e-Journal of Nondestructive Testing - ISSN 1435-4934 - www.ndt.net
10th International Symposium on Digital Industrial Radiography and Computed Tomography DIR 2025 – July 1st to 3rd, 2025, Paris (France).

Exact reconstruction for helical cone-beam X-ray CT using the ASTRA toolbox

Pavel PARAMONOV[1,2*], Joaquim SANCTORUM[1,2], Femke DANCKAERS[1,2], Jan De BEENHOUWER[1,2], and Jan SIJBERS[1,2]

[1] imec-Vision Lab, University of Antwerp, Antwerp, 2610, Belgium
[2] DynXlab: Center for 4D Quantitative X-ray Imaging and Analysis, University of Antwerp, Antwerp, 2610, Belgium

*Corresponding author, E-mail: pavel.paramonov@uantwerpen.be

Abstract
The ASTRA toolbox is a popular open-source software package for X-ray CT (XCT), supporting CT reconstruction for a wide variety of scan types. However, helical CT, which is extensively applied in medical and industrial imaging, has so far been available only through iterative reconstruction algorithms. This paper introduces an open-source Python add-on to ASTRA, implementing the state-of-the-art algorithm for helical CT - Katsevich's algorithm. Belonging to the filtered backprojection (FBP) family, it offers exact reconstruction with high computational efficiency. Our implementation leverages the GPU acceleration through ASTRA's algorithms and dedicated CUDA kernels to provide efficient backprojection, significantly reducing computation time compared to iterative methods. The developed Python package is built on top of ASTRA and includes all routines needed for helical CT: definition of scanning geometry, projection data filtering, and backprojection. To evaluate the accuracy and performance of the proposed implementation, we simulated projections of a CAD model resembling an 18650 battery. The XCT image reconstructed with Katsevich's algorithm is compared to the one reconstructed with SIRT. Our findings show that our implementation achieves high-quality image reconstruction with substantially faster processing times and reduced memory requirements.

Keywords: helical X-ray tomography, Katsevich algorithm, filtered backprojection, cone beam, ASTRA toolbox

Introduction

Helical computed tomography (CT) offers distinct advantages over conventional cone-beam CT techniques. It does not produce cone beam artifacts in the peripheral regions of reconstructed images and enables high-magnification imaging, which in turn allows for the reconstruction of fine details while capturing the entire sample within the X-ray field of view. The Katsevich algorithm is a state-of-the-art method for exact helical CT image reconstruction [1]. As a filtered backprojection (FBP) algorithm, it is computationally efficient, but its backprojection step poses implementation challenges. The ASTRA toolbox [2], a widely used open-source CT software, addresses these challenges by offering a flexible approach for defining source and detector trajectories using position vectors.
This work introduces an open-source Python implementation of the Katsevich algorithm [3], leveraging ASTRA's GPU-accelerated backprojection capabilities, along with additional GPU computations using CUDA and CuPy, and filtering steps implemented with NumPy. While ASTRA previously supported helical CT through iterative algorithms, the new implementation offers several advantages. First, it is significantly faster, requiring only a single backprojection run. Second, FBP demands less memory. Finally, it eliminates the need to tune parameters such as iteration count or voxel value constraints, simplifying the reconstruction process.

Methods

Our implementation of Katsevich's algorithm is based on [1]. The scanner source and flat detector move along helical trajectories, yielding a set of equiangular projections $g_f(\lambda, u, w)$, with λ the projection angle, u the pixel coordinate along detector columns, and w the coordinate along the rows. The algorithm consists of two parts: projection data filtering and backprojection (BP). Once the filtered projection data $g_f^F(\lambda, u, w)$ has been computed, the BP is carried out as follows:

$$f(\mathbf{r}) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{\chi(u,\ w) g_f^F(\lambda, u, w)}{v^*(\lambda, \mathbf{r})} d\lambda \bigg|_{u=u^*(\lambda,\mathbf{r}), w=w^*(\lambda,\mathbf{r})}, \quad (1)$$

with $\mathbf{r} = [x, y, z]$ the voxel coordinates, $\chi(u,\ w)$ the characteristic function of the Tam-Danielsson window, and

$$v^*(\lambda, \mathbf{r}) = R_0 - x\,cos(\lambda + \lambda_0) - y\,sin(\lambda + \lambda_0)\,, \quad (2)$$

with R_0 the source-object distance (SOD), and λ_0 the initial angle. The *BP3D_CUDA* algorithm available in ASTRA performs BP of helical data $g_f^F(\lambda, u, w)$ on the GPU, eliminating the need to explicitly calculate detector coordinates $u^*(\lambda, \mathbf{r})$ and $w^*(\lambda, \mathbf{r})$. Since it does not account for scaling with the position- and angle-dependent weight $v^*(\lambda, \mathbf{r})$, Eq.

(1) is implemented as follows: the reconstruction starts with an initial XCT image set to zero, $f_0(\mathbf{r}) \equiv 0$. For each angle λ_n, the corresponding projection is backprojected using *BP3D_CUDA*, resulting in the array $\tilde{f}_n(\mathbf{r})$. Subsequently, the values of $v^*(\lambda_n, \mathbf{r})$ are computed, and the XCT image is updated:

$$f_n(\mathbf{r}) = f_{n-1}(\mathbf{r}) + \frac{\tilde{f}_n(\mathbf{r})}{v^*(\lambda_n, \mathbf{r})}. \tag{3}$$

After backprojecting and scaling all N projections, the desired XCT image $f(\mathbf{r})$ is computed as follows:

$$f(\boldsymbol{r}) = \frac{1}{2\pi} f_N(\mathbf{r}) \Delta\lambda\,, \tag{4}$$

with $\Delta\lambda$ the angular step. These computations are performed in a dedicated CUDA kernel which combines scaling and integration.

Experiments

Experiments were conducted using a CAD model mimicking an 18650 battery (Fig. 1), consisting of three parts: the shell, spiral core, and internal media. The CAD model was projected using the CAD-ASTRA toolbox [4] with a helical scan, where the SOD was set to 80 mm, the source-detector distance was 750 mm, and the helix pitch equated 36.96 mm. The scan yielded 1000 projections covering an angular range of 16.65 radians, each projection consisting of 356×356 pixels with a 0.6 mm pitch. XCT images (260×260×899 voxels, 0.08 mm) were reconstructed with SIRT [5] and Katsevich's method, executed on a single NVIDIA A6000 GPU. SIRT took 40 seconds for 100 iterations (Fig. 1c), while Katsevich's BP took 4.2 seconds, with an additional 18.3 seconds for projection filtering on the CPU, totaling 25.8 seconds (Fig. 2).

(a) The CAD model (b) Full-view cone-beam projection (c) SIRT reconstruction

Figure 1: CAD model phantom and a central vertical slice of its helical SIRT reconstruction.

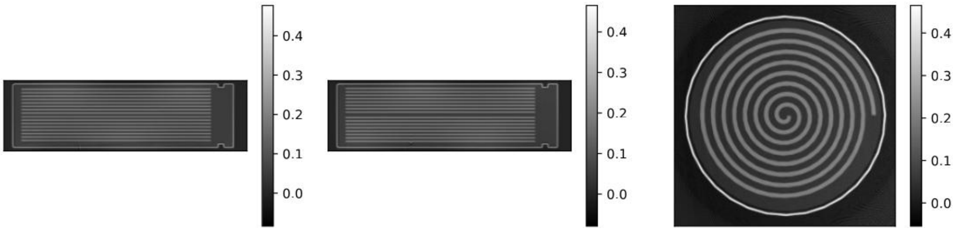

Figure 2: Orthogonal slices of the XCT image reconstructed with Katsevich's algorithm.

Conclusion

An open-source implementation of an FBP algorithm for helical CT was presented. It is based on the ASTRA toolbox and features fast GPU backprojection. This approach outperformed iterative reconstruction in execution time while delivering artifact-free reconstructions. Future work will include implementation of GPU-based projection data filtering and multi-GPU backprojection.

References

[1] F. Noo, J. Pack, D. Heuscher, Exact helical reconstruction using native cone-beam geometries, *Phys. Med. Biol.* 48: 3787−3818, 2003.

[2] W. van Aarle, W. J. Palenstijn, J. Cant, E. Janssens, F. Bleichrodt, A. Dabravolski, J, De Beenhouwer, J. Sijbers, Fast and flexible X-ray tomography using the ASTRA toolbox, *Opt. Express.* 22: 25129−25147, 2016.

[3] P. Paramonov, PyKatsevich, 2024. URL: https://github.com/astra-toolbox/helical-kats.

[4] P. Paramonov, N. Francken, J. Renders, D. Iuso, T. Elberfeld, J. De Beenhouwer, J. Sijbers, CAD-ASTRA: a versatile and efficient mesh projector for X-ray tomography with the ASTRA-toolbox, *Opt. Express*. 32: 3425–3439, 2024.

[5] J. Gregor, T. Benson, Computational analysis and improvement of SIRT, *IEEE Trans. Med. Imaging.* 27: 918-924, 2008.