EVENT RECONSTRUCTION IN JUNO

Wuming Luo
(on behalf of JUNO)
Dec. 2nd @ ACAT 2021

Institute of High Energy Physics
Chinese Academy of Sciences
OUTLINE

- Introduction of JUNO
- Traditional methods
  - Vertex reconstruction
  - Energy reconstruction
- Machine Learning based methods
  - Inputs, Models etc
- Summary

*Disclaimer:*
all studies based on Monte Carlo Simulation;
many studies still on-going;
mainly focusing on the methods
Jiangmen Underground Neutrino Observatory (JUNO):
- Determine the neutrino mass ordering
- Measure neutrino oscillation parameters to sub-percent level
- SuperNova, Solar, Atm. Geo. etc

<table>
<thead>
<tr>
<th>DETECTOR</th>
<th>TARGET MASS</th>
<th>ENERGY RESOLUTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>KamLAND</td>
<td>1000 t</td>
<td>6%/√E</td>
</tr>
<tr>
<td>D. Chooz</td>
<td>8+22 t</td>
<td>8%/√E</td>
</tr>
<tr>
<td>RENO</td>
<td>16 t</td>
<td>8%/√E</td>
</tr>
<tr>
<td>Daya Bay</td>
<td>20 t</td>
<td>5%/√E</td>
</tr>
<tr>
<td>Borexino</td>
<td>300 t</td>
<td>5%/√E</td>
</tr>
<tr>
<td>JUNO</td>
<td>20000 t</td>
<td>3%/√E</td>
</tr>
</tbody>
</table>

Need to collect large statistics
being 50km away from source
Unprecedented light level
1200 pe/MeV

Both features
• are highly expensive (civil engineering + photocathode density)
• result in extreme detector dynamic range
Liquid Scintillator  
20kton

Central Detector* 
~17,612 20” PMTs  
+ ~25,600 3” PMTs  
+ ~75% coverage

Depth: 44m

Top Tracker

Water Pool

Central Detector*  
Acrylic sphere  
Stainless steel truss

Φ: 43.5m

Wuming Luo
Particles deposit energy in Liquid Scintillator $\rightarrow$ emitting photons $\rightarrow$ detected by PMTs
Charge and time of PMTs $\rightarrow$ vertex & energy
Importance: crucial for physics sensitivity
TRADITIONAL METHODS
## Vertex Reco.

<table>
<thead>
<tr>
<th>Methods</th>
<th>PMT info.</th>
<th>pros&amp;cons</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charge Center</td>
<td>charge</td>
<td>simple and fast</td>
<td>initial value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>less accurate</td>
<td></td>
</tr>
<tr>
<td>Peak Time Fitter</td>
<td>time</td>
<td>simple</td>
<td>more accurate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>more accurate</td>
<td>initial value</td>
</tr>
<tr>
<td>Time Likelihood</td>
<td>time</td>
<td>complex and most</td>
<td>final value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>accurate</td>
<td></td>
</tr>
</tbody>
</table>
Charge weighted average position of fired PMTs

\[ \bar{r}_0 = a \cdot \frac{\sum_i q_i \cdot \bar{r}_i}{\sum_i q_i}, \]

Large bias near the edge due to photon leakage
Define “residual time”
\[ \Delta t_i(j) = t_i - \text{tof}_i(j), \quad j\text{-th iteration} \]

Apply correction to the vertex
\[ \tilde{\delta}[\tilde{r}(j)] = \frac{\sum_i \left( \frac{\Delta t_i(j) - \Delta t_{\text{peak}}(j)}{\text{tof}_i(j)} \right) \cdot (\tilde{r}_0(j) - \tilde{r}_i)}{N_{\text{peak}}(j)}, \]

Iterate until \( \Delta t \) shape converges
Define residual time

\[ t^i_{\text{res}}(\vec{r}_0, t_0) = t_i - \text{tof}_i - t_0, \]

Construct pdf \( p(t_{\text{res}}) \)

Minimize likelihood function

\[ \mathcal{L}(\vec{r}_0, t_0) = -\ln \left( \prod_i p(t^i_{\text{res}}) \right). \]
PERFORMANCE

- Bias near the detector edge
- PMT Transit Time Spread (TTS) is the dominant factor
Energy Reco.

- Simple total PE method: $E \sim \text{total PE}$
- **Maximum likelihood method**
  - optical model independent
  - calibration data driven
  - taking into account differences among PMTs
- Main factors for energy resolution:
  - photon statistics
  - **energy non-uniformity**
  - PMT dark noise
Step 1: use calibration data to construct the expected number of PhotoElectron $\hat{\mu}(r, \theta, \theta_{PMT})$ for PMTs

Step 2: maximize the likelihood function

$$\mathcal{L}(\{k_i\}|r, \theta, \phi, E_{vis}) = \prod_i \mathcal{L}(k_i|\theta, \phi, E_{vis}) = \prod_i \frac{e^{-\mu_i} \cdot \mu_i^{k_i}}{k_i}$$

$$\mu_i = E_{vis} \cdot \hat{\mu}_i$$

$\{k_i\}$ — detected PE for PMTs

$E_{vis}$ — visible energy
<table>
<thead>
<tr>
<th>Source</th>
<th>Type</th>
<th>Energy [MeV]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$^{68}\text{Ge}$</td>
<td>$\gamma$</td>
<td>$2 \times 0.511$</td>
</tr>
<tr>
<td>$^{60}\text{Co}$</td>
<td>$\gamma$</td>
<td>$1.173 + 1.333$</td>
</tr>
<tr>
<td>AmC</td>
<td>(n,H)$\gamma$</td>
<td>2.22</td>
</tr>
<tr>
<td>Laser</td>
<td>op</td>
<td>1</td>
</tr>
</tbody>
</table>

- Obvious energy non-uniformity in the total reflection region
- Laser($^{68}\text{Ge}$) is better at high(low) energy
COMBINED SOURCE

- Energy deposition of positron in LS
  - kinetic part: point-like
  - annihilation part: ball-like
- Use combined source Laser+$^{68}$Ge to mimic positron

$\hat{\mu}_{comb} = \frac{1}{E_{vis}} \cdot (E_{vis}^{Ge} \cdot \hat{\mu}^{Ge}(r, \theta, \theta_{PMT}) + E_k \cdot \hat{\mu}^{L}(r, \theta, \theta_{PMT}))$

$E_{vis} = E_{vis}^{Ge} + E_k$

*E_k — kinetic energy of e⁺
Combined source improves the energy-uniformity (consequently energy resolution) in the total reflection region.
Machine Learning methods
Applying Machine Learning to event reconstruction for Liquid Scintillator detectors

Large number of PMTs $O(10^5)$
- treating each PMT as a pixel
- ensemble of PMTs charge/time form an image

Image is highly vertex and energy dependent

Image recognition $\leftarrow$ vertex/energy reconstruction
**INPUTS**

- PMTs installed on a sphere
- **Method 1**: projection to 2D plane \(\rightarrow\) Plane CNN

![Images of PMTs and charge/first hit time channels](image_url)

(a) Map of PMTs.  
(b) Charge channel.  
(c) First hit time channel.
INPUTS

- PMTs installed on a sphere
- Method 2: HEALPix \(\rightarrow\) plane/spherical CNN
Fig. 5. The planar projection method first generates a mapping of the ID of the PMTs and the position of the pixel in the image (a). The charge (b) and first hit time (c) information can be filled in the image according to the mapping.

Fig. 6. VGG-J network architecture for CNN reconstruction with 17 weight layers: 13 convolutional and 4 dense layers. It is composed of two main blocks: a sequence of 3 \(3 \times 3\) convolutional layers (with max pooling used for coarsening) and a few dense layers at the end. The last dense layer is used to output the prediction result, which is 1 node for reconstructing energy and 3 nodes for reconstructing vertex coordinates.

Compared with the VGG network that has two layers with 4096 nodes, the amount of parameters in VGG-J network is 26 million, which has been reduced by 65%, while the reconstruction accuracy has remained at the same level.

3.3.3. ResNet-J

In order to maximize the reconstruction performance, we would like to train a network that has more layers, which may bring better learning ability. However, a deeper network is not similarly easy to optimize. This is caused by the problem of vanishing/exploding gradients. Not thoroughly optimized network may have only a lower accuracy. In order to solve this problem, we use ResNet network architecture. The main feature of ResNet is the usage of residual blocks, shown in Fig. 7, where \(x\) denotes the input of the block. In a regular NN the block yields the feature mapping \(H(x)\), while the ResNet lets the block fit another feature mapping \(F(x) = H(x) \ast x\) which is called residual mapping. Therefore, the original mapping is converted into \(F(x) + x\). It has been discussed that it is easier to optimize the residual mapping than to optimize the original one due to the effect of identity skip connections.

Compared with the original ResNet network architecture, we optimized convolutional layers and the dense layers for the reconstruction in JUNO. The final network structure is shown in Fig. 8 and contains a total of 53 layers with approximately 35 million trainable parameters. In comparison with VGG-J it has more layers to enhance the learning ability, see Table 4.

<table>
<thead>
<tr>
<th>VGG-J</th>
<th>ResNet-J</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight layers</td>
<td>17</td>
</tr>
<tr>
<td>Number of weights</td>
<td>26 million</td>
</tr>
<tr>
<td>Parameters</td>
<td>26 million</td>
</tr>
</tbody>
</table>

Table 4 Comparison of VGG-J and ResNet-J architectures.
In this work, we adapt the DeepSphere model for graphs and properly define convolutions and pooling due to the need for a projection. The previous Section for CNNs. One possible workaround is to define an arbitrary projection of PMTs in JUNO does not allow to directly use the signal as input.

### Table 5

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of epochs</td>
<td>15</td>
</tr>
<tr>
<td>Batch size</td>
<td>64</td>
</tr>
<tr>
<td>Loss</td>
<td>Mean Squared Error</td>
</tr>
</tbody>
</table>

The main idea is to encode the topology of the input domain in the features coming from spherical regions with different areas and maintain areas. So, during convolution, the same filter will capture proportional to distances on the spherical surface. So, features that are close on the sphere can be far in the 2D projection, meaning that they may not be captured by a local filter.

As it was already mentioned in Section 3.3, the spherical arrangement of the input domain in JUNO makes it very efficient. However, this comes with a few problems:

- If the spherical shape, breaking translational invariance and making learning more difficult.
- Representing their "closeness", with the spherical pixel. Then, defining the graph's nodes which will hold the vertexes' labels in the adjacency matrix, such that the connection weights encode all the information about the neighbors on the sphere.

First, we need to define the graph's nodes which will hold the vertices' labels in the adjacency matrix, such that the connection weights encode all the information about the neighbors on the sphere.

Thus, the connection weights encode all the information about the neighbors on the sphere. Each traversed link incurs in a decay factor inversely proportional to that link's weight. Thus, the higher away. However, in this procedure, further node values are propagated on its previous value, and that of all nodes that are less than neighbors on the sphere.

To model the spherical surface, and define nodes in the graph as the regions' vertices can be labeled in an hierarchical discretization meaning that they can be labeled in an hierarchical discretization. Each node should be connected only to its nearest neighbors on the sphere.

The minimum number of links between two vertices in 3D space. However, we need also a way to iteratively group input samples. A natural choice would be to directly use the PMTs as input. This can be justified by the fact that the signal from the PMTs is a function of the distance between the event and the PMT, which is invariant to the projection. However, this comes with a few problems:

- If the spherical shape, breaking translational invariance and making learning more difficult.
- Representing their "closeness", with the spherical pixel. Then, defining the graph's nodes which will hold the vertexes' labels in the adjacency matrix, such that the connection weights encode all the information about the neighbors on the sphere.

Thus, the connection weights encode all the information about the neighbors on the sphere. Each traversed link incurs in a decay factor inversely proportional to that link's weight. Thus, the higher away. However, in this procedure, further node values are propagated on its previous value, and that of all nodes that are less than neighbors on the sphere.

To model the spherical surface, and define nodes in the graph as the regions' vertices can be labeled in an hierarchical discretization meaning that they can be labeled in an hierarchical discretization.
The Cartesian distance between the true and reconstructed vertex.

The uncertainties of the fit values are shown on the plots with vertex errors, indicating the range of uncertainty. This approach is used for both the vertex and the energy reconstruction.

Bias and resolution are studied as a function of two variables. The results are sampled versus radial position and, near the edge, by the total internal reflection in the acrylic sphere.

The detector is symmetric versus rotation; therefore, the main difference in reconstruction arises from a distance between the detector center and the vertex — its radial position, which is used as a second reference in reconstruction.

The performance of the vertex reconstruction is studied as a function of both visible energy and radial position. It is reported in absolute volume spherical layers and provides equal statistics samples.

Events in the center of the detector produce a more symmetric response. The events on the edges of the detector are affected by the variable. Events in the center of the detector produce a more symmetric response. The events on the edges of the detector are affected by the variable. Events in the center of the detector produce a more symmetric response.

The detector is symmetric versus rotation; therefore, the main difference in reconstruction arises from a distance between the detector center and the vertex — its radial position, which is used as a second reference in reconstruction.

The number of triggered PMTs grow with the energy, which makes the appearance due to positron–electron annihilation. The light collection and effects of the light scattering and re-emission are well fit by Gaussian. Since the edge values are outside the region of interest of physics, which has a range of about 10 MeV, the distribution of the reconstructed energy at the edges of the dataset becomes asymmetric, as shown in Figs. 12 and 13. The topological is parameterized by the coefficients of a 022 MeV 1st-order Chebyshev polynomial, which are part of the model's learnable parameters. However, for the large JUNO dataset, the predictions distributions are not uniform, and the systematic shift introduced by the model is not negligible.

The current ResNet-J result shows that the absolute value of bias is less than 2.5% of the total positron energy and the electron mass, which is calculated as follows:

\[ E_{vis} = E_{e} + m_{e} = E_{kin} \]

where \( E_{vis} \) is the visible energy, \( E_{e} \) is the electron energy, \( m_{e} \) is the electron mass, and \( E_{kin} \) is the kinetic energy of the positron-electron pair. The light attenuation in the LS, effects of the light scattering and re-emission are taken into account, see Fig. 13. It is worth noting that while the angular resolution is high, the bias might be higher due to the limited number of triggered PMTs.

The performance of the energy reconstruction is studied as a function of number of photo-electrons, because to the first order the resolution is defined as

\[ \sigma(E) \propto E^{\gamma} \]

where \( \sigma(E) \) is the energy resolution, \( E \) is the energy, and \( \gamma \) is a constant. The energy resolution by error bars.

This approach is used for both the vertex and the energy reconstruction. The uncertainties of the fit values are shown on the plots with vertex errors, indicating the range of uncertainty. This approach is used for both the vertex and the energy reconstruction.

The uncertainties of the fit values are shown on the plots with vertex errors, indicating the range of uncertainty. This approach is used for both the vertex and the energy reconstruction. The uncertainties of the fit values are shown on the plots with vertex errors, indicating the range of uncertainty. This approach is used for both the vertex and the energy reconstruction.
**Performance**

- Vertex reconstruction with ResNet-J as an example
- Small bias throughout the whole detector
- Comparable resolution w.r.t. traditional methods

---

**Heatmap of Resolution of the vertex for different part of the detector (left) and the resolution of the whole detector for different TTS/DN options (right).**

The plots are offset along the $X$-axis within $3 \text{ cm}$ for better readability.

**An example of energy prediction distributions for different energies (right).**

The latter one is fit with Gaussian function. The predictions are produced with BDT.

**Fig. 12.**

**Fig. 13.**

**Fig. 14.**

**Z. Qian, V. Belavin, V. Bokov et al.**

Nuclear Inst. and Methods in Physics Research, A 1010 (2021) 165527
Precise event reconstruction is crucial for JUNO
Developed various traditional algorithms
Applied ML to event reco. for LS detectors, promising first look
Many further studies are still on-going, please stay tuned