



HAL
open science

Quality control of 3D reflective tomography

Jean-Baptiste Bellet, Gérard Berginc

► **To cite this version:**

Jean-Baptiste Bellet, Gérard Berginc. Quality control of 3D reflective tomography. 2017. hal-01368354v2

HAL Id: hal-01368354

<https://hal.science/hal-01368354v2>

Preprint submitted on 4 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Quality control of 3D reflective tomography

JEAN-BAPTISTE BELLET

GÉRARD BERGINC

* Institut Elie Cartan de Lorraine, Université de Lorraine and CNRS, UMR 7502, 3 rue Augustin Fresnel, BP 45112, 57073 Metz Cedex 03, France

E-mail address: jean-baptiste.bellet@univ-lorraine.fr

† Thales Optronique, 2 avenue Gay Lussac, CS 90502, 78995 Élan court Cedex, France

E-mail address: gerard.berginc@fr.thalesgroup.com.

ABSTRACT. Reflective tomography is an emerging method in three-dimensional optical imaging. It is unknown in mathematical communities, whereas justifying this process is a mathematical challenge. Thus we propose to formulate a mathematical model describing a full process of 3D reflective tomography. Then we check numerically the quality of the method over original examples, using new parameters of quality control. This work is aimed mainly at applied mathematicians or opticians who would like to discover or deepen 3D reflective tomography.

1. Introduction

1.1. Three-dimensional optical imaging

There is a considerable interest in the development of new optical imaging systems that are able to give three-dimensional images. Potential applications range across the field of defense and security for the recognition of targets, the medical field for the detection of subcutaneous and cutaneous tumors or the archaeological field for the discovery of remains in forests.

The concept of reflective tomography was introduced to overcome such challenges. The main idea is applying a tomographic algorithm on reflective data: a filtered backprojection [13]. It was observed that it works for several kinds of physical experiments [13]. It can reconstruct surfaces from phaseless images [6, 9–11, 15]; and it is known in optical engineering that it can recover partially occluded objects [7]. A new patented technology is based on 3D reflective tomography [4, 5, 8].

1.2. Mathematical gap

To the authors' knowledge, reflective tomography is mainly known in optical engineering; it is still unknown in mathematical communities, whereas it introduces interesting mathematical challenges. Indeed the standard mathematical result states that the filtered backprojection inverts the Radon transform: this is the justification of transmission tomography for complete data, where the full Radon transform is known. It is also known in the field of transmission tomography that serious difficulties appear for incomplete data (*e.g.* limited angle problems), such as instabilities [14]. But in reflective tomography, because of occlusions, the data are not modeled by a Radon transform and they are incomplete, whereas the numerical reconstructions show that the method works. As a result the two following questions are very interesting from a mathematical point of view. Find a general mathematical framework that describes reflective tomography. Prove that reflective tomography reconstructs partially the scene and describe the artifacts. A model is proposed in [3], but only for the two dimensional problem. And concerning the proof, works are in progress [1].

1.3. Proposed strategy

In this paper we discuss reflective tomography in the following way. We first derive a generic mathematical model for 3D reflective tomography, by extending the 2D model derived in [3]. We want a model which can serve as a basis for further mathematical considerations on reflective tomography; so we respect the following constraints: the model must be simple, but realistic enough. Concerning the validation of the process, we do not prove mathematical theorems, but we propose an original set of numerical tests. As usual in image processing, the main criterion of quality will be the visual rendering. To go further in this paper, we also define new quantitative criteria, usable for synthetic tests.

1.4. Main results

First, we succeed in deriving a model which describes the full process of 3D reflective tomography, in a simple but rigorous way. The process consists in generating new images of a scene from cone beam reflective projections; the full model includes the process of projecting opaque objects, the reconstruction step and the visualization step. Secondly, we succeed in defining quantitative criteria of quality; their values estimate the level of true information in the output of the process. The relevance of 3D reflective tomography is shown on many numerical examples: the observed values of quality confirm that the processed images really represent the surfaces of the initial scene.

1.5. Organization

The paper is organized as follows. In the first part, we model a full process of 3D reflective tomography. In the second part, we control the quality of reflective tomography on a wide variety of examples, including a real data set.

2. Modeling

2.1. Reflective cone beam scanning

We define a mathematical model for the cone beam scanning of a reflective-kind scene; it extends [3].

In the space \mathbb{R}^3 , we consider that an opaque object is the boundary $\Sigma = \partial\Omega$ of a bounded domain $\Omega \subset \mathbb{R}^3$; a surfacic function $f : \Sigma \rightarrow \mathbb{R}$ will be called an intensity of the object Σ . We consider a set of n separated objects, $\Sigma_i, 1 \leq i \leq n$. We also consider a wall, or background, Σ_0 , as a surface whose interior domain contains the whole scene, *i.e.* every Σ_i . This convention for the wall allows to treat the background exactly as the objects of the scene. For convenience, $\Sigma = \cup_{0 \leq i \leq n} \Sigma_i$ is the set of surfacic points; when the f_i are the intensities of the Σ_i , f_Σ is the intensity of Σ : for $v \in \Sigma_i$, $f_\Sigma(v) = f_i(v)$.

Let $\beta \in [0, 2\pi]$ be a fixed angle. We consider a device on the circle $|x| = r > 0$ in the plane $x_3 = 0$, at the position $r\theta = r(\cos \beta, \sin \beta, 0)$. We assume that the device contains a receiver array on which we record a cone beam projection of the scene, along rays through the optical center $r\theta$. The plane $\theta^\perp = \{x \in \mathbb{R}^3 : x \cdot \theta = 0\}$ passes through the origin and is parallel to the receptor array. For the ray $L(\theta, y)$ through $r\theta$ and $y = y_2\theta_\perp + y_3e_3 \in \theta^\perp$, with $\theta_\perp(\beta) = (\sin \beta, -\cos \beta, 0)$ and $e_3 = (0, 0, 1)$, the visible point of the scene is $v(\theta, y)$, which is the first intersection point of the ray $L(\theta, y)$ with the surfaces $\Sigma_i: v(\theta, y) = r\theta + \rho(\theta, y)(y - r\theta)$, with $\rho(\theta, y) = \arg \min\{\rho, \rho > 0, v = r\theta + \rho(y - r\theta) \in \Sigma\}$. We consider information on Σ : $v \in \Sigma \mapsto f_\Sigma(v, \theta)$. The first variable $v \in \Sigma$ represents a surfacic point, whereas the second variable θ indicates that information depends on the angle θ . On the ray $L(\theta, y)$, we record information $F_\theta(y)$ coming from the visible point $v(\theta, y)$: $F_\theta(y) = f_\Sigma(v(\theta, y), \theta)$. This process of reflective

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

projection is illustrated on Figure 1. Assuming that (y_2, y_3) scans a rectangle $[-a, a] \times [-b, b]$, the record is a bi-dimensional image $y \mapsto F_\theta(y)$; we call F_θ the reflective projection, associated with the angle θ .

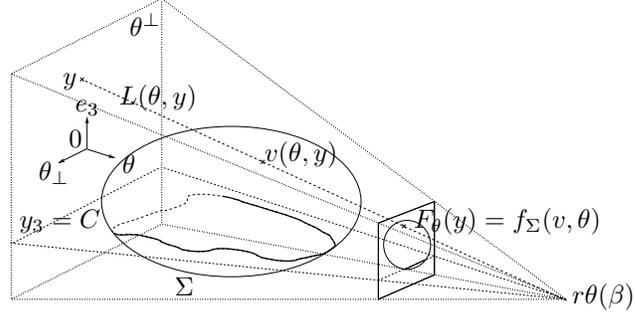


FIGURE 1. Reflective cone beam projection. The ray $L(\theta, y)$ passes through the optical center $r\theta(\beta)$ and the point $y = y_2\theta_\perp + y_3e_3 \in \theta^\perp$. The visible point is $v = v(\theta, y) \in \Sigma$, and the record is: $F_\theta(y) = f_\Sigma(v, \theta)$. A horizontal slice in the recorded image is also represented. The intersection of Σ with the 2D beam $y_3 = C$ is represented (without the background); in bold: the visible part of the surface and its projection.

In the Figure 1, we also represent a 2D beam which intersects the screen on a horizontal line $y_3 = C$. It is reasonable to assume that for such a beam, the resulting horizontal slice in the image F_θ is a piecewise smooth function. As it is observed in subsection 3.3, horizontal discontinuities in the image F_θ appear for at least two reasons:

- geometrical jumps: $y_2 \mapsto v(\theta, y_2, y_3)$ is discontinuous when the visible point v jumps from an object to another object due to an occlusion, or jumps from an object to the same object due to a concavity;
- intensity jumps: the surfacic intensity f_Σ can jump itself, even if $y_2 \mapsto v(\theta, y_2, y_3)$ is continuous.

Finally, we change the acquisition angle and we restart: this experiment is repeated for θ scanning a finite set of angles $\Theta \subset S^1 \times \{0\}$, whose cardinal is $|\Theta|$. Juxtaposing the different images, we get at the end a 3D *reflectogram* $(y, \theta) \mapsto F_\theta(y)$. In the reflectogram, each surfacic point $v \in \Sigma$ is seen partially (or eventually not seen) along the curve $\{(\theta, y) : v \in L(\theta, y)\}$; its intensity level depends on θ . The imaging challenge is transforming the reflectogram onto images that are usable for identification of the surfaces Σ_i .

2.2. Tomography solver

We recall the heuristic of 3D reflective tomography [9]: applying the Feldkamp-Davis-Kress algorithm [12] on the reflectogram.

The first step is weighting the data set:

$$F_w(\theta, y) = w(y)F_\theta(y), \text{ with } w(y) = \frac{r}{(r^2 + y_2^2 + y_3^2)^{0.5}}. \quad (2.1)$$

The next step is computing a tomographic horizontal filtering of the weighted data set F_w . We recall that a regularized kernel of the Hilbert transform is $\varphi = \mathcal{F}^{-1}(-i \operatorname{sign}(\sigma) \cdot \hat{h}(\sigma))$, where

\mathcal{F} is the Fourier transform and $\hat{h}(\sigma)$ is an even windowing function with compact support (σ is the frequency). For all $y_3 \in [-b, b]$ and for all $\theta \in \Theta$, we compute the filtering of $y_2 \mapsto F_w(\theta, y)$:

$$\partial_{y_2} F_w \star \varphi = \mathcal{F}^{-1}(|\sigma| \hat{h}(\sigma) \mathcal{F}(F_w(\theta, y))(\sigma)), \quad (2.2)$$

where the Fourier transform is taken with respect to y_2 . This definition makes sense because we assume that $y_2 \mapsto F_\theta(y)$ is piecewise smooth.

The last step is computing a weighted backprojection f_{FDK} : for each reconstruction point x , we sum over lines through x in the weighted filtered weighted reflectogram:

$$f_{\text{FDK}}(x) = \sum_{\theta \in \Theta} \frac{r^2}{(r - x \cdot \theta)^2} (\partial_{y_2} F_w \star \varphi)(\theta, y), \quad \text{with } y = \frac{rx \cdot \theta_\perp}{r - x \cdot \theta} \theta_\perp + \frac{rx_3}{r - x \cdot \theta} e_3. \quad (2.3)$$

It is known that in the limit $r \rightarrow \infty$, this reconstruction is equal to the reconstruction based on the Radon inversion, along parallel rays, and horizontal slice by horizontal slice.

Concerning practical aspects, the reflectogram F is known on a 3D discrete grid of size $|\Theta| \times N_2 \times N_3$. The horizontal filtering $\partial_{y_2} F_w \star \varphi$ is computed on the same grid, using the Fast Fourier Transform; in this paper, we will choose the Shepp-Logan filter: $\hat{h}(\sigma) = \frac{\sin \frac{\pi}{2} \frac{\sigma}{\sigma_{\max}}}{\frac{\sigma}{\sigma_{\max}}} \mathbb{1}_{-\sigma_{\max} \leq \sigma \leq \sigma_{\max}}$, where σ_{\max} is the Nyquist frequency. The computational cost for filtering is $\mathcal{O}(|\Theta| N_3 N_2 \log N_2)$. The reconstruction f_{FDK} is computed on a 3D grid of voxels. For N^3 voxels, the cost is $\mathcal{O}(N^3 |\Theta|)$; it is known to be the most consuming step. To speed up the computations, strategies of implementation on GPU are available in the literature. Basically, for the last step, each thread can be the computation of the backprojection for one voxel.

2.3. Discussion about the solver

We give an interpretation of a reflective FDK volume, based on the previous mathematical model. First the filtering step is a key point in the processing: $\partial_{y_2} F_w \star \varphi$. The derivative enhances the horizontal contrasts of the (weighted) reflectogram F_w . This step especially detects the contours that are transverse to the horizontal direction in the data set; the kernel φ is odd, so it is a zero-crossing detection.

Then the weighted backprojection accumulates the enhanced contrasts in space. For a generic point x , filtered information through x is *incoherent* and tends to offset itself by summation. But, if x is near a surfacic point v , the associated filtered data may contain *coherent* contrasts; a significant value at x occurs thanks to summation of such coherent contrasts. This discussion motivates the following interpretation: "reflective tomography especially accumulates the coherent horizontal contrasts at their true location in space". This discussion is not mathematically rigorous, but it is the starting point of a mathematical study in 2D [1].

2.4. Efficient visualization

We describe an efficient way of visualizing a volume in 3D reflective tomography [2].

The heuristic computes a 3D volume that needs to be correctly investigated to identify the objects of the original scene. In order to appreciate 3D structures by exploring the full reconstructed volume, we use volume rendering. Basically, volume rendering simulates a scanner: to display a function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$, it computes and displays a projection $P[f]$ on a 2D screen: see Figure 2. We can get several kinds of visualization, depending on the choice of the projection operator. The projection looks often like a \mathcal{L}^p -norm over the ray: for all ray L , the projection along the ray L is: $P[f](L) = \|f|_L\|$. Among these methods, the X-Ray visualization is the X-Ray transform $P[f](L) = \int_L f$, and the Maximum Intensity Projection (MIP) is $P[f](L) = \max_L(f)$.

How to display the reconstruction f_{FDK} ? We expect that surfaces of the scene produce significant values in f_{FDK} , whereas we expect low values at voxels far from the surfaces. As a result, if

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

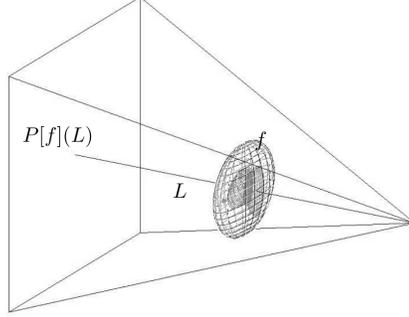


FIGURE 2. Principle of volume rendering: to display a volumetric function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$, we display a 2D projection $P[f]$ on a screen.

we want to display the surfaces, we need to select the significant values in f_{FDK} and to eliminate the others. This job is well performed by MIP: by definition, along each ray of projection, it selects and displays only the most intense voxel. Thus MIP¹ of f_{FDK} tries to visualize surfacic points of the original scene, with a significant value. Also, for the rays which do not intersect the original surfaces, we expect a low value. By the way the MIP is expected to be more than just a visualizing process: due to the selection of only one voxel per ray (and which is probably surfacic), visualizing the volume f_{FDK} by MIP implicitly reduces the noise and selects the surfaces with the best possible resolution.

We describe now how using the MIP as the visual field of a virtual observer. We consider an observer, whose optical center is located at $x_0 \in \mathbb{R}^3$; we consider that he looks at a point $x_1 \neq x_0$. We consider an orthonormal basis $(\omega_1, \omega_2, \omega_3)$ such that $\omega_1 = \frac{x_1 - x_0}{|x_1 - x_0|}$ and $\omega_3 = \omega_1 \wedge \omega_2$; we assume that the rays of the visual field are the $x_0 + \rho(\omega_1 + y_2\omega_2 + y_3\omega_3)$, $|y_2| < Y_2$, $|y_3| < Y_3$, $0 < \rho$. We assume that the part of the scene to be visualized is a domain Ω ; Ω is the whole volume by default. We can define the view V as the set of these parameters: $V = (x_0, x_1, \omega_2, Y_2, Y_3, \Omega)$. We simulate the visual field of the observer by computing, for $|y_2| < Y_2$, $|y_3| < Y_3$:

$$J_V(y_2, y_3) = \max\{f_{\text{FDK}}(x), x = x_0 + \rho(\omega_1 + y_2\omega_2 + y_3\omega_3) \in \Omega, \rho > 0\}. \quad (2.4)$$

To represent $y_2 \in [-Y_2, Y_2] \times [-Y_3, Y_3] \mapsto J_V(y_2, y_3)$, we set a RGB colormap $\Gamma : \text{Range}(J_V) \rightarrow [0, 255]^3$, and we display the image:

$$I_V = \Gamma[J_V]. \quad (2.5)$$

We would like to emphasize in the rendering the non-negative intense values of f_{FDK} , so we impose the following conditions: $\Gamma(t) = 0$ for $t < 0$ and $\|\Gamma(t)\|$ is a non-decreasing function. This colormap Γ can eventually improve the rendering by rescaling or thresholding. In this paper, we produce grayscale images by assuming that the three RGB channels are the same: $\Gamma(t) \in [0, 255]$, and to fix ideas, we make the following choice by default: once J_V is computed, we set

$$\Gamma(t) = \begin{cases} 0, & t \leq 0, \\ \frac{255}{T}t, & 0 \leq t \leq T, \text{ where } T = 0.5 \max J_V. \\ 255, & T \leq t, \end{cases} \quad (2.6)$$

With such a choice, $I_V = \Gamma[J_V]$ is a thresholded version of J_V , whose range is $[0, 255]$ due to the linear scaling.

Concerning implementation aspects, an image I_V is computed on a discrete grid of size $n_2 \times n_3$, from a discrete volume. One pixel corresponds to the projection along one ray. This may require

¹By choosing MIP of f_{FDK} , we implicitly decide to focus on the non-negative values of f_{FDK} . It could also be possible to focus on the most intense negative values, by considering the Minimum Intensity Projection, or to keep all the values by considering $|f_{\text{FDK}}|$ instead of f_{FDK} .

interpolation. When the size of f_{FDK} is N^3 and Ω is the full volume, the cost for I_V is $\mathcal{O}(n_2 n_3 N)$. To speed up the computations, it is possible to implement the MIP on GPU. For example, each thread can be the computation of one pixel of the image I_V .

In practice, a view V and a colormap Γ can be decided in advance; in that case, the image I_V is computed completely automatically. Then several views can be computed by changing the parameters: the optical center x_0 can be translated, the central point x_1 can be translated, the directions of the image ω_2, ω_3 can be rotated, the apertures Y_2 and Y_3 can be changed to zoom in/out, and the extracted sub-volume Ω can be changed. Adjusting the colormap Γ adjusts the rendering. Whereas changing the view V can be a way to obtain displacements in the scene, or videos, which helps to appreciate 3D structures.

2.5. Summary

Combining the heuristic of reconstruction and the principle of visualization, we get a computational process that can generate new views of a scene from a reflectogram: see Table 1. The reconstruction (step 1) is computed once for all; to change the view, we change V and we apply only the projection (step 2).

Input: reflectogram F , parameters of the view V
Step 1: compute the heuristic reconstruction f_{FDK} , by (2.1), (2.2), (2.3)
Step 2: display the image I_V , by projecting f_{FDK} for the view V , by (2.4) and (2.5)
Output: image I_V of the reconstructed scene, for the desired view V

TABLE 1. Global process for 3D reflective tomography: a new image I_V of the scene is displayed for the desired view V , from the reflectogram F .

We represent a reflective cone-beam scanning as $F = \Pi f_\Sigma$. For each angle θ , the scene is represented by a distribution $v \in \Sigma \mapsto f_\Sigma(v, \theta)$ supported by the surface Σ of the scene; Π is the reflective projection of these distributions. The heuristic solver computes the FDK reconstruction $f_{\text{FDK}} = \mathcal{R}^* \Phi F$ from the reflectogram F ; the operation Φ realizes (2.1) and (2.2), whereas \mathcal{R}^* is the weighted backprojection (2.3). For a desired view V , the visualization step computes the projection $J_V = \text{MIP}_V(f_{\text{FDK}})$ from (2.4), and the result is displayed with the colormap Γ : $I_V = \Gamma[J_V]$. In a word, the global heuristic generates the image I_V from the reflectogram Πf_Σ by:

$$I_V = \Gamma[\text{MIP}_V(\mathcal{R}^* \Phi(\Pi f_\Sigma))].$$

This is a full model for 3D reflective tomography. The mathematical challenge is now: prove that I_V faithfully represents the surfaces of the scene Σ , and quantify the *error* of the process. In the sequel we tackle this problem numerically.

3. Quality control

3.1. Quantitative quality criteria

The quality control of results for synthetic tests will be based on the visual perception and on quantitative criterion that we define here.

How to define such quantitative criteria? This question is more difficult than it appears. The bad idea is controlling the quality of the heuristic reconstruction f_{FDK} , by comparing it with f_Σ . Firstly this is not possible because f_{FDK} and f_Σ are different mathematical objects: f_Σ is a set of surfacic distributions which depend on the angle (the $f_i(\cdot, \theta)$) whereas f_{FDK} is a fixed volumic function. Secondly f_{FDK} is just a heuristic representation of the scene; so even on Σ , we can

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

imagine that f_{FDK} is very different from the $f_i(\cdot, \theta)$. This is a big difference with transmission tomography where f_{FDK} is close to a volumic function of attenuation. So we cannot directly define an error between f_{FDK} and f_{Σ} .

We propose instead to control the quality of the full process, by checking that an image $I = I_V$ is a right representation of the surface Σ : the main goal is to check that the final image is nice. Fundamentally we would like to check that the significant values of the image I come from voxels which are close to the surfaces of the original scene. If this is indeed the case, then it confirms two things: the most intense values of the heuristic reconstruction are located near the original surfaces, and the produced image represents an image of the surfaces. Otherwise the reconstruction contains high values, which are far from the surfaces and which introduce wrong alarms in the visualization: the final image contains wrong significant informations; they may escape visual perception, or not. We propose criteria which are based on counting the pixels in I that correspond to surfacic voxels.

For a pixel (y_2, y_3) of the image I , the displayed intensity is $I(y_2, y_3) = \Gamma f_{\text{FDK}}(x(y_2, y_3))$, where the voxel $x(y_2, y_3)$ is an arg max in (2.4). The distance between this voxel and the surfaces of the scene is $d(y_2, y_3) = \min\{|x(y_2, y_3) - z|, z \in \Sigma\}$. Assuming that a voxel is a cube of volume δ^3 , we claim that the voxel $x(y_2, y_3)$ coincides with a surfacic point when $d(y_2, y_3) < \delta$; in that case, we say that $x(y_2, y_3)$ is a *right voxel*, and that (y_2, y_3) is a *surfacic pixel*. (Eventually we could choose another threshold t for the test of being a right voxel: $d(y_2, y_3) < t$ instead of $d(y_2, y_3) < \delta$).

Let N be the number of surfacic pixels in I : $N = \sum_{(y_2, y_3)} \mathbb{1}_{d(y_2, y_3) < \delta}$; the proportion of surfacic pixels in I is $\mu = N/(n_2 n_3) \in [0, 1]$. N and μ depend on the geometry of the scene and on the selected view V . The higher μ is, the more the surfaces are spreaded in I . We can also compute the proportion of intensity explained by the surfacic pixels in I :

$$p = \frac{\sum_{(y_2, y_3)} I(y_2, y_3) \mathbb{1}_{d(y_2, y_3) < \delta}}{\sum_{(y_2, y_3)} I(y_2, y_3)} \in [0, 1].$$

We would like p to be large: the higher p is, the more the intensity in I comes from right voxels. The average of the (normalized) intensity among the surfacic pixels is $\frac{p}{\mu}$. The ratio $\kappa = \frac{p}{\mu}$ represents a normalized concentration of the intensity among the surfacic pixels. We would like this concentration κ to be large.

The analogous quantities can be defined for the non-surfacic pixels (pixels (y_2, y_3) such that $d(y_2, y_3) \geq \delta$): proportion $1 - \mu$ for the number, proportion $1 - p$ for the intensity, and $\bar{\kappa} = \frac{1-p}{1-\mu}$ for the concentration; we would like $\bar{\kappa}$ to be small. The ratio of the concentrations $\frac{\kappa}{\bar{\kappa}}$ gives information on how much the surfacic pixels are intense, comparing with the non-surfacic ones; we would like this ratio to be large.

3.2. Example

We illustrate the process on a full example.

3.2.1. Reflectogram

We simulate a reflectogram, from the famous Stanford Bunny [16]. We use the full resolution data set: 69451 faces. We use the lighting functionalities of Matlab. For that example, we color the faces of the object, using the following smooth pattern:

$$x \mapsto 1 + 0.5 \sin(20\pi |x|);$$

the pattern is computed at the vertices, and extended to the faces by interpolation. We consider a black background: intensity $f_0 = 0$. We use the Gouraud model of Matlab for the lighting

model. The object is rotated over 360 degrees (constant step); we take $|\Theta| = 1605$ images of size $N_2 \times N_3 = 397 \times 312$. We represent 6 images of this sequence in the Figure 3.

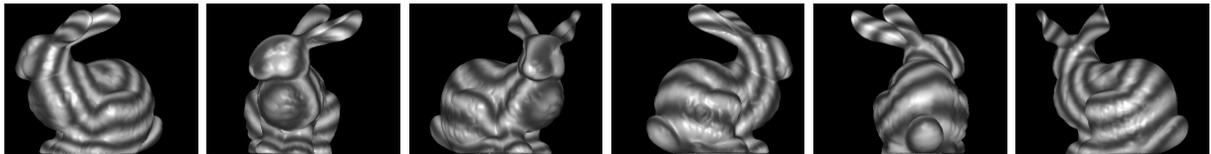


FIGURE 3. A few images of the sequence for the full example; step of 60 degrees.

3.2.2. Heuristic

We apply the process of reconstruction-visualization. For the filtering, we choose the Shepp-Logan filter. We compute two sequences of displacements around the reconstructed scene: for the first one, we turn around the vertical axis, for the second one, we turn around a horizontal direction: see Figure 4 for the resulting images. The visual perception of the results is nice. The renderings are contrasted. They represent the original scene under a semi-transparent form. The different surfaces are somehow mixed together; in particular, views from opposite sides look similar. A way of improving the visual perception is creating videos where the view changes continuously, and/or reducing the displayed volume Ω .

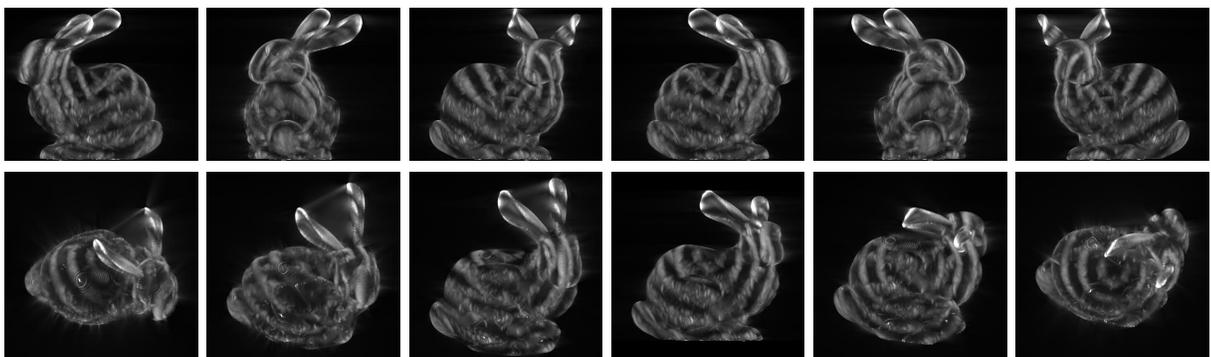


FIGURE 4. Two sequences of views, computed by the heuristic. Rotation by step of 60 degrees around the vertical axis (first line), and by step of 30 degrees around a horizontal axis (second line).

3.2.3. Quality control

We now check the quality of the rendering by controlling a set of surfacic pixels. We must select one view. The focal planes of the acquisition are vertical, so generating a view on a horizontal plane is one of the most difficult problems. That is the reason why we decide here to select an aerial view: we control the process of generating a semi-transparent aerial view from vertical reflective projections.

To check if a pixel (y_2, y_3) of the rendering is a surfacic pixel or not, we need to estimate the distance $d(y_2, y_3)$ between the voxel $x(y_2, y_3)$ and the surfaces of the scene. Here the scene is the bunny over a black wall. So $d(y_2, y_3)$ is the distance between the bunny and $x(y_2, y_3)$. First we compute a refinement of the bunny: for all face, we compute the vertices of a lattice such that each edge length of the lattice is at most the edge length δ of a voxel. At the end we get a finite set B of vertices. We consider that B is a discrete representation of the bunny for the resolution

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

δ . Then for all pixel (y_2, y_3) of the image I , we estimate the distance between the associated voxel $x(y_2, y_3)$ and the bunny as

$$d(y_2, y_3) = \min\{|x(y_2, y_3) - w|, w \in B\}.$$

After the computation of d , we decompose here I into two components; I is the superposition of: I restricted to the detected surfacic pixels $(y_2, y_3) \mapsto I(y_2, y_3)\mathbb{1}_{d(y_2, y_3) < \delta}$, and I restricted to the non-surfacic pixels $(y_2, y_3) \mapsto I(y_2, y_3)\mathbb{1}_{d(y_2, y_3) \geq \delta}$. In order to appreciate visually this decomposition we represent I and its components in the Figure 5. The component of the surfacic pixels is sharper than the total image I . But the contribution of the non-surfacic pixels helps for the perception of the object in I .



FIGURE 5. From the left to the right: image I computed by the heuristic, I after selection of the surfacic pixels, I after selection of the non-surfacic pixels.

For the quantitative criteria, we obtain the following values:

N	μ	p	κ	$\bar{\kappa}$	$\frac{\kappa}{\bar{\kappa}}$
23558	0.15	0.43	2.91	0.66	4.38

And so $\mu = 15\%$ of the pixels in the image I come from right voxels; they explain $p = 43\%$ of the total intensity of I . So the concentration of intensity for the surfacic pixels is quite strong: $\kappa = 2.91$. This is $\frac{\kappa}{\bar{\kappa}} = 4.38$ times the concentration for the non-surfacic pixels: $\bar{\kappa} = 0.66$.

3.3. Contribution of the jumps

To observe the contribution of the jumps, we create synthetic data sets whose images are piecewise constant. We increase the number of jumps from a data set to the next one.

We consider here a sphere with a dent. To create this object, we deform the sphere $|x| = 1$, in spherical coordinates (ψ, φ, ρ) , where $\psi \in [-\pi, \pi]$ is the azimuth, $\varphi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ is the elevation, and $\rho > 0$ is the radius. For all points of the sphere $(\psi, \varphi, \rho = 1)$, the point of the considered surface is $(\psi, \varphi, \rho := 1 + 0.75(r - 1)\mathbb{1}_{r < 1})$, with $0.08r := (\frac{\psi}{\pi} + 1/4)^2 + (\frac{2\varphi}{\pi} + 1/6)^2$. This object is computed from a discrete version of the sphere, discretized with 640^2 patches. For all integer m , we define on this surface the following piecewise constant pattern, in spherical coordinates: $(\psi, \varphi) \mapsto p_m(\psi)p_m(\varphi)$, with:

$$p_m(s) = 0.5 + 0.25\mathbb{1}_{(ms - \lfloor ms \rfloor) < 0.5}.$$

We project directly this pattern: for a point $v(\psi, \varphi, \rho)$ which is visible for the angle of projection θ , the information which is measured on the corresponding ray is $f(v, \theta) = p_m(\psi)p_m(\varphi)$. And we consider a black background. Such a reflectogram is simulated using plots of surfaces with Matlab. By rotating over 360 degrees, we simulate here data sets whose size is $|\Theta| \times N_2 \times N_3 = 801 \times 201 \times 201$.

Increasing m increases the number of jumps; we simulate data sets for several values of m : 0, 1, 2, 4, 8 and 16. On the first line of Figure 6, we represent one image of the reflectogram, for the successive values of m . Of course we distinguish two kinds of jumps in the images: jumps due to discontinuities in the pattern, and geometrical jumps due to the shape (interface object/background).

We apply the heuristic on these data sets. In order to improve the visual perception of the dent, we restrict the domain to a half-space Ω before visualization: our full volume being a set of voxels $(i, j, k) \in [1, 201]^3$, we keep only the $j \geq 91$. On the second line of Figure 6, we represent a vertical view of the reconstructions, associated with the view of the data sets. More jumps in the input reflectogram can improve the visual perception of the object, for both the input and the output. The heuristic computes much more than just a convex hull. A remarkable property here is that the dent is even more perceptible in the reconstruction than in the reflectogram. This is because the boundary $(\frac{\psi}{\pi} + 1/4)^2 + (\frac{2\varphi}{\pi} + 1/6)^2 = 0.08$ of the dent introduces geometrical jumps in many input images; the heuristic combines them, and so the boundary is emphasized in the reconstruction.

To compute the quantitative appreciations, we first compute the distances $d(y_2, y_3)$ from the voxels $x(y_2, y_3)$ to the vertices of the surface, discretized with 640^2 patches. The following table summarizes the results:

m	0	1	2	4	8	16
N	19844	20015	19676	18747	17440	18744
μ	0.49	0.50	0.49	0.46	0.43	0.46
p	0.68	0.69	0.69	0.70	0.70	0.71
κ	1.38	1.38	1.42	1.51	1.63	1.54
$\frac{\kappa}{\bar{\kappa}}$	2.17	2.22	2.34	2.72	3.13	2.87

It is difficult to observe significant rules from these values. We can anyway make the following comments. If we compare the first three cases with the last three cases: the proportion μ of surfacic pixels decays, their proportion p of intensity increases, and so do their concentration κ and the ratio of concentrations $\frac{\kappa}{\bar{\kappa}}$. In a word, adding jumps on the surface changes the visual perception of the scene, and it gives more weight to the surfacic pixels in the rendering.

3.4. Contribution of the smooth variations

Realistic images of realistic scenes have often jumps and smooth variations. Here we would like to observe the contribution of the smooth variations. So we create synthetic data sets whose images are smooth, and whose *frequency* increases from a data set to the next one.

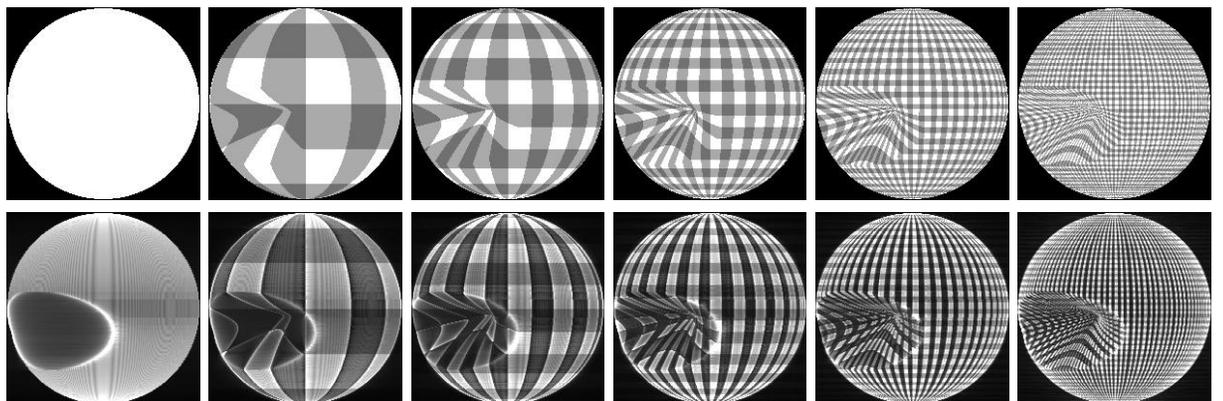


FIGURE 6. Effect of jumps for a sphere with a dent. From the left to the right: the projected pattern has more and more discontinuities, $m = 0, 2^i, 0 \leq i \leq 4$. On the first line: one image of the reflectogram; on the second line: associated heuristic reconstruction.

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

We consider the sphere $|x| = 1$, which is discretized using 640^2 patches as before. We consider a black background, and we project a smooth pattern, in spherical coordinates:

$$(\psi, \varphi) \mapsto 1 + 0.5 \cos m(\psi + \varphi),$$

with $m = 2^i, 1 \leq i \leq 6$, playing the role of a fixed frequency. By rotating over 360 degrees, we simulate here reflectograms whose size is $|\Theta| \times N_2 \times N_3 = 801 \times 201 \times 201$. The simulated images are smooth, except at the interface object/background where discontinuities occur. So we weight the data in order to obtain smooth images. For a pixel $(i, j) \in [1, 201] \times [1, 201]$, the weight is $(r + 1)^2(r - 1)^2 \mathbb{1}_{r < 1}$, with $99r(i, j) = ((i - 100)^2 + (j - 100)^2)^{0.5}$. On the first line of Figure 7, we represent one smooth image of the reflectograms that we get, for the successive values of m .

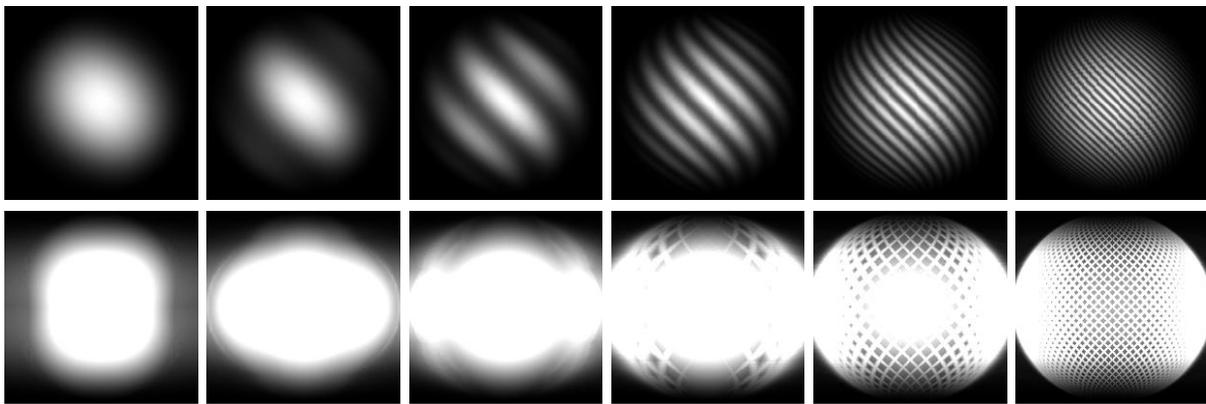


FIGURE 7. Effect of smooth variations for a sphere. From the left to the right: the frequency parameter is larger and larger: $m = 2^i, 1 \leq i \leq 6$. On the first line: one image of the reflectogram; on the second line: associated heuristic reconstruction.

We apply the heuristic on these data sets. On the second line of Figure 7, we represent a vertical view of the reconstructions, associated with the view of the data sets. And we summarize in the following Table the quantitative criteria that we get for these views:

m	2	4	8	16	32	64
N	1930	1761	1273	3881	10425	17341
μ	0.05	0.04	0.03	0.10	0.26	0.43
p	0.02	0.03	0.02	0.09	0.28	0.54
κ	0.40	0.59	0.73	0.95	1.10	1.26
$\frac{\kappa}{\bar{\kappa}}$	0.39	0.58	0.72	0.95	1.14	1.56

From the visual point of view, it is easier to identify the original sphere for large m . Increasing m also improves the result from a quantitative point of view; all the computed criteria tend indeed to increase: the proportion μ of surfacic pixels, their proportion p of intensity, their concentration κ , and the ratio of concentrations $\frac{\kappa}{\bar{\kappa}}$. For smooth data, the result of the heuristic may be poor for low frequencies components, but is meaningful when the frequency is large enough. In any case, the smooth variations of the input produce contrasts in the renderings; they can be useful for the visual perception.

3.5. Test of robustness

We show how the method deals with changes in the forward problem during the acquisition, by considering a randomized fluctuating model.

For all $\sigma = 0, 2^j, -2 \leq j \leq 2$, we consider a scan $F_\sigma(\theta, y_2, y_3)$ over 360 degrees of the Stanford Bunny; the size of this scan is $|\Theta| \times N_2 \times N_3 = 801 \times 200 \times 157$. For each angle θ , the projected surfacic pattern (with background $f_0 = 0$) is:

$$x \mapsto 1 + (0.2 + \sigma\eta_1(\theta)) \sin(\pi\sigma\eta_2(\theta) + 20\pi|x|),$$

where the $\eta_i(\theta)$ are independent realisations of the Gaussian $\mathcal{N}(0, 1)$. The $\sigma\eta_i(\theta)$ correspond to some modifications of the amplitude and the phase of the pattern. Increasing σ increases the dependency in θ for the projected pattern. Along a piece of curve $\{(\theta, y) : v \in L(\theta, y)\}$ where v is a visible point of the scene, the intensity level of v randomly varies with a standard deviation which increases when σ increases. To observe this, see the first line of Figure 8, where we represent the horizontal slice $y_3 = 0$ of the considered reflectograms.

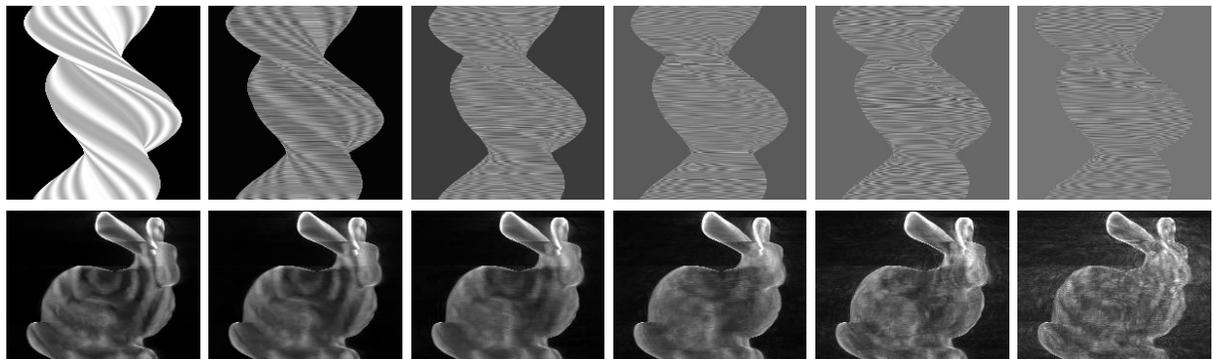


FIGURE 8. Test of robustness: from the left to the right, the level of disturbance is $\sigma = 0, 2^j, -2 \leq j \leq 2$. On the first line, slice in the reflectograms: $(\theta, y_2) \mapsto F_\sigma(\theta, y_2, 0)$. On the second line: a vertical view computed by the heuristic.

We apply the heuristic on these data sets. On the second line of Figure 8, we represent a vertical view of the reconstructions; we get the following quantitative appreciations:

σ	0.00	0.25	0.50	1.00	2.00	4.00
N	11805	11644	11410	11373	10745	8456
μ	0.38	0.37	0.36	0.36	0.34	0.27
p	0.79	0.77	0.74	0.69	0.61	0.42
κ	2.11	2.09	2.03	1.92	1.78	1.57
$\frac{\kappa}{\bar{\kappa}}$	6.36	5.82	4.94	4.00	2.99	2.00

Despite changes in the forward problem, some information stays coherent. This is for example the case of the shape of the object. The visual perceptions that we get here tend to show that the heuristic successfully combines such coherent informations, which is in agreement with the discussion of subsection 2.3. From a quantitative point of view, the computed criteria decay slowly when the level of disturbance increases. Increasing the disturbance does not seriously affect the set of right voxels in the final image, from the cardinal point of view, and from the intensity point of view too. It is also clear that the result stays relevant for large disturbances; for the last case, despite the level of disturbance is about 4 times the intensity level of the original signal, surfacic variations can still be perceived in the reconstruction.

3.6. Test of resolution

We observe the effect of increasing the resolution of the reflectogram on the reconstructed images.

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

The set-up is exactly as in the subsection 3.2, except that we consider several sizes of data sets. For an image whose horizontal length is $N_2 = \ell + 1$ pixels, we choose $|\Theta| = \lceil \ell\pi \rceil$ angles, over $[0, 2\pi)$. We choose such a sampling because it is compatible with sampling conditions of transmission tomography. On the first line of Figure 9 we represent one image of each reflectogram.

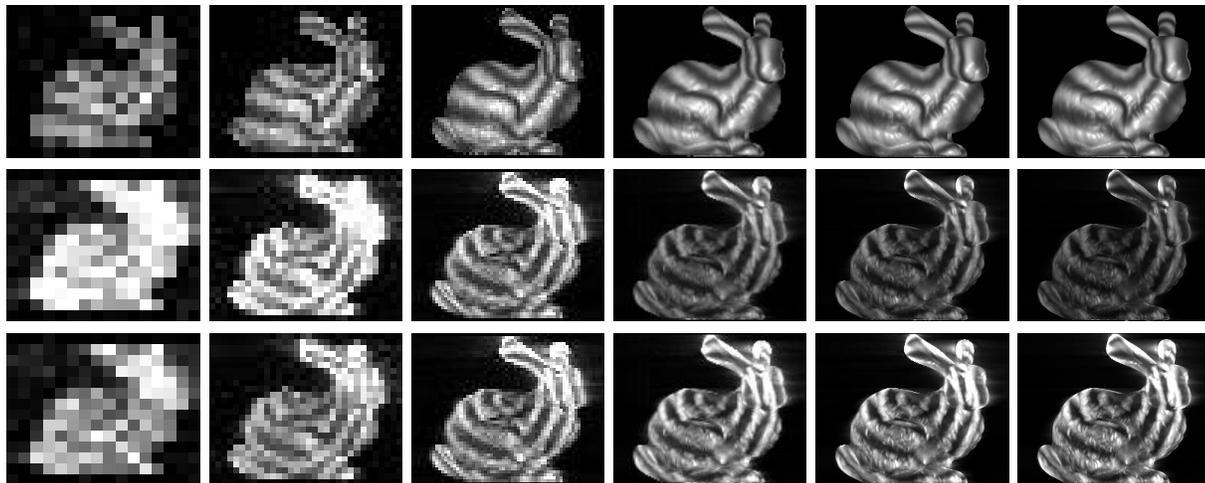


FIGURE 9. Test of resolution: from the left to the right, the length of an input image is: $N_2 = 16, 32, 64, 128, 256, 384$ (pixels). On the first line, one image of the reflectogram; on the other lines, associated reconstructions with different upper thresholds: $T = 0.5 \max(J_V)$ (second line) and $T_2 = \text{quantile}(\max(J_V, 0), 0.97)$ (third line).

On the second line we represent an associated heuristic reconstruction, with the colormap (2.6). We propose here a second choice of colormap: in (2.6), we replace the upper threshold by the quantile of $\max(J_V, 0)$ for the cumulative probability 97%: $T_2 = \text{quantile}(\max(J_V, 0), 0.97)$. We get a colormap which is based on the cumulative histogram; the result is less sensitive to the single value $\max(J_V, 0)$ of the histogram. This avoids the apparent loss of contrast when the resolution increases: see the third line of Figure 9. From the visual point of view, increasing the resolution of the dataset improves the sharpness of the reconstruction. The global shape is sharper and sharper; more and more details are perceptible to the eye: see for example the coat.

The values for quality control are (for the threshold T_2):

$ \Theta $	48	98	198	399	802	1204
N_2	16	32	64	128	256	384
N_3	14	26	52	102	202	302
N	113	360	1219	4308	15263	32141
μ	0.50	0.43	0.37	0.33	0.30	0.28
p	0.81	0.81	0.74	0.70	0.65	0.63
κ	1.61	1.87	2.02	2.13	2.21	2.28
$\frac{\kappa}{\bar{\kappa}}$	4.20	5.63	4.93	4.77	4.51	4.46

Unfortunately some parameters slowly decrease when the resolution increases: the proportion μ of surfacic pixels, their proportion p of intensity, and the ratio of the concentrations $\frac{\kappa}{\bar{\kappa}}$. From a more optimistic point of view, the number N of right pixels increases, and so does their concentration κ .

3.7. Test of stability

We realize a stability test by adding a speckle noise, for several orders of magnitude of noise.

We consider a reflectogram F as in the subsection 3.6, with a size of $|\Theta| \times N_2 \times N_3 = 801 \times 200 \times 157$; we apply the linear scaling such that the range of F becomes $[1, 2]$: $F := 1 + (F - \min F)/(\max F - \min F)$. For $\sigma = 0.05i, 0 \leq i \leq 30$, we consider a reflectogram F_σ with a speckle noise of magnitude σ : $F_\sigma = F(1 + \sigma\eta)$, where η contains $|\Theta| \times N_2 \times N_3$ (independent) realizations of the Gaussian $\mathcal{N}(0, 1)$.

In the Figure 10 we represent one image of the noisy data sets F_σ , for $\sigma = 0.3i, 0 \leq i \leq 5$, and we represent the associated reconstructions. We give in the following table the associated quantitative criteria. And in the Figure 11, we plot the criteria as functions of the level of noise, from the values $\sigma = 0.05i, 0 \leq i \leq 30$.

σ	0.00	0.30	0.60	0.90	1.20	1.50
N	8252	4033	2459	1838	1576	1400
μ	0.26	0.13	0.08	0.06	0.05	0.04
p	0.59	0.19	0.10	0.07	0.06	0.05
κ	2.23	1.45	1.24	1.17	1.13	1.08
$\frac{\kappa}{\bar{\kappa}}$	3.97	1.56	1.27	1.18	1.14	1.09

Except for the ratio of the concentrations which fastly decays at the very beginning, the criteria

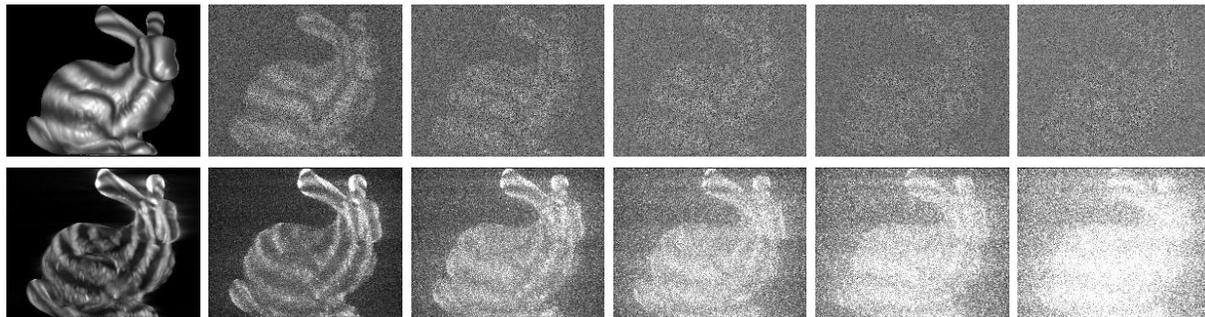


FIGURE 10. Test of stability. From the left to the right: the level of speckle noise increases: $\sigma = 0.3i, 0 \leq i \leq 5$. On the first line: one image of the reflectogram; on the second line: associated heuristic reconstruction.

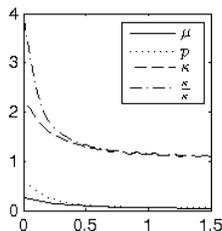


FIGURE 11. Test of stability: criteria as functions of the level of noise σ .

slowly decay when the level of noise increases. The visual perception of the reconstructed scene is stable. Accumulation of coherent information over the whole data set can tolerate noise.

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

3.8. Transverse reading

Further observations can be made by reading transversely the previous results.

The method works for several kinds of physical models. In the first example of subsection 3.3, the model is binary; it illustrates the reconstruction of a non-convex object from its shapes. And all of the examples of this subsection are based on projecting a fixed distribution. Then for the subsections 3.2, 3.6 and 3.7, it is a Gouraud model. In particular, these results include cases of isotropic diffusion without specular components, and cases with both diffusion and specular reflections. And we have even results for fluctuating models in 3.5.

Concerning the rendering, we have semi-transparent representations of full reconstructed volumes, and we illustrate in 3.3 the improvement produced by selecting a sub-domain, which is also a simple way of observing a region of interest. We have by the way proposed renderings based on two thresholding rules. This illustrates the influence of thresholding, and more generally the adjustment of the rendering by changing the colormap.

3.9. 3D optronic imaging

We finish by an example of 3D reflective tomography on experimental data. We consider a real laser system which provides active laser images of backscattered intensity by rough surfaces: see [15] for technical details of such a device. We consider a set of real optronic images, courtesy of Thales Optronique SA, obtained with this system. A sequence of 360 images of size 342×181 was measured by turning around the scene, one degree step. The considered scene is a vehicle: see Figure 12 for samples of the sequence.



FIGURE 12. A few images of a sequence of real optronic images; step of 60 degrees.

We apply the heuristic, using a home-made software, implemented in CUDA C. The execution on a Nvidia Tesla C2075 takes 2.6 seconds for the reconstruction, and is real-time for the display. In the Figure 13, we represent snapshots that we got interactively. For the first three views, the rendering uses extracted sub-volumes. The next images use the whole volume, with a view clearly different from the recorded views. In particular the last view is taken inside the vehicle. The reconstructed scene contains features and details that are useful for recognition. A notion of relief can be perceived in the rendering. By the way the scene was not completely opaque: transmission across glasses. The dashboard, which is behind the glasses, is reconstructed. The heuristic and its interpretation of accumulation of coherent contrasts still apply for data which are not purely reflective; this extends its domain of validity.

4. Conclusion

This paper deals with an emerging method in optical imaging. First we introduced a new mathematical model describing a full process of 3D reflective tomography; we recapitulated the algorithms. Then we controlled numerically the quality of the method by testing the influence of several parameters. We introduced new quality criteria, usable for synthetic tests.

Concerning the applications, the full process was checked by computing quality criteria. Based on the numerical results, the conclusion is that reflective tomography renders relevant images for many configurations, both visually and quantitatively.

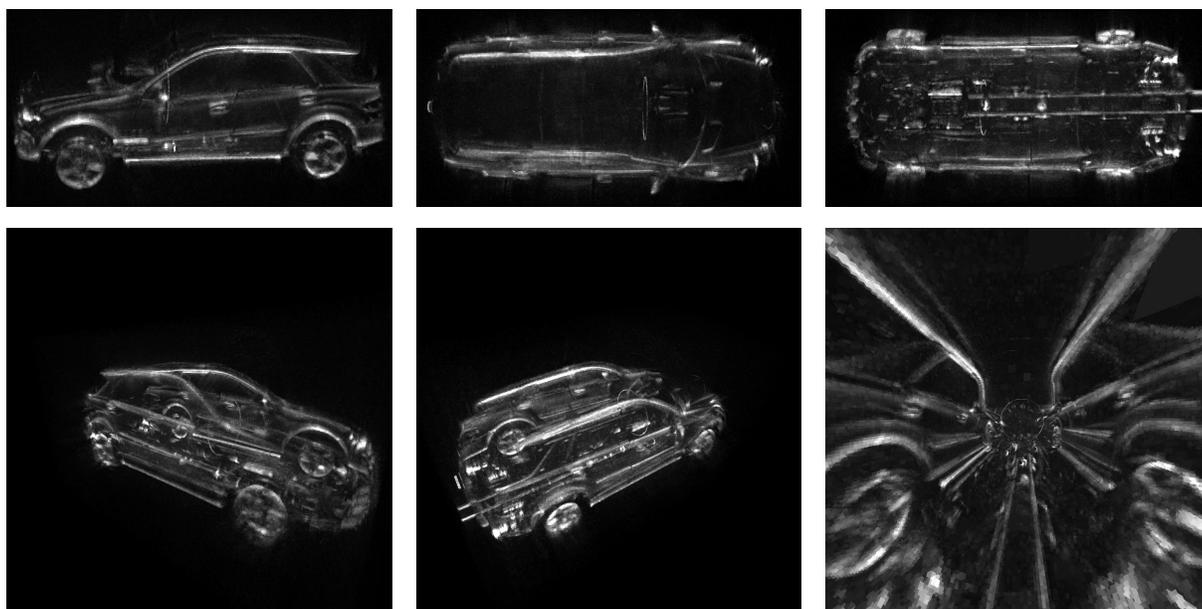


FIGURE 13. Example of computed views, from the real data set.

This paper suggests further mathematical works: proving mathematically in what extent 3D reflective tomography is relevant is still an open question. We hope that our mathematical model could serve as a starting point.

Bibliography

- [1] J.-B. Bellet. Analyse asymptotique et géométrique de la tomographie réfective. *Soumis*, 2017. <hal-01571707>.
- [2] J.-B. Bellet, I. Berechet, S. Berechet, G. Berginc, and G. Rigaud. Laser Interactive 3D Computer Graphics. *2nd International Conference on Tomography of Materials and Structures, Québec*, 2015. <hal-01175855>.
- [3] J.-B. Bellet and G. Berginc. Reflective filtered backprojection. *Comptes rendus - Mathématique*, 354:960–964, 2016.
- [4] I. Berechet, G. Berginc, and S. Berechet. Method for 3D reconstruction of an object in a scene. United States Patent, No US 2013/0100131 A1, April 2013.
- [5] S. Berechet, I. Berechet, J.-B. Bellet, and G. Berginc. Procédé de discrimination et d'identification par imagerie 3D d'objets d'une scène. Patent WO2016097168 A1, 2015.
- [6] G. Berginc. Scattering models for 1-D–2-D–3-D laser imagery. *Optical Engineering*, 56, 2016.
- [7] G. Berginc, J.-B. Bellet, I. Berechet, and S. Berechet. Optical 3D imaging and visualization of concealed objects. In *Proc. SPIE*, volume 9961, page 99610Q, 2016.
- [8] G. Berginc and M. Jouffroy. Optronic system and method dedicated to identification for formulating three-dimensional images. US patent 20110254924 A1, European patent 2333481 A1, FR 09 05720 B1, November 2009.

QUALITY CONTROL OF 3D REFLECTIVE TOMOGRAPHY

- [9] G. Berginc and M. Jouffroy. Simulation of 3D laser systems. In *Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009*, volume 2, pages 440–444. IEEE, 2009.
- [10] G. Berginc and M. Jouffroy. Simulation of 3D laser imaging. *PIERS Online*, 6(5):415–419, 2010.
- [11] G. Berginc and M. Jouffroy. 3D laser imaging. *PIERS Online*, 7(5):411–415, 2011.
- [12] L.A. Feldkamp, L.C. Davis, and J.W. Kress. Practical cone-beam algorithm. *JOSA A*, 1(6):612–619, 1984.
- [13] F.K. Knight, S.R. Kulkarni, R.M. Marino, and J.K. Parker. Tomographic Techniques Applied to Laser Radar Reflective Measurements. *Lincoln Laboratory Journal*, 2(2):143–160, 1989.
- [14] F. Natterer and F. Wübbeling. *Mathematical methods in image reconstruction*. SIAM, 2001.
- [15] G. Rigaud, J.-B. Bellet, G. Berginc, I. Berechet, and S. Berechet. Reflective Imaging Solved by the Radon Transform. *IEEE Geoscience and Remote Sensing Letters*, 13:936–938, 2016.
- [16] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 311–318. ACM, 1994.