

# Morphological Component Analysis for the Inpainting of Grazing Incidence X-Ray Diffraction Images Used for the Structural Characterization of Thin Films

## Analyse en Composantes Morphologique pour les Retouches d'Images de Diffraction des Rayons X en Incidence Rasante Utilisés pour la Caractérisation Structurale des Couches Minces

George Tzagkarakis<sup>1\*</sup>, Eleni Pavlopoulou<sup>2</sup>, Jalal Fadili<sup>3</sup>, Georges Hadziioannou<sup>2</sup> and Jean-Luc Starck<sup>1</sup>

*1 Commissariat à l'Énergie Atomique (CEA), Centre de Saclay, F-91191 Gif-Sur-Yvette cedex, France*

*2 Université Bordeaux, LCPO, UMR 5629, F-33600 Pessac, France*

*3 GREYC CNRS, UMR 6072, ENSICAEN, F-14050 Caen cedex, France*

*e-mail: georgios.tzagkarakis@cea.fr, eleni.pavlopoulou@enscbp.fr, Jalal.Fadili@greyc.ensicaen.fr, hadzii@enscbp.fr, jstarck@cea.fr*

\*Corresponding author

**Résumé** — La diffraction des rayons X en incidence rasante (GIXD) est une technique de caractérisation souvent utilisée dans l'étude de la structure des couches minces. En ce qui concerne les films organiques, le confinement du film sur le substrat conduit à des structures GIXD anisotropes à deux dimensions, telles celles observées pour les films à base de polythiophène utilisés comme couches actives dans les applications photovoltaïques. D'éventuels dysfonctionnements des détecteurs utilisés peuvent altérer la qualité des images acquises, affectant ainsi le processus d'analyse et l'information structurale qui en est dérivée. Motivés par le succès des Analyses en Composantes Morphologique (MCA) en traitement d'images, nous nous attaquons dans cette étude au problème de la récupération de l'information manquante dans les images GIXD due à un dysfonctionnement potentiel des détecteurs. Nous montrons d'abord que les structures géométriques présentes dans les images GIXD peuvent être représentées de façon parcimonieuse en utilisant une combinaison de transformées redondantes, à savoir la transformée en curvelets et en ondelettes non-décimée. Ceci permet une description simple et compacte de l'information contenue dans ces images. Ensuite, l'information manquante est récupérée en appliquant la MCA dans un cadre d'"inpainting", en exploitant la représentation parcimonieuse des données GIXD dans ces deux domaines transformés. L'évaluation expérimentale montre que l'approche proposée est très efficace pour récupérer les informations manquantes lorsqu'elles sont aléatoirement distribuées sur les pixels de l'image, ou lorsque des rangées entières sont manquantes, même lorsque la moitié du nombre total de pixels est affectée. Ce résultat indique que la MCA peut être appliquée pour remédier aux potentiels problèmes liés à la performance des détecteurs lors de l'acquisition, ce qui est d'une

grande importance dans les expériences synchrotron, puisque le temps de faisceau alloué aux utilisateurs est extrêmement limité et que toute défaillance technique peut être préjudiciable pour le cours du projet expérimental. En outre, nos résultats permettent de réduire le temps d'acquisition ou d'éviter la répétition des mesures, qui donne plus de valeur à l'approche proposée.

**Abstract** — *Grazing incidence X-ray diffraction (GIXD) is a widely used characterization technique, applied for the investigation of the structure of thin films. As far as organic films are concerned, the confinement of the film to the substrate results in anisotropic 2-dimensional GIXD patterns, such those observed for polythiophene-based films, which are used as active layers in photovoltaic applications. Potential malfunctions of the detectors utilized may distort the quality of the acquired images, affecting thus the analysis process and the structural information derived. Motivated by the success of Morphological Component Analysis (MCA) in image processing, in this study we tackle the problem of recovering the missing information in GIXD images due to potential detector's malfunction. First we show that the geometrical structures, which are present in the GIXD images can be represented sparsely by means of a combination of over-complete transforms, namely, the curvelet and the undecimated wavelet transform, resulting in a simple and compact description of their inherent information content. Then, the missing information is recovered by applying MCA in an inpainting framework, by exploiting the sparse representation of GIXD data in these two over-complete transform domains. The experimental evaluation shows that the proposed approach is highly efficient in recovering the missing information in the form of either randomly burned pixels, or whole burned rows, even at the order of 50% of the total number of pixels. Thus, our approach can be applied for healing any potential problems related to detector performance during acquisition, which is of high importance in synchrotron-based experiments, since the beamtime allocated to users is extremely limited and any technical malfunction could be detrimental for the course of the experimental project. Moreover, the non-necessity of long acquisition times or repeating measurements, which stems from our results adds extra value to the proposed approach.*

## 1 INTRODUCTION

Since the introduction of X-ray diffraction for the investigation of the unit cell of crystals in the beginning of the 20th century, X-rays turned to be an indispensable characterization tool that probes the structure of a wealth of materials, ranging from inorganic crystals and powders to organic small molecules and polymers, up to proteins and other biological samples [1, 2]. During the last 20 years, the need for characterizing the structure of 2-dimensional (2-D) systems, such as thin films that are of particular interest in microelectronics and other nanosciences, led to the emergence of grazing incidence X-ray diffraction (GIXD) [3]. GIXD exploits the principles of X-ray diffraction, however the X-ray beam probes the sample at a very small incident angle, typically below  $1^\circ$ , allowing, thus, an effective increase of the penetration depth and, consequently, the interaction of X-rays with a bigger part of the sample. On top of that, the physical confinement of one side of the film to the substrate may result in preferentially oriented scatterers that give rise to non-isotropic diffraction patterns. Such patterns are recorded when studying organic thin films, like polythiophene-based films, which are extensively used in organic photovoltaics [4].

In order to acquire and exploit all information contained in the anisotropic diffraction patterns, 2-D detectors are uti-

lized. Data acquisition is followed by sophisticated data analysis, which premises the high quality of the recorded images. For this, synchrotron-based experiments are preferred, since the high X-ray flux provided in a synchrotron facility allows for increased signal-to-noise ratio. However, due to large demand for synchrotron beamtime, the time allocated to users is extremely limited and the success of the experiments is imperative. In the quest for an alternative way of recovering information that is partially recorded in the GIXD images due to technical problems during the experiment or less acquisition time than the optimum, we turn to image processing theories.

In image processing, finding an efficient and compact representation of the data under consideration is of major importance in several distinct tasks, such as, compression, denoising, and restoration, to name a few. In the quest for a suitable transform, sparsity of the representation was recognized as a key requirement in seeking simplifying operations [5, 6]. Specifically, the design of over-complete redundant representations is now at the core of many state-of-the-art algorithms used in image compression [7], denoising [8], deconvolution [9], and restoration [10]. In each case, an image is represented as a linear combination of atoms from a dictionary, where the number of atoms is much larger than the original image dimension. Due to the redundancy, there

exist numerous ways to represent the image, with our preference being towards the sparsest one, that is, with the fewest non-zero components as being the simplest.

Focusing on images, another important task is to decompose the data into elementary building blocks. The successful separation of the image content is crucial for its effective analysis, as well as for tasks, such as, image enhancement, compression, and synthesis. Numerous methods have been proposed for the solution of the image separation problem, especially in the frameworks of blind source separation (BSS) [11] and independent component analysis (ICA) [12]. In addition, the need to recognize structures of different sizes in a given image, makes it impossible to define a priori an optimal resolution for analyzing it. Multiresolution decomposition was introduced as a simple hierarchical framework for interpreting the image information, where at different resolutions the details of an image generally characterize different physical structures.

In this direction, multiscale methods have become popular image processing tools in the last couple of decades, especially with the development of wavelets [13]. While the discrete wavelet transform (DWT) was implemented successfully in image compression, the results were far from optimal for other image processing tasks, such as, deconvolution, detection and filtering. The dual-tree complex wavelet transform [14] was introduced as a valuable enhancement of the traditional DWT, which is nearly shift invariant and, in higher dimensions, it is characterized by directional selectivity. One of the reasons to focus on the design of new redundant representations was the need to preserve the shift-invariance property, while also approximating more closely the continuous analogue.

For this purpose, several novel tailored multiscale and multidirectional redundant transforms have been introduced in the literature, including, among others, the undecimated discrete wavelet transform (UDWT) [15], the curvelet transform [16], the contourlet transform [17] and the bandlet transform [18]. Most importantly, each of these transforms adapts to specific characteristics and structures in a given image, thus, yielding highly sparse representations in the presence of the corresponding structures, in a non-linear approximation scheme. For instance, sparse approximations of piecewise smooth images with point singularities are obtained using the UDWT, which is efficient in capturing roughly isotropic features. However, this is no longer optimal in case of piecewise smooth images with singularities along smooth curves or edges. Such images are approximated more efficiently using the curvelet transform, which is highly anisotropic and thus exhibits high directional selectivity by defining an adapted multiscale geometry.

Apart from decomposing an image in terms of the physical size and orientation of its structural content via a multiscale transform, natural images are often considered to consist of homogeneous regions and oscillating patterns (*e.g.*,

texture and noise). In this later case, it is also of high importance to be able to decompose an image into its constituent components. Several methods [19–22] have been introduced for decomposing a given image into a component with bounded variation, which holds the geometrical information, and an oscillating component, which corresponds to the textural information. A recent work [23] generalized the previous approaches providing a method for decomposing an image in more than two components, while being also able to handle data corrupted with a linear operator and a non-necessarily Gaussian noise.

Tackling the decomposition problem from a different perspective, the use of sparsity as a desired property to rely on was recognized earlier [24, 25]. In this framework, *morphological component analysis* (MCA) [26] is a recent novel technique, which exploits the sparse representation of structured data in large, generally over-complete, transform domains (or dictionaries) to separate them in a set of distinct components based on their difference in morphology. The method is based on the assumption that for each morphological feature to be separated, there exists a suitable transform that enables its reconstruction via a sparse representation, while this transform is highly inefficient in representing the other morphological features. For instance, it has been shown that MCA can be used to separate the texture from the piecewise smooth component of a given image, by noticing that the former is characterized well using local cosine functions, while the latter may be well represented using curvelets [27]. The MCA algorithm has been employed successfully for the analysis of data in several areas, such as, video analysis [28], astrophysics [29], and medical imaging [30, 31], while it was extended recently in the framework of *image inpainting* [32], enabling the treatment of problems where parts of the image are missing or corrupted.

Motivated by the success of MCA in signal and image processing, the purpose of this study is to exploit the effectiveness of modern over-complete transforms and extend the applicability of MCA to the analysis of 2-D GIXD data. To the best of our knowledge, this is the first study that bridges the gap between the analysis of diffraction data and state-of-the-art image processing techniques. More specifically, first we show that the geometrical structures, which are present in GIXD images for the specific type of thin films considered in this study, can be represented efficiently by means of a combination of over-complete transforms, namely, the UDWT and the curvelet transform, resulting in a simple and compact description of their inherent structures. Then, we tackle the important problem of recovering the missing information in GIXD images, *e.g.*, due to potential malfunction of the detectors, by applying MCA in an inpainting framework.

The experimental evaluation using high-resolution GIXD images shows that the proposed approach is highly efficient in recovering missing information in the form of either randomly burned pixels, or whole burned rows, even at the

order of 50% of the total number of pixels (which is anyway an extreme practical scenario). This missing information may inhibit the post-processing of GIXD images and data evaluation. Data analysis is mainly based on reduction of the 2-D images into 1-D intensity patterns. This is done either by integrating the diffracted intensity (throughout the whole image or on specific sectors) or by plotting the intensity across the meridian and/or the horizon. In case of missing information due to burned pixels, or inaccurate statistics related to a small number of measurements due to reduced acquisition times, incomplete intensity 1-D plots will be derived. This entails the danger of allowing for data misinterpretation and derivation of false scientific outcomes, which are highly sensitive to the form of the intensity patterns. The added value of our approach stems, thus, from its application in healing potential problems related to detector performance during image recording. Additionally, MCA weakens the necessity for long acquisition times or repeating experiments. All these merits are of high importance in synchrotron-based experiments, since the beamtime allocated to users is extremely limited and any technical malfunction could be detrimental for the course of the experimental project.

The rest of the paper is organized as follows: Section 2 provides the principles of GIXD technique, experimental details on image acquisition and a brief description of the thin films under study. Section 3 overviews the basic concepts of MCA applied in the framework of inpainting. Section 4 studies the efficiency of a combination of over-complete redundant transforms, namely, the UDWT and the curvelet transform, in decomposing the given GIXD images into their constituent morphological components. Then, an evaluation of the performance of MCA, and its comparison with the performance of classical inpainting methods, in recovering the missing information in the form of individually burned pixels, or whole burned lines, distributed uniformly at random over the whole image area, is performed. Finally, Section 5 concludes and gives directions for future work.

## 2 GIXD MEASUREMENTS AND MATERIALS UNDER STUDY

GIXD experiments were performed on the Dutch-Belgian Beamline (DUBBLE CIG), station BM26B, at the European Synchrotron Radiation Facility (ESRF), Grenoble, France. In GIXD, the X-ray beam probes the sample at the grazing geometry and the diffracted intensity is recorded by a 2-D position sensitive detector, which is placed after the sample typically at a distance of around 20–40 cm (Fig. 1(a)). The energy of the X-rays was 12 eV and the angle of incidence was set at 0.15°. The diffracted intensity was recorded by a Frelon CCD camera and it was normalized by the incident photon flux and the acquisition time. Each pixel monitors the intensity as a function of the scattering vector  $\mathbf{q}$  that is

defined with respect to the center of the incident beam and has a magnitude of  $q = (4\pi/\lambda) \sin \theta$ , where  $2\theta$  is the scattering angle and  $\lambda$  is the wavelength of the X-ray beam.

In order to analyze the GIXD data several *linecuts* are performed. Usually, the 2-D images are radially averaged around the centre of the primary beam, which gives the 1-D plot of the intensity as a function of  $q$ . In a second approach, linecuts across the meridian and/or the horizon are extracted, which provide the intensity plots across the out-of-plane ( $q_z$ ) and in-plane ( $q_{xy}$ ) directions, respectively (Fig. 1(b)). Finally, azimuthal scans can be performed around a  $q$ -range of special interest, herein at around  $q = 0.37 \text{ \AA}^{-1}$  (the highest intensity ring apparent in the test images shown in Fig. 2). In this case, the diffracted intensity is plotted as a function of the polar angle  $\chi$ , which is defined with respect to the normal to the substrate (Fig. 1(b)).

The GIXD images presented herein were collected from polythiophene-based films. Poly(3-hexyl thiophene), P3HT, is a semi-conducting polymer that is widely used as a donor material for the fabrication of organic photovoltaic devices. For our study, P3HT was blended with [6,6]-phenyl-C<sub>61</sub>-butyric acid methyl ester, PCBM, a fullerene-based organic small molecule, in equal masses in chlorobenzene and 100 nm thick films were spin-coated on indium tin oxide, ITO, substrates. Two films were prepared and annealed at 160°C for 10 min (image X1) and 20 min (image X2), respectively, in order to induce small morphological changes that will result in small changes in the two GIXD images. We opt to use polythiophene-based films as a case study due to the non-isotropic features that are apparent in their GIXD patterns. We note that no discussion on the structural characteristics of the two films will be presented in this work, since our objective is to stress the potential use of image processing techniques in physico-chemical applications.

## 3 IMAGE INPAINTING USING MCA

In this section, the main principles of MCA are introduced in the framework of image inpainting. Since the performance of the MCA algorithm relies, and depends highly, on the degree of sparsity achieved for the analyzed data, we start by introducing briefly the concept of sparsity in transforms. Then, the process of decomposing an image in a set of distinct morphological components by exploiting sparse representations will be described, along with the extension of MCA as a solution to the image inpainting problem.

### 3.1 Sparse recovery in a transform domain

In the following, we consider for convenience the case of square  $N \times N$  images, although the proposed approach is extended straightforwardly in the general non-square case.

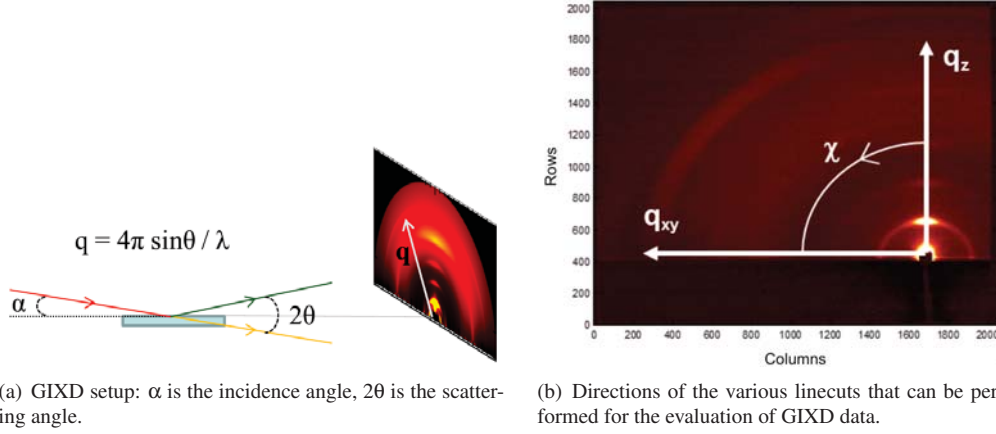


Figure 1  
Schematic of the GIXD setup.

Let  $\mathbf{X} \in \mathbb{R}^{N \times N}$  be the given image,  $\mathcal{T}\{\cdot\}$  denote a sparsifying transformation, and  $\mathbf{c} \in \mathbb{R}^L$  be the vector of transform coefficients, that is,  $\mathbf{c} = \mathcal{T}\{\mathbf{X}\}$ . Although  $\mathcal{T}\{\cdot\}$  can be linear or non-linear in the general case, however, for computational and implementation purposes, the linear transformations are highly promoted for carrying out several signal and image processing tasks (*e.g.*, restoration, denoising, deconvolution). Thus, in the subsequent analysis the linear case is considered. We note the following two remarks concerning the sparsifying transform:

- (i) In case of linearity, both the forward,  $\mathcal{T}\{\cdot\}$ , and the inverse transform,  $\mathcal{T}^{-1}\{\cdot\}$ , can be expressed in matrix form, while the transform coefficients are obtained by means of simple matrix-vector multiplications.
- (ii) In case of an over-complete redundant representation the number of transform coefficients is larger than the original image dimension, that is,  $L > N^2$ .

An image  $\mathbf{X}$  is said to be *K-sparse* in the transform domain  $\mathcal{T}$  if the coefficient vector  $\mathbf{c}$  has exactly  $K \ll L$  non-zero components. However, in practice a natural image  $\mathbf{X}$  is not strictly sparse but *compressible*, which means that the magnitude of the re-ordered transform coefficients decays at a power law,

$$|c_{r_i}| \leq C i^{-1/\delta}, \quad i = 1, \dots, L, \quad (1)$$

where  $c_{r_i}$  denotes the  $i$ -th sorted coefficient,  $C$  is a real positive constant and  $\delta > 0$  controls the rate of decay. Thanks to the rapid decay of their coefficients, compressible images are well-approximated by  $K$ -sparse images, that is, by keeping the  $K$  most significant (largest magnitude) coefficients  $c_{r_i}, i = 1, \dots, K$ . Following a typical transform coding-based approach, a way to choose the value of  $K$  is given by keeping the  $K$  transform coefficients, which contain a predetermined

percentage of their total energy. In practice, the value of this percentage is set in a heuristic way.

In contrast to a complete transform basis, where  $\mathbf{X}$  has an exact representation, an over-complete redundant transform yields several exact representations. In the later case, these representations are not equally interesting in terms of modeling and feature extraction. In particular, the representation of  $\mathbf{X}$  by means of highly sparse coefficient vectors  $\mathbf{c}$  is promoted, since it usually leads to a more concise and possibly more interpretable representation of  $\mathbf{X}$ .

However, selecting the smallest subset of transform basis functions (also called *atoms*) from a large redundant set, which will be combined linearly to reproduce the salient features of a given image, is a hard combinatorial problem. In a formal way, the requirement for obtaining the sparsest representation  $\mathbf{c}$  for a given image  $\mathbf{X}$  in a transform domain  $\mathcal{T}$  is written as follows,

$$\min_{\mathbf{c} \in \mathbb{R}^L} \|\mathbf{c}\|_0 \quad \text{subject to} \quad \mathbf{X} = \mathcal{T}^{-1}\{\mathbf{c}\}, \quad (2)$$

where  $\|\mathbf{c}\|_0$  denotes the  $\ell_0$ -pseudo-norm<sup>1</sup> defined as the number of non-zero elements of the vector  $\mathbf{c}$ . The difficulty in addressing (2) is that this optimization problem is highly non-smooth and non-convex, while it has been also proved to be NP-hard in terms of computational complexity [33].

On the other hand, the relaxation of (2) by replacing the  $\ell_0$ -pseudo-norm with the  $\ell_1$ -norm reduces to a linear program, and hence it can be solved in polynomial time,

$$\min_{\mathbf{c} \in \mathbb{R}^L} \|\mathbf{c}\|_1 \quad \text{subject to} \quad \mathbf{X} = \mathcal{T}^{-1}\{\mathbf{c}\}, \quad (3)$$

where the  $\ell_1$ -norm of  $\mathbf{c}$  is given by  $\|\mathbf{c}\|_1 = \sum_{i=1}^L |c_i|$ . Moreover, it was shown that under certain conditions, the solutions of (2) and (3) are identical [34, 35].

<sup>1</sup>Despite the fact that  $\ell_0$  is a pseudo-norm, the improper term “norm” is often used, too.

Several pursuit algorithms with empirical success have been proposed for the solution of (3), among them, the *greedy matching pursuit* (MP) [36], the *basis pursuit* (BP) [37], and their variants. Nevertheless, in applications involving large data sets, such as, the high-resolution GIXD images we deal with, MP or BP algorithms are computationally intense.

On the other hand, as discussed in [27], a single basis is often not well-adapted to large classes of highly structured data such as “natural images”. Furthermore, over the past years, new tools have emerged from modern computational harmonic analysis, such as, wavelets [5], ridgelets [38], and curvelets [39], to name a few. It is quite tempting to combine several representations to build a larger dictionary of waveforms that will enable the sparse representation of larger classes of signals.

Morphological Component Analysis (MCA) [26] was introduced recently aiming at decomposing signals in (generally overcomplete) dictionaries made of a union of bases. MCA serves as a fast alternative to other algorithms in the sparsity literature, like the ones mentioned above, where the solution of the corresponding minimization problem necessitates to deal with unknowns, that is, the sparse coefficient vectors, living in a high-dimensional space (getting larger as the dictionaries become more redundant). In contrast, MCA solves a minimization problem in terms of the morphologically distinct components directly, whose dimension is less than or equal to the dimension of the corresponding transformed version. The following sections introduce in brief the main concepts of MCA, as well as its extension in the framework of image inpainting, to be exploited for recovering the missing information in our GIXD data due to detectors malfunction.

### 3.2 Overview of MCA

In the subsequent analysis, we assume that a given image  $\mathbf{X}$  is modeled as a linear combination of  $S$  images (components) with different morphologies,  $\mathbf{X} = \sum_{s=1}^S \mathbf{X}_s$ . The fundamental assumption of MCA is morphological diversity, which relies on the sparsity of those morphological components in specific bases. In other words, MCA is based on the existence of a set of transforms (or a dictionary of bases)  $\{\mathcal{T}_1, \dots, \mathcal{T}_S\}$  such that the  $s$ -th component  $\mathbf{X}_s$  is sparsely represented in  $\mathcal{T}_s$ , while its representation in the other transform domains  $\mathcal{T}_{s'}, s' \neq s$ , is not sparse. This is ensured with high probability by an increased incoherence between the distinct dictionaries. The problem to be solved is the separation of the linear mixture  $\mathbf{X}$  into its constituent morphological components  $\mathbf{X}_s$ , relying on the discriminative power of the distinct transforms  $\mathcal{T}_s, s = 1, \dots, S$ .

By extending (3) in the multi-component case, the problem of recovering the corresponding sparse coefficient vec-

tors  $\{\mathbf{c}_s \in \mathbb{R}^{L_s}\}_{s=1, \dots, S}$  is expressed as follows,

$$\min_{\mathbf{c}_1, \dots, \mathbf{c}_S} \sum_{s=1}^S \|\mathbf{c}_s\|_1 \quad \text{subject to} \quad \mathbf{X} = \sum_{s=1}^S \mathcal{T}_s^{-1} \{\mathbf{c}_s\}. \quad (4)$$

Notice also that, generally, the coefficient vectors  $\mathbf{c}_s$  can be of different dimension  $L_s$  depending on the corresponding transform. For this reason we also keep denoting a linear transform as a general operator  $\mathcal{T}_s$ , instead of using a matrix notation  $\mathbf{T}_s$ , to avoid any inconvenience due to the potentially different dimensions among the transform coefficient vectors corresponding to distinct morphological components.

The solution of (4) should be expected to give a truly sparse decomposition if the image  $\mathbf{X}$  is indeed composed solely of the morphological components  $\mathbf{X}_s, s = 1, \dots, S$ , and thus it can actually be represented sparsely in terms of the transforms  $\mathcal{T}_s$ . However, in a real-world scenario the assumption of an exact decomposition does not hold in general. For this purpose, we compensate by reformulating the constrained optimization problem (4) as an unconstrained regularized one. Moreover, the increased memory and computational costs when we work directly with the coefficient vectors  $\mathbf{c}_s$ , whose dimensions can be much higher than the dimension of the original image in case of highly redundant transforms, are resolved by solving the optimization problem using the morphological components  $\mathbf{X}_s$  as the unknowns. The above two considerations result in the following minimization problem solved by the MCA algorithm [26]:

$$\min_{\mathbf{X}_1, \dots, \mathbf{X}_S} \sum_{s=1}^S \|\mathcal{T}_s \{\mathbf{X}_s\}\|_1 + \tau \left\| \mathbf{X} - \sum_{s=1}^S \mathbf{X}_s \right\|_2^2, \quad (5)$$

where  $\tau > 0$  is a regularization parameter, which controls the amount of distortion in representing  $\mathbf{X}$  in terms of the morphological components  $\mathbf{X}_s$ . This problem has a quadratic programming structure, for which efficient solvers exist, with the *soft-thresholding* being among them as it is described in the following section.

### 3.3 Extending MCA for image inpainting

The MCA algorithm can be easily extended in the framework of image inpainting, where the main requirement is the preservation of discontinuities (*e.g.*, edges and textures). In the following, we focus on the case where part of the original image information content is missing, in the form of occluded (“burned”) pixels. Let  $\mathbf{M}$  be a *binary mask*, where zeros indicate that the corresponding pixels in the original image  $\mathbf{X}$  have been occluded, while ones indicate valid pixels. Then, the observed image captured by the detector can be expressed as

$$\mathbf{X}_d = \mathbf{M} \circ \mathbf{X}, \quad (6)$$

where  $\mathbf{X}_d$  is the detected, possibly corrupted, image, and  $\circ$  denotes element-by-element multiplication (that is, for two matrices  $\mathbf{A}$ ,  $\mathbf{B}$  of the same dimensions we have that  $[\mathbf{A} \circ \mathbf{B}]_{ij} = [\mathbf{A}]_{ij} \cdot [\mathbf{B}]_{ij}$ ).

The standard optimization problem (5) solved by the MCA algorithm can be modified easily to account for inpainting the missing information as follows,

$$\min_{\mathbf{X}_1, \dots, \mathbf{X}_S} \sum_{s=1}^S \|\mathcal{T}_s\{\mathbf{X}_s\}\|_1 + \tau \left\| \mathbf{M} \circ \left( \mathbf{X} - \sum_{s=1}^S \mathbf{X}_s \right) \right\|_2^2. \quad (7)$$

Notice that the inclusion of the binary mask  $\mathbf{M}$  in the above objective function to be minimized prevents the sparse model we try to build from attempting to consider the invalid (missing) data. A solution to the optimization problem (7) can be obtained by employing the same iterative thresholding strategy as in the MCA algorithm. In particular, the *block-coordinate relaxation* method [41] is used. It is a fast numerical technique, which requires only the use of matrix-vector multiplications due to the linear assumption for the sparsifying transformations.

Regarding the choice of the regularization parameter  $\tau$ , which is also employed by the soft-thresholding operator used in the solution of (7), one solution is based on the following observation. At the early stages of the algorithm's execution, the estimation of the individual morphological components may be inaccurate because of the missing data. To overcome this drawback, one has to start by considering a large value of  $\tau$  in order to favor the data fidelity term (second term in (7)). The appropriate initialization of  $\tau$  is done in a rather heuristic way, with its value affecting the speed of convergence. In the present implementation, it is expressed as the minimum of maximal amplitude coefficients of the recorded image in each sparsifying transform domain. Then, the value of  $\tau$  is decreased monotonically (e.g., according to a linear or an exponential strategy as we employ here) in order to favor the sparsity-enforcing term (first term in (7)). Moreover,  $\tau$  is updated as a function of the estimated noise standard deviation (e.g., by employing a *median absolute deviation* (MAD) estimator) so as to reject noise. Algorithm 1 summarizes the main steps of MCA and gives the expressions for the regularization parameter  $\tau$  and its updating rule for solving the inpainting problem expressed by (7).

Concerning the convergence of this algorithm, MCA for image inpainting is based on an iterative thresholding process, as in the original MCA algorithm which has been proven to converge [26, 42]. As far as inpainting is concerned, the presence of the mask makes things not to be straightforward anymore. In [32], a sketch of a proof for the convergence of MCA in case of inpainting, along with the role and the effect of the mask, is provided. However, we emphasize that convergence here means that the sequence of iterates for the recovery of the individual morphological components converges, but there is not a guarantee on the properties of the minimizer with respect to the true sig-

---

**Algorithm 1** MCA for image inpainting
 

---

**Require:** Maximum iterations  $I_{\max}$ , initial threshold  $\tau^{(0)}$

**Initialize:** residual  $\mathbf{r}^{(0)} = \mathbf{0}$ , morphological components  $\{\mathbf{X}_s^{(0)} = \mathbf{0}\}_{s=1}^S$

**Set:**  $\lambda = \left( \frac{\lambda_{\max}}{\lambda_{\text{stop}} \cdot \sigma_n} \right)^{\frac{1}{I_{\max}}}$ ,

where  $\lambda_{\max} = \min \max_{s=1, \dots, S} |\mathcal{T}_s\{\mathbf{X}_d\}|$ ,  $\lambda_{\text{stop}}$  a constant depending on the noise level (set to a small value e.g.,  $10^{-6}$  in the noiseless case, and between 3 and 5 for noisy images),  $\sigma_n$  the estimated noise standard deviation (e.g., using MAD)

**while**  $\tau^{(t)} > \tau_{\min}$  **do**

  // Execute the following iteration to estimate each

  // component  $\mathbf{X}_s$  at iteration  $t$  by assuming all the

  // others  $\{\mathbf{X}_k\}_{k=1, k \neq s}^S$  are fixed:

**for**  $s = 1, \dots, S$  **do**

    - Update residual:

$$\mathbf{r}^{(t)} = \mathbf{X}_d - \sum_{k=1}^S \mathbf{X}_k^{(t-1)}$$

    - Estimate current transform coefficients  $\mathbf{c}_s^{(t)}$  and apply soft-thresholding with threshold  $\tau^{(t)}$ :

$$\mathbf{c}_s^{(t)} = \Delta_{\tau^{(t)}}(\mathcal{T}_s\{\mathbf{M} \circ \mathbf{r}^{(t)} + \mathbf{X}_s^{(t-1)}\})$$

    - Update morphological component from the selected transform coefficients:

$$\mathbf{X}_s^{(t)} = \mathcal{T}_s^{-1}\{\mathbf{c}_s^{(t)}\}$$

**end for**

  - Update threshold using an exponential decay rule:

$$\tau^{(t)} = \tau^{(t-1)} \cdot \lambda$$

**end while**

\* Soft thresholding operator:  $\Delta_{\tau}(c) = \begin{cases} 0 & , c < \tau \\ c - \tau \cdot \text{sign}(c) & , c \geq \tau. \end{cases}$

---

nal. In fact, the recovery guarantees of MCA for inpainting is a very important theoretical problem that remains largely open.

## 4 EXPERIMENTAL EVALUATION

In this section, we evaluate the efficiency of the MCA framework for the recovery of missing information in high-resolution GIXD images, and subsequently, for the analysis of 1-D intensity patterns (linecuts), which are of interest for characterizing the structure of thin films. More specifically, our test set consists of two  $2048 \times 2048$  GIXD images,  $\mathbf{X}_1$  and  $\mathbf{X}_2$ , shown in Fig. 2, whose acquisition details were described in Section 2.

As already stated, the analysis of GIXD data is based on the reduction of the 2-D images into 1-D intensity patterns. From the several linecuts mentioned in Section 2, herein we focus on three distinct types of linecuts (Fig. 1(b)):

1. Intensity as a function of the scattering vector  $\mathbf{q}$  (I vs.  $q$ )
2. Intensity as a function of the out-of-plane component of the scattering vector (I vs.  $q_z$ )

### 3. Intensity as a function of the polar angle (I vs. $\chi$ )

It is noted that in the subsequent experimental evaluation we ignore the part at the bottom of the GIXD images that corresponds to the shadow of the film substrate, since the intensity recorded in this area is not diffracted from the film under study.

The performance of MCA<sup>2</sup> is evaluated in the case of corrupted GIXD images. In particular, the simulated missing information appears in the form of either randomly burned pixels, or whole burned rows distributed also at random across the whole image. For this purpose, two types of random binary masks are generated, namely,

- *Random mask* ( $\mathbf{M}_{RM}$ ): matrix whose elements are equal to 1, except for a subset of them being 0, whose positions are distributed uniformly at random.
- *Random lines mask* ( $\mathbf{M}_{RLM}$ ): matrix whose elements are equal to 1, except for a subset of rows consisting of all-zero elements, with the indices of these “burned” rows being selected uniformly at random among all the rows of the original image.

Under this assumption, an observed corrupted GIXD image  $\mathbf{X}$  can be expressed as follows, depending on the corruption pattern,

$$\mathbf{X}_{d, RM} = \mathbf{M}_{RM} \circ \mathbf{X}, \quad \mathbf{X}_{d, RLM} = \mathbf{M}_{RLM} \circ \mathbf{X}, \quad (8)$$

where  $\mathbf{X}_{d, RM}$  and  $\mathbf{X}_{d, RLM}$  are the recorded images with randomly missing pixels and whole rows, respectively. In the subsequent experiments, we generate random masks of the above two types with the number of missing pixels varying in {20%, 30%, 40%, 50%} as a percentage of the total number of pixels. In practice, the case of randomly distributed burned lines appears in case of 2-D detectors based on wire chambers [40], while the case of randomly burned pixels is met in more modern acquisition systems, such as CCD cameras.

Fig. 3 shows corrupted instances for both images,  $\mathbf{X}_1$  and  $\mathbf{X}_2$ , where the top row corresponds to a random mask with 50% of missing information, while the bottom row corresponds to a random lines mask, also with 50% of burned pixels. In our case, a successful inpainting of the observed GIXD images using MCA, is equivalent to recovering accurately the missing information such that the associated linecuts, estimated from the reconstructed images, to be close approximations of the linecuts corresponding to the original images. Although an amount of 40% or 50% of missing pixels is considered an extreme scenario in practice, however, we validate the efficiency of MCA even under such immoderate conditions.

<sup>2</sup>Matlab code available at: <http://www.greyc.ensicaen.fr/~jfadili/demos/WaveRestore/downloads/mcalab/Home.html>

### 4.1 Morphological components using UDWT and curvelets

As mentioned in Section 3.2, the morphological diversity relies on the sparsity of the distinct morphological components in specific bases. The appropriate choice of these bases, or in general the sparsifying transform domains, is highly determined by the specific structural content of a given image. For instance, the discrete cosine transform (DCT) is appropriate in describing spatially homogeneous texture, while the local ridgelet transform [38] is efficient in describing lines of fixed size. On the other hand, for images containing isotropic features and piecewise smooth regions, the UDWT has been shown to provide a very precise description [15], while anisotropic curvilinear structures have been shown to be represented optimally sparsely by the curvelet transform [16].

A visual inspection of the GIXD images used in this study, shown in Fig. 2, implies that none of the above transforms alone will be capable of extracting the geometric structures of these images. In contrast, we observe that the recorded images are characterized by both piecewise smooth regions and curvilinear structures. This necessitates the use of a combination of transforms to represent GIXD images for this type of thin films in a sparse way. This first observation induces the combination of the UDWT and the curvelet transform to be a suitable choice. Moreover, we emphasize that the success of the MCA technique is based on the degree of incoherence between the distinct sparsifying transforms [26] (the  $s$ -th morphological component  $\mathbf{X}_s$  is sparsely represented in  $\mathcal{T}_s$ , while its representation in the other transform domains is not sparse). The requirement of incoherence holds for the {UDWT, curvelet transform} pair, as it is proved in [44] (ref. Lemma 3.3 and discussion in Section 3.5). This serves as a second strong motivation for the selection of this pair of transforms.

Unlike the DWT, which decomposes a given image at multiple scales by downsampling the approximation and detail coefficients at each decomposition level, the UDWT does not incorporate the downsampling operations. Thus, the approximation and detail coefficients at each level have the same dimension as the original image. Besides, unlike the DWT, the UDWT is shift-invariant, while it is also more robust to ringing artifacts around singularities or edges.

On the other hand, the curvelet transform, to be used for the extraction of the anisotropic structures (curves, lines), is a special member of the family of multiscale geometric transforms. Conceptually, the curvelet transform is a multiscale pyramid with many directions (angles) and locations at each length scale, and needle-shaped elements (atoms) at fine scales. More specifically, curvelets, in addition to a variable width ( $w$ ), have also a variable length ( $l$ ) and so a variable anisotropy. The length and width at fine scales are related by a scaling law,  $w = l^2$  and thus the anisotropy increases with decreasing scale like a power law.



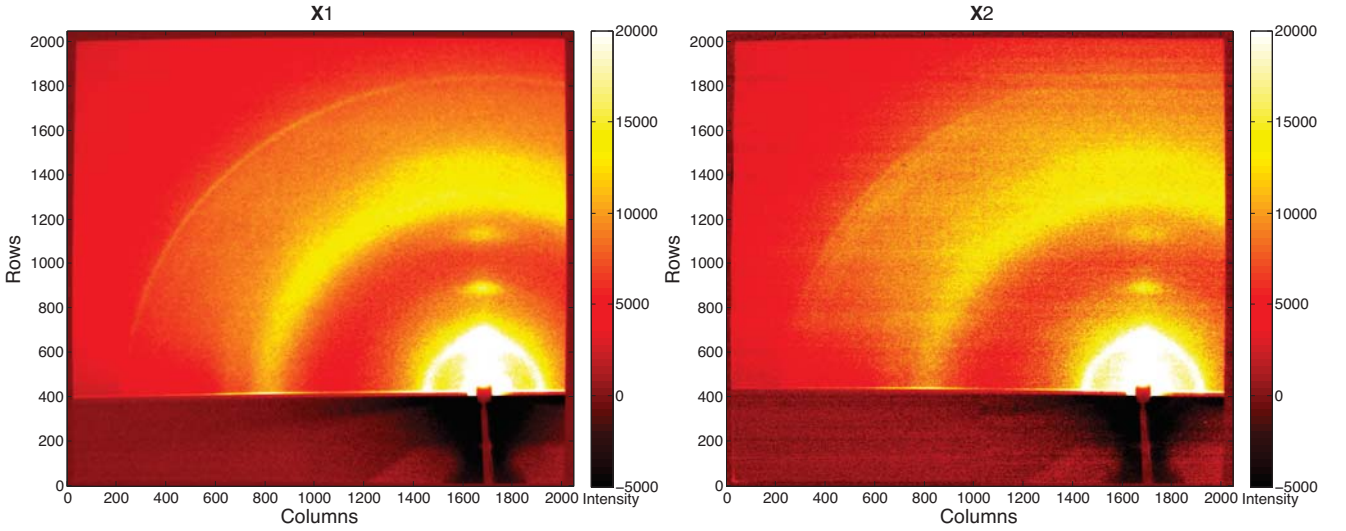


Figure 2

Original high-resolution  $2048 \times 2048$  GIXD images.

It has been shown [6, 16] that curvelets address efficiently problems where wavelets are far from ideal. For instance, they provide optimally sparse representations of objects which are characterized by smoothness except for discontinuities along a general curve with bounded curvature. Moreover, they model faithfully the geometry of wave propagation-like structures, since they may be viewed as coherent waveforms with enough frequency localization so that they behave like waves but at the same time, with enough spatial localization so that they simultaneously behave like particles. GIXD images of the kind used in this study (ref. Fig. 2) present both characteristics, that is, curvilinear structures, as well as a wave propagation-like behavior, thus motivating the use of the curvelet transform for the analysis of the corresponding morphological component.

In the following, we test the effectiveness of MCA in decomposing the given GIXD images into their morphological components. More specifically, the parameters required by Algorithm 1 are set as follows:  $I_{\max} = 150$ ,  $\lambda_{\text{stop}} = 10^{-6}$ , and  $\tau^{(0)} = \lambda_{\max}$ , where  $\lambda_{\max}$  is the minimum of maximal amplitude coefficients of the recorded image in each sparsifying transform domain. Regarding the UDWT, each image is analyzed in 3 scales using the ‘‘Symlet 6’’ (sym6) wavelet, which is near symmetric, orthogonal, and bi-orthogonal. Moreover, the spread and the oscillating nature of the associated scaling and wavelet functions, respectively, are appropriate enough to analyze images with relatively piecewise smooth content, such as the GIXD images under study. Regarding the curvelet transform, each GIXD image is decomposed in 7 scales, where the number of angles for each scale, from the coarsest to the finest one, is equal to 1, 16, 32, 32, 64, 64, 128, respectively. For the above experimental setup, and for the specific implementations we

used here, the redundancy factor (*i.e.*, the ratio of the number of transform coefficients over the number of pixels) for the UDWT is equal to 3, while for the curvelet transform the redundancy factor is equal to 2.3.

Fig. 4 shows the morphological decomposition of **X1** and **X2** in the two distinct components. A visual inspection of both images verifies that UDWT is indeed capable of representing the isotropic, piecewise smooth regions, whereas the curvelets approximate anisotropic features, such as the arcs and the edges, which appear in both images.

As mentioned above, apart from the mutual incoherence of the (possibly over-complete redundant) transforms, the second key ingredient required for MCA to work properly, is their ability to represent the information content of a given image in a precise and compact form. This is indeed the case with the UDWT and the curvelet transform when they are applied on the given GIXD images. More precisely, for the image **X1**, 50% of UDWT coefficients contains the 82% of the total energy (the sum of the absolute values of the transform coefficients), while 96% of the total energy is compressed in the 25% of the most significant (of highest magnitude) curvelet coefficients. A similar behavior is observed for the image **X2**, where 50% of the UDWT coefficients contains the 81% of the total energy, while 25% of the curvelet coefficients corresponds to 94% of their total energy.

## 4.2 Approximation accuracy of the original linecuts

The primary objective of this study is the MCA inpainting of corrupted GIXD images, with the goal of approximating as accurately as possible the original diffracted intensity, expressed in terms of the three linecuts introduced in Section 2.

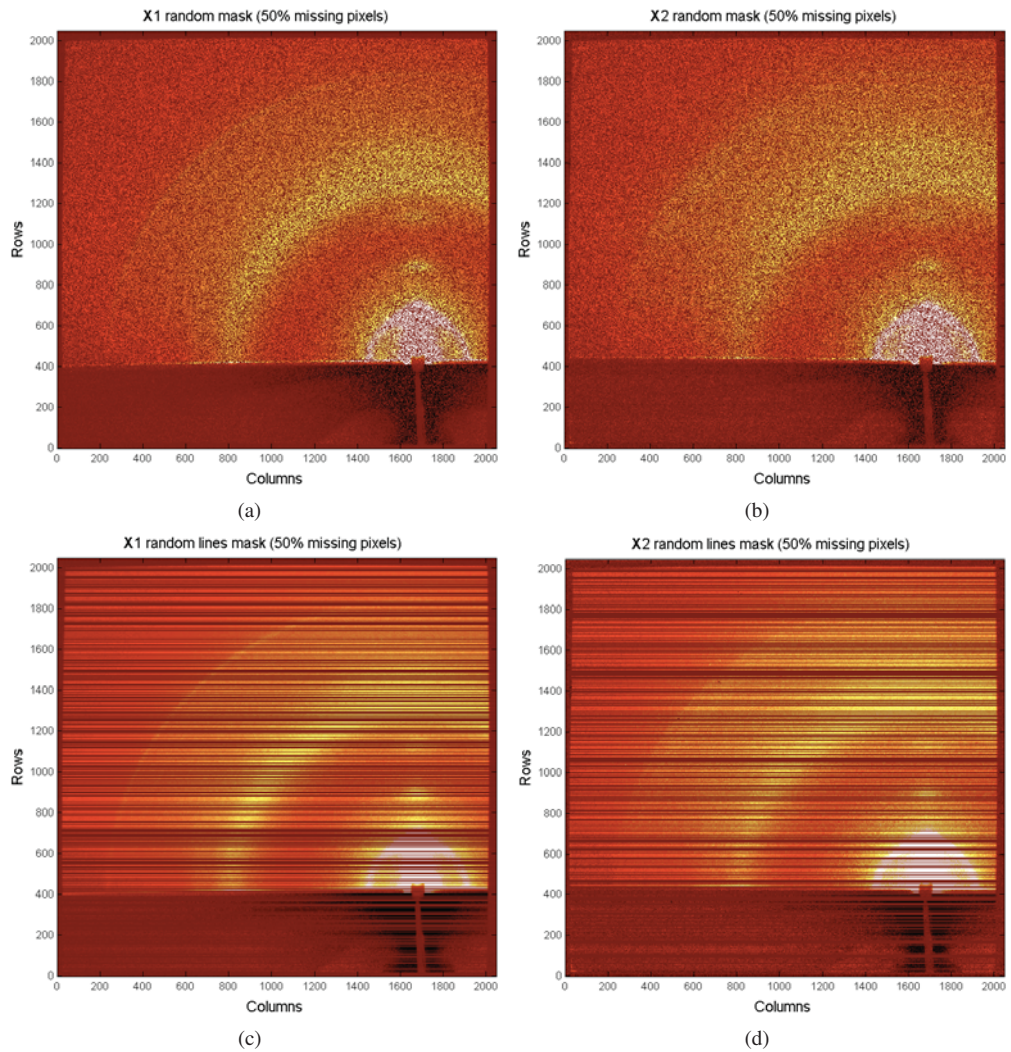


Figure 3

High-resolution  $2048 \times 2048$  GIXD images with 50% of missing pixels masked with: i) a random mask with uniformly distributed burned pixels [top row], ii) a random mask with uniformly distributed burned lines [bottom row].

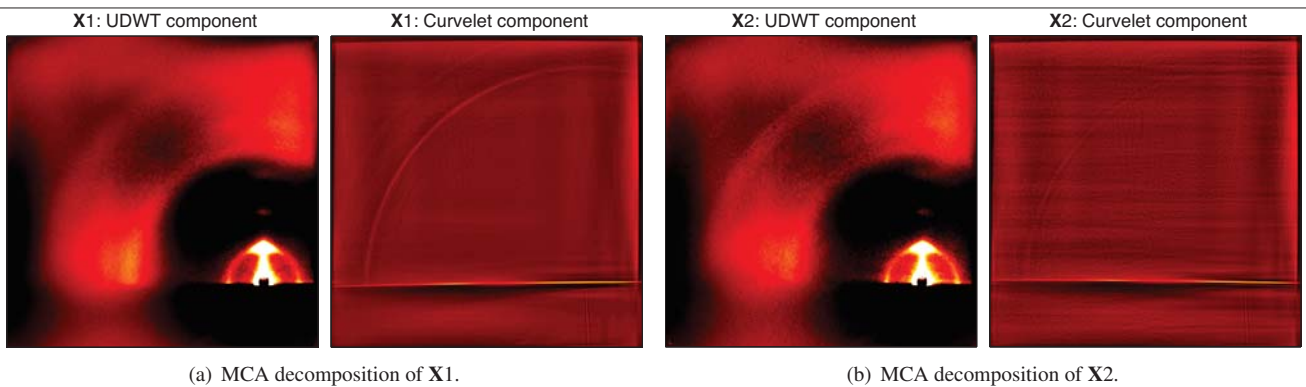


Figure 4

Analysis of X1 and X2 in two morphological components using the UDWT and the curvelet transform.

Although MCA is designed in a sparsity-based framework, solving a sparse optimization problem in suitable transform domains, however, for completeness of presentation its performance is compared with two classical inpainting techniques, namely, the method of *Fields of Experts* (FoE) [45] and a *partial differential equations*-based (PDE) approach [46]. In the former case, expressive image priors that capture the statistics of natural scenes are learned, extending traditional Markov Random Field (MRF) models by learning potential functions over extended pixel neighborhoods. In the later case, the lost information is restored guided by the anisotropic diffusion principle and the connectivity principle of human visual perception. Specifically, a fourth-order PDE model allows for the transportation of available information from the exterior towards the interior of the inpainting domain and the simultaneous diffusion of the information inside the inpainting region.

Starting with the efficiency of MCA in inpainting GIXD images corrupted by a random mask, Fig. 5 shows the inpainted  $\mathbf{X1}$  and  $\mathbf{X2}$  images corresponding to the two extreme cases of our experimental setup, that is, for 20% and 50% of randomly burned pixels. As it can be seen, the performance of MCA is excellent, even for 50% of missing information, which is a rather immoderate practical scenario. Comparing the inpainted images with their original versions, shown in Fig. 2, we observe that the smooth regions in  $\mathbf{X1}$  are preserved, while the noise-like appearance of  $\mathbf{X2}$  is suppressed slightly in its two inpainted counterparts. This is mainly due to an internal soft-thresholding step used by MCA, which tends to moderate the noise-like features.

Fig. 6 presents the corresponding linecuts (I vs.  $q$ , I vs.  $q_z$ , I vs.  $\chi$ ) for  $\mathbf{X1}$  and  $\mathbf{X2}$ , comparing the original curves with the curves obtained from the inpainted images. In case of  $\mathbf{X1}$ , the approximation of all three original linecuts is almost perfect, even for high percentages of corrupted pixels. The reconstruction performance for  $\mathbf{X2}$  is also very high, except for the extreme case at which half of the information is missing. In that case, small deviations from the original curves are apparent (for instance, in the rightmost part of the I vs.  $\chi$  linecut in Fig. 6(b)). The above observations initially derived by visual inspection of the curves in Fig. 6 are verified numerically by computing the Mean Squared Relative Approximation Error (MSRE) between the original and reconstructed linecuts, as follows,

$$\text{MSRE}(\hat{l}, l_{true}) = \text{mean} \left( \left( \frac{\hat{l} - l_{true}}{l_{true}} \right)^2 \right), \quad (9)$$

where  $l_{true}$  is the original and  $\hat{l}$  the corresponding reconstructed linecut.

Table 1 shows the MSRE (%) for the two GIXD images, along with the associated standard deviation of the approximation errors for MCA, FoE, and PDE methods. The approximation accuracy is extremely high, especially for the smoother image  $\mathbf{X1}$ , while, as we expected, it decreases

slightly as the number of missing pixels increases. Moreover, MCA clearly outperforms the other two inpainting methods in most of the cases (the minimum MSRE values are shown in bold type). In particular, the PDE approach for inpainting GIXD images of the type studied here results in the worst performance. This is not surprising, since the diffusion operation, which is inherent in every PDE-based method, tends to smooth out the interiors of the inpainted regions. This smoothing effect may detract the noise-like structures of our GIXD images, thus yielding less accurate linecuts. An additional advantage of MCA, when compared with FoE and PDE, is that, apart from restoring the lost information of a given image, it also gives the decomposition into its morphological components as a byproduct. As mentioned above, this decomposition can be further employed to extract structural information from GIXD data. For instance, a difference in the curvilinear structures corresponding to the two different annealing temperatures is apparent by a simple visual inspection of the curvelet components of  $\mathbf{X1}$  and  $\mathbf{X2}$  (ref. Fig. 4).

As a second experiment, we evaluate the efficiency of MCA when whole lines of burned pixels are distributed randomly in the recorded GIXD image. Fig. 7 shows the inpainted  $\mathbf{X1}$  and  $\mathbf{X2}$  images for 20% and 50% of burned pixels across randomly distributed lines. As in the previous scenario of uniformly random burned pixels, the performance of MCA is again very high, even for 50% of missing information. Comparing the inpainted images with their original counterparts in Fig. 2, we can see that the noise-like appearance of  $\mathbf{X2}$  is suppressed in the two inpainted images. The smoothing effect of MCA for the specific combination of sparsifying transforms we employ here (UDWT, curvelets) is more prevalent in the darker area at the bottom of the inpainted images (Fig. 7(b)). However, this is not a constraint, since, as mentioned at the beginning of this section, the dark area is ignored during the computation of the linecuts.

Fig. 8 shows the corresponding linecuts for  $\mathbf{X1}$  and  $\mathbf{X2}$ , comparing the original curves with the curves obtained from the inpainted images. As an overall conclusion, we can say that the approximation is highly accurate, especially for the first two linecuts (I vs.  $q$  and I vs.  $q_z$ ), while it diminishes slightly as the percentage of missing pixels increases. We observe that the I vs.  $\chi$  linecut presents the highest sensitivity with respect to the amount of missing information, which is more apparent near the peak and in the tails of the intensity curves for both images. In fact, the peak corresponds to the integrated intensity of the pixels around column 1680 between rows 630 – 670, while the tail corresponds to the integrated intensity around row 440 between columns 1440 – 1460 (Fig. 2). A close inspection of the randomly masked images shown in Figs. 3(c)-3(d) reveals that there is a large amount of missing pixels, effectively greater than 50%, concentrated in the corresponding areas. This inhibits the inpainting of the images in these specific areas,

resulting in an increased approximation error.

The above observations are consistent with the entries of Table 2, which compares the corresponding MSRE (%) values for the two GIXD images, along with the associated standard deviation of the approximation errors, for the MCA, FoE, and PDE inpainting methods. As in the case of randomly burned pixels, the approximation accuracy is extremely high, especially for the smoother image X1 and for the first two linecuts, while, as we expected, it decreases slightly as the number of missing pixels increases. Moreover, we can see that MCA achieves a more accurate reconstruction of the linecuts in most of the cases, when compared with FoE and PDE, while its performance is very close to FoE and PDE for the cases where they resulted in a lower MSRE. Finally, by comparing the corresponding entries in Table 1 and Table 2 we conclude that inpainting GIXD images in case of randomly missing lines is more demanding than inpainting when randomly distributed pixels are missing. Focusing on MCA, as we suggest in the next section, this reduction in reconstruction accuracy can be alleviated by incorporating additional sparsifying transforms, such as the ridgelet transform, which is more efficient in extracting meaningful information from the neighboring pixels across straight lines and edges. This is another important advantage of MCA, that is, the ability to improve the inpainting accuracy by incorporating more efficient sparsifying transformations, which are able to extract additional structural components. On the other hand, such an improvement is impossible with methods like FoE or PDE.

## 5 CONCLUSIONS AND FUTURE WORK

In this study, we exploited the efficiency of the recently introduced MCA algorithm for the decomposition of images in distinct morphological components, based on the achieved sparsity in appropriate over-complete redundant transform domains, with our objective being to solve the problem of recovering the missing information in corrupted GIXD images due to potential malfunction of the detectors.

The experimental evaluation using high-resolution GIXD images of thin polythiophene-based films, showed that the proposed approach is highly efficient in recovering the missing information in the form of either randomly burned pixels, or whole burned rows, even at the order of 50% of the total number of pixels. This led to the derivation of accurate intensity 1-D plots (linecuts) from the recovered inpainted images, that will later allow for correct data interpretation. This result can be of high impact in scattering and imaging techniques applied for materials characterization, since it indicates that the proposed MCA-based inpainting approach weakens the necessity for long acquisition times or repeating experiments, especially in synchrotron-based experiments, related to inferior detector performance during the acquisition process.

In the present work, we employed the UDWT and the curvelet transform as the most appropriate domains for the representation of the isotropic, as well as the anisotropic features, which appear in GIXD data. However, the experimental evaluation showed a smoothing effect when a GIXD image presents noise-like features, which may decrease the estimation accuracy of the corresponding linecuts. We expect that the inclusion of additional transforms to capture more features, such as the noiselet, the ridgelet or the recently introduced shearlet transform [47], which best suit noise-like features and edges, respectively, will improve the inpainting performance, and subsequently the more accurate extraction of the linecuts that are of interest to us. Moreover, in all these cases, the sparsifying transforms are fixed. However, recent works have shown that, instead of deploying a predefined set of transforms, the use of learned sparsifying dictionaries [7, 8], which adapt to the inherent image structures, often result in a superior reconstruction performance. Thus, the extension of the MCA inpainting framework in a joint dictionary learning and reconstruction framework would be also of great importance.

## ACKNOWLEDGMENT

This work is supported by CS-ORION Marie Curie Industry - Academia Partnerships and Pathways (IAPP) project funded by the European Commission in FP7 (PIAP-GA-2009-251605), and Région Aquitaine and Arkema.

We are also grateful to Dr. G. Portale for conducting the GIXD measurements in the Dutch-Belgian Beamline (DUBBLE) of the European Synchrotron Radiation Facility (ESRF), Grenoble, France. The Netherlands Organisation for Scientific Research (NWO) and ESRF are acknowledged for granting the beamtime.

## REFERENCES

- 1 Guinier A. (1994) *X-Ray Diffraction: In crystals, imperfect crystals, and amorphous bodies*, Dover Books on Physics, ISBN: 0-486-68011-8.
- 2 Warren B. E. (1991) *X-ray Diffraction*, Dover Publications Inc., ISBN: 0-486-66317-5.
- 3 Pietsch U., Holy V., and Baumbach T. (2004) *High resolution X-ray scattering, from thin films to lateral nanostructures*, Springer, ISBN: 978-0-387-40092-1.
- 4 Verploegen E., Mondal R., Bettinger C. J., Sok S., Toney M. F., and Bao Z. (2010) Effects of thermal annealing upon the morphology of polymer-fullerene blends, *Advanced Functional Materials* **20**, 3519–3529.
- 5 Mallat S. (2008) *A wavelet tour of signal processing (3rd edition: The sparse way)*, Academic Press.
- 6 Starck J.-L., Murtagh F., and Fadili J. (2010) *Sparse image and signal processing: Wavelets, curvelets, morphological diversity*, Cambridge University Press, Cambridge (GB), ISBN-10: 0521119138; 336-pp. monograph.

- 7 Murray J. and Kreutz-Delgado K. (2004) Sparse image coding using learned overcomplete dictionaries, *14th IEEE Workshop on Machine Learning for Signal Processing*, Sao Luis, Sept. 29–Oct. 1.
- 8 Elad M. and Aharon M. (2006) Image denoising via sparse and redundant representations over learned dictionaries, *IEEE Trans. on Image Processing* **15**, 12, 3736–3745.
- 9 Starck J.-L., Nguyen M. K., and Murtagh F. (2003) Wavelets and curvelets for image deconvolution: a combined approach, *Signal Processing* **83**, 2279–2283.
- 10 Starck J.-L., Donoho D., and Candès (2001) Very high quality image restoration by combining wavelets and curvelets, *Proc. SPIE, Wavelets: Application in Signal and Image Processing IX* **4478**, 9–19.
- 11 Haykin S. (2001) *Unsupervised adaptive filtering, Volume 1: Blind source separation*, John Wiley and Sons, New York.
- 12 Hyvarinen A., Karhunen J., and Oja E. (2001) *Independent component analysis*, John Wiley and Sons, New York.
- 13 Daubechies I. (1992) *Ten lectures on wavelets*, Philadelphia, PA: SIAM.
- 14 Kingsbury N. (1998) The dual-tree complex wavelet transform: A new technique for shift invariance and directional filters, *8th IEEE DSP Workshop*, Utah, Aug. 9–12.
- 15 Starck J.-L., Fadili J., and Murtagh F. (2007) The undecimated wavelet decomposition and its reconstruction, *IEEE Trans. on Image Processing* **16**, 2, 297–309.
- 16 Candès E. and Donoho D. (2002) Recovering edges in ill-posed inverse problems: Optimality of curvelet frames, *Ann. Statist.* **30**, 3, 784–842.
- 17 Do M. and Vetterli M. (2003) *Contourlets, beyond wavelets*, G. V. Welland, Ed. New York: Academic.
- 18 Mallat S. and LePennec E. (2005) Sparse geometric image representation with bandelets, *IEEE Trans. on Image Processing* **14**, 4, 423–438.
- 19 Vese L. and Osher S. (2003) Modeling textures with total variation minimization and oscillating patterns in image processing, *Journal Scientific Computing*, **19**, 1-3, 553–572.
- 20 Aujol J.-F., Aubert G., Blanc-Féraud L. and Chambolle A. (2005) Image decomposition into a bounded variation component and an oscillating component, *Journal of Mathematical Imaging and Vision*, **22**, 71–88.
- 21 Aujol J.-F., Gilboa G., Chan T. and Osher S. (2006) Structure-texture image decomposition - modeling, algorithms, and parameter selection, *Int. Journal of Computer Vision*, **67**, 1, 111–136.
- 22 Maurel P., Aujol J.-F. and Peyré G. (2011) Locally parallel texture modeling, *SIAM Journal Imaging Sciences*, **4**, 1, 413–447.
- 23 Briceño-Arias L., Combettes P., Pesquet J.-C. and Pustelnik N. (2011) Proximal algorithms for multicomponent image recovery problems, *Journal of Mathematical Imaging and Vision*, **41**, 1, 3–22.
- 24 Kreutz-Delgado K. and Rao B. (1999) Sparse basis selection, ICA, and majorization: towards a unified perspective, *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, Phoenix, AZ, USA, Mar. 15–19.
- 25 Zibulevsky M. and Pearlmutter B. (2001) Blind source separation by sparse decomposition in a signal dictionary, *Neural Computation* **13**, 4, 863–882.
- 26 Starck J.-L., Elad M., and Donoho D. (2004) Redundant multi-scale transforms and their application for morphological component analysis, *Advances in Imaging and Electron Physics* **132**, 287–348.
- 27 Starck J.-L., Elad M., and Donoho D. (2005) Image decomposition via the combination of sparse representations and a variational approach, *IEEE Trans. on Image Processing* **14**, 10, 1570–1582.
- 28 Dubois S., Péteri R. and Ménard M. (2010) Decomposition of dynamic textures using morphological component analysis: A new adaptive strategy, *Proc. Intl. Conf. on Pattern Recognition*, Istanbul, Turkey, Aug. 23–26.
- 29 Abrial P., Moudden Y., Starck J.-L., Bobin J., Fadili M. J., Afeyan B. and Nguyen M. (2007) Morphological component analysis and inpainting on the sphere: application in physics and astrophysics, *Journal of Fourier Analysis and Applications (JFAA)*, Special issue on “Analysis on the Sphere”, **13**, 6, 729–748.
- 30 Gao X., Wang Y., Li X. and Tao D. (2010) On combining morphological component analysis and concentric morphology model for mammographic mass detection, *IEEE Trans. on Information Technology in Biomedicine*, **14**, 2, 266–273.
- 31 Gaudes C. C., Van de Ville D., Petridou N., Lazeyras F. and Gowland P. (2011) Paradigm-free mapping with morphological component analysis: Getting most out of fMRI data, *Proc. of SPIE, Wavelets and Sparsity XIV*, **8138**, San Diego, CA, USA.
- 32 Elad M., Starck J.-L., Querre P., and Donoho D. (2005) Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA), *Applied and Computational Harmonic Analysis* **19**, 3, 340–358.
- 33 Natarajan B. (1995) Sparse approximate solutions to linear systems, *SIAM Journal on Computing* **24**, 227–234.
- 34 Gribonval R. and Nielsen M. (2003) Sparse representations in unions of bases, *IEEE Trans. on Information Theory* **49**, 12, 3320–3325.
- 35 Donoho D. (2004) For most large underdetermined systems of linear equations the minimal  $\ell_1$ -norm solution is also the sparsest solution, *Technical Report*, Stanford University (Available at: <http://www-stat.stanford.edu/~donoho/Reports/2004/1110EquivCorrected.pdf>).
- 36 Mallat S. and Zhang Z. (1993) Matching pursuit with time-frequency dictionaries, *IEEE Trans. on Signal Processing* **41**, 12, 3397–3415.
- 37 Chen S., Donoho D., and Saunders M. (1998) Atomic decomposition by basis pursuit, *SIAM Journal on Scientific Computing* **20**, 33–61.
- 38 Candès E. and Donoho D. (1999) Ridgelets: the key to high dimensional intermittency?, *Philosophical Transactions of the Royal Society of London A* **357**, 2495–2509.
- 39 Starck J.-L., Candès E. and Donoho D. (2002) The curvelet transform for image denoising, *IEEE Trans. on Image Processing*, **11**, 6, 131–141.
- 40 Gabriel A. and Dauvergne F. (1982) The localisation method used at EMBL, *Nuclear Instruments and Methods in Physics Research* **201**, 1, 223–224.
- 41 Bruce A., Sardy S. and Tseng P. (1998) Block coordinate relaxation methods for nonparametric signal denoising, *Proc. of the SPIE - The International Society for Optical Engineering* **3391**, 75–86.
- 42 Bobin J., Moudden Y., Fadili M.J. and Starck J.-L. (2009) Morphological diversity and sparsity for multichannel data restoration, *Journal of Mathematical Imaging and Vision*, **33**, 2, 149–



- 168.
- 43 Elad M. (2006) Why simple shrinkage is still relevant for redundant representations?, *IEEE Trans. on Information Theory*, **52**, 12, 5559–5569.
- 44 Donoho D. and Kutyniok G. (2010) Microlocal analysis of the geometric separation problem, *Tech. Report*, No. 2010-01, Dept. of Statistics, Stanford University.
- 45 Roth S. and Black M. J. (2005) Fields of Experts: a framework for learning image priors, *Proc. of IEEE Computer Vision and Pattern Recognition*, June 20–25, Providence, RI, USA.
- 46 Peiying C. and Yuandi W. (2009) A new fourth-order equation model for image inpainting, *Proc. 6th Int. Conf. on Fuzzy Systems and Knowledge Discovery*, Aug. 14–16, Shanghai, China.
- 47 Kutyniok G. and Lim W. (2011) Compactly supported shearlets are optimally sparse, *Journal of Approximation Theory*, **163**, 1564–1589.

*The date of receipt and acceptance  
will be inserted by the editor.*

Copyright © 2010 IFPEN Energies nouvelles

*Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than IFP Energies nouvelles must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee: Request permission from Documentation, IFP Energies nouvelles, fax. +33 1 47 52 70 78, or [revueogst@ifpen.fr](mailto:revueogst@ifpen.fr).*

TABLE 1

Mean Squared Relative approximation Error (MSRE) (%) between the original and reconstructed linecuts, using MCA, FoE, and PDE inpainting methods, for images X1, X2 with randomly distributed missing pixels (the standard deviation of the error is shown in parentheses).

|              |        | X1                                               |                                                  |                                                  |                                  |
|--------------|--------|--------------------------------------------------|--------------------------------------------------|--------------------------------------------------|----------------------------------|
| Linecut      | Method | Missing pixels (%)                               |                                                  |                                                  |                                  |
|              |        | 20%                                              | 30%                                              | 40%                                              | 50%                              |
| I vs. $q$    | MCA    | $3.35 \cdot 10^{-5}$<br>( $7.46 \cdot 10^{-5}$ ) | $5.24 \cdot 10^{-5}$<br>( $1.17 \cdot 10^{-4}$ ) | $1.25 \cdot 10^{-4}$<br>( $2.75 \cdot 10^{-4}$ ) | $5.45 \cdot 10^{-4}$<br>(0.0012) |
|              | FoE    | $7.89 \cdot 10^{-4}$<br>( $9.27 \cdot 10^{-4}$ ) | $4.12 \cdot 10^{-4}$<br>(0.0031)                 | 0.0011<br>(0.0020)                               | 0.0012<br>(0.0034)               |
|              | PDE    | $3.38 \cdot 10^{-4}$<br>( $7.04 \cdot 10^{-4}$ ) | $3.55 \cdot 10^{-4}$<br>( $7.47 \cdot 10^{-4}$ ) | $5.14 \cdot 10^{-4}$<br>(0.0011)                 | 0.0015<br>(0.0031)               |
| I vs. $q_z$  | MCA    | <b>0.0020</b><br>(0.0045)                        | <b>0.0031</b><br>(0.0074)                        | <b>0.0080</b><br>(0.0186)                        | <b>0.0375</b><br>(0.0689)        |
|              | FoE    | 0.0281<br>(0.0511)                               | 0.0342<br>(0.0610)                               | 0.0412<br>(0.0752)                               | 0.0435<br>(0.0771)               |
|              | PDE    | 0.0268<br>(0.0471)                               | 0.0277<br>(0.0501)                               | 0.0339<br>(0.0612)                               | 0.0890<br>(0.1610)               |
| I vs. $\chi$ | MCA    | $1.32 \cdot 10^{-4}$<br>( $1.57 \cdot 10^{-4}$ ) | $2.50 \cdot 10^{-4}$<br>( $3.83 \cdot 10^{-4}$ ) | $6.03 \cdot 10^{-4}$<br>( $7.62 \cdot 10^{-4}$ ) | <b>0.0589</b><br>(0.7151)        |
|              | FoE    | 0.0490<br>(0.7100)                               | 0.0520<br>(0.7290)                               | 0.0591<br>(0.7311)                               | 0.0712<br>(0.7500)               |
|              | PDE    | 0.0587<br>(0.7300)                               | 0.0610<br>(0.7320)                               | 0.0640<br>(0.7501)                               | 0.1671<br>(1.5100)               |
|              |        | X2                                               |                                                  |                                                  |                                  |
| Linecut      | Method | Missing pixels (%)                               |                                                  |                                                  |                                  |
|              |        | 20%                                              | 30%                                              | 40%                                              | 50%                              |
| I vs. $q$    | MCA    | <b>0.0019</b><br>(0.0049)                        | 0.0061<br>(0.0083)                               | <b>0.0028</b><br>(0.0084)                        | 0.0047<br>(0.013)                |
|              | FoE    | 0.0024<br>(0.0065)                               | <b>0.0026</b><br>(0.0071)                        | 0.0031<br>(0.0058)                               | <b>0.0039</b><br>(0.0056)        |
|              | PDE    | 0.0020<br>(0.0050)                               | 0.0039<br>(0.0052)                               | 0.0029<br>(0.0077)                               | 0.0065<br>(0.0202)               |
| I vs. $q_z$  | MCA    | <b>0.0880</b><br>(0.1622)                        | 0.0951<br>(0.1683)                               | 0.1183<br>(0.2247)                               | <b>0.2035</b><br>(0.36)          |
|              | FoE    | 0.0943<br>(0.1722)                               | 0.0962<br>(0.1734)                               | <b>0.1120</b><br>(0.1670)                        | 0.2398<br>(0.1642)               |
|              | PDE    | 0.0885<br>(0.1622)                               | <b>0.0926</b><br>(0.1752)                        | 0.1123<br>(0.2051)                               | 0.2875<br>(0.8968)               |
| I vs. $\chi$ | MCA    | <b>0.0569</b><br>(0.4892)                        | <b>0.0584</b><br>(0.4686)                        | <b>0.0572</b><br>(0.4725)                        | <b>0.0811</b><br>(0.5854)        |
|              | FoE    | 0.0572<br>(0.4892)                               | 0.0593<br>(0.4922)                               | 0.0669<br>(0.4902)                               | 0.0905<br>(0.4706)               |
|              | PDE    | 0.0601<br>(0.4892)                               | 0.0625<br>(0.4967)                               | 0.0634<br>(0.5224)                               | 0.1329<br>(1.0618)               |

TABLE 2

Mean Squared Relative approximation Error (MSRE) (%) between the original and reconstructed linecuts, using MCA, FoE, and PDE inpainting methods, for images X1, X2 with missing pixels across randomly distributed lines (the standard deviation of the error is shown in parentheses).

|              |        | X1                                                 |                                                    |                                                    |                           |
|--------------|--------|----------------------------------------------------|----------------------------------------------------|----------------------------------------------------|---------------------------|
| Linecut      | Method | Missing pixels (%)                                 |                                                    |                                                    |                           |
|              |        | 20%                                                | 30%                                                | 40%                                                | 50%                       |
| I vs. $q$    | MCA    | $5.36 \cdot 10^{-4}$<br>(0.0011)                   | $7.41 \cdot 10^{-4}$<br>(0.0015)                   | $8.32 \cdot 10^{-4}$<br>(0.0019)                   | <b>0.0140</b><br>(0.0500) |
|              | FoE    | <b><math>4.47 \cdot 10^{-4}</math></b><br>(0.0024) | $7.40 \cdot 10^{-4}$<br>(0.0018)                   | <b><math>7.79 \cdot 10^{-4}</math></b><br>(0.0015) | 0.0450<br>(0.0400)        |
|              | PDE    | $5.39 \cdot 10^{-4}$<br>(0.0010)                   | <b><math>6.60 \cdot 10^{-4}</math></b><br>(0.0012) | 0.0011<br>(0.0023)                                 | 0.0153<br>(0.0271)        |
| I vs. $q_z$  | MCA    | <b>0.0315</b><br>(0.0651)                          | 0.0467<br>(0.0950)                                 | <b>0.0623</b><br>(0.1412)                          | <b>0.1918</b><br>(0.6474) |
|              | FoE    | 0.0395<br>(0.0700)                                 | <b>0.0420</b><br>(0.0610)                          | 0.0662<br>(0.0753)                                 | 0.3001<br>(0.7211)        |
|              | PDE    | 0.0321<br>(0.0573)                                 | 0.0492<br>(0.1189)                                 | 0.0818<br>(0.2072)                                 | 0.2351<br>(0.6798)        |
| I vs. $\chi$ | MCA    | <b>0.0381</b><br>(0.3704)                          | 0.0604<br>(0.7219)                                 | <b>0.0619</b><br>(0.7308)                          | 1.6500<br>(1.3207)        |
|              | FoE    | 0.0510<br>(0.7220)                                 | <b>0.0590</b><br>(0.7310)                          | 0.0632<br>(0.7401)                                 | <b>1.2900</b><br>(0.9812) |
|              | PDE    | 0.0662<br>(0.5614)                                 | 0.0712<br>(0.7318)                                 | 0.0740<br>(0.7301)                                 | 2.4521<br>(2.9019)        |
|              |        | X2                                                 |                                                    |                                                    |                           |
| Linecut      | Method | Missing pixels (%)                                 |                                                    |                                                    |                           |
|              |        | 20%                                                | 30%                                                | 40%                                                | 50%                       |
| I vs. $q$    | MCA    | 0.0025<br>(0.0065)                                 | <b>0.0032</b><br>(0.0078)                          | 0.0040<br>(0.011)                                  | <b>0.0185</b><br>(0.0977) |
|              | FoE    | <b>0.0023</b><br>(0.0064)                          | 0.0033<br>(0.0058)                                 | <b>0.0039</b><br>(0.0056)                          | 0.0219<br>(0.0067)        |
|              | PDE    | 0.0027<br>(0.0075)                                 | 0.0033<br>(0.0073)                                 | 0.0056<br>(0.0136)                                 | 0.0372<br>(0.1010)        |
| I vs. $q_z$  | MCA    | <b>0.0987</b><br>(0.1924)                          | 0.1315<br>(0.2852)                                 | 0.1525<br>(0.3030)                                 | <b>0.3518</b><br>(1.48)   |
|              | FoE    | 0.0994<br>(0.1744)                                 | <b>0.1199</b><br>(0.1676)                          | <b>0.1483</b><br>(0.1658)                          | 0.3702<br>(0.1643)        |
|              | PDE    | 0.1027<br>(0.2577)                                 | 0.1357<br>(0.7375)                                 | 0.1977<br>(0.4981)                                 | 0.3826<br>(1.2608)        |
| I vs. $\chi$ | MCA    | <b>0.0616</b><br>(0.88)                            | 0.2303<br>(0.5817)                                 | <b>0.6640</b><br>(1.5218)                          | <b>0.9388</b><br>(1.511)  |
|              | FoE    | 0.0688<br>(0.4897)                                 | 0.2287<br>(0.4894)                                 | 0.7431<br>(0.5102)                                 | 1.1042<br>(0.4912)        |
|              | PDE    | 0.0678<br>(0.4891)                                 | <b>0.2073</b><br>(0.7357)                          | 0.7588<br>(1.3044)                                 | 1.2207<br>(1.2608)        |



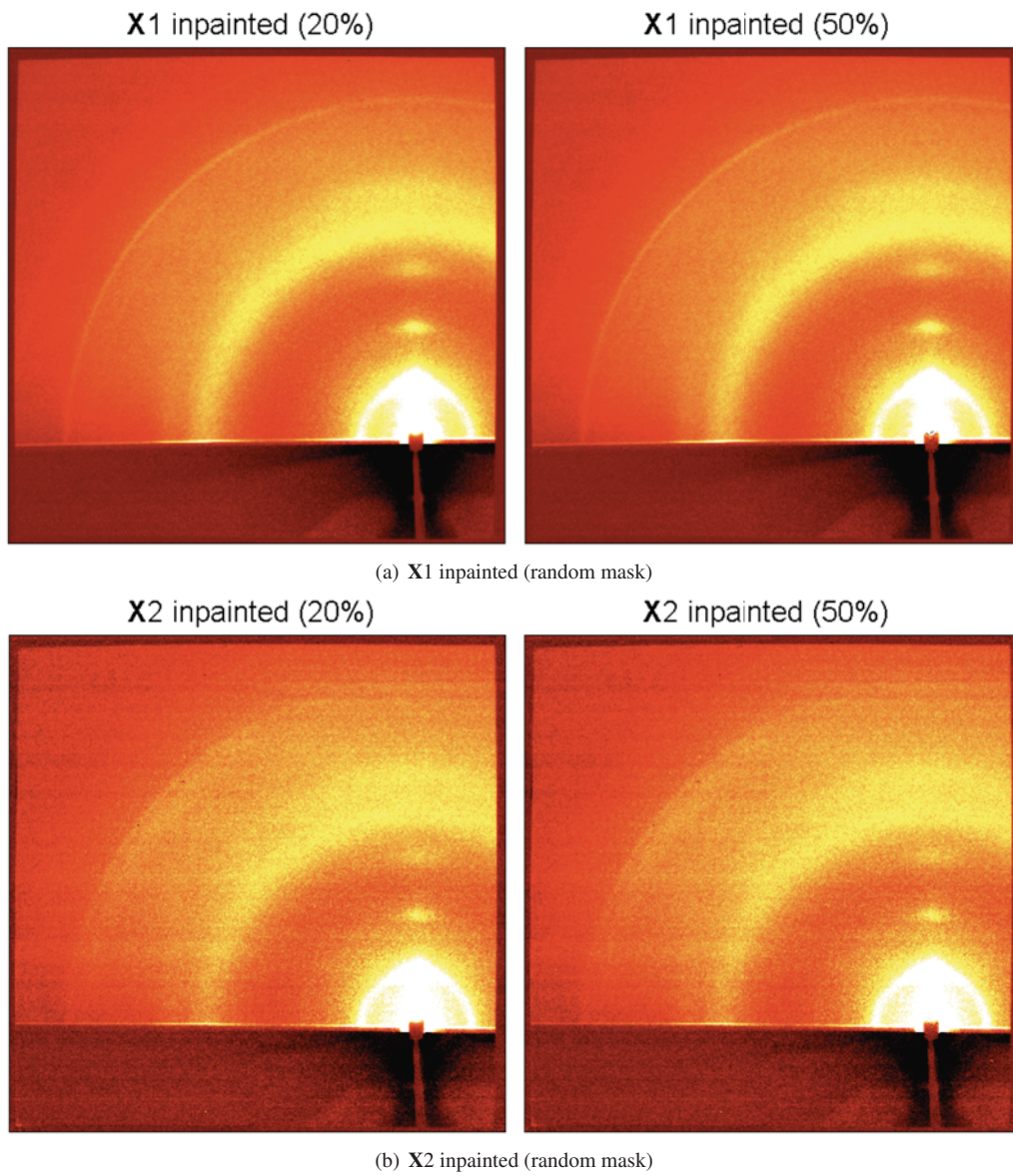


Figure 5  
Inpainted images X1, X2 using MCA for randomly distributed missing pixels (20%, 50%).

---

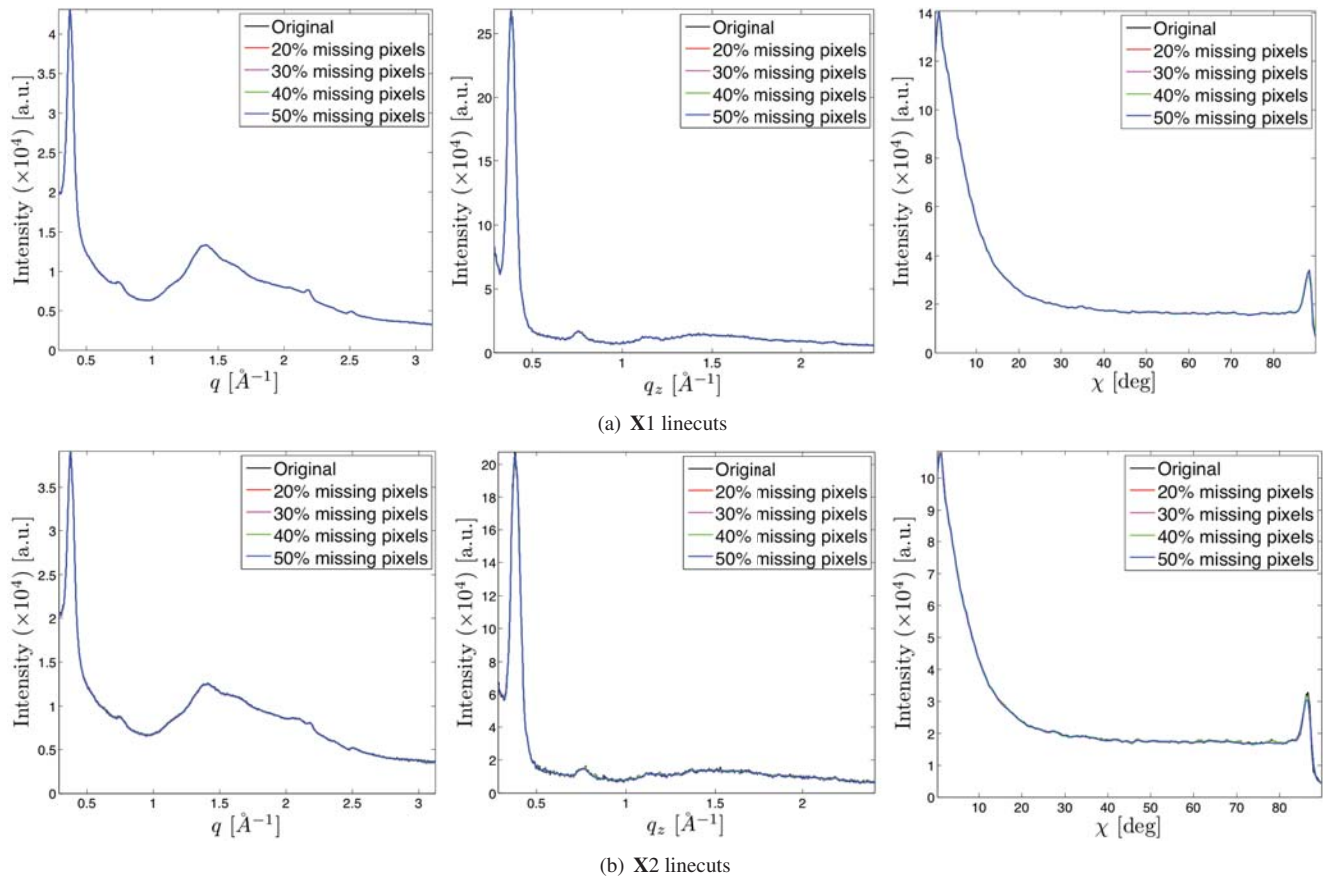


Figure 6

Linecuts of the original and inpainted (using MCA) GIXD images X1, X2 for randomly distributed missing pixels.

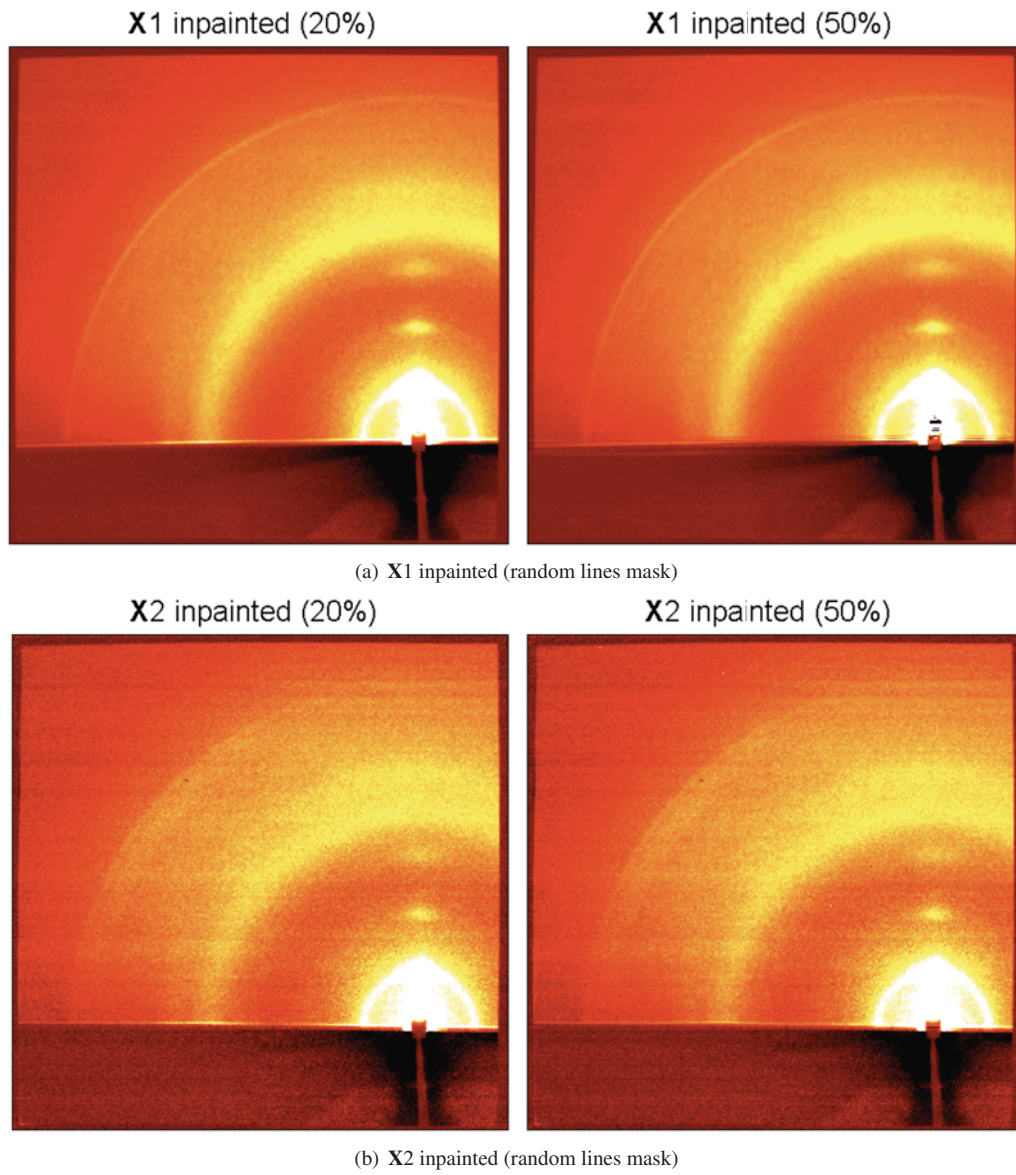


Figure 7

Inpainted images X1, X2 using MCA for randomly distributed lines of missing pixels (20%, 50%).

---

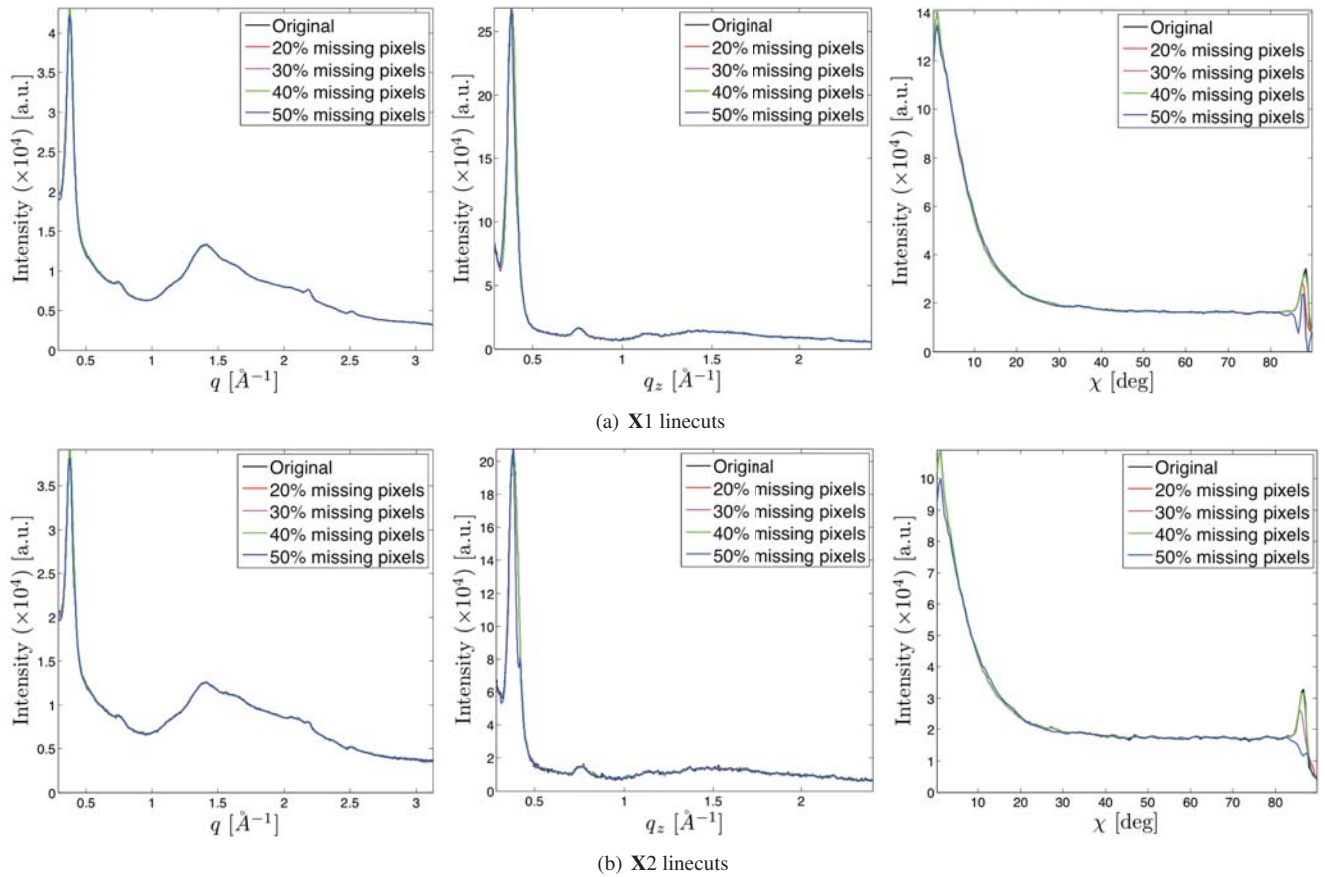


Figure 8

Linecuts of the original and inpainted (using MCA) GIXD images X1, X2 for randomly distributed lines of missing pixels.