

Master thesis and internship[BR]- Master's thesis : Study of compressive sensing in view of space imaging applications[BR]- Integration Internship

Auteur : Gramegna, Sabrina

Promoteur(s) : Georges, Marc

Faculté : Faculté des Sciences appliquées

Diplôme : Master en ingénieur civil en aérospatiale, à finalité spécialisée en "aerospace engineering"

Année académique : 2021-2022

URI/URL : <http://hdl.handle.net/2268.2/14390>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.



Study of compressive sensing in view of space imaging applications

Author:

Sabrina GRAMEGNA

Promoter:

Dr. Marc GEORGES

Committee member:

Prof. Serge HABRAKEN

Dr. Murielle KIRKOVE

Prof. Marc VAN DROOGENBROECK

Master's thesis carried out to obtain the degree of Master of Science in Aerospace
Engineering University of Liège
School of Engineering and Computer Science
Academic year 2021 – 2022

Contents

1	Introduction	1
1.1	Context	1
1.2	Methodology	3
1.3	Objective and structure of the work	3
2	State of the art	5
2.1	Mathematical approach	6
2.2	Reconstruction of the image	6
2.2.1	ℓ_1 norm minimization without noise	7
2.2.2	ℓ_1 norm minimization with noise	8
2.2.3	Total Variation minimization	8
2.2.4	Deep learning reconstruction	9
2.3	Super resolution	10
3	Simulation analysis	13
3.1	Parameters of comparison	14
3.1.1	Mean Squared Error	15
3.1.2	Peak Signal to Noise ratio	15
3.1.3	Structure Similarity Index Method	15
3.1.4	Computation time and number of iteration	17
3.2	Noise Reduced Dynamic Synthetic Coded Aperture Imaging Camera	17
3.2.1	Decoding algorithm	18
3.2.2	Simulation	18
3.2.3	Results and discussion	20
3.3	Compressive Sensing decoding algorithms	24
3.3.1	Total Variation by NonLocal Regularization [23]	24
3.3.2	Total variation based on the Augmented Lagrangian [18]	26
3.3.3	Iterative Shrinkage-Thresholding Algorithm-Net [24]	29
3.3.4	Compressed Sensing using the Convolutional neural Network [25]	32
3.3.5	Results and discussion	36
3.3.6	Total variation with noise	45

3.3.7	Final comparison	47
4	DiffuserCam	48
4.1	Reconstruction Principles	49
4.1.1	Gradient Descent	49
4.1.2	Alternating Direction Method of Multipliers	50
4.2	Construction of the DiffuserCam	51
4.3	Results and discussion	54
5	Space application	60
5.1	Multiband Ultrawide SpectroImager for Cryosphere Analysis	61
5.2	SUper-Resolved comPReSSive InStrument for Earth observation	62
5.3	Super resolved imaging spectrometer in the medium Infrared	64
5.4	UV-VIS Hyperspectral imager on orbiter for stellar spectrophotometry	66
5.5	Camera operating in the MIR-TIR for sky observation and real time detection of Near Earth Objects	69
5.6	COSMITO & PIGNOLETTO	70
6	Conclusion and future work	73
6.1	Conclusion	73
6.2	Future work	74
7	Acknowledgments	75
	Appendices	76
A	Decoding algorithm comparison	76
B	Decoding TV algorithm comparison with noise	79

List of Figures

1	Classical representation of an optical instrument composed by an object, a lens and a sensor. From [1].	1
2	Representation of the lensless instrument composed by an object, a mask, a sensor and a computer to reconstruct the image. From [1].	2
3	Diagram of the single-pixel camera. The DMD reflects only part of the light towards the single photodetector. The part of light transmitted is random, dictated by the pattern of the DMD. From [14].	5
4	Representation of the subspace of the unit vectors in the ℓ_1 norm (red octahedron) and the subspace of possible solutions (green plane). The point X represents the solution that minimize the ℓ_1 norm. Adapted from [13].	7
5	Signal reconstruction of a CS image using an iterative algorithm unrolled over a deep neural network. Adapted from [20].	10
6	Scheme representing the working principle of SR method. Adapted from [21]	11
7	Scheme of an instrument using CS and SR approach. Adapted from [22]	12
8	Images used for the simulations that constitute the set 11. The size of the images is 256×256 pixels for all the images except "Fingerprint" and "Flintstones". These last have 512×512 pixels.	14
9	Scheme of the NoRDS-CAIC showing the position of the components. From [8]	17
10	Graphical representation of a MURA mask and a Random aperture mask of dimensions 129×129 and 128×128 respectively.	19
11	SSIM maps reporting the difference between the MURA mask and the random mask for the images "Monarch", "Parrots", "Flintstones" and "Fingerprint".	20
12	Evolution of the parameters MSE, PSNR, SSIM and CPU time cost in function of the number of random masks for 4 different images.	22
13	Representation of the SSIM map when the number of masks is 1 or 40 for the image "Flintstones".	23
14	Representations of the reconstructed images of "Monarch" after the application of an increasing number of masks.	24
15	Scheme of the ISTA-Net method. Each phase is composed of a forward and backward transforms that are symmetric. Each transform is composed of a two linear convolutional operators separated by a rectified linear unit (ReLU). From [24].	30

16	Representation of a single phase of the $ISTA_{Net^+}$ algorithm. From [24].	32
17	Graphical representation of the CS- Net^+ to reconstruct the images using the DL method. From [25].	35
18	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Peppers".	37
19	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Lena". . .	38
20	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Flintstones".	39
21	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Fingerprint".	40
22	SSIM maps of the TVAL3, TVLNR, $ISTA_{Net^+}$ and CS- Net^+ when the CS ratio is 25% for $ISTA_{Net^+}$ and 20% for the other algorithms. The red square highlights the most complicated part to reconstruct. The images are "Peppers" (a), "Lena" (b), "Flintstones" (c) and "Fingerprint" (d).	42
23	Original image and images reconstructed by the TVAL3, TVLNR, $ISTA_{Net^+}$ and CS- Net^+ . The CS ratio is 25% for $ISTA_{Net^+}$ and 30% for the other algorithms. For each image, "Peppers" (a), "Lena" (b), "Flintstones" (c) and "Fingerprint" (d), the red square highlights the most complicated part to reconstruct.	44
24	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the two TV reconstruction algorithms, in function of the percentage of noise for the image "Peppers".	45
25	Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the two TV reconstruction algorithms, in function of the percentage of noise for the image "Fingerprint".	46
26	Scheme representing the structure of a DiffuserCam. Adapted from [9]	48
27	representation of components and assembly of the DiffuserCam.	52
28	Schematic representation of the phases of the image capture.	53
29	PSF of the DiffuserCam, the light source is 30 cm away and the exposure time is 3 ms. . .	53
30	Raw image of the letter C at the sensor with an exposure time of 21 ms.	54
31	ADMM and GD reconstruction of the letter A for the downsampling factor f that varies. The values of CPU time cost, PSNR and SSIM and the edge of each reconstruction are reported too.	55
32	Original images disposed on a self-illuminated screen used as object of the DiffuserCam.	56

33	ADMM and GD reconstructions of letters C, S, L when the exposure time varies at 3 ms (1), 21 ms (2) and 40 ms (3).	56
34	ADMM and GD reconstructions of letter C when the distance between the object and the sensor is 5 cm, 10 cm, 20 cm, and 30 cm.	58
35	Picture of the thicker diffuser and the corresponding PSF captured by the sensor.	59
36	Picture of the SLM and the corresponding PSF captured by the sensor.	59
37	CS optical design of the MUSICA instrument [30].	62
38	Working principle of the SURPRISE instrument. From [28].	63
39	Elements that constitute the SISSI instrument [21].	64
40	Representation of acquisition mode of the SISSI instrument [21].	65
41	Optical design of the SISSI instrument [21].	66
42	Optical scheme of the instrument [32].	67
43	Optical scheme with a doiable arm configuration [29]	68
44	Optical sceme of the instrument [32].	69
45	Optical scheme of the COSMITO instrument [30].	71
46	Optical scheme of the PIGNOLETTO instrument [30].	72

Acronyms

ADMM	Alternating Direction Method of Multipliers
ASI	Italian Space Agency
BB	Barzilai-Borwein
BCS	Block-based Compressed Sensing
CCD	Charge Couple Device
CNN	Convolutional Neural Network
CS	Compressive Sensing
CS-Net	Compressed Sensing using the Convolutional neural Network
DL	Deep Learning
DMD	Digital Micromirror Device
FOV	Field Of View
GD	Gradient Descent
IR	Infra Red
ISS	International Space Station
ISTA	Iterative Shrinkage-Thresholding Algorithm
LCD	Liquid Crystal Display
LCP	Liquid Crystal Plate
MMA	Micro-Mirror Array
MSA	Micro-Shutter Arrays
MSE	Mean Squared Error
MURA	Modified Uniformly Redundant Arrays



NEO Near Earth Objects

NLM Nonlocal Means

NLSA Non-monotone Line Search Algorithm

NoRDS-CAIC Noise Reduced Dynamic Synthetic Coded Aperture Imaging Camera

NR Nonlocal Regularization

PSF Point Spread Function

PSNR Peak Signal to Noise ratio

RIP Restricted Isometric Property

SLM Spatial Light Modulator

SR Super Resolution

SSIM Structure Similarity Index Method

TV Total Variation

TVAL3 TV based on the augmented Lagrangian

TVLNR TV based image CS recovery algorithm by nonlocal regularization

URA Uniformly Redundant Arrays

Abstract

The Compressive Sensing (CS) theory is a useful technique to retrieve a sparse signal, for example the images, from a number of measurement fewer than the minimum required by the Nyquist theorem. This application is different from the classical imaging, because the utilization of lenses is not essential, and the image is not directly acquired, a recovering algorithm is necessary.

The reconstruction is not unique, there are several methods that are investigated. Between the classical methods that use the Total Variation (TV) minimization and the Deep Learning (DL) methods, the simulations have established that the DL methods are the most suitable for a better reconstruction in less time.

This work has been proposed by the Centre Spatial de Liège (CSL), in order to analyze the utilization of the CS technique in space application. The CS space instruments use the concept of Super Resolution (SR) and single pixel imaging to retrieve the scene in spectral bands different from the visible one.

The objective is to study and analyze the basic concept of CS technique and its feasibility for the space application. At the end of this study, the CS technique appears to be an optimal solution for the space application, specially in UV and IR spectral bands, with great quality.

1 Introduction

1.1 Context

Traditionally, the method of acquisition of an image is realized using lenses or mirrors and a sensor, such as shown in Figure 1.

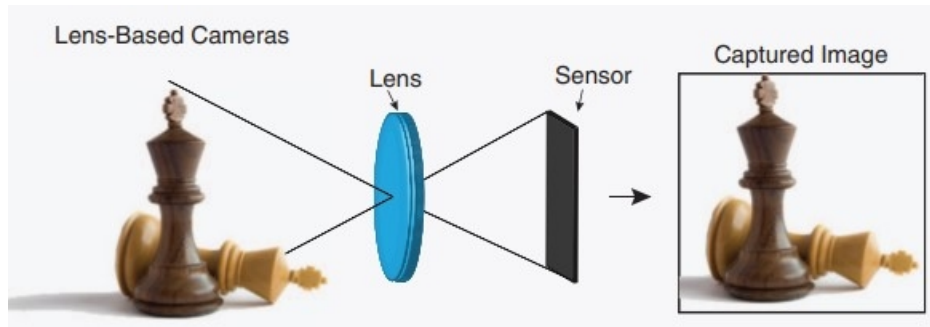


Figure 1: Classical representation of an optical instrument composed by an object, a lens and a sensor. From [1].

This kind of method is really useful when the target is in the spectrum of the visible wavelengths. Some problems appear when the object is in wavelengths far from the visible ones, like in far infrared, far and extreme UV and in γ and X-rays. The limitations are due to the difficulty or impossibility to manufacture the lenses. Also, the sensors, in these wavelengths, are extremely expensive and thus not always available. The instruments tend to be large, their dimensions makes them difficult to manufacture.

A solution can be the lensless imaging. This technique does not use a lens, but other optical elements to collect the light and send it to a sensor. The principal benefits of the lensless imaging are a small thickness and different kinds of form, the large possible spectrum (only limited by the detector sensitivity), the reduced cost, and the small amount of data. Figure 2 represents an example of lensless imaging.

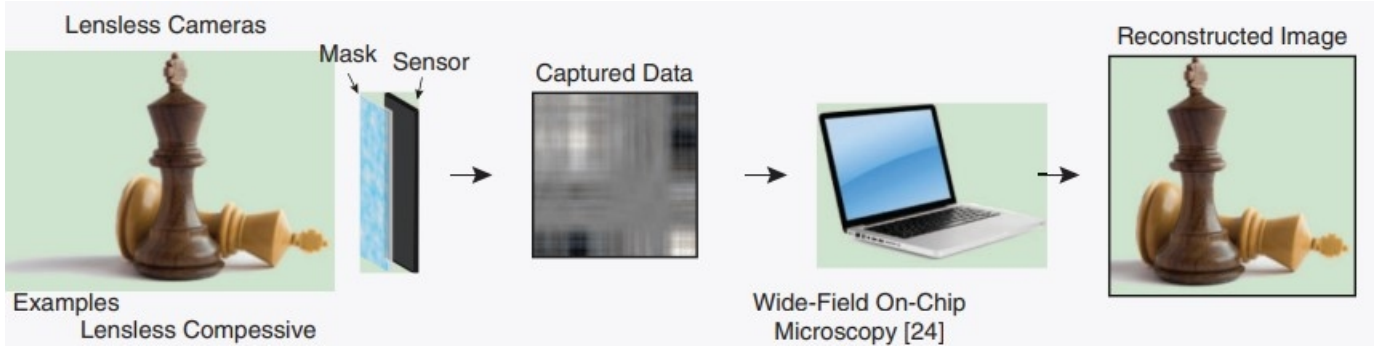


Figure 2: Representation of the lensless instrument composed by an object, a mask, a sensor and a computer to reconstruct the image. From [1].

One of the first examples of the lensless imaging is the Pinhole camera or *camera obscura* (see [1]) in which the light, coming from an object, passes through a single aperture to reach the sensor. This technique required very long exposure time to have an acceptable quality.

The idea of the pinhole camera was improved by replacing the single hole by a mask containing several apertures, the so-called "coded aperture imaging (CAI)". This technique was primarily used for X-ray and γ -ray imaging because this kind of rays contain so much energy that they interact at the level of atoms and make impossible the use of lenses and mirrors. There are several examples of space missions based on the coded aperture imaging, such as the SWIFT [2] space telescope used to observe the γ -ray bursts, the AXGAM mission [3] to study the γ -ray bursts but also the x-ray binary cyclotron lines, the SuperAGILE [4] used to observe the γ -ray sky, the EXIST [5] mission that uses several masks arranged radially to observe the hard X-ray sky and the INTEGRAL [6] mission conceived to study the hard X-ray emission from the galactic nuclear region.

In order to reduce the cost and the size of the cameras while keeping a good resolution, the coded aperture imaging is used for the creation of cameras like the FlatCam [7] or the Noise Reduced Dynamic Synthetic Coded Aperture Imaging Camera (NoRDS-CAIC) [8], which will be described in section 3.2. Another example of a camera that uses the principle of the lensless imaging is the DiffuserCam [9]. It uses a diffuser in front of the sensor to reduce the number of measurements. A practical case of a DiffuserCam will be shown in section 4.

Lensless imaging is not used only in space imaging, but there are some other applications. Let's cite, the microbial identification for the mid-infrared spectroscopy [10] and a holographic microscopy system for monitoring cells that can be used for the biological experiments in the International Space Station (ISS) [11].

1.2 Methodology

The mathematical concept that drives the coded aperture imaging is the Compressive Sensing (CS). The theory asserts that the image can be reconstructed with a great quality, taking only a few measurements with respect to the ones dictated by the Nyquist theorem. The latter expresses that the frequency at which the measurements have to be taken has to be at least twice the maximum frequency of the signal. An in-depth study of the CS and the algorithms used to reconstruct the image will be carried out in section 2.

The CS ratio establishes the rate of measurements of the image, dictated by the number of apertures in the mask. The light coming from the object passes through the mask, then each hole creates a shadow on the sensor where there is the superposition of each beam of light. The image captured by the sensor has to be decoded by a computational algorithm to reconstruct the image of the original object. This means that this technique does not allow to reproduce the images directly at the sensor level, but they need a post-processing reconstruction algorithm.

The mask used in the coded aperture imaging is a set of holes that let light pass through. The holes can either be arranged randomly, or they can be structured following a set of rules that is user defined. In this report, the structured mask that is used is the Modified Uniformly Redundant Arrays (MURA) [12]. A comparison between the MURA and random masks will be carried out in section 3.2.2. There are several methods to construct a mask. The amplitude masks can be created using a programmable Liquid Crystal Display (LCD), which allows to pass only a part of the light, or a Spatial Light Modulator (SLM), particularly the Digital Micromirror Device (DMD), in which the mirrors are arranged in such a way that part of the light is sent in the target direction, and the other part in another direction. The SLM can also be used to create the phase mask, in which the part of the light has a different phase with respect to the original one.

The data, recollected at the sensor, are used to reconstruct the original image using some algorithms described in section 3.3.

1.3 Objective and structure of the work

The objective of this project is to study the theory of the Compressive Sensing and its application in space imaging. Starting from the mathematical definition, it is interesting to analyze how a signal can be reconstructed with few measurements. The particularity is that the reconstruction is not unique, there exist a lot of decoding algorithm to retrieve the original image. Therefore, it has to be found what is the best one that gives the best results, particularly the most useful one



for the space application.

The CS could really be one of the future technique in space exploration and Earth observation. It allows to create small instruments with excellent resolution not only in visible light, but especially in wavelengths not totally experimented yet.

Therefore, this projects aims to find how the CS technique could be used for the future of the space technology, and study the possible future missions using the CS technique.

After the introduction, a state of the art about the CS will be carried out in section 2. Then a numerical analysis of the coded aperture camera, for example the NoRDS-CAIC, and a comparison between different methods to reconstruct the CS images will be carried out in section 3. Afterwards, a practical case based on the CS theory, the DiffuserCam, will be described in section 4. Finally, the use of CS for space application and the future that it can have will be discussed in section 5.

2 State of the art

Since the important information of the image is contained in less pixels than the total number of pixels N , the Compressive Sensing (CS) theory aims to take only fewer measurements M with respect to the total number of pixels to reconstruct the image. This technique has several advantages, like the diminution of the amount of data to transmit, the simplicity and reduction of the architecture, and the improvement of the single pixel detector for specific application. Indeed, the single pixel detector is used especially for the Infra Red (IR) imaging, in which the price of the traditional detector is a thousand times higher than the price of visible light detectors [13].

One of the first examples of a CS camera is the single-pixel camera developed by the Rice University [14], [15]. This camera is able to capture the image using only a single photodetector instead of an array of detectors.

The structure is very simple, it is composed of a DMD, two lenses, a single photodetector and an A/D converter. A light source illuminates the scene. The light of the scene passes through the first lens, and it is sent to the DMD, which reflects only a part of the light. Then the light goes through the second lens to be concentrated on the single photodetector. Finally, the output voltage is digitized by the A/D converter. This method, shown in Figure 3, is repeated to acquire M values, that are used to reconstruct the image.

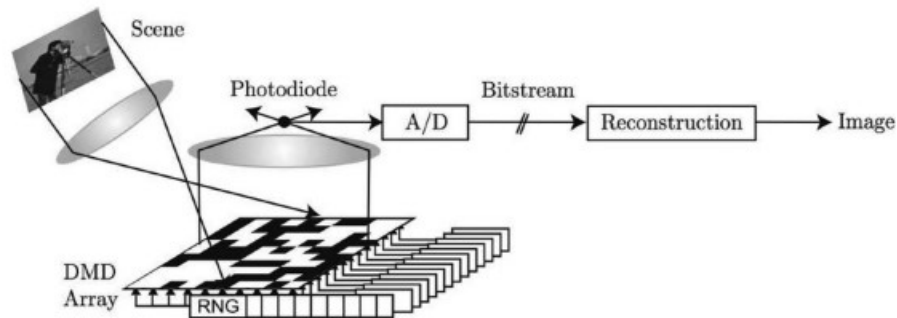


Figure 3: Diagram of the single-pixel camera. The DMD reflects only part of the light towards the single photodetector. The part of light transmitted is random, dictated by the pattern of the DMD. From [14].

2.1 Mathematical approach

The signal of the image is represented by $x \in \mathbb{R}^N$, after M measurements, with $M \ll N$ and $\frac{M}{N}$ the CS ratio, an image $y \in \mathbb{R}^M$ is formed on the sensor [15]. This last is expressed in the following form

$$y = \Phi x + e = \Phi \Psi s + e, \quad (1)$$

where Φ represents the $M \times N$ measurement matrix, that is non-adaptive. It means that it does not depend on the image x , Ψ the $N \times N$ basis matrix, s is the coefficient sparse vector (only $K \ll N$ coefficients are nonzero) and e represents the sensor noise and the model loss. The number of measurements M is related to the coefficient sparse vector in such a way that $M = O(K \log(N/K))$. The number of measurements should satisfy this relation in order to have a robust reconstruction of the vector x .

Therefore, the characteristics of the CS is that it directly takes a compressed representation y with M linear measurement without acquiring N measurements and then compresses them.

Since $M \ll N$ the problem is ill-posed, this means that there is an infinite number of solutions to reconstruct the image. The critical problems are thus to establish the matrix Φ and the reconstruction algorithm. One possible solution is that the sensing matrix $A = \Phi \Psi$ satisfies the Restricted Isometric Property (RIP) ([14], [15], [16]).

The property of order $K = 1, 2, \dots$ with the isometry constant $\delta_K \in [0, 1]$ is satisfied if

$$(1 - \delta_K) \|x\|_{\ell_2}^2 \leq \|Ax\|_{\ell_2}^2 \leq (1 + \delta_K) \|x\|_{\ell_2}^2, \quad (2)$$

where $\|x\|_{\ell_2} = \sqrt{\sum_{i=1}^N x_i^2}$ holds for all K -sparse vectors x . Therefore, if the matrix A respects the equation (2), it follows the RIP.

2.2 Reconstruction of the image

The dimensions of the image, y , at the sensor are smaller than the original image. This means that some information is lost. Since $M \ll N$, for a given y , there are multiple solutions x' such that $\Phi x' = y$. However, the matrix Φ can be designed in such a way that a sparse/compressible x' can be retrieved quite well from the values of y [15].

2.2.1 ℓ_1 norm minimization without noise

The classical solution to recover the initial image, without considering the noise, is the minimization of the ℓ_1 norm [17]:

$$\hat{x} = \arg \min \|x\|_{\ell_1} \text{ such that } Ax = y. \quad (3)$$

where $\|x\|_{\ell_1} = \sum |s_i|$. This method allows to reconstruct K -sparse and compressible vectors stably using $M \geq O(K \log(N/K))$ random measurements.

The ℓ_1 minimization can be represented in a 3-dimensional space.

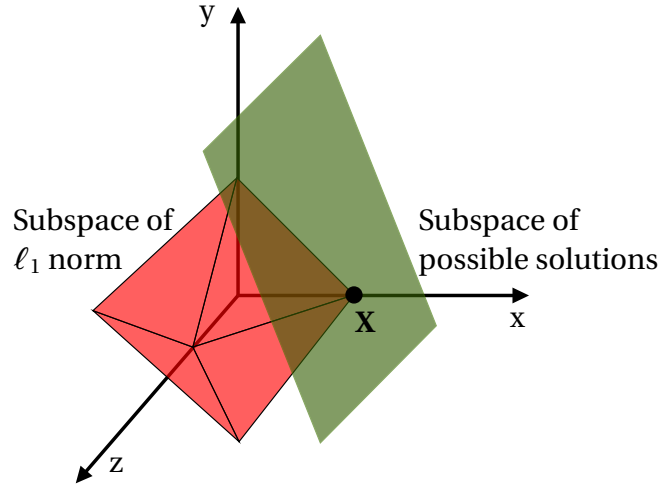


Figure 4: Representation of the subspace of the unit vectors in the ℓ_1 norm (red octahedron) and the subspace of possible solutions (green plane). The point X represents the solution that minimize the ℓ_1 norm. Adapted from [13].

The polygon in red in Figure 4 is an octahedron, and it represents the set of the unit vectors in the ℓ_1 norm. Since the vector x is sparse (it contains a lot of zero coordinates), it has to be at one of the vertices of the octahedron. All the possible solutions x that give $\Phi x = y$ constitute the green plane in Figure 4 and this plane passes through the point x . In order to constitute a solution, the plane has to intersect the octahedron only at the point x .

The measurement matrix Φ is taken randomly to have a great probability of finding a plane that intersects the octahedron only in point x . This random method does not always work because some planes pass inside the octahedron. However, the three-dimensional figure does not represent the real space of the image because it is of the order of thousands or millions of dimensions. In a space with this number of dimensions, the octahedron is called the cross polytope, and in a

multiple dimensional space it is extremely pointy. This means that if a plane passes through a vertex, it almost certainly misses the inside of the cross polytope. Therefore, thanks to the high dimensions of the images, the ℓ_1 minimization will almost always give the right solution.

2.2.2 ℓ_1 norm minimization with noise

Until now, the reconstruction method has not considered the noise, and the image on the sensor was measured without any measurement error. In reality, the signal itself and the sensor, for example, are sources of errors. Thus, the methodology has to be stable even in presence of noise. The noise e in equation (1) is a stochastic or deterministic error $\|e\|_{\ell_2} \leq \varepsilon$, where ε is a limitation on the noise magnitude.

The noise is taken into condition in (3) as:

$$\hat{s} = \arg \min \|s'\|_{\ell_1} \quad \text{such that} \quad \|y - \Phi\Psi s'\|_{\ell_2} < \varepsilon. \quad (4)$$

This new solution has to satisfy the equation

$$\|\hat{s} - s\|_{\ell_2} < C_N \varepsilon + C_K \sigma_K(x), \quad (5)$$

where C_N and C_K are respectively the noise and the approximation error constants, and $\sigma_K(x)$ is the ℓ_2 error that occurs when s is approximated using its largest K terms [15].

2.2.3 Total Variation minimization

Another method for the reconstruction of the CS images is the Total Variation (TV) minimization. This method is better than the ℓ_1 minimization because it recovers the images with better quality: the edges and boundaries are more faithful, that is a key characteristic for the reconstruction of the images [18].

The TV method aims to reconstruct the image x using the compressibility of the images with respect to their discrete gradient. The term $x_{m,n}$ represents the pixel of x at the position m, n . The discrete gradients $D_1 x_{m,n}$ and $D_2 x_{m,n}$ are defined as

$$\begin{cases} D_1 x_{m,n} = x_{m+1,n} - x_{m,n} \\ D_2 x_{m,n} = x_{m,n+1} - x_{m,n} \end{cases}. \quad (6)$$

The TV and its minimization are given by

$$\|x\|_{\text{TV}} = \sum \sqrt{(D_1 x)^2 + (D_2 x)^2} = \sum |(\nabla x)_{m,n}|. \quad (7)$$

$$(\text{TV}) \min \|x\|_{\text{TV}} \text{ such that } \|y - \Phi \Psi s\|_{\ell_2} \leq \varepsilon. \quad (8)$$

The equation (8) is a non-linear and non-differential method that makes the resolution too complex. One possible solution is to use the least-squares problem combined with the TV method in the following equation

$$\min_x \|y - \Phi x\|_{\ell_2}^2 + \lambda \mathcal{R}(x). \quad (9)$$

The first term, $\min_x \|y - \Phi x\|_{\ell_2}^2$, represents the least-squares problem and estimates the image accuracy. The second term $\mathcal{R}(x)$, the TV model $\mathcal{R}(x) = \|x\|_{\text{TV}}$, represents the regularization term that enforces an image prior. The constant λ is a tradeoff between the least-square problem and the TV method, and thus between the accuracy and regularization. The equation (9) is now a convex problem that can be resolved easily [1].

Among all the methods that use the TV for the reconstruction of the image, the TV based image CS recovery algorithm by nonlocal regularization (TVLNR) and an improved TV based on the augmented Lagrangian method algorithm (TVAL3) are used in the simulations in section 3.

2.2.4 Deep learning reconstruction

The previous algorithms are not applicable in practice because the computation time and costs are too high. This will be demonstrated in the next section 3, during the comparison between the algorithms.

Deep Learning (DL) algorithms are a real innovation in this field because they directly learn the inverse mapping to the original image from the CS measurements. The results are better both in accuracy and computation time and costs domains [19].

The DL algorithms are based on neural network models that have been developed to solve some complex optimization problems, such as the ℓ_1 minimization described in section 2.2.1.

The DL for the CS reconstruction methods consists in simulating the classical iterative CS algorithms using the neural network structure [20]. Figure 5 represents the scheme of the DL for the iterative reconstruction.



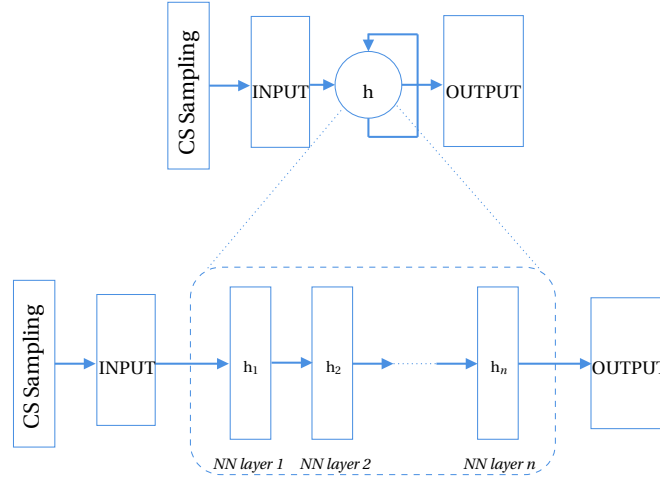


Figure 5: Signal reconstruction of a CS image using an iterative algorithm unrolled over a deep neural network. Adapted from [20].

The technique that is shown in Figure 5, called algorithm unrolling, identifies each iteration into a network layer and assemble some layers together. All parameters, including the transformation matrix, are discriminately learned, unlike the traditional iterative methods where the value of parameters are chosen at the beginning. After the unrolling, the training data is fed through the network and the stochastic gradient descent is used to improve and optimize the parameters [20].

The improvement of the reconstruction iterative method with the DL is given by the fact that the data is processed faster with the neural network layers. Moreover, the number of layers in DL is smaller with respect to the number of iterations in the classical methods and so the computation time is reduced drastically [19].

Among all the DL methods, the improved Iterative Shrinkage-Thresholding Algorithm (ISTA) cast into deep network form (ISTA-Net⁺) and the imaged Compressed Sensing using the Convolutional neural Network (CS-Net⁺) are used in the simulation section 3.

2.3 Super resolution

The Super Resolution (SR) technique aims to improve the quality of a given image. In order to achieve the objective, the best is to increase the density of the pixel of the detector. This type of solution is optimal, despite the increase in cost. Actually, the high resolutions detectors with tiny pixel size are really expensive, especially in wavelengths different from the visible ones.

A cheaper solution is using an SLM with a number of elements $N \times N$ times greater than the number of pixels of the detector. In this way, each pixel of the detector is associated with $N \times N$ elements of the SLM. Since the dimensions of the detector are lower than the dimensions of the SLM, a specific number of pixels of the SLM are focused on a macropixel of the detector that acts like a single pixel camera. Figure 6 shows the principle of the SR.

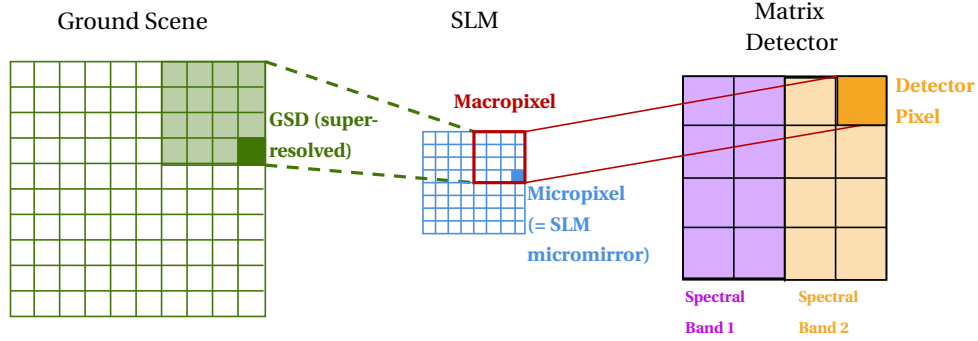


Figure 6: Scheme representing the working principle of SR method. Adapted from [21]

In the case of IR images, in which the more suitable sensor is a single pixel detector, it is not possible to increase the number of pixels to improve the resolution. Therefore, the dimensions of each element of the SLM have to be equal or larger to the size of the diffraction-limited spot blur created by the foreoptics at the level of the image plane to assure the acquisition of the highest spatial frequency presented in the image [21].

In space application, the production of the high spatial resolution IR images remains a problem. One possible solution is to combine the SR and the CS techniques. In this manner, CS directly acquires compressed data stream without using compression board on the satellite and, at the same time, it allows to acquire super resolved images [22]. Figure 7 shows an idea of a super-resolved instrument.

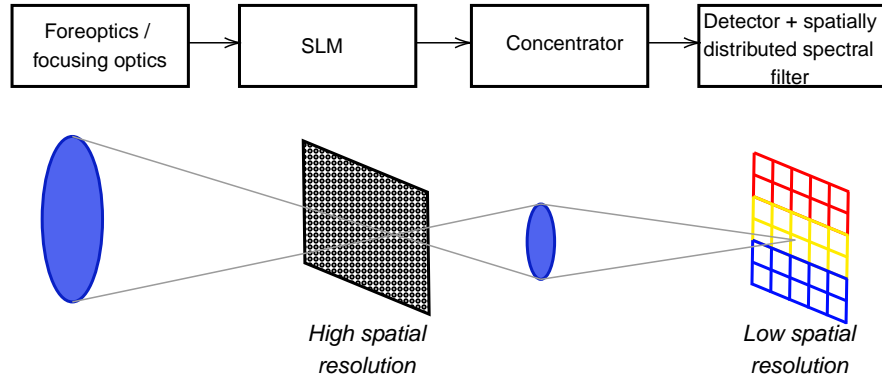


Figure 7: Scheme of an instrument using CS and SR approach. Adapted from [22]

In Figure 7, four principal blocks are presents, starting from the left:

- The foreoptics/focusing optics: it acquires and focuses the light on the next block;
- SLM: the image coming from the previous block passes through a binary spatial mask produced by the SLM. The mask has a great number of pixels that corresponds to the number of pixels of the final image;
- Concentrator: it concentrates the image coming from the SLM on the detector;
- Detector+filter: the detector has lower dimensions than the SLM and the final image. The filter allows the separation of the different spectral bands.

3 Simulation analysis

In this section, the concept of Code Aperture Imaging is simulated thanks to the Noise Reduced Dynamic Synthetic Coded Aperture Imaging Camera (NoRDS-CAIC) decoding algorithm. It will be described in details in section 3.2. The algorithm developed by [8] is studied and optimized for the Compressive Sensing in coded aperture imaging. However, the CS is not used only for the coded aperture imaging, but also, for example, for the DiffuserCam (see section 4), and for the single pixel imaging, like in space application (see section 5).

Therefore, four different algorithms (TVLNR, TVAL3, ISTA-Net⁺ and CS-Net⁺), conceived for the application of CS in all domains, are compared in section 3.3. These algorithms are executed using the same set of 11 gray test images for a fair comparison. The original images are shown in Figure 8.

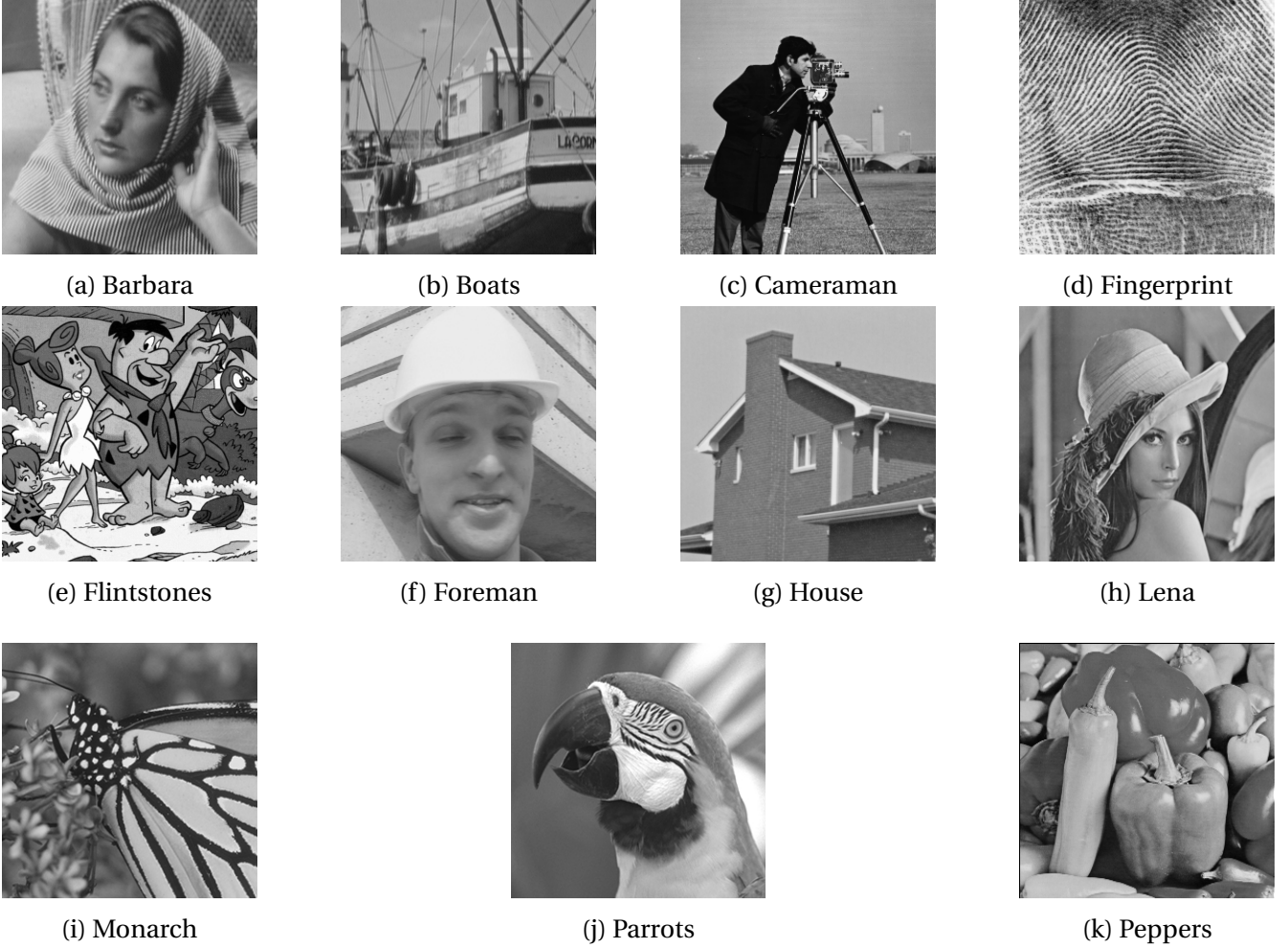


Figure 8: Images used for the simulations that constitute the set 11. The size of the images is 256×256 pixels for all the images except "Fingerprint" and "Flintstones". These last have 512×512 pixels.

All the decoding algorithms come from the original developers [8], [18], [23], [24], and [25], who have shared them freely. In this work, the algorithms have been adapted to work on a device equipped with an Intel Core i7-8565 CPU, an 8 GB RAM and NVIDIA GeForce MX230 GPU. The different parameters used to compute the comparison are defined in the next section 3.1

3.1 Parameters of comparison

The parameters used for the evaluation of the quality of the images are the Mean Squared Error (MSE), the Peak Signal to Noise ratio (PSNR), and the Structure Similarity Index Method (SSIM) [26]. Furthermore, other parameters, like the computation time (CPU time cost) and the number of iterations for the algorithms using the TV method, are also taken into account to

compare the decoding algorithms.

3.1.1 Mean Squared Error

The MSE estimates the absolute error between the original image and the decoded one. Each pixel of the images is compared as follows

$$\text{MSE} = \frac{1}{N_x N_y} \sum_{n=1}^{N_x} \sum_{m=1}^{N_y} [\hat{g}(n, m) - g(n, m)]^2, \quad (10)$$

where N_x, N_y represents the dimensions of the image, in this case the image is squared of dimension N , \hat{g} represents the reconstructed image and g the original image. The best quality is achieved when the value is the closest to zero.

For the simulations, the MATLAB function *immse* is used. This function computes automatically the equation (10).

3.1.2 Peak Signal to Noise ratio

The PSNR gives the ratio between the maximum possible signal power and the power of the distorting noise, represented by the MSE. The value is the logarithm term of decibel scale because the dynamic range varies between the largest and the smallest possible values.

The PSNR is computed as follows,

$$\text{PSNR} = 10 \log_{10} \frac{\text{peakval}^2}{\text{MSE}} \quad (11)$$

where the MSE is the previous parameter and the peakval (Peak Value) is the maximal value of the image. For an 8-bit image, the peakval is 255 and if the image is normalized, the peakval is 1. The quality is optimal when the value is as large as possible.

The equation (11) is implemented in the MATLAB function *psnr*, that will be used for the simulations.

3.1.3 Structure Similarity Index Method

The SSIM expresses the quality of the image perceived by the human eye. Contrary to the previous parameters that compare the pixels between the two images one by one, the SSIM compare the images in their totality.



The mathematical expression of the SSIM is given by:

$$\text{SSIM}(\hat{g}, g) = [l(\hat{g}, g)]^\alpha \cdot [c(\hat{g}, g)]^\beta \cdot [s(\hat{g}, g)]^\gamma \quad (12)$$

where l is the luminance, that expresses the brightness between the two images, c is the contrast, that evaluates the ranges between the brightest and darkest region of the two images, and s is the structure, that compares the local luminance to find the similarity and the dissimilarity between the images; α, β, γ are positive constants. The luminance l , the contrast c and the structure s of an image are defined as follows:

$$l(\hat{g}, g) = \frac{2\mu_{\hat{g}}\mu_g + C_1}{\mu_{\hat{g}}^2 + \mu_g^2 + C_1} \quad (13)$$

$$c(\hat{g}, g) = \frac{2\sigma_{\hat{g}}\sigma_g + C_2}{\sigma_{\hat{g}}^2 + \sigma_g^2 + C_2} \quad (14)$$

$$s(\hat{g}, g) = \frac{\sigma_{\hat{g},g} + C_3}{\sigma_{\hat{g}}\sigma_g + C_3} \quad (15)$$

where $\mu_{\hat{g}}, \mu_g$ are the local means, $\sigma_{\hat{g}}, \sigma_g$ are the standard deviations and $\sigma_{\hat{g},g}$ is the cross-covariance. C_1, C_2, C_3 are regularization constants, non-negative real numbers, that avoid the instability in some parts of the images where the local mean or the standard deviation is close to zero. These constants have to be close to zero.

For the simulations, the MATLAB function *ssim* is used. The function uses the equation (12) to compute the SSIM. By default, α, β, γ are imposed equal to 1. Therefore, the value of the SSIM becomes

$$\text{SSIM}(\hat{g}, g) = \frac{(2\mu_{\hat{g}}\mu_g + C_1)(2\sigma_{\hat{g}}\sigma_g + C_2)}{(\mu_{\hat{g}}^2 + \mu_g^2 + C_1)(\sigma_{\hat{g}}^2 + \sigma_g^2 + C_2)} \quad (16)$$

The regularization constants are defined in MATLAB as:

$$C_1 = (0.01L)^2 \quad (17)$$

$$C_2 = (0.03L)^2 \quad (18)$$

$$C_3 = C_2/2 \quad (19)$$

where L is called the DynamicRange, and it depends on the class of the image, that represents the



number of bits uses to represent the image. If this one is single or double class $L = 1$, if it is of uint8 class $L = 255$. In this work, the class of images is double.

The equation (16) gives the values of the SSIM normalized. The best quality is obtained when the value is the closest possible to 1.

The MATLAB function also gives the local SSIM map. This map represents the zones where the reconstructed image report the larger differences with respect to the original image. The pixels are in gray scale, varying from black to white with increasing value of the SSIM. Thus, the black areas are the parts where the reconstruction is not well computed.

3.1.4 Computation time and number of iteration

The CPU time cost indicates the time that the algorithm spends to decode the compressed images. For the classical iterative CS algorithms, the CPU time cost depends on the number of iterations used to reconstruct the image. Therefore, it increases with the number of iterations. The results in section 3.3.5 will demonstrate that there is a huge difference in CPU time cost between the classical iterative methods and the DL algorithms.

3.2 Noise Reduced Dynamic Synthetic Coded Aperture Imaging Camera

The NoRDS-CAIC consists in a lensless camera that uses the concept of the coded aperture imaging. The structure of the NoRDS-CAIC is very simple, it is composed of a programmable Liquid Crystal Display (LCD) and a Charge Couple Device (CCD) to detect the photons. Therefore, the final structure has modest dimensions, the longest dimension is 19 cm. Figure 9 shows the structure of the NoRDS-CAIC [8].

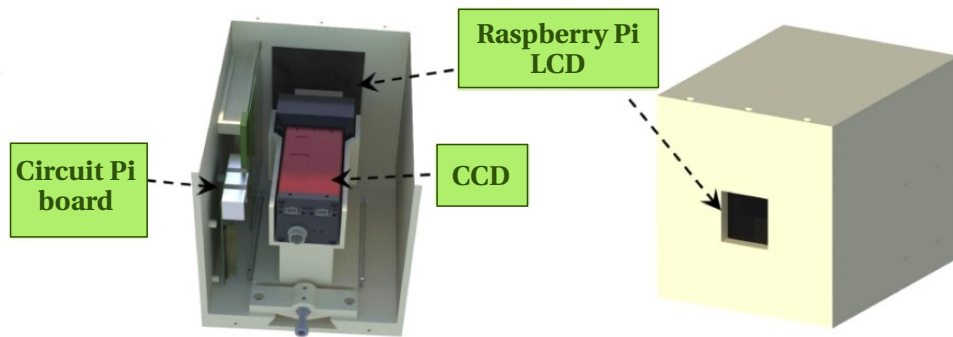


Figure 9: Scheme of the NoRDS-CAIC showing the position of the components. From [8]

The LCD has the function to create a random amplitude mask in front of the detector in order to block part of the light. The innovation of the NoRDS-CAIC with respect to a FlatCam, for example, is that the programmable LCD is capable to rapidly display different random masks one after the other. The number of masks, in fact, will affect the quality of the image, as the results, in section 3.2.3, demonstrate.

3.2.1 Decoding algorithm

The reconstruction of the image is based on the minimization of the loss function E

$$E(x, y) = \sum_k |I(x, y; k) - I_g(x, y; k)|^2 \quad (20)$$

where k is the number of masks, I is the image itself, I_g is the reconstructed image. The image and the reconstructed one are defined as

$$I_g(x, y; k) = O_g(x, y; k)H(x, y; k) + N'(x, y; k) \quad (21)$$

$$I(x, y; k) = O(x, y)H(x, y; k) \quad (22)$$

where O is the target, O_g is the guessed target, H is the PSF and N' is the noise. The guessed target is modified and improved after each mask.

3.2.2 Simulation

The simulation is computed using the source code provided by the authors of the NoRDS-CAIC in [8]. For the simulation, four images with different dimensions are chosen from the set of 11 images reported in Figure 8, and it will be used in the next section 3.3.5. The first two images are "Monarch" and "Parrots" with 256×256 pixels, and the other two are "Flintstones" and "Fingerprint" with 512×512 pixels.

The basic parameters, the wavelength, the distance between the object and the mask, the distance between the mask and the CDD and the pixel size are left unchanged with respect to the initial code.

For the analysis of the code aperture imaging, it is interesting to compare the random mask with a coded mask like the MURA. The codification of the MURA mask is operated in MATLAB using the following algorithm given in the original paper [12]. The dimension L of the mask is a

prime number such that

$$L = 4m + 1 \quad m = 1, 2, 3, \dots, \quad (23)$$

Since the dimensions of the images are 256×256 and 512×512 , the dimensions of the MURA masks are $L = 65 \Rightarrow m = 16$ and $L = 129 \Rightarrow m = 32$ respectively, in order to have the same dimensions of the random mask.

In MATLAB, the mask is coded as a matrix

$$A = \{A_{i,j}\}_{i,j=1}^L, \quad (24)$$

whose coefficients are defined as follows:

$$A_{ij} = \begin{cases} 0 & \text{if } i = 0, \\ 1 & \text{if } j = 0, i \neq 0, \\ 1 & \text{if } C_i C_j = +1, \\ 0 & \text{otherwise} \end{cases} \quad (25)$$

where $C_{i,j} = 1$ if there exists an integer x , $1 \leq x < L$, such that $i, j = \text{mod}_r x^2$, otherwise $C_{i,j} = -1$.

Figure 10 shows the graphical representations of a MURA mask and a random mask.

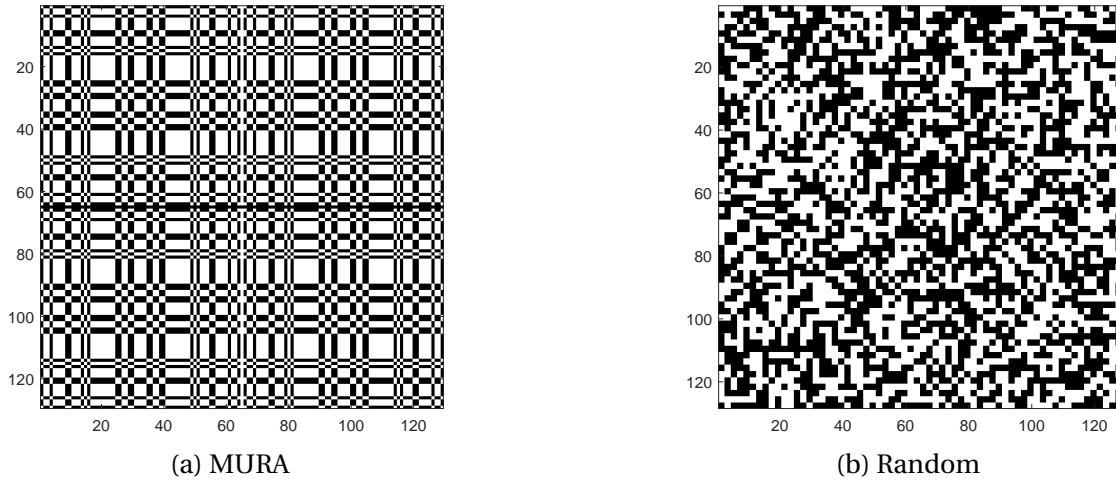


Figure 10: Graphical representation of a MURA mask and a Random aperture mask of dimensions 129×129 and 128×128 respectively.

3.2.3 Results and discussion

The first simulations study the difference between a random and MURA masks. Table 1 reports the value of SSIM, MSE, PSNR and CPU time cost for the four images using the two different masks, MURA and random mask. The difference is also evaluated comparing the SSIM maps.

	Monarch		Parrots		Flintstones		Fingerprint	
	MURA	Random	MURA	Random	MURA	Random	MURA	Random
SSIM	0.68	0.86	0.67	0.85	0.37	0.58	0.31	0.65
MSE	0.007	0.002	0.058	0.025	0.072	0.051	0.050	0.036
PSNR	21.58	26.98	12.36	15.99	11.40	12.91	12.89	14.42
CPU time cost	0.25	0.14	0.27	0.25	0.55	0.41	0.50	0.67

Table 1: Comparison of SSIM, MSE, PSNR and CPU time cost using the MURA and random masks for the images "Monarch", "Parrots", "Flintstones" and "Fingerprint".

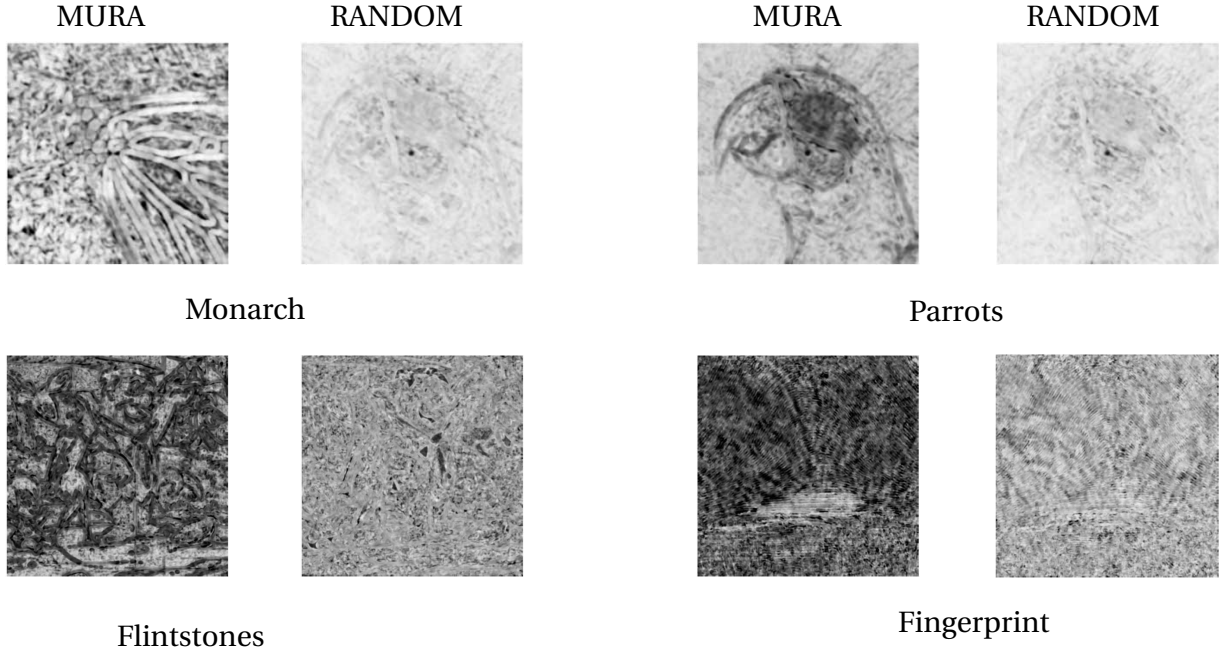


Figure 11: SSIM maps reporting the difference between the MURA mask and the random mask for the images "Monarch", "Parrots", "Flintstones" and "Fingerprint".

The results of the simulation in Table 1 show a clear difference between the two masks. The

random mask shows better values for all the parameters and all the images. The only drawback is that the values are not stable because the masks are random and the distribution of the light over the sensor is not always the same. However, for the next simulations the random mask is chosen because of its better results and, since there are more than one mask, the stability increases. The red values highlight the better values for all parameters. The image "Monarch" results to be the best one, even if the values of the image "Parrots" are comparable. Therefore, the decoding algorithm results to have better performance when the size of the images is smaller.

The difference between the two masks is also shown in Figure 11. The black parts are the most critical and the worst reconstructed ones. The images decoded with a MURA mask are mostly black, and so they are not well-defined.

Therefore, since it has been shown that the MURA masks are not suitable for the NoRDS-CAIC, the random masks are used for the rest of the section.

The next step is to increase the number of the random masks to see the effect on the different comparison parameters. The graphs 12 show the evolution of the different parameters when the number of masks increase up to 40.

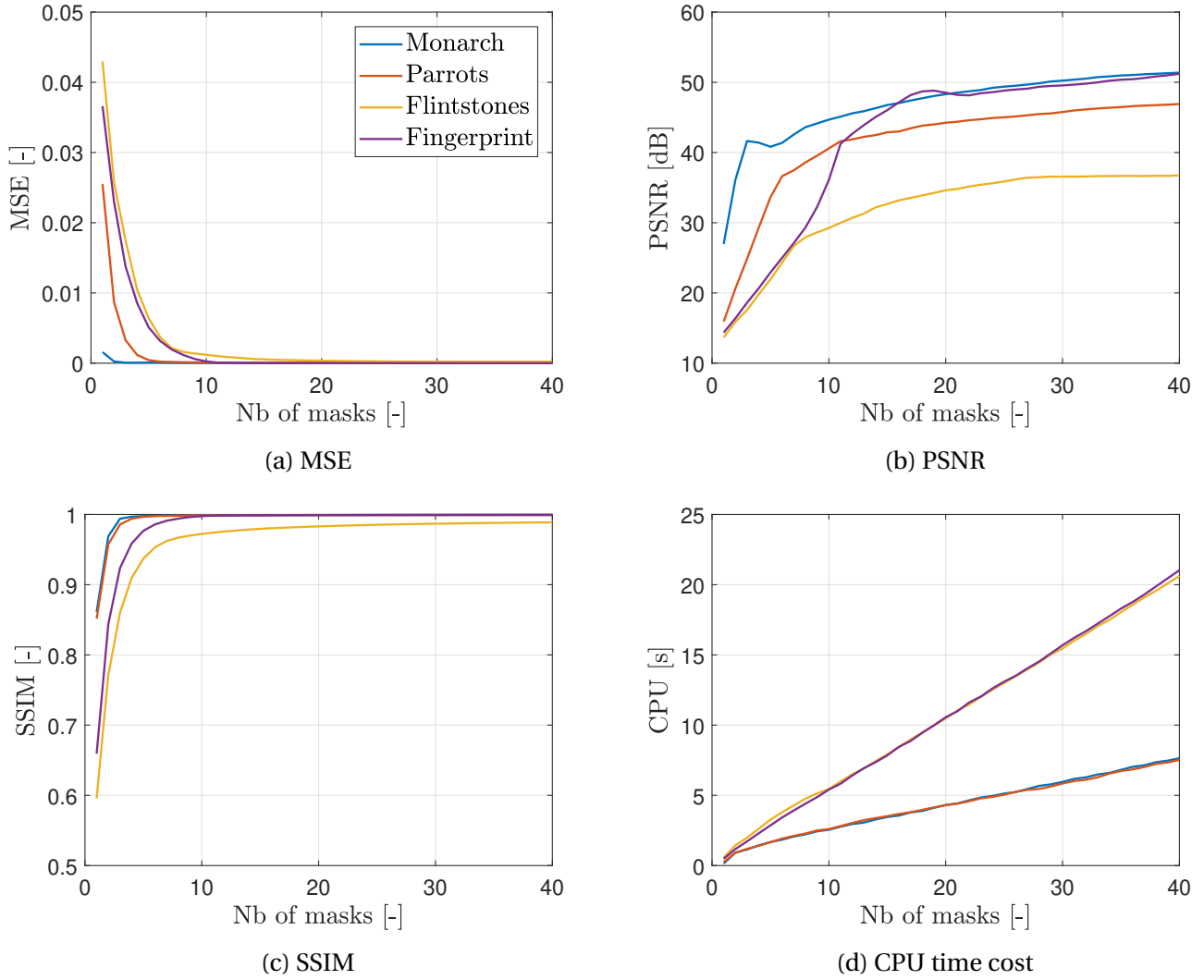
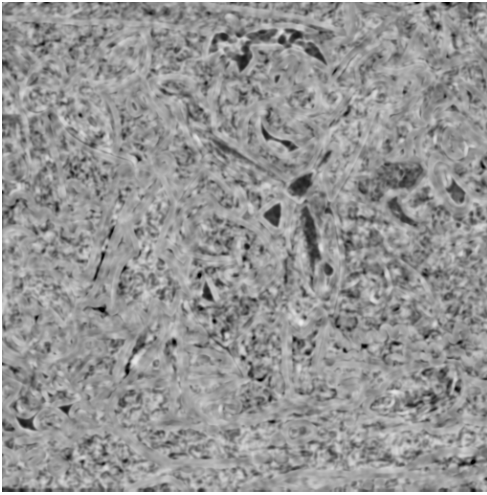


Figure 12: Evolution of the parameters MSE, PSNR, SSIM and CPU time cost in function of the number of random masks for 4 different images.

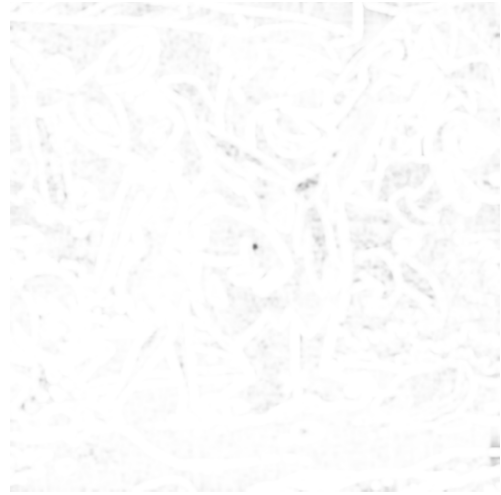
Starting from the evolution of the MSE in function of the number of masks in Figure 12a, it is demonstrated that the absolute error decreases rapidly with the increasing of the number of masks. At already 15 different random masks, the MSE is almost zero for all the images. The evolution of the PSNR in Figure 12b is not stable, like the evolution of the MSE. The difference is caused by the randomness of the distribution of the holes on the mask. Thus, when the number of masks is low the values are not yet stable. Regardless, the value increases with the number of masks, and the stability of values is reached when the number of masks is larger than 20. The evolution of the SSIM in Figure 12c is opposite to the evolution of the MSE. The value increases rapidly with the number of masks, at already 10 masks the value reaches the final value

for each image. The "Flintstones" image does not reach a value of 1 like the other images, therefore the black pixels are still present and the image is not completely reconstructed. For this reason, the "Flintstones" image is chosen to show the difference in the SSIM map when one mask is used [13a](#) and when 40 masks are used [13b](#) in Figure 13. For the other images, whose value of SSIM is almost 1, the SSIM map after 40 masks has all pixels white, that means that the image is well-reconstructed.

When only one mask is used, there are a lot of black pixels, which is not the case when the number of masks increases. After 40 masks, the number of black pixels is very low, but still present.



(a) 1 mask



(b) 40 masks

Figure 13: Representation of the SSIM map when the number of masks is 1 or 40 for the image "Flintstones".

The evolution of the CPU time cost in Figure [12d](#) is linear with the number of the masks, and it is greater for the "Flintstones" and "Fingerprint". This result was expected because there are as iterations as the number of masks, and if the image has greater dimensions, there are more pixels to decode.

Finally, only 20 different random masks are required to obtain excellent results for every parameter with a relatively low CPU time cost.

Figure [14](#) reports the reconstructed images when the number of masks increases.

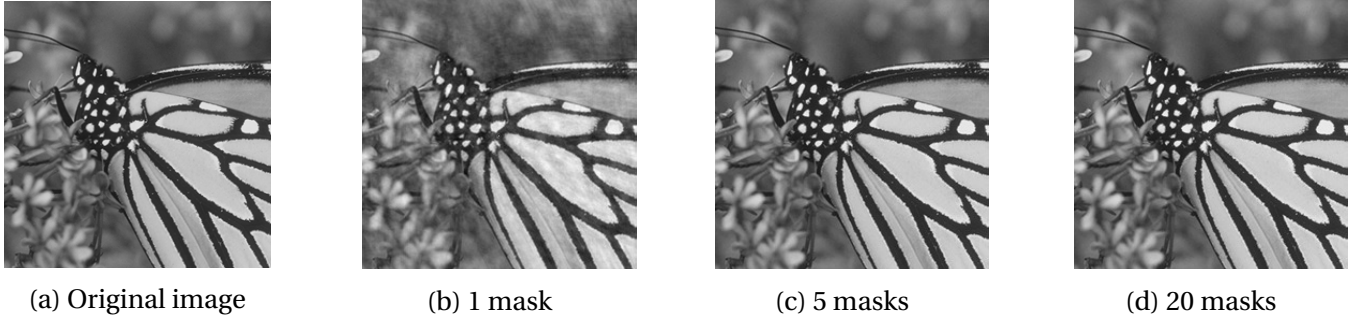


Figure 14: Representations of the reconstructed images of "Monarch" after the application of an increasing number of masks.

The Figure 14 highlights the difference in the reconstruction when the number of masks increases. Above 20 masks, there are no significant improvements in the reconstruction, therefore only the reconstruction after 20 masks is shown in Figure 14. When only one mask is used, in the Figure 14b, the details are not well reconstructed and the contrast between the black lines and the white parts is not optimum. When the number of masks is 20, in Figure 14d, the reconstruction is almost perfect, all the details are well-defined, and the contrast is maximum.

3.3 Compressive Sensing decoding algorithms

In this section, the decoding algorithms for the reconstruction of CS images will be analyzed in detail. The first two algorithms are based on the classical TV methods, and the last two algorithms use the DL method. The mathematical developments that are at the basis of the algorithms will be presented. During the simulations, the CS ratio varies from 1% to 50% in order to show the improvement when the number of measurements increases.

The algorithms are slightly modified to use the same 11 gray images, and the parameters described in section 3.1 are computed in the same way for all the algorithms.

At the end, after the comparison between the algorithms, the best will be chosen with the purpose of the space imaging.

3.3.1 Total Variation by NonLocal Regularization [23]

The TVLNR is a decoding algorithm based on the TV method explained in section 2.2.3. The particularity of this method is that it uses the nonlocal means (NLM) filtering that is a convenient tool for the reconstruction of the images. In fact, the NLM filtering is especially efficient in sharp-

ening edges and preserving image details.

The concept of NML filtering is to establish a nonlocal regularization (NR) to recover CS images. The mathematical expression of the NR is given by

$$\mathcal{NR}(\mathbf{x}) = \sum_{x_i \in \mathbf{x}} \|x_i - \mathcal{NL}\mathcal{F}(x_i)\|_2^2 = \sum_{x_i \in \mathbf{x}} \|x_i - \mathbf{w}_i^T \boldsymbol{\kappa}_i\|_2^2 \quad (26)$$

where x_i is a single pixel of the image \mathbf{x} , $\mathcal{NL}\mathcal{F}(x_i)$ is the NLM filtered intensity value of the pixel x_i , $\boldsymbol{\kappa}_i$ is the column vector containing the central pixels around x_i in the search window and \mathbf{w}_i is the column vector containing the corresponding weights.

The previous equation (26) can be written in matrix form as

$$\mathcal{NR}(\mathbf{x}) = \|\mathbf{x} - \mathbf{W}\mathbf{x}\|_2^2, \quad \mathbf{W}_{i,j} = \begin{cases} W_{i,j}, & \text{if } x_j \in \boldsymbol{\kappa}_i \\ 0, & \text{otherwise} \end{cases}. \quad (27)$$

The improved TV based image CS recovery algorithm by nonlocal regularization is given by the combination of equation (7) and equation (27), such that

$$\min_x \|x\|_{\text{TV}} + \alpha \|\mathbf{x} - \mathbf{W}\mathbf{x}\|_2^2. \quad (28)$$

The equation (28) is not easy to solve directly because of non-differentiability and non-linearity of the two terms. The solution is to introduce two variables \mathbf{u} and \mathbf{w} such that

$$D\mathbf{u} = \mathbf{x} \quad \text{and} \quad \mathbf{u} = \mathbf{w} \quad (29)$$

The solution of the minimization can be found using the augmented Lagrangian function, that is defined as:

$$\mathcal{L}_{\mathcal{A}}(x, u, w) = \|x\|_1 - \mathbf{v}^T(D\mathbf{u} - x) + \frac{\beta}{2} \|D\mathbf{u} - x\|_2^2 + \alpha \|w - Ww\|_2^2 - \gamma^T(u - w) + \frac{\theta}{2} \|u - w\|_2^2 + \frac{\mu}{2} \|Au - y\| - \lambda^T(Au - y), \quad (30)$$

where β, θ, μ are the regularization parameters and $\mathbf{v}, \gamma, \lambda$ are the Lagrangian multipliers.

The final solution is given by minimizing one variable at a time among x, u, w while maintaining the two others fixed.



The iterative process of the minimization is given by

$$(x^{(k+1)}, u^{(k+1)}, w^{(k+1)}) = \min_{x, u, w} \mathcal{L}_{\mathcal{A}}(x, u, w). \quad (31)$$

The algorithm implementing the TVLNR method to reconstruct the CS images is given by

Algorithm 1 Algorithm to reconstruct a CS image with TVLNR method

Input: The observed measurement \mathbf{b} , the measurement matrix \mathbf{A} and $\beta, \mu, \theta, \alpha$.

Initialization: $\mathbf{u}_0 = \mathbf{A}^T \mathbf{b}$, $\nu_0 = \gamma_0 = \lambda_0 = 0$; $\mathbf{w}_0 = \mathbf{x}_0 = 0$.

while Outer stopping criteria unsatisfied **do**

while Inner stopping criteria unsatisfied **do**

 Minimize \mathbf{w} while the other parameters are fixes;

 Minimize \mathbf{u} while the other parameters are fixes;

 Compute the weight \mathbf{W} ;

 Minimize \mathbf{x} while the other parameters are fixes;

end while

 Update multipliers; ν, γ, λ

end while

Output: Final CS restored image; $\tilde{\mathbf{u}}$.

The algorithm 1 is implemented in MATLAB by the authors of [23]. The parameters are chosen as follows: $\mu = 2^8 = 256$, $\theta = 2$, $\beta = 2^5 = 32$ and $\alpha = 16$. If the values of θ and β are higher, the number of iterations exceeds the maximum number of iterations imposed. This latter has been increased to 1000 with respect to 300 of the original code, because when the CS ratio is too low (1% for example), 300 iterations are too few to obtain good values. The augmentation of the maximum number of iterations causes an increase in the computation time too.

To assure that the algorithm is stable in the presence of noise, the latter is implemented in the same way as the TVAL3 algorithm in order to have a fair comparison.

3.3.2 Total variation based on the Augmented Lagrangian [18]

The TVAL3 algorithm is also based on the method of TV described in section 2.2.3, and on augmented Lagrangian multiplier approach.

The particularity of the TVAL3 is that it uses a splitting and alternating minimization idea, the non-monotone line search and Barzilai-Borwein steps to improve the convergence.

The minimization of TV is defined as

$$\min_{x_i, u} = \sum_i \|x_i\|_{\text{TV}}, \text{ such that } Au = y \text{ and } D_i u = x_i \text{ for all } i, \quad (32)$$

where x is the image, u represents the pixel, D is the discrete gradient, A is the measurement matrix and y is the observed measurement.

The minimization of the augmented Lagrangian function is computed using an alternative way: the variable-splitting technique. The TV regularization term is split into two terms using a new slack variable x , as the equation (33) reports. The term $\|x_i\| - v_i^T (D_i u - x_i) + \frac{\beta_i}{2} \|D_i u - x_i\|_2^2$, present in the augmented Lagrangian function, derives from the variable-splitting technique. The function is thus given by:

$$\mathcal{L}_A(x_i, u) = \sum_i \left(\|x_i\| - v_i^T (D_i u - x_i) + \frac{\beta_i}{2} \|D_i u - x_i\|_2^2 \right) - \lambda^T (Au - y) + \frac{\mu}{2} \|Au - y\|_2^2 \quad (33)$$

where v, λ are the Lagrangian multipliers, and β, μ are regularization parameters.

The reconstruction is determined by the minimization of the augmented Lagrangian function $\mathcal{L}_A(x_i, u)$. The difficulty comes from the non-linearity and non-differentiability of the previous equation, like in TVLNR method.

Thanks to the introduction of the variable x , the variables x_i and u can be minimized separately, while updating the Lagrangian multipliers.

The minimization of x_i is defined using the Shrinkage-like formula as follows:

$$x_{i,k+1} = \text{shrike}(D_i u_k; v_i, \beta_i) = \max \left\{ \left\| D_i u_k - \frac{v_i}{\beta_i} \right\| - \frac{1}{\beta_i}, 0 \right\} \frac{(D_i u_k - v_i / \beta_i)}{\|D_i u_k - v_i / \beta_i\|}. \quad (34)$$

The non-monotone line search and Barzilai-Borwein steps are used for the minimization of the variable u , that is equivalent to

$$\min_u Q_k(u) \triangleq \sum_i \left(-v_i^T (D_i u - x_{i,k+1}) + \frac{\beta_i}{2} \|D_i u - x_{i,k+1}\|_2^2 \right) - \lambda^T (Au - y) + \frac{\mu}{2} \|Au - y\|_2^2. \quad (35)$$

$Q_k(u)$ is a quadratic function whose gradient is $d_k(u)$.

In order to minimize the function (35) the iterative one-step steepest descent method is used as



follows

$$u_{k+1} = u_k - \alpha_k d_k. \quad (36)$$

The parameter α_k is defined, by the Barzilai-Borwein (BB) method, in two different ways:

$$\alpha_k = \frac{s_k^T s_k}{s_k^T b_k} \quad (37)$$

$$\alpha_k = \frac{s_k^T b_k}{b_k^T b_k} \quad (38)$$

where $s_k = u_k - u_{k-1}$ and $b_k = dk(u_k) - dk(u_{k-1})$.

This method is validated by the non-monotone line search algorithm (NLSA). The NLSA requires the verification of the Armijo condition, defined as:

$$Q_k(u_k - \alpha_k d_k) \leq C_k - \delta \alpha_k d_k^T d_k \quad (39)$$

where C_k is given by

$$\begin{aligned} C_{k+1} &= (\eta P_k C_k + Q_k(u_{k+1})) / P_{k+1} \\ P_{k+1} &= \eta P_k + 1 \end{aligned} \quad (40)$$

and the variables δ and η are between 0 and 1.

All the previous equations are implemented in the final algorithms for the TVAL3 reconstruction method.

Algorithm 2 Alternating Minimization Scheme used in the final algorithm of TVAL3

```

Initialize  $0 < \delta, \rho, \eta < 1$  and starting points  $x_{i,0}, u_0$ ;
Set  $Q_0 = 1$  and  $C_0 = \mathcal{L}_A(x_{i,0}, u_0)$ ;
while Inner stopping criteria unsatisfied do
    Compute  $x_{i,k+1}$  based on shrinkage-like formula;
    set  $\alpha_k$  through BB-like formula;
    while Nonmonotone Armijo condition unsatisfied do
        Backtrack  $\alpha_k = \rho \alpha_k$ ;
    end while
    Compute  $u_{k+1}$  by one-step steepest descent method
    Set  $C_{k+1}$ 
end while

```

The previous algorithm is used in the final algorithm for TVAL3 reconstruction defined below.

Algorithm 3 Algorithm to reconstruct a CS image with TVAL3 method

Initialize $v_i^0, \beta_i^0, \lambda^0, \mu^0$, and starting point x_i^0, u^0 for all i ;
while Outer stopping criteria unsatisfied **do**
 Set $x_{i,0}^{k+1} = x^k$ and $u_0^{k+1} = u^k$;
 Find minimizers x_i^{k+1} and u^{k+1} of the augmented Lagrangian function by means of the Algorithm 2 starting from $x_{i,0}^{k+1}$ and u_0^{k+1} ;
 Update multipliers to attain v_i^{k+1}, λ^{k+1} ;
 Choose new regularization parameters $\beta_i^{k+1} \geq \beta_i^k$ and $\mu^{k+1} \geq \mu^k$;
end while
Final CS restored image.

The algorithm is implemented in MATLAB by the authors of [18]. The parameters are chosen as follows: $\mu = 2^{12} = 4096$, $\beta = 2^6 = 64$, v_i^0 , $\lambda^0 = 0$, $\delta = 10^{-5}$, $\rho = 0.6$ and $\eta = 0.9995$. In this case, the number of iterations has been left the same, 300, because the simulations don't reach the maximum number.

The original codes have been modified to allow reading images in MATLAB, because the original ones are in color and the test images are gray.

In addition to the previous simulations, a second simulation is computed when the noise affects the reconstruction of the images.

The values when the noise is added are compared with the TVLNR method in section 3.3.6.

3.3.3 Iterative Shrinkage-Thresholding Algorithm-Net [24]

The ISTA-Net algorithm is based on the traditional ISTA that aims to minimize the ℓ_1 norm. The ISTA algorithm is used in deep learning in such a way that each phase of the ISTA-Net corresponds to an ISTA-like iteration, as shown in Figure 15, and all parameters are not fixed at the beginning.

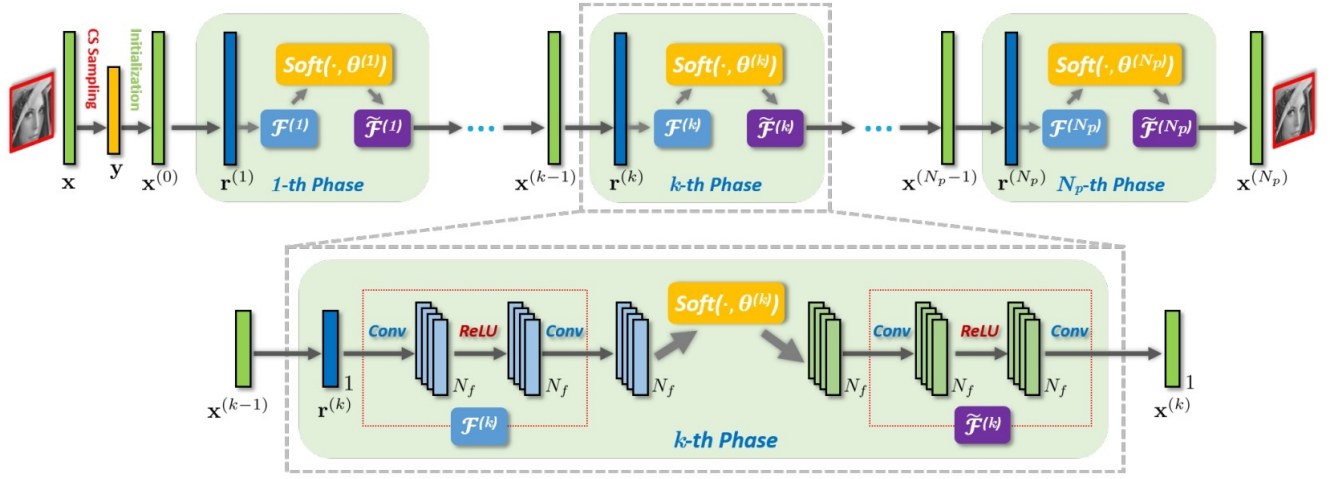


Figure 15: Scheme of the ISTA-Net method. Each phase is composed of a forward and backward transforms that are symmetric. Each transform is composed of a two linear convolutional operators separated by a rectified linear unit (ReLU). From [24].

The computational complexity of the ISTA-Net corresponds to multiple iterations of the ISTA algorithm, and it is more than 100 times faster than the correspondent non-DL algorithms. The ISTA-Net is further improved in ISTA-Net⁺ to upgrade the CS decoding using the fact that the residuals of images are more compressible.

The optimization problem that ISTA-Net⁺ aims to solve is the equation (9), with some modifications, as follows:

$$\min_x \frac{1}{2} \|\Phi x - y\|_2^2 + \lambda \|\Psi x\|_1. \quad (41)$$

where Ψx represents the transform coefficients of x and Ψ is the transform.

The algorithm of ISTA solves the equation (41) implementing the following iterative way:

$$\mathbf{r}^{(k)} = x^{(k-1)} - \rho \Phi^T (\Phi x^{(k-1)} - y) \quad (42)$$

$$x^{(k)} = \underset{text{x}}{\operatorname{argmin}} \frac{1}{2} \|x - \mathbf{r}^{(k)}\|_2^2 + \lambda \|\Psi x\|_1 \quad (43)$$

where k is the iterative index and ρ is the step size.

The challenge in ISTA resolution is to find *a priori* the transform Ψ and the parameters ρ and λ . The solution of equation (43) requires a lot of iterations to have good reconstructions, which implies an increasing in computation time.

ISTA-Net uses a non-linear transform function $\mathcal{F}(\cdot)$ with learnable parameters. It consists of two linear convolutional operators separated by a Rectified Linear Unit (ReLU) as shown in Figure 15, where the convolutional operators are constituted by N_f filters. Finally, the function $\mathcal{F}(\cdot)$ is expressed in matrix form,

$$\mathcal{F}(\mathbf{x}) = \mathbf{BReLU}(\mathbf{Ax}) \quad (44)$$

where \mathbf{A} and \mathbf{B} are the two convolutional operators.

Therefore, the recursive equation (43) of ISTA algorithm becomes

$$\mathbf{x}^{(k)} = \arg\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{r}^{(k)}\|_2^2 + \lambda \|\mathcal{F}(\mathbf{x})\|_1. \quad (45)$$

In ISTA-Net, both equations (42) and (45) are included into two different modules of the k -th phase of the algorithm.

The first module, $\mathbf{r}^{(k)}$ **Module**, aims to solve equation (42) and so find the value of $\mathbf{r}^{(k)}$. The result of the first module is the input for the second module, $\mathbf{x}^{(k)}$ **Module**, that aims to resolve equation (45) to compute $\mathbf{x}^{(k)}$. Supposing that $\mathbf{r}^{(k)}$ and $\mathcal{F}(\mathbf{r}^{(k)})$ are the mean values of \mathbf{x} and $\mathcal{F}(\mathbf{x})$, thus the following approximation is made:

$$\|\mathcal{F}(\mathbf{x}) - \mathcal{F}(\mathbf{r}^{(k)})\|_2^2 \approx \alpha \|\mathbf{x} - \mathbf{r}^{(k)}\|_2^2 \quad (46)$$

where α is a scalar. Introducing the equation (46) into the equation (45) the computation of \mathbf{x} is given by:

$$\mathbf{x}^{(k)} = \arg\min_{\mathbf{x}} \frac{1}{2} \|\mathcal{F}(\mathbf{x}) - \mathcal{F}(\mathbf{r}^{(k)})\|_2^2 + \theta \|\mathcal{F}(\mathbf{x})\|_1, \quad (47)$$

where θ is given by $\theta = \lambda\alpha$.

The characterization of the DL algorithm is that the parameters are learnable. For the ISTA-Net, the parameters are grouped in $\Theta = \{\rho^{(k)}, \theta^{(k)}, \mathcal{F}^{(k)}, \tilde{\mathcal{F}}^{(k)}\}_{k=1}^{N_p}$, where $\tilde{\mathcal{F}}$ is the inverse of \mathcal{F} and N_p is the number of total phases.

In the improved version of ISTA-Net, ISTA-Net^+ , $\mathbf{x}^{(k)}$ is defined as

$$\mathbf{x}^{(k)} = \mathbf{r}^{(k)} + \mathbf{w}^{(k)} + \mathbf{e}^{(k)} \quad (48)$$

where $\mathbf{e}^{(k)}$ represents the noise and $\mathbf{w}^{(k)}$ represents some missing frequency component in $\mathbf{r}^{(k)}$. The term $\mathbf{w}^{(k)}$ is defined as $\mathbf{w}^{(k)} = \mathcal{R}(\mathbf{x}^{(k)})$, where $\mathcal{R}(\cdot)$ is a linear operator determined by $\mathcal{R} = \mathcal{G} \circ \mathcal{D}$, \mathcal{D} corresponds to N_f filters and \mathcal{G} to 1 filter.

In ISTA-Net^+ , the transform function \mathcal{F} becomes $\mathcal{F} = \mathcal{H} \circ \mathcal{D}$, where \mathcal{H} is composed by two linear



convolutional operators and one ReLU. In this case, the optimization problem is

$$\min_{\mathbf{x}} \frac{1}{2} \left\| \mathcal{H}(\mathcal{D}(\mathbf{x})) - \mathcal{H}(\mathcal{D}(\mathbf{r}^{(k)})) \right\|_2^2 + \theta \|\mathcal{H}(\mathcal{D}(\mathbf{x}))\|_1 \quad (49)$$

Each phase of ISTA-Net⁺ is represented in the Figure 16.

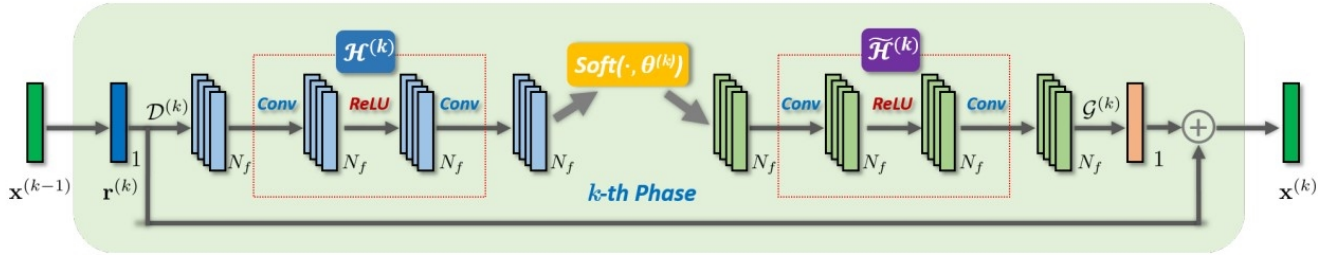


Figure 16: Representation of a single phase of the ISTA-Net⁺ algorithm. From [24].

The algorithm of ISTA-Net⁺ is implemented in Python by the authors of the article [24]. Since it is a DL algorithm, it is not necessary to define the parameters, like in the previous TV algorithms. Only the number of phases is chosen at the beginning. It is $N_p = 9$, because even if the number of phases is larger, the quality of the reconstruction remains almost constant. In this work, some changes are made to the original code. These changes concern principally the TensorFlow, a platform for the machine learning, because the original code uses the first version of the platform. In the actual version of TensorFlow, some functions are changed or modified, therefore the code has been adapted. The computations of PSNR, MSE, SSIM are implemented in MATLAB, and not Python, to have fair comparison.

3.3.4 Compressed Sensing using the Convolutional neural Network [25]

The CS-Net⁺ is based on the Convolutional Neural Network (CNN). The goal of the CS-Net⁺ algorithm is to compute an optimal sampling matrix and to elaborate a reconstruction algorithm that is fast and performs well.

In the algorithm there are three kinds of sampling matrix that can be learned: the floating-point matrix, the $\{0,1\}$ -binary matrix and the $\{-1, +1\}$ -bipolar matrix. The reconstruction method consists in an initial linear reconstruction to generate the initial reconstructed image. Then a non-linear reconstruction improves the quality of the initial reconstructed image using the deep residual network.

One important characteristic of this algorithm is that the sampling matrices are learned from the images in such a way to preserve more information about the images and have a better recon-

struction. The sampling matrices can also be used in the traditional CS reconstruction method to increase their quality.

The reconstruction is based on the block-based compressed sampling, the initial reconstruction and the non-linear reconstruction. These three steps constitute the Block-based Compressed Sensing (BCS).

Initially, the image is divided into non-overlapping blocks of size $B \times B \times l$ where l is the number of channels. The CS measurements can be expressed as

$$y_j = \Phi_B x_j \quad (50)$$

where x_j is the block j of the global image, y_j is the corresponding measurement and Φ_B is the sampling matrix of dimensions $n_B \times lB^2$. Each row of the sampling matrix Φ_B is considered as a filter, the number of filter, n_b , is defined as follows

$$n_b = \left\lceil \frac{M}{N} lB^2 \right\rceil. \quad (51)$$

n_B constitutes also the number of measurements because at each filter corresponds one measurement.

The block-based compressed sampling can be represented by the operation $S(x)$:

$$y = S(x) = W_s * x \quad (52)$$

where $*$ is the convolution operation and W_s represents the n_B filters.

The reconstruction of the image is given by

$$R(y) = D(I(y)), \quad (53)$$

where y are the CS measurements, I represents the initial reconstruction, D the deep reconstruction and R the final reconstruction.

The initial reconstruction is expressed as:

$$\tilde{I}(y) = W_{int} * y \quad (54)$$

where W_{int} represents the lB^2 filters of dimensions $1 \times 1 \times n_B$. The reconstruction of each block is



a vector representing a column of $\tilde{I}(y)$. Then, a combination layer is created to reshape the vectors from dimensions $1 \times 1 \times lB^2$ to a $B \times B \times l$ block and concatenate the blocks to obtain the initial reconstructed image. The formulation of the initial reconstructed image is given by:

$$\tilde{x} = I(y) = \kappa \begin{pmatrix} \gamma(\tilde{I}_{11}(y)) & \cdots & \gamma(\tilde{I}_{1w}(y)) \\ \vdots & \ddots & \vdots \\ \gamma(\tilde{I}_{h1}(y)) & \cdots & \gamma(\tilde{I}_{hw}(y)) \end{pmatrix} \quad (55)$$

where h and w represent the number of blocks in row and column respectively, γ is the reshape function and κ is the concatenation function.

The deep reconstruction includes three operations: feature extraction, non-linear mapping and feature aggregation. The first one produces high dimensional feature using an activation layer in the following way,

$$D_e(\tilde{x}) = Act(W_e * \tilde{x} + B_e) \quad (56)$$

where \tilde{x} comes from the equation (55), W_e represents the d filters of dimensions $f \times f \times l$, B_e is the biases and Act represents the activation function. The latter is the rectified Linear Unit (ReLU), the same used by the ISTA-Net⁺ method.

The non-linear mapping is given by alternating subsequently the residual block, the convolution layer and the activation layer. The mathematical expression is given by:

$$D_{m1}^i(\tilde{x}) = Act\left(D_{m2}^{i-1}(\tilde{x}) + W_{m1}^i * D_{m2}^{i-1}(\tilde{x}) + B_{m1}^i\right) D_{m1}^i(\tilde{x}) = Act\left(D_{m2}^{i-1}(\tilde{x}) + W_{m1}^i * D_{m2}^{i-1}(\tilde{x}) + B_{m1}^i\right) \quad (57)$$

$$D_{m2}^i(\tilde{x}) = Act\left(W_{m2}^i * D_{m1}^i(\tilde{x}) + B_{m2}^i\right), \quad (58)$$

where $i \in \{1, 2, \dots, n\}$, W_{m1}^i, W_{m2}^i represents the d filters, B_{m1}^i, B_{m2}^i are the biases and Act is the ReLU activation function. For the value $i = 0$ the deep reconstruction is given by $D_{m2}^0(\tilde{x}) = D_e(\tilde{x})$.

The final output is given by the feature aggregation operation starting from the high dimensional feature. The operation $D_a(\tilde{x})$ is given by:

$$D_a(\tilde{x}) = W_a * D_{m2}^n(\tilde{x}) + B_a \quad (59)$$

where W_a represents the l filters and B_a the bias. In order to speed up the convergence, a link



between the initial reconstructed image and the previous output is made as follows:

$$D(\tilde{x}) = (\tilde{x}) + D_a(\tilde{x}). \quad (60)$$

All the steps of the reconstruction of the image using the CS-Net⁺ are represented in the Figure 17.

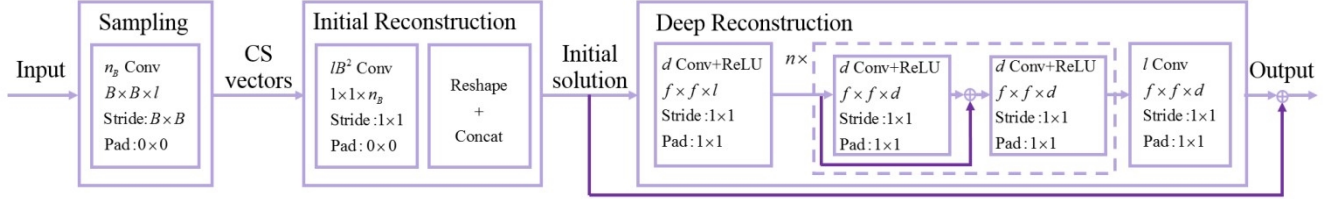


Figure 17: Graphical representation of the CS-Net⁺ to reconstruct the images using the DL method. From [25].

In order to train the sampling and reconstruction network, the loss functions of the initial reconstruction image and the final reconstruction image have to be minimized simultaneously. The respectively equations are given by:

$$l_{int}(\theta, \phi) = \frac{1}{2K} \sum_{i=1}^K \|I(S(x_i; \theta); \phi) - x_i\|_F^2 \quad (61)$$

and

$$l_{deep}(\theta, \phi) = \frac{1}{2K} \sum_{i=1}^K \|R(S(x_i; \theta); \phi) - x_i\|_F^2 \quad (62)$$

where θ, ϕ are the parameters of sampling and reconstruction network respectively, $S(x_i; \theta)$ represents the CS measurements, $I(S(x_i; \theta); \phi), R(S(x_i; \theta); \phi)$ are the initial and final reconstructed outputs.

The algorithm of CS-Net⁺ is implemented in MATLAB by the authors of [25]. The sampling matrix is the floating-point matrix, because it gives better results with respect to the other matrices, see [25]. The CNN is implemented using the toolbox "MatConvnet" in MATLAB, that is a useful tool for computer vision application.

Some values are fixed as the block size $B = 32$, the number of channel $l = 1$, the spatial size of a kernel $f = 3$, the number of features maps $d = 64$, and the number of non-linear mapping layer $n = 5$.

In this work, some changes are made to the original code. These changes concern principally the way the images are read. Indeed, the original images were colored and the code was not able to read the image in the '.tif' format, therefore the code has been modified to read gray images in '.tif' format, like the test images.

3.3.5 Results and discussion

In this section, the algorithms described in sections 3.3.1, 3.3.2, 3.3.3 and 3.3.4 will be compared. The quality parameters MSE, PSNR, SSIM, CPU time cost and the number of iteration for TV algorithms are evaluated for the CS ratios of 1%, 5%, 10%, 20%, 30%, 40% and 50%. For the ISTA-Net⁺, the CS ratios given by the original code are 1%, 4%, 10%, 25%, 40%, 50%.

All the values of the quality parameters are reported in the Table in annex A. In order to have a visual evolution of the parameters of comparison, only four images are chosen, "Peppers", "Lena", "Flintstones" and "Fingerprints". For each image, the graphs below show the evolution of the parameters, comparing the different methods of CS reconstruction.

It has been noticed that the CPU time cost does not depend on the CS ratio, thus the Table 2 reports the average value of computation time that the algorithms spend to reconstruct the image.

	TVLNR	TVAL3	ISTA-Net ⁺	CS-Net ⁺
Peppers	642	4.01	0.15	0.20
Lena	640	3.91	0.15	0.20
Flintstones	3238	25.11	0.546	0.80
Fingerprint	4415	27.18	0.977	0.80

Table 2: Average values of CPU time cost for the reconstruction of Peppers, Lena, Flintstones and Fingerprint.

The graphs below represent the evolution of the MSE, PSNR, SSIM and the number of iterations in function of the CS ratio for each image.

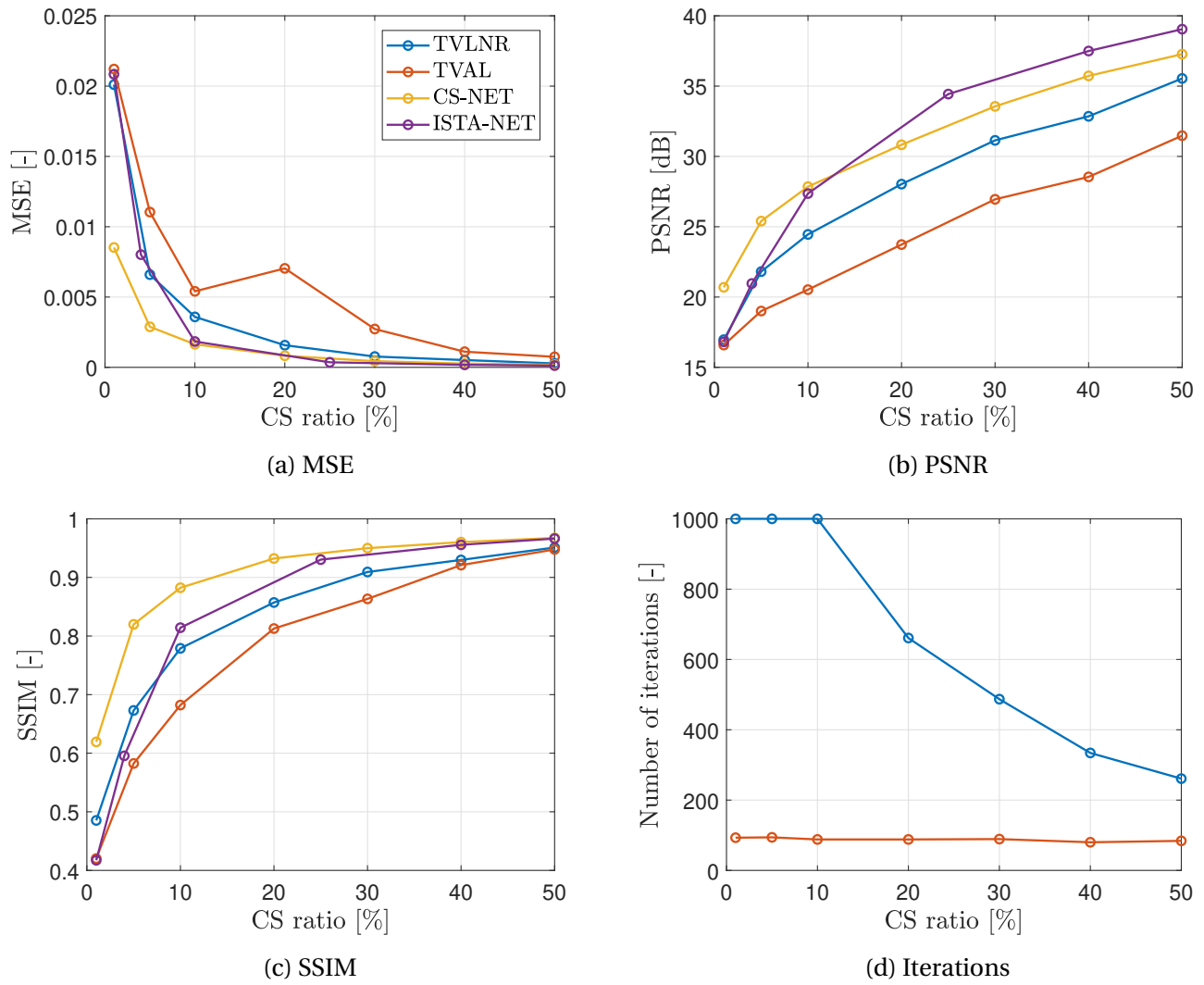


Figure 18: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Peppers".

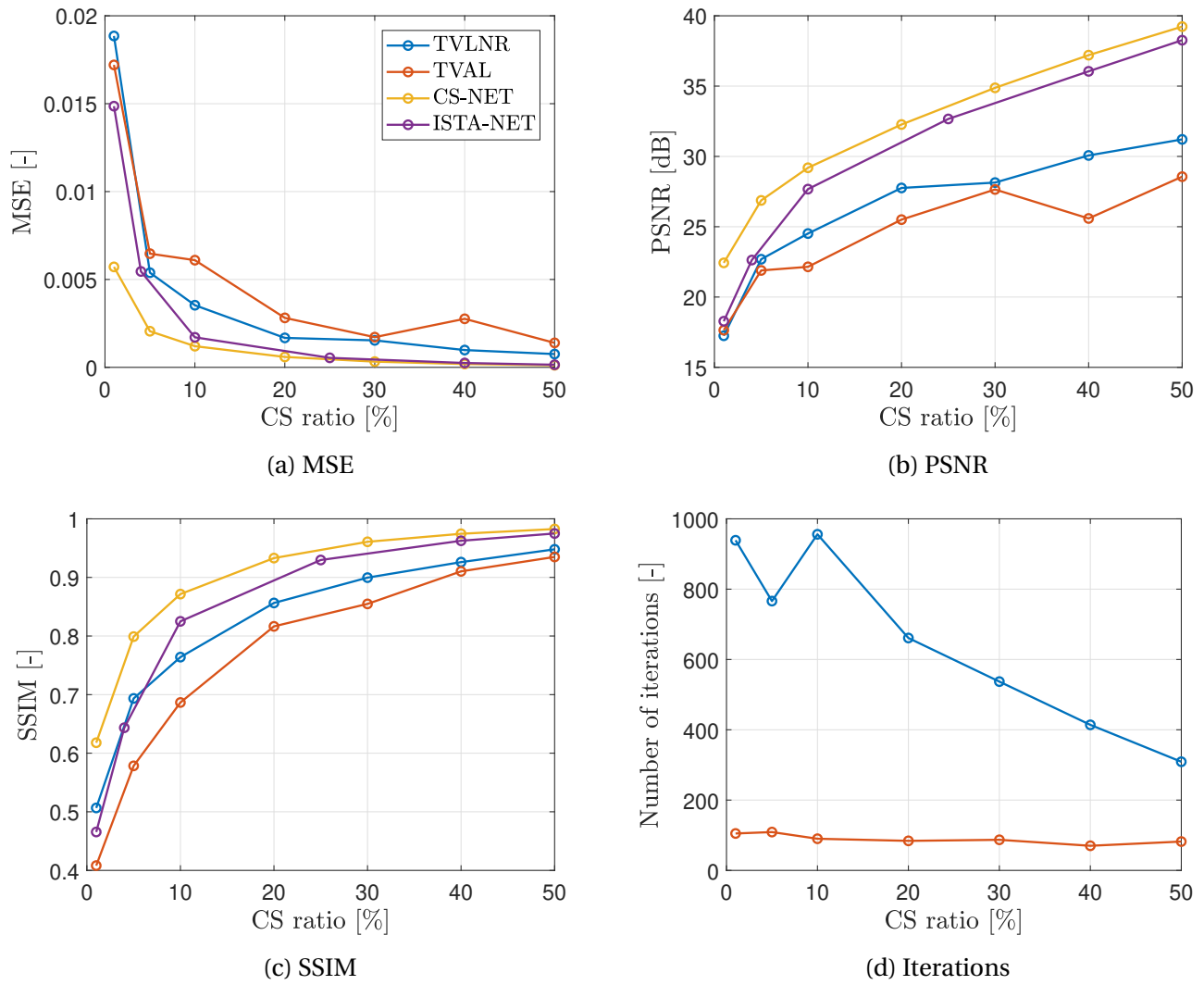


Figure 19: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Lena".

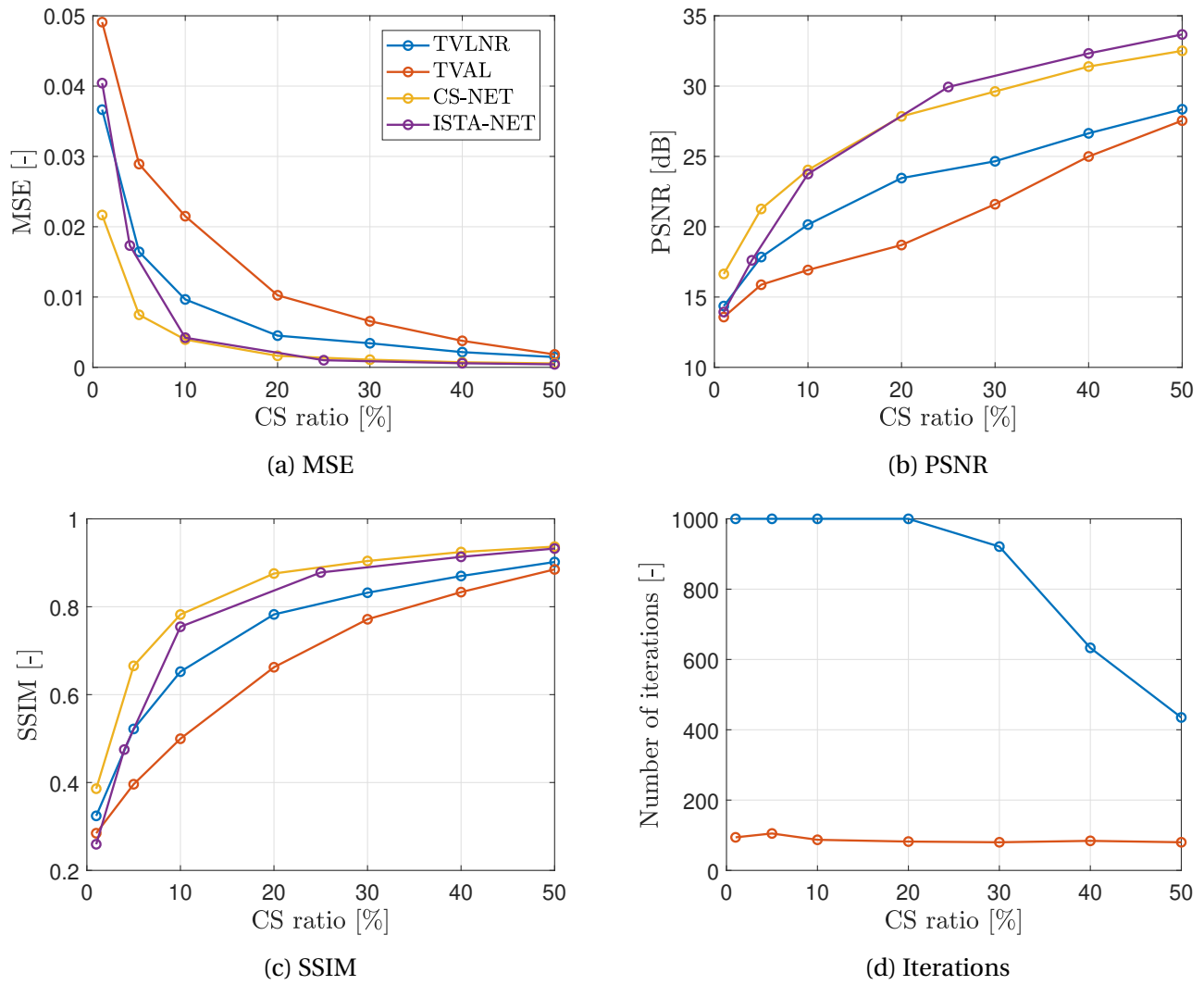


Figure 20: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Flintstones".

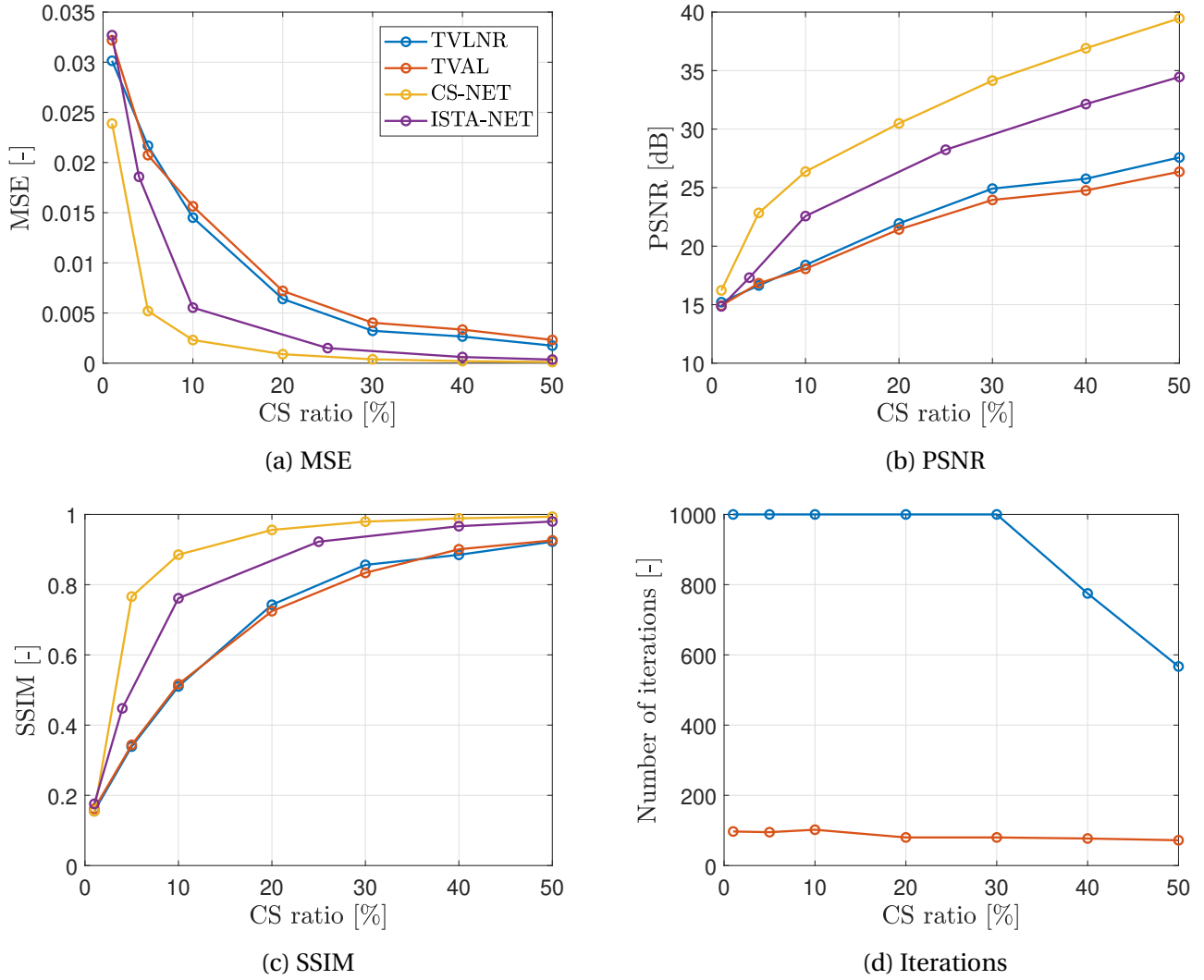


Figure 21: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the different reconstruction algorithms, in function of the CS ratio for the image "Fingerprint".

The tables in annex A and Figures 18-21 above show that the values of MSE and PSNR are inversely proportional, the MSE decreases and the PSNR increases with the increase of the CS ratio. The SSIM has the same behavior of the PSNR. This evolution is due to the fact that, when the CS ratio increases, the number of measurements is higher and so the information about the initial image is greater. Moreover, good results are already obtained when the CS ratio is about 30% for all the algorithms.

Comparing the two TV algorithms, TVLNR and TVAL3, the TVLNR gives good results for the PSNR and the SSIM, but the CPU time cost is too high to make it suitable. However, the best results are obtained with the DL algorithms, as it was expected.

Between the two DL algorithms, CS-Net⁺ is the best in all parameters except the CPU time cost that is slightly higher than the ISTA-Net⁺, although still tiny. Thus, both algorithms are powerful and suitable, the choice will be between the quality and the computation time.

The evolution of PSNR and SSIM are not always similar for the same image. Taking in consideration Figures 18b and 18c, and Figures 20b and 20c, the PSNR is higher for ISTA-Net⁺ with respect to CS-Net⁺, however the SSIM is higher for CS-Net⁺ with respect to ISTA-Net⁺. This difference is caused by the fact that for the PSNR there is a comparison pixel-by-pixel, and it focuses on colors. Alternatively, the SSIM focuses on the global image, and the best image is the one with the SSIM higher for the human eye.

Table 3 at the end of the chapter reports the global comparison between the algorithms, and establish as the one with the best performance.

Once the parameters are studied, the SSIM map can be used to observe the parts of the images that are complicated to reconstruct. The SSIM map makes, thus, a comparison between the algorithm based on the value of SSIM. In Figure 22, the SSIM maps are reported for all the algorithms when the CS ratio is 20%, and 25% for the ISTA-Net⁺.

It is chosen a low CS ratio because, if the values of SSIM are too high, the image is mostly composed by white pixels and the critical parts are not identifiable. Moreover, when the ratio is 20% the difference between the values of SSIM for each algorithm is more marked.

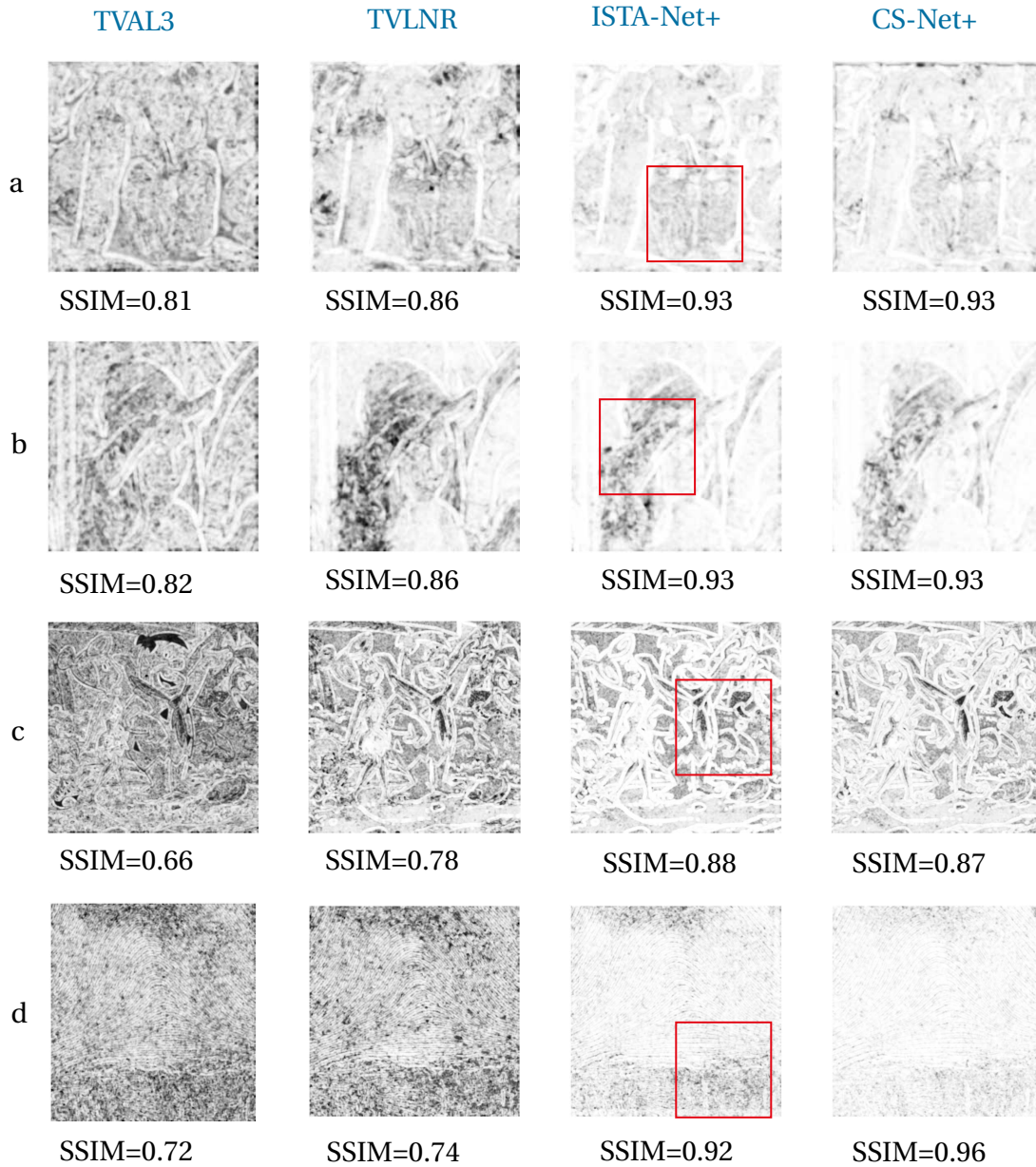


Figure 22: SSIM maps of the TVAL3, TVLNR, ISTA-Net⁺ and CS-Net⁺ when the CS ratio is 25% for ISTA-Net⁺ and 20% for the other algorithms. The red square highlights the most complicated part to reconstruct. The images are "Peppers" (a), "Lena" (b), "Flintstones" (c) and "Fingerprint" (d).

Figure 22 shows clearly that the DL methods have a better reconstruction with respect to the TV methods. The DL images have more white pixels, therefore in these regions the reconstruction is very accurate. The worst algorithm is clearly the TVAL3, the image is composed mostly by black pixels. The difference between the two DL algorithms is not really clear in Figure 22 because both have a large number of white pixels. However, one must consider that the ISTA-Net⁺ has

5% more of measurements with respect to CS-Net⁺.

The red squares in Figure 22 highlight the critical part of the reconstruction. The TV algorithms also have other critical parts, but the square highlights the critical part that is almost common to all the algorithms.

The corresponding reconstructed images are shown in the Figure 23, particularly the critical part for each image is magnified to see clearly the difference with the original image.



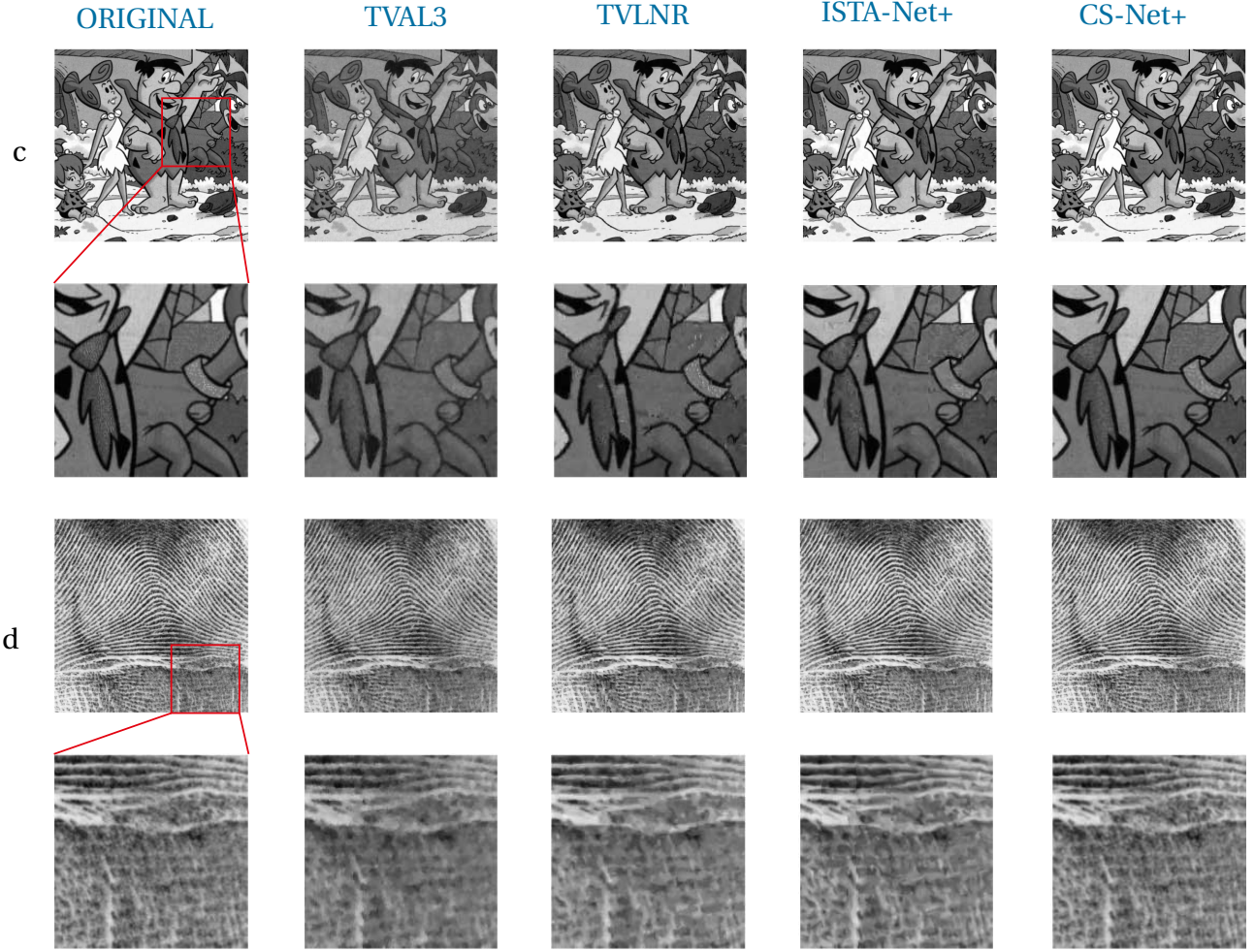


Figure 23: Original image and images reconstructed by the TVAL3, TVLNR, ISTA-Net⁺ and CS-Net⁺. The CS ratio is 25% for ISTA-Net⁺ and 30% for the other algorithms. For each image, "Peppers" (a), "Lena" (b), "Flintstones" (c) and "Fingerprint" (d), the red square highlights the most complicated part to reconstruct.

Figure 23 (a, b, c, d) reports the reconstructed images of "Peppers", "Lena", "Flintstones", and "Fingerprint" respectively, when the CS ratio is 30%, and 25% for the ISTA-Net⁺. The images show that the reconstruction is not yet perfect, especially for TV algorithms. The magnification of the critical part shows that also the DL algorithm have some difficulties in front of critical part. The image "Flintstones" shows that the white dots present in the original images are not well reconstructed. For this detail, the CS-Net⁺ has a better reconstruction with respect to the ISTA-Net⁺, even if the PSNR is larger for the latter. This feature explains the difference between the PSNR and the SSIM explained before.

3.3.6 Total variation with noise

As explained in sections 3.3.1 and 3.3.2, the simulations for TVLNR and TVAL3 are also computed adding some noise to the measurements. The percentage of noise varies from 1% to 10% while the value of the CS ratio is maintained fixed at 40%.

The values for all figures are reported in Table B in appendices. However, two images, "Peppers" and "Flintstones", are chosen to illustrate and discuss the behavior of the algorithms in the presence of noise. The evolution of values of the parameters of comparisons are reported in Figures 24 and 25.

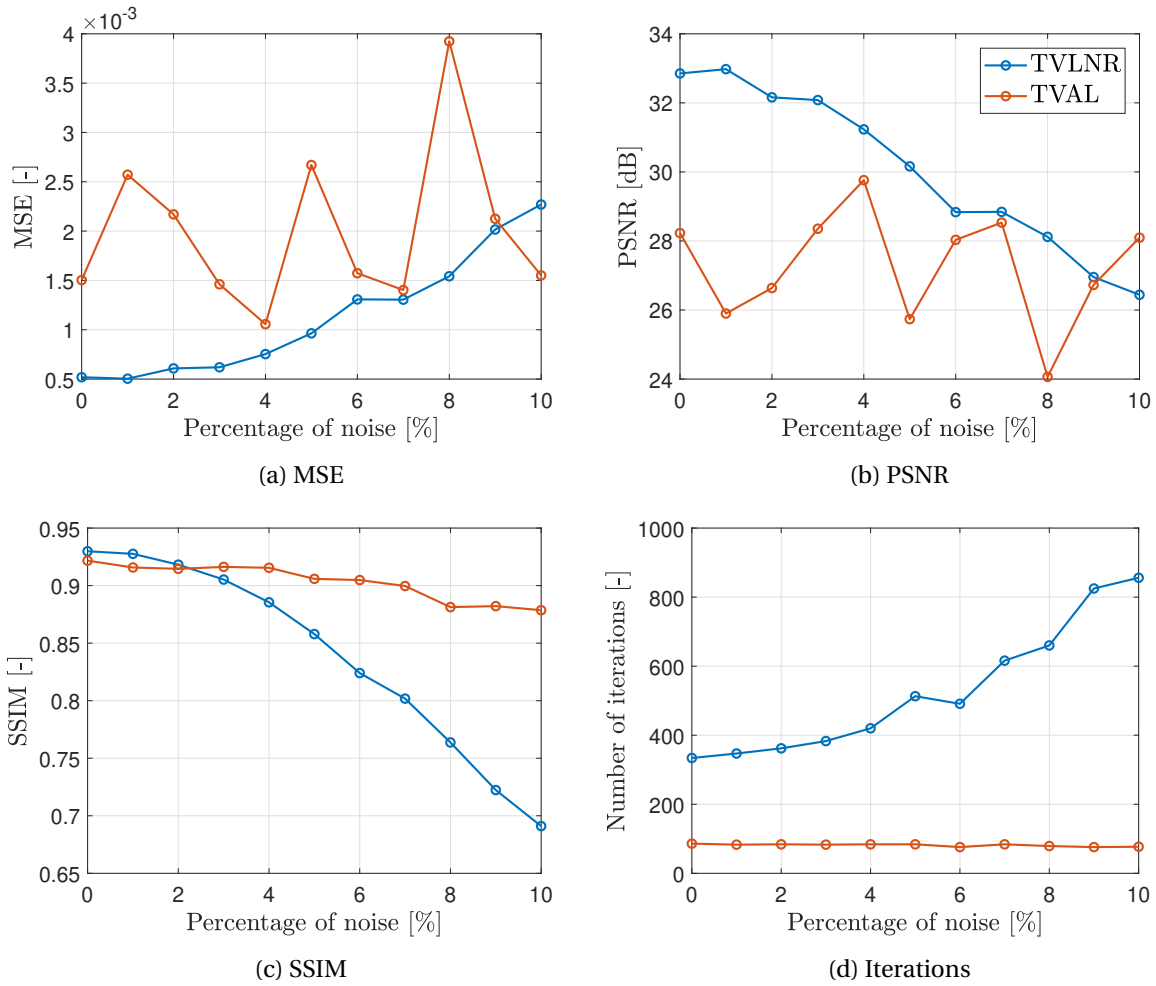


Figure 24: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the two TV reconstruction algorithms, in function of the percentage of noise for the image "Peppers".

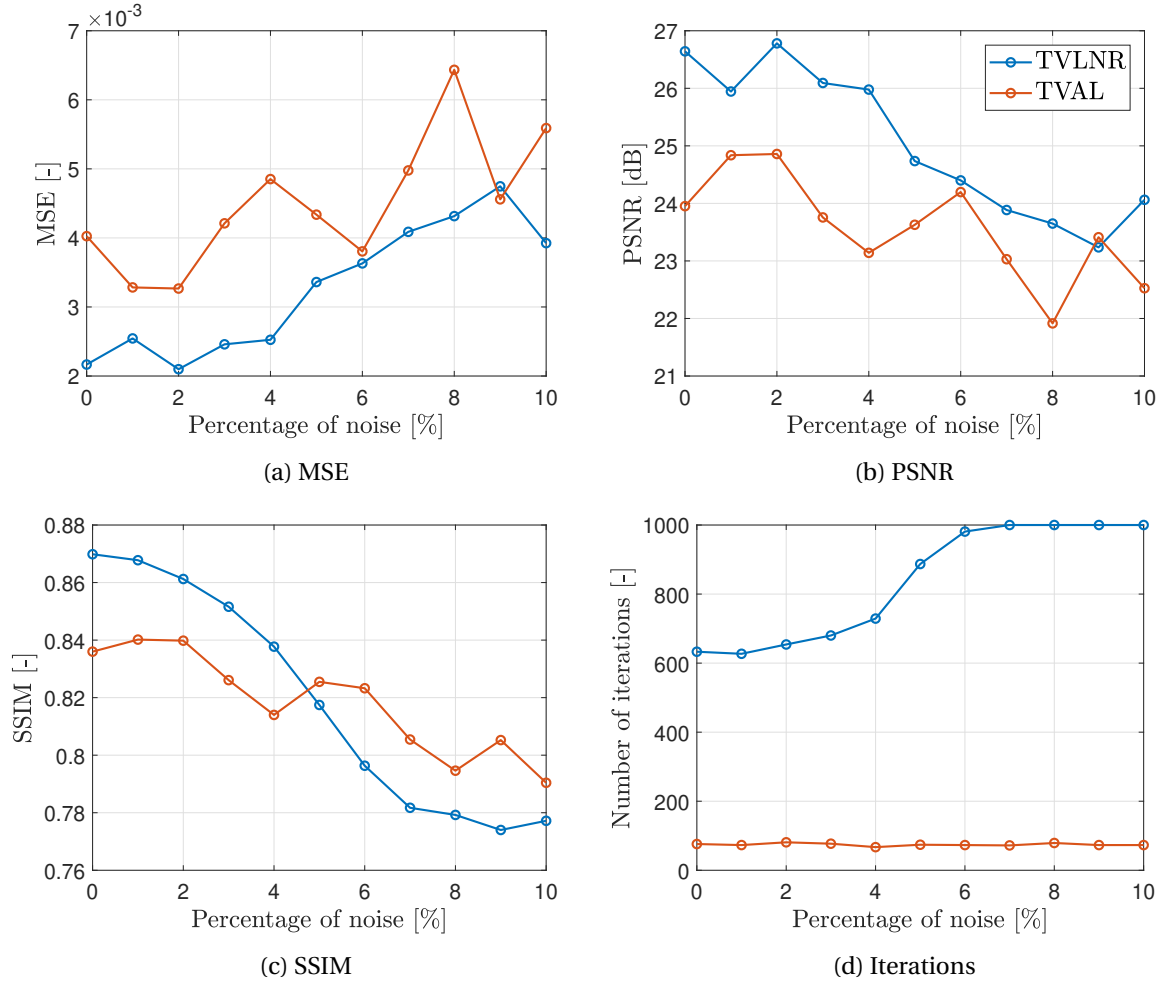


Figure 25: Evolution of the MSE, PSNR, SSIM and the number of iterations, computed using the two TV reconstruction algorithms, in function of the percentage of noise for the image "Fingerprint".

The graphs show that the quality is lost when the noise increases. The decrease of quality of PSNR and MSE is not linear as one might expect. This lack of linearity is caused by the fact that the PSNR and the MSE compare the images for each pixel and not in their entirety. The linearity is clearly present in the evolution of SSIM, where the images are compared in their entirety. The number of iterations increases almost linearly for the TVLNR, but in the case of TVAL3 remains nearly constant when the amount of noise increases.

Globally, the TVAL3 shows better results in terms of quality, but the algorithm is not stable, the evolution is not linear when the noise increases. Therefore, to have better results, it is necessary to repeat the simulations several times and retrieve average results.

3.3.7 Final comparison

All previous discussions on the evolution of the quality parameters for each algorithm can be used to designate the best algorithm suitable for the CS space application. The four algorithms, TVLNR, TVAL3, ISTA-Net⁺ and CS-Net⁺, are compared, and the performance scores are attributed to them for each of parameter of quality. These performance scores are quantified by stars (★): the greater is the number of star for each parameter, the better is its performance. Table 3 reports the comparisons of performance of the algorithms.

Parameters \ Algorithms	TVLNR	TVAL3	ISTA-Net ⁺	CS-Net ⁺
MSE	★★	★	★★★★★	★★★★★
PSNR	★★	★	★★★★★	★★★★★
SSIM	★★	★	★★★★	★★★★★
N° of iterations	★	★★★★	/	
CPU time cost	0	★	★★★★★	★★★★★
Noise behavior	★	★★	/	
Total performance	★★★	★★★	★★★★	★★★★★

Table 3: Qualitative summary of the performance for different algorithms.

The best quality performance are obtained by the CS-Net⁺. It could be the more suitable for the space application thanks to the great levels of SSIM and PSNR and to the low CPU time cost that allows to obtain the image almost instantaneously.

At the second place there is the ISTA-Net⁺. It has slightly inferior performances than the CS-Net⁺, the CPU time cost is better with respect to the values of CS-Net⁺, but still comparable. For the availability to anybody and ease of coding in Python with respect to MATLAB, it could be also suitable for a future space application.

Finally, the classic recovery algorithms for CS imaging, TVLNR and TVAL3, have really large CPU time cost, especially TVLNR method, for a PSNR and SSIM that are not so good, compared to DL algorithms. For this reason, they do not seem suitable for a space implementation.

Therefore, the DL algorithms make the CS imaging more efficient and realistic for a space application in a future that is not so far, as explained in section 5.

4 DiffuserCam

One of the objectives of this work is to implement the CS in an experimental case. It was a bit ambitious to envisage an imaging system for space application. Therefore, based on literature review, the DiffuserCam has been chosen as an application, that can be set up in the CSL laboratory in a realistic time frame.

The DiffuserCam is a practical example of a lensless camera using the Compressive Sensing method and which seemed easy to set up.

The structure of the DiffuserCam is characterized by a diffuser (a thin, transparent, lightly scattering material), that replaces the lens of a classical camera and a classical sensor [9]. Figure 26 represents the scheme of a DiffuserCam.

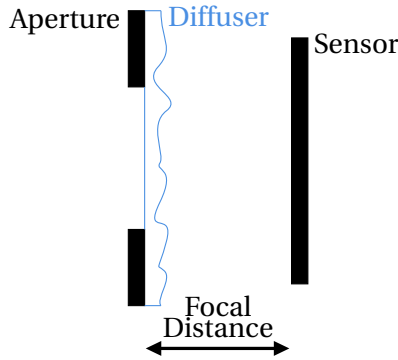


Figure 26: Scheme representing the structure of a DiffuserCam. Adapted from [9]

As explained in [9], such a diffuser has a so-called "focal distance" that represents the distance between the diffuser and the sensor. The light coming from the object passes through the diffuser where it is blocked by the aperture. The image at the sensor is not defined, but it is used in the reconstruction algorithms to retrieve the initial image.

The section 4.2 will explain how to practically retrieve the focal distance, reported in Figure 26, of a diffuser.

This kind of camera has several advantages:

- **Lightweight:** in the DiffuserCam the lenses or mirror that constitute a camera are replaced by a thin and flat material. This kind of structure makes the imaging system lighter.

- **Lower accuracy:** The structure of the diffuser is random by nature, so it does not need accuracy for the fabrication. In the next section, the results obtained using the tape like diffuser will be shown.
- **3D imaging:** The authors of the article [9] demonstrate that the principle of the DiffuserCam can be used for the reconstruction of 3D images. This concept can be useful for the microscopy application, for example.

The section 4.1 describes some reconstruction methods that will be used with the experimental set-up.

4.1 Reconstruction Principles

The imaging system is based on the CS concept. The scene \mathbf{x} is mapped into a sensor through the diffuser. The measurements by the sensor \mathbf{y} are defined as:

$$\mathbf{y} = A\mathbf{x} \quad (63)$$

where A is the measurement matrix. In the CS approach, the matrix A is non-invertible, thus the basic reconstruction of the image is given by solving the least-squares problem:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \frac{1}{2} \|A\mathbf{x} - \mathbf{y}\|_2^2. \quad (64)$$

The DiffuserCam uses two different methods to solve the minimization in the equation (64): the Gradient Descent (GD) method and the Alternating Direction Method of Multipliers (ADMM). These methods use the concept of CS algorithms of Total Variation and Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) to reconstruct the image.

4.1.1 Gradient Descent

The Gradient Descent method implements the gradient of the minimization with respect to the variable \mathbf{x} , to optimize (64). Since the gradient indicates the direction of the steepest rise, the reconstruction goes in the opposite direction:

$$\mathbf{x}_0 = \text{initial guess} \quad (65)$$

$$\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k - \alpha_k \nabla \hat{\mathbf{x}}(\mathbf{x}_k) \quad (66)$$

where $\nabla \hat{\mathbf{x}}(\mathbf{x}_k)$ is the gradient and α is a step size that determines the velocity of descent. In this problem, the gradient is defined as

$$\nabla \hat{\mathbf{x}}(\mathbf{x}) = A^H(A\mathbf{x} - \mathbf{y}) \quad (67)$$

where A^H is the adjoint of A .

The step size should be constant, but sufficient small to guarantee the convergence. Therefore, the largest α is $0 < \alpha < \frac{2}{\|A^H A\|_2}$.

Theoretically, the convergence is guaranteed after an infinite number of iterations, but in reality after a certain number the improvement is so small that it can be chosen as the maximum number of iterations.

The previous method can be speeded up using the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA). It integrates a new operator, the projection step p_L , that links two successive points of the reconstruction.

The algorithm for the reconstruction of the image is so described in following the same order by

$$\begin{aligned} \mathbf{x}_0 &= I/2, & t_1 &= 1, & v_0 &= \mathbf{x}_0 \\ \text{for } k &= 0 \text{ to } \max_{\text{iter}} : \\ & v_k \leftarrow p_L(\mathbf{x}_k) \\ & t_{k+1} \leftarrow \frac{1 + \sqrt{1 + 4t_k^2}}{2} \\ & \mathbf{x}_{k+1} \leftarrow v_k + \frac{t_k - 1}{t_{k+1}}(v_k - v_{k-1}) \end{aligned} \quad (68)$$

where \mathbf{x}_0 is the initial guess of the reconstructed image, it can have random values, but in this case it is imposed to be the half of a uniform image.

4.1.2 Alternating Direction Method of Multipliers

The GD is a reliable method that guarantee the convergence, but it is slow and limited to 2D images. To manage larger images (3D for example) and for faster reconstruction, the Alternating Direction Method of Multipliers method is preferable.

Starting from the original problem:

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \geq 0}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y} - A\mathbf{x}\| \quad (69)$$

the minimization of $\hat{\mathbf{x}}$ can be split into two separable minimizations, taking into account the action of the DiffuserCam on the measurement matrix, $A = CM$, where M represents the convolution and C the crop. The separation is defined as follows:

$$\begin{aligned} \hat{\mathbf{x}} = \underset{w \geq 0, v}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y} - Cv\|_2^2 \\ \text{s.t. } v = M\mathbf{x}, w = \mathbf{x} \end{aligned} \quad (70)$$

Using the concept of the CS, that the images are sparse, the ℓ_1 norm of the gradient is added, giving the expression of the TV methods:

$$\begin{aligned} \hat{\mathbf{x}} = \underset{w \geq 0, u, v}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y} - Cv\|_2^2 + \tau \|u\|_1 \\ \text{s.t. } v = M\mathbf{x}, u = \Psi\mathbf{x}, w = \mathbf{x} \end{aligned} \quad (71)$$

where ψ is a derivative operator.

As for the TV methods explained in sections 3.3.1 and 3.3.2, the ADMM uses the minimization of the augmented Lagrangian function as follows

$$\begin{aligned} \min_{u, v, w, \mathbf{x}} \mathcal{L}(\{u, v, w, \mathbf{x}\}, \{\xi, \eta, \rho\}) = & \frac{1}{2} \|\mathbf{y} - Cv\|_2^2 + \tau \|u\|_1 \\ & + \frac{\mu_1}{2} \|M\mathbf{x} - v\|_2^2 + \xi^\top (M\mathbf{x} - v) \\ & + \frac{\mu_2}{2} \|\Psi\mathbf{x} - u\|_2^2 + \eta^\top (\Psi\mathbf{x} - u) \\ & + \frac{\mu_3}{2} \|\mathbf{x} - w\|_2^2 + \rho^\top (\mathbf{x} - w) \\ & + \mathbb{1}_+(w), \end{aligned} \quad (72)$$

where $\mathbb{1}_+(w)$ assures that $w \geq 0$.

4.2 Construction of the DiffuserCam

The construction of the DiffuserCam requires several elements:

- **Digital sensor:** The digital sensor is obtained by a camera without the optical elements, in this case, it comes from a CMOS Thorlabs DCC3240C camera with 1280×1024 pixels. A



software allows to read in live the light coming on the sensor.

- **Diffuser:** As explained before, the diffuser is a thin, transparent, scattering material. In the DiffuserCam, the diffuser consists of a simple transparent tape.
- **Optical diaphragm:** The optical diaphragm, or aperture stop, stops the incoming light and allows to pass only a small part. In this case, the aperture of the diaphragm is smaller than the dimensions of the sensor, and the diffuser is attached to it.
- **Point source:** A LED of incoherent light is used as point source to obtain the Point Spread Function (PSF).
- **Object:** An image of a letter on a self illuminated screen is used as an object in front of the sensor.
- **Darkness:** The environment of the calibration and the measurements has to be dark to have better results. Only the point source or the object are sources of light.

Figure 27 shows the components of the DiffuserCam separately and the final construction of the DiffuserCam.

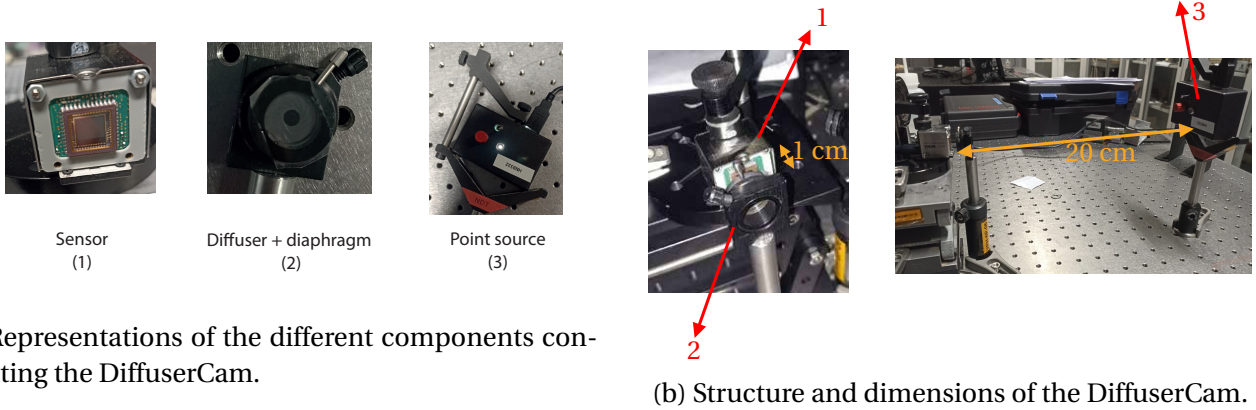


Figure 27: representation of components and assembly of the DiffuserCam.

The first important task in the construction of the DiffuserCam is to find the focal distance of the diffuser. In order to find the focal distance, the point source is placed far away from the sensor, around 20 cm, and the diffuser moves closer or further from the sensor to have a good PSF. The PSF can be considered good when the sensor captures a clear caustic pattern. In this set-up, the focal distance is found to be around 1 cm.

Once that the focal distance is found, the distance between the sensor and the diffuser is fixed. The

diaphragm is closed to find a correct aperture in front of the sensor, the diameter of the aperture is about 5 mm.

Figure 28 shows a schematic representation of the measurement of the PSF and the recording of images.

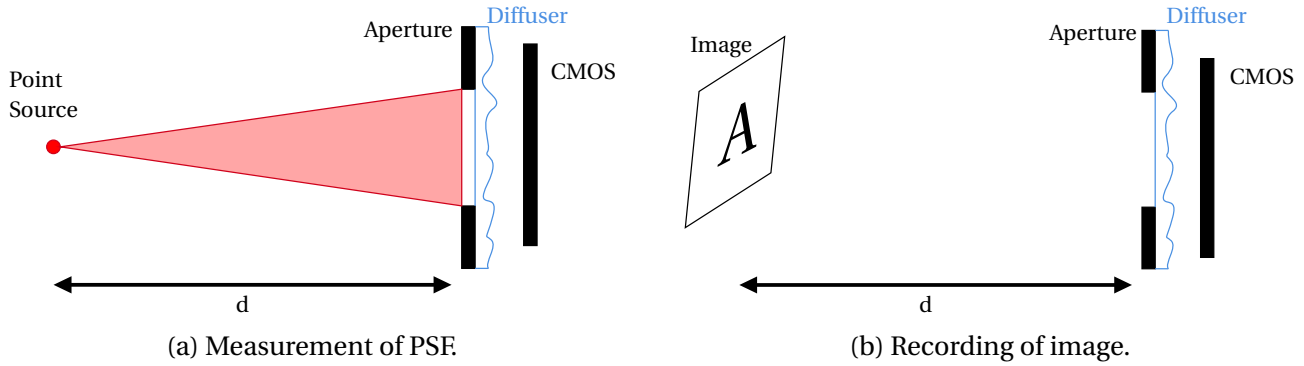


Figure 28: Schematic representation of the phases of the image capture.

When all the distances and dimensions are fixed, the calibration (measurement of PSF) can be performed. There are some precautions to take in the capture of the PSF: the point source has to be sufficiently bright and the smallest possible. The brightness at the sensor has to be high enough without saturating the sensor. The saturation can be controlled by changing the exposure time of the camera. Finally, the exposure time used to capture the PSF is 3 ms. Figure 29 shows the PSF at the sensor.

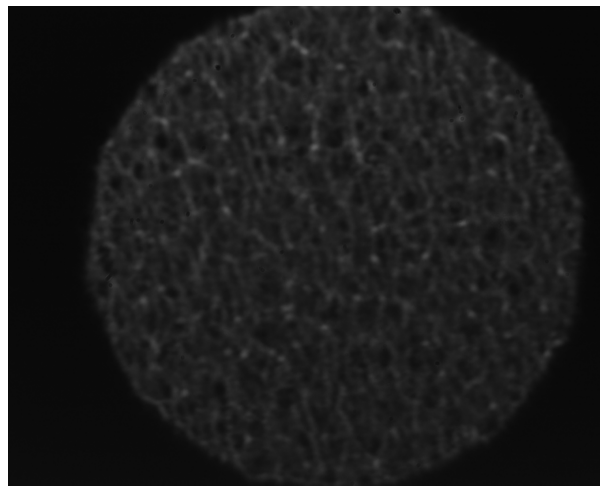


Figure 29: PSF of the DiffuserCam, the light source is 30 cm away and the exposure time is 3 ms.

The images are taken using a self-illuminated screen at the same distance of the light source.

The only light, that arrives at the sensor, is the light coming from the object. The exposure time is increased in order to collect more light at the sensor, and always without saturating the sensor. Several values of the exposure time are tested to find the best value: 21 ms. Figure 30 shows the raw images captured at the sensor.



Figure 30: Raw image of the letter C at the sensor with an exposure time of 21 ms.

At the end, the PSF and the raw image are inserted in the GD and ADMM algorithms to reconstruct the original images. The next section reports the results of the DiffuserCam.

4.3 Results and discussion

Several factors are investigated during the simulation. The first important factor is the number of iterations for each method. It has been noticed that for the GD method, it is necessary to have numerous iterations, about 140, with an intermediate reconstruction each 20 iterations, to have a good reconstruction.

For the ADMM method, already at 20 iterations with an intermediate reconstruction each 4 images, the reconstruction reaches already good results. Increasing the number of iterations does not increase significantly the quality. Therefore, it has been fixed at 20.

The other important factor is the downsampling factor imposed to be $f = 1/2^k$, where k is a positive integer. It is noticed that when f decreases, the quality of the image and the CPU time cost decreases at the same time.

The images, for $f = 0.5, 0.25, 0.125$ and 0.0625 , are compared using PSNR and SSIM methods of comparison, but also the shape of the reconstructed images with the function *edge* that com-

putes the edge of the images. Figure 31 shows the reconstructed images and the edges for each reconstruction.

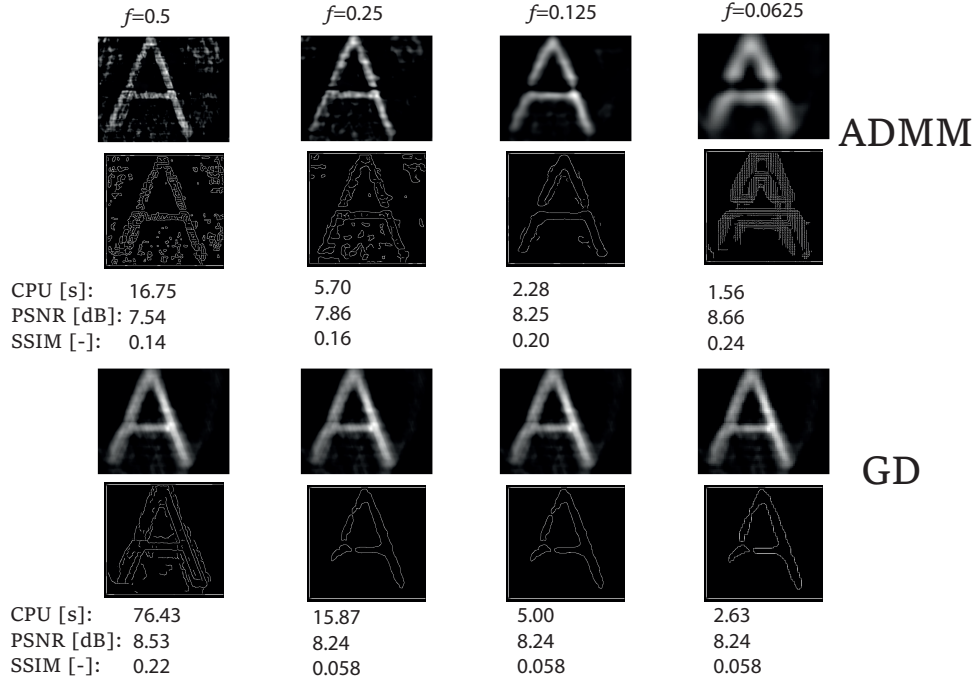


Figure 31: ADMM and GD reconstruction of the letter A for the downsampling factor f that varies. The values of CPU time cost, PSNR and SSIM and the edge of each reconstruction are reported too.

Figure 31 shows that, when the factor f is 0.5, the letter A is completely reconstructed, but the CPU time cost is too high. When the value of f decreases, the reconstruction is not complete, but the CPU time cost decreases hugely. Finally, for lower value $f = 0.0625$, the blur of the image increases.

The evolution of the PSNR and SSIM, contrarily, increases when f decreases. This behavior can be explained because the PSNR evaluates the reconstruction pixel-by-pixel and the difference between the white and black pixels increases in the last images. The SSIM takes into account the contrast, the brightness and the structure of the reconstructed image, and they are more obvious when the f is small. Therefore, the PSNR and SSIM are not useful to evaluate the reconstruction for the DiffuserCam because they do not take into account the blur of the images and the reconstruction of the edges.

A trade-off between the quality and the CPU time cost has to be done. For the ADMM algorithm, since the CPU time cost is relatively low, the downsampling factor of $f = 0.25$ is chosen because the image is less blurred compared to the $f = 0.125$.

For the GD method, the difference in quality when the $f = 0.25$ or 0.125 is not really obvious. When the $f = 0.0625$ the image is blurred, thus a final value of $f = 0.125$ is chosen for GD method.

One of the most important parameters to study is the exposure time, that represents the period of time during which the sensor recollects the light. For the PSF measurements, the exposure time has to be small enough to not saturate the sensor. For the images, where the luminosity of the screen is weaker, the exposure time has to be increased. In order to find the best value for the exposure time, several values for different images are tested, and the results are reported in Figure 33.

CSL

Figure 32: Original images disposed on a self-illuminated screen used as object of the DiffuserCam.

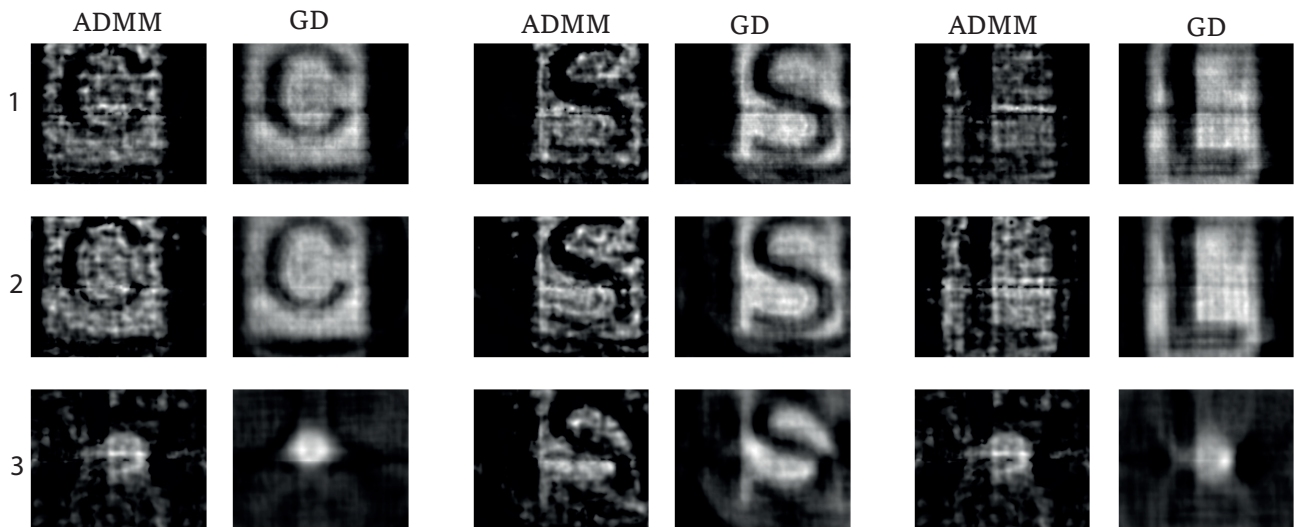


Figure 33: ADMM and GD reconstructions of letters C, S, L when the exposure time varies at 3 ms (1), 21 ms (2) and 40 ms (3).

Figure 33 shows how the reconstruction change in function of the exposure time of the sensor. When the exposure time is 3 ms, the 1st line in the figure, the reconstruction is well-defined, but the contrast between the white and black part is not quite remarked. The white part, as well as the black part, is not uniform, there are some pixels of the other color.

The best reconstruction is clearly visible when the exposure time is 21 ms in the 2nd line of the figure. The contrast between the white and the black is evident, and the white and black parts are more uniform.

The worst reconstruction appears when the exposure time is 40 ms in the 3rd line of the figure. The reconstruction is hardly visible for the ADMM method and completely inaccurate for the GD method, except for the letter S. This issue is caused by the fact that when the exposure time is too high, the sensor is rapidly saturated. The saturated part of the sensor is remarkable by the luminous spot at the center of the images, where the reconstruction is completely wrong.

The evaluation of the PSNR and SSIM is not possible this time because the dimensions of the original image and the reconstructed one are not the same, and the lateral position of the object in front of the sensor is not constant.

The number of iteration for the ADMM method has been increased to obtain optimal results, therefore the CPU time cost increases as well. Finally, the average CPU time cost for the ADMM and GD methods is respectively 15.19 s and 3.65 s.

The last parameter to study is the distance between the object and the sensor. The distance of the object (the letter C) is chosen at 5 cm, 10 cm, 20 cm (the same of the PSF) and 30 cm. Figure 34 shows the reconstructions when the distance from the sensor increases.

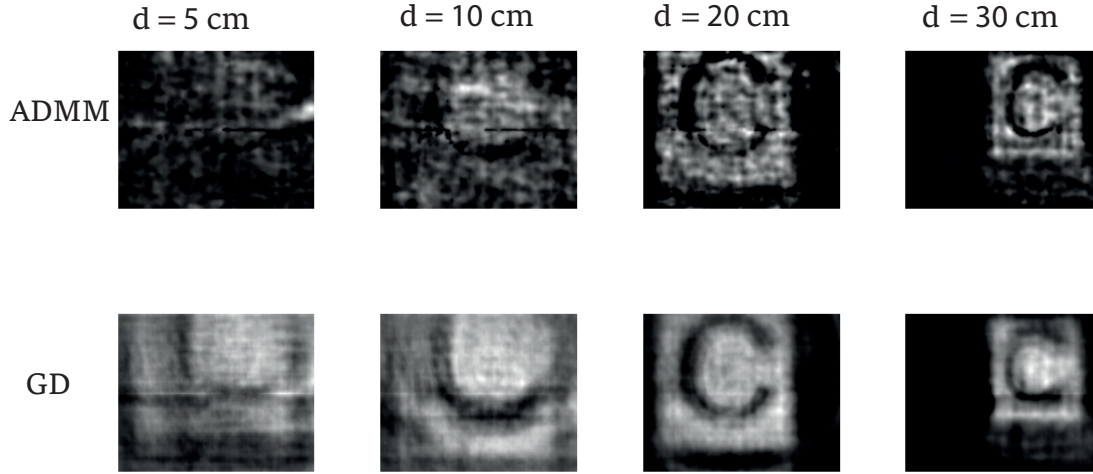


Figure 34: ADMM and GD reconstructions of letter C when the distance between the object and the sensor is 5 cm, 10 cm, 20 cm, and 30 cm.

Figure 34 shows that the best reconstruction is achieved when the distance between the object and the sensor is the same at which the PSF is taken, i.e. 20 cm. The GD method seems to reconstruct better the image with respect to the ADMM when the distance changes. As was to be expected, the field of view increases with the distance, such that at a smaller distance only a part of the light reaches the sensor and the letter is partially reconstructed.

Therefore, to assure a good functioning of the DiffuserCam, the values of the downsampling factor f , the exposure time and the distance between the object and the sensor have to be accurately chosen.

Some modification can be brought to the structure of the DiffuserCam. For example, the tape can be replaced by another diffuser. Some tests are done with a thicker diffuser and a SLM LC2012 used as phase plate.

The thicker diffuser creates a PSF that is too fine and not clear. To have a caustic pattern well resolved, similarly to the one generated by the tape, the sensor should have smaller pixels and the light has to be brighter. Figure 35 shows the diffuser and the pattern generated on the sensor.

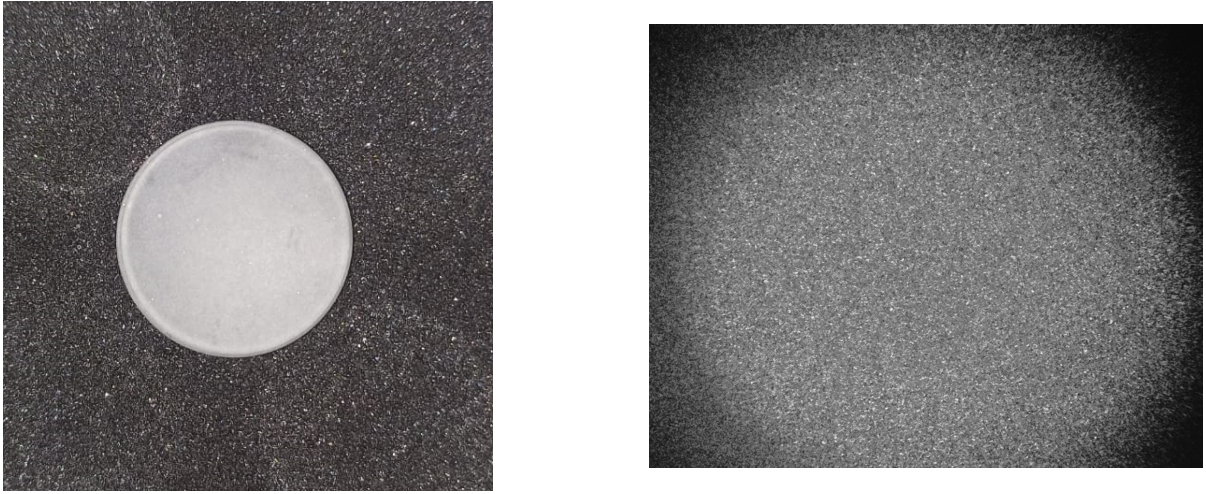


Figure 35: Picture of the thicker diffuser and the corresponding PSF captured by the sensor.

The SLM can generate a random binary phase plate like in the case of the diffuser. However, in practice, it is not useful for a DiffuserCam because the PSF is small and quite regular. In the frame of this work, we were not able to create a truly random pattern that is convenient for the DiffuserCam.

The solution can be to have an SLM that creates a phase plate where one can choose the value for each pixel. In the case of the SLM LC2012, the pattern generator generates itself a random pattern that cannot be controlled.

Figure 36 reports the SLM and the corresponding PSF.

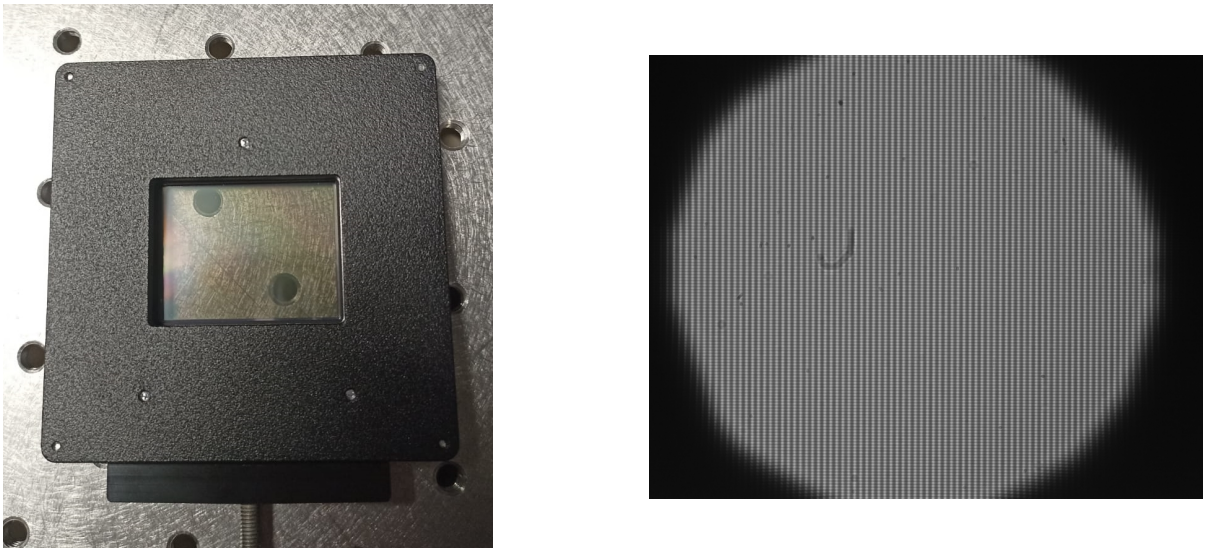


Figure 36: Picture of the SLM and the corresponding PSF captured by the sensor.

5 Space application

Nowadays, the study of Compressive Sensing technique is not limited to the computer simulations or laboratory applications like the DiffuserCam, the single pixel camera and the NoRDS-CAIC described in the previous sections.

The current studies aim to use the CS in the space domain. The request for the high resolution images determine an increase of the amount of data to manage in terms of memory and data link. CS would be able to directly acquire compressed signals, and extract the information directly on-board the satellite. In this way, a preselection of the reconstructions can be performed, in view of decreasing the data to be transferred.

Therefore, CS based instruments can save mass, power and memory because the compression board is no more necessary, and the compressed data occupy less memory.

The concept at the basis of CS instruments is the same reported in section 2.3. Unfortunately, building CS instruments is not so easy in view of space application. The SLM and the detector, that are in the earth of the instruments, are the most critical parts.

There are different kinds of SLM available: the Liquid Crystal Plate (LCP), the Micro-Shutter Arrays (MSA) and the Digital Micromirrors Arrays (DMD) [27]. The LCP induces phases delays of the incoming light for each measurement of the CS instrument. The MSA is capable of creating programmable amplitude masks that changes for each measurement. The working principle of the DMD is the same explained for the single pixel camera in section 2.

The MSA and the DMD, particularly the Micro-Mirror Array (MMA), have been tested for the space application [28]. The characteristics for both SLM are reported in table 4.

	MSA	MMA
Working principle	transmission	reflection
Switching speed	few frame per second	32 kHz
N of elements	171×365	2048×1080
Size of elements μm	100×200	[5 14]
Cryogenic temperature	yes	yes
Fill factor	80%	92%
Radiation test	yes	yes

Table 4: Characterization of the different SLM adapted for the space application [27].

The MMA appears to be the most adapted for the CS space application because of its higher switching speed and lower pixel dimension.

The other basic element of the CS instrument is the detector. It is less critical than the SLM, however a high speed of acquisition is required to reduce the time of measurement. Usually a single photodetector is the most adapted and the features depend on the spectral region of observation. Furthermore, it is possible to use an array of detectors to analyze a larger spectral band and/or reduce the computation time for the reconstruction.

The possible space applications for CS instruments has been studied by an Italian consortium in the frame of an ESA project [29]. The results emphasize three application fields: space science, planetary exploration and Earth observation. For the space science, that includes the solar observation and the stellar spectroscopy. The spectral region could be very wide, from UV to far IR. The design of CS instruments has some advantages, such as the mass, data and power reductions, and it requires a large integration time.

About the planetary exploration, the bandwidth limitation fills well with the selective bandwidth of the CS instruments. Moreover, in the UV and TIR spectral range, the cost is reduced because of the limited number of detectors.

Concerning the Earth observation, the CS instruments are very interesting for the spectral ranges of the IR region because high resolution FPA are difficult to find or are expensive. Although, some critical problems arise from the fast integration time due to high velocity of the satellite with respect to the planet. In fact, considering the pushbroom architecture, for example, the instrument should be able to collect all the information to reconstruct the linear Field Of View (FOV) in the dwell time (time during which the sensor is on the same point). This concept will be analyzed for the description of the MUSICA instrument below.

The next sections report a description of some CS instruments that could have a future application in space domain.

5.1 Multiband Ultrawide SpectroImager for Cryosphere Analysis

The Multiband Ultrawide SpectroImager for Cryosphere Analysis (MUSICA) is conceived by the Italian Space Agency (ASI) to study the cryosphere [30]. The telescope is designed to operate



in the visible spectral range (4 bands at 450-950 nm) and in thermal IR (3 bands at 8-12 nm). The CS technique is applied only in TIR region, with a set of 32 MMA grouped on a single detector pixel [31].

The telescope operates in pushbroom mode, moving across track and observing a linear field. For each element, the number M of measurements, that corresponds to the M rows of the measuring matrix A , are performed. After the acquisition, the reconstruction can be achieved. The authors of [31] use the TVAL algorithm, although it has demonstrated that the DL algorithms are faster and reconstruct with better quality. Moreover, the simulations have been demonstrated that the classical pushbroom mode reaches better quality.

This means that further studies are needed to increase the quality, but also to reduce the dwell time because the last mask observe a scene that is completely different from the first one. The best is that the DMD change its pattern in the shortest possible time.

The reconstruction algorithms have to be improved also, because the TVAL is not the best solution to have fast and good resolution at the same time.

Table 5 and Figure 37 report the telescope parameters and the CS optical design respectively.

Instrument parameters	
Acquisition mode	Pushbroom
Spectral range [μm]	VIS-NIR (0.45-0.95) & TIR (8-12)
Altitude [km]	500
Focal length [mm]	400

Table 5: Principal parameters of the MU-SICA instrument.

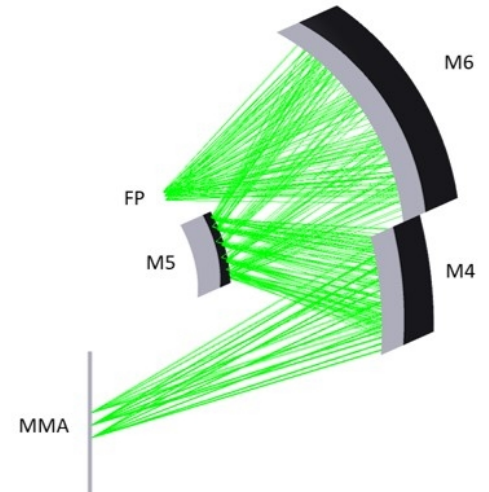


Figure 37: CS optical design of the MU-SICA instrument [30].

5.2 Super-Resolved comPReSSive InStrument for Earth observation

The Super-Resolved comPReSSive InStrument for Earth observation applications (SURPRISE) is designed to use the principle of super resolution, where the DMD is used to increase the number

of pixels of the reconstructed images, as explained in section 2.3. The instrument observes the Earth in the visible and medium infrared [28].

Contrary to the MUSICA, the SURPRISE is a whiskbroom spectral imager that operates in two different spectral bands: VIS-NIR and MIR with respectively ten and two channels. One measurement is achieved for each coding mask displayed on the SLM, the set of measurements is used to reconstruct an image that has the same dimensions of the SLM. The working principle of the SURPRISE is reported in Figure 38.

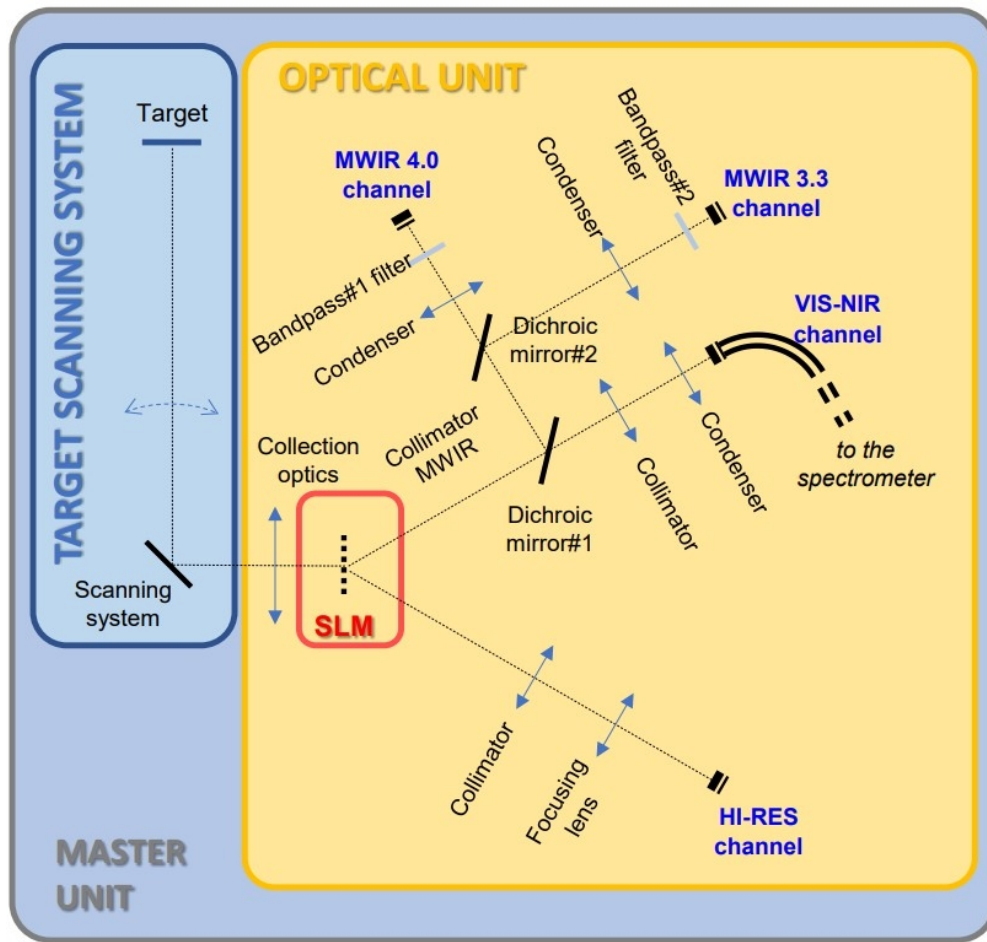


Figure 38: Working principle of the SURPRISE instrument. From [28].

The scanning system, in the left part of the Figure 38, scans the scene along two axes. Then the fore-optics reports the target scene on the field stop where the DMD is placed. After that, the dichroic mirrors operate a spectral splitting and then the signal passes through the condensers. Finally, the signal is spectrally filtered by the spectrometers to reach the appropriate spectral band

of the detector.

The second line that parts from the SLM represents the high-resolution camera used for alignment and validation objectives.

Currently, the SURPRISE is in the process of assembly with the purpose to reach a 4-fold super-resolution factor and the compressive ratio of 50%.

Table 6, reports the principal parameters of the SURPRISE instrument.

Instrument parameters	
Acquisition mode	Whiskbroom
Spectral range [μm]	VIS-NIR(0.4-0.9) & MIR (3.1-4)
Altitude [km]	36000

Table 6: Principal parameters of the SURPRISE instrument.

5.3 Super resolved imaging spectrometer in the medium Infrared

The super resolved imaging spectrometer in the medium Infrared (SISSI) project aims to observe the Earth in the MIR region to characterize high temperature events, which requires high spatial resolution [21]. The working principle of the instrument is based on the concept of super resolution, see section 2.3, and CS in order to obtain high resolution while decreasing the mass, power and memory budgets.

The architecture of the instrument is composed by the collection optics, that observe the scene, the SLM, that implements the coding mask, the condenser, that collects the light coming from each macropixel of the SLM to send it to each pixel of the detector, and finally the detector. On the detector, some spectral filters collect the same scene in different spectral bands. Therefore, when the platform moves, the scene passes each spectral band on the detector, and it is reconstructed in each band. Figure 39 reports the elements of the architecture of the instrument [21].

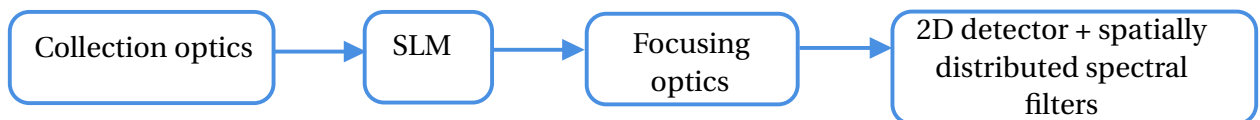


Figure 39: Elements that constitute the SISSI instrument [21].

Figure 41 shows the acquisition mode of the instrument that is a spectrometer using the spectrally resolved pushbroom acquisition mode in different spectral bands to acquire the images in both along and cross-track.

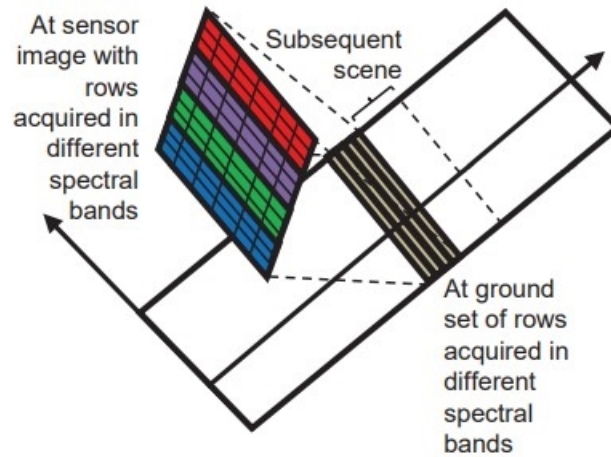


Figure 40: Representation of acquisition mode of the SISSI instrument [21].

The article [21] reports an initial optical design of the instrument. For the collection and focusing optics, reflective optical surfaces are chosen to minimize the chromatic aberration and optimizing the transmittance. The optical design is reported in Figure 41.

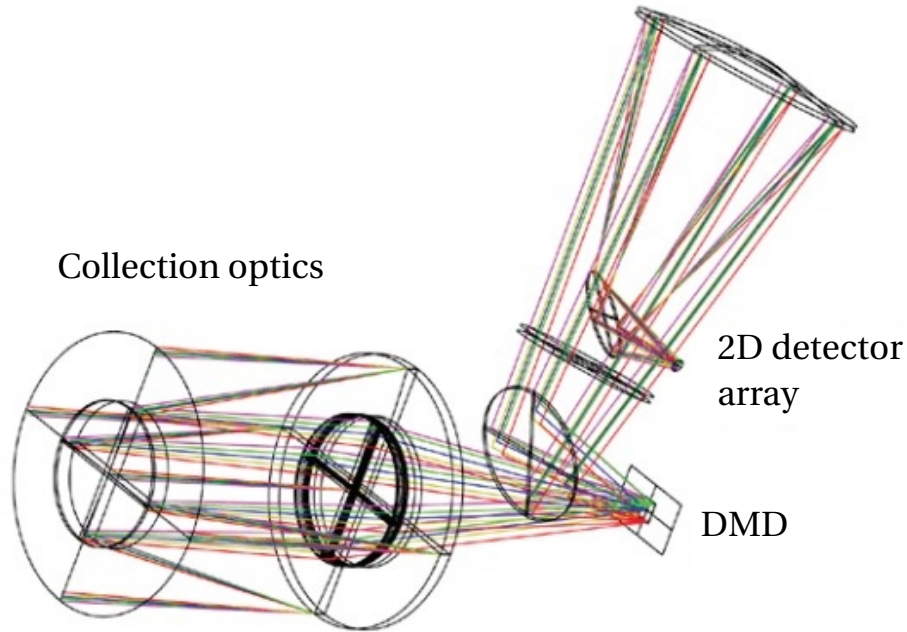


Figure 41: Optical design of the SISSI instrument [21].

In this first version of the instrument, the energy per detector pixel is about 85% and the transmittance is about 30%, without considering the DMD.

Table 7, reports the principal parameters of the SURPRISE instrument.

Telescope parameters	
Acquisition mode	Pushbroom
Spectral range [μm]	MWIR (3-4)
Altitude [km]	700

Table 7: Principal parameters of the SISSI instrument [21].

5.4 UV-VIS Hyperspectral imager on orbiter for stellar spectrophotometry

The UV-VIS hyperspectral imager on orbiter for stellar spectrophotometry is conceived for space science observation [32]. The instrument is a spectrometer that has some features similar to the Hubble Space Telescope, working from a geostationary or sun-synchronous orbit.

Since the acquisition domain for this kind of purpose is highly sparse, the CS application could be very powerful for the downlink bandwidth, mass and power reduction.

The instrument is composed by several elements reported in Figure 42.

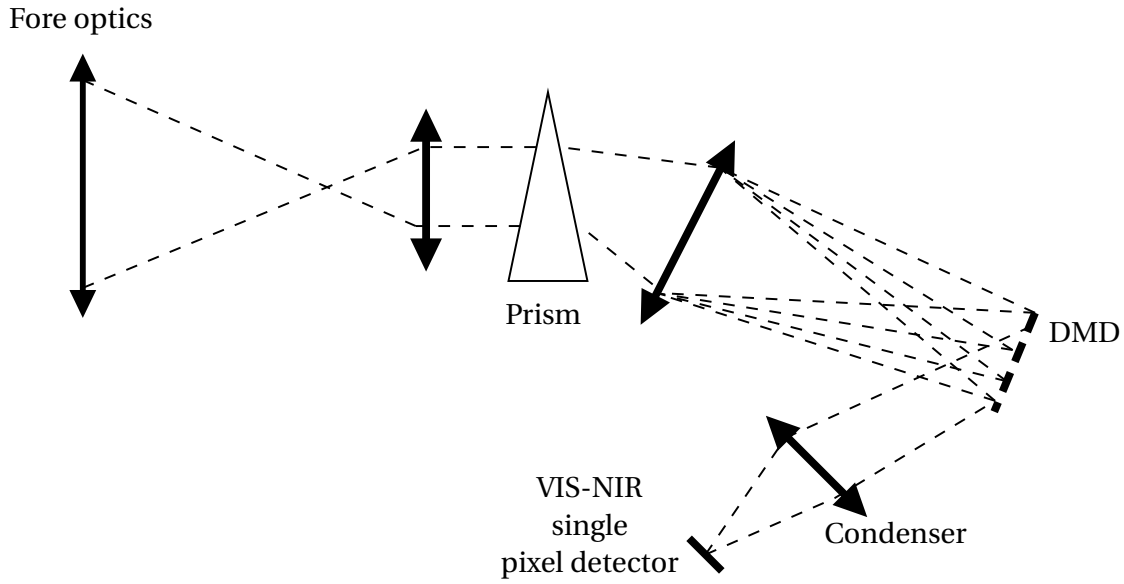


Figure 42: Optical scheme of the instrument [32].

The instrument is composed by a telescope (fore optics in the figure), a prism imaging spectrometer, the SLM (DMD in the figure, with a tilt angle of $\pm 12^\circ$), a condenser and a single pixel detector. The light recollected by the telescope passes through the prism to be split in several spectral bands, the image of each star is spread into its spectrum. After that, the light passes through the DMD which applies a different mask for each measurement. Finally, the light from the DMD reaches the condenser to be recollected in one point where the single pixel detector is positioned.

The fore optics is constituted of a Schmidt-Cassegrain with a parabolic surface for the primary mirror and a hyperbolic surface for the secondary mirror, both coated in aluminium.

A system of baffles is thought to prevent any possible straylight that could reach the detector. For this kind of application, the study of the straylight is extremely important because the light coming from the stars is weak and the scene is sparse.

From the envisaged mass, power and volume budgets, the CS instrument demonstrates a high decrease in the power budgets, about 15 W less than the corresponding classical instrument, although for the mass and volume the difference is not so remarkable. The power reduction is principally due to the suppression of the data compression board.

The instrument is also conceived in double arm configuration to exploit the DMD mirrors in off position. This configuration allows to increase the integration time and so improve the quality with a double acquisition, or to define the position of the stars by adding a narrow band-pass filter.

The optical design with the second arm is reported in Figure 43.

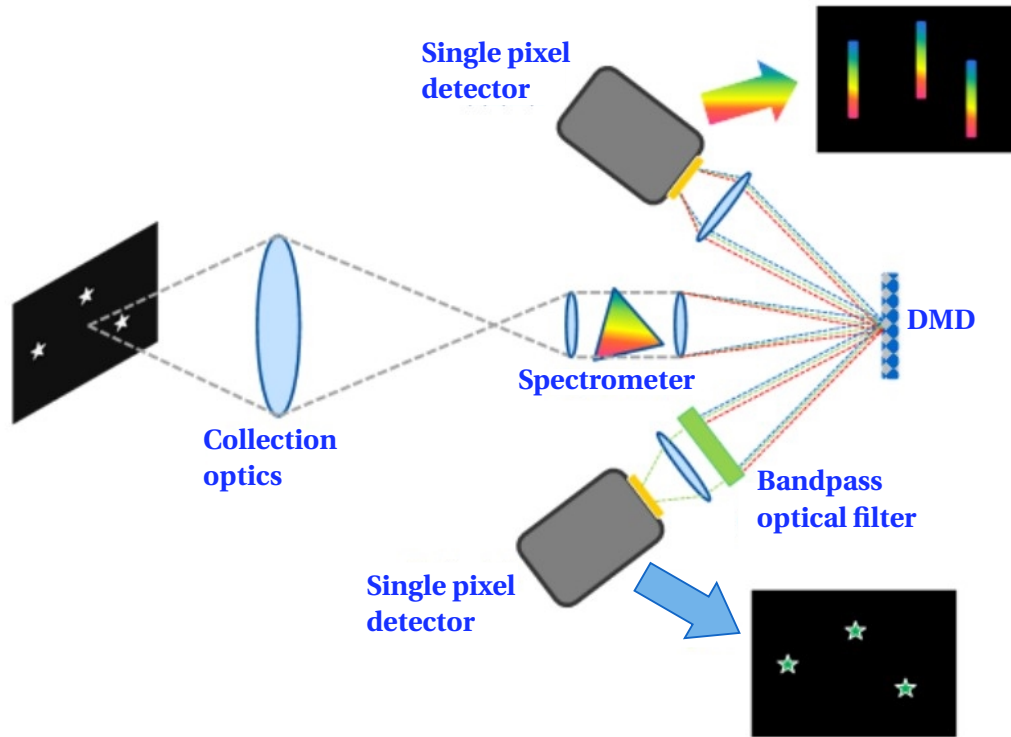


Figure 43: Optical scheme with a double arm configuration [29]

The only difference with respect to the previous configuration is the second arm that is composed by a condenser lens, a narrow-band spectral filter and a single pixel detector.

From the simulations, it has been shown that the configuration with two arms offers better results in terms of quality, but also, the possibility to establish the size limits of Near Earth Objects (NEO) that can be detected.

In fact, the information retrieved from the second arm can be used as input for the reconstruction of the first arm. It will be possible by exploiting specialized algorithms that have the possibility of processing the signal on-board.

5.5 Camera operating in the MIR-TIR for sky observation and real time detection of Near Earth Objects

The instrument is a single pixel CS-based camera that operates at 3-5 μm , whose purpose is to detect the NEOs [32]. The temperature of the payload is about 150 °K, this is a great issue for the DMD that has a minimal survival temperature of 203 °K. The problem is overcome using a photolithographic deposition of a pre-defined pattern on a glass support transparent in MIR band. The change in pattern is obtained by using an X-Y linear stage.

Figure 44 reports the schematic diagram of the instrument.

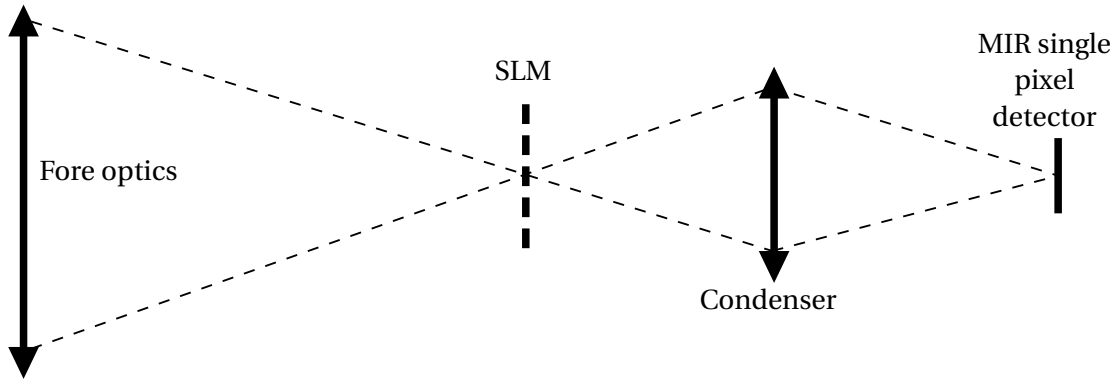


Figure 44: Optical scheme of the instrument [32].

The light, coming from a portion of the sky, enters the telescope which creates a panchromatic image on the coding mask situated in the image plane. The size and the number of pixels of the mask depends on the size of the final image. In order to have a rapid change of pattern, the linear X-Y stage has to move of 50 μm (the size of the pixel pitch) in 0.2 ms and then stop for 0.8 ms to acquire the signal. After that, the light passes through a condenser to be sent to the single pixel detector.

At the moment, the sensor and the linear stage are the most critical elements because the sensor works in counting photons mode and it has a large sensitive area, and the linear stage should have a very high speed of movement. A trade-off between the performance and feasibility has to be done: the solution is to increase the integration time to have a more stable stage and collects a larger amount of signal at the detector, but at the same time, this implies a smaller number of measurements.

In order to test the instrument, some simulations are performed. The utilization of a DMD,

even at the lowest possible temperature, is impossible because of its emissivity. Therefore, the matrix that simulates the coding mask is coded as a block circulant with circulant blocks (BCCB) drawn from a symmetric Bernoulli distribution [32].

In this way, the simulations computed with a CS ratio of 10% have done the minimum diameter of NEOs in function of the distance and body temperature. The instrument is capable of detecting NEOs of 3.5 km at 0.5 AU with a body temperature of 400 K.

5.6 COSMITO & PIGNOLETTO

The COmpressive Sampling Multispectral Imaging camera for remote Observation (COSMITO) and the PIGNOLETTO are two CS instruments under development [30].

The COSMITO telescope operates in VIS-NIR spectral band (440-1740 nm), the CS is implemented using a DMD and the capture of the images is possible thanks to a spectrometer and a multispectral detector. Figure 45 shows the optical scheme of the instrument.

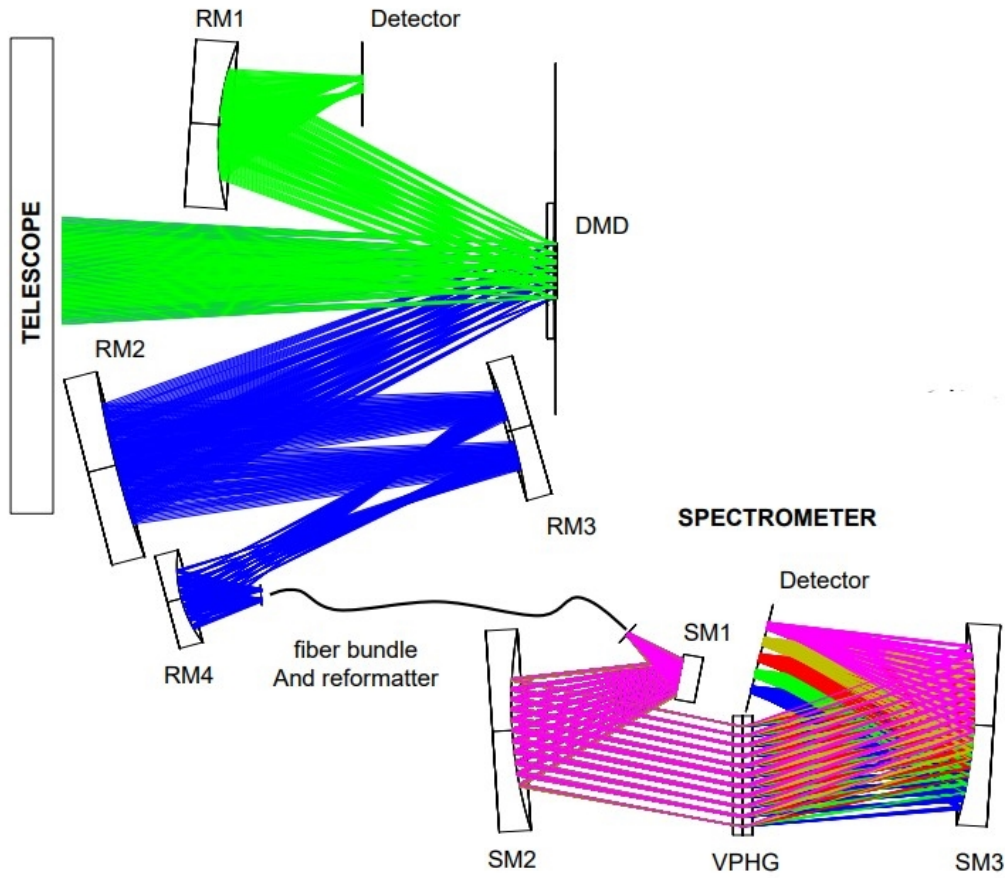


Figure 45: Optical scheme of the COSMITO instrument [30].

The telescope recollects the light and it sends it to the DMD. From the DMD, the light is split into two different channels, the first is monochromatic and reaches the single pixel detector, and the second is hyperspectral and the light passes through the spectrometer to reach the multi-spectral detector.

Some simulations are performed using the pushbroom acquisition model and the TVAL reconstruction algorithm [33].

The PIGNOLETTO aims to monitor the land and precision agriculture using the CS technique. The instrument is a wideband camera studied to operate at 50 m of altitude from a drone. It is composed of three parallel spectrometers operating in the VIS, NIR and SWIR spectral bands ($0.4\text{-}2.5\ \mu\text{m}$). Figure 46 shows the design of the instrument.

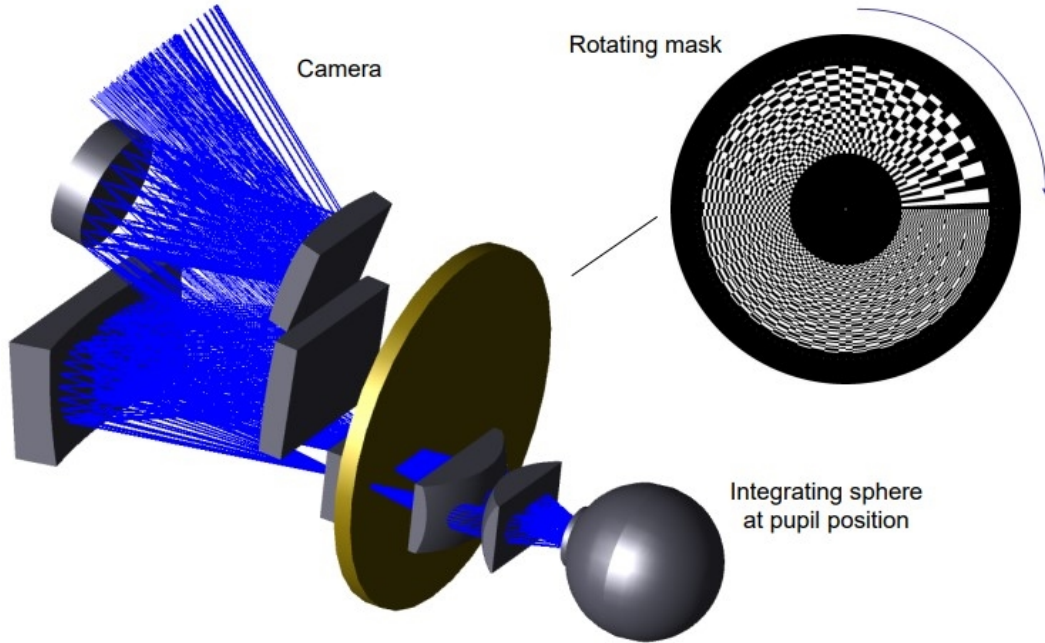


Figure 46: Optical scheme of the PIGNOLETTO instrument [30].

The CS technique is applied using the pushbroom mode and a rotating coded mask. The mask is composed by 64 Hadamard bases that rotate at 8 HZ in order to follow the movement of the drone where the instrument is placed.

An integrating sphere, positioned at the pupil position, makes possible the separation between the camera and the spectrometers, in fact, the entrance slit of these last is situated in the integrating sphere.

In conclusion, several projects using the CS technology are under development. They are not still ready for the spatial application, the SLM and the detectors have to be tested and the implementation of the DL reconstruction algorithms, such as ISTA-Net⁺ and CS-Net⁺, has to be analyzed deeper for the space application. The CS technology has great potential for the space application, after some tests it will be operating at its full potential.

6 Conclusion and future work

6.1 Conclusion

The purpose of this work is to study the relatively new technique of Compressive Sensing for a possible application in space applications. The classical imaging approach could be expensive and complicated in certain spectral bands like the IR and high energy bands. Therefore, the CS technique could be an optimum substitute for the observation in this spectral band.

Since the CS application requires a number of measurements lower than the minimum imposed by the frequency of Nyquist, the mathematical approach is different from the classical optics. The mathematical problem is undetermined and there are several resolution methods to retrieve the original signal: from classical approach, like the ℓ_1 and TV minimization, to the most recent deep learning approach.

In this work, the classical methods, TVAL3 and TVLNR, and the deep learning methods, ISTA-Net⁺ and CS-Net⁺, have been compared to establish the best reconstruction method. After several simulations, where the CS ratio varies from 1% to 50%, it has been demonstrated that the DL approach is the best in terms of quality and computation time. This recent method of reconstruction will make the application of the CS technique more practicable for the space application.

The CS technique allows also the construction of lensless cameras, making the optic system easier and lighter. In this work, two examples of lensless imaging are reported. The first one, the NoRDS-CAIC, has been analyzed via MATLAB simulations. The camera consists only of a programmable LCD screen and a sensor. The LCD disposes of several random masks in front of the sensor, and the algorithm retrieves the original image from the data collected by the sensor. It has been demonstrated that the quality increases with the number of masks and the camera is capable of obtaining good results in short period of time.

The second example is the DiffuserCam that has been constructed and tested in the CSL laboratory. This kind of camera is capable to reconstruct the original image using simple elements (a diffuser and a sensor) in few seconds.

At present, there are several CS instruments under study in view of space application. The studies prove that this new imaging approach has several advantages in terms of reduction of mem-



ory, mass and power budget. The principal modification with respect to the classical instrument is the suppression of the compressing board that saves mass and power. The instruments are not yet ready to be launched in space, but the results are encouraging, therefore the future is not so far.

6.2 Future work

In this thesis, the basic concept of the CS is investigated. The application field of the technique is wide, and its application could have several advantages in the space field.

To pursue the work done, some experimental tests linked to the space application could be performed. The first simulation is the single pixel camera, on which most of the CS instruments already conceived are based. The manipulation will use a DMD, with a relatively high frame rate, a collimator and a single pixel detector to simulate the space instrument. The results coming from the single pixel detector can be implemented in the reconstruction algorithms analyzed in section 3.3 to reconstruct the original image.

Due to its small range of operation, the DiffuserCam can be well suited for applications such as spatial rovers and other close up cameras. This type of camera requires less memory and data link compared to conventional instruments. Moreover, the absence of lenses and mirror makes it less fragile. However, the application has to be improved with better reconstruction methods, for example the deep learning methods, and a mechanism that changes the distance between the ground and the rover in order to have always the same distance between the detector and the observed scene.

Finally, the CS instruments, like the DMD and the detector, have to be tested for the critical space conditions such as the radiation environment for the absence of the atmosphere or the critical temperature present in space.

7 Acknowledgments

First of all, I would to express my gratitude to my promoter Dr. Marc Georges and his team for the proposition of this interesting subject and for the help apported in the implementation of the DiffuserCam in the laboratory.

I would like to thank my parents for the opportunity to study in Belgium and frequent the faculty of my dream. I would to thank also my sister and brother for the encouragement in each choice and the moral support in these years.

Finally, I would to thank my Italian friend to be always present without considering the physical distance; and the friends of the "Centrale Des Courses" for the help in the proofreading of this report and to be a second family in Liège.

Appendices

A Decoding algorithm comparison

TVLNR								TVAL3						
CS ratio [%]	1	5	10	20	30	40	50	1	5	10	20	30	40	50
House	1.1e-2	3.1e-3	3.2e-3	6.3e-4	3.5e-4	2.5e-4	1.6e-4	1.3e-2	5.4e-3	3.4e-3	1.6e-3	1.6e-3	2.3e-3	1.5e-3
Peppers	2.0e-2	6.6e-3	3.6e-3	1.6e-3	7.7e-4	5.2e-4	2.8e-4	2.1e-2	1.1e-2	5.4e-3	7.0e-3	2.7e-3	1.1e-3	7.5e-4
Parrots	1.5e-2	5.4e-3	4.1e-3	2.4e-3	2.2e-3	1.0e-3	7.1e-4	2.1e-2	1.1e-2	4.6e-3	4.3e-3	2.8e-3	2.2e-3	1.6e-3
Monarch	3.4e-2	1.2e-2	6.5e-3	3.3e-3	2.1e-3	2.1e-3	1.3e-3	3.1e-2	1.3e-2	7.3e-3	2.8e-3	1.9e-3	3.5e-3	1.7e-3
Lena	1.9e-2	5.4e-3	3.5e-3	1.7e-3	1.5e-3	9.8e-4	7.6e-4	1.7e-2	6.5e-3	6.1e-3	2.8e-3	1.7e-3	2.8e-3	1.4e-3
Foreman	1.3e-2	1.4e-3	7.3e-4	3.1e-4	1.8e-4	1.1e-4	7.4e-5	3.0e-2	2.9e-2	1.4e-2	9.6e-3	1.1e-2	9.8e-3	8.1e-3
Flintstones	3.7e-2	1.6e-2	9.7e-3	4.5e-3	3.4e-3	2.2e-3	1.5e-3	4.9e-2	2.9e-2	2.1e-2	1.0e-2	6.6e-3	3.8e-3	1.8e-3
Fingerprint	3.0e-2	2.2e-2	1.4e-2	6.4e-3	3.2e-3	2.7e-3	1.7e-3	3.2e-2	2.1e-2	1.6e-2	7.2e-3	4.0e-3	3.3e-3	2.3e-3
Cameraman	1.6e-2	7.6e-3	5.2e-3	2.9e-3	2.0e-3	2.0e-3	1.3e-3	3.1e-2	2.9e-2	1.1e-2	3.9e-3	2.4e-3	8.1e-4	3.9e-4
Boats	1.9e-2	1.3e-2	1.0e-2	2.7e-3	1.8e-3	1.2e-3	6.2e-4	2.1e-2	8.9e-3	5.7e-3	3.9e-3	1.3e-3	1.2e-3	1.1e-3
Barbara	1.9e-2	1.2e-2	1.2e-2	8.2e-3	6.4e-3	5.4e-3	2.0e-3	1.8e-2	9.3e-3	7.1e-3	4.9e-3	3.3e-3	3.0e-3	2.0e-3

ISTA-Net ⁺							CS-Net ⁺						
CS ratio [%]	1	4	10	25	40	50	1	5	10	20	30	40	50
House	1.0e-2	3.7e-3	8.3e-4	2.4e-4	1.4e-4	9.8e-5	3.9e-3	9.9e-4	5.5e-4	2.9e-4	1.6e-4	1.1e-4	6.3e-5
Peppers	2.1e-2	8.0e-3	1.8e-3	3.6e-4	1.8e-4	1.2e-4	8.5e-3	2.9e-3	1.6e-3	8.3e-4	4.4e-4	2.7e-4	1.9e-4
Parrots	1.5e-2	6.4e-3	2.2e-3	6.4e-4	2.9e-4	1.9e-4	6.0e-3	2.8e-3	1.5e-3	7.3e-4	4.1e-4	2.4e-4	1.5e-4
Monarch	3.2e-2	1.1e-2	2.5e-3	4.6e-4	1.7e-4	1e-4	1.6e-2	2.8e-3	1.4e-3	5.3e-4	3.1e-4	1.9e-4	1.3e-4
Lena	1.5e-2	5.5e-3	1.7e-3	5.4e-4	2.5e-4	1.5e-4	5.7e-3	2.1e-3	1.2e-3	5.9e-4	3.2e-4	1.9e-4	1.2e-4
Foreman	9.0e-3	2.3e-3	4.5e-4	1.2e-4	6.3e-5	4.3e-5	2.2e-2	7.5e-3	3.9e-3	1.6e-3	1.1e-3	7.3e-4	5.6e-4
Flintstones	4.0e-2	1.7e-2	4.2e-3	1.0e-3	5.9e-4	4.3e-4	2.2e-2	7.5e-3	3.9e-3	1.6e-3	1.1e-3	7.3e-4	5.6e-4
Fingerprint	3.3e-2	1.9e-2	5.5e-3	1.5e-3	6.1e-4	3.6e-4	2.4e-2	5.2e-3	2.3e-3	8.9e-4	3.8e-4	2.1e-4	1.1e-4
Cameraman	1.8e-2	9.0e-3	4.3e-3	1.2e-3	5.9e-4	3.7e-4	9.3e-3	4.5e-3	2.8e-3	1.4e-3	8.5e-4	5.7e-4	3.9e-4
Boats	1.4e-2	6.0e-3	1.8e-3	4.3e-4	1.7e-4	1.0e-4	6.3e-3	1.9e-3	1.0e-3	4.6e-4	2.6e-4	1.6e-4	9.0e-5
Barbara	1.4e-2	7.9e-3	4.4e-3	1.2e-3	3.9e-4	2.2e-4	6.7e-3	4.2e-3	3.6e-3	2.1e-3	7.5e-4	3.4e-4	1.5e-4

Table 8: MSE

TVLNR								TVAL3						
CS ratio [%]	1	5	10	20	30	40	50	1	5	10	20	30	40	50
House	19.53	25.03	24.85	32.03	34.54	36.08	37.95	16.85	19.75	26.28	26.50	27.59	28.06	28.73
Peppers	16.97	21.81	24.45	28.02	31.14	32.85	35.54	16.59	19.01	20.53	23.73	26.95	28.54	31.47
Parrots	18.11	22.64	23.92	26.26	26.64	29.81	31.49	16.89	19.50	23.34	23.62	25.46	26.67	27.91
Monarch	14.66	19.05	21.89	24.83	26.78	26.86	28.72	15.38	18.70	21.10	23.77	26.86	27.50	27.50
Lena	17.25	22.69	24.52	27.76	28.14	30.07	31.21	17.64	21.89	22.15	25.51	27.65	25.59	28.56
Foreman	18.85	28.65	31.34	35.03	37.34	39.62	41.30	13.30	15.61	16.99	19.98	20.18	20.68	21.03
Flintstones	14.36	17.84	20.15	23.46	24.65	26.64	28.35	13.58	15.88	16.92	18.70	21.60	24.99	27.54
Fingerprint	15.21	16.64	18.38	21.94	24.91	25.76	27.58	14.92	16.83	18.06	21.43	23.95	24.76	26.36
Cameraman	17.83	21.19	22.86	25.44	26.91	26.99	28.96	15.08	15.33	19.66	24.08	26.11	30.90	34.08
Boats	17.16	18.92	19.85	25.71	27.56	29.21	32.08	16.79	20.48	22.48	24.12	28.74	29.20	29.76
Barbara	17.09	19.05	19.24	20.86	21.92	22.70	26.97	17.34	20.32	21.49	23.14	24.79	25.26	27.09

ISTA-Net ⁺							CS-Net ⁺						
CS ratio [%]	1	4	10	25	40	50	1	5	10	20	30	40	50
House	19.91	24.32	30.81	36.23	38.61	40.06	24.12	30.06	32.59	35.41	37.93	39.58	41.98
Peppers	16.81	20.96	27.36	34.43	37.49	39.05	20.69	25.40	27.85	30.83	33.56	35.73	37.27
Parrots	18.13	21.97	26.56	31.96	35.25	37.27	22.23	25.60	28.11	31.35	33.85	36.10	38.23
Monarch	14.99	19.53	25.97	33.35	37.70	39.99	18.07	25.55	28.59	32.77	35.08	37.09	38.88
Lena	18.28	22.63	27.67	32.66	36.05	38.26	22.43	26.87	29.19	32.27	34.88	37.20	39.23
Foreman	20.44	26.43	33.46	39.07	41.99	43.66	26.77	32.38	35.05	38.69	40.65	42.27	43.86
Flintstones	13.93	17.61	23.75	29.94	32.32	33.67	16.64	21.26	24.04	27.85	29.62	31.39	32.51
Fingerprint	14.85	17.31	22.57	28.24	32.14	34.46	16.22	22.84	26.37	30.48	34.14	36.91	39.46
Cameraman	17.33	20.45	23.65	29.17	32.29	34.36	20.32	23.46	25.56	28.63	30.70	32.47	34.04
Boats	18.51	22.22	27.42	33.66	37.67	39.84	21.98	27.27	29.99	33.41	35.86	38.07	40.44
Barbara	18.42	21.01	23.61	29.26	34.07	36.61	21.77	23.76	24.41	26.69	31.23	34.73	38.30

Table 9: PSNR

TVLNR								TVAL3						
CS ratio [%]	1	5	10	20	30	40	50	1	5	10	20	30	40	50
House	0.61	0.76	0.82	0.87	0.91	0.93	0.94	0.40	0.61	0.69	0.83	0.88	0.91	0.93
Peppers	0.48	0.67	0.78	0.86	0.91	0.93	0.95	0.42	0.58	0.68	0.81	0.86	0.92	0.95
Parrots	0.65	0.76	0.82	0.88	0.91	0.94	0.95	0.47	0.65	0.73	0.82	0.89	0.92	0.94
Monarch	0.41	0.65	0.75	0.86	0.90	0.93	0.95	0.34	0.54	0.68	0.82	0.89	0.92	0.95
Lena	0.51	0.69	0.76	0.86	0.90	0.93	0.95	0.41	0.58	0.68	0.82	0.85	0.91	0.93
Foreman	0.65	0.82	0.87	0.92	0.95	0.96	0.97	0.48	0.66	0.77	0.86	0.90	0.93	0.94
Flintstones	0.32	0.52	0.65	0.78	0.83	0.87	0.90	0.28	0.39	0.50	0.66	0.77	0.83	0.88
Fingerprint	0.15	0.34	0.51	0.74	0.86	0.88	0.92	0.16	0.34	0.52	0.72	0.83	0.90	0.93
Cameraman	0.58	0.69	0.75	0.82	0.85	0.89	0.92	0.34	0.49	0.57	0.70	0.79	0.87	0.93
Boats	0.42	0.59	0.68	0.82	0.88	0.92	0.95	0.33	0.51	0.63	0.77	0.84	0.89	0.92
Barbara	0.40	0.51	0.58	0.70	0.78	0.85	0.90	0.34	0.46	0.53	0.63	0.74	0.82	0.88

ISTA-Net ⁺							CS-Net ⁺						
CS ratio [%]	1	4	10	25	40	50	1	5	10	20	30	40	50
House	0.56	0.69	0.84	0.91	0.94	0.96	0.69	0.83	0.87	0.91	0.93	0.95	0.97
Peppers	0.42	0.59	0.81	0.93	0.96	0.97	0.62	0.82	0.88	0.93	0.95	0.96	0.97
Parrots	0.55	0.70	0.85	0.93	0.96	0.97	0.71	0.84	0.89	0.93	0.95	0.97	0.98
Monarch	0.39	0.61	0.84	0.95	0.97	0.98	0.51	0.83	0.90	0.96	0.97	0.98	0.99
Lena	0.46	0.64	0.82	0.93	0.96	0.97	0.62	0.80	0.87	0.93	0.96	0.97	0.98
Foreman	0.59	0.75	0.89	0.95	0.97	0.98	0.77	0.89	0.93	0.96	0.97	0.98	0.99
Flintstones	0.26	0.47	0.75	0.88	0.91	0.93	0.39	0.66	0.78	0.88	0.90	0.92	0.94
Fingerprint	0.18	0.45	0.76	0.92	0.97	0.98	0.15	0.77	0.88	0.96	0.98	0.99	0.99
Cameraman	0.49	0.61	0.76	0.88	0.93	0.95	0.62	0.75	0.83	0.90	0.92	0.94	0.96
Boats	0.42	0.59	0.79	0.93	0.97	0.98	0.54	0.78	0.87	0.93	0.96	0.97	0.98
Barbara	0.38	0.53	0.69	0.90	0.96	0.97	0.50	0.66	0.72	0.83	0.92	0.96	0.98

Table 10: SSIM

TVLNR								TVAL3						
CS ratio [%]	1	5	10	20	30	40	50	1	5	10	20	30	40	50
House	887	288	243	151	121	105	93	102	93	98	81	78	80	81
Peppers	1000	1000	1000	661	487	334	261	93	94	88	88	89	80	84
Parrots	903	573	890	802	589	482	372	99	86	84	75	74	75	79
Monarch	1000	1000	1000	1000	845	565	407	101	94	98	85	87	77	79
Lena	939	766	956	661	537	414	309	105	109	90	84	87	70	82
Foreman	897	274	229	136	115	100	85	86	90	85	69	67	77	64
Flintstones	1000	1000	1000	1000	921	633	435	94	105	87	82	80	84	80
Fingerprint	1000	1000	1000	1000	1000	775	567	97	95	102	80	80	77	72
Cameraman	987	824	1000	1000	951	626	424	93	93	95	90	83	81	82
Boats	915	804	839	578	388	237	181	105	101	88	81	77	78	69
Barbara	931	1000	1000	1000	938	653	473	109	92	86	89	82	72	82

Table 11: Number of iterations

	TVLNR	TVAL3	ISTA-Net ⁺	CS-Net ⁺
House	246	3.92	0.14	0.19
Peppers	642	4.01	0.15	0.20
Parrots	691	3.73	0.15	0.20
Monarch	829	3.83	0.14	0.19
Lena	640	3.91	0.15	0.20
Foreman	235	3.49	0.14	0.20
Flintstones	3238	25.11	0.546	0.80
Fingerprint	4415	27.18	0.977	0.80
Cameraman	798	3.83	0.14	0.19
Boats	535	3.62	0.15	0.20
Barbara	881	3.80	0.15	0.19

Table 12: CPU

B Decoding TV algorithm comparison with noise

Error percentage [%]	TVLNR						TVAL3					
	0	2	4	6	8	10	0	2	4	6	8	10
House	2.5e-4	3.2e-4	1.0e-3	1.8e-3	2.9e-3	3.9e-3	1.6e-3	1.7e-3	2.1e-3	2.2e-3	3.2e-3	2.7e-3
Peppers	5.2e-4	6.1e-4	7.5e-4	1.3e-3	1.5e-3	2.3e-3	1.5e-3	2.2e-3	1.1e-3	1.6e-3	3.9e-3	1.6e-3
Parrots	1.0e-3	1.2e-3	1.3e-3	1.7e-3	2.5e-3	3.2e-3	2.6e-3	3.2e-3	2.2e-3	1.9e-3	2.3e-3	3.9e-3
Monarch	2.1e-3	1.6e-3	2.3e-3	2.2e-3	3.1e-3	3.6e-3	3.1e-3	2.8e-3	2.0e-3	3.8e-3	1.5e-3	1.1e-3
Lena	9.8e-4	8.4e-4	1.1e-3	1.5e-3	2.0e-3	2.5e-3	7.7e-4	8.6e-4	2.3e-3	2.7e-3	1.4e-3	1.4e-3
Foreman	1.1e-4	3.4e-4	9.0e-4	1.7e-3	3.4e-3	4.7e-3	7.5e-3	8.0e-3	8.2e-3	1.0e-2	1.0e-2	1.7e-2
Flintstones	2.2e-3	2.1e-3	2.5e-3	3.6e-3	4.3e-3	3.9e-3	4.0e-3	3.3e-3	4.9e-3	3.8e-3	6.4e-3	5.6e-3
Fingerprint	2.7e-3	2.8e-3	3.1e-3	3.5e-3	3.7e-3	4.0e-3	2.1e-3	3.9e-3	2.5e-3	3.9e-3	2.5e-3	3.7e-3
Cameraman	2.0e-3	1.5e-3	1.8e-3	2.2e-3	2.5e-3	2.8e-3	8.0e-4	9.4e-4	1.2e-3	1.2e-3	1.2e-3	2.1e-3
Boats	1.2e-3	7.9e-4	3.4e-3	4.3e-3	6.3e-3	6.5e-3	1.2e-3	2.8e-3	1.7e-3	9.7e-4	1.0e-3	1.9e-3
Barbara	5.4e-3	4.6e-3	4.3e-3	5.3e-3	8.3e-3	5.8e-3	3.5e-3	3.5e-3	1.8e-3	4.7e-3	6.5e-3	3.3e-3

Table 13: MSE

Error percentage [%]	TVLNR						TVAL3					
	0	2	4	6	8	10	0	2	4	6	8	10
House	36.08	34.88	29.99	27.48	25.33	24.12	27.95	27.62	26.87	26.49	24.97	25.65
Peppers	32.85	32.16	31.23	28.83	28.12	26.44	28.23	26.64	29.76	28.03	24.06	28.09
Parrots	29.81	29.37	28.76	27.77	25.95	24.97	25.79	24.91	26.48	27.28	26.44	24.14
Monarch	26.86	28.07	26.29	26.67	25.08	24.44	25.09	25.55	27.03	24.15	28.13	29.67
Lena	30.07	30.77	29.72	28.26	27.02	25.99	31.11	30.64	26.42	25.66	28.63	28.59
Foreman	39.62	34.66	30.45	27.57	24.69	23.29	21.26	20.98	20.86	19.95	19.86	17.69
Flintstones	26.64	26.78	25.98	24.39	23.65	24.06	23.95	24.86	23.14	24.19	21.91	22.53
Fingerprint	25.76	25.53	25.03	24.56	24.36	23.97	26.71	24.12	26.03	24.14	26.06	24.31
Cameraman	26.99	28.29	27.37	26.59	26.08	25.48	30.95	30.29	29.18	29.35	29.33	26.83
Boats	29.21	30.99	24.71	23.65	22.03	21.87	29.38	25.46	27.58	30.14	29.83	27.18
Barbara	22.70	23.37	23.65	22.78	20.81	22.36	24.55	24.55	27.42	23.28	21.90	24.79

Table 14: PSNR

Error percentage [%]	TVLNR						TVAL3					
	0	2	4	6	8	10	0	2	4	6	8	10
House	0.93	0.91	0.85	0.78	0.69	0.61	0.91	0.90	0.89	0.88	0.86	0.84
Peppers	0.93	0.92	0.88	0.82	0.76	0.69	0.92	0.91	0.91	0.90	0.88	0.88
Parrots	0.94	0.92	0.88	0.81	0.73	0.65	0.92	0.92	0.91	0.90	0.88	0.85
Monarch	0.93	0.93	0.90	0.85	0.80	0.75	0.92	0.92	0.93	0.91	0.91	0.90
Lena	0.93	0.91	0.87	0.82	0.75	0.67	0.91	0.91	0.90	0.89	0.89	0.88
Foreman	0.96	0.94	0.87	0.78	0.67	0.57	0.93	0.92	0.91	0.89	0.87	0.83
Flintstones	0.87	0.86	0.84	0.79	0.78	0.78	0.84	0.84	0.81	0.82	0.79	0.79
Fingerprint	0.88	0.88	0.87	0.85	0.84	0.83	0.90	0.89	0.90	0.88	0.90	0.89
Cameraman	0.89	0.88	0.84	0.78	0.71	0.65	0.87	0.86	0.85	0.84	0.82	0.79
Boats	0.92	0.91	0.87	0.80	0.74	0.68	0.89	0.88	0.89	0.88	0.87	0.85
Barbara	0.85	0.84	0.81	0.77	0.71	0.70	0.82	0.82	0.82	0.81	0.79	0.79

Table 15: SSIM

Error percentage [%]	TVLNR						TVAL3					
	0	2	4	6	8	10	0	2	4	6	8	10
House	105	111	199	265	370	507	78	70	82	81	88	78
Peppers	334	362	420	491	660	856	86	84	84	76	79	77
Parrots	482	474	510	577	604	737	83	81	81	81	70	75
Monarch	565	591	617	701	784	891	78	78	83	86	82	81
Lena	414	420	453	530	597	751	79	85	82	90	78	85
Foreman	100	101	105	187	286	441	79	77	77	74	74	78
Flintstones	633	654	729	981	1000	1000	76	81	67	73	79	73
Fingerprint	643	677	737	993	1000	1000	77	75	77	75	81	82
Cameraman	626	654	702	782	857	1000	71	81	75	80	86	75
Boats	237	287	335	444	565	709	86	84	72	87	75	76
Barbara	653	666	710	786	925	1000	80	79	83	80	80	82

Table 16: Number of iterations

References

- [1] V. Boominathan et al. “Lensless Imaging: A computational renaissance”. In: *IEEE Signal Processing Magazine* 33.5 (Sept. 2016), pp. 23–35. ISSN: 1053-5888. DOI: [10.1109/MSP.2016.2581921](https://doi.org/10.1109/MSP.2016.2581921). <http://ieeexplore.ieee.org/document/7559956/> (visited on 03/21/2022).
- [2] N. Gehrels et al. “The *Swift* Gamma-Ray Burst Mission”. In: *The Astrophysical Journal* 611.2 (Aug. 20, 2004), pp. 1005–1020. ISSN: 0004-637X, 1538-4357. DOI: [10.1086/422091](https://doi.org/10.1086/422091). <https://iopscience.iop.org/article/10.1086/422091> (visited on 03/25/2022).
- [3] T. Tumer et al. “All-sky X-ray and Gamma-ray Astronomy Monitor (AXGAM)”. In: *IEEE Transactions on Nuclear Science* 44.3 (June 1997), pp. 572–576. ISSN: 0018-9499, 1558-1578. DOI: [10.1109/23.603713](https://doi.org/10.1109/23.603713). <https://ieeexplore.ieee.org/document/603713/> (visited on 03/25/2022).
- [4] M. Feroci et al. “SuperAGILE: the hard X-ray Imager for the AGILE space mission”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 581.3 (Nov. 2007), pp. 728–754. ISSN: 01689002. DOI: [10.1016/j.nima.2007.07.147](https://doi.org/10.1016/j.nima.2007.07.147). arXiv: [0708.0123](https://arxiv.org/abs/0708.0123). <http://arxiv.org/abs/0708.0123> (visited on 03/25/2022).
- [5] J. E. Grindlay and J. Hong. “Optimizing wide-field coded aperture imaging: radial mask holes and scanning”. In: *Optics for EUV, X-Ray, and Gamma-Ray Astronomy*. Vol. 5168. International Society for Optics and Photonics. 2004, pp. 402–410.
- [6] G. Belanger et al. “Detection of hard X-ray emission from the Galactic nuclear region with INTEGRAL”. In: *The Astrophysical Journal* 601.2 (Feb. 1, 2004), pp. L163–L166. ISSN: 0004-637X, 1538-4357. DOI: [10.1086/381738](https://doi.org/10.1086/381738). arXiv: [astro-ph/0311147](https://arxiv.org/abs/astro-ph/0311147). <http://arxiv.org/abs/astro-ph/0311147> (visited on 03/25/2022).
- [7] M. S. Asif et al. “FlatCam: Thin, Bare-Sensor Cameras using Coded Aperture and Computation”. In: *arXiv:1509.00116 [cs]* (Jan. 27, 2016). arXiv: [1509.00116](https://arxiv.org/abs/1509.00116). <http://arxiv.org/abs/1509.00116> (visited on 03/25/2022).
- [8] Z. Jiang et al. “Programmable liquid crystal display based noise reduced dynamic synthetic coded aperture imaging camera (NoRDS-CAIC)”. In: *Optics Express* 28.4 (Feb. 17, 2020), p. 5221. ISSN: 1094-4087. DOI: [10.1364/OE.385547](https://doi.org/10.1364/OE.385547). <https://opg.optica.org/abstract.cfm?URI=oe-28-4-5221> (visited on 04/09/2022).

- [9] N. Antipa et al. “DiffuserCam: lensless single-exposure 3D imaging”. In: *Optica* 5.1 (Jan. 20, 2018), pp. 1–9. ISSN: 2334-2536. DOI: [10 . 1364 / OPTICA . 5 . 000001](https://doi.org/10.1364/OPTICA.5.000001). <https://opg.optica.org/optica/abstract.cfm?uri=optica-5-1-1> (visited on 04/09/2022).
- [10] J. Legaludec et al. “Mid-infrared multispectral lensless imaging for wide-field and label-free microbial identification”. In: *Biomedical Spectroscopy, Microscopy, and Imaging*. Biomedical Spectroscopy, Microscopy, and Imaging. Ed. by J. Popp and C. Gergely. Online Only, France: SPIE, Apr. 1, 2020, p. 10. ISBN: 9781510634909 9781510634916. DOI: [10 . 1117 / 12 . 2557502](https://doi.org/10.1117/12.2557502). <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11359/2557502/Mid-infrared-multispectral-lensless-imaging-for-wide-field-and-label/10.1117/12.2557502.full> (visited on 04/09/2022).
- [11] K. Delikoyun et al. “Lensless Digital in-Line Holographic Microscopy for Space Biotechnology Applications”. In: *2019 9th International Conference on Recent Advances in Space Technologies (RAST)*. 2019 9th International Conference on Recent Advances in Space Technologies (RAST). Istanbul, Turkey: IEEE, June 2019, pp. 937–940. ISBN: 9781538694480. DOI: [10 . 1109 / RAST . 2019 . 8767842](https://doi.org/10.1109/RAST.2019.8767842). <https://ieeexplore.ieee.org/document/8767842/> (visited on 04/09/2022).
- [12] S. R. Gottesman and E. E. Fenimore. “New family of binary arrays for coded aperture imaging”. In: *Applied Optics* 28.20 (Oct. 15, 1989), p. 4344. ISSN: 0003-6935, 1539-4522. DOI: [10 . 1364 / AO . 28 . 004344](https://doi.org/10.1364/AO.28.004344). <https://opg.optica.org/abstract.cfm?URI=ao-28-20-4344> (visited on 03/25/2022).
- [13] D. Mackenzie. “Compressed sensing makes every pixel count”. In: 2009. <https://www.ams.org/publicoutreach/math-history/hap7-pixel.pdf>.
- [14] F. Magalhães et al. “Active illumination single-pixel camera based on compressive sensing”. In: *Applied Optics* 50.4 (Feb. 1, 2011), p. 405. ISSN: 0003-6935, 1539-4522. DOI: [10 . 1364 / AO . 50 . 000405](https://doi.org/10.1364/AO.50.000405). <https://opg.optica.org/abstract.cfm?URI=ao-50-4-405> (visited on 04/09/2022).
- [15] M. F. Duarte et al. “Single-pixel imaging via compressive sampling”. In: *IEEE Signal Processing Magazine* 25.2 (Mar. 2008), pp. 83–91. ISSN: 1053-5888. DOI: [10 . 1109 / MSP . 2007 . 914730](https://doi.org/10.1109/MSP.2007.914730). <http://ieeexplore.ieee.org/document/4472247/> (visited on 04/09/2022).

- [16] R. F. Marcia. “Compressed sensing for practical optical imaging systems: a tutorial”. In: *Optical Engineering* 50.7 (July 1, 2011), p. 072601. ISSN: 0091-3286. DOI: [10.1117/1.3596602](https://doi.org/10.1117/1.3596602). <http://opticalengineering.spiedigitallibrary.org/article.aspx?doi=10.1117/1.3596602> (visited on 04/09/2022).
- [17] E. J. Candes and M. B. Wakin. “An Introduction To Compressive Sampling”. In: *IEEE Signal Processing Magazine* 25.2 (2008), pp. 21–30. DOI: [10.1109/MSP.2007.914731](https://doi.org/10.1109/MSP.2007.914731).
- [18] C. Li. “An efficient algorithm for total variation regularization with applications to the single pixel camera and compressive sensing”. In: 2010.
- [19] P. Hanumanth, P. Bhavana, and S. Subbarayappa. “Application of deep learning and compressed sensing for reconstruction of images”. In: *Journal of Physics: Conference Series* 1706.1 (Dec. 1, 2020), p. 012068. ISSN: 1742-6588, 1742-6596. DOI: [10.1088/1742-6596/1706/1/012068](https://doi.org/10.1088/1742-6596/1706/1/012068). <https://iopscience.iop.org/article/10.1088/1742-6596/1706/1/012068> (visited on 04/09/2022).
- [20] A. L. Machidon and V. Pejovic. “Deep Learning Techniques for Compressive Sensing-Based Reconstruction and Inference – A Ubiquitous Systems Perspective”. In: *arXiv:2105.13191 [cs, eess]* (May 26, 2021). arXiv: [2105.13191](https://arxiv.org/abs/2105.13191). <http://arxiv.org/abs/2105.13191> (visited on 04/09/2022).
- [21] V. Raimondi et al. “A feasibility study for a compressive sensing imager in the medium infrared for hotspot detection”. In: *International Conference on Space Optics — ICSO 2020*. International Conference on Space Optics — ICSO 2021. Ed. by Z. Sodnik, B. Cugny, and N. Karafolas. Online Only, France: SPIE, June 11, 2021, p. 188. ISBN: 9781510645486 9781510645493. DOI: [10.1117/12.2599938](https://doi.org/10.1117/12.2599938). <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11852/2599938/A-feasibility-study-for-a-compressive-sensing-imager-in-the/10.1117/12.2599938.full> (visited on 03/21/2022).
- [22] C. Lastrì et al. “Exploring the Potential of Compressive Sensing and Super-Resolution for Space Applications in the MIR-TIR”. In: *The 15th International Workshop on Advanced Infrared Technology and Applications*. The 15th International Workshop on Advanced Infrared Technology and Applications. MDPI, Sept. 27, 2019, p. 35. DOI: [10.3390/proceedings2019027035](https://doi.org/10.3390/proceedings2019027035). <https://www.mdpi.com/2504-3900/27/1/35> (visited on 04/09/2022).

- [23] Jian Zhang et al. “Improved total variation based image compressive sensing recovery by nonlocal regularization”. In: *2013 IEEE International Symposium on Circuits and Systems (ISCAS2013)*. 2013 IEEE International Symposium on Circuits and Systems (ISCAS). Beijing: IEEE, May 2013, pp. 2836–2839. ISBN: 9781467357623 9781467357609 9781467357616. DOI: [10.1109/ISCAS.2013.6572469](https://doi.org/10.1109/ISCAS.2013.6572469). <http://ieeexplore.ieee.org/document/6572469/> (visited on 04/09/2022).
- [24] J. Zhang and B. Ghanem. “ISTA-Net: Interpretable Optimization-Inspired Deep Network for Image Compressive Sensing”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT: IEEE, June 2018, pp. 1828–1837. ISBN: 9781538664209. DOI: [10.1109/CVPR.2018.00196](https://doi.org/10.1109/CVPR.2018.00196). <https://ieeexplore.ieee.org/document/8578294/> (visited on 04/09/2022).
- [25] W. Shi et al. “Image Compressed Sensing Using Convolutional Neural Network”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 375–388. ISSN: 1057-7149, 1941-0042. DOI: [10.1109/TIP.2019.2928136](https://doi.org/10.1109/TIP.2019.2928136). <https://ieeexplore.ieee.org/document/8765626/> (visited on 04/09/2022).
- [26] U. Sara, M. Akter, and M. S. Uddin. “Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study”. In: *Journal of Computer and Communications* 07.3 (2019), pp. 8–18. ISSN: 2327-5219, 2327-5227. DOI: [10.4236/jcc.2019.73002](https://doi.org/10.4236/jcc.2019.73002). <http://www.scirp.org/journal/doi.aspx?DOI=10.4236/jcc.2019.73002> (visited on 04/09/2022).
- [27] G. Coluccia et al. “Optical Compressive Imaging Technologies for Space Big Data”. In: *IEEE Transactions on Big Data* 6.3 (2020), pp. 430–442. DOI: [10.1109/TBDATA.2019.2907135](https://doi.org/10.1109/TBDATA.2019.2907135).
- [28] V. Raimondi et al. “Compressive sensing instrumental concepts for space applications”. In: *Unconventional Optical Imaging III*. Unconventional Optical Imaging III. Ed. by M. P. Georges, G. Popescu, and N. Verrier. Strasbourg, France: SPIE, May 20, 2022, p. 23. ISBN: 9781510651487 9781510651494. DOI: [10.1117/12.2625305](https://doi.org/10.1117/12.2625305). <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/12136/2625305/Compressive-sensing-instrumental-concepts-for-space-applications/10.1117/12.2625305.full> (visited on 05/28/2022).

- [29] E. contract N° 4000116423/15/NL/BJ/gp. “Optical Compressive Sensing (CS) Technologies for Space Applications (OCS-TECH)”. In: *Executive Summary* (2018). https://nebula.esa.int/sites/default/files/neb_study/1265/C4000116423ExS.pdf.
- [30] G. Pariani et al. “Application of compressive sensing for image acquisition in different environments”. In: *Unconventional Optical Imaging III*. Unconventional Optical Imaging III. Ed. by M. P. Georges, G. Popescu, and N. Verrier. Strasbourg, France: SPIE, May 20, 2022, p. 92. ISBN: 9781510656031. DOI: 10.1117/12.2622301. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/PC12136/2622301/Application-of-compressive-sensing-for-image-acquisition-in-different-environments/10.1117/12.2622301.full> (visited on 05/28/2022).
- [31] L. Oggioni and G. Pariani. “Compressive sensing for earth observation: the effect of a moving scene”. In: *Unconventional Optical Imaging III*. Unconventional Optical Imaging III. Ed. by M. P. Georges, G. Popescu, and N. Verrier. Strasbourg, France: SPIE, May 20, 2022, p. 25. ISBN: 9781510651487 9781510651494. DOI: 10.1117/12.2621681. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/12136/2621681/Compressive-sensing-for-earth-observation--the-effect-of-a/10.1117/12.2621681.full> (visited on 05/28/2022).
- [32] D. Guzzi et al. “Optical compressive sensing technologies for space applications: instrumental concepts and performance analysis”. In: *International Conference on Space Optics — ICSO 2018*. International Conference on Space Optics - ICSO 2018. Ed. by N. Karafolas, Z. Sodnik, and B. Cugny. Chania, Greece: SPIE, July 12, 2019, p. 226. ISBN: 9781510630772 9781510630789. DOI: 10.1117/12.2536146. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11180/2536146/Optical-compressive-sensing-technologies-for-space-applications--instrumental-concepts/10.1117/12.2536146.full> (visited on 04/09/2022).
- [33] G. Pariani et al. “Compressive sampling for multispectral imaging in the vis-NIR-TIR: optical design of space telescopes”. In: *Space Telescopes and Instrumentation 2018: Optical, Infrared, and Millimeter Wave*. Space Telescopes and Instrumentation 2018: Optical, Infrared, and Millimeter Wave. Ed. by H. A. MacEwen et al. Austin, United States: SPIE, July 6, 2018, p. 198. ISBN: 9781510619494 9781510619500. DOI: 10.1117/12.2312008. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10698/2312008/Compressive-sampling-for-multispectral-imaging-in-the-vis-NIR-TIR/10.1117/12.2312008.full> (visited on 05/30/2022).

- [34] R. F. Marcia and R. M. Willett. “Compressive coded aperture superresolution image reconstruction”. In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2008, pp. 833–836. DOI: [10.1109/ICASSP.2008.4517739](https://doi.org/10.1109/ICASSP.2008.4517739).
- [35] W.-P. Zhou et al. “Fast compression and reconstruction of astronomical images based on compressed sensing”. In: *Research in Astronomy and Astrophysics* 14.9 (Aug. 2014), pp. 1207–1214. DOI: [10.1088/1674-4527/14/9/011](https://doi.org/10.1088/1674-4527/14/9/011). <https://doi.org/10.1088/1674-4527/14/9/011>.
- [36] G. Huang et al. “Lensless Imaging by Compressive Sensing”. In: Sept. 2013. DOI: [10.1109/ICIP.2013.6738433](https://doi.org/10.1109/ICIP.2013.6738433).
- [37] W. Chi and N. George. “Optical imaging with phase-coded aperture”. In: *Optics Express* 19.5 (Feb. 28, 2011), p. 4294. ISSN: 1094-4087. DOI: [10.1364/OE.19.004294](https://doi.org/10.1364/OE.19.004294). <https://opg.optica.org/oe/abstract.cfm?uri=oe-19-5-4294> (visited on 04/09/2022).
- [38] J. Bobin, J.-L. Starck, and R. Ottensamer. “Compressed Sensing in Astronomy”. In: *IEEE Journal of Selected Topics in Signal Processing* 2.5 (Oct. 2008), pp. 718–726. ISSN: 1932-4553. DOI: [10.1109/JSTSP.2008.2005337](https://doi.org/10.1109/JSTSP.2008.2005337). <http://ieeexplore.ieee.org/document/4703508/> (visited on 04/09/2022).
- [39] L. Oggioni, D. Sanchez del Rio Kandel, and G. Pariani. “Earth Observation via Compressive Sensing: The Effect of Satellite Motion”. In: *Remote Sensing* 14.2 (Jan. 12, 2022), p. 333. ISSN: 2072-4292. DOI: [10.3390/rs14020333](https://doi.org/10.3390/rs14020333). <https://www.mdpi.com/2072-4292/14/2/333> (visited on 04/09/2022).
- [40] R. Yao and Y. Zhang. “Compressive sensing for small moving space object detection in astronomical images”. In: *Journal of Systems Engineering and Electronics* 23.3 (June 2012), pp. 378–384. ISSN: 1004-4132. DOI: [10.1109/JSEE.2012.00047](https://doi.org/10.1109/JSEE.2012.00047). <http://ieeexplore.ieee.org/document/6232096/> (visited on 04/09/2022).
- [41] P. Chen et al. “Lensless Computational Imaging Technology Using Deep Convolutional Network”. In: *Sensors* 20.9 (May 6, 2020), p. 2661. ISSN: 1424-8220. DOI: [10.3390/s20092661](https://doi.org/10.3390/s20092661). <https://www.mdpi.com/1424-8220/20/9/2661> (visited on 04/09/2022).
- [42] A. Zomet and S. Nayar. “Lensless Imaging with a Controllable Aperture”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR’06)*. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR’06). Vol. 1. New York, NY, USA: IEEE, 2006, pp. 339–346. ISBN: 9780769525976.

- DOI: 10.1109/CVPR.2006.175. <http://ieeexplore.ieee.org/document/1640778/> (visited on 04/09/2022).
- [43] X. Zhang et al. “Infrared Image Super Resolution by Combining Compressive Sensing and Deep Learning”. In: *Sensors* 18.8 (Aug. 7, 2018), p. 2587. ISSN: 1424-8220. DOI: 10.3390/s18082587. <http://www.mdpi.com/1424-8220/18/8/2587> (visited on 04/09/2022).
- [44] X. Zhang et al. “MEMS-based super-resolution remote sensing system using compressive sensing”. In: *Optics Communications* 426 (Nov. 2018), pp. 410–417. ISSN: 00304018. DOI: 10.1016/j.optcom.2018.05.046. <https://linkinghub.elsevier.com/retrieve/pii/S0030401818304280> (visited on 04/09/2022).
- [45] Huaijin Chen et al. “FPA-CS: Focal plane array-based compressive imaging in short-wave infrared”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA: IEEE, June 2015, pp. 2358–2366. ISBN: 9781467369640. DOI: 10.1109/CVPR.2015.7298849. <http://ieeexplore.ieee.org/document/7298849/> (visited on 04/09/2022).
- [46] E. E. Fenimore and T. M. Cannon. “Coded aperture imaging with uniformly redundant arrays”. In: *Applied Optics* 17.3 (Feb. 1, 1978), pp. 337–347. ISSN: 2155-3165. DOI: 10.1364/AO.17.000337. <https://opg.optica.org/ao/abstract.cfm?uri=ao-17-3-337> (visited on 03/25/2022).
- [47] H. Arguello et al. “Higher-order computational model for coded aperture spectral imaging”. In: *Applied Optics* 52.10 (Apr. 1, 2013), p. D12. ISSN: 1559-128X, 2155-3165. DOI: 10.1364/AO.52.000D12. <https://opg.optica.org/abstract.cfm?URI=ao-52-10-D12> (visited on 03/25/2022).
- [48] I. Reshetouski et al. “Lensless Imaging with Focusing Sparse URA Masks in Long-Wave Infrared and Its Application for Human Detection”. In: *Computer Vision – ECCV 2020*. Ed. by A. Vedaldi et al. Cham: Springer International Publishing, 2020, pp. 237–253. ISBN: 9783030585297. DOI: 10.1007/978-3-030-58529-7_15.
- [49] M. J. Cieślak, K. A. Gamage, and R. Glover. “Coded-aperture imaging systems: Past, present and future development – A review”. In: *Radiation Measurements* 92 (Sept. 2016), pp. 59–71. ISSN: 13504487. DOI: 10.1016/j.radmeas.2016.08.002. <https://linkinghub.elsevier.com/retrieve/pii/S1350448716301524> (visited on 03/25/2022).

- [50] G. Daniel and O. Limousin. “Extended sources reconstructions by means of coded mask aperture systems and deep learning algorithm”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1012 (Oct. 2021), p. 165600. ISSN: 01689002. DOI: [10 . 1016 / j . nima . 2021 . 165600](https://doi.org/10.1016/j.nima.2021.165600). [https : //linkinghub.elsevier.com/retrieve/pii/S0168900221005854](https://linkinghub.elsevier.com/retrieve/pii/S0168900221005854) (visited on 03/25/2022).
- [51] Y. Pfeffer. “A Micro-Mirror Array based System for Compressive Sensing of Hyperspectral Data”. In: 2010.