State of the art in texture analysis and synthesis

Angélique Drémeau, Christine Guillemot, Olivier Le Meur, Emilie Bosc, Luce Morin, Muriel Pressigout, Marco Cagnazzo, Erica d'Acunto

Projet PERSEE
« Schémas Perceptuels et Codage vidéo 2D et 3D »
n° ANR-09-BLAN-0170

Livrable **D2.1** *15/10/2010*

State of the art in texture analysis and synthesis

Angélique	DRÉMEAU	IRISA
Christine	GUILLEMOT	IRISA
Olivier	LE MEUR	IRISA
Emilie	BOSC	INSA
Luce	MORIN	INSA
Muriel	PRESSIGOUT	INSA
Marco	CAGNAZZO	LTCI
Erica	D'ACUNTO	LTCI

Contents

1 Texture-based analysis **3**
1.1 2D structure extraction . . . 3
1.2 3D structure extraction . . . 4
1.2.1 Depth maps extraction . . . 4
1.2.2 Vanishing points, Planar regions . . . 7

2 Texture synthesis and inpainting **9**
2.1 Texture prediction . . . 9
2.2 Epitomes . . . 10
2.3 Graph-cut . . . 12
2.4 Diffusion equation . . . 12
2.4.1 Isotropic diffusion . . . 12
2.4.2 Anisotropic diffusion . . . 13
2.4.3 Noise reduction Applications . . . 15
2.5 Sparse representation . . . 17
2.5.1 Optimization problems . . . 17
2.5.2 Sparse decomposition algorithms . . . 22
2.5.3 Prediction based on sparse representations . . . 30
2.6 Depth-aided texture synthesis . . . 31

References **31**

1 Texture-based analysis

1.1 2D structure extraction

The issue of extracting some structures, in the sense of some relevant part of images or videos is really interesting in both the context of 2D and 3D video coding. The possibility of extracting and tracking an object can be exploited to achieve a more efficient prediction. The techniques of structure extraction include all contours, motion and regions extraction.

Here we focus on some techniques for 2D video coding for contour and motion extraction and trajectory estimation.

The paper [25] by Dubuisson and Jain presents an innovative (at the date of the publication) approach to the extraction of the contour of a moving object. The method is based on the fusion of a motion segmentation technique using image subtraction and a coulor segmentation technique based on the split-and-merge paradigm and edge information obtained from using the Canny edge detector [12]. The advantages of this method are that it can detect large moving objects no matter how complicated the background is, and it requires only three image frames that need not be consecutive provided that the moving object is entirely contained in the three frames.

Another method for tracking, in an image sequence, complex object is presented in the article by Bouthemy, Chaumette, Moreau and Marchand [11]. The approach relies on the estimation of the 2D object image motion along with the computation of the 3D object pose. The proposed method fulfills real-time constraints along with reliability and robustness requirements.

As mentioned, the structure extraction can be exploited also for the extraction and classification of motion in an image sequence basing on motion trajectories. The paper by Yang, Ahuja and Tabb [80] presents an algorithm that first perform a multiscale segmentation to generate homogeneous regions in each frame. Regions between consecutive frames are then matched to obtain two-view correspondences. Affine transformations are computed from each pair of corresponding regions to define pixel matches. Pixels matches over consecutive image pairs are concatenated to obtain pixel-level motion trajectories across the image sequence. Motion patterns are learned from the extracted trajectories using a time-delay neural network. Experimental results show that motion patterns of hand gestures can be extracted and recognized accurately using motion trajectories.

Object segmentation and trajectory estimation can be also exploited to create a successful object tracking system [57]. A new efficient motion detection algorithm referred to as the flux tensor is used to detect moving objects in infrared video without requiring background modeling or contour extraction. The flux tensor-based motion detector when applied to infrared video is more accurate than thresholding "hot-spots", and is insensitive to shadows as well as illumination changes in the visible channel. The object segmentation algorithm uses level set-based geodesic active contour evolution that incorporates the fusion of visible color and infrared edge informations in a novel manner. Touching or overlapping objects are further refined during the segmentation process using an appropriate shape based model. Multiple object tracking

is extended to handle groups of objects and occlusion events by Kalman filter-based cluster trajectory analysis and watershed segmentation.

Most of the existing salient object detection approaches detect visually conspicuous structures in images, while it can be useful also to find regions that may be important for indexing in a video database system. In [5] is described a method to automatically extract important video objects for object-based indexing. First there's a segmentation of each frame to obtain homogeneous regions in terms of color and texture. Then, there's the extraction of a set of regional and inter-regional color, shape, texture and motion features for all regions, which are classified as being important or not using SVMs trained on a few hundreds of example regions. Finally, each important region is tracked within each shot for trajectory generation and consistency check.

1.2 3D structure extraction

1.2.1 Depth maps extraction

Recently the 3D video coding is assuming a relevant importance in the contest of video processing and coding. Most of the times when we refer to 3D content, we are considering stereo or multiview sequences with depth maps.

The extraction of depth maps is a very important issue for the context of 3D video processing and coding. Here we try to resume some relevant or innovative techniques for depth maps extraction that have been developed in the last years.

A relative depth layer extraction for monoscopic video using multi-line filters and a layer selection algorithm is presented in the paper [14]. Main ideas are to extract multiple linear trajectory signals from videos and to determine their relative depths using the concept of motion parallax. The proposed superficial line model used for detecting slow moving objects provides sufficient taps within few frames to reduce frame buffer, while the closest-hit line model used for detecting fast motion objects provides few enough taps to prevent blurring. To increase the correctness of layer map, three-level layer map codecision is used to compensate low texture region defect.

Obtaining an accurate and precise depth map is the ultimate goal for 3D shape recovery.

For depth map estimation, one of the most vital parts is the initial selection of the focus measure and processing the images with the selected focus measure. Although, many focus measures have been proposed in the literature but not much attention has been paid to the factors affecting those focus measures as well as the manner the images are processed with those focus measures. In the paper by Malika and Choi [52], for accurate calculation of depth map, the authors consider the effects of illumination on the depth map as well as the selection of the window size for application of the focus measures. The resulting depth map can further be used in techniques and algorithms leading to recovery of three-dimensional structure of the object which is required in many high-level vision applications. It is shown that the illumination effects can directly result in incorrect estimation of depth map if proper window size is not selected during focus measure computation. Further, it is shown that the images need some kind of pre-processing to enhance the dark regions and shadows in the image.

For this purpose, an adaptive enhancement algorithm is proposed for pre-processing. Without such pre-processing for image enhancement and without the use of proper window size for the estimation of depth maps, it is not possible to obtain the accurate depth map.

Three-dimensional shape recovery from one or multiple observations is a challenging problem of computer vision. In this paper by Malika and Choi [51], it is presented a new focus measure for the estimation of a depth map using image focus. This depth map can subsequently be used in techniques and algorithms leading to the recovery of a three-dimensional structure of the object, as required by high level vision applications. The proposed focus measure has shown robustness in the presence of noise as compared to the earlier focus measures. This new focus measure is based on an optical transfer function implemented in the Fourier domain. The results of the proposed focus measure have shown drastic improvements in estimation of a depth map, with respect to the earlier focus measures.

With the development of 3DTV, the conversion of existing 2D videos to 3D videos becomes an important component of 3D content production. One of the key steps in 2D to 3D conversion is how to generate a dense depth map. In the paper [38], it is presented a novel depth extraction method based on motion and geometric information for 2D to 3D conversion, which consists of two major depth extraction modules, the depth from motion and depth from geometrical perspective. The H.264 motion estimation result is utilized and cooperates with moving object detection to diminish block effect and generates a motion-based depth map. On the other hand, a geometry-based depth map is generated by edge detection and Hough transform. Finally, the motion-based depth map and the geometry-based depth map are integrated into one depth map by a depth fusion algorithm.

The emergence of three dimensional (3D) video applications, based on Depth Image Based Rendering (DIBR) has brought up more requirements of bandwidth, due to the need of depth information. This additional bandwidth requirement need to be tackled to enable the widespread of 3D video applications based on DIBR. Exploiting visual correlations between the colour image and the depth image, in depth image coding, will reduce the requirement of high bandwidth required to transmit the additional depth information. In the paper by De Silva, Fernando and Yasakethu [64], an object based depth image coding technique is presented which is suitable for low bit rate 3D-TV applications that are based on Depth Image Based Rendering. The proposed method achieves at most 50% bit rate reduction at low bit rates.

A novel dense depth map estimation algorithm is proposed by Çigla and Alatan [40] in order to meet the requirements of N-view plus N-depth representation, which is one of the standardization efforts for the upcoming 3D display technologies. Hence, extraction of multiple depth maps is achieved from multi-view video. Starting from the piecewise planarity assumption of the scene, estimation of 3D structure of the patches, obtained through color-based over-segmentation, is achieved by plane- and angle-sweeping for every view independently. Markov Random Field (MRF) modeling is utilized for each view in pixel-wise manner in order to relax and refine the estimated planar models while incorporating visibility and consistency constraints. In this algorithm, the fusion of multiple depth maps is performed by updating belief values on

the observed nodes based on depth and color consistency during the refinement step. The proposed method handles untextured surfaces, as well as depth discontinuities at object boundaries. The experimental results illustrate reliability and the robustness of the proposed algorithm for different type of scenes.

Experimental prototypes of entire 3DTV processing chains have been demonstrated successfully during the last few years. In this context the paper by Kauff, Atzpadin and Fehn [44], discusses an advanced approach for a 3DTV service, which is based on the concept of video-plus-depth data representations. It particularly considers aspects of interoperability and multi-view adaptation for the case that different multi-baseline geometries are used for multi-view capturing and 3D display. Future 3DTV displays will use different techniques for 3D reproduction, such as standard stereo systems with glasses, tracked auto-stereoscopic single-user displays supporting HMP viewing or auto-stereoscopic multi-view displays allowing several users to watch the same 3D scene from different perspectives. Furthermore, future 3DTV systems should be backwards compatible to existing 2D digital broadcast services. In this context, the estimation of suitable depth maps from stereo or multi-baseline systems is certainly one of the most challenging tasks. The only geometrical restriction on the multi-baseline configuration is that the cameras are arranged along an almost straight or slightly curved line and that they converge to one common 3D point. First a rectification process is performed: a rotation of the two cameras within a stereo pair is used in order to obtain a pair of rectified cameras with parallel optical axes.

After the rectification a disparity matching is applied to each pair of rectified cameras using a hybrid-recursive-matching (HRM) algorithm.

After checking and correcting the mismatching, a post processing analyzes the failures due to occlusions and ambiguities.

Starting from a disparity map it is possible to calculate a depth map by using geometrical relations between the rectified cameras.

So an alternative method to exploit the correlation between different cameras in order to provide depth information is using the disparity between the cameras. In the paper by Daribo, Kaaniche, Miled, Cagnazzo and Pesquet-Popescu [20] the authors propose a dense motion/disparity estimation algorithm, designed to replace the classical temporal/inter-view unit within the MVC extension of H.264/MPEG- 4 AVC, which uses a block-based motion/disparity estimation. The dense disparity estimation problem has been formulated by Miled, Pesquet and Parent[55] as a convex programming problem within a global variational framework. Numerical studies have shown that variational-based disparity estimation methods are among the most powerful techniques meanwhile preserving the depth discontinuities. A quantitative comparison with results from other stereo algorithms, shows that this approach is competitive with state-of-the-art methods. This naturally motivates the choice to integrate this global convex variational framework, along with the motion and disparity estimation, within a multiview video coder. The retained variational convex optimization approach generates smooth displacement vector fields with ideally infinite precision. The estimation problem is solved through the minimization of a global objective function, which is the sum of Displaced Frame Differences (DFD), under various convex constraints. The problem of coding the resulting dense motion and disparity vectors is

a challenging issue because of the high bitrate needed to transmit such fields. What is proposed in [20], is to reduce the bitrate needed for the coding of the dense displacement fields by preforming a RD segmentation and coding. This is achieved by optimizing a Lagrangian cost function which takes into account the accuracy and the coding cost of the displacement field. The dense estimation framework followed by the segmentation step can replace therefore the block-based motion/disparity estimation stage in the MVC extension.

In the paper by Alvarez, Deriche, Sanchez and Weickert [2], it is presented an energy based approach to estimate a dense disparity map from a set of two weakly calibrated stereoscopic images while preserving its discontinuities resulting from image boundaries. First it is derived a simplified expression for the disparity that allows to estimate it from a stereo pair of images using an energy minimization approach. It is assumed that the epipolar geometry is known, and this information is included in the energy model. Discontinuities are preserved by means of a regularization term based on the Nagel-Enkelmann operator. The associated Euler-Lagrange equation of the energy functional is investigated, and the solution of the underlying partial differential equation (PDE) is found using a gradient descent method. The resulting parabolic problem has a unique solution. In order to reduce the risk to be trapped within some irrelevant local minima during the iterations, it is used a focusing strategy based on a linear scale-space.

1.2.2 Vanishing points, Planar regions

Motivations Taking into account geometrical cues human observers are sensitive to can help to better representing an object in 3D. Incorporating constraints extracted from this estimated geometry should consequently improve the perceptual quality of the rendered image, as suggested in [65]. As exemples, 3D lines, planes, vanishing points can be extracted. The computer vision community has proposed many methods to extract some geometrical features in field such as augmented reality, for example.

Commonly used methods for 3D cues extraction Deducing 3D structure from image sequences has been a central problem in computer vision for many years. 3D reconstruction needs correspondences between features in consecutive frames. Different approaches can help finding such correspondences in a relative reliable way. In optical-flow-based methods, a velocity vector is computed for each pixel. In feature-based methods, primitives such as lines and points are extracted and matched to corresponding primitives along the frames of the sequence.
This problem has been addressed in various fields: it is known that the knowledge of the scene structure can help the rendering process by the addition of appropriate 3D constraints. Those constraints are meant to ensure a certain consistency of the rendered scene and improve the perceptual quality. For example, the computer vision community uses methods that extract scene topology information in augmented reality applications (analysis of the orientation for re-illumination, planes structure, presence of vanishing points, detection of occlusions). A large number of examples of the use of 3D cues extraction for improving image quality can be found in the literature, in

various fields. In what follows, we refer some pieces.

The necessity to have a knowledge of the scene structure is imperative when dealing with augmented reality, say, for the superimposition of images onto target planes in the scene. Smith et al. addressed this problem in [65] and proposed methods to improve the rendering of augmented scenes. Three types of augmented tasks are presented:

- the planar surface augmentation, where a 2D homography is computationally determined. Here, some artefacts are avoided by identifying and tracking occlusion boundaries, when the augmented plane is not a foreground plane.The method for surface augmentation is based on three main technologies: bundle adjustment, geometrically constrained optimization and line tracking.
- the connected planes augmentation, where the line of intersection of the two planes is extrapolated from close views. Here, for perceptual purposes, this line must be accurately located. A temporal extrapolation is also needed so that the line appear rigidly fixed in all frames (the constraint used for accuracy is the parallelism constraint).
- the 3D augmentation, where reference points are determined to augment a 3D object within the scene. For instance, the authors wish to add a 3D box on the roof of a building. They need a 3D coordinate system that they extract by using a perspective camera and tracking the vanishing points of three sets of parallel lines. The 3D intersection points of the lines are estimated, then an affine system can be computed.

A few works about how and what 3D information can be retrieved are now presented. This is not exhaustive as 3D reconstruction deals with many applications domains, data and methods.

The quality of reconstruction can be improved via use of planarity constraints according to [66] and [1]. In [1], the proposed method is based on the extended Kalman filter (EKF) and experiments were performed on synthetic and real imagery. It shows an improvement in accuracy and stability to noise. The method models planar structure without any assumption about the initial orientation of the plane with respect to the camera. The method can be applied in real time applications.

In [30], an approach is proposed to embed planar structural components from points lying on the same plane. It allows the discovery of planes in a sequence of images. The authors confirm a gain in time of computation (and it can be used in real-time application), of scalabitlity and a higher level of description of the scene. The method is based on a visual Simultaneous Localization and Mapping (SLAM) system. SLAM is a technique used by robots and autonomous vehicles to build up a map within an unknown environment (without a priori knowledge) and update the map while keeping track of their current location. With SLAM techniques, various features can be

estimated. Though these methods can not be directly applied to video coding, they can be helpful to retrieve 3D information from multiviews.

In [48], a quasi-dense approach to 3D surface model acquisition from uncalibrated images is proposed. The algorithm includes the computation of correspondence information and geometry from a subsample of points of interest. The modelling requires few largely separated images. This method delivers a high level of quality of 3D points reconstruction on which a surface can be reconstructed.
In [**?**], the method is supposed to avoid visual distortions through the use of 3D scene characteristics inherent in image sequences(detection of edges, vanishing points and vanishing lines). It is based on a robust estimation of such 3D structural properties.

Conclusion The knowledge of the scene structure helps in the improvement of the rendered images in fields such as augmented reality. Methods used in augmented reality are reliable because the constraints are chosen to respect perceptual cues. Many tools are used to extract and enhance the 3D model of a scene. Those tools can inspire new methods in 3D video compression that preserve the topology: the exploitation of the multi view data can lead to the extraction of 3D constraints that could be used during the coding process or the rendering process for example. In the image coding community, the idea already came out with [**?**]. MVD data provide a large amount of information from which 3D cues, special features, can be extracted and exploited. It is worth adding appropriate geometric contraints (parallelism constraint or detection of vanishing points, consistency over frames, respect of occlusion areas, planarity constraint) to 3D video codecs in order to improve the perceptual quality of the rendered views, and to the rendering process.

2 Texture synthesis and inpainting

2.1 Texture prediction

Spatial and temporal texture prediction can be regarded as particular and simple texture synthesis techniques. Along the temporal dimension, the motion information allows the selection of one predictor (a texture patch) along the motion trajectory of objects in the scene. In spatial prediction, a set of directional predictors, which propagate neighbouring pixel values along specified directions, is often used. These predictors are well suited if the block to be predicted contains structures with orientations close to the ones defining the predictors.

In particular we focus on texture prediction techniques developed for H.264.

Intra prediction is an effective method for reducing the coded information of an image or an intra frame within a video sequence. The conventional method today is to create a sample predictor block by extrapolating the reconstructed pixels surrounding the target block to be coded. The sample predictor block is subtracted from the target block and the resulting residual coded using transformation, quantization and entropy coding. This is an effective method for sample predictor block creation in

most sequences. However the extrapolation method is not able to represent sample prediction blocks with complex texture. Furthermore, pixels that are far from the surrounding pixels are usually badly predicted. In the paper by Tan, Boon and Suzuki [67], a new method for sample predictor creation by template matching in a region of reconstructed pixels is presented.

Efficient intra prediction is an important aspect of video coding with high compression efficiency. H.264/AVC applies directional prediction from neighbouring pixels on an adjustable block size for local decorrelation. In the paper by Ballè and Wien [3], it is presented an extended prediction scheme in the context of H.264/AVC that comprises two additional prediction methods exploiting self-similar properties of the encoded texture. A new macroblock type is implemented, allowing for exible selection of the available prediction methods for sub-partitions of the macroblock. Depending on the content of the encoded video sequence, substantial gains in rate-distortion performance are achieved.

Texture prediction techniques can also be extended to the case of 3D video coding by combining the texture prediction and the depth maps approaches. In the paper [49] authors developed a smoothed reference inter-layer texture prediction mode for bit depth scalability based on the Scalable Video Coding extension of the H.264/MPEG-4 AVC standard. With this approach, the base layer encodes an 8-bit signal that can be decoded by any existing H.264/MPEG-4 AVC decoder and the enhancement layer encodes a higher bit depth signal (e.g. 10/12-bit) which requires a bit depth scalable decoder. The approach presented uses base layer motion vectors to conduct motion compensation upon enhancement layer reference frames. Then, the motion compensated block is tone mapped and summed with the co-located base layer residue block prior to being inverse tone mapped to obtain a smoothed reference predictor. In addition to the original inter-/intra-layer prediction modes, the smoothed reference prediction mode enables inter-layer texture prediction for blocks with inter-coded co-located block. The proposed method is designed to improve the coding efficiency for sequences with non-linear tone mapping.

2.2 Epitomes

The epitome of a video sequence is a spatially or temporally compact representation of the video that retains the video's essential textural, shape, and motion components. The video epitome is a three-dimensional construct that can represent the video in both a spatially and temporally compact form.

The epitomes can be considered as a simple appearance and shape model. The epitome model was first introduced for single images: the epitome of an image is its miniature, condensed version containing the essence of the textural and shape properties of the image. The size of the epitome is considerably smaller than the size of the image or object it represents, but the epitome still contains most constitutive elements needed to reconstruct the image. A collection of images often shares an epitome, e.g., when images are a few consecutive frames from a video sequence, or when they are photographs of similar objects. A particular image in a collection is defined by its epitome and a smooth mapping from the epitome to the image pixels.

In the paper by Jojic, Frey and Kannan[42], when the epitomic representation is used within a hierarchical generative model, appropriate inference algorithms can be derived to extract epitome from a single image or a collection of images and at the same time perform various inference tasks, such as image segmentation, motion estimation, object removal and super-resolution.

As mentioned, "epitomes" were introduced as patch-based probability models that are learned by compiling together a large number of examples of patches from input images. Moreover epitomes can be used to model video data. In the paper by Cheung and Frey[17] there are described significant computational speedups that can be incorporated into the epitome inference and learning algorithm. In the case of videos, epitomes are estimated so as to model most of the small spacetime cubes from the input data. Then, the epitome can be used for various modelling and reconstruction tasks. Besides computational efficiency, an interesting advantage of the epitome as a representation is that it can be reliably estimated even from videos with large amounts of missing data.

Epitomes can be used to achieve a high reduction of transmission bandwidth and memory space for images by factoring their repeated content as shown in the paper by Wang, Wexler, Ofek and Hoppe [73]. A transform map and a condensed epitome are created such that all image blocks can be reconstructed from transformed epitome patches. The transforms may include affine deformation and color scaling to account for perspective and tonal variations across the image. The factored representation allows efficient random-access through a simple indirection, and can therefore be used for real-time texture mapping without expansion in memory. This scheme is orthogonal to traditional image compression, in the sense that the epitome is amenable to further compression such as DXT.

The epitomes can be efficient method to perform a good reconstruction in the case of dropped frames in a network, to realize denoising and impainting techniques, but epitomes can be applied for labelling problems. In the paper by Warrell and Prince[74] is considered the problem of parsing facial features from an image labelling perspective. The authors use an Adaboost-based unary classifier, and develop a family of priors based on 'epitomes' which are shown to be particularly effective in capturing the non-stationary aspects of face label distributions.

It is also possible to realize an epitome tree model which combines the epitome with a tree structured belief network prior [75]. The authors consider three existing priors, and show how each can be extended using the epitome. The simplest prior assumes patches of labels are drawn independently from either a mixture model or an epitome. Next we investigate a 'conditional epitome' model, which substitutes an epitome for a conditional mixture model. Finally, we develop an 'epitome tree' model. Each model is combined with a per pixel classifier to perform segmentation. In each case, the epitomized form of the prior provides superior segmentation performance, with the epitome tree performing best overall.

2.3 Graph-cut

2.4 Diffusion equation

The diffusion equation is a partial differential equation which describes density fluctuations in a material undergoing diffusion.
Results given in this part (see figure 2 and 3) are obtained by our own software, that is available for research purpose on `http://www.irisa.fr/temics/staff/lemeur/`.

2.4.1 Isotropic diffusion

The isotropic diffusion directly stems from the heat equation, given below:

$$\begin{cases} I_{(t=0)} = \widetilde{I} \\ \frac{\partial I}{\partial t} = \Delta I \end{cases} \tag{1}$$

where $\widetilde{I}$ is a degraded version of the original picture I.
Linear isotropic diffusion not only smoothes noise but also simultaneously blurs features such as edges. The isotropic filtering behaves as a low-pass filtering suppressing high frequencies in the image I. For many applications, the blur effect is not acceptable. The need to find more complex and more adpative filtering methods has quickly appeared.
To overcome the aforementionned limitations of linear isotropic diffusion, a first idea was to propose a non-linear extension of the heat equation. Perona and Malik [59] were the first to make this kind of extension. This extension relies on the re-formulation of the heat equation:

$$\frac{\partial I}{\partial t} = \Delta I = div\left(\nabla I\right) \tag{2}$$

where $div(.)$ is the divergence operator.
Perona and Malik [59] proposed to add a function $g(.)$ (called diffusivity function) in order to control the amount of diffusion. The new diffusion equation was then given by:

$$\frac{\partial I}{\partial t} = div\left(g(|\nabla I|^2)\nabla I\right) \tag{3}$$

where the function $g(.)$ is bounded in $[0, 1]$ and is a decreasing function vanishing on edges (high gradients) and close to 1 on flat regions (low gradients). Therefore, the diffusion process is lessened (or stopped) near the edges and is strong on homogenous areas.
The diffusivity functions proposed by Perona and Malik have the following expressions:

$$\begin{array}{ll} g(s^2) = \frac{1}{1+\frac{s^2}{\lambda^2}} & \lambda > 0 \\ g(s^2) = exp(-\frac{s^2}{\lambda^2}) & \lambda > 0 \end{array} \tag{4}$$

Another diffusivity function was proposed by Charbonnier [15]:

$$g(s^2) = \frac{1}{\sqrt{1+\frac{s^2}{\lambda^2}}} \quad \lambda > 0 \tag{5}$$

where λ is a fixed gradient threshold separating low contrast and high contrast diffusion areas. The optimal value of this threshold depends on the problem.

In order to be more robust to noise and to improve the diffusion process, some authors replace the diffusivity function $g((|\nabla I|^2))$ by $g((|\nabla I * G_\sigma|^2))$. The diffusion equation becomes:

$$\frac{\partial I}{\partial t} = div\left(g(|\nabla I * G_\sigma|^2)\nabla I\right) \tag{6}$$

where $G_\sigma = \frac{1}{2\pi\sigma^2} exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$ is a normalized 2D Gaussian kernel of variance σ. By convolving the gradient with a Gaussian, a smoothed version of the image I is used in the evolution process. It is then more robust to noise. Moreover, as a larger neighborhood is involved in the computation of the local image structures, the diffusion process is improved in term of local diffusion geometry.

2.4.2 Anisotropic diffusion

Non-linear isotropic diffusion presents significant advantages compared to isotropic diffusion. Indeed, although that the diffusion process is still isotropic, there is almost no diffusion when the gradient is greater than the given threshold ($|\nabla I|^2 > \lambda^2$). Edges are preserved.
A better solution is the anisotropic diffusion. Contrary to isotropic diffusion (linear or not), the anisotropic diffusion favors a given direction. Two kinds of diffusion scheme can be considered: a diffusion process based on the divergence operator and a diffusion process based on second-derivative expressions.

Divergence-based diffusion Weickert [78] proposed a generalization of divergence-based diffusion. The diffusion equation is given by:

$$\frac{\partial I}{\partial t} = div\left(\mathbf{D}\nabla I\right) \tag{7}$$

where $\mathbf{D}$ is a diffusion tensor (2×2 matrices). The diffusion tensor is used to steer the filtering process in a given direction. The principal directions of smoothing are based on the description of the local structures. The diffusion tensor D is deterined by the eigen vectors and eigen values of the structure tensor.

The structure tensor of a coordinate in the image I is a symmetric positive semi-definite 2×2 tensor given by ([77]):

$$J_p(\nabla I) = G_p * \left(\nabla I \nabla I^T\right) \tag{8}$$

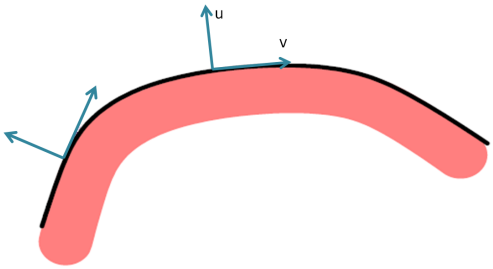


Figure 1: Position of vectors **u** and **v**.

where G_p is a Gaussian kernel of variance p. Principal axis transformation gives the eigenvectors and eigenvalues of $J_p(\nabla I)$:

$$J_p(\nabla I) = \begin{bmatrix} \mathbf{u}^* & \mathbf{v}^* \end{bmatrix} \begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix} \begin{bmatrix} \mathbf{u}^{*T} \\ \mathbf{v}^{*T} \end{bmatrix} \tag{9}$$

The eigen vectors $\mathbf{u}^*$ and $\mathbf{v}^*$ give the local image orientations with $\mathbf{u}^* = [u_x, u_y]^T$. The eigen values $\mu_1 \geq \mu_2$ give the average contrast in those directions (the asterix upperscript is explained below). The first eigen vector $\mathbf{u}^*$ of the structure tensor points in the direction of the highest gray level fluctuation.
Finally, the corresponding diffusion tensor $\mathbf{D}$ is given by:

$$\mathbf{D} = \lambda_1 \mathbf{u}^* \mathbf{u}^{*T} + \lambda_2 \mathbf{v}^* \mathbf{v}^{*T} \tag{10}$$

where $\mathbf{u}^*$ and $\mathbf{v}^*$ are smoothed version of $\mathbf{u}$ and $\mathbf{v}$. $\mathbf{u}$ is orthogonal to the edges and $\mathbf{v}$ is tangent to the isophote lines (contours in the image) (see figure 1). The λ_1 and λ_2 values, that steers the diffusion play a fundamental role in the way the diffusion acts. For instance, we can have $\lambda_1 \approx \lambda_2$. In this case, the diffusion will be isotropic (any direction is favoured). When $\lambda_2 >> \lambda_1$, the diffusion is anisotropic, directed along the isophote.
Two kinds of anisotropic diffusion were introduced by Weickert:

- Edge-Enhancing Diffusion (EED) [76] designed to smooth noise while enhancing edges:

$$\begin{aligned} \lambda_1 &= \begin{cases} 1 & |\nabla I|^2 = 0 \\ 1 - exp(-\frac{C}{((|\nabla I|^2)/\lambda_e)^4}) & |\nabla I|^2 > 0 \end{cases} \\ \lambda_2 &= 1 \end{aligned} \tag{11}$$

  where, $C = 3.311488$.

- Coherence-Enhancing Diffusion (CED) [79] designed to enhance line-like texture:

$$\begin{aligned} \lambda_1 &= \alpha \\ \lambda_2 &= \begin{cases} \alpha & \mu_1 = \mu_2 \\ \alpha + (1-\alpha)exp(-\frac{C}{(\mu_1-\mu_2)^2}) & else \end{cases} \end{aligned} \tag{12}$$

  where C is a positive value and $\alpha \in [0, 1]$.

Laplacian-based diffusion The Laplacian-based diffusion relies on the decomposition of the diffusion equation as followed [46]:

$$\frac{\partial I}{\partial t} = c_1 I_{uu} + c_2 I_{vv} \tag{13}$$

where $\mathbf{u} \perp \mathbf{v}$ and I_{xx} is the second derivatives of I in the direction x. The diffusion behavior is entirely defined by the directions $\mathbf{u}$ and $\mathbf{v}$ and the corresponding weights c_1 and c_2. The previous formulation can also written as

$$\frac{\partial I}{\partial t} = trace(\mathbf{TH}) \tag{14}$$

where $\mathbf{H}$ is the hessian of I given by

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \tag{15}$$

and

$$\mathbf{T} = c_1 \mathbf{u}\mathbf{u}^T + c_2 \mathbf{v}\mathbf{v}^T \tag{16}$$

is a 2×2 symmetric matrix whose eigenvalues are c_1 and c_2 and respective eigenvectors are $\mathbf{u}$ and $\mathbf{v}$.
From the previous introduced notation (see figure 1), when $c_1 = 0$ and $c_2 = 1$, the mean curvature flow is obtained:

$$\frac{\partial I}{\partial t} = I_{vv} \tag{17}$$

Such approach acts only along the isophote lines (edges). In the next section, some examples are given in figure 3.

2.4.3 Noise reduction Applications

One of the most interesting applications is the image restoration. Figure 2 shows an example of noise reduction based on different diffusion algorithms.
Results show that image features are better preserved by the non-linear isotropic diffusion and by the anisotropic diffusion than with the isotropic one. The image is significanlty blurred by the isotropic scheme (the noise as well as the edges) whereas the two other approaches succeed in removing the noise and in keeping the edges. An

(a)

(b) PSNR=23.06dB (c) PSNR=24.93dB (d) PSNR=28dB

(e) (f) (g)

Figure 2: (a) Original picture corrupted with a gaussian Noise; (b), (c) and (d) are the filtered picture with a linear isotropic diffusion, a non linear isotropic diffusion (Charbonnier diffusivity function $\lambda^2 = 0.01$) and the anisotropic diffusion proposed by Tschumperlé [71], respectively. The last row shows the residual picture (difference between original picture uncorrupted by noise and the filtered picture).

objective quality measure (PSNR) is given for each case. Figure 3 shows an example of noise reduction based on an anisotropic diffusion. The scheme used is the one proposed by Tschumperlé [71] and described in the section called *Laplacian-based diffusion*. The goal of this figure is to show how the anisotropic diffusion can be steered by the coefficient c_1 and c_2 of the following equation:

$$\mathbf{T} = c_1 \mathbf{u}\mathbf{u}^T + c_2 \mathbf{v}\mathbf{v}^T \tag{18}$$

2.5 Sparse representation

2.5.1 Optimization problems

Sparse decompositions aim at describing a signal as the combination of a small number of vectors. In the sequel, we will adopt the following definitions and notations.

We call *sparsity* of the vector $\mathbf{x} = [x_1, \ldots, x_m]^T$ the number of zero coefficients in $\mathbf{x}$. The *diversity* of $\mathbf{x}$ corresponds to the number of nonzero coefficients in $\mathbf{x}$. If $\mathbf{x}$ is M-dimensional, we obtain thus

$$\text{diversity} = M - \text{sparsity}. \tag{19}$$

The sparse expansion is considered in the general framework where the dictionary is eventually redundant: the selection set of vectors is expressed as a matrix $\mathbf{D}$ with M columns $\mathbf{d}_k \in \mathbb{R}^N$, $M \geq N$. Each column corresponds to a vector, also called *atom*.

This redundancy has a sizeable interest in the description of complex signals as audio signals or images. Let us consider a N-dimensional signal $\mathbf{y} = [y_1, \ldots, y_N]^T$ and its sparse approximation defined as the combination of L atoms. For a given sparsity (*i.e.,* for a given L), the more the dictionary is redundant, the more the sparse approximation can be close to the signal $\mathbf{y}$. Reciprocally, for a given approximation error, a redundant dictionary could decrease the number of atoms required to approximate the signal $\mathbf{y}$.

The sparse model can be expressed as follows:

$$\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{n}. \tag{20}$$

The product $\mathbf{D}\mathbf{x}$ is the sparse approximation of $\mathbf{y}$ in dictionary $\mathbf{D}$. The gap between this approximation and the real signal $\mathbf{y}$ is seen as a noise, denoted $\mathbf{n}$. If this gap is null, the product $\mathbf{D}\mathbf{x}$ is called *sparse representation* of $\mathbf{y}$ in $\mathbf{D}$.

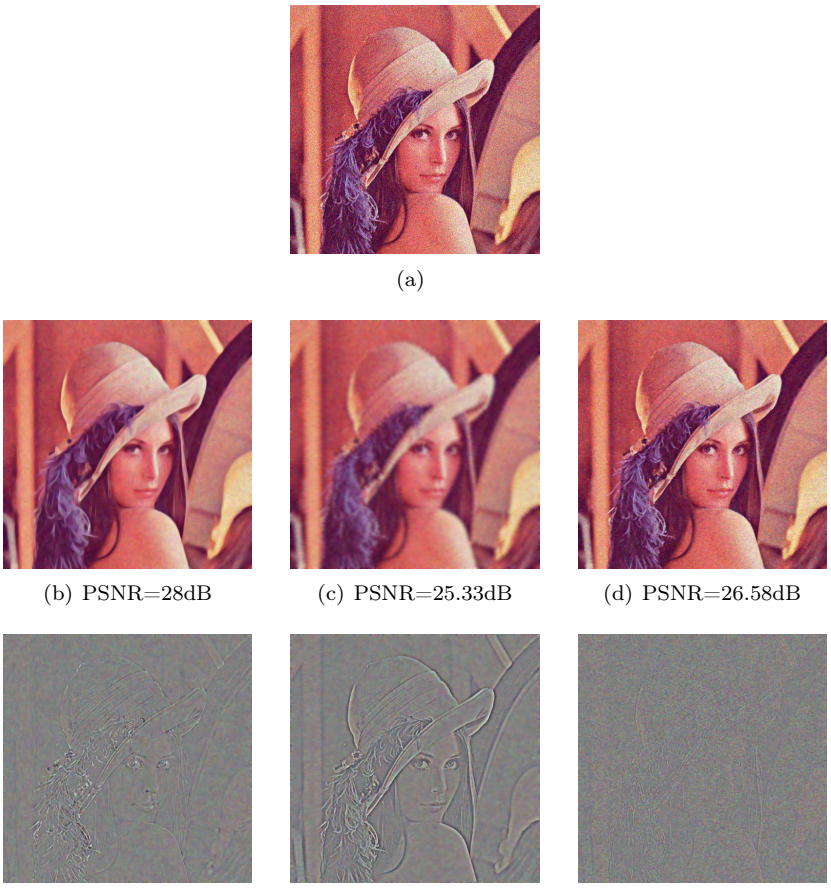
Finally, we denote $\mathbf{s} = [s_1, \ldots, s_M]^T$ the *support* of the sparse expansion, such as

$$s_i = \begin{cases} 1 & \text{if } x_i \neq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{21}$$

Sparsity measures We find in the literature several sparsity measures.

The "ideal" measure is formalized as the ℓ_0-pseudo-norm and is written as $\|.\|_0$. This pseudo-norm counts the number of nonzero coefficients:

$$\|\mathbf{x}\|_0 = \sum_{i=1}^{M} |x_i|^0 = |\mathcal{I}|, \tag{22}$$



(a)

(b) PSNR=28dB (c) PSNR=25.33dB (d) PSNR=26.58dB

(e) (f) (g)

Figure 3: (a) Original picture corrupted with a gaussian Noise; (b), (c) and (d) are the filtered picture with the anisotropic diffusion proposed by Tschumperlé [71]: $\mathbf{T} = c_1\mathbf{u}\mathbf{u}^T + c_2\mathbf{v}\mathbf{v}^T$; (b) $c_1 = 0$ and $c_2 = 1$, Mean curvature flow; (c) $c_1 = 1$ and $c_2 = 0$; (d) c_i are function of the eigen values, respectively. The last row shows the residual picture (difference between original picture uncorrupted by noise and the filtered picture).

where $\mathcal{I} = \{i | x_i \neq 0\}$.

Note that a norm $\mathcal{G}$ on a set $\mathcal{X}^M$ satisfies the three following properties:

- Positivity : $\forall \mathbf{x} \in \mathcal{X}^M \setminus \{0_{\mathcal{X}}^M\},\ \ \mathcal{G}(\mathbf{x}) > 0$,
- Homogeneity : $\forall \mathbf{x} \in \mathcal{X}^M,\ \ \forall \lambda \in \mathcal{K},\ \ \mathcal{G}(\lambda \mathbf{x}) = |\lambda| \mathcal{G}(\mathbf{x})$,
- Triangle inequality : $\forall (\mathbf{x}_1, \mathbf{x}_2) \in (\mathcal{X}^M)^2,\ \ \mathcal{G}(\mathbf{x}_1 + \mathbf{x}_2) \leq \mathcal{G}(\mathbf{x}_1) + \mathcal{G}(\mathbf{x}_2)$.

The sparsity measure (22) does not satisfy the homogeneity condition, it is thus defined as a pseudo-norm.

We find also "relaxed" sparsity measures, corresponding to ℓ_p-(quasi-)norms of $\mathbf{x}$:

$$\|\mathbf{x}\|_p = (\sum_{i=1}^{M} |x_i|^p)^{\frac{1}{p}}, \quad 0 < p < \infty. \tag{23}$$

For $0 < p < 1$, the triangle inequality is not satisfied, but is replaced by the quasi-triangle inequality

$$\forall (\mathbf{x}_1, \mathbf{x}_2) \in (\mathcal{X}^M)^2,\ \ \|\mathbf{x}_1 + \mathbf{x}_2\|_p^p \leq \|\mathbf{x}_1\|_p^p + \|\mathbf{x}_2\|_p^p,$$

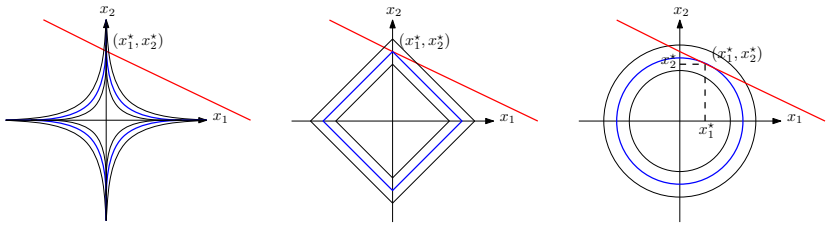
the measure (23) constitutes so a quasi-norm. On the other hand, for $p > 1$, the three properties are verified, the measure (23) is definitely a norm. In the sequel, we will generally use the term "norm" to express the three notions of pseudo-norm, quasi-norm et norm.

Several considerations can lead to choose one or the other norm. Among them, the convexity is useful for the resolution of some inverse sparse problems, as we will see in the next subsections. According to the value of the parameter p, a ℓ_p-norm will present level curves convex or not: for $p \geq 1$, they will be convex, for $p < 1$, not. Figure 4 illustrates this point. Let us consider a one-dimensional signal y and a dictionary $\mathbf{D}$ made up of 2 one-dimensional atoms ($N = 1$, $M = 2$). The representation of y in $\mathbf{D}$ will be the two-dimensional vector $\mathbf{x} = [x_1, x_2]$. We look for the solution of the following problem :

$$\mathcal{P}_p : \qquad \min_{\mathbf{x}} \|\mathbf{x}\|_p \quad \text{subject to} \quad \mathbf{D}\mathbf{x} = y. \tag{24}$$

A graphical interpretation of this problem is given by Figure 4 for three different "classes" of values for p: $p < 1$, $p = 1$, $p > 1$. To each of this class corresponds a graph (resp. figures 4(a), (b) et (c)). The level curves of the norms are represented, as well as the straight line defined by $y = \mathbf{D}\mathbf{x}$ (in red). The solution of problem (24) is the point $\mathbf{x}^\star = [x_1^\star, x_2^\star]$, located at the intersection of the straight line defined by $y = \mathbf{D}\mathbf{x}$ and the smallest-valued level curve (in blue on the figures).

If the convexity can be interesting for the resolution of some optimization problems (see next sections 2.5.1 and 2.5.1), Figure 4 illustrates at the same time the fact that any norm does not give the same result and the same performance in terms of sparsity. Hence, while for $p \leq 1$, the solution will be actually sparse ($x_1 = 0$, $x_2 \neq 1$), for



(a) $p < 1$ (b) $p = 1$ (c) $p > 1$

Figure 4: (Non-)convexity of ℓ_p-norms, for ($p < 1$, figure (a)) $p \geq 1$, figures (b) and (c)

$p > 1$, it will have a dimension upper than the one of the signal we want to represent ($x_i \neq 0 \ \ \forall i \in \{1, 2\}$).

In practice, several contributions propose to replace the ℓ_0-norm by the ℓ_1-norm. Attractive because of its convexity, the ℓ_1-norm presents also the interest to be equivalent to the ℓ_0-norm under some conditions (see Theorem 1, next section). It is not the case for ℓ_p-norms with $p > 1$, which are convex but do not favour sparsity. In the sequel, we restrain our study to values of p comprised between 0 and 1.

Sparse representations The "standard" inverse problem related to sparse representations consists in looking for the vector $\mathbf{x}$ containing the lowest number of nonzero coefficients which leads to an exact reconstruction of vector $\mathbf{y}$ in dictionary $\mathbf{D}$. We already saw the formulation of this problem in previous section:

$$\mathcal{P}_p : \qquad \min_{\mathbf{x}} \|\mathbf{x}\|_p \quad \text{subbject to} \quad \mathbf{D}\mathbf{x} = \mathbf{y}. \tag{24}$$

During the last years, several studies allowed to characterize the solutions admitted by the problem $(\mathcal{P}_p)$ according to the value of p. In particular, the contributions of Donoho and Elad [22], contemporary with those of Gribonval and Nielsen [32], have been interested in the conditions ensuring the uniqueness and equivalence of the solutions of problems $(\mathcal{P}_p)$, for resp. $p \in \{0, 1\}$ et $p \in [0, 1]$. Theorem 1 sums up their results.

Theorem 1 *Assume* $\mathbf{D}$ *is an arbitrary dictionary in a finite or infinite Hilbert space. If* $\mathbf{y} = \sum_i x_i \mathbf{d}_i$ *with*

$$\|\mathbf{x}\|_0 < \frac{1}{2}(1 + \frac{1}{\mu(\mathbf{D})}), \tag{25}$$

then $\mathbf{x}$ *is the unique and simultaneous solution of the minimization problems* $(\mathcal{P}_p)$*, for* $0 \leq p \leq 1$.

The parameter $\mu(\mathbf{D})$ is called *coherence* of the dictionary $\mathbf{D}$ and is defined as $\mu(\mathbf{D}) \triangleq \max_{k \neq k'} |\langle \mathbf{d}_k, \mathbf{d}_{k'} \rangle|$. The coherence measures the maximal proximity between atoms

of a dictionary, its value is comprised between 0 and 1 if the atoms are normed to 1. Hence, if $\mathbf{D}$ is an orthonormal basis, $\mu(\mathbf{D}) = 0$ and Theorem 1 states that any vector $\mathbf{x}$ can be recovered univocaly whatever the sparsity; on the other hand, if two atoms are colinear (equal if they are normed to 1), $\mu(\mathbf{D}) = 1$ and no vector $\mathbf{x}$ can be recovered univocally.

Other conditions were highlighted by Fuchs [28] and Tropp [70] to ensure the equivalence of problems $(\mathcal{P}_1)$ et $(\mathcal{P}_0)$.

Theorem 2 *Assume* $\mathbf{y}$ *has an expansion* $\mathbf{y} = \sum_{i=1}^{L} x_i \mathbf{d}_i$ *in an arbitrary dictionary. For* $\mathcal{I} = \{i | x_i \neq 0\}$*, denote* $\Phi_{\mathcal{I}}$ *the linear operator* $\Phi_{\mathcal{I}}\mathbf{x} = \sum_{i \in \mathcal{I}} x_i \mathbf{d}_i$*. If*

$$\sup_{i \notin \mathcal{I}} \|\Phi_{\mathcal{I}}^{+} \mathbf{d}_i\|_1 < 1, \tag{26}$$

where $.^+$ *is the pseudo-inverse, then* $\mathbf{x}$*, the solution of problem* $(\mathcal{P}_1)$*, is also the solution of problem* $(\mathcal{P}_0)$*.*

The consequences of Theorems 1 and 2 are very important: the problem $(\mathcal{P}_0)$ is NP-hard, *i.e.,* all possible combinations of atoms have to be considered to find the optimal solution, which is not conceivable in high dimension. Theorems 1 and 2 allow and motivate the use of relaxed measures for the sparsity. In particular, the ℓ_1-norm enables the use of linear programming to solve $(\mathcal{P}_1)$ (the equivalence between ℓ_1-norm optimization and linear programming is proved in [8]). This is the approach adopted by Chen *et al.* in [16] for the design of the Basis Pursuit (BP) algorithm.

Sparse approximations In practice, the problem of sparse representations is useful only in case of redundant dictionaries. If the dictionary $\mathbf{D}$ has more lines than columns ($M < N$), problem $(\mathcal{P}_p)$ has possibly no solution and if $\mathbf{D}$ is a basis, the solution is straightforward.

The sparse approximation offers a more general research framework. It consists in looking for the sparsest vector resulting in the closest approximation to the real signal $\mathbf{y}$. This problem relies on the two notions of approximation quality and sparsity of the decomposition vector. The approximation quality is in general measured by a square error between the real signal $\mathbf{y}$ and the sparse approximation $\mathbf{Dx}$:$\|\mathbf{y} - \mathbf{Dx}\|_2^2$. Sparsity is taken into account according to the measures presented in the subsection 2.5.1.

Three main types of optimization problems can be then considered.

Sparsification The first one is analogous to problem (24). We want to find the sparsest vector $\mathbf{x}$ under the constraint that the approximation error is lower than a given threshold $\epsilon \geq 0$. This can be formalized as follows:

$$\mathcal{P}_p^P : \quad \min_{\mathbf{x}} \|\mathbf{x}\|_p \quad \text{subject to} \quad \|\mathbf{y} - \mathbf{Dx}\|_2^2 \leq \epsilon. \tag{27}$$

Approximation Inversely, we can look for the sparse vector $\mathbf{x}$ which leads to the closest sparse approximation to $\mathbf{y}$, under the constraint that $\mathbf{x}$ has a sparsity upper than a given threshold. This can be expressed as follows:

$$\mathcal{P}_p^A : \quad \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{Dx}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_p \leq L. \tag{28}$$

If the sparsity of $\mathbf{x}$ is measured by the ℓ_0-norm, the parameter L corresponds to the maximal number of nonzero coefficients in $\mathbf{x}$.

Regularization Finally, the vector $\mathbf{x}$ can be considered as the solution of a compromise between approximation quality and sparsity. The compromise is fixed by a parameter $\lambda \geq 0$ and the problem can be formalized as follows:

$$\mathcal{P}_p^R : \qquad \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_p. \tag{29}$$

A factor $\frac{1}{2}$ is sometimes added as weighting factor of the approximation error $\|\mathbf{y}-\mathbf{D}\mathbf{x}\|_2^2$ to facilitate the derivation of algorithms. Note that this formalization can be seen as a Lagrangian version of the approximation and regularization problems (27)-(28).

2.5.2 Sparse decomposition algorithms

Each of the formalization (27) to (29) leads to the design of one or several different algorithms. In this section, we review the main existing algorithms. We first begin by the particular case where the dictionary $\mathbf{D}$ is an orthonormal basis.

Particular case: orthonormal basis The use of an orthonormal basis (*i.e.,* $\|\mathbf{d}_k\|_2 = 1$ $\forall k \in \{1, \ldots, M\}$, $\langle \mathbf{d}_k, \mathbf{d}'_k \rangle = 0$ if $k \neq k'$, and $N = M$) as approximation dictionary is a simple case allowing a fast resolution of approximation problems $(\mathcal{P}_0^P)$, $(\mathcal{P}_0^A)$ and $(\mathcal{P}_0^R)$ under the "ideal" sparsity constraint $p = 0$.

The resolution is thus immediate for the sparsification and approximation problems by noticing that

$$\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 = \|\mathbf{x} - \mathbf{D}^T\mathbf{y}\|_2^2. \tag{30}$$

Hence, the solution $\mathbf{x}^\star$ to these problems is obtained by thresholding the coefficients of the product $\mathbf{D}^T\mathbf{y}$:

$$\mathbf{x}^\star = \mathcal{T}_\lambda(\mathbf{D}^T\mathbf{y}), \tag{31}$$

where $\mathcal{T}_\lambda$ is a thresholding operator.

In the sparsification case $(\mathcal{P}_0^P)$, the solution $\mathbf{x}^\star$ is obtained by setting to zero successively, in the increasing order of their values, the coefficients of the product $\mathbf{D}^T\mathbf{y}$, until the constraint $\|\mathbf{x}^\star - \mathbf{D}^T\mathbf{y}\|_2^2 \leq \epsilon$ is satisfied. In the approximation case $(\mathcal{P}_0^A)$, the solution $\mathbf{x}^\star$ is obtained by keeping the L largest coefficients of the product $\mathbf{D}^T\mathbf{y}$.

The resolution of the regularization problem $(\mathcal{P}_0^R)$ is less obvious but reduces also to a simple thresholding operation ([63]):

$$x_i^\star = \mathcal{T}_\lambda^0((\mathbf{D}^T\mathbf{y})_i) = \begin{cases} (\mathbf{D}^T\mathbf{y})_i & \text{if } |(\mathbf{D}^T\mathbf{y})_i| > \sqrt{\lambda}, \\ 0 & \text{otherwise,} \end{cases} \tag{32}$$

where $(\mathbf{D}^T\mathbf{y})_i$ is the i^{th} coefficient of the product $\mathbf{D}^T\mathbf{y}$.

A similar result has been obtained by Donoho in [23] for the regularization problem $(\mathcal{P}_1^R)$. In case of an orthonormal basis, he showed that this problem can be solved by a soft-thresholding:

$$x_i^\star = \mathcal{T}_\lambda^1((\mathbf{D}^T\mathbf{y})_i) = \begin{cases} (\mathbf{D}^T\mathbf{y})_i - \lambda/2 & \text{si} \quad (\mathbf{D}^T\mathbf{y})_i > \lambda/2, \\ 0 & \text{si} \quad |(\mathbf{D}^T\mathbf{y})_i| \leq \lambda/2, \\ (\mathbf{D}^T\mathbf{y})_i - \lambda/2 & \text{si} \quad (\mathbf{D}^T\mathbf{y})_i < -\lambda/2. \end{cases} \tag{33}$$

General case: redundant dictionaries In the general case where $\mathbf{D}$ is a redundant dictionary ($M > N$), problems $(\mathcal{P}_0^P)$, $(\mathcal{P}_0^A)$ and $(\mathcal{P}_0^R)$, which use the "ideal" sparsity measure, are NP-hard. Suboptimal algorithms are then designed to approximate "at best" the optimal solution.

Gradient descent The simpliest resolution method is the gradient descent. This method is general and can be used in minimization problems involving a differentiable function. In our case, the gradient descent can be applied to the regularization problem $(\mathcal{P}_1^R)$. The resulting algorithm is iterative and suggests then the following update equation

$$\hat{\mathbf{x}}^{(n+1)} = \hat{\mathbf{x}}^{(n)} - \eta \nabla f(\mathbf{x}, \mathbf{y}, \mathbf{D}) \mid_{\hat{\mathbf{x}}^{(n)}}, \tag{34}$$

where $f(\mathbf{x}, \mathbf{y}, \mathbf{D}) = \frac{1}{2}\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_1$, ∇ is the gradient operator and η is the scale parameter. Used in the first works on sparse approximations, this method was rapidly supplanted by other efficient algorithms.

Iterative thresholding algorithms Solving the regularization problem $(\mathcal{P}_p^R)$ with $p = 0$ or $p = 1$, the iterative thresholding algorithms are an extension of the hard- and soft-thresholding algorithms introduced in previous subsection 2.5.2 to the general case where $\mathbf{D}$ is an arbitrary redundant dictionary.

We find in the literature several different versions of iterative thresholding algorithms.

The first relevant works were realized by Kingsbury et Reeves. In [45], they derive an iterative thresholding method allowing to solve problem $(\mathcal{P}_0^R)$. However, their contribution is done without a clear connection to the objective function $(\mathcal{P}_0^R)$. We find a more explicit version of their results in [9]. Blumensath and Davies show thus that the problem $(\mathcal{P}_0^R)$ can be solved by the following update equation:

$$x_i^{(n+1)} = \mathcal{T}_\lambda^0(x_i^{(n)} + \mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\mathbf{x}^{(n)})), \tag{35}$$

where $\mathcal{T}_\lambda^0$ is defined by equation (32).

A similar result is proved for problem $(\mathcal{P}_1^R)$. In [21], Daubechies *et al.* give rise to a solution obtained by iterating the following expression:

$$x_i^{(n+1)} = \mathcal{T}_\lambda^1(x_i^{(n)} + \mathbf{d}_i^T(\mathbf{y} - \mathbf{D}\mathbf{x}^{(n)})), \tag{36}$$

where $\mathcal{T}_\lambda^1$ is defines by equation (33). In the same idea, we can mention the contributions of Combettes and Pesquet in [18].

Algorithme 1: Matching Pursuit

0. Initialization : $\mathbf{r}^{(0)} = \mathbf{y}$.

Until the stopping criterion is reached, repeat:

1. Support update (Selection of the atom the most correlated with the residual)

$$\hat{s}_j^{(n)} = \begin{cases} 1 & \text{if } j = \arg\max_i \langle \mathbf{r}^{(n-1)}, \mathbf{d}_i \rangle^2, \\ \hat{s}_j^{(n-1)} & \text{otherwise.} \end{cases} \tag{37}$$

2. Coefficient update (Computation of the corresponding coefficient)

$$\hat{x}_j^{(n)} = \begin{cases} \hat{x}_j^{(n-1)} + \langle \mathbf{r}^{(n-1)}, \mathbf{d}_j \rangle & \text{if } j = \arg\max_i \langle \mathbf{r}^{(n-1)}, \mathbf{d}_i \rangle^2, \\ \hat{x}_j^{(n-1)} & \text{otherwise.} \end{cases} \tag{38}$$

3. Residual update: $\mathbf{r}^{(n)} = \mathbf{r}^{(n-1)} - \langle \mathbf{r}^{(n-1)}, \mathbf{d}_j \rangle \mathbf{d}_j$.

Pursuit algorithms The pursuit algorithms, or greedy algorithms, deal with problems $(\mathcal{P}_0^P)$ or $(\mathcal{P}_0^A)$. Their process is iterative: at each iteration, at most one atom is added to the decomposition. This is done until the stopping criterion is reached, which can be a maximal approximation error ϵ, or a maximal atom number in the decomposition, L.

The decisions are taken locally, there is no guarantee to obtain a global optimum. However, these algorithms are in general simple and fast, they are in practice very often used.

There exist many pursuit algorithms. We review the most popular of them:

- Matching Pursuit (MP), first called CLEAN and introduced in 1974 in the astrophysical community in [37], then in the statistical community in [39] and finally by Mallat and Zhang in the signal processing society in [53],
- Orthogonal Matching Pursuit (OMP), evolution of MP proposed by Pati *et al.* in [58].

Variants and extensions of these two algorithms were derived afterwards. Let us mention for example the algorithms Optimized Orthogonal Matching Pursuit (OOMP, [62]), Complementary Matching Pursuit (CMP, [60]) and Complementary Orthogonal Matching Pursuit (COMP, [61]).

Most greedy algorithms estimate successively the support of the sparse decomposition $\mathbf{s}$ and the coefficients of the sparse vector $\mathbf{x}$. The algorithm MP relies on a selection of the atoms the most correlated with the signal. Its general process is given by Algorithm 1.

Note that an atom can be chosen several times, so that the algorithm requires sometimes numerous iterations to reach the stopping criterion. However, the algorithm is ensured to converge if the dictionary $\mathbf{D}$ spans the entire space of signals (cf. [43]).

Algorithme 2: Orthogonal Matching Pursuit

0. Initialization : $\mathbf{r}^{(0)} = \mathbf{y}$.
Until the stopping criterion is reached, repeat
1. Support update (Selection of the atom the most correlated with the residual)

$$\hat{s}_j^{(n)} = \begin{cases} 1 & \text{if } j = \arg\max_i \langle \mathbf{r}^{(n-1)}, \mathbf{d}_i \rangle^2, \\ \hat{s}_j^{(n-1)} & \text{otherwise.} \end{cases} \tag{39}$$

2. Coefficient update (Computation of the corresponding coefficients)

$$\hat{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}} = \mathbf{D}^+_{\hat{\mathbf{s}}^{(n)}} \mathbf{y}, \tag{40}$$

where $\mathbf{D}^+_{\hat{\mathbf{s}}^{(n)}}$ is the pseudo-inverse of $\mathbf{D}_{\hat{\mathbf{s}}^{(n)}}$, matrix made up of the columns $\mathbf{d}_i$ such as $s_i \neq 0$.
3. Residual udpate: $\mathbf{r}^{(n)} = \mathbf{y} - \mathbf{D}\hat{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}}$.

The algorithm OMP performs the same support update as MP but calculates the coefficients of the sparse vector in a different way. Thus, instead of estimating only one coefficient per iteration (from the projection of the residual on the selected atom), OMP recalculates all coefficients by projecting the signal $\mathbf{y}$ on the space spanned by all selected atoms. The operation is performed using a Gram-Schmidt orthogonalization. Thus a selected atom can not be selected again and, at the N^{th} iteration, the N selected and orthogonalized atoms constitute an orthogonal basis of $\mathbb{R}^N$ able to represent with a null error the signal $\mathbf{y}$. OMP is thus ensured to converge in a finite number of iterations, at most equal to the signal dimension, N. OMP is described in Algorithm 2.

Several works have been interested to the reconstruction ability of the algorithms MP and OMP. Tropp [70], and Gribonval and Vandergheynst [33] give rise to some conditions ensuring exact recovery by MP and OMP.

Theorem 3 *Assume* $\mathbf{y}$ *has an expansion* $\mathbf{y} = \sum_{i=1}^{L} x_i \mathbf{d}_i$ *in an arbitrary dictionary. For* $\mathcal{I} = \{i | x_i \neq 0\}$*, denote* $\Phi_{\mathcal{I}}$ *the linear operator* $\Phi_{\mathcal{I}}\mathbf{x} = \sum_{i \in \mathcal{I}} x_i \mathbf{d}_i$*. If*

$$\sup_{i \notin \mathcal{I}} \|\Phi^+_{\mathcal{I}} \mathbf{d}_i\|_1 < 1, \tag{41}$$

where $\Phi^+_{\mathcal{I}}$ *is the pseudo-inverse of* $\Phi_{\mathcal{I}}$*, then MP and OMP recover the decomposition,* i.e., *at each iteration* n*, a "correct" atom* $\mathbf{d}_n$ *is selected* ($n \in \mathcal{I}$).

Thus, under the same conditions, the algorithms MP and OMP recover the solution of $(\mathcal{P}_0^P)$ or $(\mathcal{P}_0^A)$, and BP the solution of $(\mathcal{P}_1)$ (cf. Theorem 2).

As we saw, pursuit algorithms do not allow to select more than one atom per iteration. This limitation is avoided by the stagewise pursuit algorithms. Among the most popular of them, we find the Stagewise OMP algorithm (StOMP, [24]), the Subspace

Algorithme 3: Stagewise Orthogonal Matching Pursuit

0. Initialization : $\mathbf{r}^{(0)} = \mathbf{y}$.
Until the stopping criterion is reached, repeat
1. Support update

$$\hat{s}_j^{(n)} = \begin{cases} 1 & \text{if } \langle \mathbf{r}^{(n-1)}, \mathbf{d}_j \rangle^2 > T^{(n)}, \\ \hat{s}_j^{(n-1)} & \text{otherwise.} \end{cases} \tag{42}$$

2. Coefficient update (Computation of the corresponding coefficients)

$$\hat{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}} = \mathbf{D}^+_{\hat{\mathbf{s}}^{(n)}} \mathbf{y}, \tag{43}$$

where $\mathbf{D}^+_{\hat{\mathbf{s}}^{(n)}}$ is the pseudo-inverse of $\mathbf{D}_{\hat{\mathbf{s}}^{(n)}}$, matrix made up of the columns $\mathbf{d}_i$ such as $s_i \neq 0$.
3. Residual update: $\mathbf{r}^{(n)} = \mathbf{y} - \mathbf{D}\hat{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}}$.

Pursuit algorithm (SP, [19]), similar to the Compressive Sampling Matching Pursuit algorithm (CoSaMP, [56]), or the Morphological Component Analysis algorithm (MCA, [10]). We detail here the algorithms StOMP and CoSaMP/SP.

StOMP can be seen as a variant of OMP. The computation of the sparse coefficients is identical, but the choice of the atoms added to the support of the sparse decomposition at each iteration is performed by a thresholding operation with parameter $T^{(n)}$ on $\langle \mathbf{r}^{(n-1)}, \mathbf{d}_i \rangle^2$. Donoho *et al.* propose in [24] two different approaches to specify the value of parameter $T^{(n)}$ at each iteration. Algorithm 3 presents the main operations of StOMP.

The algorithm CoSaMP/SP deals exclusively with problem ($\mathcal{P}_0^A$). But it offers an additional freedom degree to solve it: the atom deselection. To this purpose, CoSaMP/SP relies on the knowledge of the maximal number of nonzero coefficients L. Its principle is described in Algorithm 4.

Algorithms CoSaMP and SP distinguish by the choice of parameter P. In SP, it is equal to L, the maximal number of nonzero coefficients. In CoSaMP it is set to $2L$.

Convex optimization algorithms Convex optimization algorithms deal with regularization problem ($\mathcal{P}_1^R$). They are divided into two main classes: the fixed-point algorithms as the Focal Underdetermined System Solver algorithm (FOCUSS, [31]) and the algorithms based on quadratic programming as algorithms Basis Pursuit Denoising (BPD, [16]) and Global Matched Filter (GMF, [29]). We do not detail them here, but some important results deserve to be mentioned in comparison with previous algorithms.

Many numerical simulations tend to prove ([16]) that if the signal $\mathbf{y}$ has a very sparse expansion in dictionary $\mathbf{D}$ and $\mathbf{D}$ is well-structured, then the sparse decomposition can perfectly be recovered by algorithms BPD and GMF. This observation has led to a series of theoretical results for different dictionaries (cf. for exemple [32], [70]). One of the most general result is the one obtained by Fuchs in [28].

Algorithme 4: SP/CoSaMP

0. Initialization : $\mathbf{r}^{(0)} = \mathbf{y}$.
Until the stopping criterion is reached, repeat
1. Selection of the P atoms the most correlated with the residual

$$\tilde{\mathbf{s}}^{(n)} = \arg\max_{\mathbf{s}} \sum_i s_i \langle \mathbf{r}^{(n)}, \mathbf{d}_i \rangle^2 \text{ subject to } \|\mathbf{s}\|_0 = P, \tag{44}$$

2. Computation of the corresponding coefficient

$$\tilde{\mathbf{x}}_{\tilde{\mathbf{s}}^{(n)}} = \mathbf{D}^+_{\tilde{\mathbf{s}}^{(n)}} \mathbf{y}, \tag{45}$$

where $\mathbf{D}^+_{\tilde{\mathbf{s}}^{(n)}}$ is the pseudo-inverse of $\mathbf{D}_{\tilde{\mathbf{s}}^{(n)}}$, matrix made up of the columns $\mathbf{d}_i$ such as $s_i \neq 0$.
3. Support update (Selection of the L atoms corresponding to the L largest coefficients of $\tilde{\mathbf{x}}$)

$$\hat{\mathbf{s}}^{(n)} = \arg\max_{\mathbf{s}} \sum_i s_i |\tilde{x}^{(n)}_j| \text{ soumis à } \|\mathbf{s}\|_0 = L, \tag{46}$$

4. Coefficient update

$$\hat{\mathbf{x}}^{(n)} = \tilde{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}}, \tag{47}$$

5. Residual update: $\mathbf{r}^{(n+1)} = \mathbf{y} - \mathbf{D}\hat{\mathbf{x}}_{\hat{\mathbf{s}}^{(n)}}$.

Theorem 4 *Assume* $\mathbf{y}$ *has an expansion* $\mathbf{y} = \sum_{i=1}^{L} x_i \mathbf{d}_i$ *in an arbitrary dictionary. For* $\mathcal{I} = \{i | x_i \neq 0\}$*, denote* $\Phi_{\mathcal{I}}$ *the linear operator* $\Phi_{\mathcal{I}}\mathbf{x} = \sum_{i \in \mathcal{I}} x_i \mathbf{d}_i$*. If* λ *is small enough and*

$$|\langle (\Phi^+_{\mathcal{I}})^* sign(\mathbf{x}), \mathbf{d}_i \rangle| < 1, \quad \forall i \notin \mathcal{I}, \tag{48}$$

where $\Phi^+_{\mathcal{I}}$ *is the pseudo-inverse of* $\Phi_{\mathcal{I}}$ *and* $(\Phi^+_{\mathcal{I}})^*$ *the adjoint of* $\Phi^+_{\mathcal{I}}$*, then BPD recovers the expansion: each nonzero component of* $\mathbf{x}$ *corresponds to an index* $i \in \mathcal{I}$*.*

The convex optimization algorithms present in general a good performance in terms of approximation quality vs. decomposition sparsity, in comparison with the greedy algorithms. But this is achieved at the expense of a higher complexity. While MP and OMP have respectively a complexity of order M and M^2 per iteration, GMF requires near to N^3 operations.

Bayesians algorithms Recently, new approaches based on a Bayesian framework have been proposed. These approaches assume the following model:

$$\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{n}, \tag{49}$$

where $\mathbf{n}$ is a white Gaussian noise with variance σ_n^2.

The contributions distinguish first by the a priori probability distribution of the vector $\mathbf{x}$.

The most intuitive is the Laplacian distribution which implements the sparsity measure based on the ℓ_1-norm ([7]). For each component x_i of $\mathbf{x}$,

$$p(x_i) \propto \exp(-\lambda|x_i|), \tag{50}$$

where $\propto$ denotes equality up to a normalization factor.

Given the signal $\mathbf{y}$ and the dictionary $\mathbf{D}$, we prove easily that the problem $(\mathcal{P}_1^R)$ corresponds to a Maximum A Posteriori (MAP) estimation problem on $\mathbf{x}$ using prior (50):

$$\begin{aligned} \mathbf{x}^\star_{MAP} &= \arg\max_{\mathbf{x}} \log p(\mathbf{y}, \mathbf{x}, \mathbf{D}), \\ &= \arg\max_{\mathbf{x}} \log p(\mathbf{y}|\mathbf{x}, \mathbf{D}) p(\mathbf{x}), \\ &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_1. \end{aligned} \tag{51}$$

This relation between inverse sparse problem and Bayesian approximation offers interesting analysis perspectives. In particular, rather than maximizing the posterior probability distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$, we can look for an approximation of it. A solution to the sparse problem can then be obtained on a classical way by a a posteriori maximization (51), or by a minimization of the mean square error (MMSE):

$$\mathbf{x}^\star_{MMSE} = \int_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}, \mathbf{D}) d\mathbf{x}. \tag{52}$$

This latter approach presents the advantage of integrating over all possible realizations of $\mathbf{x}$ rather than making a hard decision by estimating $\mathbf{x}$ as the maximum. In a general way, estimating the posterior distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$ allows to take into account the uncertainties inherent in the approximation of $\mathbf{x}$.

The computation of the posterior distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$ is impossible under a Laplacian prior since the Laplacian prior is not conjugate to a Gaussian likelihood [1], and hence the associated Bayesian inference may not be performed in closed form ([7]).

To overcome this difficulty, several substitute distributions have been proposed.

The first one considers the following mixture, for each component x_i of vector $\mathbf{x}$:

$$p(x_i) = (1 - p_i)\delta_0 + p_i \mathcal{N}(0, \sigma_i^2), \tag{53}$$

where δ_0 is a Dirac distribution, $\mathcal{N}(0, \sigma_i^2)$ a Gaussian law with zero mean and variance σ_i^2, p_i a probability regulating the mixture. In [36], He et Carin assume that the variables σ_i^{-2} and p_i are random and follow respectively a Gamma law and a Beta

[1] In Bayesian probability theory, a class of prior probability distributions $p(\theta)$ is said to be conjugate to a class of likelihood functions $p(\mathbf{z}|\theta)$ if the resulting posterior distributions $p(\theta|\mathbf{z})$ are in the same family as $p(\theta)$.

law. A Monte Carlo Markov Chain method (MCMC, [13]) is then used to estimate the posterior distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$.

Another approach considers a mixture of two Gaussians, where the components x_i depend on Bernoulli variables s_i as follows:

$$p(x_i) = \sum_{s \in \{0,1\}} p(s_i = s)\mathcal{N}(0, \sigma^2(s_i = s)). \tag{54}$$

We find this model in [4]. The authors use then a belief propagation method to estimate the posterior distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$. This technique relies on factor graphs, which enable a fast computation of global multivariate functions by exploiting the way in which the global function factors into a product of simpler local functions, each of which depends on a subset of variables ([47]). In [4], the authors show that, particularized to the model (54), the resulting algorithm has a complexity $\mathcal{O}(M \log^2(M))$. Thus the proposed algorithm remains competitive in comparison with pursuit or convex optimization algorithms.

A slightly more complex model is adopted in [81]. The authors propose an iterative algorithm estimating successively the variables $\mathbf{x} = [x_1, \ldots, x_M]^T$ and $\mathbf{s} = [s_1, \ldots, s_M]^T$ with a Maximum A Posteriori approach. The estimation of $\mathbf{x}$ for a given $\mathbf{s}$ is relatively easy due to the Gaussianity of $\mathbf{x}$. On the other hand, the estimation of $\mathbf{s}$ for a given $\mathbf{x}$ is more subtle. The authors resort to a gradient descent. In this purpose, they "convert" each discrete variable s_i to a continuous one via a mixture of two Gaussians centered around 0 and 1 with sufficiently small variances σ_0^2 and σ_1^2:

$$p(s_i) = \pi\mathcal{N}(0, \sigma_0^2) + (1 - \pi)\mathcal{N}(0, \sigma_1^2), \tag{55}$$

where π is a parameter comprised between 0 and 1 regulating the mixture. The iterative process they obtain can be compared to an Expectation-Maximization algorithm (EM). In terms of complexity, their approach is much more costly than the pursuit or convex optimization algorithms.

Finally, we find in [68] and [41] the following model for each component x_i:

$$p(x_i|\sigma_i) = \mathcal{N}(0, \sigma_i^2), \tag{56}$$

where σ_i^{-2} follow a Gamma law for all $i \in \{1, \ldots, M\}$. We set that the parameter σ_n^{-2} (σ_n^2 is the noise variance, cf. equation (49)) follows also a Gamma law. Assuming the variables $\boldsymbol{\sigma} = [\sigma_1, \ldots, \sigma_M]^T$ and σ_n are known, the posteriori distribution of $\mathbf{x}$ can be expressed analytically as a Gaussian distribution with mean μ and variance Σ such as :

$$\mu = \sigma_n^{-2}\Sigma\mathbf{D}^T\mathbf{y}, \tag{57}$$

$$\Sigma = (\sigma_n^{-2}\mathbf{D}^T\mathbf{D} + \mathbf{A})^{-1}, \tag{58}$$

where $\mathbf{A} = \mathrm{diag}(\sigma_1, \ldots, \sigma_M)$. Thus, the estimation of the posterior distribution $p(\mathbf{x}|\mathbf{y}, \mathbf{D})$ reduces to the estimation of the parameters σ_n and $\boldsymbol{\sigma}$. In [41], Ji *et al.* propose to use the Relevant Vector Machine algorithm (RVM) introduced by Tipping

in [68]. RVM performs a type-II Maximum of Likelihood (ML) (or evidence maximization) procedure, *i.e.,* a maximization of the *marginal likelihood* $p(\mathbf{y}|\boldsymbol{\sigma}, \sigma_n)$ ([6][50]). Because of the matrix inversion (58), the algorithm complexity is quite high, $\mathcal{O}(M^3)$. This defect has motivated the design of a fast RVM ([69]), with an inferior complexity $\mathcal{O}(MN^2)$. Finally, RVM presents the interest to make the atom deselection possible (with $\sigma_i = 0$), as algorithms SP and CoSaMP.

2.5.3 Prediction based on sparse representations

The idea of prediction based on sparse representations relies on the assumption that the missing data we want to predict and the observed data have a sparse representation in a given dictionary. Sparsity defines thus a prior on the signal made up of the concatenation of the observed and the missing data. This can be formalized as follows. Let $\mathbf{y} = [\mathbf{y}_o^T, \mathbf{y}_m^T]^T$ be the concatenation of $\mathbf{y}_o \in \mathbb{R}^{N_o}$, observed data, and $\mathbf{y}_m \in \mathbb{R}^{N_m}$, missing data. $\mathbf{y}$ is assumed to have a sparse representation in dictionary $\mathbf{D}$. The missing data $\mathbf{y}_m$ is thus estimated as

$$\mathbf{y}_m = \mathbf{D}_m \mathbf{x}^\star, \tag{59}$$

where the sparse representation $\mathbf{x}^\star$ is calculated from the observed data as the solution of problem $(\mathcal{P}_p^P)$, $(\mathcal{P}_p^A)$ or $(\mathcal{P}_p^R)$. For example, for the regularization problem $(\mathcal{P}_p^R)$,

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \|\mathbf{y}_o - \mathbf{D}_o \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_o.$$

$\mathbf{D}_o \in \mathbb{R}^{N_o \times M}$ (resp. $\mathbf{D}_m \in \mathbb{R}^{N_m \times M}$) is the dictionary whose rows correspond to the elements in $\mathbf{y}_o$ (resp. $\mathbf{y}_m$).

For a proper choice of the dictionary, it has been shown that such an approach can offer very good performance in prediction or inpainting problems, see *e.g.*, [34, 35, 26, 27, 54, 72].

In [34] and [35], Guleryuz considers an overcomplete dictionary made up of orthonormal bases and proposes an iterative implementation of the sparse representation problem applied to inpainting. Another approach is presented by Elad *et al.* in [26]. The proposed implementation involves a different type of dictionary, made up of atoms capturing either “cartoon" or “texture" areas. Elad *et al.* add also a total variation (TV) penalty term to the standard sparse representation problem. Finally, in [27], Fadili *et al.* introduce an implementation based on the expectation-maximization (EM) algorithm.

Several contributions also consider the problem of prediction based on sparse representations in the context of image/video coding, see *e.g.,* [54], [72]. These contributions mainly distinguish by the choice of the dictionary used to “sparsely” represent the signal and the choice of the data used for the prediction. In [54], Martin *et al.* consider an overcomplete dictionary made up of discrete real Fourier and cosine functions, while Türkan *et al.* [72] construct a dictionary from image patches taken in a large causal area and consider seven possible causal neighborhoods.

2.6 Depth-aided texture synthesis

References

[1] J. Alon and S. Sclaroff. Recursive estimation of motion and planar structure. In *cvpr*, page 2550, 2000.

[2] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. Dense disparity map estimation respecting image discontinuities: A pde and scale-space based approach. *Elsevier Signal Processing: Image Communication*, 13(1):3–21, March 2002.

[3] J. Ballè and M. Wien. Extended texture prediction for h.264/avc intra coding. In *IEEE International Conference on Image Processing*, volume 6, pages 93 – 96, October 2007.

[4] D. Baron, S. Sarvotham, and R. G. Baraniuk. Bayesian compressive sensing via belief propagation. Technical report, available at http://arxiv.org: 0812.4627v2.pdf, June 2009.

[5] M. Bastan, U. Gudukbay, and O. Ulusoy. Segmentationbased extraction of important objects from video for object-based indexing. In *IEEE International Conference On Multimedia and Expo*, volume 1, pages 1357 – 1360, June 2008.

[6] J. O. Berger. *Statistical decision theory and Bayesian analysis*. Springer-Verlag, 2nd edition, 1985.

[7] J. M. Bernardo and A. F. M. Smith. *Bayesian Theory*. Wiley & Sons Ltd., 1994.

[8] P. Bloomfield and W. L. Steiger. *Least Absolute Deviations: Theory, Applications and Algorithms*. Birkhäuser Boston, 1983.

[9] T. Blumensath and M. E. Davies. Iterative thresholding for sparse approximations. *Journal of Fourier Analysis and Applications*, 14(5-6):629–654, December 2008.

[10] J. Bobin, J.-L. Starck, J. M. Fadili, Y. Moudden, and D. L. Donoho. Morphological component analysis: An adaptive thresholding strategy. *IEEE Trans. On Image Processing*, 16(11):2675–2681, 2007.

[11] P. Bouthemy, F. Chaumette, V. Moreau, and E. Marchand. Robust real-time visual tracking using a 2d-3d model- based approach. In *International Conference On Computer Vision*, volume 1, pages 262–268, September 1999.

[12] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679 – 698, November 1986.

[13] G. Casella and C. P. Robert. *Monte Carlo Statistical Methods*. Springer-Verlag New York, LLC, 2nd edition, 2004.

[14] J. Chang, C. Cheng, S. Chien, and L. Chen. Relative depth layer extraction for monoscopic video by use of multidimensional filter. In *IEEE International Conference On Multimedia and Expo*, pages 221–224, July 2006.

[15] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 168–172, 1994.

[16] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1998.

[17] V. Cheung and B. Frey. Video epitomes. *International Journal Of Computer Vision*, 76(2), February 2008.

[18] P. L. Combettes and J.-C. Pesquet. A proximal decomposition method for solving convex variational inverse problems. *Inverse problems*, 24(6):1–31, 2008.

[19] W. Dai and O. Milenkovic. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. on Information Theory*, 55(5):2230–2249, May 2009.

[20] I. Daribo, M. Kaaniche, W. Miled, M. Cagnazzo, and B. Pesquet-Popescu. Dense disparity estimation in multiview video coding. In *IEEE International Workshop on Multimedia Signal Processing*, pages 1–6, October 2009.

[21] I. Daubechies, M. Defrise, and C. DeMol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on pure and applied mathematics*, 57(11):1413–1457, 2004.

[22] D. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization. *Proc. Nat. Acad. Sci.*, 100(5):2197–2202, March 2003.

[23] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. On Information Theory*, 41(3):613–627, May 1995.

[24] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. Technical report, Stanford University, March 2006.

[25] M. Dubuisson and A. Jain. Contour extraction of moving objects in complex outdoor scenes. *International Journal Of Computer Vision*, 14(1), January 1995.

[26] M. Elad, J-L Starck, P. Querre, and D. L. Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis (mca). *Journal on Applied and Computational Harmonic Analysis (ACHA)*, 19:340–358, November 2005.

[27] M.J. Fadili and J-L. Starck. An em algorithm for sparse representation-based image inpainting. In *Proc. IEEE Int'l Conference on Image Processing (ICIP)*, volume 2, pages II – 61–4, September 2005.

[28] J.-J. Fuchs. On sparse representations in arbitrary redundant bases. *IEEE Trans. On Information Theory*, 50(6):1341–1344, June 2004.

[29] J.J. Fuchs. On the application of the global matched filter to doa estimation with uniform circular arrays. *IEEE Trans. On Signal Processing*, 49(4):702–709, 2001.

[30] A. P Gee, D. Chekhlov, W. Mayol, and A. Calway. Discovering planes and collapsing the state space in visual slam. In *British Machine Vision Conference (BMVC)*, 2007.

[31] I. F. Gorodnitsky and B. D. Bhaskar. Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm. *IEEE Trans. On Signal Processing*, 45(3):600–616, March 1997.

[32] R. Gribonval and M. Nielsen. Sparse representations in unions of bases. *IEEE Trans. On Information Theory*, 49(12):3320–3325, December 2003.

[33] R. Gribonval and P. Vandergheynst. On the exponential convergence of matching pursuits in quasi-incoherent dictionaries. *IEEE Trans. On Information Theory*, 52(1):255–261, 2006.

[34] O. G. Guleryuz. Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising: Part i - theory. *IEEE Trans. On Image Processing*, 15:555–571, 2004.

[35] O. G. Guleryuz. Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising: Part ii - adaptive algorithms. *IEEE Trans. On Image Processing*, 15:555–571, 2004.

[36] L. He and L. Carin. Exploiting structure in wavelet-based bayesian compressive sensing. *IEEE Trans. On Signal Processing*, 57(9):3488–3497, September 2009.

[37] J. A. Högbom. Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics Supplement*, 15:417–426, 1974.

[38] X. Huang, L. Wang, J. Huangk, D. Li, and M. Zhang. A depth extraction method based on motion and geometry for 2d to 3d conversion. In *Third International Symposium on Intelligent Information Technology Application*, volume 3, pages 294–298, November 2009.

[39] P. J. Huber. Projection pursuit. *The Annals of Statistics*, 13:435–475, 1985.

[40] C. Çigla and A. Alatan. Multi-view dense depth map estimation. In *International Conference on Immersive Telecommunications*, volume 2, pages 1–6, May 2009.

[41] S. Ji, Y. Xue, and L. Carin. Bayesian compressive sensing. *IEEE Trans. On Signal Processing*, 56(6):2346–2356, June 2008.

[42] N. Jojic, B. Frey, and A. Kannan. Epitomic analysis of appearance and shape. In *International Conference On Computer Vision*, volume 2, page 34, October 2003.

[43] L. K. Jones. On a conjecture of huber concerning the convergence of projection pursuit regression. *Annals of statistics*, 15(2):880–882, 1987.

[44] P. Kauff, N. Atzpadin, C. Fehn, M. Muller, O. Schreer, and A. Smolic. Depth map creation and image-based rendering for advanced 3dtv services providing interoperability and scalability. *Elsevier Signal Processing: Image Communication*, 22(2):217–234, February 2007.

[45] N. G. Kingsbury and T. H. Reeves. Overcomplete image coding using iterative projection-based noise shaping. In *Proc. IEEE Int'l Conference on Image Processing (ICIP)*, volume 3, pages 597–600, 2002.

[46] P. Kornprobst, R. Deriche, and G. Aubert. Nonlinear operators in image restoration. In *In Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 325–331, June 1997.

[47] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. On Information Theory*, 47(2):498–519, February 2001.

[48] M. Lhuillier and L. Quan. Reconstruction quasi-dense et modèles 3D à partir d'une séquence d'images. 2004.

[49] Z. Ma, J. Luo, P. Yin, C. Gomila, and Y. Wang. Smoothed reference inter-layer texture prediction for bit depth scalable video coding. In *SPIE Visual Communications and Image Processing Conference*, volume 7543, January 2010.

[50] D. J. C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, May 1992.

[51] A. Malika and T. Choi. A novel algorithm for estimation of depth map using image focus for 3d shape recovery in the presence of noise. *Pattern Recognition*, 41(7):2200–2225, July 2008.

[52] A. Malika and T. Choi. Consideration of illumination effects and optimization of window size for accurate calculation of depth map for 3d shape recovery. In *International Conference on Pattern Recognition*, volume 40, pages 154–170, January 7.

[53] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. On Signal Processing*, 41(12):3397–3415, December 1993.

[54] A. Martin, J-J. Fuchs, C. Guillemot, and D. Thoreau. Sparse representations for image prediction. In *Proc. European Signal Processing Conference (EUSIPCO), Poznan, Poland.*, September 2007.

[55] W. Miled, J. Pesquet, and M. Parent. Disparity map estimation using a total variation bound. In *Canadian Conference on Computer and Robot Vision*, pages 48–55, June 2006.

[56] D. Needell and J. A. Tropp. Cosamp: iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, May 2009.

[57] K. Palaniappan, F. Bunyak, S. Nath, and G. Seetharaman. Flux tensor constrained geodesic active contours with sensor fusion for persistent object tracking. *Journal Of Multimedia*, 2(4):20–33, August 2007.

[58] Y. C. Pati, R. Rezaiifar, Y. C. Pati R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proc. Asilomar Conference on Signals, Systems, and Computers*, pages 40–44, 1993.

[59] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[60] G. Rath and C. Guillemot. A complementary matching pursuit algorithm for sparse approximation. In *Proc. European Signal Processing Conference (EUSIPCO)*, 2008.

[61] G. Rath and C. Guillemot. Sparse approximation with an orthogonal complementary matching pursuit algorithm. In *Proc. IEEE Int'l Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 3325 – 3328, 2009.

[62] L. Rebollo-Neira and D. Lowe. Optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9(4):137–140, April 2002.

[63] O. G. Sezer, O. Harmanci, and O. G. Guleryuz. Sparse orthonormal transforms for image compression. In *Proc. IEEE Int'l Conference on Image Processing (ICIP), San Diego, CA.*, October 2008.

[64] D. De Silva, W. Fernando, and S. Yasakethu. Object based coding of the depth maps for 3d video coding. *IEEE Transactions on Consumer Electronics*, 55(3):1699 – 1706, August 2009.

[65] R. A. Smith, A. W Fitzgibbon, and A. Zisserman. Improving augmented reality using image and scene constraints. In *British machine vision conference, Nottingham, UK*, pages 295 – 304, 1999.

[66] R. Szeliski and P. Torr. Geometrically constrained structure from motion: Points on planes. *3D Structure from Multiple Images of Large-Scale Environments*, pages 171–186, 1998.

[67] T. Tan, C. Boon, and Y. Suzuki. Intra prediction by template matching. In *IEEE International Conference on Image Processing*, pages 1693 – 1696, October 2006.

[68] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[69] M. E. Tipping and A. C. Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proc. Int'l Workshop on Artificial Intelligence and Statistics*, 2003.

[70] J. Tropp. Greed is good : algorithmic results for sparse approximation. *IEEE Trans. On Information Theory*, 50(10):2231–2242, October 2004.

[71] D. Tschumperlé. *PDE-based regularization of multivalued images and applications.* PhD thesis, University of Nice-Sophia Antipolis/France, 2002.

[72] M. Turkan and C. Guillemot. Sparse approximation with adaptive dictionary for image prediction. In *Proc. IEEE Int'l Conference on Image Processing (ICIP)*, pages 25–28, November 2009.

[73] H. Wang, Y. Wexler, . Ofek, and H. Hoppe. Factoring repeated content within and among images. *ACM Transactions on Graphics*, 27(3), August 2008.

[74] J. Warrell and S. Prince. Labelfaces: Parsing facial features by multiclass labeling with an epitome prior. In *IEEE International Conference on Image Processing*, pages 2481–2484, August 2009.

[75] J. Warrell, S. Prince, and A. Moore. Epitomized priors for multi-labeling problems. In *International Conference on Pattern Recognition*, June 2009.

[76] J. Weickert. Theoretical foundations of anisotropic diffusion in image processing. *Computing Supplement*, 11:221–236, 1996.

[77] J. Weickert. A review of nonlinear diffusion filltering. scale-space theory in computer vision. *Lecture Notes in Comp. Science (Springer, Berlin)*, 1252:3–28, 1997.

[78] J. Weickert. *Anisotropic diffusion in image processing.* Teubner-Verlag, 1998.

[79] J. Weickert. Coherence-enhancing diffusion filtering. *International Journal of Computer Vision*, 32(2-3):111–127, 1999.

[80] M. Yang, N. Ahuja, and M. Tabb. Extraction of 2d motion trajectories and its application to hand gesture recognition. In *IEEE Transaction on Pattern Analysis and Machine Intelligence*, volume 24, pages 1061–1074, August 2002.

[81] H. Zayyani, M. Babaie-Zadeh, and C. Jutten. An iterative bayesian algorithm for sparse component analysis in presence of noise. *IEEE Trans. On Signal Processing*, 57(11):4378–4390, November 2009.