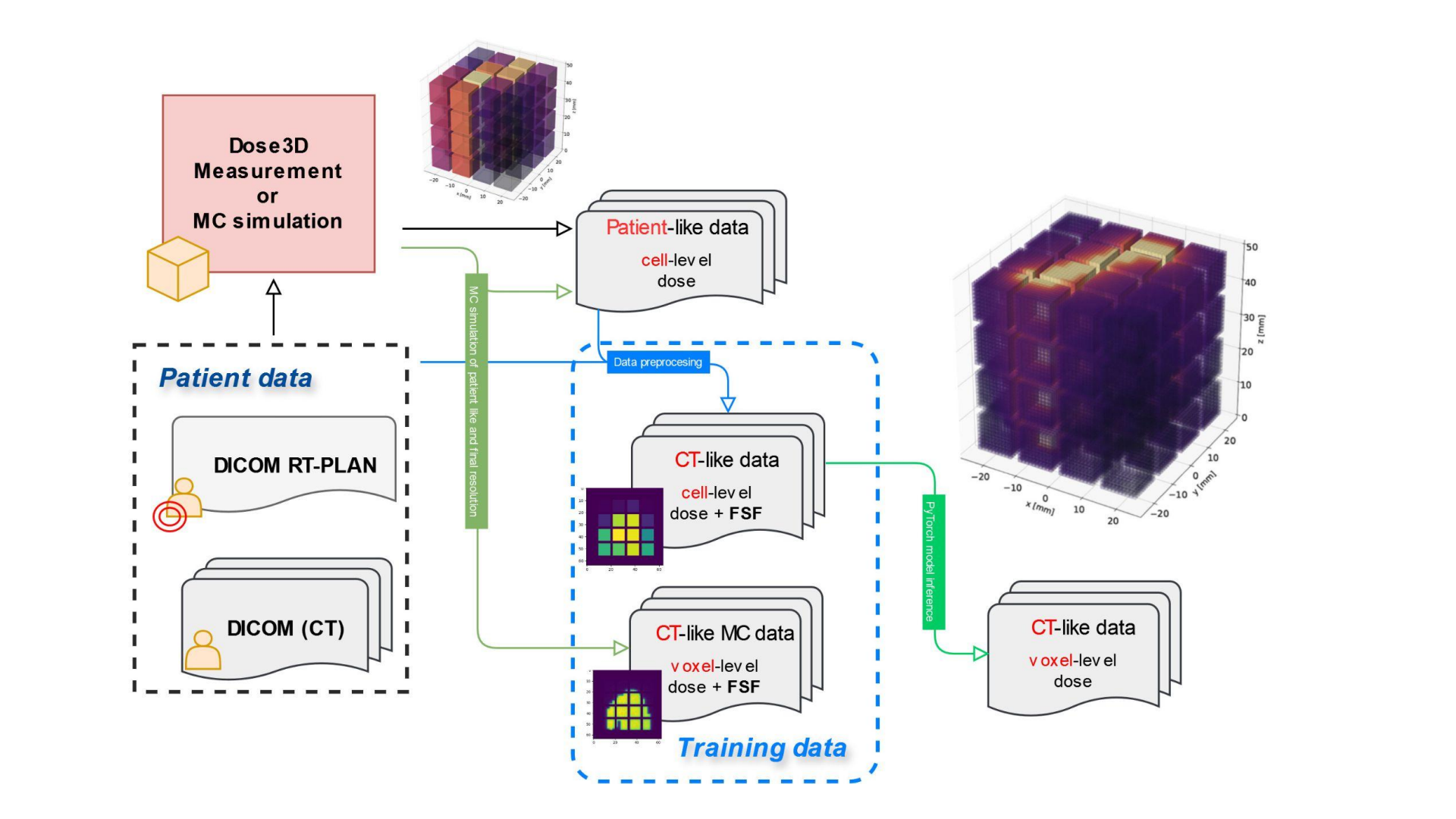
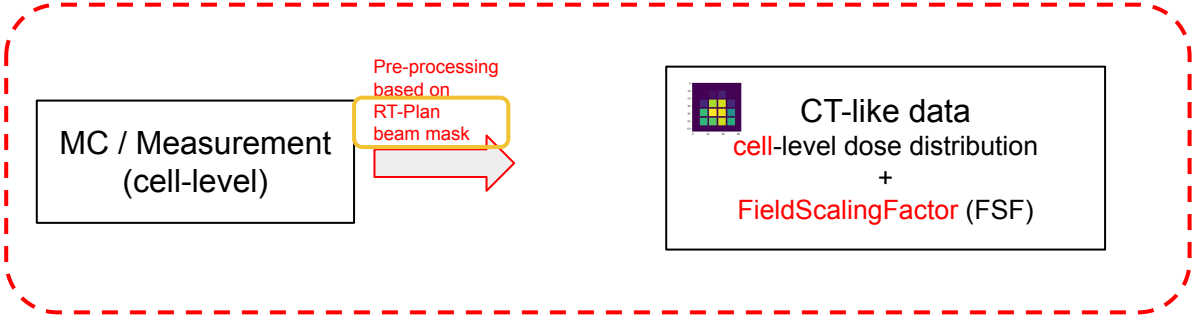

Dose3D
Measurement
or
MC simulation
Patient-like data
cell-level
dose
MC simulation of patient like and final resolution
Patient data
DICOM RT-PLAN
DICOM (CT)
Data preprocesing
CT-like data
cell-level
dose + FSF
CT-like MC data
voxel-level
dose + FSF
Training data
PyTorch model inference
CT-like data
voxel-level
dose
x [mm]
y [mm]
z [mm]

Pre-processing based on RT-Plan beam mask (research task: **G. Esser (FT)**)

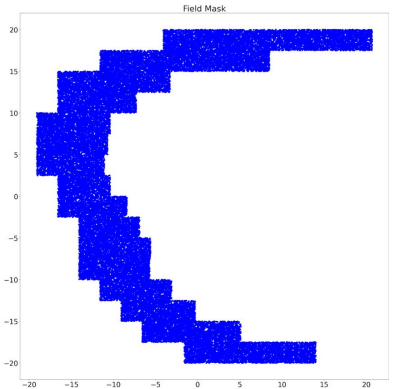


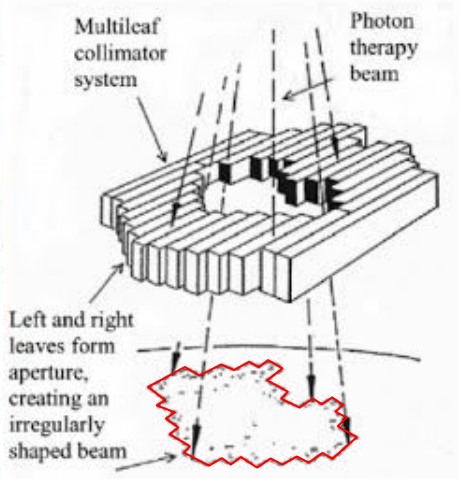
Input:

1) RT-Plan (beam mask)

> example prostate_imrt_beam0_cp0.dat

```
# Rotation: 0.0
# Particles: 1000
# Jaws: X1[mm],X2[mm],Y1[mm],Y2[mm]
-48,53,-43,43
# MLC: Y1[mm],Y2[mm]
-33.7,-33.7
-33.7,-33.7
-33.7,-33.7
-33.7,-33.7
-33.7,-33.7
-33.7,-33.7
```



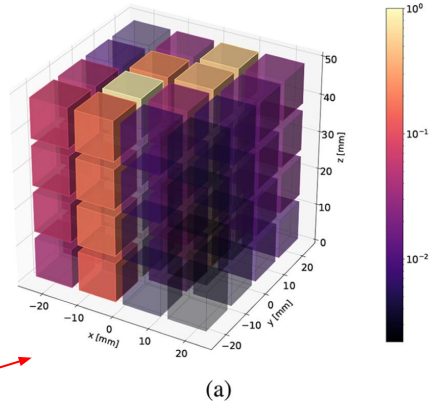


Pre-processing based on RT-Plan beam mask (research task: **G. Esser (FT)**)

MC / Measurement (cell-level)

Pre-processing based on RT-Plan beam mask

CT-like data
cell-level dose distribution
+
FieldScalingFactor (FSF)



(a)

Input:

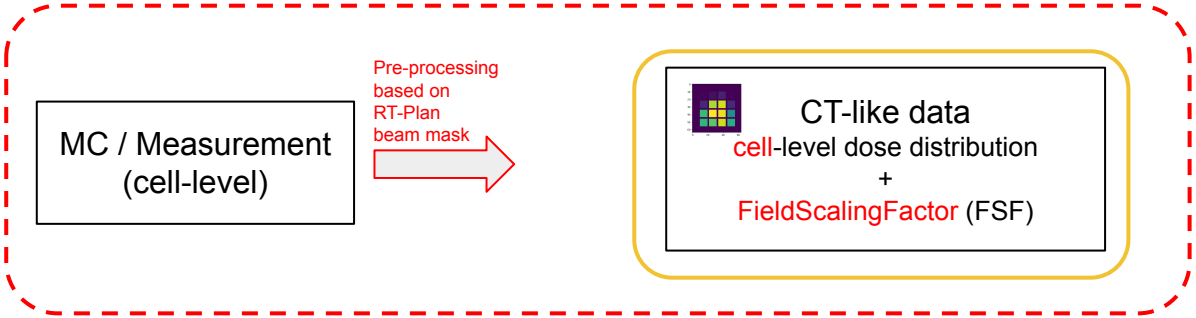
1) RT-Plan (beam mask)
2) Patient MC data (cell-level)
> example 4x4x2, bartek_beam0_run01/cp-0_d3ddetector_cell.csv

```
Cell IdX,Cell IdY,Cell IdZ,X [mm],Y [mm],Z [mm],Dose,FieldScalingFactor
0,1,1,-18,-6,18,3.58825e-08,0.11964
3,1,1,18,-6,18,3.03597e-10,0.119782
1,0,1,-6,-18,18,3.64189e-08,0.165187
2,1,0,6,-6,6,1.26026e-09,0.957306
```

3) Patient Cell Material (HU) 0.28947 (pmma)
3) Environment Material (HU) 0.0211788 (air)

Tool: voxel_plotter_ct_csv.py
> Input:
cp-0_d3ddetector_cell.csv
or
voxel/img0001(0064).csv

Pre-processing based on RT-Plan beam mask (research task: **G. Esser (FT)**)



Output:

1) **cell**-level → slice-like / CT

> example **64 x 64 x 64, 1 mm pix size,** bartek_beam0_run01/cp-0/cell/img0001(0064).csv

- 64 slices
- 64 x 64 pixels (1mm size)
- 1 mm between slices

```
X [mm],Y [mm],Z [mm],IdX,IdY,IdZ,Material,Dose [Gy],FieldScalingFactor
-32.5,-32.5,-32.5,-1,-1,-1,0.0211788,0,0
-32.5,-32.5,-31.5,-1,-1,-1,0.0211788,0,0
-32.5,-32.5,-30.5,-1,-1,-1,0.0211788,0,0
-32.5,-32.5,-29.5,-1,-1,-1,0.0211788,0,0
-32.5,-32.5,-28.5,-1,-1,-1,0.0211788,0,0
-32.5,-32.5,-27.5,-1,-1,-1,0.0211788,0,0
```

> Now we have 64 files as output, but simply merge all to single file

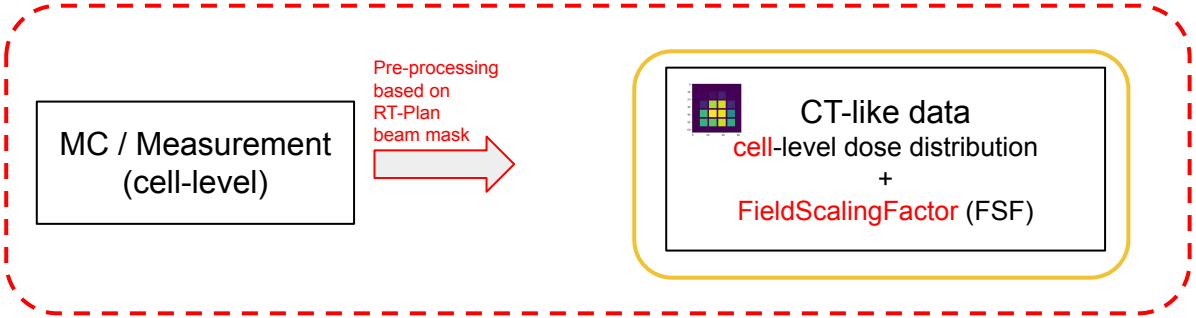
> Each pixel within a cell is assigned the dose value of that cell;

> Id's outside of cell are -1,-1,-1

> Coordinate system and indexing (see next slides)

> FSF calculation (see next slides)

Pre-processing based on RT-Plan beam mask (research task: **G. Esser (FT)**)



Remarks:

Coordinate system for slicing:

> The (0,0,0) is so called the iso-centre
> CT tube is positioned around the iso-centre, with a configurable size in *mm*
in a given example: sizeX=64, sizeY=64, sizeZ=64,
> Slicing is performed along the X-axis,
> The 1st slice (img0001.csv) corresponds to the minimum X position,

Cell's indexing:

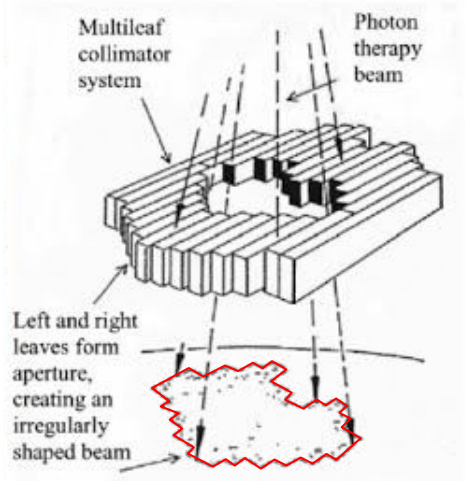
> To be read-in from input data (cell-level)
> The scheme from measurement will be different than from MC
> The goal is to *match* pixels from the CT-like slices to the certain cell from input data

Pre-processing based on RT-Plan beam mask (research task: **G. Esser (FT)**)

MC / Measurement (cell-level)

Pre-processing based on RT-Plan beam mask

CT-like data
cell-level dose distribution
+
FieldScalingFactor (FSF)



Remarks:

FSF calculation

> Calculated for each voxel of active volume, as a function of position, **P(z,y,z)**

> CPP @ G4RT implementation:
ControlPoint::GetMlcWeightedInfluenceFactor#L498

$$FSF = y1MLC + y2MLC$$

$$y(1,2)MLC = \sum_{leaf=1}^{64} \lambda_{leaf}$$

$$\lambda_{leaf} = \angle(r_{mlc} - r_{leaf})$$

$$r_{mlc} = MLC_{centre} - P$$
$$r_{leaf} = MLC_{leaf} - P$$

P – point of interests from active volume in patient

MLC_{centre} – centre of the MLC positioning, usually $(0,0,-340)$

Normalization to the patient configuration:

$$FSF_{pnorm} = {FSF}/{wx * nx + wy * ny + wz * nz}$$

wx = 1, wy = 2, wz = 3 represent the dimensional weights

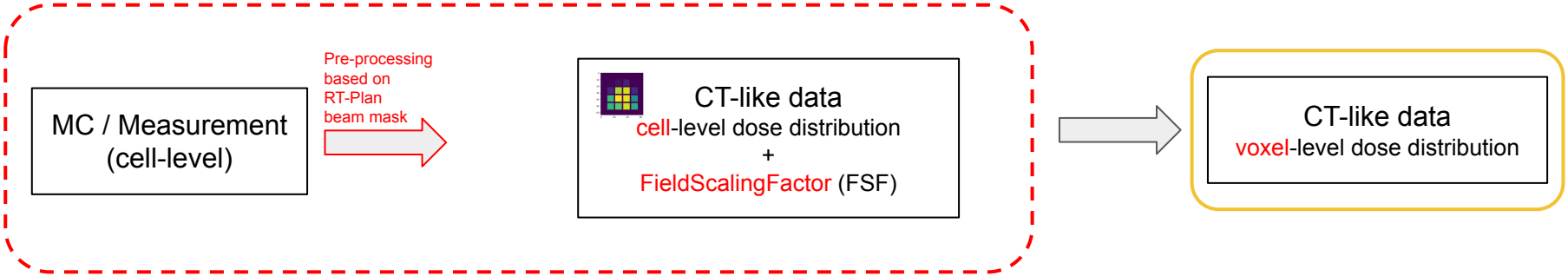
nx, ny, nz are for number of cells in the detector in each dim, e.g. 4x4x2

Finally it is normalized with min-max scaling:

$$FSF_{norm} = {FSF_{pnorm} - min}/{\max - min}$$

where min=0.02 and max=0.98

PyTorch model inference (**Kamila**)



Input:

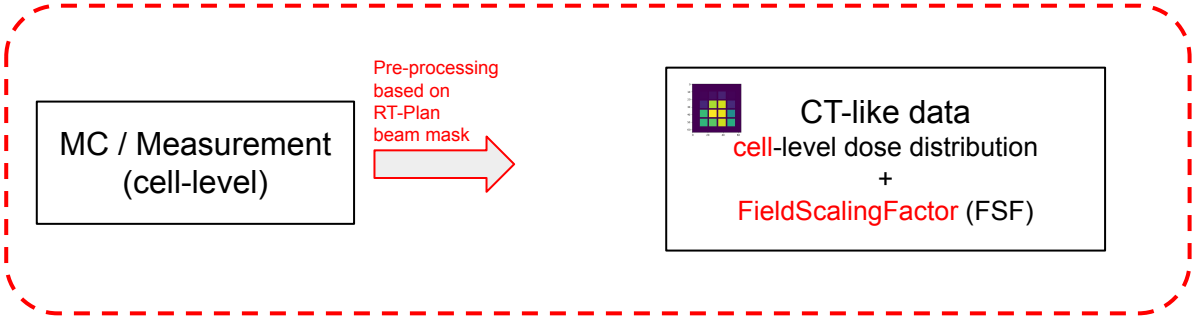
1) CT-like data witch cell-level dose distribution and FSF

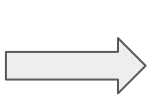
2) Trained torch model

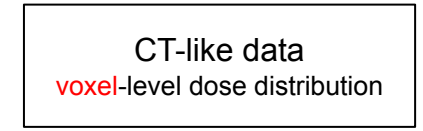
Output:

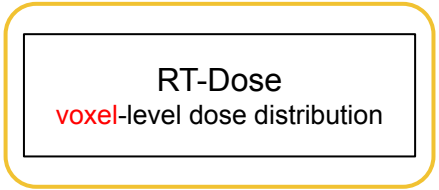
CT-like data witch **voxel**-level dose distribution
written to the geometric scheme taken from **cell**-level data
> Stored as a 3-dimensional matrix in CSV format.

PyTorch model inference (**projekt inżynierski J. Kawka**)









Input:

1) CT-like data witch **voxel**-level dose distribution

2) Patient meta data

Output:

CT-like data witch **voxel**-level dose distribution

> Stored as **DICOM-RT** format.