# Create Poly Correction Tables

#### Research Work Notes

## Use IIR to correct poly tables – 16/11/24

Optimizing to target is working – but poorly!

1. The target itself is usually not improving - First check why?
2. Make more bold steps

#### Check how the target is missed…

The correction is not at the exact position to improve the target!

First make the errors apparent – check their direction!

Find location and direction of most change in the new [image, radius] raster

#### New flow: Several steps per target

First select a target to correct

Then take several steps until it is significantly improved (at least 40%)

For each target there are 4 spatial options: Each of the 2 tubes, Left & Right

One of the tubes may not be effective.

## Use IIR to correct poly tables – 2/11/24

Some ideas:

1. Separate the “abs diff from target” score from all other scores
   1. Try accepting steps only by “average abs diff”
   2. Make sure that the max point that was targeted really improved
2. Maybe try correcting both tubes – and see which one is better?
3. Select patches of different radius for deviations of different size
   1. Prepare several initial patches
4. Avoid traps of repeatedly trying to correct failed corrections

## Use IIR to correct poly tables – 2/11/24

Add global log to help understand how steps are selected and what is the result

## Use IIR to correct poly tables – 30/10/24

Set a single target-level and strive to bring all rings to this level.

Find point with biggest deviation from target – and select relevant correction.

Direction of correction (up or down) is evident from the selected deviation.

After correction & recon, consider both local and global deviation score.

Later this method will also help to decide width and amplitude of correction.

Avoid marginal spaces – at least initially.

#### Create new “score” – distance from flat target at ring

Dump Deviation [Image, Radius] as displayable matrix

Apparently, there are very strong deviations on the margins –

Try to cut out the margins! Peel function seems to work OK.

#### First verify that problems are correctly identified

Even after peel – result seems to be wrong!

self.avgDev=-1.0128281116485596, iImage=92, iRad=0, maxDev=tensor(-458.4808)

Second run even worse:

self.avgDev=-0.9682614803314209, iImage=92, iRad=0, maxDev=tensor(-999.0014)

The inner rings should be deterministic – as there are only 4 pixels!

Something is basically wrong with the average per radius computations!

## Check IR and IIR in a single automatic loop – 29/10/24

In Identify Ring Source.py

First activate new recon option:

Set config file “d:\Config\Poly\Impulse.txt”: Tube, row, detector

Set BP dump name at: “d:\Config\Poly\BPDumpFileName.txt”

To something like: “d:/PolyCalib/Impulse\Poli\_AI\_t1\_r70\_d300\_width256\_height256\_zoom2.float.rvol”

## Correcting table by Reverse-IR function – 23-27/10/24

Work with new “impulse response values” to check full loop of IIR

D:\PolyCalib\Impulse

Poli\_AI\_t1\_r70\_d300\_width256\_height256\_zoom2.float.rvol

Load it and analyze it – to find source of ring!