Implementation of New Track Pattern Recognition Methods into FairShip

Mikhail Hushchyn, Oleg Alenkin, Eric Van Herwijnen, Andrey Ustyuzhanin

11th SHiP Collaboration Meeting, 8 June 2017, CERN
Straw Tracker

Stations 1&2

Magnet

Stations 3&4

HNL

\( \pi \)

\( \mu \)

Y-view  U-view  V-view  YUVY-views

Y U V Y

X

Y

Layer 1  Layer 2  Layer 1  Layer 2

OD 9.83
ID 9.76

17.6  8.8

4.4

PLANE 1  PLANE 2
The PR Procedure Overview

1. Look for tracks in the 16 layers with horizontal straw tubes (Y-views, y-z plane).

2. For each track found in the Y-view, intersect the plane defined by this track and the x-axis with the stereo (U,V) hits. This gives (z,x) coordinates.

3. Look for tracks in 16 layers of stereo-views in x-z plane using (z, x) coordinates of the intersections.

4. Combine tracks before and after the magnet. Reconstruct a particle momentum and charge.
New PR in FairShip

New track pattern recognition methods are implemented into FairShip:


› The code description and instructions how to run it are provided.

› It have not integrated into FairShip/macro/ShipReco.py yet.
The PR procedure is divided into separate blocks.

The blocks interconnect with each other using defined APIs.

No global variables.

Changes in one block don’t affect the others.

Easy to update/upgrade.

Each function is described.
<table>
<thead>
<tr>
<th>Method**</th>
<th>Events Passed PR, %</th>
<th>Track Efficiency*, %</th>
<th>Left-Right Amb. Res., %</th>
<th>Time, ms/event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>94.1</td>
<td>99.2</td>
<td>94.5</td>
<td>18</td>
</tr>
<tr>
<td>Hough Transform</td>
<td>98.7</td>
<td>99.3</td>
<td>94.9</td>
<td>34</td>
</tr>
<tr>
<td>Artificial Retina</td>
<td>99.3</td>
<td>99.8</td>
<td>98.3</td>
<td>38</td>
</tr>
<tr>
<td>Reconstructor</td>
<td>96.8</td>
<td>99.0</td>
<td>82.9</td>
<td>39</td>
</tr>
</tbody>
</table>

* Track Efficiency is $\frac{N_{\text{correct reco hits}}}{N_{\text{reco hits}}}$

** Description of the methods are in backup slides.
Events Passed the PR

Events passing the pattern recognition

<table>
<thead>
<tr>
<th>EventsPassed</th>
<th>Entries 24410</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0</td>
</tr>
<tr>
<td>RMS</td>
<td>0</td>
</tr>
</tbody>
</table>

Baseline

HT

Entries Passed
Entries 24286
Mean 0
RMS 0
Momentum Reconstruction Accuracy

\[ \frac{(p - p_{\text{true}})}{p} \]

**Baseline**
- Entries: 5274
- Mean: -0.03264
- RMS: 0.04793

**HT**
- Entries: 5442
- Mean: -0.03118
- RMS: 0.03454
Momentum Reconstruction Accuracy vs Momentum

(p - p-true)/p

AR

Baseline

HT

Entries: 5445
Mean: 26.84
Mean y: -0.03262
RMS: 21.41
RMS y: 0.02047

Entries: 5442
Mean: 26.89
Mean y: -0.03718
RMS: 21.44
RMS y: 0.03456
Momentum Rec. Acc. vs Track Slope in YZ

\[(p - p_{\text{true}})/p\] from track direction in YZ plane

**AR**

- Entries: 5445
- Mean: -0.02246
- Mean \(y\): -0.03262
- RMS: 1.715
- RMS \(y\): 0.02045

**HT**

- Entries: 5442
- Mean: -0.02228
- Mean \(y\): -0.03118
- RMS: 1.711
- RMS \(y\): 0.03454
Track Efficiency vs Momentum

Fraction of hits the same as MC hits, total

<table>
<thead>
<tr>
<th></th>
<th>AR</th>
<th>HT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entries</td>
<td>5445</td>
<td>5442</td>
</tr>
<tr>
<td>Mean</td>
<td>26.84</td>
<td>26.89</td>
</tr>
<tr>
<td>Mean y</td>
<td>0.9977</td>
<td>0.9931</td>
</tr>
<tr>
<td>RMS</td>
<td>21.41</td>
<td>21.44</td>
</tr>
<tr>
<td>RMS y</td>
<td>0.01952</td>
<td>0.02892</td>
</tr>
</tbody>
</table>

Baseline

Fraction of hits the same as MC hits, total

HT

Fraction of hits the same as MC hits, total
RAM Leakage

Track PR methods use `ROOT.genfit.DAF()` to fit recognized tracks.

Experiments show that the fit is main source of RAM leakage:

- approx. 150 MB per 1k events in new implementation of the PR methods
- approx. 60 MB per 1k events in `macro/ShipReco.py` (current PR)

Other steps of PR use negligible amount of RAM.
## Conclusion

<table>
<thead>
<tr>
<th>Method**</th>
<th>Events Passed PR, %</th>
<th>Track Efficiency*, %</th>
<th>Left-Right Amb. Res., %</th>
<th>Time, ms/event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>94.1</td>
<td>99.2</td>
<td>94.5</td>
<td>18</td>
</tr>
<tr>
<td>Hough Transform</td>
<td>98.7</td>
<td>99.3</td>
<td>94.9</td>
<td>34</td>
</tr>
<tr>
<td>Artificial Retina</td>
<td>99.3</td>
<td>99.8</td>
<td>98.3</td>
<td>38</td>
</tr>
<tr>
<td>Reconstructor</td>
<td>96.8</td>
<td>99.0</td>
<td>82.9</td>
<td>39</td>
</tr>
</tbody>
</table>

- The best and more accurate PR method is AR
- Speed-Efficiency balance
- The methods are implemented into FairShip
Backup
Track Efficiency vs Angle between Tracks in YZ

Fraction of hits the same as MC hits, total

AR
frac_total_angle
Entries 5434
Mean 2.177
Mean y 0.9977
RMS 1.722
RMS y 0.01955

Baseline
frac_total_angle
Entries 5176
Mean 2.226
Mean y 0.9934
RMS 1.673
RMS y 0.02862

HT
frac_total_angle
Entries 5417
Mean 2.171
Mean y 0.9932
RMS 1.712
RMS y 0.02669
The PR Procedure Overview

1. Look for tracks in the 16 layers with horizontal straw tubes (Y-views, y-z plane).

2. For each track found in the Y-view, intersect the plane defined by this track and the x-axis with the stereo (U,V) hits. This gives (z,x) coordinates.

3. Look for tracks in 16 layers of stereo-views in x-z plane using (z,x) coordinates of the intersections.

4. Combine tracks before and after the magnet. Reconstruct a particle momentum and charge.
Baseline [1]

Track PR procedure in a plane:
1. Estimate line parameters \((k, b)\) for two hits.
2. Find track hits within a window width from the line.
3. Repeat steps 1 and 2 for each pair of hits.
4. Select two tracks with the largest number of unique hits.

Hough Transform

\[(x_0, y_0)\]

\[b = y_0 - x_0k\]

\[y = k_0x + b_0\]

\[(k_0, b_0)\]
Hough Transform

Track PR procedure in a plane:
1. Estimate line parameters \((k, b)\) for two hits.
2. Use Hough Transform to find track hits within \((dk, db)\) parameters region.
3. Repeat steps 1 and 2 for each pair of hits.
4. Select two tracks with the largest number of unique hits.
The artificial retina function is defined as:

$$ R(\theta) = \sum_i e^{-\frac{\rho^2(\theta, x_i)}{\sigma^2}} $$

where $\rho(\theta, x_i)$ is distance between the i-th hit and a track with parameters $\theta$.

For 2D tracks:

$$ \rho(\theta, x_i) = y_i - (kx_i + b) \quad \theta = [k, b] $$
Artificial Retina

Track PR procedure in a plane:

1. For each pair of hits estimate line parameters \((k, b)\) and calculate AR function value.
2. Use pair with the highest value as init point of AR function maximization procedure.
3. Fit track parameters by maximizing AR function value.
4. Select hits within a window width from the fitted track. Exclude the hits from the sample.
5. Repeat steps 1-4.
Let's search whole track before and after the magnet simultaneously.

A track parametrization $\theta$: $\theta = (l_0, l_1, y, m, x)$
RANSAC

Searching for one track:
1. The RANSAC selects a random subset of the hits.
2. The linear model is fitted using this subset.
3. The error of the data with respect to the fitted model is calculated.
4. The number of inlier candidates is calculated.
5. Steps 1-4 are repeated until the maximum number of iterations.
6. A model with maximum number of inliers is returned.
Reconstructor

Track PR procedure:

1. Project hits to the magnet plane as shown.
2. Estimate the $y$ parameters of tracks.
3. Use RANSAC to estimate track parameters and hits before and after the magnet for $Y$ and Stereo views as described in slide 14.
4. Select two tracks with the largest number of unique hits.