OML: an online multi-particle locating method for high-resolution single-event effects studies

Yan-Hao Jia^{1,2,3} · Jian-Wei Liao^{1,2,3} · Hai-Bo Yang^{1,2,3} · Qi-Hao Duan^{1,3} · Long-Jie Wang³ · Jiang-Yong Du^{1,4} · Hong-Lin Zhang^{1,2,3} · Cheng-Xin Zhao^{1,2,3}

Received: 23 December 2023 / Revised: 5 March 2024 / Accepted: 7 March 2024 / Published online: 10 October 2024 © The Author(s), under exclusive licence to China Science Publishing & Media Ltd. (Science Press), Shanghai Institute of Applied Physics, the Chinese Academy of Sciences, Chinese Nuclear Society 2024

Abstract

Identifying sensitive areas in integrated circuits susceptible to single-event effects (SEE) is crucial for improving radiation hardness. This study presents an online multi-track location (OML) framework to enhance the high-resolution online trajectory detection for the Hi'Beam-SEE system, which aims to localize SEE-sensitive positions on the IC at the micrometer scale and in real time. We employed a reparameterization method to accelerate the inference speed, merging the branches of the backbone of the location in the deployment scenario. Additionally, we designed an irregular convolution kernel, an attention mechanism, and a fused loss function to improve the positioning accuracy. OML demonstrates exceptional real-time processing capabilities, achieving a positioning accuracy of 1.83 µm in processing data generated by the Hi'Beam-SEE system at 163 frames per second per GPU.

Keywords Single-event effects · Integrated circuits · Silicon pixel Sensors · Artificial intelligence · Gaseous detector

1 Introduction

Integrating multiple integrated-circuit devices into spacecraft components poses a significant challenge due to the intense radiation environment in space [1]. These devices are highly susceptible to bombardment by high-energy particles. The ionizing effect of these particles leads to the generation

This work was supported by the National Natural Science Foundation of China(Nos. U2032209, 12222512, 12375193, 12305210), the National Key Research and Development Program of China (No. 2021YFA1601300), the CAS "Light of West China" Program, the CAS Pioneer Hundred Talent Program, the Guangdong Major Project of Basic and Applied Basic Research (No. 2020B0301030008).

Cheng-Xin Zhao chengxin.zhao@impcas.ac.cn

- ¹ Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou 730000, China
- ² Advanced Energy Science and Technology Guangdong Laboratory, Huizhou 516003, China
- ³ University of Chinese Academy of Sciences, Beijing 100049, China
- ⁴ Northwest Normal University, Lanzhou 730070, China

of numerous electron-hole pairs. When these charges accumulate in the sensitive regions of integrated-circuit devices, they result in abnormal circuit behavior or failure [2]. Malfunctions induced by charge deposition by a single particle are known as single-event effects (SEEs). Depending on the underlying mechanisms, SEE can be categorized into singleevent burnout (SEB), single-event gate rupture (SEGR), single-event upset (SEU), and single-event latchup (SEL) [3, 4].

To ensure the stable operation of integrated circuits in space, conducting SEE testing is imperative [5, 6]. Ground experiments using heavy-ion accelerators are among the most critical methods for evaluating SEE [7]. This approach involves irradiating circuits or chips with different energies and types of heavy-ion beams generated by a heavy-ion accelerator, inducing SEE, and obtaining crucial parameters such as the threshold linear energy transfer (LET) and cross section. The Heavy Ion Research Facility in Lanzhou (HIRFL) is currently China's largest and most diverse heavyion research facility in terms of ion species, and the highestenergy [8–11]. The SEE experimental terminal at the HIRFL expands the beam to a specific size, thereby obtaining a uniform distribution of ions within a particular irradiation area. These tests were used to study the average SEE parameters in the irradiated area. However, the radiation-resistant



integrated circuit industry aims to accurately determine the area sensitive to SEE during testing, enabling accurate radiation hardening and reducing the development cycle. The heavy-ion microbeam terminal restricts the accelerator beam to the micrometer scale, enabling scanning of specified regions with extremely high spatial resolution [12]. Therefore, precise localization of radiation-sensitive units in integrated circuits is possible, facilitating more effective radiation hardening and shortening the iteration process. However, the study of SEE precise positioning of an entire integrated circuit with a heavy-ion microbeam has been conducted by moving the irradiated area from point to point [13–15]. Consequently, extensive tests on integrated circuits require significant amounts of time and have become unrealistic [16–18].

Therefore, Hi' Beam-SEE, a precise SEE positioning system based on a charge-collecting pixel sensor for the SEE experimental terminal at the HIRFL, has been proposed [19–21]. Figure 1 illustrates the general structure of the Hi'Beam-SEE. This system enables real-time tracking of the trajectory of each ion in the beam, thereby determining its position on the device under test (DUT). By recording the positions of the particle hits that cause the SEE, the sensitive locations of the SEE on the DUT can be identified. The core component of the Hi'Beam-SEE is a heavy-ion positioning system [22], which must accurately determine the position of each ion in the beam. The system consists of two mutually perpendicular detection units, each comprising a stable electric field provided by a cage and a readout anode with a charge-collecting pixel sensor. Electron-ion pairs are generated by the ionization of the gas when a heavy ion passes through a detection unit. Electrons drift toward the charge-collecting sensor of the anode under the influence of the electric field. The trajectory projection of every incident particle can be obtained by collecting electrons.



Record the Current Hit Position

Fig.1 (Color online) The entire structure of the Hi'Beam-SEE system

The hit position of each particle on the DUT can be fitted by combining the trajectory projections in the two directions. The pixel sensor in the first-generation Hi'Beam-SEE system was a Topmetal-M chip [23, 24]. In future, we intend to adopt an IMPix-S series chip. In addition to the Hi'Beam-SEE, similar principles have been applied in other HiBeam series, such as Hi'Beam-A for beam monitoring in ultra-high vacuum environments and Hi'Beam-T for heavy-ion physics experiment terminals [25].

When an SEE occurs, it is referred to as an event. The event rate was defined as the number of SEEs that occurred per second. For each event, we recorded the data from the sensors to extract the ion trajectory. The raw frame rate of the Hi'Beam-SEE is 2.5 KHz, generating 1.28 GB of data per second. Therefore, offline storage of such a large volume of data is impractical, necessitating the development of online algorithms for real-time extraction of heavy-ion positions. Assuming that SEE occurs on the device under test (DUT) with an extreme probability of 1/100, the online algorithm needs to process 100 frames per second (fps).

Neural networks [26], due to their distinctive inductive bias and global modeling capabilities, have demonstrated effectiveness in distinguishing features from the background. In physics research and applications, neural networks have proven proficient in pulse shape recognition, beam trajectory segmentation, lesion detection in CT imaging, and heavy-ion cancer treatment [27, 28]. Notable examples include successful utilization of the CNN method for CT reconstruction by propagating noise solely from a single projection, and [29]'s proposition of an end-to-end neural network for feature extraction, regression of beam trajectory paths through segmentation, and fitting. A neural network trained with an extensive dataset of high-quality data has significant potential for real-time trajectory localization, aligning well with the requirements of the Hi'Beam-SEE system. In this study, we designed and implemented an online multi-particle locating (OML) algorithm for single-event effect studies using Hi'Beam-SEE. Our contributions are as follows:

- We proposed our OML method to extract the position of each particle in Hi'Beam-SEE. The OML achieves a positioning accuracy of 1.83 μm and a processing speed of 163 fps on a single GPU.
- We built a beam trajectory dataset, which contains more than 3000 images, covering trajectories with various densities and locations.

The second section discusses potential options for trajectory positioning, including statistical methods, traditional computer vision, lane detection, and object detection methods. The third section provides a comprehensive introduction to OML's network structure and elucidates its principles in detail, including track location and track fitting. The fourth section discusses the of the dataset and presents experimental results. Finally, the fifth section summarizes our research findings and outlines directions for subsequent optimization.

2 Potential options

In pursuit of implementing online data processing for track locating, several methods are potential candidates. These include statistical analysis, computer vision, lane detection, and object detection methods. Each method is introduced separately, along with its respective advantages and limitations.

2.1 Statistical method

Statistical methods require offline data storage and utilize Gaussian fitting and the center-of-gravity method to locate and fit trajectories [30]. The center-of-gravity method involves summing all the pixel values in each column, thereby compressing one frame into one row, creating a onedimensional distribution of the frame's values. Subsequently, N-order Gaussian fitting is applied to this one-dimensional distribution to obtain μ and σ for each trajectory in every frame. For instance, single-frame data with four trajectories require a sum of four one-dimensional Gaussian distributions. The region of each track was manually determined based on the μ and σ of the Gaussian distribution. Then, the center of gravity of each row is calculated within the region of the trajectory, and the slope (k) and intercept (b)are extracted by fitting each row's center of gravity. Statistical methods do not rely on image features to analyze data and can directly identify the positions of trajectories from raw data. However, the fitting process in this method requires manual interaction. Furthermore, this method generates substantial offline data, which makes it inadequate for our requirements of online automated positioning.

2.2 Computer vision methods

Prior to the advent of artificial intelligence, the detection of track-like objects in an image relied on leveraging visual information. The core approach involved extracting visual cues through various image processing techniques, such as utilizing the HSI color model [31] and edge extraction algorithms [32, 33], where the color and shape of the track boundary are the keys to the process [34, 35]. Furthermore, post-processing methods, such as Markov random fields and conditional random fields, were employed to enhance the detection accuracy [36, 36]. With advancements in machine learning, several methodologies have been proposed that utilize algorithms, such as template matching and support vector machines [38, 39]. For instance, SCNN [40] introduced a specialized convolution operation in the segmentation model to utilize features more efficiently. Other studies have focused on developing lightweight methods suitable for realtime applications [41–43]. However, these methods rely on image quality and encounter difficulties when processing images with strong background noise.

2.3 Lane detection methods

The reconstructed beam tracks closely resemble lane lines. Hence, lane detection methods are good candidates for trajectory location after adjusting part of the network structure according to the beam track features. The main algorithm design concepts are as follows: Beam trajectory detection in the segmentation problem can be abstracted by classifying $N \times M$ pixels in the image. However, this approach is extremely slow and requires an additional segmentation head to the backend of the network. [44] converted this problem into a row-wise detection problem in which detecting the presence of beam trajectory features in each row is equivalent to classifying N rows with M dimensions, allowing the network to explicitly establish relationships between different rows. This approach bridges the original semantic gap caused by low-level pixel-level modeling and high-level trajectory long-line structures and substantially simplifying the $N \times M$ classification problem. Another idea is based on the work of high-level semantic information in [45] to guide low-level semantic information. The network can realize detection more accurately by fusing beam trajectories, such as long bars, color highlighting after visualization, slope uniformity, and other features of different levels. Traditional lane detection methods usually define two lane lines with a set of prior points and then recognize lane lines with similar shapes and locations. In addition, these methods typically have difficulty in processing multiple trajectories simultaneously.

2.4 Object detection

The task of object detection aims to identify all targets (objects) of interest in an image and determine their categories and locations. Object detection involves three main methods: proposal-based, anchor-based, and anchor-free methods. The proposal-based method searches all generated regions to determine whether any region includes an object [46]. This method can achieve high accuracy but requires significant time. An anchor-based method was used to match a predefined set of anchors [47]. Due to the different sizes of the anchors, the network can recognize objects of different sizes. The anchor-based method is faster but requires a non-maximum suppression process for post-processing. The anchor-free method is similar to a segmentation method. It outputs feature maps of different sizes to determine whether an object is included in the feature map [48]. However, this method still requires post-processing and has lower accuracy compared to the proposal-based method. High robustness is required for the object detection method model to achieve high speed and accuracy.

2.5 Comparison and discussion

- The trajectories generally exhibit elongated shapes with the width being due to charge diffusion during the drift process. In addition, they are susceptible to irregular background noise, which appears as discrete energy points at considerable distances with similar intensity to the trajectories. Therefore, directly using the computer vision method to determine the continuity conditions makes it challenging to locate and return the overall beam trajectory.
- The number of trajectories in a single frame is uncertain; each frame may contain ten or more trajectories. Lane detection methods, influenced by predefined prior lane, cannot simultaneously locate all trajectories.
- The beam trajectory detection task requires high speed and positioning accuracy with online processing. Therefore, we cannot use the statistical method that requires manual interaction, the anchor-free method with low accuracy, or the proposal-based method with slow inference speed. We need to design a new network with an efficient structure that can be accelerated on different platforms to achieve the speed and accuracy requirements.
- FML [49] is previous work from our group designed for • the previous Hi'Beam-SEE [19], from which the real trajectory feature is that a single frame contains one or a few (typically no more than 5) large-angle trajectories. Because of the track features with large inclination angles, we designed a predefined slot structure for FML based on the lane detection method to improve its processing speed. In the upgraded Hi'Beam-SEE in this work, the data feature is that a single frame contains multiple tracks (up to more than ten) with small angles. We conducted experiments and analyzed the results, which revealed that it is difficult for FML training to converge when the number of tracks increases, especially when multiple adjacent tracks can be easily misclassified as a single track, resulting in a high miss detection rate. In addition, the slot-structure design of FML grids the frame to improve the processing speed (a grid contains multiple pixels, and the network first determines whether the grid includes a track and then determines the pixels in each grid that may belong to a track). However, because the coordinates of the tracks in the column direction are relatively close, it is easy to match local tracks from different rows on the same

column. Therefore, an upgraded Hi'Beam-SEE requires a new method to process multiple small-angle tracks simultaneously.

Hence, the online multi-particle locating (OML) algorithm for extracting the positions of particles is designed based on anchor-based object detection method.

3 OML's architecture

3.1 The backbone

The input image size is 128×512×3 (128 and 512 are the rows and columns of the image, respectively, and 3 is the RGB color channel of the image) and is computed in the first set of blocks in each stage by undergoing a set of convolutions with a stride of 2. By observing network structures, including ResNet152 [50], Swin Transformer [51], and ConvNeXT [52], we found that the addition of convolution operations in the third and fourth stages can effectively improve the network's ability to generalize the data. This is because the expression of high-dimensional features in images is more critical than the expression of low-dimensional features. Therefore, we applied this relevant experience and repeated the convolution operation for 4, 6, and 16 times in stages 2, 3, and 4, respectively. After each stage, the size of the feature map is halved, and the dimensions of the feature map are doubled. Table 1 demonstrates the architectural specification of the backbone.

The upper right corner of Fig. 2 shows the backbone structure. The structures of the training, inference, and merge-and-infer phases are shown from left to right. The module structure in the training phase has three branches: the middle branch (main branch) uses a convolution kernel of size 3×3 , batch normalization (BN), and a rectified linear unit (ReLu) function, and the first 3×3 convolution layers of each module use a step size of 2. The bypass branch uses a convolution kernel of size 1×1 and BN. The other bypass branch used a separate BN operation. The main branch sums the results of the two bypass branches in dimensions after the second convolution and BN operations (only the channel is changed, not the dimensions).

The BN layer accelerates the convergence speed of the model by normalizing the data and reducing overfitting, gradient explosion, and gradient disappearance. The principle is illustrated in Eq. (1). μ_i , σ_i are the mean and variance of the channel *i*, respectively. *e* is a minimum parameter set to prevent σ_i from being zero, and γ_i , β_i are used to adjust the variance and mean value of the data distribution in each batch.



Fig. 2 The structure of the anchor-box-based object detection approach is as follows: The backbone part uses a parameterizable VGG-style sub-module. The neck part obtains information from different backbone stages and combines it. In the final stage, high-level features are superimposed on low-level features to improve the per-

formance. In the head part, the feature map containing real beam trajectories is first cropped, and after strip convolution, it is combined with the resized scene. Finally, the coordinates and shapes of the particle trajectories are regressed, and the loss is calculated

 Table 1 (Color online) Architectural specification of the backbone

-			
Stage	Input size	Output size	Dimension of stage
1	128×512	64×256	1×64
2	64×256	32×128	4×128
3	32×128	16×64	6×256
4(optional)	16×64	8×32	16×512

Here 4×128 means stage-2 has 4 layers each with 128 channels

$$y_i = \frac{x_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}} \cdot \gamma_i + \beta_i \tag{1}$$

When the training phase is completed, μ , σ , γ , β become constant. The BN layer can then be integrated into the weight update process of the convolution kernel, and the final weight formula after fusion is

$$W'_{i,:,:,:} = \frac{\gamma_i}{\sigma_i} W_{i,:,:,:}, \quad b'_i = \beta_i - \frac{\mu_i \gamma_i}{\sigma_i}, \tag{2}$$

where W and B are the convolution kernel weights and bias after fusion, respectively. Therefore, the calculation formula after integrating the convolution, BN, and ReLu layers was as follows:

$$y_i = \text{Max}(W'_{i,..,i}x_i + b'_i, 0).$$
 (3)

The Conv1×1+BN branch can be regarded as a convolution BN fusion layer with a core size of 3×3 (only the center point has a 1×1 convolution kernel weight, and the eight surrounding points are 0). The BN branch can be regarded as a 3×3 convolution BN fusion layer with a weight of one. The process in Eq. (3) and Fig. 3 is called reparameterization.

After reparameterization, we obtain the module structure in the inference phase, which improves the speed of inference by removing the bypass branches and maintaining the backbone. Subsequently, we integrate the different layer weights obtained from their training into the fusion layer. Therefore, the final fusion layer formula is

$$y_i = \text{Max}(W''_{i,:,:,:}x_i + b''_i, 0),$$
(4)

$$W''_{i,:,::} = W'_{i,:,::} + W^{\text{Conv+BN}}_{i,:,::} + W^{\text{BN}}_{i,:,::},$$
(5)

$$b_i'' = b_i' + b_i^{\text{Conv+BN}} + b_i^{\text{BN}}, \tag{6}$$

Through reparameterization and layer fusion, we have integrated the nine-layer module structure into three layers, resulting in an accelerated structure for the merge-and-infer



Fig. 3 (Color online) After unifying the network size of the bypass branch to be consistent with the main branch, add their weights. This operation can remove the bypass branches and connection layers and unify the scale of the convolution layer while retaining the data characteristics learned by multiple bypass branches

phase. This structure can significantly speed up network operations and effectively reduce hardware overhead.

3.2 The neck

The feature pyramid network (FPN) [53] has become the standard neck model because it integrates seamlessly with backbones such as ResNet, and can fuse different levels of features, making it applicable to various application areas such as target detection and segmentation. Utilizing small- or medium-sized backbone networks can reduce computational costs and improve efficiency. However, shallow backbones typically provide fewer rich semantic features for object detection. Most SOTA (state-of-the-art) methods use deep CNN structures to improve the accuracy of irregular object detection but this approach is not suitable for Hi'Beam-SEE. Because the last-stage's feature map size of our beam data is 16×64 , the trajectories in the images are compressed into features with only a few pixels in the last layer of the feature pyramid, which is insufficient for accurate trajectory recognition. Considering that the features of particle trajectories in Hi'Beam-SEE are different from those of ordinary target detection objects, it is vital for the neck to extract the accurate features of the particle beam trajectories with long shapes and a high-noise background, allowing the head to efficiently use the features. Additionally, the detection of irregular objects requires high-level semantics to distinguish objects in the background from low- or medium-level features for accurate object localization. For this reason, we designed a new neck structure called feature fusion component (FFC) to create a feature map with less feature loss when transforming between different scales, and to retain more detailed and integrated features. As shown in Fig. 4, FFC obtains original information from different stages in the backbone. The features of the middle stage are extracted as the main features of the DW-Conv module. The larger the size of the convolution kernel, the larger the receptive field, especially on the particle beam trajectory dataset. High- and low-level features were aligned in size using bilinear difference and regular convolution, respectively. The features are superimposed and can contain information of different scales after a 1×1 convolutional fusion layer.



Fig. 4 FFC detailed structure

3.3 Head part

The head requires features extracted from the backbone and neck for different detection and segmentation tasks. As the particle trajectories have various shapes, colors, locations, and degrees of association with the background, compared with the traditional model, the loss function must be redesigned to improve the object localization accuracy. Referring to the concepts of conditional convolution and prior LaneNet works, we use position prior knowledge to aid in detecting trajectories and propose a composite loss function consisting of multiple parts, which can further improve the detection accuracy of the network.

3.3.1 ROI extractor

The trajectories respond more strongly to high-level features, whereas low-level features relate more to the interaction information with background noise and edge information. Therefore, it is important to allow the network to recognize trajectories as a whole by simultaneously using different stage features. In addition, it is difficult to use only a detection head and a single-stage feature for trajectory location. In addition, more comprehensive and finer features can help improve the localization accuracy. Therefore, our aim is to utilize Conv's gradually increasing receptive field and gradually decreasing feature hierarchy, which has a semantic hierarchy from low to high, and build a fused feature pyramid network with different levels of semantic information.

By clustering the particle trajectory data in the dataset, we can assign trajectory priors to each feature map and make the network more interested in the regions where trajectories are likely to exist (ROI). However, additional contextual information regarding these features is required. In corner cases, the particle trajectories may be incomplete. Therefore, the network lacks local semantic information to determine whether a trajectory exists and whether the background noise and trajectory are misclassified. Hence, determining whether a pixel belongs to a trajectory must be aided by pixels surrounding the trajectory.

This was also demonstrated in non-local experiments, showing that the adequate utilization of long-range dependencies can improve performance. Therefore, we incorporated more background noise to interact with the trajectories, enabling the network to distinguish them better. Through convolution operations on the ROI, further connections can be made between the pixels of the trajectory and possibly reconstruct incomplete areas. The enhanced ROI was then subjected to an attention operation with the original feature map to establish a mapping relationship between the ROI and the background. This approach further exploits background noise, resulting in a more robust model.



Fig. 5 ROI extractor detailed structure



Fig. 6 (Color online) IoU loss calculation method

Figure 5 presents the detailed structure of the ROI Extractor. To address the difference between the slender shapes of the trajectories and the rectangular detection box in classical object detection methods, we used bar-shaped convolution for feature extraction. We found that a convolution kernel with a kernel size of 13 works best, and we deployed these convolution kernels in both row and column directions within the ROI Extractor.

3.3.2 Loss function

Conventional loss function designs typically involve classification and regression components. However, our findings reveal that this approach does not enable the network to describe trajectories precisely through multiple backpropagation iterations. This limitation arises from the use of the conventional L1 loss function, which calculates the regression loss by dispersing the points during computation instead of considering them as a cohesive unit. Consequently, the regression accuracy failed to satisfy the experimental requirements of Hi'Beam-SEE, particularly in scenarios with high-noise backgrounds. To address this issue, we introduce a composite loss function consisting of three components: IoU (Intersection over Union) loss, cls (classification) loss, and xykb (start position with (x, y), slope, and intercept) loss.

IoU loss is the most important aspect for predicting and regressing a trajectory as a whole. As shown in Fig. 6, the blue line is the set of real trajectory line coordinates, and the red line is the set of network-predicted trajectory line coordinates. First, we consider a horizontal coordinate x_i , where the corresponding real coordinate is x_i^G the predicted coordinate is x_i^P , and its horizontal deviation is d_i . If a small range of perturbations is added to x_i^G and x_i^P as e, the IoU of point x_i is calculated as

$$IoU_{x_i} = \frac{d_i^{Offset}}{d_i^{Horizontal}}$$

$$= \frac{\min(x_i^{P}, x_i^{G}) - \max(x_i^{P}, x_i^{G})}{\max(x_i^{P} + e, x_i^{G} + e) - \min(x_i^{P} - e, x_i^{G} - e)}$$
(7)

If the trajectory line is discretized into a set of *N* points, the IoU loss is

$$IoU_{Line} = 1 - \sum_{i=1}^{N} IoU_{x_i} = 1 - \frac{\sum_{i=1}^{N} d_i^{Offset}}{\sum_{i=1}^{N} d_i^{Horizontal}}$$
(8)

To better detect the trajectory using prior knowledge, we also need xykb loss to predict the coordinates, slope, and intercept of the starting point of each trajectory. In addition, we added focal loss for the classification of different types of particle trajectories. The total loss formula is as follows:

$$Loss_{Total} = Loss_{IoU} + Loss_{xykb} + Loss_{focal}$$
(9)

3.4 Track fitting

The detected trajectory bounding boxes are represented in the form of vertex coordinates. Subsequently, the center of gravity of each row was calculated along the trajectory within the bounding box. Least-squares fitting was then performed on all the rows' centers of gravity, yielding the corresponding trajectory slope and intercept. The fitting accuracy of each row was assessed by calculating the standard deviation ΔP between the center of gravity of each row and the fitted point in each row. Then, the positioning accuracy of each trajectory was calculated using Eq. (11), where PA



Fig.7 (Color online) The visualization of the track fitting process. The red line represents the ground truth for each row, and the blue line represents the fitted trajectory

is the positioning accuracy of the trajectory, $\overline{\Delta P}$ is the mean of the standard deviation of all the rows in the bounding box. The visualization results are presented in Fig. 7.

$$y_{\exp} = k \times x_m + b \tag{10}$$

$$PA = \sqrt{\frac{1}{n(n-1)} \sum_{i=1}^{n} (\Delta P_i - \overline{\Delta P})}$$
(11)

4 Experiments and results

The process of creating a dataset and visualizing the detection results is presented in this section. CLRNet was selected as the comparison detection method and uses the yolov7base version as the baseline for comparison [54].

4.1 Dataset

The raw data contained either the heavy-ion trajectories collected at HIRFL or the laser trajectories collected in the laboratories at low density. These two trajectory types exhibit different image features, necessitating two separate categories for labeling. The OML must effectively handle various particle flux densities to ensure that the Hi'Beam-SEE fits different test terminals. Therefore, images covering various cases were generated based on the raw data to build the dataset.

The dataset image generation involves the following steps. First, background noise reduction is performed on the raw data using a 9×9 convolutional kernel. Next, particle tracks extracted from preprocessed raw data were merged to generate dataset images covering trajectories with varying types, numbers, and spacing. Also, part of the preprocessed images containing a single trajectory was directly included in the dataset. Finally, Gaussian noise was randomly injected into the image to simulate the noise that may have arisen from different experimental terminals. The final dataset contained more than 3000 images. Figure 8 shows example images of the 12 cases covered by the final dataset: images with low-, medium-, or high-density heavy-ion trajectories or laser trajectories; images with mixed-type trajectories at different densities. The dataset was then split into training and test parts in a ratio of 8:2. We tested the experimental performance of our model using the test set.

4.2 Comparison experiments and ablation study

This subsection compares the performance of CLRNet, yolov7-base, the center-of-gravity method, and OML.

Fig. 8 Visualization the example cases. a Image with single laser trajectory. b Image with single heavy-ion trajectory. c Image with mediumdensity laser trajectories with different locations. d Image with medium-density heavyion trajectories with different locations. e. Image with high-density laser trajectories at different locations. f Image with high-density heavy-ion trajectories with different locations. g Image with low-density mixed heavy-ion and laser trajectories. h Image with medium-density mixed heavy-ion and laser trajectories. i. Image with highdensity mixed heavy-ion and laser trajectories. j. Image with high-density mixed heavy-ion and laser trajectories



Table 2 Quantitative comparison of beam trajectory detection

Method	Miss detection rate (%)	False detection rate (%)	Speed (fps)	Positioning accuracy (µ m)
CLRNet	41.7	4.4	270	1.54
Yolov7-base	6.3	4.5	128	2.89
Center-of- Gravity Method	0	0	Offline	1.40
OML	0.4	0	163	1.83

Additionally, ablation experiments for each component of OML were conducted separately to verify the effectiveness of the design. The performance metrics included the miss detection rate, false detection rate, speed, and positioning accuracy. The miss detection rate is defined as the proportion of undetected trajectories to the total number of trajectories; the false detection rate is the proportion of incorrectly detected trajectories to the total number of detected trajectories; the speed is the number of frames containing trajectories per second that OML can process; and the positioning accuracy is the average value of all trajectories in the test dataset.

According to the results presented in Table 2, our OML demonstrates good performance in terms of miss detection and false detection rates, maintaining high processing speed and positioning accuracy. Notably, the yolov7-base model

struggles, particularly in non-vertical trajectory scenarios, primarily because of its object detection principle's inability to accurately detect bounding boxes with rotation angles, resulting in increased IoU loss. However, CLRNet satisfactorily fulfills the detection requirements in scenarios with small tilt angles and provides a fast processing speed. However, CLRNet faces difficulties in dealing with multiple trajectories, which leads to a high missing detection rate. The center-of-gravity method cannot locate trajectories online, although it has good sounding performance.

In the ablation study, yolov7-base was utilized as the baseline model, gradually replacing its components with the corresponding parts of OML. Table 3 lists the results of this study. The results indicate that after reparameterization, the backbone component exhibits significant improvements in positioning accuracy while reducing the parameter size, thereby achieving a higher model inference speed of 188 fps. However, the obtained accuracy still fails to meet the demands of real-time detection. Substituting the FFC neck, which enables the fusion of different stage features, and the ROI head, which focuses on the entire trajectory, further improves accuracy. Finally, the overall model was constructed by optimizing the loss function, striking a balance between detection accuracy and speed that aligned with our expectations.

Figure 9 shows the visualized final detection results obtained using the OML, corresponding to the example images in Fig. 7. The results demonstrate the capability of

Fig. 9 Visualization of OML test results



Table 3	Ablation	study	of	OML
---------	----------	-------	----	-----

Backbone	Neck	Head	Loss	Miss detection rate (%)	False detection rate (%)	Speed (fps)	Positioning accuracy (µ m)
ResNet50	FPN	yolov7 head	L1	6.3	4.5	128	2.89
Fusion(Ours)	FPN	yolov7 head	L1	6.1	4.7	188	2.58
Fusion(Ours)	FFC(Ours)	yolov7 head	L1	4.6	3.5	177	2.36
Fusion(Ours)	FFC(Ours)	ROI(Ours)	L1	2.8	1.9	169	2.16
Fusion(Ours)	FFC(Ours)	ROI(Ours)	Loss_Total(Ours)	0.4	0	163	1.83

OML to accurately detect both heavy ion and laser trajectories, even in the presence of high background noise. The model exhibited exceptional robustness when faced with trajectories of different angles and densities. However, it is worth noting that OML still encounters certain challenges, as illustrated by the heavy-ion trajectories on the right side of Fig. 9f. The excessive intensity of background noise poses difficulties during data preprocessing, making it challenging for OML to locate the center of the trajectory precisely, resulting in low positioning accuracy. However, such corner cases occur with a relatively low probability. Overall, OML satisfies the detection requirements of the Hi'Beam-SEE.

5 Conclusion

To perform massive and high-resolution SEE studies on integrated circuits at HIRFL, the Hi'Beam-SEE, a precise SEE positioning system based on charge-collecting pixel sensors, was designed. The OML (online multi-particle locating) algorithm in Hi'Beam-SEE extracts the positions of the particle hits on the integrated circuits in real-time, avoiding the storage of large amounts of data. The OML consists of track-locating and track-fitting parts. The track location part targets the trajectories in the frames. It consists of a reparameterization backbone network, an FFC neck capable of aggregating features at different levels, an ROI head focusing on overall trajectory information, and a novel composite loss function to adapt to various densities and particle trajectory types. The track-fitting component extracts the position of each trajectory using the center-ofgravity method. The OML is trained and tested on a dataset with actual and fused frames. The evaluation results show that the OML achieves a miss detection rate of only 0.4% and no false detection rate, with a processing speed of 163 fps and a positioning accuracy of 1.83 µm, demonstrating the effectiveness of OML in meeting the real-time SEE detection requirements of HIRFL. The OML demonstrates excellent potential for real-time track processing in gaseous detectors, such as the Time Projection Chamber, and it will continue to be optimized for more scenarios in future.

Author contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Yan-Hao Jia, Jiang-Yong Du, Long-Jie Wang, Hong-Lin Zhang, and Jian-Wei Liao. The first draft of the manuscript was written by Yan-Hao Jia, Hai-Bo Yang, and Cheng-Xin Zhao, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Data availability The data that support the findings of this study are openly available in Science Data Bank at https://cstr.cn/31253.11. sciencedb.j00186.00178 and https://www.doi.org/10.57760/sciencedb.j00186.00178.

Declarations

Conflict of interest Cheng-Xin Zhao is an editorial board member for Nuclear Science and Techniques and was not involved in the editorial review, or the decision to publish this article. All authors declare that there are no conflict of interest.

References

- C. Zeitlin, Space radiation shielding, in *Handbook of Bioastro*nautics. ed. by L.R. Young, J.P. Sutton (Springer, Cham, 2021). https://doi.org/10.1007/978-3-319-12191-8_28
- R. Gaillard, Single event effects: mechanisms and classification, in Soft Errors in Modern Electronic Systems. Frontiers in Electronic Testing, vol. 41, ed. by M. Nicolaidis (Springer, Boston, 2011). https://doi.org/10.1007/978-1-4419-6993-4_2
- E. Normand, Single-event effects in avionics. IEEE Trans. Nucl. Sci. 43, 461–474 (1996). https://doi.org/10.1109/23.490893
- F.W. Sexton, Destructive single-event effects in semiconductor devices and ICs. IEEE Trans. Nucl. Sci. 50, 603–621 (2003). https://doi.org/10.1109/TNS.2003.813137
- W. Yang, X. Du, Y. Li et al., Single-event-effect propagation investigation on nanoscale system on chip by applying heavyion microbeam and event tree analysis. Nucl. Sci. Tech. 32, 106 (2021). https://doi.org/10.1007/s41365-021-00943-6
- J. Liu, Z. Zhou, D. Wang et al., Prototype of single-event effect localization system with CMOS pixel sensor. Nucl. Sci. Tech. 33, 136 (2022). https://doi.org/10.1007/s41365-022-01128-5

- G. Aad, T. Abajyan, B. Abbott et al., The ATLAS Experiment at the CERN Large Hadron Collider. JINST 3, S08003 (2008). https://doi.org/10.1088/1748-0221/3/08/S08003
- H. Wang, Z. Wang, C. Gao et al., Design and tests of the prototype beam monitor of the CSR external target experiment. Nucl. Sci. Tech. 33, 36 (2022). https://doi.org/10.1007/ s41365-022-01021-1
- C. Yuan, W. Zhang, T. Ma et al., Design and implementation of accelerator control monitoring system. Nucl. Sci. Tech. 34, 56 (2023). https://doi.org/10.1007/s41365-023-01209-z
- T. Liu, H. Song, Y. Yu et al., Toward real-time digital pulse process algorithms for CsI(Tl) detector array at external target facility in HIRFL-CSR. Nucl. Sci. Tech. 34, 131 (2023). https://doi.org/ 10.1007/s41365-023-01272-6
- B. Zhang, L. Liu, H. Pei et al., Classifier for centrality determination with zero-degree calorimeter at the cooling-storage-ring external-target experiment. Nucl. Sci. Tech. 34, 176 (2023). https://doi.org/10.1007/s41365-023-01338-5
- T. Liu, Z. Yang, J. Guo et al., Application of SEU imaging for analysis of device architecture using a 25 MeV/u Kr-86 ion microbeam at HIRFL. Nucl. Instrum. Meth. Phys. B 404, 254–258 (2017). https://doi.org/10.1016/j.nimb.2017.01.069
- B. Walasek-Hohne, Scintillating screen applications in accelerator beam diagnostics. IEEE Trans. Nucl. Sci. 59, 2307–2312 (2012). https://doi.org/10.1109/TNS.2012.2200696
- J. Bosser, J. Mann, G. Ferioli et al., Optical transition radiation proton beam profile monitor. Nucl. Instrum. Meth. Phys. A 238, 45–52 (1985). https://doi.org/10.1016/0168-9002(85)91025-3
- C. Gonzalez, F. Pedersen, An ultra low noise AC beam transformer for deceleration and diagnostics of low intensity beams. in *Proceedings of the 1999 Particle Accelerator Conference (Cat. No.99CH36366)*, (New York, NY, USA, 1999), pp. 474–476. https://doi.org/10.1109/PAC.1999.795736
- H. Weisberg, E. Gill, P. Ingrassia et al., An ionization profile monitor for the Brookhaven AGS. IEEE Trans. Nucl. Sci. 30, 2179–2181 (1983). https://doi.org/10.1109/TNS.1983.4332753
- C.D. Johnson, L. Thorndahl, The CPS gas-ionization beam scanner. IEEE Trans. Nucl. Sci. 16, 909–913 (1969). https://doi.org/ 10.1109/TNS.1969.4325399
- H. Xie, K. Gu, Y. Wei et al., A noninvasive Ionization Profile Monitor for transverse beam cooling and orbit oscillation study in HIRFL-CSR. Nucl. Sci. Tech. **31**, 40 (2020). https://doi.org/10. 1007/s41365-020-0743-7
- H. Yang, H. Zhang, C. Gao et al., Hi'Beam-S: a monolithic silicon pixel sensor-based prototype particle tracking system for HIAF. IEEE Trans. Nucl. Sci. 68, 2794–2800 (2021). https://doi.org/10. 1109/TNS.2021.3128542
- H. Zhang, Y. Zhang, H. Yang et al., Hi'Beam-A: a pixelated beam monitor for the accelerator of a heavy-ion therapy facility. IEEE Trans. Nucl. Sci. 68, 2081–2087 (2021). https://doi.org/10.1109/ TNS.2021.3085030
- Y. Zhang, H. Yang, H. Zhang et al., Hi'Beam-T: a TPC with pixel readout for heavy-ion beam monitoring. in *Poster Presented at* 23rd Virtual IEEE Real Time Conference, Aug 1-5, (2022). https:// indico.cern.ch/event/1109460/contributions/4893245/
- 22. J.W. Liao, X.L. Wei, H.B. Yang et al., A pixel sensor-based heavy ion positioning system for high-resolution single event effects studies. Nucl. Instrum. Meth. Phys. A **1065**, 169538 (2024). https://doi.org/10.1016/j.nima.2024.169538
- W. Ren, W. Zhou, B. You et al., Topmetal-M: a novel pixel sensor for compact tracking applications. Nucl. Instrum. Meth. A 981, 164557 (2020). https://doi.org/10.1016/j.nima.2020.164557
- H. Yang, F. Mai, J. Liao et al., Heavy-ion beam test of a monolithic silicon pixel sensor with a new 130 nm High-Resistivity CMOS process. Nucl. Instrum. Meth. Phys. A **1039**, 167049 (2022). https://doi.org/10.1016/j.nima.2022.167049

- R. He, X.Y. Niu, Y. Wang et al., Advances in nuclear detection and readout techniques. Nucl. Sci. Tech. 34, 205 (2023). https:// doi.org/10.1007/s41365-023-01359-0
- Y. Lecun, Y. Bengio, G. Hinton, Deep learning. Nature 521, 436 (2015). https://doi.org/10.1038/nature14539
- D. Guest, K. Cranmer, D. Whiteson, Deep learning and its application to LHC physics. Annu. Rev. Nucl. Part. Sci. 68, 161–181 (2018). https://doi.org/10.1146/annurev-nucl-101917-021019
- M. Hibat-Allah, Optimizing the synergy between physics and machine learning. Nat. Mach. Intell. 3, 925 (2021). https://doi. org/10.1038/s42256-021-00416-w
- P. Ai, D. Wang, X. Sun et al., A deep learning approach to multitrack location and orientation in gaseous drift chambers. Nucl. Instrum. Meth. Phys. A **984**, 164640 (2020). https://doi.org/10. 1016/j.nima.2020.164640
- Z. Li, Y. Fan, Z. Wang et al., A new method for directly locating single-event latchups using silicon pixel sensors in a gas detector. Nucl. Instrum. Meth. Phys. A 962, 163697 (2023). https://doi.org/ 10.1016/j.nima.2020.163697
- I. S. Choi, C. K. Cheong, A road lane detection algorithm using HSI color information and ROI-LB. in *Information & Control Symposium*, (2009)
- B. Yu, A.K. Jain, Lane boundary detection using a multiresolution Hough transform. in *Proceedings of International Conference on Image Processing*, (Santa Barbara, CA, USA, 1997), pp. 748-751. https://doi.org/10.1109/ICIP.1997.638604
- Y. Wang, D. Shen, E.K. Teoh, Lane detection using spline model. Pattern Recogn. Lett. 21, 677–689 (2000). https://doi.org/10.1016/ S0167-8655(00)00021-0
- J. Sui, Y. Ma, W. Yang et al., Diffusion enhancement for cloud removal in ultra-resolution remote sensing imagery. IEEE Trans. Geosci. Remote Sensing 62, 5405914 (2024). https://doi.org/10. 1109/TGRS.2024.3411671
- J. Sui, X. Ma, X. Zhang et al., GCRDN: global context-driven residual dense network for remote sensing image super-resolution. IEEE J. Sel. Top. Appl. 16, 4457–4468 (2023). https://doi.org/10. 1109/JSTARS.2023.3273081
- P. Krhenbühl, V. Koltun, Efficient inference in fully connected CRFs with Gaussian edge potentials. Curran Associates Inc. (2012). arXiv:1210.5644
- 37. K.C. Kluge and S. Lakshmanan, Lane boundary detection using deformable templates: Effects of image subsampling on detected lane edges. in *Recent Developments in Computer Vision. ACCV* 1995. Lecture Notes in Computer Science, ed. by S.Z. Li, D.P. Mital, E.K. Teoh, H. Wang, vol 1035. (Springer, Berlin, Heidelberg). https://doi.org/10.1007/3-540-60793-5_87
- S. Shajahan, B. Chithra, Structural morphology based automatic virus particle detection using robust segmentation and decision tree classification. in *International Conference On Innovations & Advances In Science, Engineering And Technology* [IC - IASET 2014]
- M. Banerjee, M.K. Kundu, P. Mitra, Corner detection using support vector machines. in *Proceedings of the 17th International Conference on Pattern Recognition*, 2004. ICPR 2004., (Cambridge, UK, 2004), pp. 819-822. https://doi.org/10.1109/ICPR. 2004.1334384
- X.G. Pan, J.P. Shi, P. Luo et al., Spatial as deep: Spatial CNN for traffic scene understanding. Proc. AAAI Conf. Artif. Intell. 32(1), 7276–7283 (2018). https://doi.org/10.1609/aaai.v32i1.12301
- H. Li, C. Jia, P. Jin et al., FreestyleRet: Retrieving images from style-diversified queries. (2023). https://doi.org/10.48550/arXiv. 2312.02428

- Y. Hou, Z. Ma, C. Liu et al., Learning lightweight lane detection CNNs by self attention distillation. in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), (Seoul, Korea (South), 2019), pp. 1013-1021. https://doi.org/10.1109/ICCV. 2019.00110
- 43. D. Neven, B. De. Brabandere, S. Georgoulis et al., Towards endto-end lane detection: an instance segmentation approach. in 2018 IEEE Intelligent Vehicles Symposium (IV), (Changshu, China, 2018), pp. 286-291 https://doi.org/10.1109/IVS.2018.8500547
- Z.Q. Qin, P.Y. Zhang, X. Li et al., Ultra fast deep lane detection with hybrid anchor driven ordinal classification. IEEE Trans. Pattern Anal. Mach. Intell. 46, 2555–2568 (2024). https://doi.org/10. 1109/TPAMI.2022.3182097
- 45. T. Zheng, Y.F. Huang, X. Li et al., CLRNet: cross layer refinement network for lane detection. in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (New Orleans, LA, USA, 2022), pp. 888-897. https://doi.org/10.1109/CVPR5 2688.2022.00097
- R. Girshick, Fast R-CNN. in 2015 IEEE International Conference on Computer Vision (ICCV), (Santiago, Chile, 2015), pp. 1440–1448. https://doi.org/10.1109/ICCV.2015.169
- J. Redmon, S. Divvala, R. Girshick et al., You only look once: unified, real-time object detection. in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (Las Vegas, NV, USA, 2016), pp. 779-788. https://doi.org/10.1109/CVPR. 2016.91
- Z. Tian, C. Shen, H. Chen et al., FCOS: Fully convolutional onestage object detection. in 2019 IEEE/CVF International Conference on Computer Vision (ICCV), (Seoul, Korea (South), 2019), pp. 9626-9635. https://doi.org/10.1109/ICCV.2019.00972
- 49. Y.X. Hu, H.B. Yang, H. Zhang et al., An online fast multi-track locating algorithm for high-resolution single-event effect test platform. Nucl. Sci. Tech. 34, 72 (2023). https://doi.org/10.1007/ s41365-023-01222-2
- K. He, X. Zhang, S. Ren et al., Deep residual learning for image recognition. in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (Las Vegas, NV, USA, 2016), pp. 770-778. https://doi.org/10.1109/CVPR.2016.90
- Z. Liu, Y. Lin, Y. Cao et al., Swin transformer: Hierarchical vision transformer using shifted windows. in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), (Montreal, QC, Canada, 2021), pp. 9992-10002. https://doi.org/10.1109/ICCV4 8922.2021.00986
- Z. Liu, H. Mao, C. Wu et al., A ConvNet for the 2020s. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, (2022), pp. 11966-11976. https://doi.org/10.1109/CVPR52688.2022.01167
- T. Lin, P. Dollar, B. Girshick et al., Feature pyramid networks for object detection. in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (Honolulu, HI, USA, 2017), pp. 936-944. https://doi.org/10.1109/CVPR.2017.106
- C. Wang, A. Bochkovskiy, H.Y.M. Liao, YOLOv7: Trainable bagof-freebies sets new state-of-the-art for real-time object detectors. in 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (Vancouver, BC, Canada, 2023), pp. 7464-7475. https://doi.org/10.1109/CVPR52729.2023.00721

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.