Simultaneous Absorption and Environment Light Reconstruction in Optical Tomography Problem

Afanasyev V.
Keldysh Institite of Applied Math RAS
4, Miusskaya sq.
Russia, 125047, Moscow
vafanasjev@graphics.cs.msu.ru

Ignatenko A.
CMC MSU
Leninskie Gory, MSU
Russia, 119991, Moscow
ignatenko@graphics.cs.msu.ru

Voloboy A.
Keldysh Institite of Applied Math RAS
4, Miusskaya sq.
Russia, 125047, Moscow
voloboy@gin.keldysh.ru

ABSTRACT

Classic tomography algorithms applied in optical tomography require the light source pre-calibration and do not allow refining the light map in tomography algorithm. This article shows an approach to environment light reconstruction during the ART algorithm execution. It makes the optical tomography scanning process more fast and simple, allowing to exclude the light calibration stage.

Keywords

ART, tomography, lighting reconstruction

1. INTRODUCTION

Tomography concept

Computed tomography is a class of problems of object internal structure reconstruction using a set of its projections. The closest application area of computed tomography methods is X-ray tomography.

The X-ray tomography device consists of emitter, detector and a place for observable object between them. Emitter irradiates the X-rays with fixed intensity, they are absorbed inside the object and the detector registers residual ray intensity. During the object rotation on some trajectory a set of projections is created and using them the internal object structure is reconstructed.

Different configurations of ray beam are possible: it can be flat or volumetric, parallel or cone. Depending on this, flat or volumetric tomographic reconstruction is used.

The radiation absorption inside the material obeys Beer's law:

$$I = I_0 e^{-\int_a^b k(x)dx} \quad (1)$$

where I is residual intensity received by detector, I_0 is initial emitter intensity irradiated in this direction, $k(x)$ is distribution of absorption index along the ray.

If we take a set of parallel lines perpendicular to flat detector and integrate the absorption index along every line, and consider all directions of these lines, we get the Radon transform for examined volume as it is shown on fig. 1.

$$R(P,n) = \int_{-\infty}^{\infty} f(P + \vec{n}t)dt \qquad (2)$$

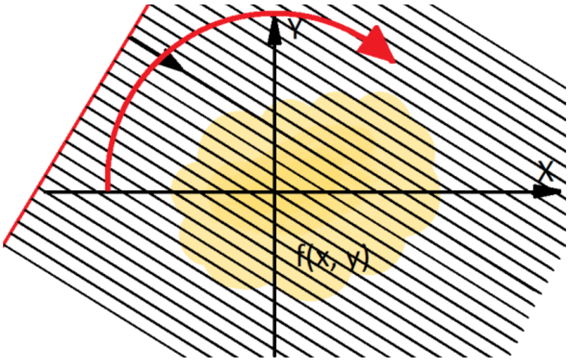

Figure 1. Radon transform

If we put some restrictions on function to reconstruct, the inverse Radon transform exists **[Hel99a]**. But if there is a noise or some other inaccuracy in initial data, it is unstable. Also counting reverse Radon transform is computationally inefficient.

There is a number of approximate tomography algorithms, for example, algorithm based on inverse Fourier transform, convolution and back projection, algebraic reconstruction techniques (ART). ART group of methods are the most flexible, so one of them is used in this study.

ART is based on sequential correction of resulting function stored in voxel map using its projections one-by-one. Every projection value which is an integral of initial function along the corresponding ray affects the resulting function along the same ray according to some law. That makes observable integral along the ray closer to the real measured value of this integral every time. Then, the iterative process stops on some trigger.

Optical tomography

Optical tomography uses the same principle as X-ray tomography, but there are light rays instead of X-rays. This makes significant difference and adds some physical effects and difficulties in handling them.

- Light is reflected and refracted when it meets transparent object
- Complete absorption of light is a usual case
- Refraction changes the light direction, and usually for some particular area inside transparent object only part of all directions are available for observation
- Defocusing on convex refracting surfaces
- Rays that were parallel in the air, after refraction cross the target volume from multiple directions

Since here, we will consider only objects with flat surfaces to avoid defocusing problem which is not the point of this study. Also in order to simplify the equations, working with single wavelength will be considered because processing polychromatic spectrum data does not refer directly to the subject of this article.

The optical tomography scanner (fig. 2) consists of areal diffuse light source, camera and axis to place and rotate the observable object. The installation is covered with opaque housing to prevent object illumination from outside.

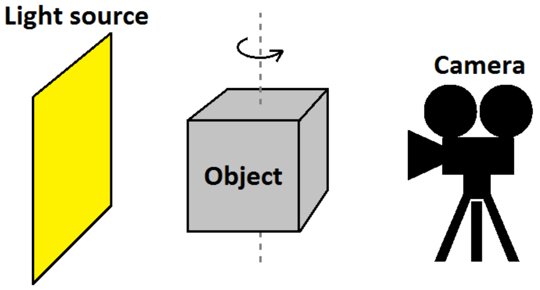

Figure 2. Optical tomography scanner

In this article the approximate ART (Algebraic Reconstruction Technique) algorithm will be used for volumetric absorption reconstruction. Unlike classic ART **[Gor70a]** this algorithm uses approximate correction of absorption along the ray inside target volume. It is based on difference between observable pixel brightness and its expected value, which was calculated using current absorption distribution inside the target volume, without direct calculation of observable integral absorption value.

2. ENVIRONMENT LIGHT PROBLEM. EXISTING APPROACHES

Tomographic reconstruction requires exact knowing of integral absorption along all rays crossing the volume of interest. In ideal case, light should leave the light source, then, after two refractions and absorption inside the object it should be captured by camera. Also light intensity distribution on the light source should be known. Then using Beer's law and Fresnel equations the residual intensity can be evaluated:

$$I = I_0 T^2 e^{-K} \qquad (3)$$

Where I is light source intensity in the ray hit point, I_0 is pixel value measured with camera, T is Fresnel transmittance coefficient for particular angle of incidence, $K = -\int_a^b k(x)dx$ is integral absorption along the ray, this value will be used in tomography.

$$K = -\ln \frac{I}{I_0 T^2} \qquad (4)$$

The light source can be more complex than it is shown on the scheme. For example, it can consist of primary light source like LED and reflective diffusor. The scanner has several internal parts and though the installation is closed from external light, camera can receive some light that was reflected multiple times inside the scanner. Observable object itself can reflect light back to diffusor that increases its brightness. Finally, this excess light is reflected from observable object to camera and it contributes into the pixel value. So, in order to find a precise value K of absorption inside the object, all illumination around the object and its reflections should be considered.

Simple light acquisition

The simplest method to get some approximation of real light is to make a photo of background behind observable object, when this object is removed. Generally, it's impossible to make one photo for the scanner and to use it for every scanned object. Scanning every new object may require its own light settings that can be tuned only after taking some photos of observable object. So, there are 2 ways to carry out scanning and tomography. Note that tomography requires having data about lighting before it started.

1. Separate sequential scanning and tomography:
   - Placing object to scanner
   - Tuning light using sample photos of object
   - Scanning object (taking photos of it, rotating 360° around vertical axis)
   - Removing object
   - Taking photo of background

- Carrying out tomography

2. Simultaneous scanning and tomography:
   - Placing object to scanner
   - Tuning light using sample photos of object
   - Removing object
   - Taking photo of background
   - Placing object again
   - Scanning and tomography (every taken photo is immediately applied in algorithm, scanner and computer work simultaneously)

As we can see, in first case there is no parallelism, in second case taking photo of background requires extra placing and removing object in case of tuning light. Every variant takes some additional time or human actions.

Another problem is the size of light source. Normally, light source is bigger than camera frame. Otherwise, if the light source is smaller or of the same size, there will be dark areas on photos: on the object or around it and this will decrease the amount of information for tomography. Taking photo of the background gives us only part of light source, though, most part of observable light comes from this area.

Simple calibration

Possible solution to the problem is light source calibration using parametric model **[Afa14a]**. Light distribution on the light source inside and outside the frame is calculated using a set of parameters. Then, these parameters are optimized by comparing rendered scene with real photos. The geometric model of light source, some other data like LED radiation pattern can help to reduce the number of parameters. This approach can be used without removing object from scanner, the only requirement is to use frame(s) with some piece of background area or knowing average absorption of the object.

A disadvantage of parametric light model is inexact matching of computed light intensity and real background in visible light source area. Solution to this problem, which is used in practice, is combining last two methods: we use real photo in visible area and tuned parametric model in invisible area. It takes additional time, but gives better result than any of two “pure” methods.

If for some reason the parametric model is too complex to be tuned fine, or side reflections (fig. 3) have significant brightness, the environment light reconstruction can help to make more accurate light model.

3. SEPARATE LIGHT RECONSTRUCTION

Existing approaches

There are some studies that solved the problem of environment light reconstruction using artificial objects like reflective balls **[Hey05a]** or more generally, using reflections from any scene objects **[Gib01a]**. Usually more general algorithms represent light as panorama or a set of point light sources. They use an iterative technique to reconstruct lighting. A similar technique is used in the following algorithm which also takes light transmission into account, and it is a base for more complex algorithm discussed later.

Implemented algorithm

An algorithm using both transmitted and reflected light to reconstruct light panorama was implemented.

The observable object is required to be reflective and can be also transparent. The environment can be defined as 3D scene model or as simple spherical panorama. In the tests of this study a sphere with finite radius was used, because the available real scanner has orthographic camera and does not have any precise draft of internal geometry. For example, light can be positioned manually.

The algorithm takes the following data:

- A set of greyscale photos of observable object
- Camera calibration for all photos
- 3D model of environment (not necessary)

The algorithm output is a panorama light map which can be applied to initial scene as a texture.

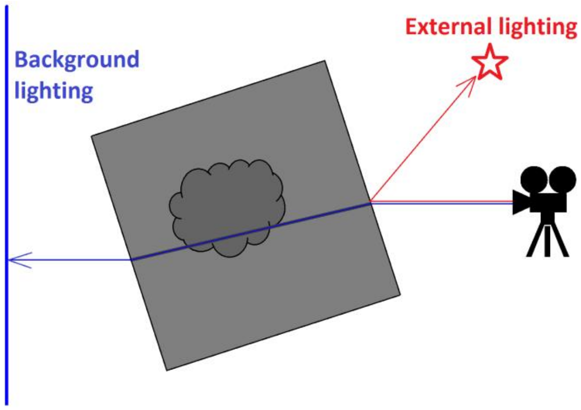

Figure 3. Rays path from camera to environment

It works iteratively, using a sequence of given images in some defined order. For example, it can be a trivial order with continuous camera movement. The initial panorama is black. For each frame there are the following steps:

1. **Rendering step.** A ray is traced from camera into scene as it usually is done in ray-tracing. It hits some surfaces, reflects, transmits, splits into

multiple rays and finally every of these rays cross the environment in some point corresponding to a panorama point. Radiance from all rays is gathered (summed, multiplied by Fresnel coefficient and transmittance coefficient in case of absorption) and assigned to current pixel brightness I.

2. **Counting correction.** Existing brightness value in pixel is compared with observable brightness I_0 taken from photo. A difference $\Delta I = I - I_0$ is counted. Then, a correction value C is calculated based on difference. For example, $C = t\Delta I$, where $0 < t \leq 1$.
3. **Correction step.** The second time a ray is traced, having the same route. Each ray has its own correction value and there are the following rules for correction propagation:
    - If the ray hits environment, its correction is added to panorama value in this point
    - After hitting a surface of transparent object the ray splits into 2 rays. Corrections of new rays are proportional to their impact (to Fresnel coefficients) and sum of them is correction of initial ray.
    - If total internal reflection occurs, the ray correction is preserved.

It is important that during correction step rays repeat the trajectory of the rays on rendering step and hit the same panorama points. It can be achieved by just saving hit points and ray impacts on the first step without tracing one more time, if architecture allows this. Also hitting one panorama pixel by 2 rays on one frame should be considered: double correction will lead to wrong result.

Panorama light reconstruction on scanner with areal background light showed interesting results. The object to reconstruct can be an immersion glass cube with a gemstone inside it (typical object for tomography reconstruction: in this case the inclusion models inside the stone are the main target). On the figure 4 you can see an example of analogic object: a cube of epoxide glue with some wires inside it.

Figure 4. Object example

On the first iterations algorithm makes some phantom light source behind the camera and the real light source behind. That should be expected, because the route from camera to this panorama area contains only one reflection from front cube face.

The figures 5 and 6 show sphere maps of light intensity. Camera looks to the point in center of real light labeled on figure 5. If we move half an image leftwards, we will get into the point behind camera. Top and bottom of picture are sphere poles.

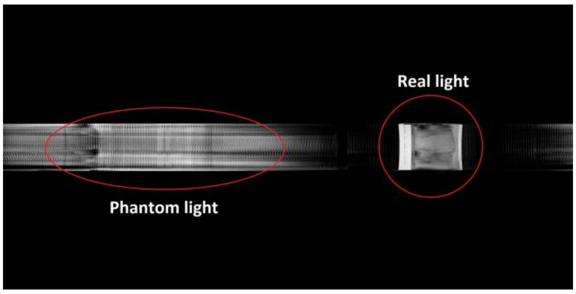

Figure 5. First iterations of light reconstruction.

But then the process converges to correct result, eliminating brightness on the place of phantom light.

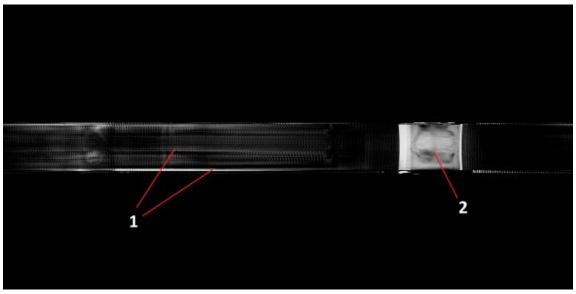

Figure 6. Final result of reconstruction

The final map is not ideal, because not all real effects were considered in the model. We can see different types of artefacts:

1. Some residual light on the place of phantom light source behind the camera. It remains because the used model of cube has inaccuracies and its edges produce big difference between photos and renders every time. Bright stripe behind camera has the same explanation.
2. A grey image on the light source that resembles original object: the semi-transparent cube and some dirt inside it. The absorption wasn't included into the model specially and some effect of really present absorption should have appeared: proportional decreasing of light brightness behind the cube is such effect.

It should be mentioned that the process has no convergence in a strict sense due to model inconsistency that is result of its incompleteness. The light map cannot represent absorption effect correctly, so, after a number of iterations light map will just make small but continuous oscillations.

Comparison of the light map reconstruction algorithm to parametric model **[Afa14a]** shows that the new algorithm gives better result in visible areas of light source (expectedly). It has almost the same result as the photo in visible light areas (difference is less than 1 grade of 255). But in the areas behind the object parametric model are still much better due to absorption map inside the object not being taken into account.

The advantage is that the new algorithm reconstructed invisible parts of diffusor using reflections from side cube faces. The data that was previously only extrapolation with parametric model, now is image-based. The new algorithm generally does not need the model of light source, its projection to sphere is reconstructed automatically.

4. SIMULTANEOUS LIGHT AND ABSORPTION RECONSTRUCTION

The next step is combining light reconstruction and tomography in a single process.

Algorithm on the frame and pixel level is the same as described in light reconstruction. The difference is addition of absorption impact. It should be taken in account in rendering step and in correction step.

Rendering step is obvious: brightness is calculated and summed as usual, but also the integral absorption inside target volume is counted and final brightness of the ray is multiplied by this value.

Correction step requires separation of absorption and brightness correction. For the ray which crossed target volume and leaves the medium, we should choose, how to share correction between these two opposite effects: absorption that decreases intensity of light brought with the ray and brightness of panorama pixels situated somewhere farther along this ray. The problem and the main difference from separation between transmitted and reflected rays is multiplication of brightness and transmittance coefficient, as we can see in equation 1.

$$I = I_0 A \qquad (5)$$

If we want to make some correction ΔI we cannot just separate it like the following:

$$(I + \Delta I) = (I_0 + \Delta I_0)(A + \Delta A) \qquad (6)$$

This problem is not solved yet, correction separation here is regulated with manually set coefficients for now and is inaccurate. The source of this issue is differential nature of correction. Carrying the desired value of brightness with all rays seems solve it, because in correction formula (6) we get rid of multiplication two sums. But this replacement will cause problems with storing brightness corrections in panorama.

5. IMPLEMENTATION AND RESULTS

Figure 7 shows an example of one horizontal layer of absorption map built from photos of glass cube. Bright areas represent dirt inside the glass.

Figure 7. Absorption voxel map slice

Figure 8 shows rendering of reconstructed absorption map in reconstructed lighting compared to real photo.

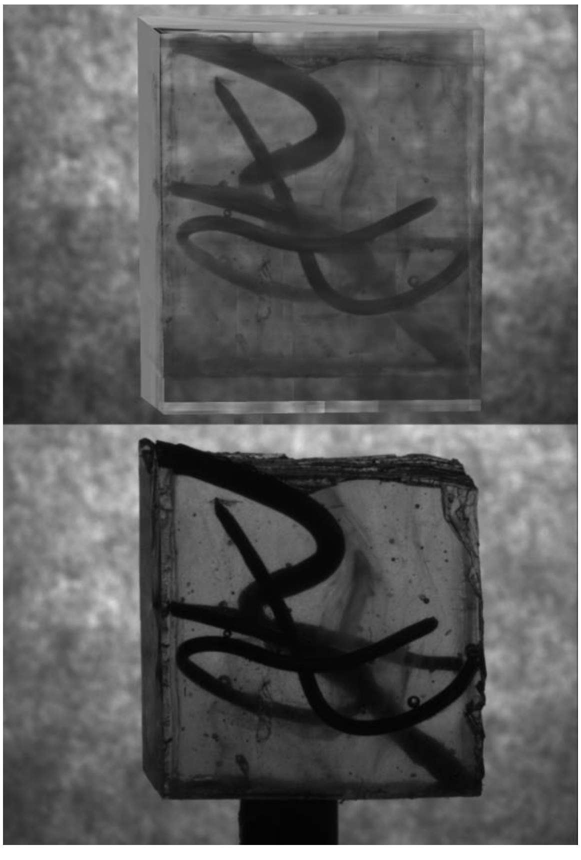

Figure 8. Rendering (top) and photo (bottom)

The algorithm is implemented on the base of ray tracing algorithm. Tomography procedure is built in ray tracing mechanism. NVidia OptiX GPU ray tracing engine was used **[Par10a]**.

Absorption index is stored as voxel map of float32 values. Its size is 512^3 voxels. Lighting panorama has resolution 4096×2048 and float32 type is used also. These arrays are stored in GPU memory.

The algorithm reaches acceptable absorption map quality after 1000 iterations (the same as algorithm using pre-calibrated light). The algorithm execution takes less than 10 minutes on GTX 980 for a cube occupying nearly all the visible area. The test sample with wires inside epoxide glue took 2.5 minutes on GTX Titan X. However, the light map quality is not enough after this due to the effect of phantom light described before, and about 10000 iterations are necessary to get acceptable lighting map.

6. CONCLUSION AND DISCUSSION

So, the algorithm was created which allows to build an absorption map without having previously calibrated light. It allows to exclude the physical manipulations with the scanner and observable object before scanning.

This study contains several unresolved problems: phantom light remaining in panorama, inexact separation of correction between brightness and absorption, slow convergence of light reconstruction. Though these issues do not disturb fast building of absorption map, which was the main purpose, solving them will help to make algorithm more accurate and fast.

Also this study considers only transparent materials. Although, real objects may contain some other effects that influence light travelling inside the material. Scattering, more complex absorption, reflections from internal structures among them. These effects can be taken into account by changing the physical model and light transport model: for example, a modified Beer’s law can be used. The further research will be directed to reconstruction of areas which have complex interaction with light, like cracks inside gemstones. Currently such objects are not reconstructed with acceptable quality.

7. REFERENCES

[Hel99a] Helgason S. Radon Transform Second Edition. Cambridge, pp.15-20, 1999.

[Gor70a] Gordon R., Bender R., Herman GT. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography. Journal of Theoretical Biology.29. No.3, pp. 471–81. 1970.

[Afa14a] Afanasiev V., Ignatenko A., Tisevich I. Secondary light source calibration using photos. Information techno;ogies in design and production. No. 2, pp. 41-45. 2014.

[Hey05a] Heymann, Sebastian, et al. Illumination reconstruction from real-time video for interactive augmented reality. Proc. of International Workshop on Image Analysis for Multimedia Interactive Services. 2005.

[Gib01a] Gibson S., Howard T.J., Hubbold R.J. Flexible Image-Based Photometric Reconstruction using Virtual Light Sources. Eurographics 20, no.3, Manchester. 2001.

[Par10a] Parker S., Bigler J., Dietrich A. OptiX: a general purpose ray tracing engine. ACM SIGGRAPH 2010 papers, Article No. 66. 2010.