On the Design of a Sparsifying Dictionary for Compressive Image Feature Extraction

Marco Trevisi, Ricardo Carmona-Galán, Jorge Fernández-Berni, Ángel Rodríguez-Vázquez
Instituto de Microelectrónica de Sevilla (IMSE-CNM)
CSIC-Universidad de Sevilla, Spain.
E-mail: {trevisi, rcarmona, berni, angel}@imse-cnm.csic.es

Abstract— **Compressive sensing is an alternative to Nyquist-rate sampling when the signal to be acquired is known to be sparse or compressible. A sparse signal has a small number of nonzero components compared to its total length. This property can either exist either in the sampling domain, i. e. time or space, or with respect to a transform basis. There is a parallel between representing a signal in a compressed domain and feature extraction. In both cases, there is an effort to reduce the amount of resources required to describe a large set of data. A given feature is often represented by a set of parameters, which only acquire a relevant value in a few points in the image plane. Although there are some works reported on feature extraction from compressed samples, none of them considers the implementation of the feature extractor as a part of the sensor itself. Our approach is to introduce a sparsifying dictionary, feasibly implementable at the focal plane, which describes the image in terms of features. This allows a standard reconstruction algorithm to directly recover the interesting image features, discarding the irrelevant information. In order to validate the approach, we have integrated a Harris-Stephens corner detector into the compressive sampling process. We have evaluated the accuracy of the reconstructed corners compared to applying the detector to a reconstructed image.**

Keywords—compressive sampling; image feature extraction; sparse representation;

I. INTRODUCTION

Compressive sensing is a theoretical framework that provides the support for the reconstruction of undersampled signals. If the original signal can be sparsely represented in some domain, like natural images [1], then it is possible to recover it from a much smaller number of samples than that indicated by Nyquist-Shannon theorem. Therefore, if X is a collection of either spatial or temporal samples of a signal, a set of measurements Y can be obtained through:

$$Y = \Phi X \tag{1}$$

where Φ is the so-called measurement matrix. If signal X can be sparsely represented by coefficients α in a different base, we can rewrite Eq. (1) as:

$$Y = \Phi \Psi \alpha \tag{2}$$

where Ψ is the sparsifying dictionary. The key requirement for achieving a successful reconstruction, i. e. approximating X given the much smaller set Y, is the sparsity of the input signal. The way in which the samples of the original signal are linearly combined to from the compressed samples is encoded into the measurement matrix. In other words, matrix Φ contains the compressive strategy. On the other side, the sparsifying dictionary Ψ transforms the original signal into a sparse version, referred to a transform basis. The inverse problem defined by Eq. (1) is undetermined as long as the elements in Y are fewer than those in X. Although underdetermined problems are considered ill-posed, as there is no univocal solution to them, compressive sensing theory can lead to a unique solution by the means of convex optimization. The condition for this to be achieved is that the product of Φ and Ψ holds the restricted isometry property (RIP) [2].

In this paper, we are evaluating the incorporation of feature extraction right at the sparsifying dictionary. The initial hypothesis is that if the relevant information is contained in a small number of pixels, i. e. those where a particular feature scores a noticeable value, the reconstruction of the features of an image can be realized on a smaller number of compressed samples than the reconstruction of the original image. This will seamlessly integrate feature extraction with the process of sampling and eliminating the need further processing after reconstruction. There are some examples in literature of dedicated image sensors implementing compressive sampling [3] [4], but they are mainly concerned with the generation of the samples at the focal plane. Concerning compressive feature detection, work on the characteristics of the measurement matrix has been reported to provide good results in detecting features [5]. Other studies concentrate in the propagation of properties from the original image to the compressed samples [6] [7]. Others apply the concept of compressive sampling at higher cognitive tasks, like object classification, by focusing on the reconstruction [8] [9]. To the best of our knowledge there are no previous attempts to generate compressed samples in a way that image features can be directly extracted with a standard reconstruction algorithm.

II. DESIGN OF THE SPARSIFYING DICTIONARY

Representing a signal in a transform basis involves the choice of a dictionary. In a sparse representation, most of the information contained in the signal is represented by only a few coefficients in the transform domain. The sparsifying dictionary contains a set of elements that are employed to represent each signal by means of linear combinations. When this approach is applied to image sensing, the dictionaries employed are usually based on the wavelet and cosine

transforms, as they are the most suitable for image compression [10]. Knowledge regarding the kind of image to be sampled or regarding the content that one might be looking for can be used to create a sparsifying dictionary. The choice of dictionary that extracts features does not aim to represent a compressed form of the whole informational content of the sampled image. It rather focuses on the features of interest so that a standard reconstruction algorithm can process and recover only the relevant information contained in the image.

In principle, we suppose that adjusting the sparsifying dictionary in order to reconstruct only a given set of features will reduce the amount of information that a reconstruction problem must handle. This will lead to faster reconstruction time, and the need of a smaller number of compressed samples. The sparsifying dictionary can be seen as a mask applied at the focal plane of a dedicated sensor implementing a compressive sensing strategy. To demonstrate this statement we have employed the Harris-Stephens corner detection algorithm [11]. This method is based on the comparison of an image patch with their neighboring overlapping patches, in terms of the sum of squared differences:

$$E(i,j) = \sum_{u,v} w(u,v)|I(u+i,v+j) - I(u,v)|^2 \quad (3)$$

where $w(u,v)$ are the weights over the window where the image intensity is evaluated, $I(u,v)$ are the image intensity values in this window and $I(u+i,v+j)$ are the image intensity values on a window that is shifted i pixels in the vertical direction and j pixels in the horizontal. Usually, the (i,j) pairs in which the difference is evaluated are: $(1,0)$, $(-1,0)$, $(0,1)$ and $(0,-1)$. This means that a flat region will render small differences in all directions, an edge will render a small change in on direction and a noticeable large one in the other, and a corner yields large changes in both directions.

Eq. (3) can be approximated by using Taylor expansion, and then written in a matrix form:

$$E(i,j) = \begin{bmatrix} i \\ j \end{bmatrix} A [i \quad j] \quad (4)$$

where A contains the products of the derivatives evaluated at the central pixel of the window $w(u,v)$. If a circularly weighted window is employed to have an isotropic response:

$$A = \begin{bmatrix} I_i^2 & I_i I_j \\ I_i I_j & I_j^2 \end{bmatrix} \quad (5)$$

where I_i and I_j are the partial derivatives of the image intensity in the i and j directions, respectively. Hence, the presence of a corner is reported by two large eigenvalues of matrix A. The response of every pixel can be defined as:

$$R = \mathrm{Det}(A) - k[\mathrm{Tr}(A)]^2 \quad (6)$$

where k is employed to tune the sensitivity to changes of the algorithm. Usual values are in the 0.04-0.15 range. In order to evaluate this response we need to compute the partial derivatives of the image intensity at every single pixel, what can be done with the help of the following masks:

$$d_i = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{and} \quad d_j = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (7)$$

which represent the partial derivatives in the vertical and horizontal directions. Therefore:

$$I_i = I * d_i \quad \text{and} \quad I_j = I * d_j \quad (8)$$

In order to test the procedure with an implementation of the NESTA algorithm [12], we have to convert images of size $M \times N$ into column vectors of size $N \times 1$. We will do that by rearranging image columns into one single column. Therefore the first M components of the column-vector image I^c are the first column of the matrix image I. The next M components are the second column, and so on. By defining these $M \times M$ matrices:

$$m_i = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & -1 & 0 \end{bmatrix} \quad (9)$$

and:

$$m_j = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 1 \end{bmatrix} \quad (10)$$

we can write these $MN \times MN$ matrices:

$$D_i = \begin{bmatrix} m_i & m_i & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ m_i & m_i & m_i & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & m_i & m_i & m_i & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & m_i & m_i & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & m_i & m_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & m_i & m_i & m_i \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & m_i & m_i \end{bmatrix} \quad (11)$$

$$D_j = \begin{bmatrix} \mathbf{0} & m_j & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ -m_j & \mathbf{0} & m_j & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -m_j & \mathbf{0} & m_j & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -m_j & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & m_j \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -m_j & \mathbf{0} \end{bmatrix} \quad (12)$$

where each $\mathbf{0}$ is a $M \times M$ matrix of zeros, and therefore Eq. (8) can be written as a matrix product:

$$I_i^c = D_i I^c \qquad \text{and} \qquad I_j^c = D_j I^c \quad (13)$$

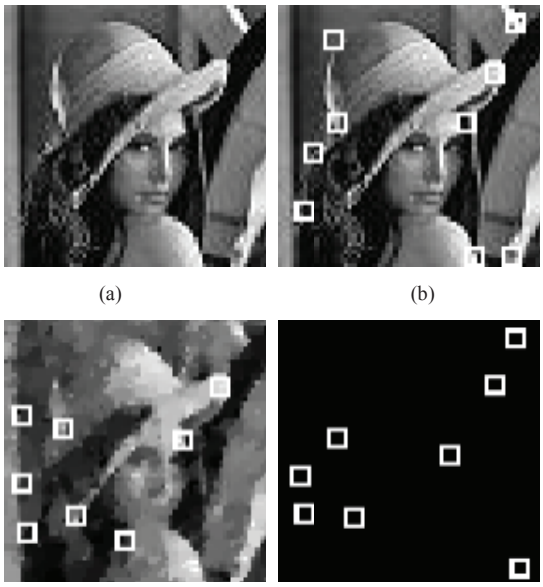
These matrices, D_i and D_j can be employed as a sparsifying dictionary, Ψ, as they hold the RIP when multiplied to the measurement matrix Φ. We can therefore introduce the computation of the partial derivatives of the image intensity right at the sampling point. Then, we can use the NESTA algorithm to reconstruct directly the derivatives instead of the rendering the image and then computing the derivatives.

III. Feature Reconstruction and Evaluation

In order to analyze the benefits of using the derivative masks D_i and D_j to recover a set of corners from a compressed-sampled image we have devised an experiment using a 64 × 64 grayscale picture of Lena (Fig. 1(a)). In order to establish the ground truth, Harris corners have been detected over the original image (Fig. 1(b)). As an illustration of the type of images that we will be obtaining, Fig. 1(c) displays the reconstruction of the original Lena image by using 1024 compressed samples —being 4096 the total number of pixels of the original image. Fig. 1(d) shows the reconstructed corners directly from 1024 compressed samples that have been obtained using the sparsifying dictionaries that contain the derivative masks. In order to evaluate the results we will take into account the distance between the original corners and the ones obtained by the different methods. We have defined an average distance given by:

$$\bar{d} = \frac{1}{N}\sum_{i=1}^{N} \min_{\substack{p_i \in P_0 \\ p_j \in P_e}} \|p_i - p_j\| \quad (14)$$

where p_i one of the N corners belonging to P_0, which is the set of corners extracted from the original image, i. e. the ground truth. The contribution to the average distance is given by the closest p_j, a corner belonging to the set of estimated points P_e. We will be counting the number of false positives and false negatives as well.



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

Fig. 1. (a) Original 64 × 64 image; (b) Ground truth, i. e. Harris corners detected over the original image; (c) Harris corners detected over the reconstructed image; (d) Corners directly extrated from compressed samples.

We have tested the extraction of Harris corners by direct reconstruction for different numbers of compressed samples. In Fig. 2 we can see how the average distance to the ground-truth corners decays with the number of samples. It can be appreciated also that direct reconstruction (blue circles) is more accurate than performing Harris corner detection over the reconstructed image (red crosses), especially for a small set of compressed samples.

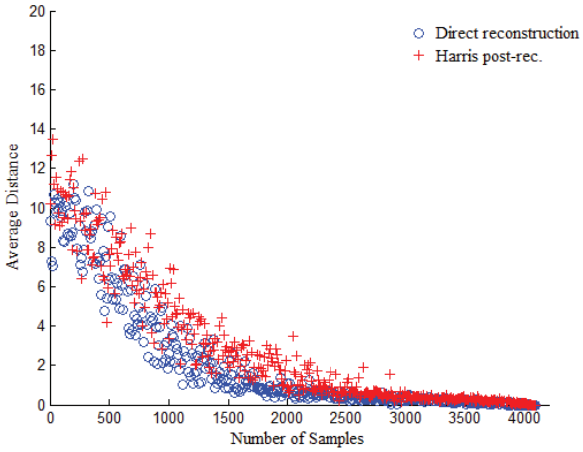


Fig. 2. Average distance vs. number of samples.

This information would be incomplete unless we take into account the number of false positives and false negatives. While the direct reconstruction of Harris corners seems to perform better than extracting Harris corners on the already reconstructed image, if it were to produce more false results it would be overall worse. Hence, Fig. 3 plots the number of false negatives, i. e. corners that were present in the ground-truth image but are missing in the reconstructed version, vs. the number of compressed samples. The number of false negatives is slightly smaller for the direct reconstruction. Fig. 4 displays the number of false positives, i. e. corners that were not in the ground-truth image but are present in the reconstructed version, vs. the number of compressed samples again. The number of false positives is also slightly smaller for the direct reconstruction. In fact, even though the graphics are somehow cluttered, direct reconstruction of the derivatives of the original image leads to an average of 11% less false negatives and 8 % less false positives. We can conclude that not only the location of the corners is more accurately determined when using the derivative masks as sparsifying dictionaries, but this method also leads to better overall results by decreasing the number of false positives and false negatives.

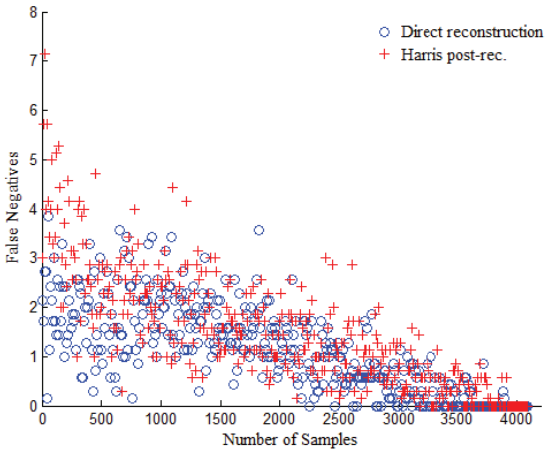


Fig. 3. False negatives vs. number of samples

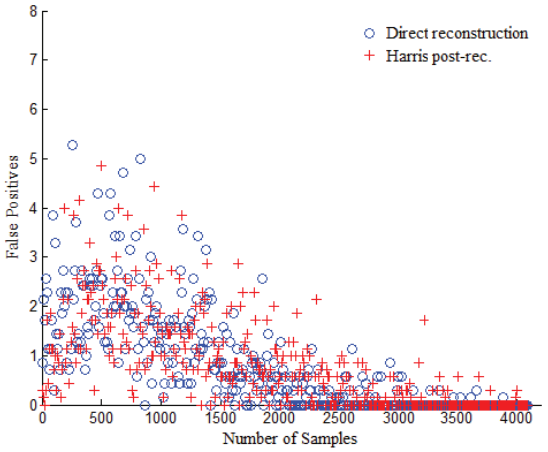


Fig. 4. False positives vs. number of samples

The downside of applying this method is that, to actually locate Harris-Stephens corners, the two derivatives are necessary. Fortunately, this sends the computational load to the reconstruction side, alleviating the work at the sensor plane.

IV. Conclusions

Image description based on features can be naturally inserted into the compressive sensing framework. We have successfully generated a set of compressed samples from which features can be directly reconstructed, without having to recreate the original image first. As the relevant points in a feature-based description are much fewer than the number of pixels of the image, the number of compressed samples that are required for feature extraction under a targeted accuracy is going to be smaller. Experimental evidence has been obtained by simulation. The direct reconstruction of Harris corners from a set of compressed samples generated with derivative masks as sparsifying dictionaries yields better results. The study the characteristics of generic a set of features to be translated into a sparsifying dictionary would be the subject of further research.

Acknowledgment

This work has been funded by the Spanish Government through projects TEC2012-38921-C02 MINECO (European Region Development Fund, ERDF/FEDER), IPT-2011-1625-430000 MINECO and IPC-20111009 CDTI (ERDF/FEDER), by Junta de Andalucía through project TIC 2338-2013 CEICE and by the Office of Naval Research (USA) through grant N000141410355.

References

[1] J. Romberg. “Imaging via Compressive Sampling”. *IEEE signal Processing Magazine*, Vol. 25, No. 2, pp. 15-20. March, 2008.

[2] R. G. Baraniuk, V. Cevher, M. B. Wakin. “Low-Dimensional Models for Dimensionality Reduction and Signal Recovery: A Geometric Perspective”. *Proc. of the IEEE*, Vol. 98, No. 6, pp. 959-971, Jun 2010.

[3] V. Majidzadeh et al. “A (256x256) Pixel 76.7mW CMOS Imager /Compressor Based on Real-Time In-Pixel Compressive Sensing”. *IEEE Int. Symp. on Circuits and Systems* (ISCAS), pp. 2956-2959, May 2010.

[4] Y. Oike, A. El Gamal. “CMOS Image Sensor With Per-Column ΣΔ ADC and Programmable Compressed Sensing”. *IEEE Journal of Solid-State Circuits*, Vol. 48, No. 1, pp. 318 - 328, Jan. 2013.

[5] A. Elenyan, K. Kose, A. Cetin. “Image Feature Extraction Using Compressive Sensing”. *Image Processing and Communications Challenges 5*, pp. 177-184, Springer. 2014.

[6] M. Davenport, et al. “The Smashed Filter for Compressive Classification and Target Recognition”. *Proc. SPIE, Computational Imaging V*, Vol. 6498, pp. 64980H , San Jose, California, January 2007.

[7] B. Gardiner. *Compressive Image Feature Extraction by Means of Folding*. Master thesis. Massachusetts Institute of Tech. June 2012.

[8] P. Nagesh and B. Li. “A Compressive Sensing Approach for Expression-Invariant Face Recognition”. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1518-1525, June 2009.

[9] L. Liu, P. Fieguth. “Texture Classification using Compressed Sensing”. *Canadian Conf. Comp. and Robot Vision* (CRV), pp. 71-78 June 2010.

[10] M. Antonini, M. Barlaud, P. Mathieu, I. Daubechies. “Image Coding Using Wavelet Transform”. *IEEE Transactions on Image Processing*, Vol. 1, No. 2, pp. 205-220, Apr. 1992.

[11] C. Harris, M. Stephens, “A Combined Corner and Edge Detection”. *Proc. of the 4th Alvey Vision Conference*, pp. 147-151, 1988.

[12] S. Becker, J. Bobin, E. Candès. “NESTA: a Fast and Accurate First-Order Method for Sparse Recovery”. *SIAM J. Imaging Sci.*, Vol. 4, No. 1, pp. 1–39. Jan 2011.