Track and Vertex Reconstruction for Dark Photon Search at DarkSHINE

Zejia Lu, Huayang wang, Yifan Zhu, Yulei Zhang, Liang Li

Shanghai Jiao Tong University

July 22nd , 2025

2nd Workshop on Tracking in Particle Physics Experiment

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Content

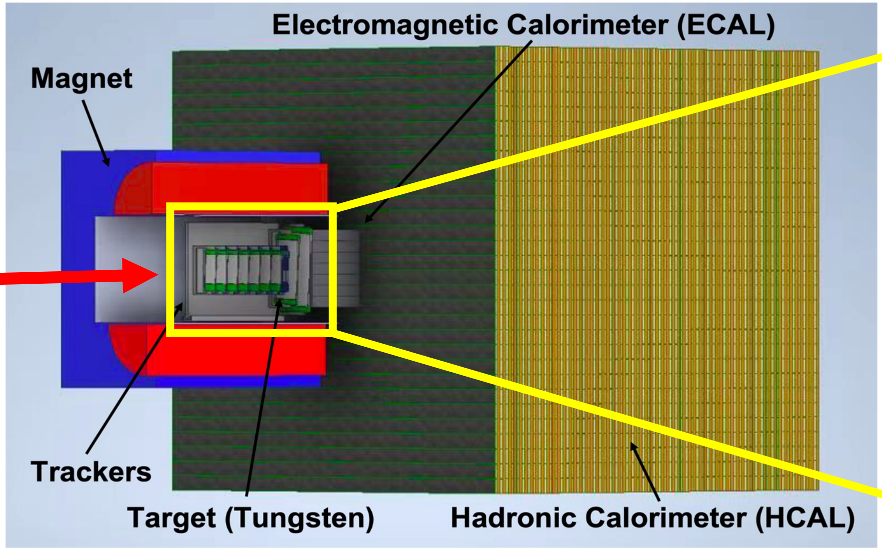
- Introduction
- Tracking
  - Traditional method
  - GNN method
- Vertexing
  - Vertexing method
  - Application on dark photon search
- Summary

DarkSHINE Experiment

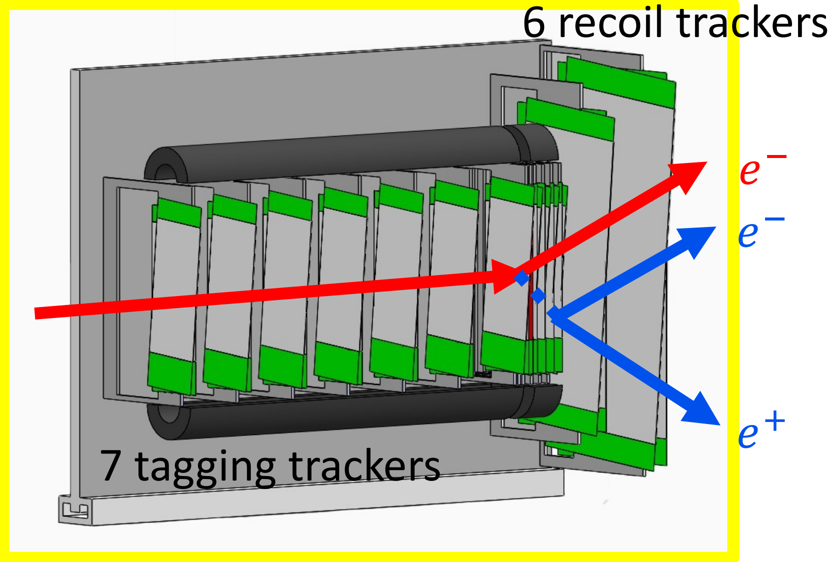
- DarkSHINE is a proposed **fixed-target** experiment that utilizing **electron beam from SHINE** aimed at searching the light dark matter.
- The DarkSHINE detector system is consisted of **tagging tracker**, **recoil tracker**, **ECAL** and **HCAL**.

8 GeV e^- beam

from Shanghai High Repetition-Rate XFEL and Extreme Light Facility (**SHINE**)



- **Stagger strip trackers**



Tracker parameter	Strip width	Sensor thickness
	30 µm	150 µm

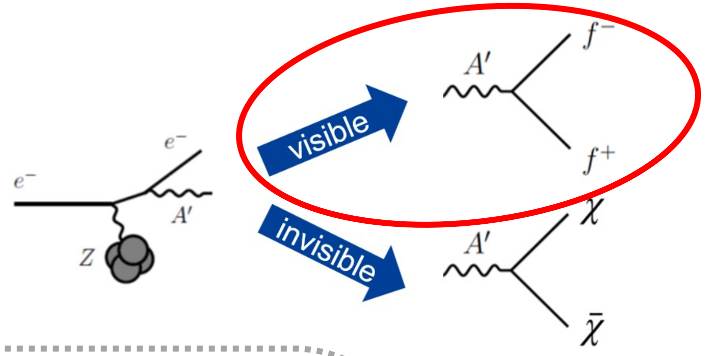
Dark Photon Search

- **Light dark matter search: dark photon**

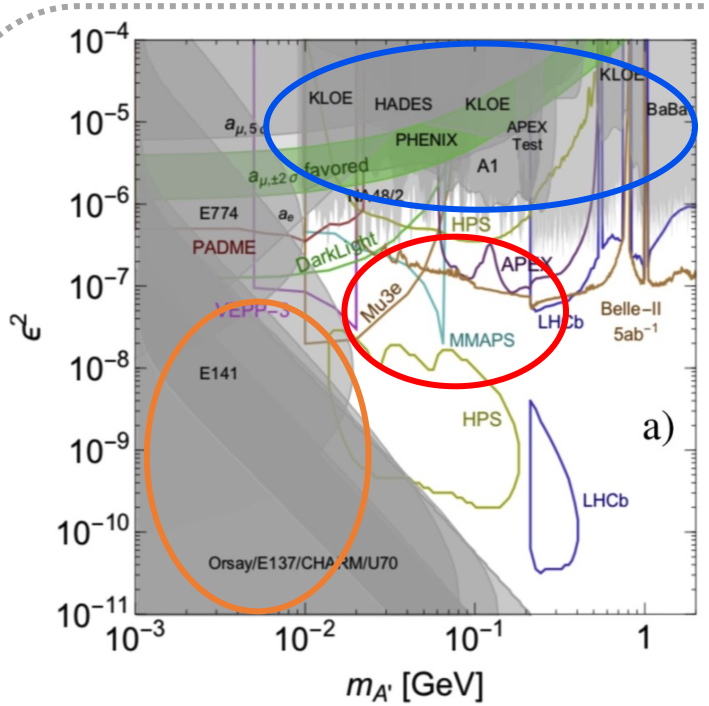
$$\mathcal{L} = \mathcal{L}_{SM} + \boxed{\frac{1}{2}\frac{\epsilon}{\cos\theta_W} F^{Y,\mu\nu} F'_{\mu\nu}} + \frac{1}{4} F'^{,\mu\nu} F'_{\mu\nu} + m_{A'}^2 A'^2$$

Two model parameters:
- Coupling constant ϵ
- Dark photon mass $m_{A'}$

We will focus on **visible decay**



According to the decay channel of dark photon, we have invisible and visible decay.



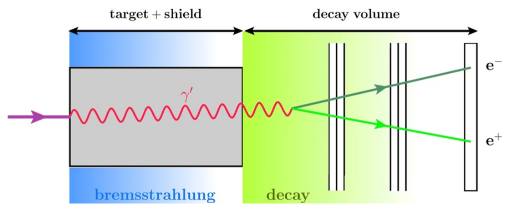
arixv:2006.04640

Bump hunting : High production rate

Challenging region :
- Signal rate too low for bump hunting
- Lifetime too short for beam-dump experiment

-> **Displaced vertex reconstruction needed!**

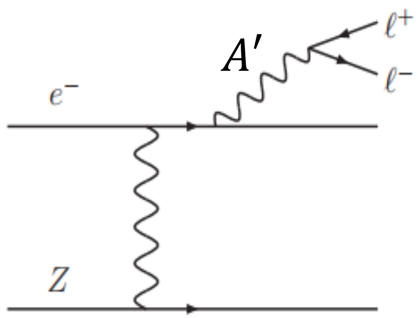
Beam-dump experiment : Long decay length



Signal and Background

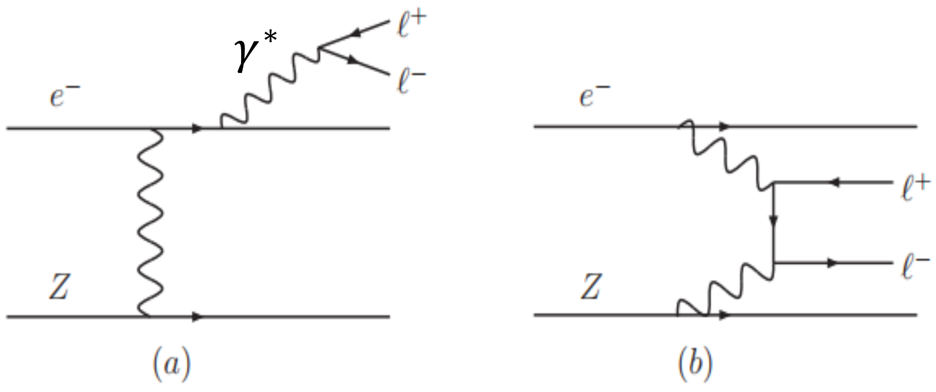
arixv:0906.0580

Signal



A' production and decay

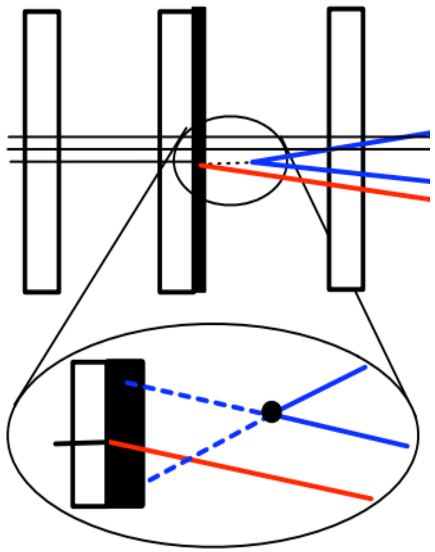
Main background
(both virtual $\gamma \to e^+e^-$)



γ^* trident reaction

Bethe-Heitler trident reaction

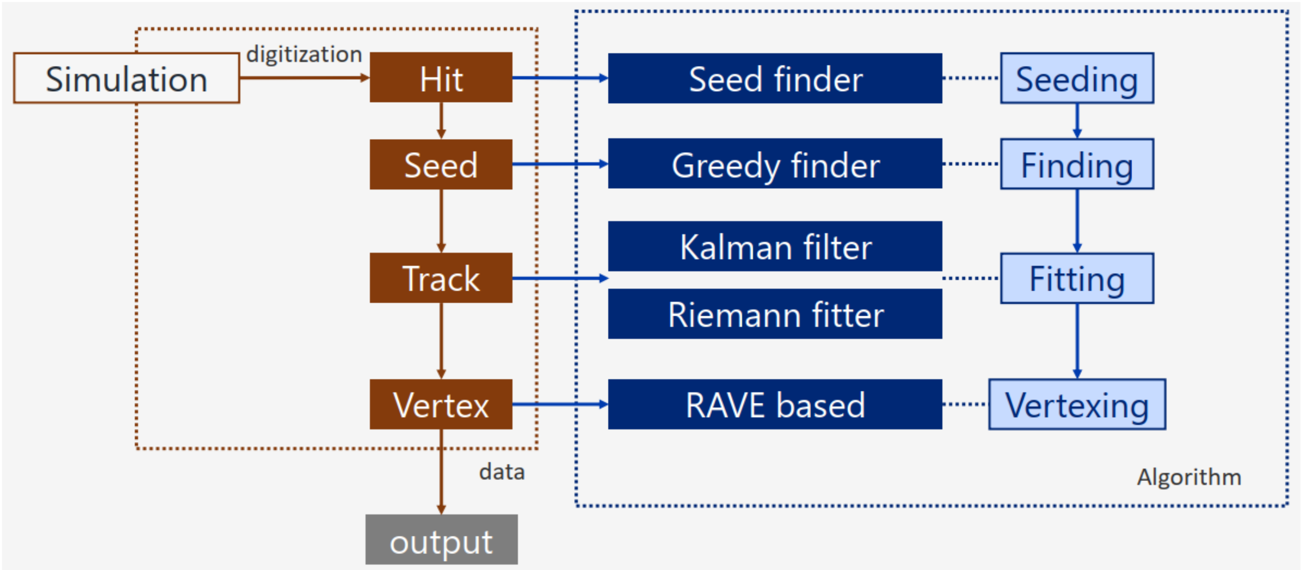
Displaced vertex

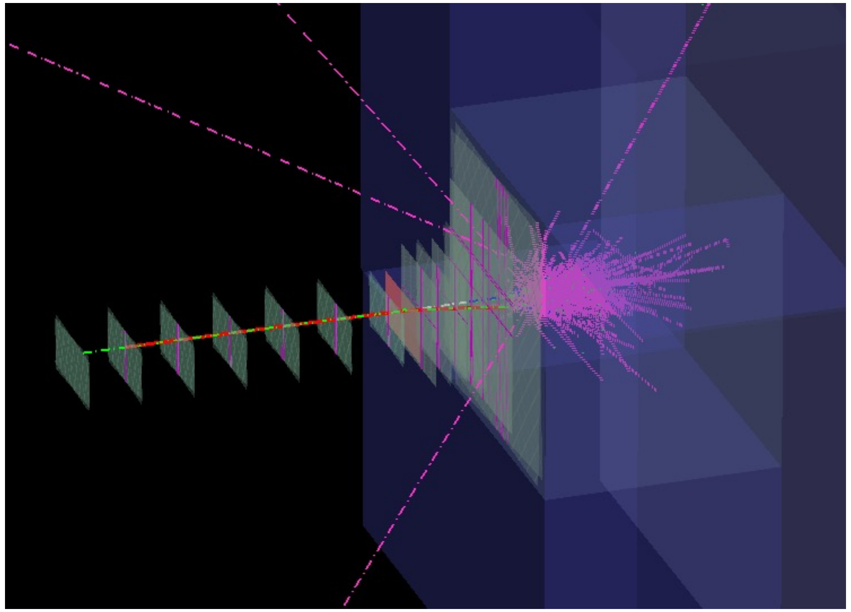


- To differentiate signal from background, the key is to **reconstruct the displaced vertex through tracking and vertexing.**

Tracking and Vertexing

- We apply **full chain reconstruction** from hits to tracks and to vertexes. We adopt Kalman Filter algorithm for both tracking (**GenFit**) and vertexing (**Rave**).
- The simulation is based on **Geant4. CalcHEP** generator is used for signal production.



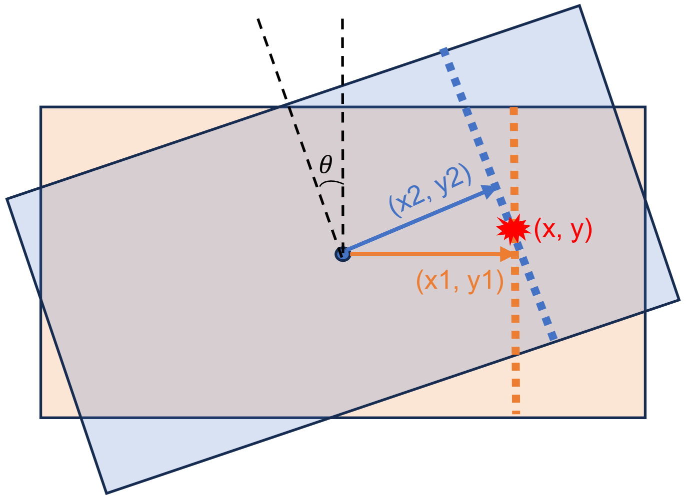


Hit Reconstruction

Strip clustering

Merge u, v layer

- Cluster strip hits on single layer.
- Use mean shift clustering, based on the weighted distance (energy deposition of hits) to the center of cluster.



$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} x_1 \cos\theta_1 - y_1 \sin\theta_1 \\ x_2 \cos\theta_2 - y_2 \sin\theta_2 \end{pmatrix}$$

$$\boldsymbol{A} = \begin{pmatrix} \cos\theta_1 & -\sin\theta_1 \\ \cos\theta_2 & -\sin\theta_2 \end{pmatrix}$$

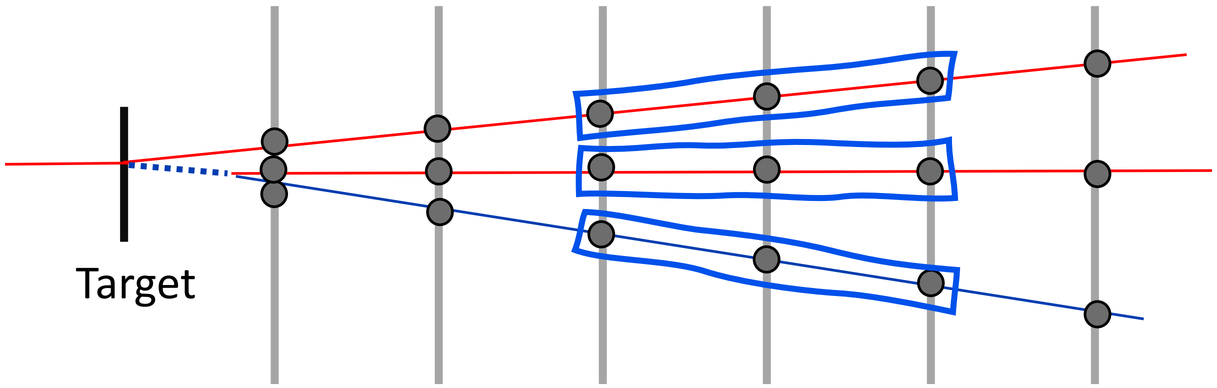
$$\begin{pmatrix} x \\ y \end{pmatrix} = \boldsymbol{A}^{-1} \begin{pmatrix} u \\ v \end{pmatrix}$$

$$\begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{pmatrix} = \boldsymbol{A}^{-1} \begin{pmatrix} \sigma_u^2 & \sigma_{uv} \\ \sigma_{uv} & \sigma_v^2 \end{pmatrix} (\boldsymbol{A}^{-1})^T$$

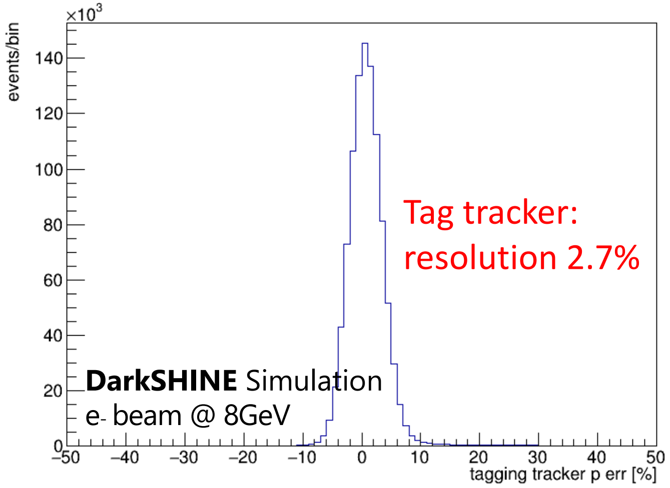
- Use transformation of coordinates, can handle any cases.
- With error propagation.

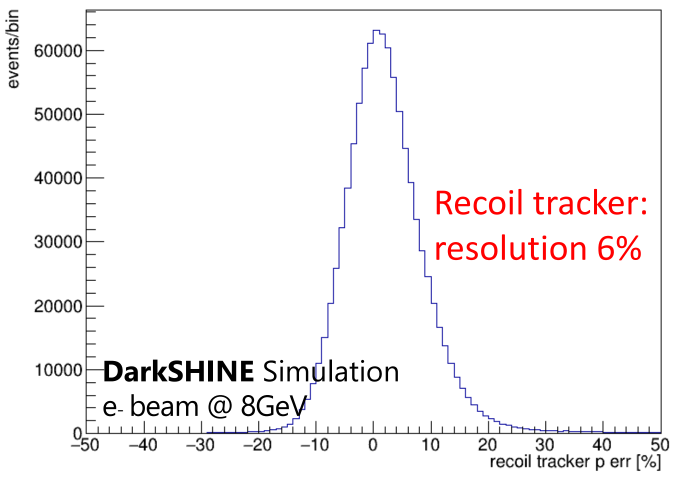
Track Finding and Fitting

- Track finding
  - Greedy algorithm + circle fit (Kasa method).
  - Good tracking efficiency for single-track
    - 97% in single-track case
    - 70% in 3-track case, improvement needed



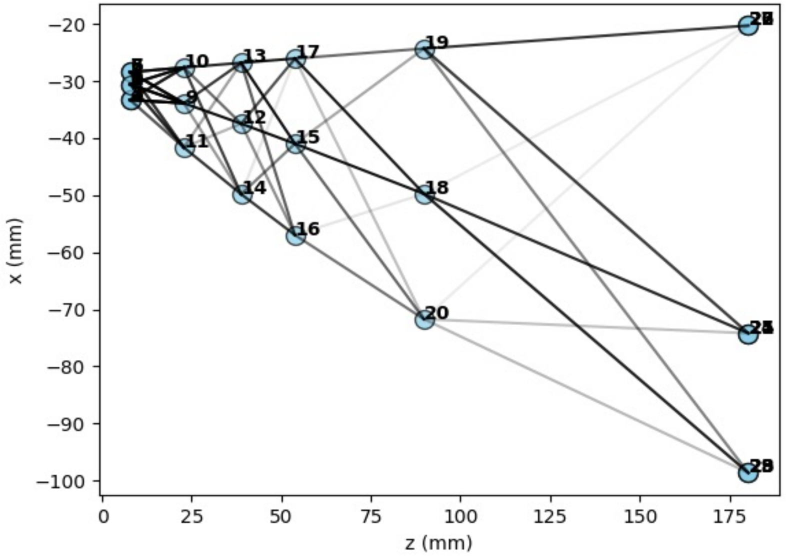
- Track fitting
  - Kalman filter algorithm based on GenFit.
  - Track momentum resolution
    - 3% at tagging tracker
    - 6% at recoil tracker.

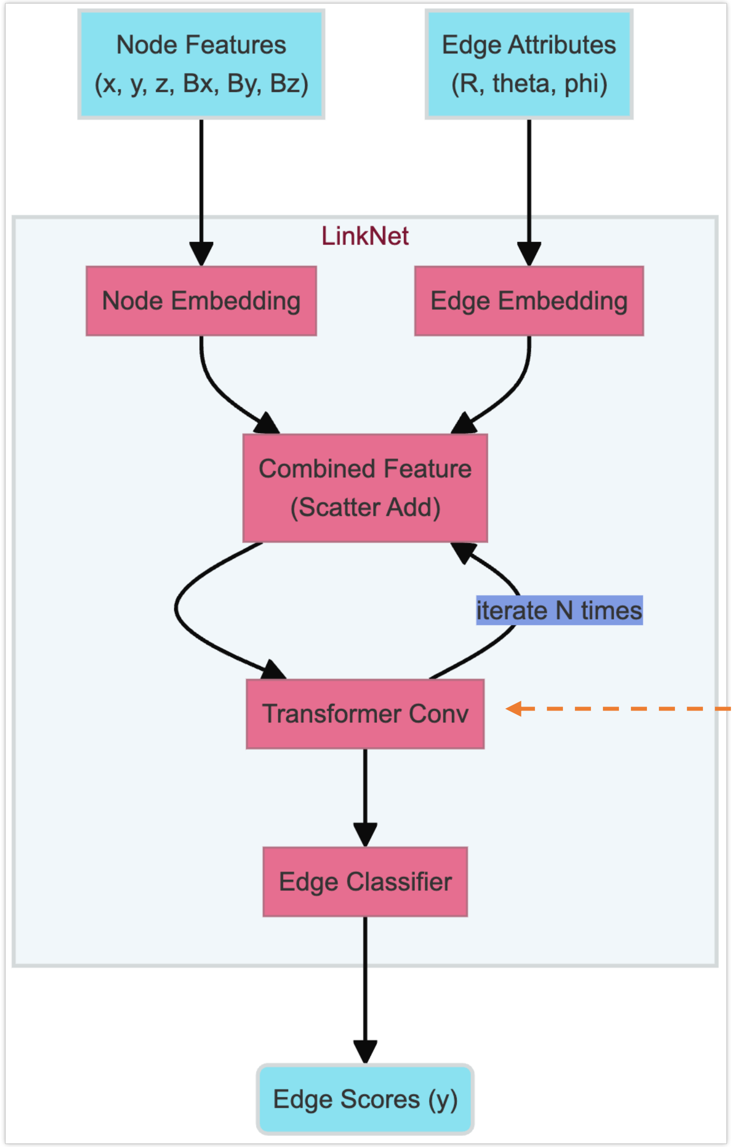




GNN Tracking: Build the Network

- We connect hits from adjacent tracker module to form a graph for each event.
- The task of GNN is to predict the truth edge that connects hits from the same track.





LinkNet (*classification*):

- To predict if **an edge is a true particle trajectory** passing through the **two connected node**

Transformer Conv

Combine the strengths of GNN and transformer model

GNN Tracking: Training

Loss function

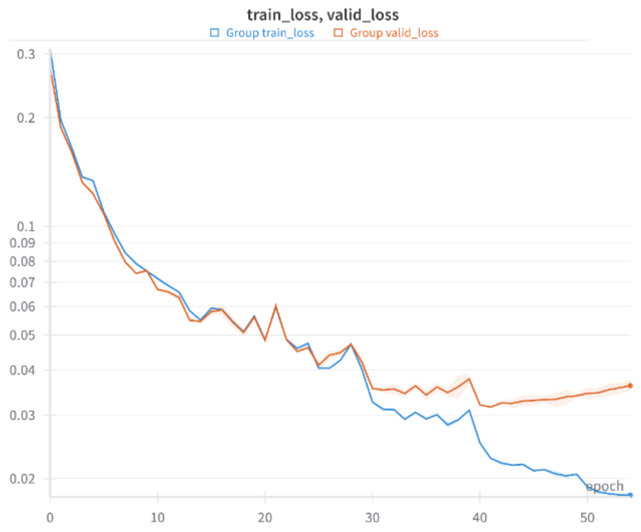
- Binary cross-entropy with logit
- $L = \sum_i [-\hat{y}_i \log(\sigma(y_i)) - (1 - \hat{y}_i) \log(1 - \sigma(y_i))]$

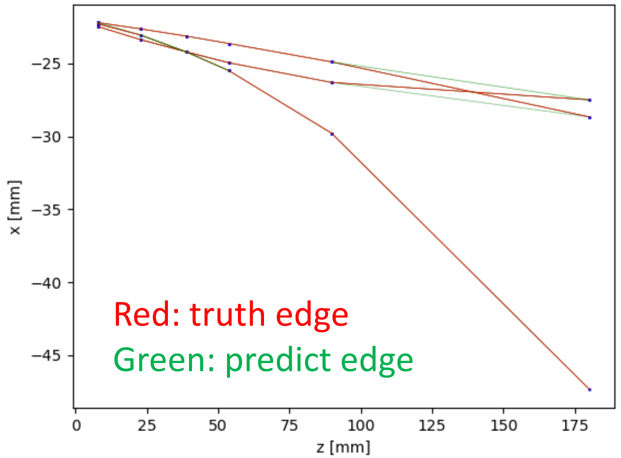
Optimization

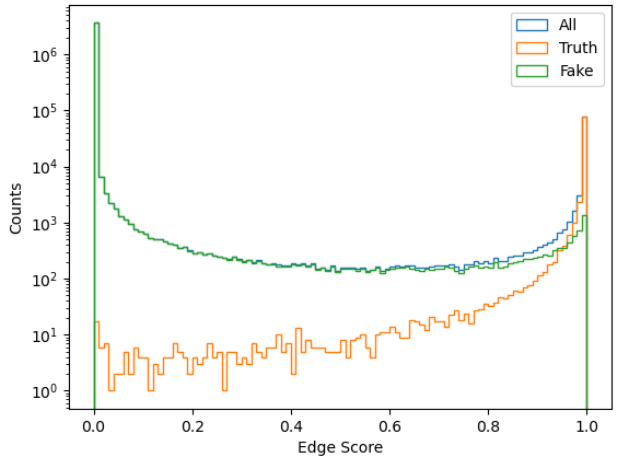
- Adam algorithm

Dataset

- 10^5 dark photon visible decay events
- $\text{train: validation: test} = 3:1:1$

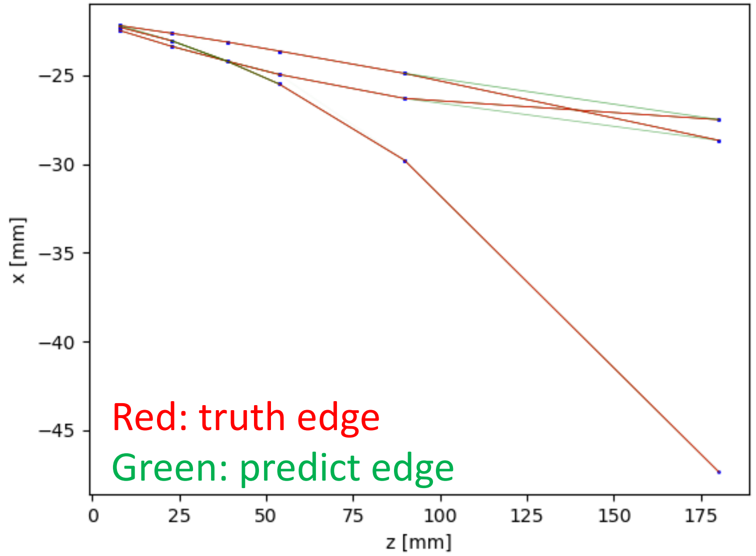


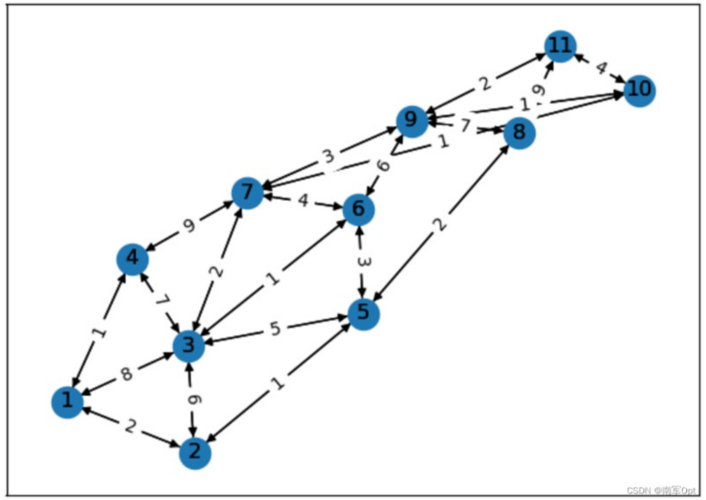




GNN Tracking: Result

- The output of GNN is the score of each edge. We use the Dijkstra algorithm to cluster the edges into track.



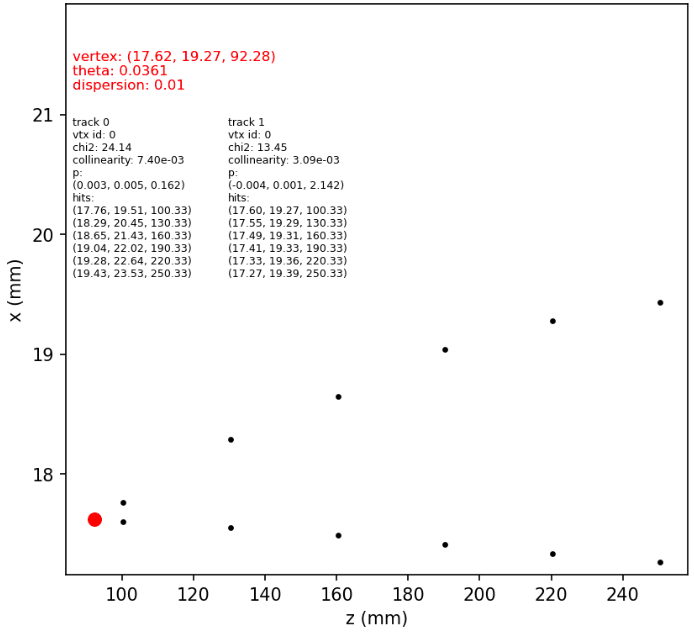


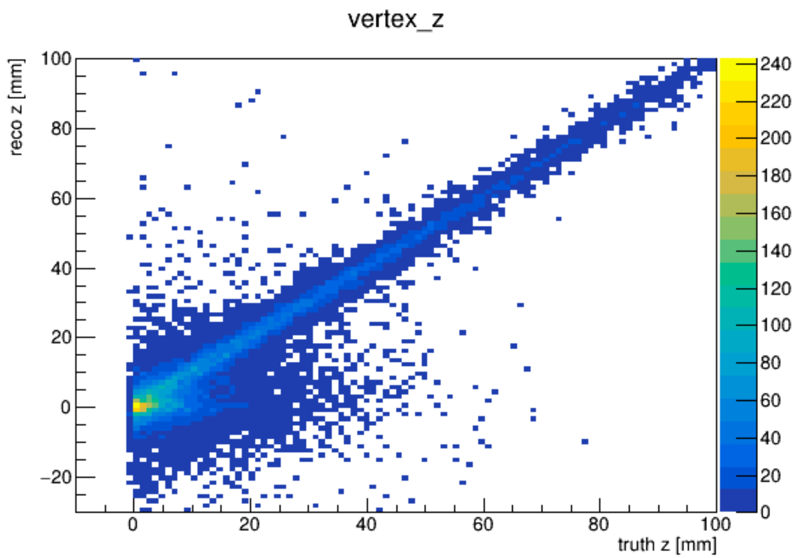
The **3-track** reconstruction efficiency comparison

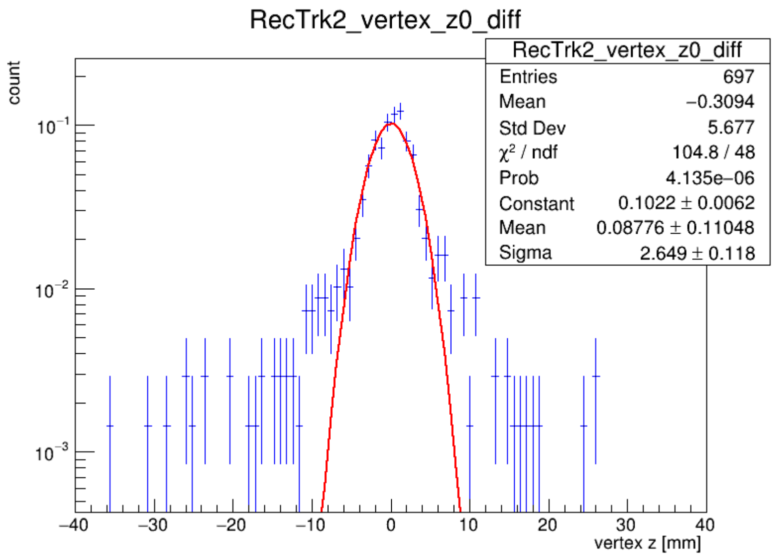
$m_{A'}$ [MeV]	20	50	100	200	500
Greedy + circle fit	60%	75%	77%	75%	69%
GNN + Dijkstra	82%	85%	85%	85%	84%

Vertexing

- Implemented based on rave (https://rave.hepforge.org/)
  - Use GenFit track as input (in the preliminary study, we use truth track finding result).
  - Several vertexing methods (in Rave):
    - Kalman filter
    - Adaptive Vertex Fitter



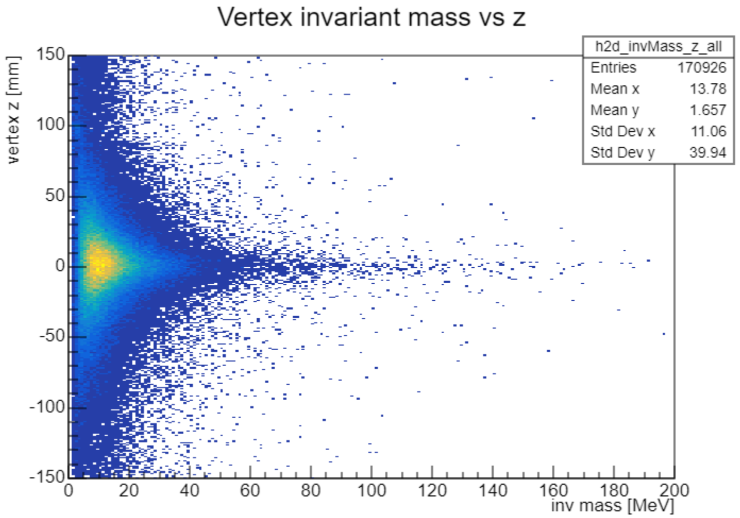




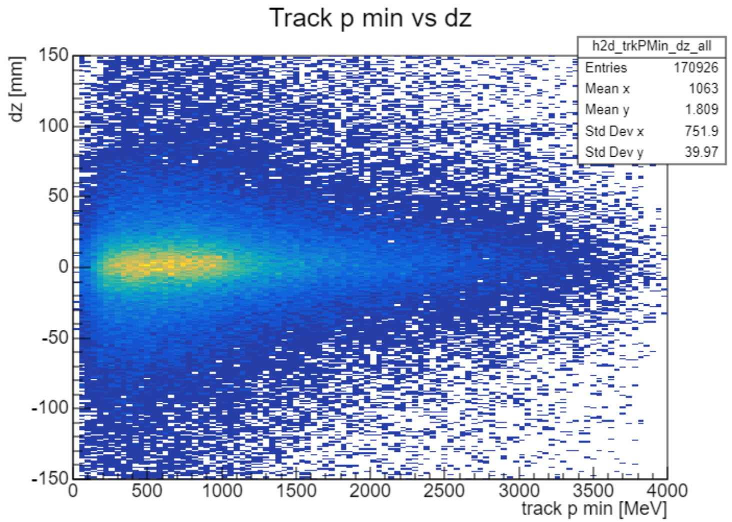
Signal vertex resolution: **2.6 mm**

Vertexing

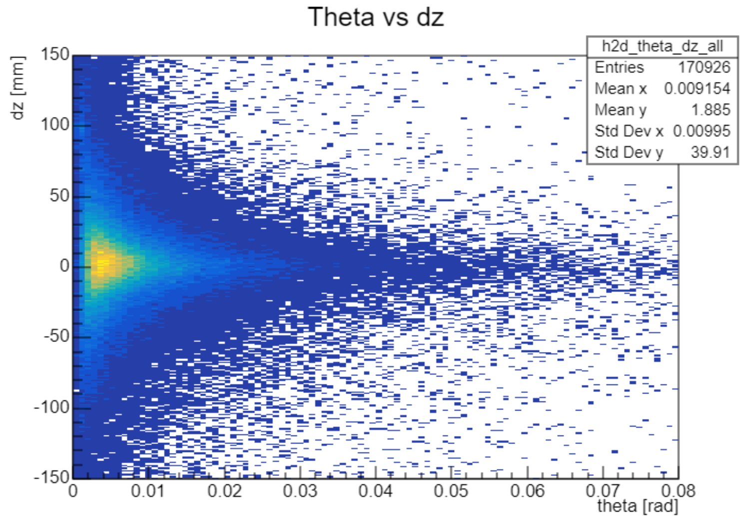
- We select several vertexing parameters to improve the vertex resolution:



Invariant mass



Track momentum



Track intersection angle

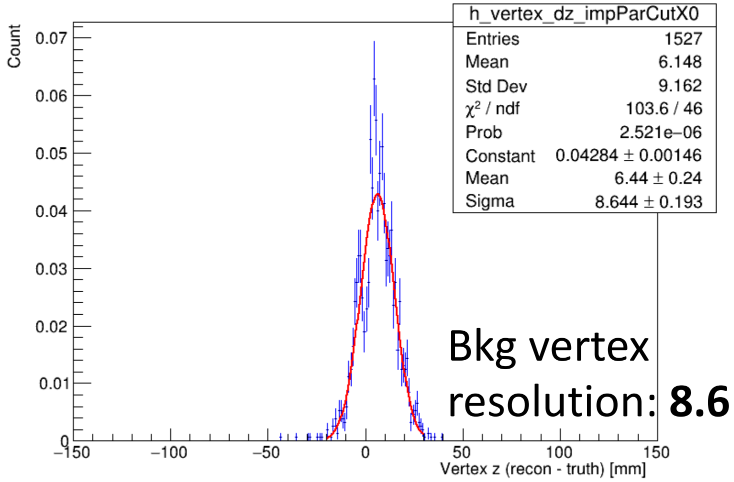
- The vertex z resolution is highly dependent on the angle, the momentum of tracks (multiple scattering effect).
- We add cut on these variables to improve the vertex z resolution.

Vertexing Application: Visible Decay Search

- We study the vertex resolution on the main background of visible decay.

Cut summary	
Kinematics cut	Track p > 0.5 GeV
	Invariant mass > 20 MeV
Reconstruction track cut	Shared hit num = 0
	Impact parameter cut
Vertex parameter cut	Recon vertex num >= 1
	Vertex dispersion < 0.4 mm
	Vertex theta > 0.012 rad
	Vertex projection – IP distance < 0.3 mm

Vertexing result of $e^-N \to e^-Ne^+e^-$ process:



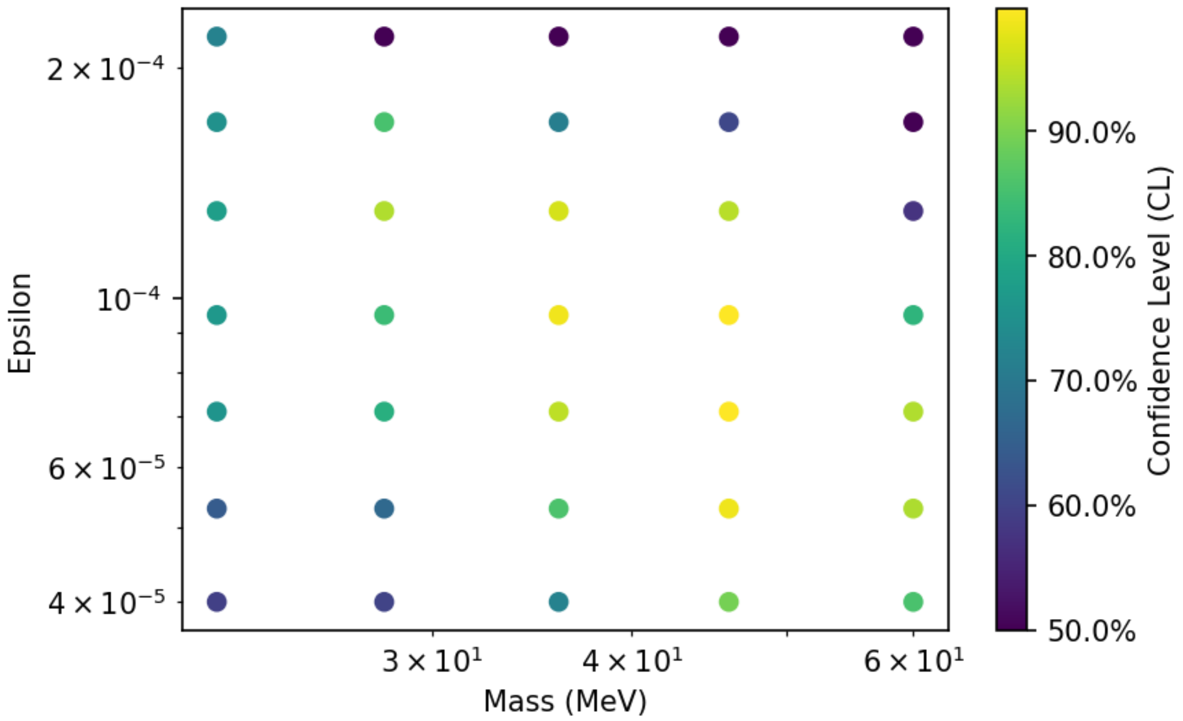
Bkg vertex resolution: **8.6 mm**

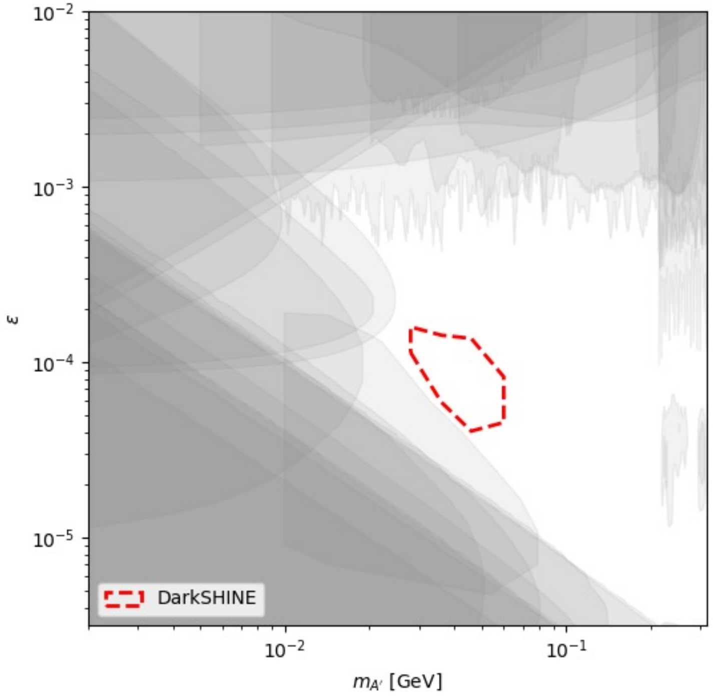
EOT: 3E14

Event type	**eBrem + conv**	**ePairProd**
Event num	7.5E12	4.3E11
Event num (after cut)	2.2E10	1.2E9
0.1 background range	**> 6.8 sigma (65mm, 99mm)**	

Exclusion Limit

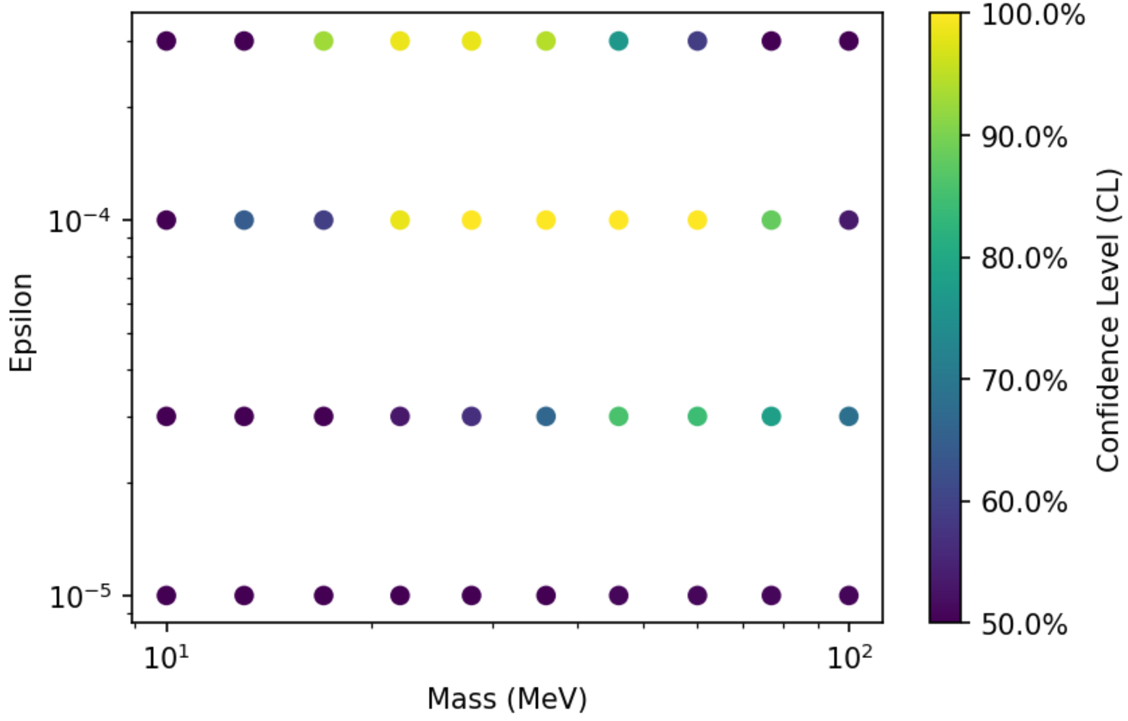
- From the vertex reconstruction result of $e^- N \to e^- N e^+ e^-$ process, assuming the reconstructed vertex is Gaussian distribution, we define the displaced vertex region at (65, 99) mm from the target.
- The exclusion limit for DarkSHINE (1 year run, 3E14 EOT) is estimated.

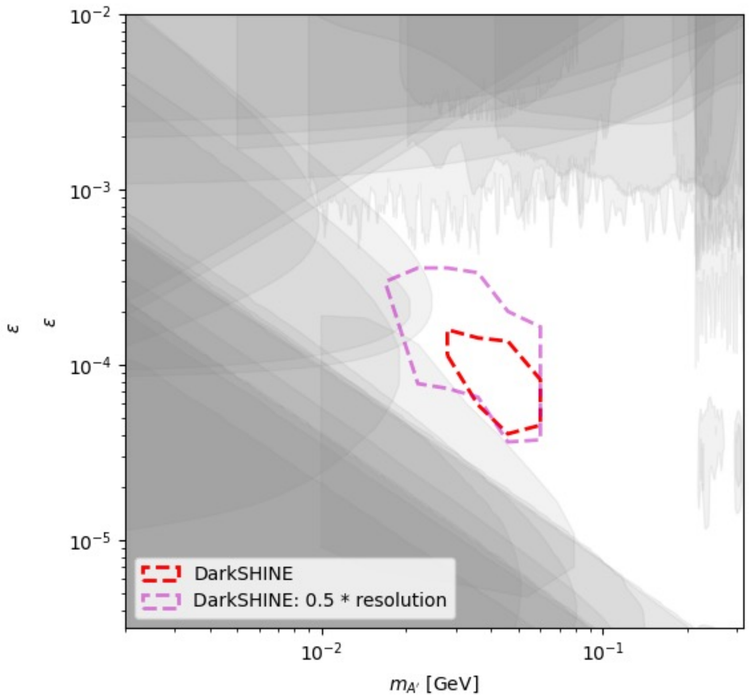




Exclusion Limit

- If the vertex resolution can be **improved by a factor of 2** (4.3mm), the signal region can be further extended.





Summary

- We have developed the full tracking and vertexing framework at DarkSHINE.
- **Tracking**
  - A simple track finding algorithm (Greedy find + circle fit) is implemented with good tracking efficiency **(~97%)** in single track case.
  - Track fitting algorithm with Kalman Filter based on GenFit is implemented.
  - GNN track finding is developed to improved tracking efficiency in 3-track case **(from ~70% to ~85%)**.
- **Vertexing**
  - Vertex reconstruction based on GenFit and Rave has been developed with mm-level resolution on vertex Z.
  - DarkSHINE can reach state-of-art sensitivity for dark photon visible decay search using improved vertex reconstruction.
    - $m_{A'}$ **(30 MeV, 60 MeV)**
    - ϵ **(4e-5, 2e-4)**

上海交通大學

SHANGHAI JIAO TONG UNIVERSITY

Thank You

饮 水 思 源　爱 国 荣 校