

# Recent developments of CASToR at J-PET: an overview for GATE users

Aurélien Coussat

April 25, 2023

# Introduction to CASToR

# What is CASToR?

- ▶ Customizable and Advanced Software for Tomographic Reconstruction
- ▶ Written in C++ under the GNU GPL v3
- ▶ Generic platform to implement iterative reconstruction algorithms
- ▶ Supports several modalities
  - ▶ Positron Emission Tomography (PET)
  - ▶ Single-Photon Emission Computed Tomography (SPECT)
  - ▶ Computed Tomography (CT)
- ▶ More information:
  - ▶ <https://castor-project.org/>
  - ▶ <https://gitlab.com/castor-collaboration>



# CASToR datafiles

## Header (.Cdh)

Data filename:

benchmark\_pet\_list-mode\_tof.cdf

Number of events: 396627

Data mode: list-mode

Data type: PET

Start time (s): 822

Duration (s): 360

Scanner name: PET\_GE\_SIGNA\_PET-MR

Calibration factor: 1.45762e+08

Isotope: F-18

Attenuation correction flag: 1

Normalization correction flag: 1

Scatter correction flag: 1

Random correction flag: 1

## Data (.Cdf)

Table 3: Mandatory/Optional fields of a PET list-mode event

	Symbol	Description	Type	Mandatory
1	t	Time in ms	uint32.t	yes
2	a	Attenuation correction factor	FLTNBDATA	no
3	s	Un-normalized scatter intensity rate of the corresponding event (count/s) dependent on TOF	FLTNBDATA	no
4	r	Un-normalized random intensity rate of the corresponding event (count/s)	FLTNBDATA	no
5	n	Normalization factor of the corresponding event	FLTNBDATA	no
6	TOF	Difference in arrival time between c1 and c2 (ps)	FLTNBDATA	no
7	k	Number of contributing crystal pairs	uint16.t	if <i>Maximum number of lines</i> > 1
	For [k]			
8.1	c1	Crystal ID1	uint32.t	yes
8.2	c2	Crystal ID2	uint32.t	yes
	For [r]	r = Maximum number of lines per event minus k		
8.1	0	Garbage	32bits	yes
8.2	0	Garbage	32bits	yes

https:

[//castor-project.org/sites/default/files/2020-09/CASToR\\_general\\_documentation.pdf](https://castor-project.org/sites/default/files/2020-09/CASToR_general_documentation.pdf)

## Example CASToR command

```
castor-recon \  
  -df input.Cdh \  
  -dim 200,200,200 \  
  -vox 2.5,2.5,2.5 \  
  -th 0 \  
  -it 30:1 \  
  -dout output \  
  -ignore-corr fdur \  
  -norm norm.Cdh \  
  -proj multiSiddon,1,10 \  
  -vb 2
```

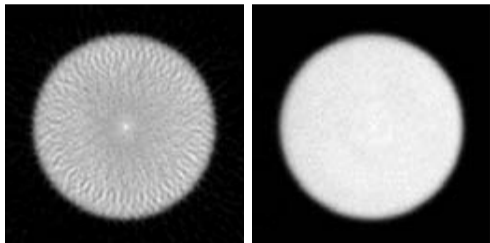
## CASToR utilities

CASToR comes with several utilities:

- ▶ `castor-datafileExplorer`
- ▶ `castor-datafileShuffler`
- ▶ `castor-scannerLUTExplorer`
- ▶ `castor-GATEMacToGeom` (`.mac` → `.geom`)
- ▶ `castor-GATERootToCastor` (`.root` → `.Cdh` and `.Cdf`)
- ▶ ...

# J-PET contributions

## Normalization in one minute



*Courtesy of Badawi et al., 1999.*

- ▶ Assuming a perfect source:  $C_{\text{LOR}} = A$
- ▶ However, because of effects,  
 $C_{\text{LOR}} = F_{\text{LOR}} \times A$
- ▶ We want to design  $\text{NF}_{\text{LOR}}$  such that  
 $C_{\text{LOR}} \times \text{NF}_{\text{LOR}} = A$
- ▶ Two options:
  - ▶ Direct normalization:  
 $\text{NF}_{\text{LOR}} = \frac{A}{C_{\text{LOR}}}$
  - ▶ Component-based normalization



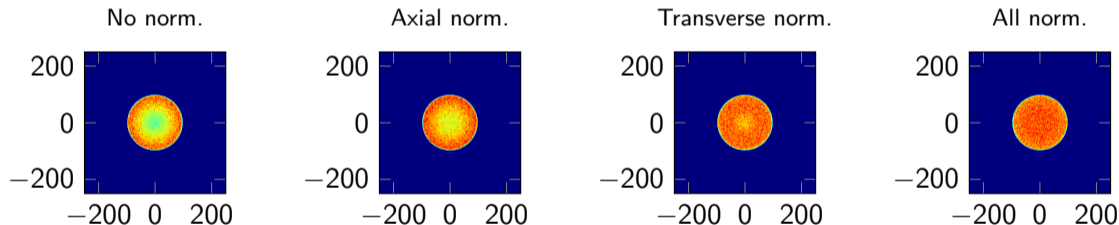
## Direct normalization with castor-norm

- ▶ New utility to compute direct normalization factors
- ▶ Asked several times by CASToR users
- ▶ [https://gitlab.com/castor-collaboration/castor/-/merge\\_requests/4](https://gitlab.com/castor-collaboration/castor/-/merge_requests/4)

```
castor-norm \  
-df normalization_scan.Cdh \  
-img normalization_phantom.img \  
-sc Scanner \  
-fout output
```

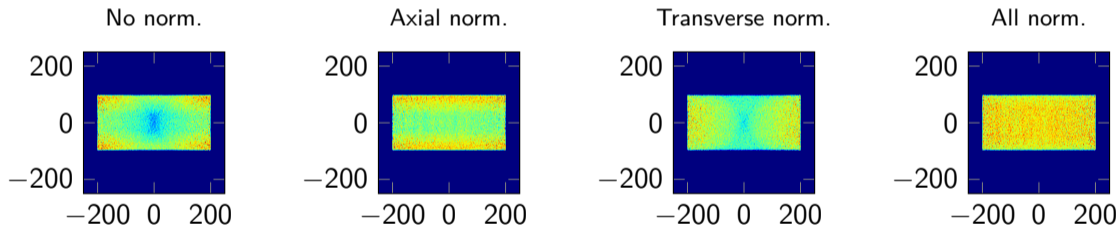
## Component-based normalization (reconstruction – z plane)

$$NF_{\text{LOR}} = \text{Axial norm.} \times \text{Transverse norm.} \times \text{Other effects...}$$



## Component-based normalization (reconstruction – x plane)

$$NF_{\text{LOR}} = \text{Axial norm.} \times \text{Transverse norm.} \times \text{Other effects...}$$



## castor-datafileMerger

```
castor-datafileMerger \  
-df input.Cdh \  
-norm with_normalization.Cdh \  
-fout output
```

input.Cdh		
<u>time</u>	<u>c1</u>	<u>c2</u>
1	1	2
2	1	3
3	2	3

with_normalization.Cdh		
<u>norm</u>	<u>c1</u>	<u>c2</u>
0.8	1	3
1.1	2	3
1.2	2	1

castor-datafileMerger

output.Cdh			
<u>time</u>	<u>norm</u>	<u>c1</u>	<u>c2</u>
1	1.2	1	2
2	0.8	1	3
3	1.1	2	3

## Python tools

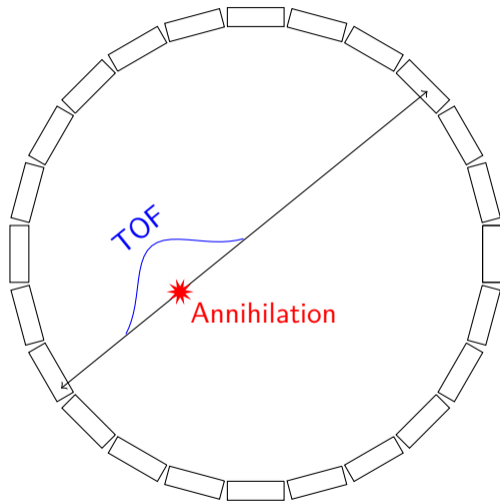
Several internal tools, among which

- ▶ `add_normalization_factors.py`, Python replica of `castor-datafileMerger`
- ▶ `replicate_castor_datafile.py`, generates CASToR datafiles for statistical bootstrapping
- ▶ `update_castor_datafile.py`, arbitrarily edits CASToR datafiles

```
def double_normalization(row):  
    row[CASToRCDFField.NORMALIZATION] *= 2  
    return row  
  
update_castor_datafile(cdh_path, output_cdh, output_cdf,  
                       double_normalization)
```

Towards a more generic Python CASToR library?

## Per-event TOF kernels



- ▶ Implemented per-event time-of-flight (TOF) kernels (courtesy of Damian Trybek)
- ▶ Allows to set a different TOF kernel for each event
- ▶ In the future: not only Gaussian kernels
- ▶ New parameter to `castor-GATERootToCastor`:  
`-TOF_branch`
- ▶ [https://gitlab.com/castor-collaboration/castor/-/merge\\_requests/8](https://gitlab.com/castor-collaboration/castor/-/merge_requests/8)

# Improved functional tests of CASToR

- ▶ Courtesy of Damian Trybek
- ▶ pytest wrapper around CASToR pipelines
- ▶ Get “faster feedback” when developing features for CASToR

```
aurelien@acoussat-latitude-3520: ~/Documents/redmine/20...
(castor-examples) aurelien@acoussat-latitude-3520:~/Documents/redmine/2043/castor-examples$ pytest . --castor-config /home/aurelien/Documents/redmine/2043/castor/config/ --castor-path /home/aurelien/Documents/redmine/2043/castor-build/
===== test session starts =====
platform linux -- Python 3.10.8, pytest-7.2.0, pluggy-1.0.0
rootdir: /home/aurelien/Documents/redmine/2043
plugins: xdist-3.2.1, check-2.0.0
collected 25 items

tests/test_castor_provided_examples.py .ss.....s.....ss. [ 80%]
tests/test_jpet_examples.py ..... [100%]

===== warnings summary =====
castor-examples/tests/test_castor_provided_examples.py::test_ct_histogram
castor-examples/tests/test_castor_provided_examples.py::test_pet_histogram
castor-examples/tests/test_castor_provided_examples.py::test_pet_listmode_reco
/home/aurelien/Documents/redmine/2043/castor-examples/tests/tools/command_tool
s.py:57: RuntimeWarning: invalid value encountered in divide
      abs_vec = np.abs(mean_difference_per_slice / mean_reference_per_slice)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
===== 20 passed, 5 skipped, 3 warnings in 395.41s (0:06:35) =====
(castor-examples) aurelien@acoussat-latitude-3520:~/Documents/redmine/2043/castor-examples$
```

# Conclusion



## Conclusion

- ▶ Development is (as always) ongoing!
  - ▶ Tools to interact with GATE output and CASToR datafiles
  - ▶ Tests for CASToR pipelines
- ▶ Some scripts can be contributed to `CASToR_tools`<sup>1</sup>
- ▶ Feel free to participate in the CASToR forum<sup>2</sup>

---

<sup>1</sup>[https://gitlab.com/castor-collaboration/CASToR\\_tools](https://gitlab.com/castor-collaboration/CASToR_tools)

<sup>2</sup><https://castor-project.discourse.group/>