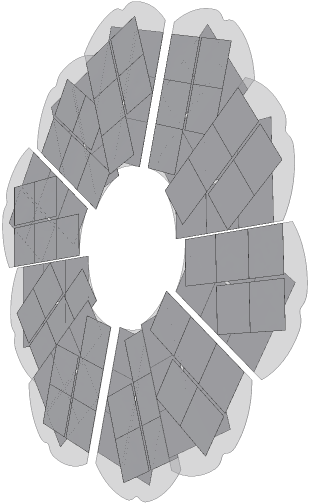
Alignment within one module of $\overline{\text{P}}$ANDA Luminosity Detector

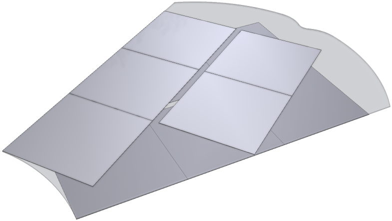
R. Klasen, M. Fritsch, P. Jasinski

Helmholtz-Institut Mainz
Johannes Gutenberg-Universität Mainz

$\overline{\text{P}}$ANDA XLV. Collaboration Meeting
June 25, 2013

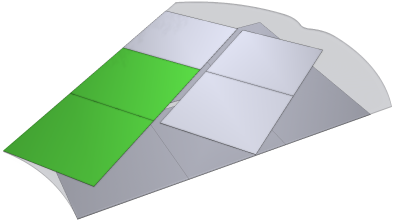
Luminosity pixel detectors





Task

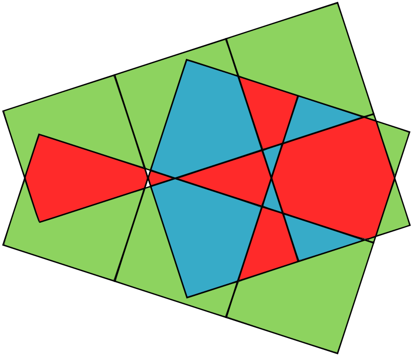
- pixel sensors glued on thin diamond plane from both sides
- quality fluctuations $\rightarrow$ only few sensors grouped to arrays of $(2 \times 4)cm^2$ (green), rest will be $(2 \times 2)cm^2$ sensors (grey)
- full disc coverage $\rightarrow$ 36° angle between sensors
- exact position and rotation of sensors must be known for track reconstruction



Pixel coincidence approach

Using overlapping area

- sensors will overlap partially
- use overlapping area to determine the alignment of each sensor pair



Pixel coincidence approach

Pixel coincidence approach

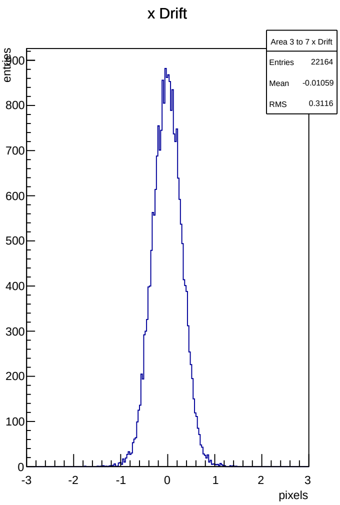
- charged particle will (ideally) activate single pixel in every sensor
- pixels on front and back must be close (depends on angle of entry and energy of particle)
- sensors have discrete pixels → approach will have certain maximum accuracy

Procedure

- simulation of perfectly aligned sensors (no shift in x and y, no rotation)
- using 1 million events to determine maximum achievable accuracy
- accuracy roughly estimated! (only centers of mass in pixel clusters are compared)

First result

units in diagrams in pixels with pixel size of $(80 \times 80)\mu m^2$



x: Avg. $\approx 0.8\mu m$, RMS $\approx 25\mu m$, y: Avg. $\approx -1.6\mu m$, RMS $\approx 25\mu m$

That means:

Perfectly aligned sensor:

- mean very close to 0
- pixel centers from front and back shifted by $\approx 1\mu m$ among each other

$\Rightarrow$ alignment with charged particles tracks possible!

Iterative Closest Point Algorithm

- algorithm to align two clouds of points in three dimensions (i.e. translation, rotation, scaling)
- here: determine transformation matrices from one sensor to every other sensor
- 9 overlapping areas $\rightarrow$ 9 matrices

$\Rightarrow$ position of every sensor with respect to reference sensor is known

Iterative Closest Point Algorithm

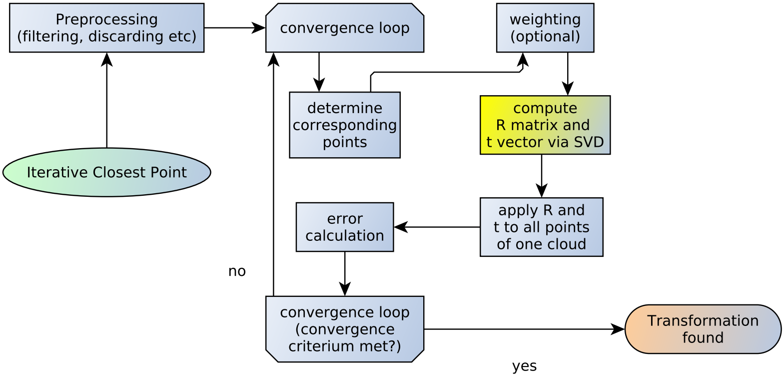
Inputs:

- points from two raw scans
- initial estimation of the transformation
- criteria for stopping the iteration

Output:

- refined transformation

Iterative Closest Point Algorithm



Singular Value Decomposition

compute
R matrix and
t vector via SVD

Singular Value Decomposition

- is a factorization of a matrix
- decompose transformation matrix
  $\rightarrow$ rotation + translation + scaling

Procedure

- subtract center of mass from every point
- point sets then are:

$$X' = x_i - \mu_x = x'_i$$
$$P' = p_i - \mu_p = p'_i$$

Singular Value Decomposition

- construct matrix W:

$$W = \sum_{i=1}^{N_p} x'_i p'^T_i \tag{1}$$

- and denote singular value decomposition (SVD) of W by:

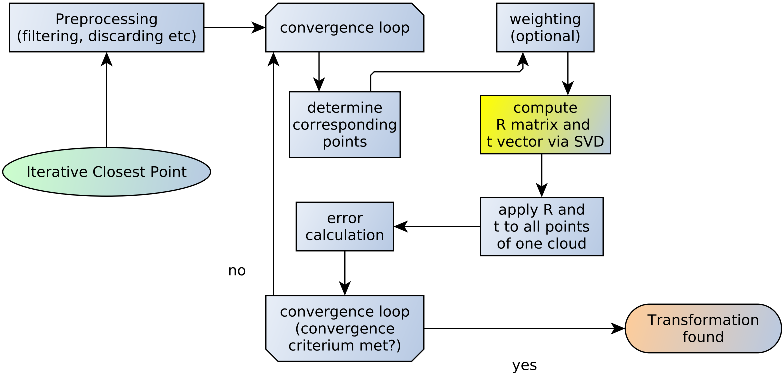
$$W = U \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{pmatrix} V^T$$

where $U, V \in \mathbb{R}^{3\times 3}$ are unitary and $\sigma_1, \sigma_2, \sigma_3$ are singular values of W

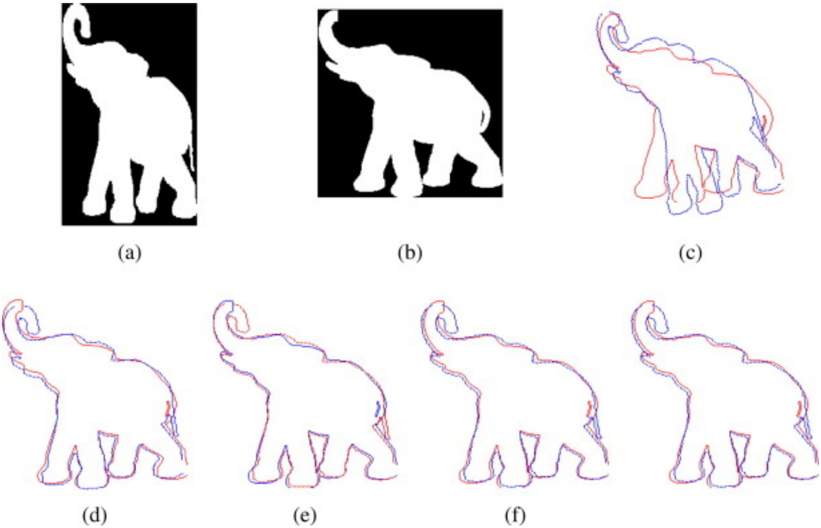
- rotation and translation are:

$$R = UV^T$$
$$t = \mu_x - R\mu_p$$

Iterative Closest Point Algorithm



Iterative Closest Point Algorithm (Visualization)[2]



[2]http://www.sciencedirect.com/science/article/pii/S0167865510000292

Conclusion

Pixel coincidence method

- using charged particles tracks and responding pixels
- simultaneously firing pixels must be close
- high statistic $\rightarrow$ high accuracy

Iterative Closest Point Algorithm

- finds transformation matrix to transform one set of points onto a corresponding set of points
- robust against inaccuracies (pixel not firing or noise)

this method is still being tested, but looks very promising