

Restoration of diffraction patterns through different imaging planes

EDUARDO X MIQUELES,^{a*} JEAN M POLLI^a AND HARRY WESTFAHL JR^a

^a*Brazilian Synchrotron Light Laboratory, CNPEM, Campinas, São Paulo, Brazil.*

E-mail: eduardo.miqueles@lnls.br

Abstract

In this manuscript, the mathematical aspects for the image restoration of a direct conversion x-rays detector are described. The detector is determined by a collection of small planar area detectors, attached to a single computing node through a very specific physical geometry. By image restoration we define the image formed on a virtual plane that exists in front of the physical device at which a physical experiment is under analysis. We demonstrate feasibility of this imaging technique through simulated and real examples, arguing that the computation of standard physical quantities can be easily retrieved either on the virtual plane or in the detection set of planes. A set of routines were implemented using a combination of Python and CUDA programming language, so that the restoration is done rapidly for a block of images lying at the RAM of the detector. The implemented software is under usage at the 4th generation Brazilian synchrotron, Sirius, where a multicore system can distribute the restoration process so that applied imaging techniques as Ptychography or XPCS can be efficiently used.

1. Introduction

The imaging process using conventional planar detectors to measure diffraction scattering in a synchrotron facility is an extensively discussed subject in the literature, e.g. with standard techniques as SAXS or modern ones as coherent diffraction imaging (CDI) (Meneau *et al.*, 2021). Typically, it is assumed that the detectors are developed in such a way that the reciprocal space is directly measured on a flat imaging plane so that each pixel is directly related to a physical wave vector through well known relations involving the experimental distance of the scatter point, numerical aperture, the incident energy among other experimental factors. We usually refer to *a digital image* as the matrix of recorded intensities on each sensor lying on this flat imaging plane.

Replacing the hypothesis of a flat detector by one where we record the x-ray intensities on a sequence of distributed small flat imaging planes (here denoted as a *stripe*), the definition of a *digital image* now becomes non trivial. It also becomes not immediate the physical relation of what we use to call *pixel* and the frequencies in the reciprocal space. A similar effort was reported in (Le Bourlot *et al.*, 2012), where data is distributed in 6 flat panels (here denoted as a *stripe*) using a specific geometric configuration. We can have two situations for a set of distributed stripes, as indicated in Figures 1.(a) and (b). In the example (a) we illustrate a device determined by two horizontal stripes with an overlapping area (with their presentation either in the front view or in the side view). There is a physical distance between each stripe, which later will become a parameter among several others. In the second example (b), the same device proposed in (a) is clockwise mounted in order to define a bigger detection area, but keeping the same reasoning of the small device. The devices presented in Figure 1.(c) and (d) use the same reasoning, but with planar stripes with a gap separating them. Throughout this manuscript we use the notation of Definition 1.

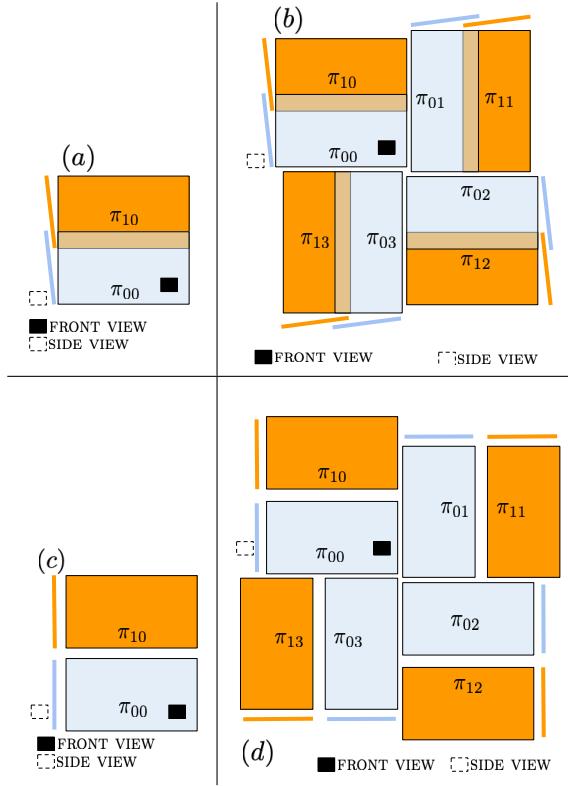


Fig. 1. Detection device in two different configurations (a) Composition of two stripes with an overlapping area (b) Composition of 4 modules in the same geometry and with 8 stripes. The number of stripes can be extended and the illustration presents only two stripes per module, for practical purposes.

Definition 1 We denote the area detector presented in Fig.1.(a) and (b) as $\Pi_{n,1}$ and $\Pi_{n,4}$ respectively. The area detector presented in Fig.1.(c) and (d) are denoted as $\Pi_{p,1}$ and $\Pi_{p,4}$, respectively. We use letters p and n to differentiate planar and non-planar detectors.

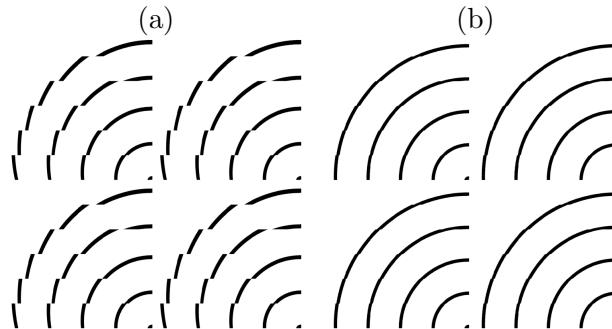


Fig. 2. Simulated example of raw data for $\Pi_{n,4}$ using two different distances from sample to detector (a) 40mm (b) 300mm.

The detection devices $\{\Pi_{n,1}, \Pi_{n,4}, \Pi_{p,1}, \Pi_{p,4}\}$ can be theoretically extended for an arbitrary number of stripes and in this manuscript we present the mathematical formulation for 6 stripes, i.e., stripes $\{\pi_{00}, \dots, \pi_{05}\}$. In practice, the extended device of Figure 1.(b) will have 24 stripes π_{ij} , $i = 0, 1, 2, 3$ indicating the module (clockwise direction) and j indicating the stripe. It is important to note that a given stripe is a portion of a mathematical plane, physically determined by a semiconductor sensor mounted on top of a set of ASIC chips and composed by a given amount of pixels. The discussion about the chips and their composition for a given stripe are beyond the scope of this work. We refer to (PITEC, 2022; Ponchut *et al.*, 2021) for a more robust discussion about sensors and the complete imaging setup. An electronic feature for those devices, imposing a mathematical difficulty is given by the fact that each stripe provide us with a digital image stored in row-major order. This situation is illustrated in Figure 2 for a simulated example of a sample producing ring patterns on the area detector.

In this manuscript, we demonstrate the feasibility of measuring diffraction patterns using the proposed detection geometry indicating all possible variables affecting the determination of a virtual image through the mathematical remapping process. For different types of image, either simulated or real ones, a discussion about the alignment

process, pixel size, point of normal incidence are also discussed, where we stress the main difficulties for an imaging process through this very specific area detector device. For completeness, we pose the problem that we aim to solve in this work in the following manner:

Problem 1 *Given a image set \mathbf{I} (as the one presented in Fig.2), find a remapping operator \mathbf{R} to restore the original image to a readable one, say $\mathbf{R}[\mathbf{I}]$ so that further algorithms \mathbb{P} can provide reliable results using $\mathbb{P}\mathbf{R}[\mathbf{I}]$.*

Coherent-diffraction imaging algorithms are good examples for the algorithmic family defining the operator \mathbb{P} , where one can use iterative techniques to find the missing phase information through the diffraction pattern set \mathbf{I} . These algorithms are beyond the scope of this work, and we refer to (Chapman *et al.*, 2006; Pfeiffer, 2018; Bauschke *et al.*, 2002) for a more focused discussion on experimental and reconstruction techniques.

2. Single composition

Starting with the detection device of Figure 1.(a), we illustrate once again its geometry with only four stripes, as depicted in Figure 3.(a), where we have introduced a local Cartesian coordinate system at the scatter point. A given ray emanating from this scatter point will not reach a given set of pixels lying behind on the overlapping area, but this assertion depends mainly on two factors, the distance between the so-called *imaging plane* (as indicated in Figure 3.(a)) and the scatter point and on point of normal incidence on this very same plane. The distance L_0 is the distance of the virtual plane, where we expect to construct a *virtual image*. This should be the same distance where a conventional *planar* image should be formed using conventional flat detectors.

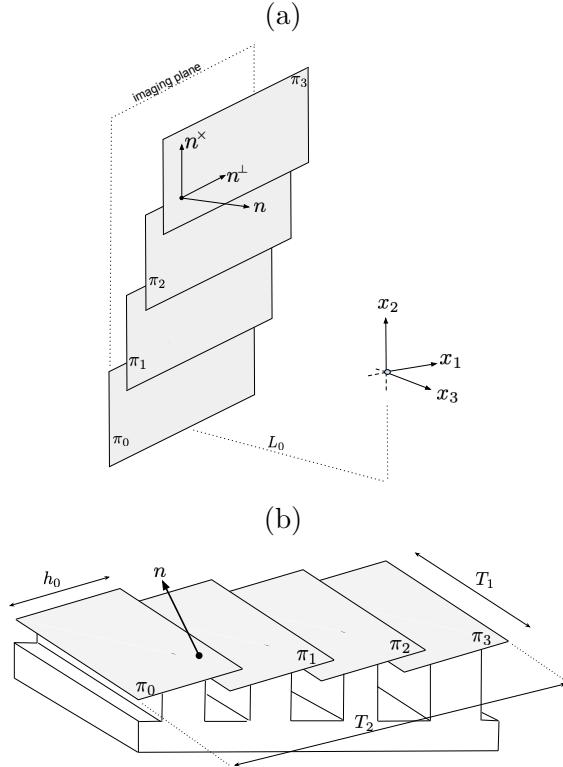


Fig. 3. Non-planar geometry illustrated only with 4 stripes π_j for convenience.

A full mathematical description - for the remapping operation from detected pixels lying on each stripe to a virtual imaging plane - is presented at the Appendix A. The description presented there (using basic linear algebra concepts) indicates the existence of a nonlinear (and invertible) operator \mathbf{R} such that

$$\mathbf{q}^\perp = \mathbf{R}(L_0, \mathbf{n}, \mathbf{v}) \quad (1)$$

for all points $\mathbf{q}^\perp = (q_x, q_y)$ on the virtual imaging plane. This relation primarily depends on the distance L_0 , the sequence of normal vectors \mathbf{n} for each stripe and the beam direction \mathbf{v} reaching the sample at the origin point.

3. Multiple compositions

In order to compose multiple modules (we define a *module* as a family of stripes) to obtain a bigger area detector, we have to consider a fixed point on the virtual imaging plane so that the grid of remapped points can have a formal reference point.

Non-planar

Considering the non-planar detector $\Pi_{n,4}$ we take the three-dimensional point $\mathbf{s} = (s_1, s_2, -L)$ as the center of the virtual image formed in front of the physical device (aligned with beamline incident direct beam). Taking Equation (A.21) and replacing \mathbf{z} by \mathbf{s} , we look for the pair $(t_{\text{ref}}, m_{\text{ref}})$ such that

$$\mathbf{s} - \mathbf{p} = t_{\text{ref}} \mathbf{n}^\times + m_{\text{ref}} \mathbf{n}^\perp \quad (2)$$

with $\mathbf{p} = \mathbf{y} - L\mathbf{n}$. Here, $\{\mathbf{n}, \mathbf{n}^\times, \mathbf{n}^\perp\}$ depends on the orientation of the stripe π_{ij} , but we avoid redundant notation to make the description easier. From the above equation, we can easily retrieve t_{ref} and m_{ref} from

$$\begin{aligned} t_{\text{ref}} &= n_2^\perp(s_1 - p_1) - n_1^\perp(s_2 - p_2), \\ m_{\text{ref}} &= -n_2^\times(s_1 - p_1) + n_1^\times(s_2 - p_2) \end{aligned} \quad (3)$$

with y_1, y_2 given by

$$\begin{aligned} y_1 &= (1 - n_1^2)z_1 - n_1 n_2 z_2 - n_1 n_3 z_3 \\ y_2 &= -n_2 n_1 z_1 + (1 - n_2^2)z_2 - n_2 n_3 z_3 \end{aligned} \quad (4)$$

Having $(t_{\text{ref}}, m_{\text{ref}})$ we use Equation (A.24) with (t, m) replaced by

$$t = t_{ij} + t_{\text{ref}} + y_{ij}, \quad m = m_{ij} - \bar{m} + m_{\text{ref}} + x_{ij} \quad (5)$$

to obtain $\mathbf{q}^\perp = (q_x, q_y)$ lying at the virtual imaging plane. Here, \bar{m} is the last point of the m -mesh, so that each module is shifted in a clockwise orientation from the image center. Also, t_{ij} and m_{ij} are the standard meshes for a given stripe (see equation (C.33)). From equation (A.5), variables $\{x_{ij}, y_{ij}\}$ indicate possible horizontal/vertical displacements for each stripe π_{ij} , which may occur due to factory misalignment.

Planar

The composition for the planar detector $\Pi_{p,4}$ can be obtained as a consequence of setting the appropriate variables on the detector $\Pi_{n,4}$. For instance, taking the angle of all stripes to be zero, the longitudinal distance δ equal to zero (see Figure 9) and moving properly the distances x_{ij}, y_{ij} we can force $\Pi_{p,4}$ to be equal to $\Pi_{n,4}$. The problem with this approach is that we still project the measured data onto a virtual plane in order to form an image. Instead of projecting, it is easier to adapt a different geometry coordinate system, based on pixels translations only, so that a bigger image can group the complete measured information. The logic implementation for this multiple composition, based only on vertical and horizontal translations of each measured stripe information, is intuitive and simple; reason for which we do not present it here.

4. Offset alignment strategy

After applying all the strategies described in Sections 3 and 2, we obtain images that can still present non-aligned features. This is due to the fact that deviations in all the described variables from the detector structure can impact the image composition. This is illustrated in Figure 4, where a feature determined by the union of all curves $\{\Omega_0, \dots, \Omega_7\}$ present a strong discontinuity set through the boundary of each stripe. There are many numerical strategies in the literature providing tools to identify the geometric behaviour of each stripe, such as Conex (Gommès & Goderis, 2010), Fit2d(Hammersley *et al.*, 1997), Dioptas (Prescher & Prakapenka, 2015) and Pyfai(Ashiotis *et al.*, 2015). Those tools assume that a given pattern can be measured on each stripe, such as the diffraction pattern obtained with Lysosyme crystals, providing us information about the stripe angles. Nonetheless, those tools do not provide

information about the complete imaging system, specially regarding the offset of each stripe.

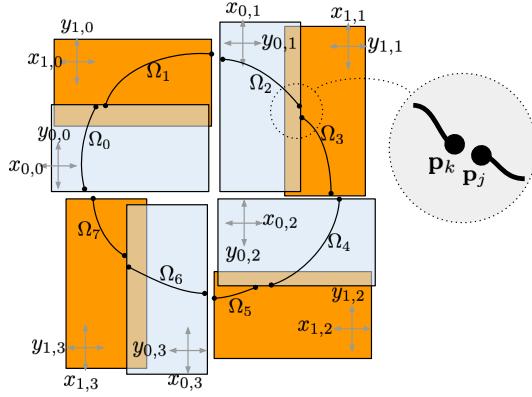


Fig. 4. Generic device determined by four misaligned modules providing a discontinuous curve $\Omega_0 \cup \dots \cup \Omega_7$.

The variables to be optimized are described below, for all devices discussed in this manuscript. We indicate by $\text{dim}(a)$ the size of the unknown array a :

1. Detector $\Pi_{n,4}$ (4 modules, each having 6 stripes): *total of 151 variables*

| <i>variable</i> | <i>description</i> | <i>dim</i> |
|-----------------|--|------------|
| a | angle deviations for each module | 4 |
| r_x | stripe rotations w.r.t to x -axis | 24 |
| r_y | stripe rotations w.r.t to y -axis | 24 |
| r_z | stripe rotations w.r.t to z -axis | 24 |
| ΔL | deviations on the distance value L_j | 24 |
| x | stripe offsets w.r.t to x axis | 24 |
| y | stripe offsets w.r.t to y axis | 24 |
| a_v | angles for the normal vector (Fig.10) | 3 |

2. Detector $\Pi_{n,1}$ (1 module having 6 stripes): *total of 39 variables*

| <i>variable</i> | <i>description</i> | <i>dim</i> |
|-----------------|--|------------|
| r_x | stripe rotations w.r.t to x -axis | 6 |
| r_y | stripe rotations w.r.t to y -axis | 6 |
| r_z | stripe rotations w.r.t to z -axis | 6 |
| ΔL | deviations on the distance value L_j | 6 |
| x | stripe offsets w.r.t to x axis | 6 |
| y | stripe offsets w.r.t to y axis | 6 |
| a_v | angles for the normal vector (Fig.10) | 3 |

3. Detector $\Pi_{p,4}$ (4 modules, each having 6 stripes): *total of 68 variables*

| <i>variable</i> | <i>description</i> | <i>dim</i> |
|-----------------|----------------------------------|------------|
| a | angle deviations for each module | 4 |
| x | stripe offsets w.r.t to x axis | 24 |
| y | stripe offsets w.r.t to y axis | 24 |
| g | gap distance between stripes | 20 |

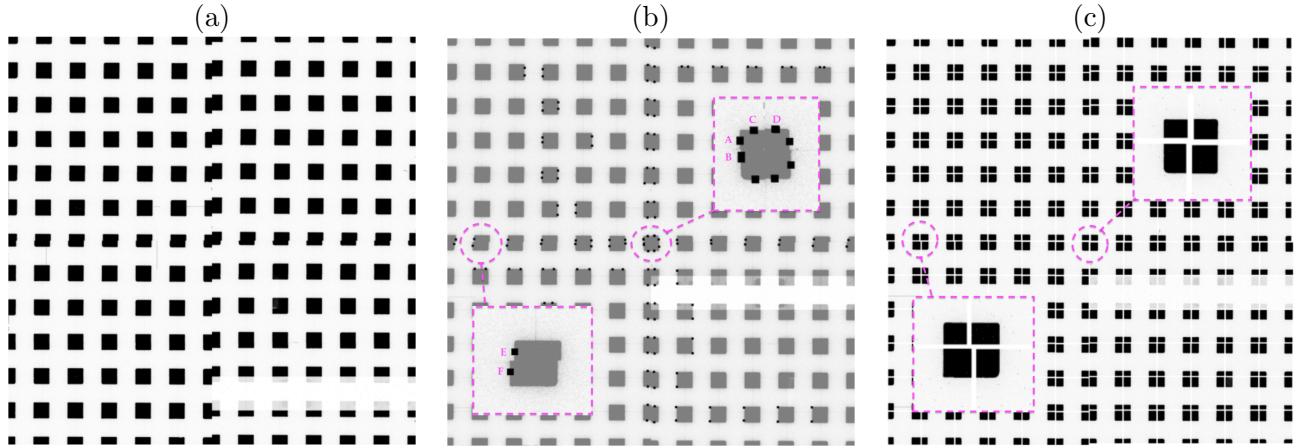
4. Detector $\Pi_{p,1}$ (1 module having 6 stripes): *total of 17 variables*

| <i>variable</i> | <i>description</i> | <i>dim</i> |
|-----------------|----------------------------------|------------|
| x | stripe offsets w.r.t to x axis | 6 |
| y | stripe offsets w.r.t to y axis | 6 |
| g | gap distance between stripes | 5 |

Each stripe angle can be determined by metrology, so that we eliminate 3×24 variables from our alignment problem in the case of the non-planar device $\Pi_{n,4}$. The influence of each variable on the final imaging system is influenced by the distance from sample to detector. These are the cases of ΔL and $\{rx, ry, rz\}$ in the non-planar situation.

For distances sample/detector typically greater than $500mm$, we can see that the influence of the angle variables decrease considerably. Therefore, as indicated by the discussion in (B.28), the effective pixel size converges to the pixel size of the detector and the discontinuity errors on the final image are strongly influenced only by horizontal/vertical shifts of each stripe, as well as the gap distance for the planar case. Operating with a detector distance greater than $500mm$, this is the case of beamlines CATERETE (Meneau *et al.*, 2021) and CARNAUBA (Tolentino *et al.*, 2021).

Annotation prior to optimization In order to optimize the offset (and gaps) for the detector device family $\{\Pi_{p,1}, \Pi_{p,4}, \Pi_{n,1}, \Pi_{n,4}\}$, we first have to annotate (manually or automatically) a given set of points in the image restored domain. This situation is illustrated in Figure 4, where two annotated points p_k and p_j indicate a matching pattern that must be respected. Since each annotation corresponds uniquely to a



pixel in the device domain, we know how to map the distances of annotated features in either domains, i.e.,

$$\mathbf{p}_k = \mathbf{R}_{(u,w)}[\mathbf{d}_k], \quad \dim(u) = M, \quad \dim(w) = N \quad (6)$$

with $\mathbf{R}_{(u,w)}$ the remapping operation depending on $M+N$ variables (151, 39, 68 or 17), and u the ones that were selected to be optimized. The remaining variables w assume project values or are determined by other experimental strategies. For simplicity, we indicate \mathbf{R}_u , dropping variable w , which is fixed.

Keeping fixed the annotated pixels $\{\mathbf{d}_k\}$ lying on the device domain, we want to find variable u such that $\{\mathbf{p}_k\}$ satisfies a matching pattern equation. For some of the samples that produce interesting patterns on the measured image, as the Lysosyme crystal or the Siemens star, it is more difficult (but not impossible) to determine a formal equation for the pattern because we rely on the slope (or curvature) of the discontinuity on the boundary of each stripe. To tackle this problem, the LNLS detectors group(DET/DAP, 2022) have produced a template consisting on a sequence of well known squares, that is easy to be screwed in the front of the detector. Those squares produce a sequence of misaligned shadows on each sensor in such a way that when restored in the virtual imaging plane, we should be able to *see* well behaved square features. In this sense, we can collect a set of annotation points in the restored

domain such that the following set of equations is satisfied

$$\begin{cases} (\mathbf{p}_k)_x = (\mathbf{p}_j)_x, & k, j \in \mathcal{V} \\ (\mathbf{p}_r)_y = (\mathbf{p}_i)_y, & r, i \in \mathcal{H} \\ \text{dist}(\mathbf{p}_l, \mathbf{p}_o) = D, & l, o \in \mathcal{E} \end{cases} \quad (7)$$

The annotation sets \mathcal{V} , \mathcal{H} and \mathcal{E} , standing for vertical, horizontal and Euclidean annotations respectively, are sufficient to constrain square features on the restored image. Hence, we look for a solution u^* , in the least square sense, for the following optimization problem:

$$\begin{aligned} u^* = \operatorname{argmin}_{u \in \mathbb{R}^M} & \sum_{k,j \in \mathcal{V}} (\mathbf{R}_u[\mathbf{d}_k]_x - \mathbf{R}_u[\mathbf{d}_j]_x)^2 \\ & + \sum_{r,i \in \mathcal{H}} (\mathbf{R}_u[\mathbf{d}_r]_y - \mathbf{R}_u[\mathbf{d}_i]_y)^2 \\ & + \sum_{l,o \in \mathcal{E}} (\|\mathbf{R}_u[\mathbf{d}_l] - \mathbf{R}_u[\mathbf{d}_o]\| - D)^2 \end{aligned} \quad (8)$$

5. Numerical examples

We start with the optimization of the device $\Pi_{n,4}$ using the screwed template in front of the detector head. The template itself lies on the imaging virtual plane, similar to the one presented in Figure 3. Using a conventional x-rays source the template produce the detected pattern presented in Figure 4.(a), which is similar to the simulated pattern of Figure 2.(a). In order to optimize the geometry, we have used an in-house developed web system to annotate points on the image determined by each module rotated by $\{0, 90^\circ, 180^\circ, 270^\circ\}$ so that annotation could be made easier. An example of annotation is presented in Figure 4.(b), where a collection of points (indicated by tiny square dots) indicate a possible match, either on the horizontal or vertical axis. For instance, the pair of pixels annotated pixels $\{E, F\}$ must have the same horizontal coordinate after the remapping operation (A.24). The same reasoning applies for the pair of points $\{A, B\}$ and $\{C, D\}$, which must have the same vertical coordinates after the remapping operation. The optimization process discussed in (8) using the SCIPY (Virtanen *et al.*,

2020) optimization tool package (with COBYLA as the minimization method) runs in less than one minute, providing us with an optimal offset configuration such that the image presented in Figure 4.(c) can be obtained after the remapping operation.

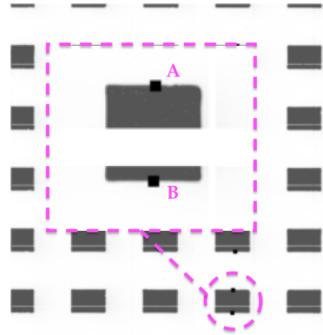


Fig. 5. Annotation example for planar detectors $\Pi_{p,4}$ and $\Pi_{p,1}$. Points A, B must have a remapped known Euclidean distance.

The same optimization methodology was also applied to the detector $\Pi_{p,4}$. The annotation set is presented in Figure 5, where now the Euclidean distance between points $\{A, B\}$ is imposed (since the template geometry is known *a priori*) in the minimization process (8). In order to validate the optimized gaps and offset, a Lysosyme crystal ring measurement is presented in Figure 5.(a) using the obtained optimal geometric condition. This image was obtained at the EMA beamline(dos Reis *et al.*, 2020). A raw image obtained with the device $\Pi_{p,1}$ is presented in Figure 5.(b) with data from CARNAÚBA(Tolentino *et al.*, 2021) beamline, with his correspondent remapped image in Figure 5.(c). For completeness, an example with the detector $\Pi_{n,1}$ is presented in Figure 6 using a Lab6 crystal structure at the Crystallography beamline MANACÁ(Nascimento *et al.*, 2021) at a distance of 40mm. Experimental stations $\{\text{EMA, CARNAÚBA, MANACÁ}\}$ are from Sirius, the Brazilian Synchrotron light source.

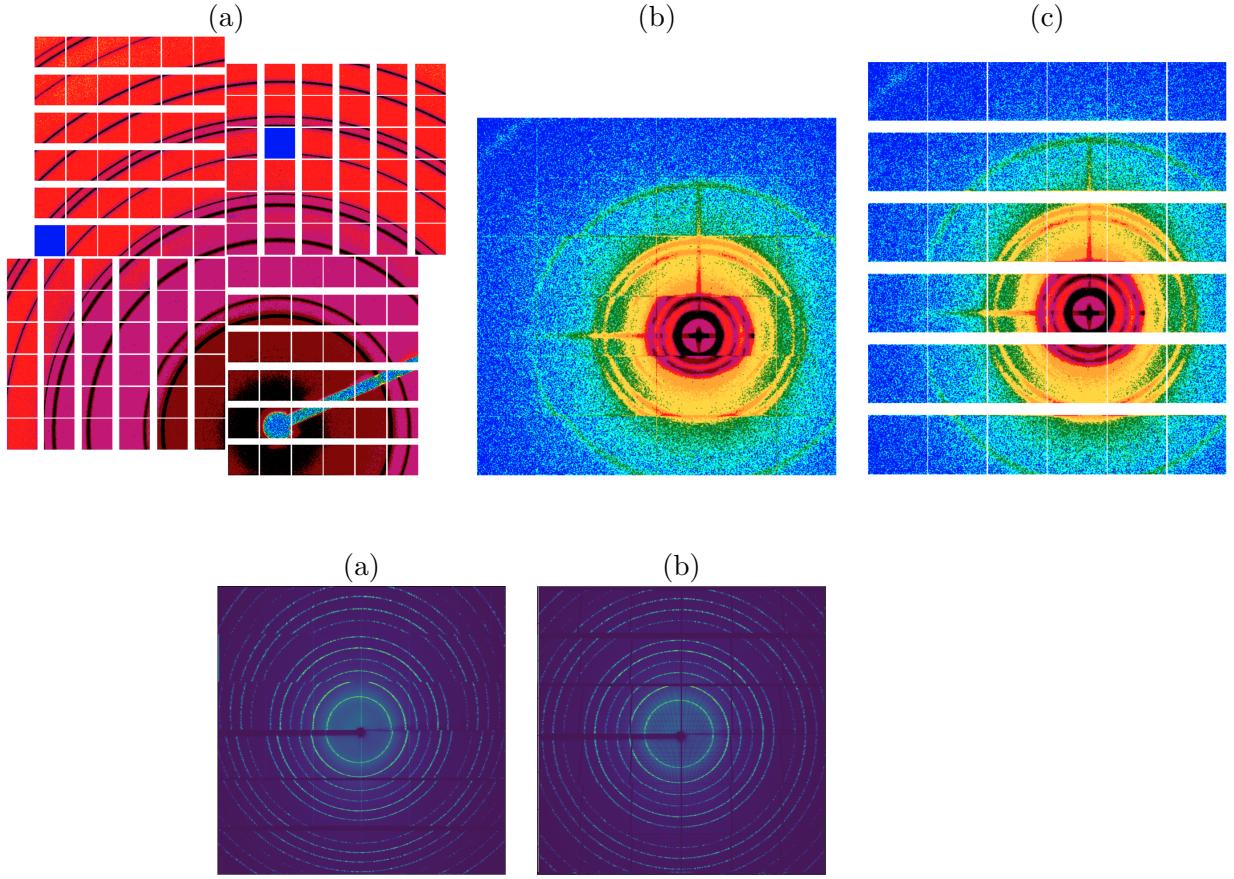


Fig. 6. Remapping operation for the detector $\Pi_{n,1}$ using a LaB6 crystal structure at a distance of 40 mm. (a) Non consistent raw data (b) Restored data.

6. Discussion

Validating the restored imaging set depends on the experimental technique behind measurements. For example, the image presented in Figure 5.(a) can be validated using the aforementioned azimuthal integration software (Hammersley *et al.*, 1997; Prescher & Prakapenka, 2015; Ashiotis *et al.*, 2015). In order to validate the proposed image restoration, for the most difficult case of $\Pi_{n,4}$, a coherent diffraction imaging data obtained at the CATERETÉ(Meneau *et al.*, 2021) beamline was used. Figure 7.(a) depicts the raw data of a Siemens star (ss) - with each module rotated, in accordance with the simulation of Figure 2. After the restoration process, we obtain the

pattern presented in Figure 7.(b), where now each feature (lines with different slopes) of the ss in the frequency domain are presented as a family of continuous curves. A ptychographic scanning procedure using the reconstruction algorithms presented in (Baraldi *et al.*, 2020) was used to validate the restoration algorithm. In a first attempt, each module of $\Pi_{n,4}$ was rotated to $0, 90^\circ, 180^\circ, 270^\circ$ so that an approximation of the diffraction pattern could be used for the ptychographic numerical scheme. The result is presented in Figure 7.(c), where we see that only features of the ss lying on each module separately, could be properly recovered, due to the overlap of the diffraction imaging set on these regions. The ptychographic reconstruction using the restoration algorithm presented in (24) is presented in Figure 7.(d), where reconstruction consistency is higher and small features of the ss could be recovered.

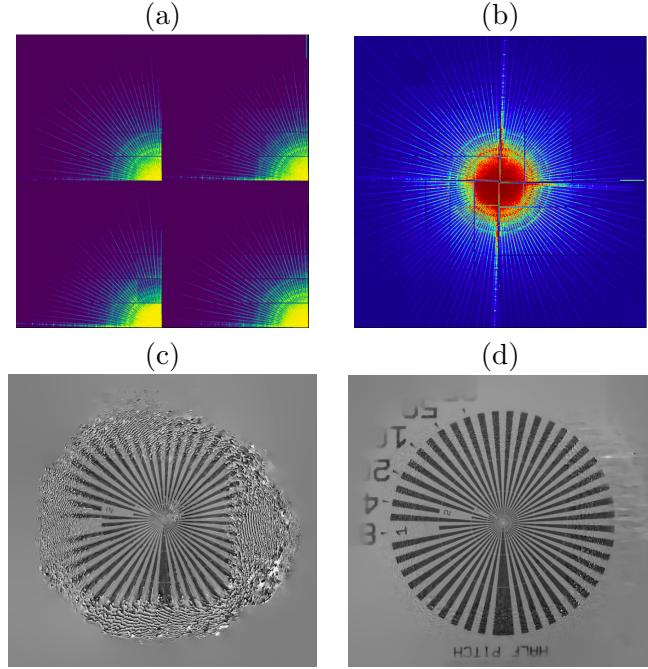


Fig. 7. (a) Real data obtained at CATERETE beamline (b) Restored diffraction imaging data (c) Ptychographic reconstruction without data restoration (d) Ptychographic reconstruction with data restoration.

Each stripe composing the detector family $\{\Pi_{p,4}, \Pi_{p,1}, \Pi_{n,4}, \Pi_{n,1}\}$ was constructed

using the gap-less philosophy, which means that 256×256 chips have large pixels at the boundaries of each intersection. This situation is illustrated in Figures 11 and 13. Using the measured information at these large pixels requires a mathematical operation that uses the information from neighbours pixels with the help of a probabilistic distribution of electronic charges within the pixel area. This study was out the scope of this work and all images were restored ignoring the boundary/large pixels since they are considered suspicious for some reconstruction numerical schemes.

The implementation of the restoration process was done using Python as the main source code for obtaining the geometry. In fact, the detector geometry does not change over large periods, and it is easier to implement (and maintain) this process using a high-level programming language. In this sense, all meshes providing the remapping operation from the detector space, to a image lying at virtual imaging space, is first obtained at Python. This process itself can take roughly 1 second for images having 3072×3072 pixels. The restoration evaluation over the obtained mesh is faster, and can be required to be used over a sequence of experimental images. For this task, the evaluation process was implemented in CUDA and distribution of data, e.g., for a ptychographic scan of 1000 images, is made using multiprocessing strategies according to the number of available GPUS.

Appendix A Remmapping operation

The equation of a plane determining the stripe π_j is

$$\pi_j = \{\mathbf{x} \in \mathbb{R}^3 : \mathbf{n}^T \mathbf{x} = -L_j\} \quad (9)$$

where L_j is a known distance. In fact, we assume that L_0 can be approximated a

priori and δ (see Figure 9) is the physical distance between each stripe, i.e.,

$$L_j = L_0 + j\delta \quad (10)$$

From \mathbf{n} we can easily set the orthogonal basis $\{\mathbf{n}, \mathbf{n}^\times, \mathbf{n}^\perp\}$ according to

$$\mathbf{n}^\times = (-n_2 n_1, n_3^2 + n_1^2, -n_2 n_3)^T \quad (11)$$

and

$$\mathbf{n}^\perp = (n_3, 0, -n_1)^T \quad (12)$$

in such a way that $\{\mathbf{n}^\times, \mathbf{n}^\perp\}$ lies within the strip π_j , for all j .

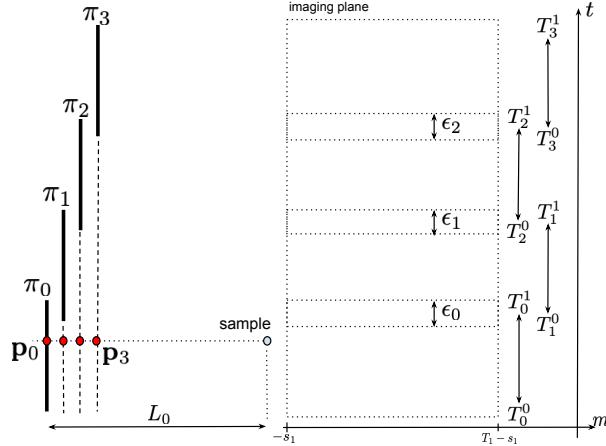


Fig. 8. Points of normal intersection over each stripe.

Considering the scatter point located at the point \mathbf{z} in the coordinate system $x_1x_2x_3$, the point of normal incidence (PONI) of z at each stripe π_j is referred as \mathbf{p}_j given by

$$\mathbf{p}_j = \mathbf{z} + \lambda \mathbf{n} \quad (13)$$

where λ is such that $\mathbf{p}_j \in \pi_j$, i.e.,

$$(\mathbf{z} + \lambda \mathbf{n})^T \mathbf{n} = -L_j \iff \lambda = \mathbf{z}^T \mathbf{n} - L_j$$

The point of normal incidence becomes

$$\mathbf{p}_j = (I + \mathbf{n} \mathbf{n}^T) \mathbf{z} - L_j \mathbf{n} \quad (14)$$

or, for the case of $\mathbf{z} = \mathbf{0}$,

$$\mathbf{p}_j = -L_j \mathbf{n} \quad (15)$$

We denote $\mathbf{G} = I + \mathbf{n}\mathbf{n}^T$ as the projection matrix with direction \mathbf{n} . In the majority of cases, $\mathbf{z} = \mathbf{0}$, except for scatter points lying the in vicinity of the origin. Figure 8 presents a side view of PONIS $\{\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\}$ assuming 4 parallel stripes π_j .

Each point \mathbf{p} lying in the stripe π_j is a linear combination of normal vectors $\{\mathbf{n}^\times, \mathbf{n}^\perp\}$ starting at the PONI \mathbf{p}_j , i.e.,

$$\mathbf{p}(m, t) = \mathbf{p}_j + t\mathbf{n}^\times + m\mathbf{n}^\perp \quad (16)$$

where (m, t) determine the physical positions of what we can call a *detection active area*. A detailed discussion for the mesh device, determining each pair (t, m) can be found in Appendix C.

A.1. Remapping operation

Each stripe has its own normal vector \mathbf{n} and therefore a local orthogonal basis $\{\mathbf{n}^\perp, \mathbf{n}^\times, \mathbf{n}\}$ can be easily determined. Since each stripe is mounted on a physical *bed* - see Figure 3.(b) - there may exist small deviations on the angles determining \mathbf{n} , as well as an horizontal/vertical offsets of each stripe. The normal vector \mathbf{n} is obtained after rotations of the normal vector $(0, 0, 1)^T$ with respect to angles r_x, r_y, r_z , that is

$$\mathbf{n} = \mathbf{n}(r_x, r_y, r_z) = \mathbf{R}_z \mathbf{R}_y \mathbf{R}_x \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad (17)$$

where $\{\mathbf{R}_x, \mathbf{R}_y, \mathbf{R}_z\}$ are 3D standard rotation matrices related to angles $\{r_x, r_y, r_z\}$ respectively. We assume that \mathbf{n} is *not* constant for all stripes π_{ij} , so that there are a total of 3×24 varying angles. For simplicity, we take $\mathbf{n} = \mathbf{n}_j$ as one of the six normal vectors, orthogonal to a given stripe π_j .

The virtual detector plane, lying in the virtual imaging plane - see Figure 1.(a) -

depends on its own normal vector direction \mathbf{v} , therefore having equation given by

$$\pi_{\text{ref}}(\mathbf{v}) = \{\mathbf{x} \in \mathbb{R}^3 : \mathbf{x}^T \mathbf{v} = -L\} \quad (18)$$

with $L = L_0$ the approximate footprint distance to the sample (scattering point). Here \mathbf{v} is a *tilt* of vector $\mathbf{e}_3 = (0, 0, 1)^T$. We can parameterize \mathbf{v} using spherical coordinates to define appropriate tilting angles for the imaging plane. In the ideal case, $\mathbf{v} = \mathbf{e}_3$. According to Figure 9, each stripe π_j has a distance L_j to the plane passing through the sample and parallel to all stripes. Due to a simple geometric reasoning, we can obtain L_j as a function of θ_j , L and δ , i.e.,

$$L_j = L / \cos \theta_j - j\delta \quad (19)$$

with $j \in [0, 5]$. Assuming M stripes, each with P squared $J \times J$ detectors, our digital matrix will have order $PJ \times MJ$. As indicated by (16), each stripe have points \mathbf{p} over a non-uniform grid (m_u, t_r) according to

$$\mathbf{p}(t_r, m_u) = \mathbf{p}_j + t_r \mathbf{n}^\times + m_u \mathbf{n}^\perp \quad (20)$$

$$= \mathbf{Gz} - L_j \mathbf{n} + t_r \mathbf{n}^\times + m_u \mathbf{n}^\perp \quad (21)$$

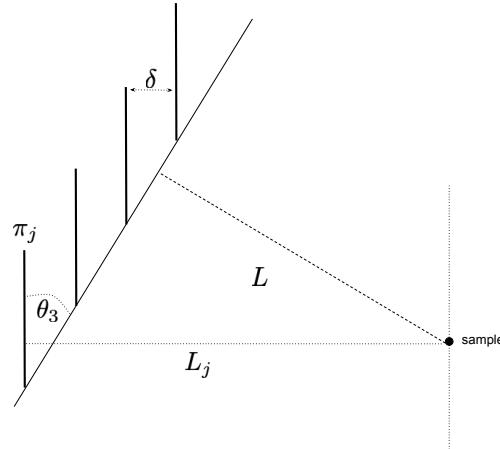


Fig. 9. Distance L_j between each stripe π_j and the sample position, according the a given reference distance L .

We now finally project each point \mathbf{p} determined by (21), on the direction $\mathbf{p} - \mathbf{z}$, over the detector plane π_{ref} , where L and \mathbf{v} are given. Such a point is determined by the existance of a constant λ such that

$$\mathbf{q} = \lambda(\mathbf{p} - \mathbf{z}) \quad (22)$$

In this case, it is easy to show that λ is $\lambda = -L/(\mathbf{p} - \mathbf{z}) \cdot \mathbf{v}$. We are only interested in the q_x, q_y positions of $\mathbf{q} = (q_x q_y, q_z)$, since they will define the virtual image that is going to be processed later. Taking $\mathbf{q}^\perp = (q_x, q_y)$ and considering $\mathbf{z} = \mathbf{0}$ for simplicity, the remapping operation from the physical device to the virtual imaging plane becomes the non-linear operation

$$\mathbf{q}^\perp = \mathbf{R}(L, \mathbf{n}, \mathbf{v}) \quad (23)$$

or

$$\begin{pmatrix} q_x \\ q_y \end{pmatrix} = \begin{pmatrix} -L \frac{(-L_j n_1 + t n_1^\times + m n_1^\perp)}{-L_j (\mathbf{n} \cdot \mathbf{v}) + t (\mathbf{n}^\times \cdot \mathbf{v}) + m (\mathbf{n}^\perp \cdot \mathbf{v})} \\ -L \frac{(-L_j n_2 + t n_2^\times + m n_2^\perp)}{-L_j (\mathbf{n} \cdot \mathbf{v}) + t (\mathbf{n}^\times \cdot \mathbf{v}) + m (\mathbf{n}^\perp \cdot \mathbf{v})} \end{pmatrix} \quad (24)$$

Solving (24) for (t, m) we obtain a two-dimensional linear system with a non-singular matrix (a_{ij}) and linear term (b_j) , i.e.,

$$\mathbf{A} \begin{pmatrix} m \\ t \end{pmatrix} = \mathbf{b} \iff \begin{pmatrix} m \\ t \end{pmatrix} = \mathbf{A}^{-1} \mathbf{b} \quad (25)$$

with \mathbf{A}^{-1} being defined as the inversion of the matrix \mathbf{A} over the linear term \mathbf{b} , and

$$\begin{aligned} a_{00} &= q_x(\mathbf{n}^\perp \cdot \mathbf{v}) + L n_1^\perp, & a_{01} &= q_x(\mathbf{n}^\times \cdot \mathbf{v}) + L n_1^\times \\ a_{10} &= q_y(\mathbf{n}^\perp \cdot \mathbf{v}) + L n_2^\perp, & a_{11} &= q_y(\mathbf{n}^\times \cdot \mathbf{v}) + L n_2^\times \\ b_0 &= L L_j n_1 + x L_j (\mathbf{n} \cdot \mathbf{v}), & b_1 &= L L_j n_2 + y L_j (\mathbf{n} \cdot \mathbf{v}) \end{aligned} \quad (26)$$

giving us the final relation between a point (q_x, q_y) on the virtual imaging plane $\pi_{\text{ref}}(\mathbf{v})$ - tilted through $\mathbf{v} \in \mathbb{R}^3$ - and a point (m, t) on the physical stripe π_j , for all j .

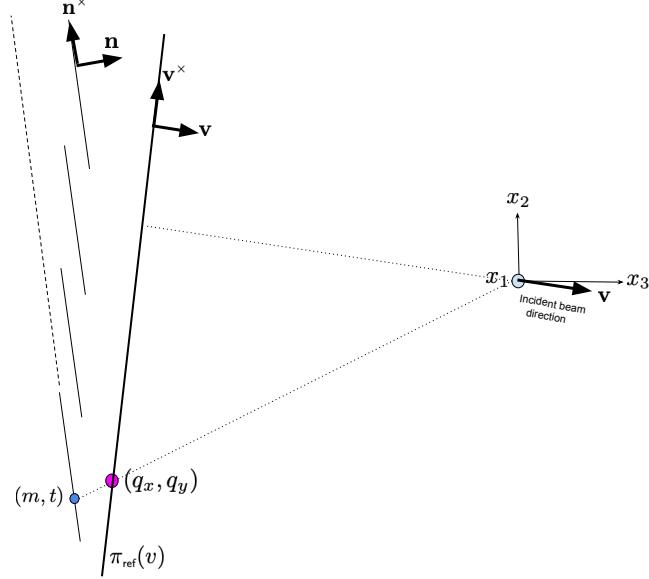


Fig. 10. Geometry from side view perspective (x_2x_3 plane) and remapping of $\mathbf{p}(m, t)$ on the virtual imaging plane $\mathbf{q}^\perp = (q_x, q_y)$.

Appendix B Effective pixel size

The geometry of our problem, on the plane x_2x_3 is presented in Figure 10. Here we consider that a single point, with coordinates (m, t) at a given stripe, is mapped to a unique point (q_x, q_y) under the non-linear transformation (24). It is important to note that, under the non-linear mapping (24), values of q_x and q_y are dominated by L and L_j , which are much greater than the inner products $\mathbf{n} \cdot \mathbf{v}$, $\mathbf{n}^\perp \cdot \mathbf{v}$ and $\mathbf{n}^\times \cdot \mathbf{v}$. Since $\mathbf{n} \cdot \mathbf{v} = \cos \alpha(\mathbf{n}, \mathbf{v})$ with $\alpha \sim \pi$ the angle between \mathbf{n} and \mathbf{v} , we can state that a first order approximation for \mathbf{q}^\perp is

$$\begin{pmatrix} q_x \\ q_y \end{pmatrix} \sim -\frac{L}{L_j} \begin{pmatrix} -L_j n_1 + t n_1^\times + m n_1^\perp \\ -L_j n_2 + t n_2^\times + m n_2^\perp \end{pmatrix} \quad (27)$$

which holds because $\mathbf{n}^\times \cdot \mathbf{v} \sim 0$ and $\mathbf{n}^\perp \cdot \mathbf{v} \sim 0$. The non-linearity is increased for small distances as the angle between the basis $\{\mathbf{n}, \mathbf{n}^\perp, \mathbf{n}^\times\}$ and the vector \mathbf{v} could not be neglected.

The Jacobian matrix affect the expansion or shrinking of the constructed image on the virtual imaging plane. It happens that $\Delta q_x \Delta q_y \sim |\det J| \Delta m \Delta t > \Delta m \Delta t$; therefore, we will have a contraction of the remapped image. For large distances, we use (27) to observe that the Jacobian is approximated by

$$J \sim -\frac{L}{L_j} \begin{pmatrix} n_1^\times & n_1^\perp \\ n_2^\times & n_2^\perp \end{pmatrix} = -\frac{L}{L_j} \begin{pmatrix} n_3 & -n_2 n_1 \\ 0 & n_3^2 + n_1^2 \end{pmatrix}.$$

Hence,

$$|\det J| \sim \frac{L}{L_j} n_3 (n_3^2 + n_1^2) \quad (28)$$

We can also show that in the limit of $L \rightarrow \infty$, $\Delta q_x \Delta q_y \rightarrow \Delta t \Delta m$, which is specially important for beamlines with larger distances.

Boundary pixels: The logical measured image, has several unreliable pixels, mostly in the vertical and horizontal boundaries between each chip. For each one of these pixels, we have set their values to -1 in order to avoid their processing. Setting an appropriate value for these pixels is a secondary problem, which can be solved later. We can set a parameter determining the number of pixels that are removed from each chip boundary. We identify boundary pixels at Figure 11 from Appendix C.

Appendix C Detector physical mesh

For many sensors, the detection active area is a synonym to a pixel and here this is also the case, with the exception of a detective active area located on the borders of each stripe. Figure 11 illustrate two stripes $\{\pi_0, \pi_1\}$, each one determined by two

square chips with non-esquispaced logical detection points (*bump bonds*), as denoted by the $J \times J$ mesh, with $J = 6$. Here we can see the larger detection active area at the borders of each stripe, but also in the boundary of chips, without physical gaps.

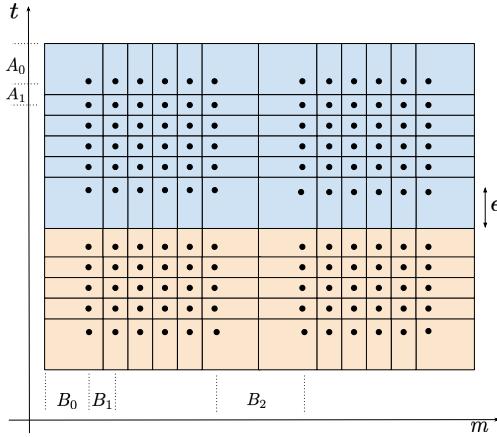


Fig. 11. Mesh of physical points at the device. Simple example of 4 chips (2 per stripe) of 6×6 points. Distances $\{\mathbf{A}, \mathbf{B}, \epsilon\}$ can be estimated *a priori*.

The distance between bump bonds is considered as a reference for the construction of our mesh, as indicated by Figure 12. For a single composition of 6 chips, and on the direction axis m , we define a temporary mesh of points \bar{m} based on the bumping bond distance, given by

$$\begin{aligned} \bar{m}_{u+1} - \bar{m}_u &= B_1, \quad u \geq 1 \\ \bar{m}_1 - \bar{m}_0 &= B_0 \\ \bar{m}_J - \bar{m}_{J-1} &= B_2 \quad (\text{boundary between chips}) \end{aligned} \tag{29}$$

with values $\{B_0, B_1, B_2\}$ given *a priori*. As for the direction axis t , since the gap ϵ is larger than a physical pixel (that is, the detection active area), there is a negative step from the last bump bond of a chip and the next one. Hence, we define a temporary mesh of points \bar{t} also based on the bumping bond distances,

$$\begin{aligned} \bar{t}_{r+1} - \bar{t}_r &= A_1, \quad r \geq 1 \\ \bar{t}_1 - \bar{t}_0 &= A_0 \\ \bar{t}_J - \bar{t}_{J-1} &= -A_2 \quad (\text{boundary between chips}) \end{aligned} \tag{30}$$

with values $\{A_0, A_1, A_2\}$ also given *a priori*. Axis \bar{t} and \bar{m} are presented in Figure 12.

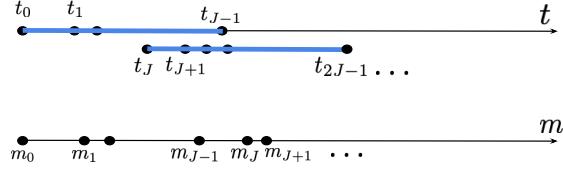


Fig. 12. Axis m and t at the point of normal incidence plane.

Considering the imaging plane as a reference - see Figure 3 - we assume that distances \bar{u} , \bar{r} (pixel unity) from the bottom-left corner, can be estimated *a priori*. Also, the physical lengths T_1 , T_2 (width and height) - see Figure 1.(b) - can also be estimated before measurements, having an impact on the definition of the mesh (m, t) . Here, (\bar{u}, \bar{r}) indicates the center of the beam at the imaging plane, not to be confused with the point of normal incidence. Since $(m_{\bar{u}}, t_{\bar{r}})$ is an approximation of the center (sample position on the imaging plane), we now build our coordinate system (m, t) as

$$t_r = \bar{t}_r - t_{\bar{r}}, \quad m_u = \bar{m}_u - m_{\bar{u}} \quad (31)$$

Also, assuming that the gap between strips is covered by a distance ϵ_j - see Figure 8 - each point p determined by the linear combination (16) lying at the strip π_j , has the t axis varying on a fixed interval,

$$t \in [T_j^0, T_j^1]$$

Here, we have assumed that ϵ_j remains constant for all overlapping regions.

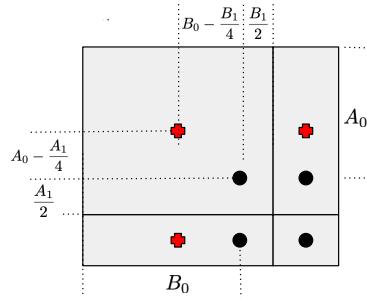


Fig. 13. Virtual shift (red) for the mesh of bump bonds (black).

If the reference for building a mesh of points is not the sequence of bump bonds, but the center of each physical pixel, there is a constant shift of $\pm\hat{m}$ on the left/right borders and $\pm\hat{t}$ at the bottom/top of each bump bond, as indicated by Figure 13.

Here,

$$\hat{t} = A_0 - \frac{A_1}{4}, \quad \hat{m} = B_0 - \frac{B_1}{4} \quad (32)$$

Therefore, we take the same mesh defined in (33) except that

$$\begin{aligned} t_{kJ} &= \bar{t}_{kJ} - t_{\bar{r}} - \hat{t}, \\ m_{kJ} &= \bar{m}_{kJ} - m_{\bar{u}} - \hat{m}, \\ t_{k(J-1)} &= \bar{t}_{k(J-1)} - t_{\bar{r}} + \hat{t}, \\ m_{k(J-1)} &= \bar{m}_{k(J-1)} - m_{\bar{u}} + \hat{m}, \quad k = 0, 1, 2, \dots \end{aligned} \quad (33)$$

Acknowledgments

I would like to acknowledge the Brazilian Ministry of Science, Technology, and Innovation (MCTI) for funding this work, through the Brazilian Center for Research in Energy and Materials (CNPEM). The datasets presented in this work was kindly provided by the imaging beamlines CATERETÉ, CARNAUBA and EMA, from CNPEM/Sirius (the 4th generation Brazilian synchrotron). I also thank Renato Candido Reis and Gustavo Siqueira Gomes from the LNLS detectors group for providing a physical template, so that a set of optimization codes could be properly developed. Special thanks to Matheus L. Bernardi from the Control Software group for developing a friendly web interface for the annotation of pixels prior to offset optimization. I also would like to thank Daniel Tavares, head of the Data Acquisition and Processing Division at LNLS, for suggestions and proofreading the article.

About the detector

The geometric problem presented in this work was requested as an algorithmic subproblem, among many other developments between the LNLS group of detectors (DET/DAP, 2022) and the partner company Pitec(PITEC, 2022). The geometric restoration presented in this manuscript is part of the software distributed along with the detector and necessary for its operation.

References

- Ashtiotis, G., Deschildre, A., Nawaz, Z., Wright, J. P., Karkoulis, D., Picca, F. E. & Kieffer, J. (2015). *Journal of applied crystallography*, **48**(2), 510–519.
- Baraldi, G. L., Dias, C. S., Silva, F. M., Tolentino, H. C. & Miqueles, E. X. (2020). *Journal of Applied Crystallography*, **53**(6), 1550–1558.
- Bauschke, H. H., Combettes, P. L. & Luke, D. R. (2002). *JOSA A*, **19**(7), 1334–1345.
- Chapman, H. N., Barty, A., Marchesini, S., Noy, A., Hau-Riege, S. P., Cui, C., Howells, M. R., Rosen, R., He, H., Spence, J. C. et al. (2006). *JOSA A*, **23**(5), 1179–1200.
- Gommès, C. J. & Goderis, B. (2010). *Journal of Applied Crystallography*, **43**(2), 352–355.
- Hammersley, A. et al. (1997). *European synchrotron radiation facility internal report ESRF97HA02T*, **68**, 58.

- Le Bourlot, C., Landois, P., Djaziri, S., Renault, P.-O., Le Bourhis, E., Goudeau, P., Pinault, M., Mayne-L'Hermite, M., Bacroix, B., Faurie, D. *et al.* (2012). *Journal of Applied Crystallography*, **45**(1), 38–47.
- Meneau, F., Passos, A., Garcia, P., Vinaches, P., Manoel, L., Kalile, T., Zerba, J., Rodrigues, G., Miqueles, E., Baraldi, G. *et al.* (2021). In *Acta Crystallographica A-Foundation and Advances*, vol. 77, pp. C283–C283. Int. Union Crystallography 2 Abbey SQ, Chester, CH1 2HU.
- Nascimento, A., Araujo, E., Hagio, C., Almeida, S., Rodrigues, A. C., Barretto Barbosa Trivella, D., Bruder, M., Rustiguel, J. K., Dislich Ropke, C., Ravanelli Pessa, L. *et al.* (2021). *Synchrotron Radiation News*, **34**(5), 3–10.
- Pfeiffer, F. (2018). *Nature Photonics*, **12**(1), 9–17.
- Ponchut, C., Tartoni, N. & Pennicard, D. (2021). *Radiation Measurements*, **140**, 106459.
- Prescher, C. & Prakapenka, V. B. (2015). *High Pressure Research*, **35**(3), 223–230.
- dos Reis, R. D., Kaneko, U. F., Francisco, B. A., Fonseca Jr, J., Eleoterio, M. A. & Souza-Neto, N. M. (2020). In *Journal of Physics: Conference Series*, vol. 1609, p. 012015. IOP Publishing.
- DET/DAP, (2022). Detector's group from *Data Acquisition and Processing Division*, CNPEM, *Brazilian Synchrotron Light Laboratory*. <https://www.lnls.cnpm.br/grupos/det-en/>. [Online; accessed 2022-10-17].
- PITEC, (2022). Communication and imaging systems. <https://www.pitec.co>. [Online; accessed 2022-10-17].
- Tolentino, H. C., Geraldes, R. R., Moreno, G. B., Pinto, A. C., Bueno, C. S., Kofukuda, L. M., Sotero, A. P., Neto, A. C., Lena, F. R., Wilendorf, W. H. *et al.* (2021). In *X-Ray Nanoimaging: Instruments and Methods V*, vol. 11839, pp. 5–14. SPIE.
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, É. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P. & SciPy 1.0 Contributors (2020). *Nature Methods*, **17**, 261–272.