral Time Series based on Hidden Markov Models". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 [2022], pp. 1–11. DOI: 10.1109/TGRS.2021.3101127).

ities of correct classification are used for anomaly classification[128]. The precision, and recall are defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP + FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP + FN}},$$

where TP, TN, FP, and FN are the numbers of true positives, true negatives, false positives, and false negatives[129]. On the other hand, the probability of correct classification for class c (denoted as P_c) is defined as

$$P_c = \frac{1}{R_c} \sum_{r=1}^{R_c} \delta(v_r, \hat{v}_r), \tag{58}$$

where R_c is the total number of training vectors of the class $c \in \{1, ..., C\}$, v_r and $\hat{v}_r$ are the true and estimated labels of the r-th training vector of class c and $\delta(\cdot)$ is an indicator such that $\delta(v_r, \hat{v}_r) = 1$, if $v_r = \hat{v}_r$, and zero otherwise.

Study Area The research site displayed in Figure 45 is located in Beauce, North of France. This site contains a lot of crop fields such as rapeseed and wheat. A set of 13 multispectral Sentinel-2 images was selected between October 2017 and June 2018.

The dataset was processed to level 2A using the MAJA[130] processing chain [131].

---

[128] The higher the value of the metric, the better the detection or classification.

[129] The detection threshold was determined using the point of the AUC curve located *the closest to the ideal point* $(0, 1)$.

[130] MAJA is available on the PEPS (Plateforme d'Exploitation des Produits Sentinel) platform of the French National Center for Space Studies (Centre National d'Études Spatiales, CNES).

[131] Olivier Hagolle et al. "A multi-temporal and multi-spectral method to estimate aerosol optical thickness over land, for the atmospheric correction of FormoSat-2, LandSat, VENμS and Sentinel-2 images". In: *Remote Sensing* 7.3 (2015), pp. 2668–2691.

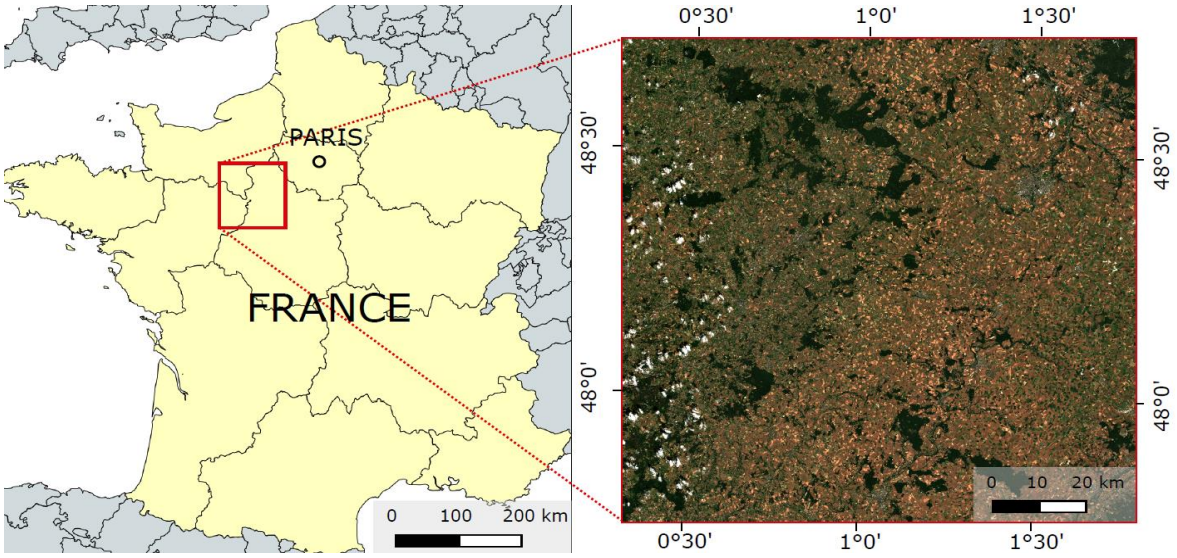


Figure 45. Study area located in Beauce, North of France.

A set of 1924 rapeseed parcels was extracted from the images with the support of agronomists, as illustrated in Figure 44(a). The resulting dataset was analyzed with the aim of labeling part of the data for evaluation purposes. The anomalies found in the data were related to vegetation phenology problems, crop heterogeneity, boundary errors, wrong crop type, and shadow perturbations. Vegetation phenology problems (illustrated in Figure 46(a)) are associated with problems during the plant development such as early/late growth, early/late senescence (plant degradation), and early flowering. On the other hand, heterogeneity corresponds to the spatial heterogeneous development of the crop, which can indicate the presence of crop or soil diseases and can affect the crop at any moment of growth. Boundary errors and wrong crop type are anomalies describing problems related to the database delineation and wrong target crop. Note that the rapeseed data and their corresponding labels are the same than those used in [104]. It is important to note that the labeling used in the real simulations was performed with the help of agronomist experts by visual-interpretation using all the available images and by using all the time series

of the different VI features to compare any analyzed parcel to the rest of the dataset [104].

The partitioning of the dataset for AD was randomly performed as follows: $N = 500$ normal parcels were selected for AD-HMM learning and the remaining 1424 parcels (697 normal parcels and 727 anomalies) were considered for the testing phase. The test parcels detected as anomalies were then considered as input data for the classifiers. Based on the anomalies found in the database, the anomaly classification step considers the following classes: late growth, early senescence, late senescence, and other, where "other" is a class containing all the anomalies that do not belong to the previous three classes. These anomalies are non-agronomic anomalies (such as errors in parcel boundaries, wrong crop type, or shadow perturbations) or agronomic anomalies affecting a very small number of parcels (such as early growth, early flowering, and crop heterogeneity).

For illustration purposes, Figure 46 displays (a) the expected temporal profile of the NDVI median with anomalies related to problems in the vegetation phenology such as early/late growth, early/late senescence, and early flowering, and (b) the distribution of a set of 500 normal and abnormal NDVI medians. As can be seen in the histograms and in the zoomed portion, the normal and abnormal data have different distributions, allowing anomalies to be classified. Note that the different vegetation phenological stages for the rapeseed crops indicated in the top of Figure 46(a) are located between the vertical gray dotted lines and were selected based on [82104].

Analysis of vegetation indices To analyze the impact of using different time series of VIs for AD, Figure 47 compares the performance of the proposed AD-HMM algorithm using the median and IQR of different combinations of VIs introduced in Table 9. More precisely, Figure 47 shows the AUC values obtained for different VI combinations after averaging the results of 10 Monte Carlo runs. Note that all the VIs were scaled such that each column of the feature matrix take its values in the interval

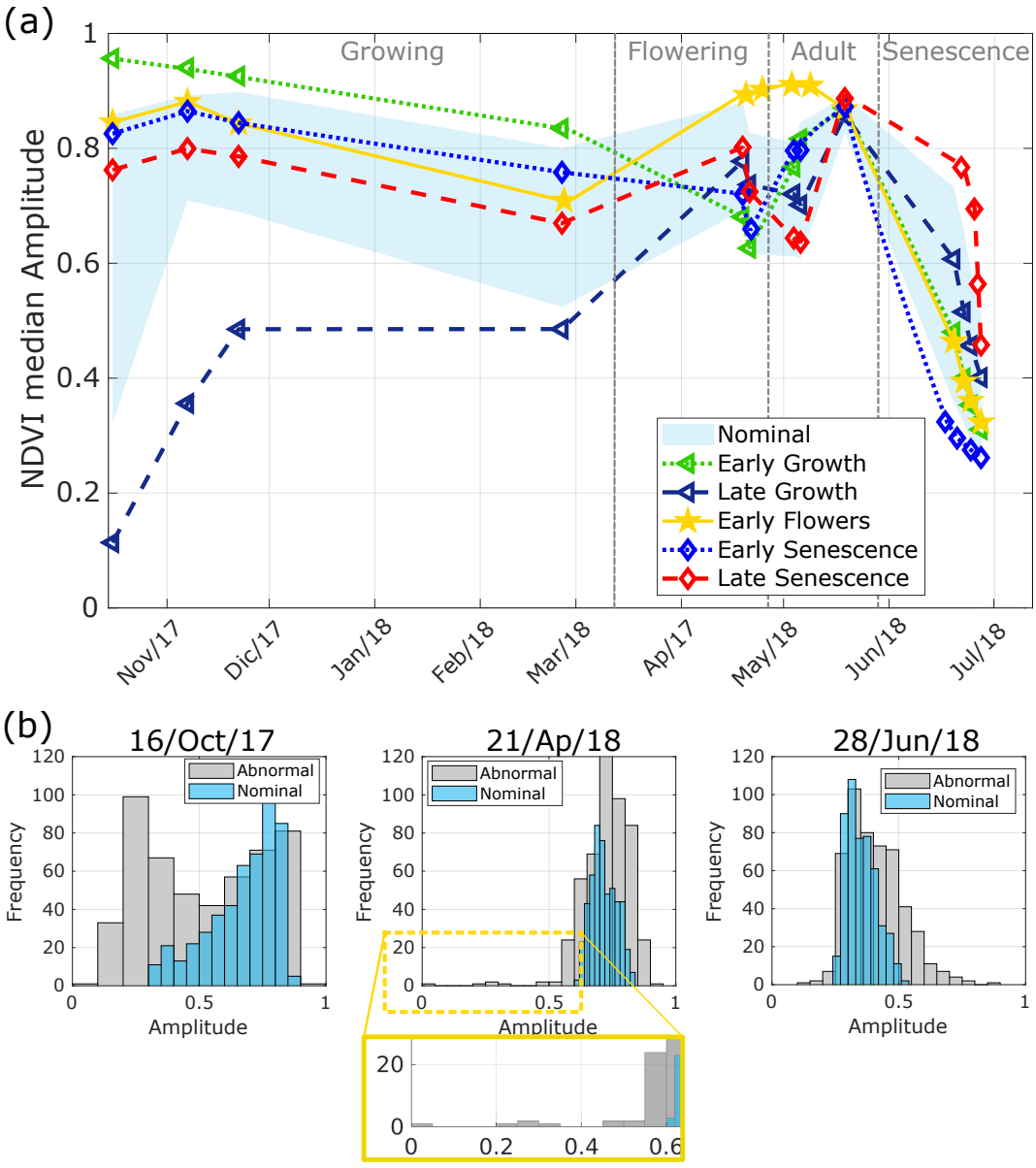


Figure 46. Temporal profiles and distribution of normal (blue) and abnormal data. a) Five typical time series profiles for agronomic anomalies are shown, where the shaded blue section corresponds to the normal time series. b) Histogram of 500 time series of normal (blue) and abnormal (gray) NDVI median for three dates, which illustrates how the distribution of abnormal data deviates respect to the normal data, leading potential anomalies to be detected by the proposed approach.

$(0, 1)$ (minimum-maximum scaling). One can observe that the performance of AD-HMM is very similar when using different VI combinations. Therefore, the remaining analyses will be performed with NDVI only. When using NDVI only, the estimated number of Gaussians in the mixture model was fixed to $M = 6$.

The overall performance of the proposed AD-HMM algorithm depends on the parameters N_s, L, and D, that need to be adjusted. The number of states was varied in the set $\{3, 9, 12, 15, 18\}$ whereas the values of N_s and L were chosen in the set $\{10, 25, 50, 100, 200\}$. For $D = 18$, the averaged AUC metrics vary in the interval $[0.80, 0.83]$, and the best performance for all the VIs was obtained when $N_s = 100$ and $L = 10$. These values were selected in the rest of the analysis, in particular to display Figure 47.

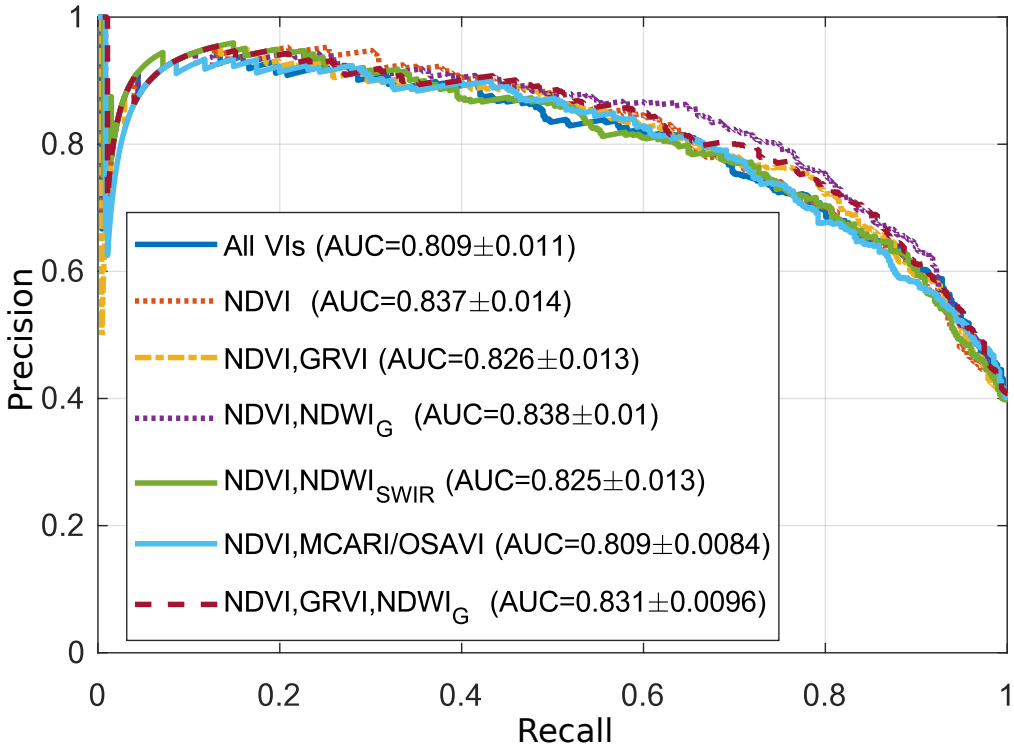


Figure 47. Performance of the AD-HMM detection using the median and IQR of different temporal vegetation indices, where the AUC value of each VI combination is shown in the legend.

AD Results The experiments conducted on real data using the proposed AD approach were compared to different algorithms including IF-N, OC-SVM-N, and HMAD. All the algorithms were run using the NDVI features. The parameters of the algorithms were set by cross validation. For OC-SVM-N, the outlier ratio was set to $\nu = 0.1$, and the kernel parameter for the RBF kernel was estimated using Jaakkola's heuristic [132]. The IF-N algorithm was run using 1000 isolation trees and a sub-sampling ratio of 256. The outlier fraction used in HMAD was set to $\nu = 0.3$. Figure 48 summarizes the performance of the different algorithms, where the proposed approach obtains slightly better results than OC-SVM-N and IF-N. The poor performance obtained with HMAD is probably due to the fact that the anomalies affecting crop parcels are not limited to mean changes but also lead to variance changes for which the HMAD algorithm is not adapted.

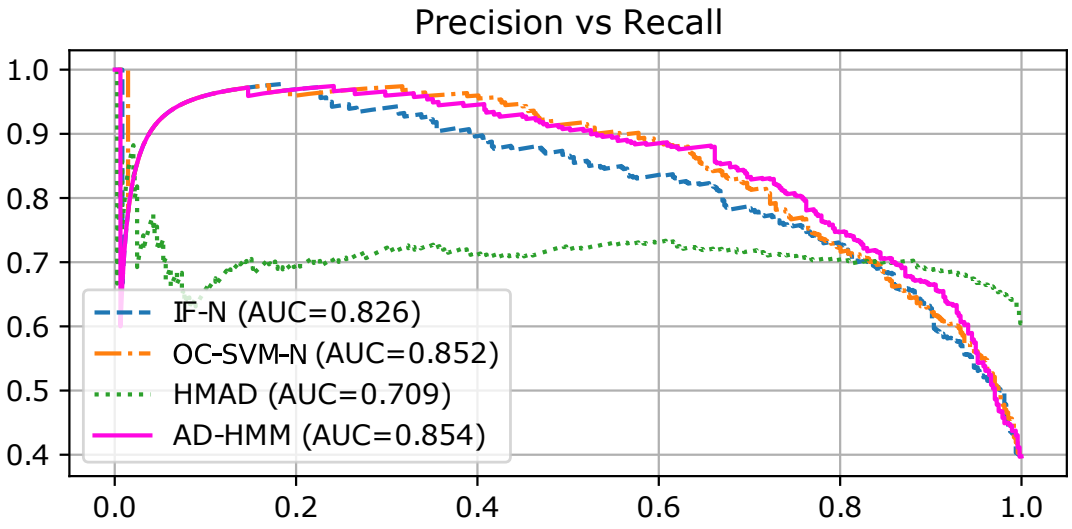


Figure 48. Performance of different AD methods to detect abnormal parcels in the real dataset.

[132] Tommi Jaakkola, Mark Diekhans, and David Haussler. "Using the Fisher kernel method to detect remote protein homologies." In: *Proc. Int. Conf. on Intelligent Systems for Molecular Biology*. Heidelberg, Germany, Aug. 1999, pp. 149–158.

Time Anomaly Localization The signals detected as abnormal in the previous step were then analyzed to localize the anomalies affecting the crops. For an easier interpretation, the acquisition dates presented in Figure 46 were transformed into integer values following the time intervals associated with the different phenological stages of rapeseed crops as follows: *Growing*$= \{t|1 \leq t \leq 4\}$, *Flowering*$= \{t|5 \leq t \leq 6\}$, *Adult*$= \{t|7 \leq t \leq 9\}$, and *Senescence*$= \{t|10 \leq t \leq 13\}$.

Figure 49 shows the results obtained for three time series with growth and wrong type problems, where the estimated value of $S \in \{$*Growing, Flowering, Adult-phase, Senescence*$\}$ by the proposed AD-HMM is indicated in the top of each figure as "Detected Stage" whereas the class of the anomaly is referred to as "True Class". The lattice box on the plots in Figure 49 highlights the detected stage. Note that based on the learned models, the proposed approach can estimate when the temporal structure deviates from the normal behavior, even, for subtle deviations as shown in the plot of the middle for *Late Senescence* problems.

Anomaly Classification After AD and Localization The last step of the proposed algorithm classifies some classes of anomalies detected in the rapeseed crops. To evaluate the classification performance on the available samples, the leave-one-out cross-validation (LOOC) strategy was considered. LOOC consists in leaving one vector out of the database, training the classifier with all the remaining samples, testing the classifier with the vector removed from the database and repeating these operations R times, where R is the size of the database. This strategy was selected given the few number of training samples available for anomaly classification. The classifiers considered in this section were the random forest (RF) algorithm with 100 trees and a maximum number of features set to the square root of the number of columns of the feature matrix, the k-nearest neighbor (k-NN) classifier with $k = 3$, and the support vector machine algorithm with linear (SVM-LN) and Gaussian (SVM-RBF) kernels, with a regularization parameter $C = 1$. The dif-

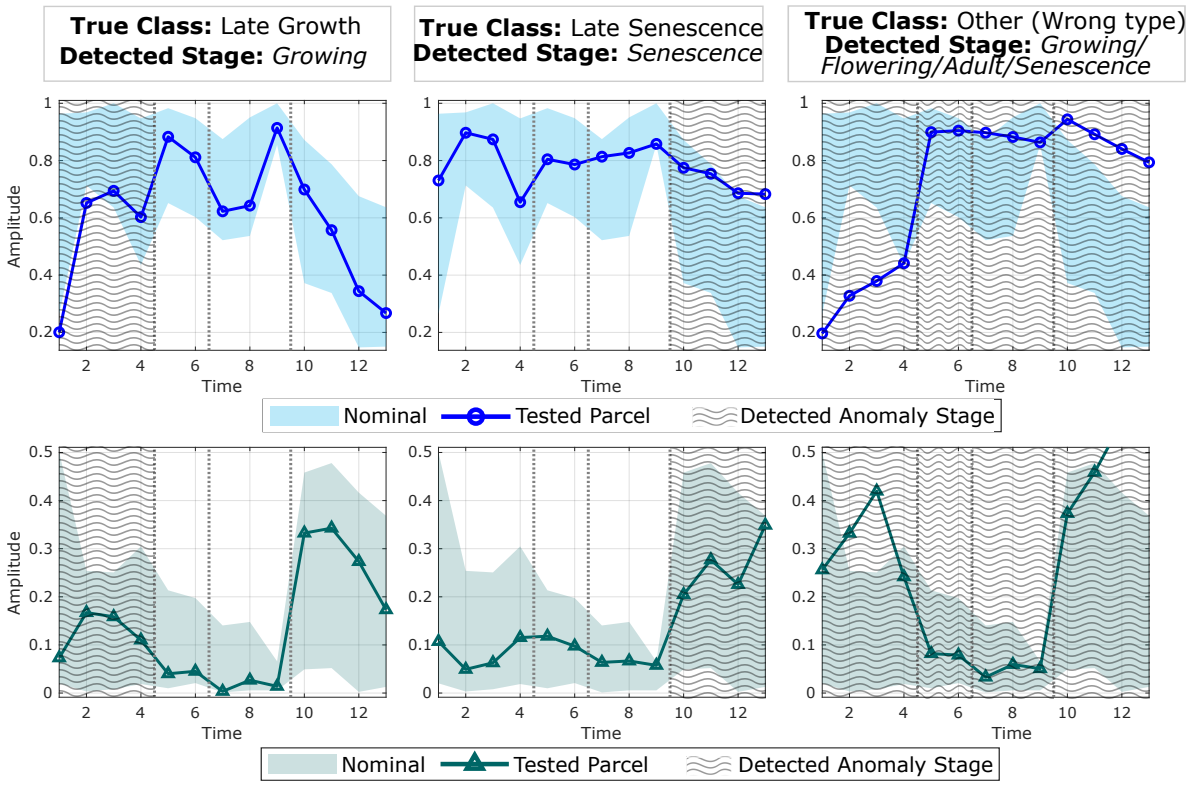


Figure 49. Time anomaly localization for three tested parcels of rapeseed crops affected by different anomalies. Each plot displays the median (top) and IQR (bottom) of the NDVI features. The box in the top indicates the class of the anomaly and the detected stage. The lattice box highlights the detected stage.

ferent parameters values of the classifiers were chosen in order to obtain the best performance. The multi-class strategy used in the SVM-based classifiers was based on the One-Against-One voting strategy [133]. In addition, the synthetic minority over-sampling technique (SMOTE) was used to oversample the training set to mitigate the unbalanced nature of the dataset [134].

---

[133] F. Melgani and L. Bruzzone. "Classification of hyperspectral remote sensing images with support vector machines". In: *IEEE Transactions on Geoscience and Remote Sensing* 42.8 (2004), pp. 1778–1790.

[134] Nitesh V. Chawla et al. "SMOTE: Synthetic Minority over-Sampling Technique". In: *J. Artif. Int. Res.* 16.1 (2002), 321–357.

Table 12 shows the estimated probability of correct classification, overall accuracy, and kappa coefficient for the different classifiers obtained using 10 Monte Carlo runs (where the highest values are highlighted in bold and the corresponding standard deviations are indicated into brackets). Note that the number of samples is presented in average. As can be observed, the highest value of P_c is obtained for the SVM-RBF classifier, whereas the better OA and value of kappa were obtained from the RF classifier. These two classifiers provide the overall best classification performance. The resulting confusion matrix of the SVM-RBF classifier for the 10th realization is shown in Table 13, where 618 samples were detected using AD-HMM. Note that the class *other*, which contains anomalies such as heterogeneity, wrong crop type, errors in parcel boundaries, and shadow perturbations, allows us to be close to a real scenario where anomalies that cannot be explained by abnormal plant growing are often present. It is important to mention here that in the rapeseed crops of this study, those classes (wrong crop type, heterogeneity, and shadow perturbations) affect either the whole time series or some time intervals in a random way, yielding anomalies located in any time interval. This lack of structured patterns increases the complexity of the classification, which explains the relatively poor classification performance obtained for this class. Additional information resulting from other data, e.g., from synthetic aperture radar images, might be considered to improve the classification performance. This work is currently under investigation.

Conclusions

A method for detecting, localizing, and classifying anomalies that affect agricultural crops based on hidden Markov models (HMM) and machine learning was presented. The proposed anomaly detection based on HMM (AD-HMM) exploited the temporal structure of time series of vegetation indices extracted from multispectral images to perform both point (parcel-wise) and contextual (temporal-wise) anomaly detec-

Table 12. Performance results for the different classifiers (Leave-One-Out Cross validation). Note that the number of samples is presented in average.

#	Classes	# Samples	SVM-LN	SVM-RBF	KNN	RF
1	Late Growth	179,1 (5,8)	0,79 (0,01)	**0,82** (0,00)	0,81 (0,01)	0,81 (0,01)
2	Early Senescence	51,3 (3,0)	0,91 (0,03)	**0,94** (0,02)	0,84 (0,04)	0,72 (0,02)
3	Late Senescence	28,8 (1,0)	0,97 (0,01)	**1,00** (0,01)	0,92 (0,01)	0,73 (0,03)
4	Other	352,3 (1,8)	0,59 (0,01)	0,60 (0,00)	0,61 (0,01)	**0,76** (0,01)
Average Pc			0,81 (0,01)	**0,84** (0,00)	0,79 (0,01)	0,75 (0,01)
OA (%)			69,16 (0,42)	71,32 (0,43)	70,26 (0,90)	**77,04** (0,49)
kappa			0,54 (0,01)	0,57 (0,01)	0,54 (0,01)	**0,62** (0,01)

Table 13. Confusion matrix for the SVM-RBF classifier for realization # 10 (Pc: 0.83, OA: 70.1%, kappa: 0.57)

Detected AD-HMM: **618**		**Predicted Class**			
Classes		**Late Growth**	**Early Senescence**	**Late Senescence**	**Other**
True Class	Late Growth	**151**	6	16	11
	Early Senescence	1	**49**	0	4
	Late Senescence	0	0	**28**	0
	Other	55	61	26	**210**

tion. The proposed method also allowed the detected anomalies to be temporally localized and classified into pre-defined classes, information that is valuable for crop monitoring. A comparison with classical anomaly detection algorithms, in terms of precision and recall, provided very promising results. An interesting property of the proposed anomaly detection algorithm is its capacity of localizing and classifying the anomalies located within each time series by exploiting the previously learned HMM models, where the outcomes of these steps provide a complementary knowledge to the farmers and producers for monitoring their crops.

Further investigation should be conducted to evaluate the interest of the proposed approach for detecting anomalies in other kinds of crops to characterize their dynamic behavior. Moreover, it would be interesting to generalize the proposed approach to non-homogeneous Markov chains in order to handle transitions between states defined by non-stationary time series. Another interesting further work is the

extension of the proposed AD-HMM to time series of vegetation indices estimated from multiple remote sensing sources, e.g., extracted from synthetic aperture radar (SAR) images or vegetation optical depth (VOD) retrievals derived from microwave sensors (these vegetation indices have been used in phenology studies in [84]). Finally, it could also be interesting to investigate the application of the proposed approach to features estimated from other kinds of sensors such as compressive multitemporal/multispectral sensors [135][136], which acquire the images using a compressed format.

[135] J. M. Ramirez and H. Arguello. "Spectral Image Classification From Multi-Sensor Compressive Measurements". In: *IEEE Transactions on Geoscience and Remote Sensing* 58.1 (2020), pp. 626–636. DOI: 10.1109/TGRS.2019.2938724.

[136] H. Vargas and H. Arguello. "A Low-Rank Model for Compressive Spectral Image Classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 57.12 (2019), pp. 9888–9899.