Occlusion Segmentation: Restore and Segment Invisible Areas for Particle Objects

Jinshi Liu, Zhaohui Jiang, *Member, IEEE*, Weihua Gui, *Member, IEEE*, Zhiwen Chen, *Member, IEEE*, and Chaobo Zhang, *Member, IEEE*

***Abstract*—The occlusion problem has consistently posed a significant challenge in the field of segmentation. Most existing segmentation methods require additional annotations and fail to capture the contour information of occluded regions, thus not truly addressing the occlusion issue. Although segmentation tasks involving particle objects also suffer from occlusion problems, the homogeneity of particle objects offers new possibilities for overcoming this challenge. In this paper, we propose an occlusion segmentation framework for particle objects that does not require additional annotations. This framework only necessitates instance-level segmentation labels to obtain complete contour information of particle objects, including occluded regions. First, we decompose the occlusion segmentation task into a generic instance segmentation task and an occlusion repair task for occluded objects. Then, in order to train the occlusion repair model with only instance segmentation-level labels, we quantitatively analyze the occlusion phenomenon, including the mathematical descriptions of occlusion relationships, degrees, and distributions. Next, we geometrically transform and layer overlay the unobscured samples to construct occlusion samples containing labeling information of the occluded regions. These sample sets are used to train a generative model that predicts the contour information of occluded regions. Finally, we fine-tune or post-process the pre-segmentation model with the particle objects containing restored complete contour information to achieve the final occlusion segmentation. We conducted extensive ablation experiments on both the ore-particle dataset and publicly available cell-particle datasets. The experimental results validate the effectiveness, accuracy, and generalizability of our method.**

***Note to Practitioners*—Particle segmentation has been faced with the occlusion problem. In this paper, inspired by the similarity between particle objects, we propose a self-supervised occlusion segmentation framework that does not require additional annotation of occlusion layers. Our approach requires only instance segmentation level annotation without more complex additional manual annotation, which is crucial for practical applications. In addition, we decouple the complex occlusion relation modeling into a binary classification problem without knowing precisely the occlusion hierarchy between particles, which further reduces the difficulty of practical applications. Then, we also propose shading transformations to characterize the inter-particle shading distribution to construct shading sample sets from existing samples. Finally, we use these learned and constructed occlusion sample sets to pre-train the generative model for regenerating the occluded objects to complete the final occlusion segmentation. Although in a generic segmentation task, our approach may have some limitations because the segmented objects may not have an apparent similarity. However, our approach using self-supervision and the objects' properties provides valuable ideas for solving the occlusion problem. In the future, we will solve the occlusion problem regarding the properties of each class of objects rather than just considering the similarity among particle objects.**

***Index Terms*—Occlusion Segmentation, Self-supervised Learning, Occlusion Modeling, Particle Objects, Generative Model.**

I. INTRODUCTION

PARTICLES are the minor constituent units in nature that describe natural objects, just as pixels are to images and voxels are to volume data. From a microscopic perspective, atoms, molecules, cells, etc., are the smallest particles that make up all matter. From a macroscopic viewpoint, small objects with specific shapes, such as gravel, droplets, and bubbles, are also particles. By studying the properties of particles in terms of size, number, shape, and distribution, we can theoretically dissect any complex object in nature gradually. For example, in process monitoring in the process industry, the particle size distribution of an ore pile is calculated by segmenting and extracting the particle objects in the ore image [1]. In biomedicine, the edges of cells are extracted by segmentation to identify cell types [2]. With the gradual implementation and application of computer vision technology, particle segmentation has gradually become a challenging and distinctive vision task [3], [4].

Like existing general-purpose segmentation tasks, particle segmentation faces the problem of mutual occlusion between objects due to the large number of objects to be segmented, as shown in Fig. 1. In generic segmentation tasks, occlusions obscure parts of objects, and due to the significantly different contour characteristics of segmented objects (as they are inherently distinct), it becomes challenging to identify effective patterns. This results in two major difficulties:

(1) The occluded information increases the difficulty for conventional segmentation models to extract the visible contours of objects, as the contours of occluded objects have been altered. To address this issue, some existing methods have attempted to improve segmentation accuracy in the presence of occlusions by incorporating a large amount of high-quality manually labeled data that includes occlusion relationships [5], [6]. However, in practical applications, due

This work was supported by the National Major Scientific Research Equipment of China (No. 61927803). (Corresponding author: Zhaohui Jiang).

Jinshi Liu, Zhaohui Jiang, Weihua Gui, and Zhiwen Chen are with the School of Automation, Central South University, Changsha 410083, China. Weihua Gui and Chaobo Zhang are with the Pengcheng Laboratory, Shenzhen 518055, China. (e-mail: ljs11528@csu.edu.cn; jzh0903@csu.edu.cn; gwh@csu.edu.cn; zhiwen.chen@csu.edu.cn; zhangchb@pcl.ac.cn.)

to factors such as cost, it is often difficult to provide the required large amount of manual label data.

(2) The occluded information contains parts of the actual contour information of objects, and conventional segmentation models are almost unable to predict the complete contour information. According to our research, there are currently few segmentation methods that can accurately extract the complete contour information of objects in the presence of occlusions. Most current segmentation models for occluded objects only achieve segmentation of the visible regions. Requiring the segmentation model to predict the contours of occluded parts while also segmenting the visible region's contours poses significant challenges to model design and data labeling.

In contrast, in particle segmentation tasks, occlusions differ somewhat from those in generic segmentation tasks. particle segmentation objects often have similar contour characteristics because they belong to the same category of objects and typically appear in groups or piles, sharing certain similarities in appearance contours. Therefore, this paper leverages the homogeneity of particle objects to focus on solving the occlusion problem in particle segmentation. We first transform the occlusion segmentation task into a generalized instance segmentation task with an occlusion repair task for occluded objects. Then we utilize the homogeneity of particle objects to construct the existing unobscured data samples with labeling information via planar geometric transformation and layer superposition to train and evaluate the occlusion repair model. The main contributions of this paper are as follows:

1. We propose a self-supervised occlusion segmentation framework for particle objects that does not require additional labeling of occlusion layers. This method only requires instance-level segmentation annotations without the need for more complex additional manual annotations.

2. We model the complex occlusion relationships as a binary classification problem of distinguishing whether an object is occluded or not, without the need to precisely know the layer relationships between particles. Additionally, we propose an occlusion transformation (randomly combined planar geometric transformations) to quantify and represent the degree of occlusion between particles.

3. We learn from existing samples which have complete contour information to construct a high-confidence occlusion sample set, and pre-train a generative model for the occluded regions based on this. The final occlusion segmentation is completed via fine-tuning the pre-segmentation model and post-processing.

II. RELATED WORK

A. Occlusion Segmentation Without Additional Annotation

To our knowledge, there are currently few segmentation methods specifically addressing the occlusion problem. Some studies consider occlusion merely as an interference that hinders recognition and impacts overall accuracy. The primary goal remains to identify the visible, unoccluded pixels without obtaining the complete contours of objects, thus these methods can still be used without introducing additional annotations. For example, Zhou et al. designed a feature extractor to extract enough scene information to solve the occlusion problem [7]. However, this method is extraordinarily sample-dependent and cannot guarantee the accuracy of the results. Liu et al. proposed optimizing pseudo-label performance via physical methods such as skeleton guidance, and then improving segmentation performance under occlusion via semi-supervised hybrid training [8]. Although this method does not rely on a large number of high-quality samples, it still cannot accurately identify the pixel classes of the occluded regions. Liang et al. introduced a Proposal-free Network to segment objects under occlusion, but their approach cannot capture the complete contour information of the occluded parts [9]. Xie et al. leverage scale-invariant feature transform flow and bilateral representation to solve inconsistency under occlusion [10]. Chen et al. determine the occlusion attribution among different object instances solely based on similarity [11]. Wada et al. combined semantic segmentation and instance segmentation, predicting the occluded areas as a separate category [12]. Purkait et al. shifted their thinking by proposing grouped semantic segmentation method to perform complete segmentation of objects in the scene separately. While these methods do not require additional annotation information, they cannot extract the contour information of objects in the occluded regions. Some researchers have proposed unsupervised methods; Koichiro et al. proposed a slanted plane model to obtain occlusion boundaries. Current segmentation methods often have low accuracy and struggle to handle occlusion interference effectively. Our new method uses self-supervised learning to create a fresh occlusion dataset with label information from the existing unoccluded dataset. This new dataset follows the basic occlusion law, ensuring reliable model learning without the need for additional manual labeling. Instead, the model can be trained under supervision using the automatically generated new masking dataset.

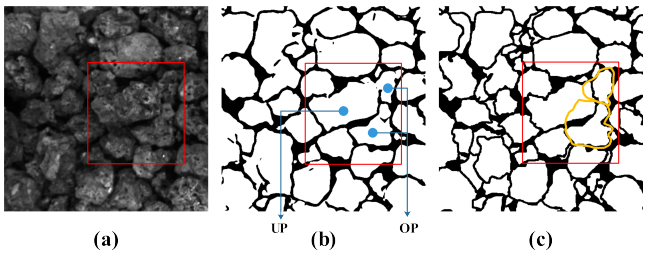


Figure 1. Object occlusion leads to incorrect segmentation. (a) is a input image; (b) is a segmented image with UNet; (c) is GroundTruth. UP represents a kind of unobscured particles, and OP indicates a kind of obscured particles. The yellow curve represents the real boundary of OPs.

B. Occlusion Segmentation Requiring Additional Annotation

Some researchers have enhanced segmentation accuracy under occlusion by introducing a substantial amount of additional annotated information to disentangle complex

occlusion relationships and directly predict the contours of occluded regions [13]. For instance, Chen et al. utilized GANs to represent the mapping between regular segmentation and occlusion segmentation, generating results based on regular segmentation inputs [14]. This method, while more reliable than previous approaches, heavily depends on the performance of the sample data for both training and inference. Purkait et al. adopted a different strategy by proposing a grouped semantic segmentation method, which performs complete segmentation of objects in a scene separately [15]. Although this method achieves better recognition results in occluded regions, it requires additional annotations and inferences for the occlusion order of objects. John et al. introduced the Layout Consistent Random Field to segment partially occluded objects, but their approach necessitates additional annotations to specify the occlusion relationships between adjacent nodes [16]. Even though current segmentation methods can identify object contours in occluded areas, they need a lot of detailed annotation information [5], [17], [18]. This kind of information is hard to gather via manual labeling in real-world applications. To solve this problem, our method suggests a new self-supervised learning approach to automatically create many samples with labels from an existing dataset. These samples are produced as layered overlays. Real occlusion happens when different targets overlap in layers. Because of this, the new samples can provide a large amount of high-quality labeling information, even without manual annotation. This ensures that the target model can be trained effectively.

C. Self-supervised Learning

Supervised learning depends on a lot of high-quality labeled data, which has held back its progress. Luckily, research now focuses on self-supervised learning, which needs fewer labels and samples. To improve occlusion segmentation research, we're thinking about using self-supervised learning because it requires fewer labels and samples. The term "self-supervised learning" first appeared in robotics, where training data is automatically labeled using input sensor signals. Current self-supervised learning research is split into two main categories: generative and contrastive [19]. Generative self-supervised learning uses methods such as GAN, VAE, and diffusion to restore the original data distribution without making assumptions about downstream tasks [20]–[22]. Generative models can learn the deep structure and features of data by creating or generating data, providing rich representations with strong diversity and adaptability. Additionally, they are often easier to understand because they directly generate or reconstruct data. However, training generative models (e.g., GANs) is complex and unstable, requiring large amounts of computational resources and data. They may suffer from issues like Mode Collapse, needing fine-tuning and stable training techniques [23]. Contrastive self-supervised learning is closely related to metric learning, effectively capturing the global and local structure of the data, which improves the model's performance in downstream tasks. This approach does not require manual labeling of data and can fully use a large amount of unlabeled data for training, reducing the cost and difficulty of data labeling. Since the contrast learning method needs to maximize the similarity between different views, the model is more robust to data distortion and noise, and can better adapt to the complexity of practical applications. However, it usually requires the computation of a large number of similarity matrices, especially when training on large-scale datasets, with high computational and storage costs. Effective contrast learning tasks and data enhancement strategies require extensive experimentation and tuning to ensure that the model learns meaningful features. The selection of appropriate negative samples significantly impacts the model's performance, with improper selection potentially leading to poor training results. The occlusion segmentation problem deals with finding the outline and position of a target when there are overlapping targets. Using instance segmentation seems like a logical approach to address the occlusion problem. Generative self-supervised learning can provide extra labeling information, which is a unique advantage. Therefore, we propose to combine instance segmentation with generative self-supervised learning to extract complete contour information of objects, including occluded regions, without needing additional annotations. This effectively solves the occlusion problem in particle segmentation by leveraging a large amount of occlusion data with labeling information obtained via image restoration models.

III. METHODOLOGY

A. Motivation

Without additional labeling of the occluded region (with only instance segmentation-level labels), the model itself cannot directly obtain any information about the occluded region. Therefore, most existing segmentation methods cannot accurately predict the invisible region when other objects partially obscure the object. So, we consider generating those invisible regions by learning essential information from the existing labels. The segmentation model can then make more accurate predictions for those generated invisible regions. Precisely, our SOPS network architecture consists of the following: (1) A pre-segmentation model f_{pre} for generating coarse segmentation images. (2) A self-supervised image restoration sub-network for recovering the obscured mask. (3) Fine-tuning or post-processing of the f_{pre} using the output of the repair sub-network.

B. Pre-Segmentation Model

The structure of the decoder needs to be discussed categorically. One is that the pre-segmentation model contains a classification branch, such as some instance segmentation models Mask RCNN, etc [24]. We will denote the pre-segmentation models as f_{sac}. The other is that the pre-segmentation model does not contain a classification branch

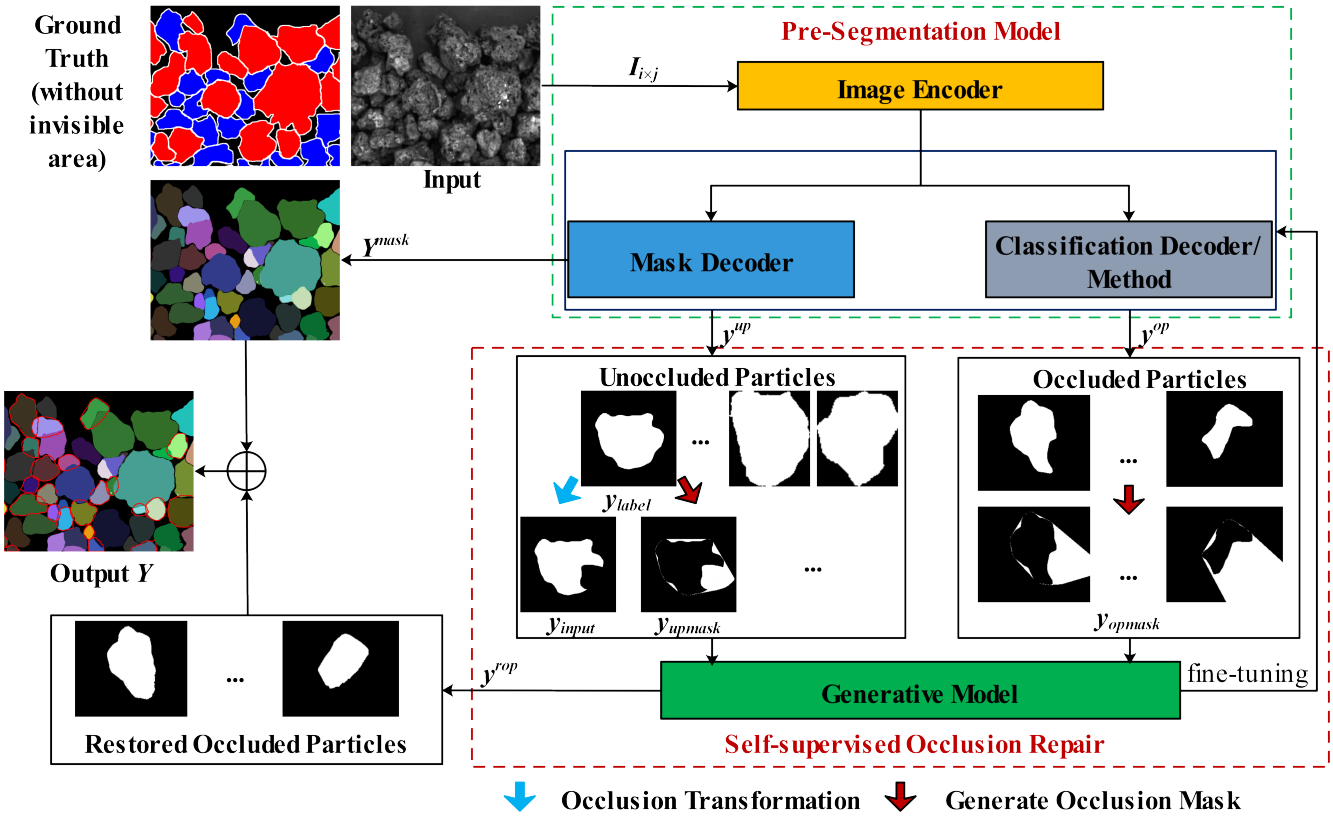


Figure 2. The structure of our SOPS. The "/" in the figure indicates that the class of particles is classified by an unsupervised clustering method if a pre-segmentation model without classification branches is used. The blue arrows in the figure indicate the proposed occlusion transformation and the red arrows indicate the proposed occlusion mask generation algorithm.

f_{onlys}, such as SAM and other segmentation models [25]. The following discussion on encoders is for f_{sac} as a pre-segmentation model.

1) Encoder:

a) Feature Extractor: For feature extraction of the input image, we first use ResNet-50, pre-trained on the large-scale image classification task ImageNet [26], for parameter initialization. Then the feature maps of Conv2 (Layer1), Conv3 (Layer2), Conv4 (Layer3), and Conv5 (Layer4) of ResNet are taken out and put into the FPN for operation [27]. FPN is an efficient backbone using a top-down architecture to build an intra-network feature pyramid from a single input scale via lateral connections. Therefore, we use the ResNet-FPN backbone for feature extraction with significant improvements in accuracy and speed.

b) Head: For the network head, we pay close attention to the architecture proposed in the previous work. We use a convolutional network as a mask prediction head. Then we use the Faster R-CNN box head as the boundary prediction head [28].

$$\begin{cases} Y_n^{class}, Y_n^{reg} = f_{class}(f_{ex}(I_{i\times j})) \\ Y_n^{mask} = f_{mask}(f_{ex}(I_{i\times j})) \end{cases} \quad (1)$$

where $Y_n^{class} = [y_0^c, y_1^c, \ldots, y_{n-1}^c]$ indicates the categorie of objects. $Y_n^{mask} = [y_0^m, y_1^m, \ldots, y_{n-1}^m]$ indicates the mask of objects. f_{ex} indicate the feature extraction.

2) Decoder:

a) Mask Decoder: Mask Decoder is a decoder used to generate masks. As a more general structure in the network, the decoder structure in CNN networks such as FCN and UNet can be used [29], [30]. In this paper, the decoder structure in UNet is used. We note that our Mask Encoder has a straightforward structure. More complex designs have the potential to improve performance but are not the focus of this work.

b) classification Decoder/Method: When $f_{pre} = f_{sac}$, the subsequent training of the f_{pre} can be regarded as fine-tuning. When $f_{pre} = f_{onlys}$, the f_{pre} does not need to be trained again, and the subsequent work can be regarded as post-processing of the f_{pre} results. Moreover, to apply the f_{pre} without the classification branch, we use an unsupervised method to distinguish the class of particles in the image (occluded or unoccluded).

Specifically, we consider each particle object as an individual connected domain. Then we take the connected domain as a sample and define the feature vector of this sample according to the geometric features of the particle as $F = (\frac{p_h}{p_c}, \frac{a_h}{a_c}, \frac{d_f}{d_s})$, where p_h, p_c and a_h, a_c represent the perimeter and area of the connected hull and connected domains, respectively. d_f indicates the farthest distance from the convex defect to the convex hull. d_s denotes the

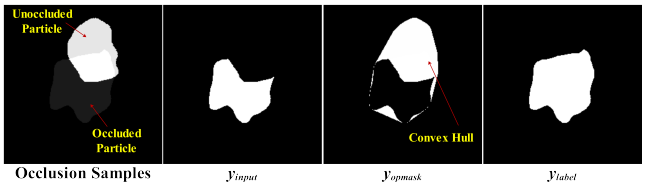


Figure 3. Occlusion samples and their components.

equivalent diameter of the connected domain area. Finally, we apply the K-means clustering algorithm to unsupervised learning of the samples to obtain the classification results of the particles [8].

C. Occlusion Modeling

1) Occlusion Relationship: The occlusion problem in particle segmentation contains two sub-problems: the occlusion relationship and the occlusion distribution. Modeling the occlusion relationship refers to the problem of figuring out the occluder and the occluded. In practice, however, a particle object can be both an occluder and an occluded object (it can occlude another particle while being impeded by other particles). Therefore, we model the problem of characterizing the inter-particle occlusion relationship as a dichotomous problem, i.e., we need to distinguish whether a particle object is an occluded or unoccluded one. An object is considered an occluded object if a pixel of any other particle object appears on the complete image projection of a particle object and, conversely, an unoccluded one. The advantage of such modeling is that such a definition allows one to distinguish between the two on the image clearly and reduces the difficulty of constructing a sample set. The classification of particle objects can be achieved using only the instance segmentation level of annotation. Therefore, we design a boundary prediction branch in the network head, which outputs the class of particle objects inside the bounding box while predicting the position and size of the bounding box. According to the class of particle objects, we also classify the output of the mask prediction branch into two categories: occluded particle (OP) or unoccluded particle (UP). OPs refer to particle objects that are partially occluded in an image, and their projection on the image is incomplete; UPs refer to particle objects that are not obscured by any other objects in the image, and they can present a complete projection in the image.

$$\begin{cases} y_n^{up} = Y_n^{mask} \cdot Y_n^{class} \\ y_n^{op} = Y_n^{mask} \cdot \tilde{Y}_n^{class} \end{cases} \tag{2}$$

where y_n^{up} refers to the set of UPs, and y_n^{op} refers to the set of OPs. Y_n^{class} refers to the category of particles (0 or 1), and $\tilde{Y}_n^{class}$ represents the contrary categories of particles (1 or 0).

2) Occlusion Distribution: We model the occlusion distribution to find out what kind of distribution characteristics the formation of the occlusion region on the image has. In fact, with the above modeling of the occlusion relationship, we can consider the occlusion formation as the result of the intersection of the occluded and unoccluded objects formed on the image by some geometric transformations. Therefore, we define the occlusion distribution as the geometric transformation in the plane of two objects with an occlusion relationship. The geometric transformations in the plane include translation $(\Delta x, \Delta y)$, rotation θ, scaling (s_x, s_y) and miscut $(\tan\alpha, \tan\beta)$ parameters. Let the coordinates of the image before the transformation be (x_0, y_0) and the image coordinates after the transformation be (x, y). When the image is only rotated, we have $x_0 = r\cos\gamma, y_0 = r\sin\gamma$ and $x = r\cos(\gamma+\theta), y = r\sin(\gamma+\theta)$, where (r, γ) indicates the polar coordinates. So, we can get

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = R \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \tag{3}$$

When the image is scaled, translated, and miscut, we have $x = s_x x_0 + \Delta x + y_0 \tan\alpha, y = s_y y_0 + \Delta y + x_0 \tan\beta$, then we have

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & \tan\alpha & \Delta x \\ \tan\beta & s_y & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = T \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \tag{4}$$

To introduce the occlusion relationship, we unify the above transformations as

$$\begin{bmatrix} x \\ y \\ c \end{bmatrix} = \begin{bmatrix} s_x & \tan\alpha & \Delta x \\ \tan\beta & s_y & \Delta y \\ 0 & 0 & c \end{bmatrix} R \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = P \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \tag{5}$$

where c represents the categorie of particle objects. In addition, due to the constraints of image size and occlusion relationship, there are boundary values for the occlusion transformation between two particle objects.

$$s.t. \begin{cases} r_x - S_x < x < r_x + r_w - S_x \\ r_y - S_y < y < r_y + r_h - S_y \\ -\pi < \theta, \alpha, \beta < \pi \end{cases} \tag{6}$$

where r_x, r_y, r_w and r_h denote the center point coordinates and width and height of the minimum outer rectangle of y_q^{bot}, which are mainly used to characterize the approximate contour of OPs. Constraining (x, y) to this range ensures that the constructed occlusion samples have overlapping areas as much as possible. s_x and s_y denote the size of the y_p^{top} cropped image, usually set to 256 or 512.

Therefore, a region with occlusion can be considered as the result of a superposition of both the OP and UP images after a series of geometric transformation, which we call the occlusion transformation P.

$$y^{top} = \sum_{i=1}^{k} P \cdot y_i^{up} \tag{7}$$

where y_i^{up} indicates the mask label of particle objects. k indicates that the particle is obscured by k other particles. y^{top} denotes the particle that obscures other particles in the constructed occlusion sample.

Up to this point, the samples can be obtained by determining the parameters of the occlusion transformation via random sampling. If a particle has been obscured by multiple particles simultaneously, it is only necessary to reconsider the already generated sample as an obscured sample. Then the occlusion transformation can be performed with a randomly sampled UP, and so on.

D. Self-supervisied Occlusion Repair

1) Occlusion Samples: We now understand the occlusion problem better, but we still need to deal with the interference it causes. Our goal is to extract important information from particle objects that are not obscured, without needing extra annotations to address the occlusion problem. However, getting information directly from unobscured particle objects doesn't guarantee its accuracy and validity. So, we'll use the occlusion problem model to create a sample set that shows the occlusion relationship and distribution of unobscured particle objects. After that, we'll use this sample set to train an image restoration model that can fix the obscured areas.

Specifically, we first randomly sample two particle objects from the unobscured particle objects Y_n^{up} and use one of them as the unobscured object y_p^{top} and the other as the obscured object y_q^{bot}. Next, the y_p^{top} and y_q^{bot} are transformed on an image $F_{p,q}$ by the occlusion transformation, and their intersection $I_{p,q}$ is created. The y_q^{bot} is labeled as y_{label}, and the difference between the y_q^{bot} and intersection $I_{p,q}$ is labeled as y_{input}. Since there are often some blank pixels in the boundaries between particle objects in the ordinary semantic segmentation results, these pixels break the continuity of the distribution between particles. The pixels in the occluded region should have obvious continuity. Therefore, to compensate for this continuity, we design the mask as the difference between the convex hull of the concatenated set of y_p^{top} and y_q^{bot} and the input, as shown in Fig. 3.

$$\begin{cases} y_{input} = y_q^{bot} - y_p^{top} \cap y_q^{bot} \\ y_{upmask} = f_c(y_p^{top} \cup y_q^{bot}) - y_{input} \\ y_{label} = y_q^{bot} \end{cases} \tag{8}$$

where f_c indicates the computation of convex hulls.

2) Generate Occlusion Mask: In constructing the occlusion samples, the occlusion masks y_{opmask} of the training samples can be easily obtained by the occlusion transformation and Equation. 8. However, when the occlusion samples are input in the generator, the y_{opmask} of the occlusion samples cannot be obtained directly. Therefore, we design an unsupervised method to directly obtain the y_{opmask} of the occluded samples to enable the generator to repair the occluded regions. First, we use Graham's Scan method to compute the convex hull of each particle objects with a space complexity of O(1) and a time complexity of O(nlgn) [31]. Then, if there is an intersection between the convex hull of the current OP and other convex hull, the Equation. 9 is executed to calculate the y_{opmask} of the OP. The reason why convex hull are used for discrimination is because in instance segmentation, the edges between different objects may be background, so two objects that do not have intersection in the segmentation result may still have an occlusion relationship.

$$\begin{aligned} y_{opmask} &= f_c(y^{op} \cup y^{closer}) - y^{op} \\ y^{closer}(y^{op}) &= \underset{j \in [0,M)}{\arg} f_c(y^{op}) \cap f_c(y_j^{op}) \neq \emptyset \end{aligned} \tag{9}$$

where M denotes the number of OPs. y^{closer} indicates those particles that have intersection with the convex hull of the current y^{op}.

3) Occlusion Recovery Sub-network: We feed the generated set of occlusion samples into the generative network, as shown in Fig. 4. By training, we obtain the generator G that can repair the obscured regions of particles. Then the OP samples are fed into G one by one to obtain the mask of restored OPs (ROP) y^{rop}. The y^{rop} is more realistic than the y^{op}, although not necessarily closer to the label. However, we would like to use the y^{rop} to optimize the mask branch's prediction results continuously. Therefore, when the f_{pre} is f_{sac}, we use the generated y^{rop} to fine-tune its mask branch on the pre-segmentation model. It is then added to the mask branch to help it produce results closer to the y^{rop} at the initial prediction. When the f_{pre} is f_{onlys}, the generated y^{rop} is replaced sequentially with the positions of the y^{op} in the prediction results.

$$\begin{cases} Y = f_{fine}(f_{sac}, \sum_{i=0}^{M-1} y_i^{rop} + Y^{mask}) & , if\ f_{pre} = f_{sac} \\ Y = \sum_{i=0}^{M-1} y_i^{rop} + Y^{mask} & , if\ f_{pre} = f_{onlys} \end{cases} \tag{10}$$

where f_{fine} denotes the fine-tuning of the f_{pre}.

We selected PEN-Net from many generative models and trained its generator as our generator [32]. PEN-Net can restore an image by encoding contextual semantics from full-resolution input and decoding the learned semantic features into images. The pseudo-code of our proposed self-supervision occlusion repair (SOR) is shown in Alg. 1.

E. Loss

The overall loss of the perspective segmentation network contains the loss of the mask prediction branch, the loss of the SOR pre-training model, and the loss of the SOR pre-training model fine-tuning. In the perspective segmentation network, the SOR model actually performs the function of online fine-tuning and testing. The SOR pre-training model is obtained by training in advance with the labels of the original image. Therefore, the loss of the SOR pre-training model is defined as follows:

$$L_{pre}^{sor} = \lambda_0 L_{rec} + \lambda_1 L_{adv} \tag{11}$$

where L_{rec} indicates the reconstruction loss between the generated result and the label, e.g. L_1 loss. L_{adv} denotes the adversarial loss that distinguishes the restored graph from the original graph, as defined in GAN [33].

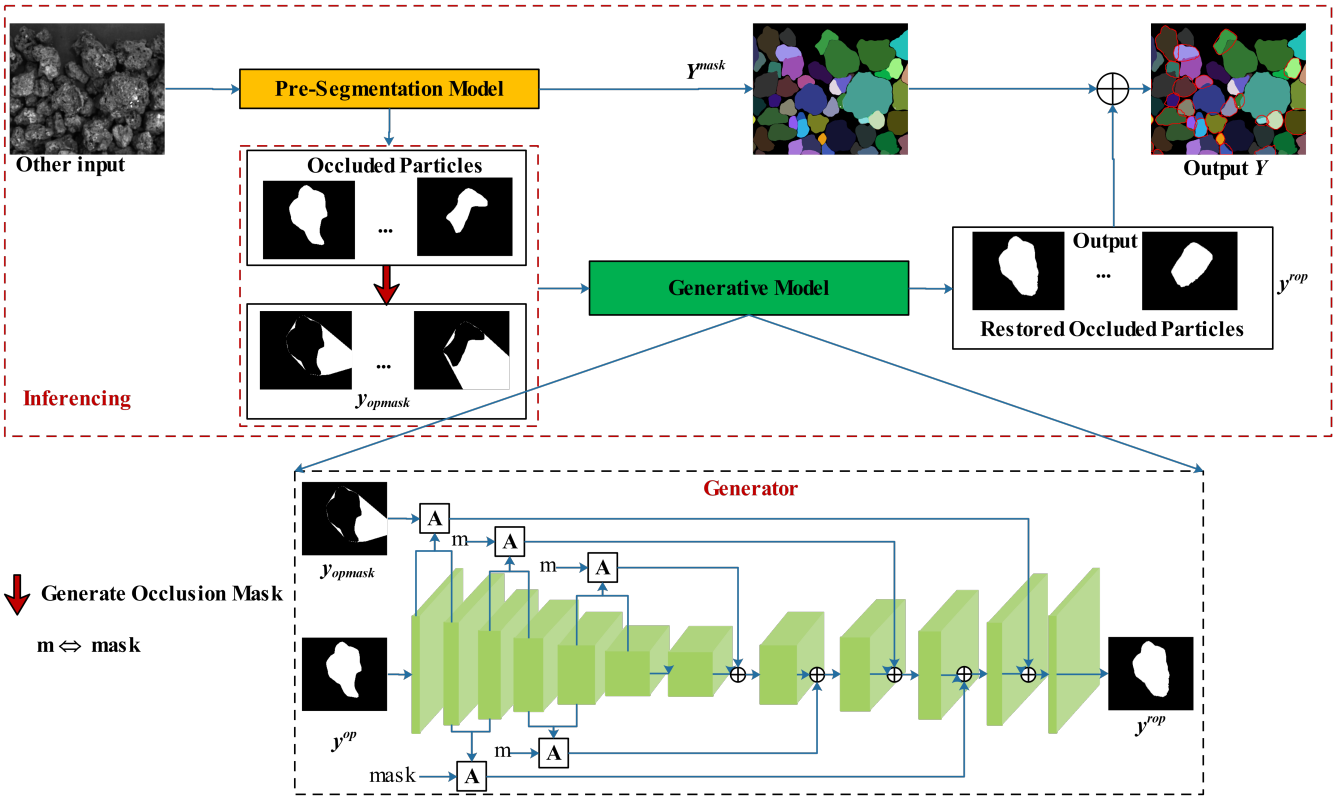


Figure 4. Inference process for our method. A denotes a sub-network capable of learning regional attention from deep semantic features, and is called Attention Transfer Network (ATN) in PEN-Net. By using ATN repeatedly it is possible to obtain higher resolution when restoring images.

The loss of the mask prediction branch usually uses the traditional cross-entropy loss. In training the perspective segmentation network, it is impossible to directly train the UPs with L_1 loss in the SOR pre-training model because these data are not labeled. Therefore, we exploit the homology and homogeneity of homogeneous particle objects, where these particles have similar geometric features (Hu moment) on the image [34]. We want to use this similarity to constrain the training of the sub-network. The loss for subsequent training of the SOR pre-training model is defined as follows:

$$L_{fine}^{sor} = \lambda_2 L_{CE} + \lambda_3 \sum_{j=0}^{M-1} Hu(y_j^{rop}, \overline{y}^{up}) \tag{12}$$
$$Hu(A,B) = e^{-\sum_{i=0}^{6} \left| \frac{\ln|h_i^A| - \ln|h_i^B|}{\ln|h_i^A| + \ln|h_i^B|} \right|}$$

where h_i denotes a term in the Hu moment. L_{CE} denotes the cross-entropy function. $\overline{y}^{up}$ represents the average mask of UPs. M denotes the number of OP. $\lambda_0, \lambda_1, \lambda_2, \lambda_3$ represent the coefficients.

Algorithm 1 Self-supervised Occlusion Repair Algorithm

Input: Occluded particles y^{op} and unoccluded particles y^{up}
Output: The restored occluded particles y^{rop}

for each $i \in [0, len(y^{up})]$ **do**
  $y^{bot} = y_i^{up}$
  for each $j \in [0, len(y^{up})]$ **do**
    if $i! = j$ **then**
      $y^{top} = P \cdot y_j^{up}$
      $OS(y_{input}, y_{upmask}, y_{label}) = y^{top} + y^{bot}$
    end if
  end for
end for
Training a Generative Model f_{GM} by OS.
$y^{rop} = f_{GM}(y^{op})$

IV. Experiment

A. Dataset and Experimental Platform

Ore and cell segmentation has been one of the hot issues in industrial and biomedical research. Moreover, ores and cells are also very classical particle objects. Therefore, we consider validating our method with ore and cell datasets. According to our research, there is no publicly available dataset for ore particle segmentation. So, we made a dataset for ore particle segmentation by ourselves. To obtain the ore images, we installed industrial cameras and other equipment on the feed belt of a blast furnace in a steel plant. Fifty-six 2176*808 non-duplicate images were selected from the collected images as the tagging dataset. The label drawing tool manually labeled the 56 images by tagging pixels as background, obscured particles, and unobscured particles. These samples were made into a training set, a validation set, and a test set in the ratio of 6:1:1. 2100 256*256 labeled

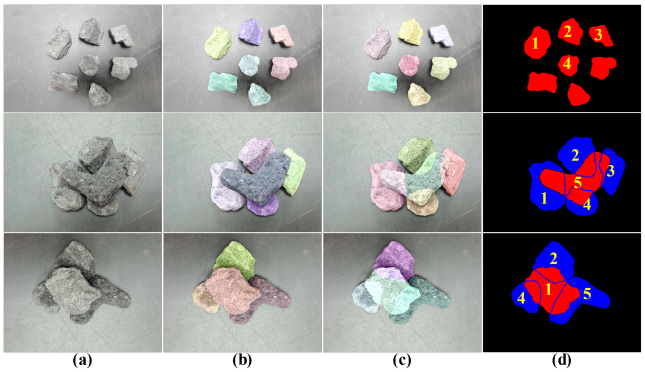


Figure 5. Test results under the annotation of the occlusion segmentation level. (a) origin images; (b) SAM; (c) SAM+Ours; (d) GroundTruth. The same number in different samples indicates the same ore particle.

images were obtained by data enhancement.

In addition, we have tested on LIVECELL, a currently open-source biomedical cell segmentation dataset [35]. This paper configures the experimental hardware environment as Intel core i9-9900K 3.6 GHz 32 GB and NVIDIA GeForce RTX2080TI 11 GB. The software environment is configured as Python 3.7.6, PyTorch 1.4.0, CUDA 10.1, and CUDNN 7.6.5. Adam is employed for optimization with the learning rate set to 1×10^{-4}, where a cosine annealing learning rate scheduling policy is adopted to adjust the learning rate with a minimum value of 1×10^{-6} in the last epoch. The weight decay is set to 1×10^{-8}.

B. Evaluation Metrics

Our segmentation results can be considered as containing two parts. One part is the regular segmentation results of unobscured particles with label supervision, which we evaluate with $mIoU$. The other part is the segmentation and prediction result of unlabeled particles without label supervision, which we evaluate by the image moment Hu. The particle objects we focus on can change the shape of the particle objects projected onto the image as we adopt different poses and devices. The Hu moment of an image is precisely an image feature with translation, rotation, and scale invariance. Therefore, due to the high similarity of shape features between similar particles (particle homogeneity), we can use the Hu moments of the extracted intact particles to assess the degree of recovery of those obscured particles that have been repaired. So, we evaluate the two components separately and then fuse them to obtain the occlusion segmentation accuracy (OSA).

$$OSA = \frac{1}{N+M}\left(\sum_{i=0}^{N-1} IoU_i + \frac{1}{N}\sum_{j=0}^{M-1}\sum_{k=0}^{N-1} Hu(y_j^{rop}, y_k^{up})\right) \tag{13}$$

where h_i represents the Hu moments, which contains seven components. y_k^{up} represents the mask of UP, and y_j^{rop} indicates the mask predicted by the generative model. IoU_i indicates the intersection over union. N denotes the number of UP, and M denotes the number of OP.

Table I
COMPARISON OF ACCURACY AND EFFICIENCY OF DIFFERENT OCCLUSION SEGMENTATION METHODS

Dataset	Method	mIoU(↑)*	Hu(↑)	OSA(↑)⋆	FPS
Ore	UNet [30]	83.37	62.42	77.83	5.01
	Mask RCNN [24]	82.22	73.23	79.84	3.23
	BCNet [5]	83.54	78.55	82.22	1.81
	SAM [25]	84.63	76.14	82.38	0.72
	ORCNN [6]	83.28	75.21	81.15	2.10
	Mask RCNN+Ours	83.31	83.90	83.47	1.65
	SAM+Ours	**85.15**	**85.36**	**85.21**	0.64
Cell	UNet	84.40	63.29	82.62	5.84
	Mask RCNN	83.34	76.18	82.74	4.52
	BCNet	84.28	82.97	84.17	2.79
	SAM	84.42	79.06	83.97	1.01
	ORCNN	83.56	78.35	83.12	3.21
	Mask RCNN+Ours	84.39	86.64	84.98	2.52
	SAM+Ours	**85.97**	**87.83**	**86.13**	0.83

*. Here mIoU is calculated only for unobscured objects (Bold: best; Underline: second best).
⋆. In this experiment, the percentage of UP and OP were 73.6% and 26.4% in the ore dataset, respectively; the percentage of UP and OP were 91.6% and 8.4% in cell dataset, respectively. Therefore, OSA = 0.736*mIoU+0.264*Hu in the ore dataset and OSA = 0.916*mIoU+0.084*Hu in the cell dataset.

C. Overall Accuracy and Efficiency Comparison

1) Comparison with other methods under the annotation of the instance segmentation level: First, we selected the classical algorithms UNet and Mask RCNN [24], [30] in supervised semantic and instance segmentation to compare the overall segmentation accuracy. We used them as the baseline for some experiments, as shown in Fig. 6. By observing the performance of the ore dataset in Tab. I, our methods with Mask RCNN and SAM as the f_{pre} improve 1.09% and 2.93%, respectively, over Mask RCNN in mIoU. Then, we selected the current cutting-edge occlusion segmentation methods BCNet and ORCNN to compare their performance on the occlusion problem [5], [6]. In the previous paper, we proposed to use Hu to evaluate the degree of recovery of OPs. Moreover, our methods with Mask RCNN and SAM as the f_{pre} improve 6.76% and 9.22% over BCNet on Hu, respectively. SAM is well known as one of the most cutting-edge big models for vision today [25]. Our comparative experiments also show that a better-performing pre-trained model also leads to better overall performance of our model. Thus our approach with SAM as the f_{pre} achieves the most significant advantage in terms of overall accuracy. In terms of efficiency, the FPS of our method is mostly the same compared to the original method too.

2) Test results under the annotation of the occlusion segmentation level: To further demonstrate the advantages of our method in occlusion segmentation, we tested with the same set of coke particle images in both the occluded and unoccluded cases. With Fig. 5, it can be seen that our method does perform better in terms of its ability to recover the occluded region compared to other masking methods, and is visually closer to the real label.

We analyze the following two reasons why the proposed method performs better than existing occlusion segmentation

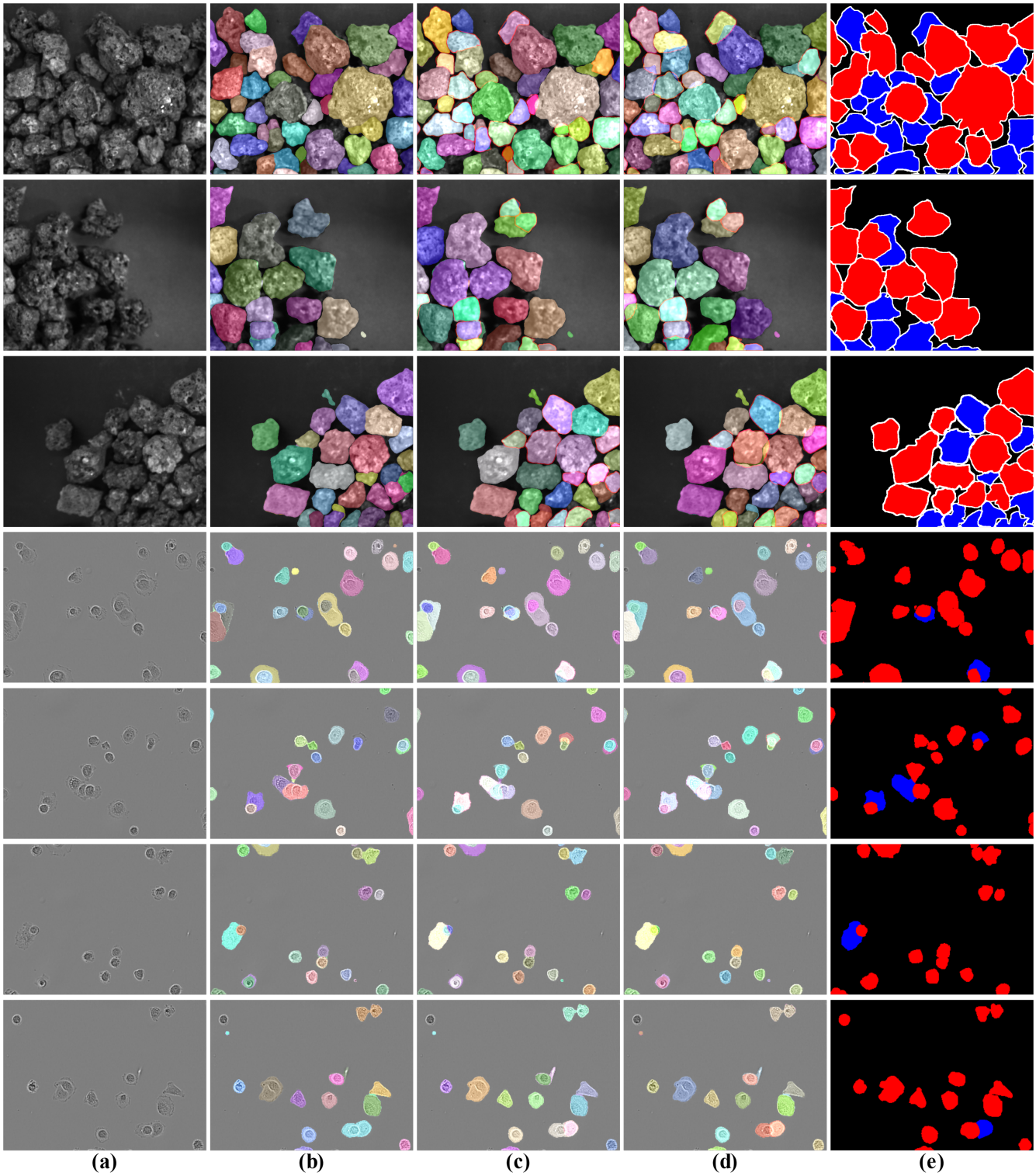


Figure 6. Comparison of results for selected test sets. (a) original inputs; (b) SAM; (c) SAM + RePaint; (d) Ours; (e) GroundTruth without invisible area, where the blue area indicates obscured particles and the red area indicates unobscured particles. The areas marked by red lines in the figure indicate the recognition results of invisible areas by occlusion segmentation. The first three images are from the ore particle dataset, and the last four images are from the cellular particle dataset livecell.

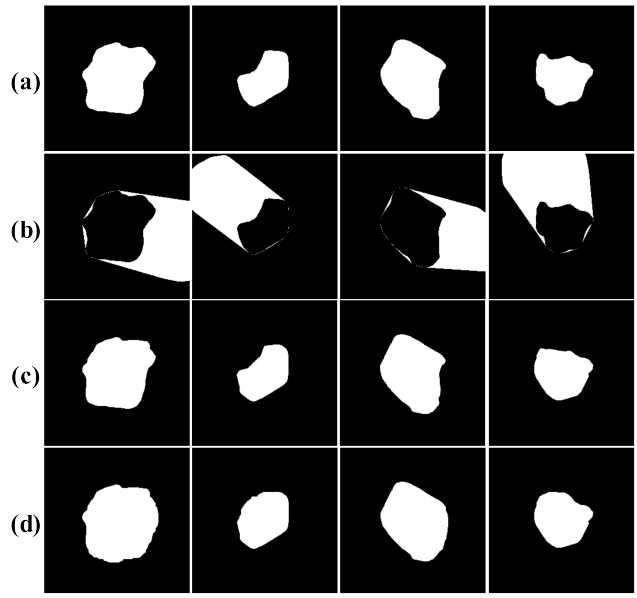


Figure 7. Comparison of the restoration effect of different generation models on obscured particles in ore particle dataset. (a) OP y^{op}; (b) occlusion mask y_{opmask}; (c) y^{rop} by RePaint; (d) y^{rop} by PEN-Net.

methods. (a) Our method are more lenient on sample labeling. Most existing methods require additional annotation of the occlusion layers in the samples in advance. They must learn enough information from these additional annotations to maintain high-accuracy segmentation performance. Instead, as in our case, we just fuzzy classify the occlusion relation (only into OP and UP). This fuzzy classification labeling requirement can be easily satisfied in general instance segmentation. (b) Our framework can leverage the segmentation capabilities of a powerful f_{pre}, after which we can achieve our more advanced task goals via fine-tuning or post-processing by a small number of samples. In contrast, existing occlusion segmentation methods often require retraining and thus still cannot escape from the reliance on high-quality samples.

D. Ablation Experiments

1) Impact of different generative models on the results: The generative model in this paper mainly undertakes to repair the occluded area. We compare and analyze the framework of the generative model and the generator's design in two aspects. First, we selected several mainstream generative frameworks for comparative analysis: PICNet (based on VAE [36]), Repaint (based on diffusion [37]), PConv, and PEN-Net (based on GAN) [32], [38]–[40], as shown in Fig. 7. In the case of using the same generator, PEN-Net is the best in Hu and OSA in the dataset used in this paper, as shown in Tab. II. Then, we use different generators in PEN-Net to perform a comparative analysis. By observing Tab. II, we use the generator combination of UNet+ATN to perform the best in all metrics.

We analyze several reasons for the generation of this experimental result. (a) In this paper, the complexity of the generation task is low because both the input and output results are binary images. Some more complex training architectures (VAE, diffusion, etc.) do not necessarily achieve better training results. The classical GAN architecture still achieves better results in this paper. (b) Our mask is more complex, so the generator may need to consider how to learn more valuable features from the mask. From the experimental results, ATN, a sub-network that can better aggregate the input and mask's deeper features, can improve overall performance.

Table II
EFFECT OF DIFFERENT GENERATIVE MODELS ON OCCLUSION SEGMENTATION

Dataset	Model*	Generator	mIoU(↑)	Hu(↑)	OSA(↑)
Ore	PICNet [38]	UNet	83.12	80.82	82.51
	PEN-Net [32]	UNet	83.08	81.57	82.68
	PConv [40]	UNet	82.22	79.64	81.53
	RePaint [39]	UNet	83.01	81.30	82.55
	PEN-Net(we used)	**UNet+ATN**	**83.31**	**83.90**	**83.47**
Cell	PICNet	UNet	84.15	83.13	83.88
	PEN-Net	UNet	84.04	84.63	84.19
	PConv	UNet	83.15	82.37	82.94
	RePaint	UNet	84.09	84.44	84.18
	PEN-Net(we used)	**UNet+ATN**	**84.39**	**86.64**	**84.98**

*. We use different generative model frameworks for training and inference. In this ablation experiment, our network backbone and f_{pre} is ResNet-50+RPN and Mask RCNN.

Table III
COMPARISON OF ACCURACY AND EFFICIENCY OF DIFFERENT FEATURE EXTRACTION MODULES

Backbone*	mIoU(↑)	Hu(↑)	OSA(↑)	FPS
ResNet-34	81.23	83.36	81.79	2.97
ResNet-50	82.45	83.64	82.76	1.88
ResNet-101	83.54	83.99	83.65	0.69
ResNet-34+FPN	82.22	83.52	82.56	2.74
ResNet-50+FPN(Ours)	83.31	83.90	83.47	1.65
ResNet-101+FPN	83.77	84.28	83.90	0.56

*. In this ablation experiment, we use the PEN-Net as the generative model, as shown in Fig. 4.

2) Effect of feature extraction models on the results: When we choose Mask RCNN as the f_{pre}, the results differ when different feature extraction modules are chosen. We conducted a comparison experiment with ResNet and FPN [27], [41], as shown in Tab. III. It can be seen that ResNet+FPN has better feature extraction results in the method of this paper. Moreover, the accuracy of segmentation results gradually increases as the parameters of ResNet keep increasing. However, the improved accuracy of ResNet-101 compared with ResNet-50 is minimal, and it loses a lot of efficiencies, which seems to reach the bottleneck. FPN uses a top-down architecture to build an intra-network feature pyramid by connecting laterally from a single input scale. The addition of FPN allows ResNet to extract multi-scale features more efficiently. Therefore, ResNet-50+FPN is chosen as our feature extraction module in this paper.

E. Limitations and Future Research

The most ingenious processing in this paper is to use self-supervised learning to learn complete contour information from homogeneous complete particle objects, and then repair and extract those occluded contour information. This method allows for the reconstruction and extraction of occluded contour information. However, if the contour information between the unoccluded and occluded objects lacks sufficient similarity, it may adversely affect the outcomes of the self-supervised learning process. Consequently, acquiring adequate contour information of the occluded object via self-supervised learning alone can be challenging. To address this limitation, future work could involve incorporating a small number of labeled samples to provide additional contour information about the occluded objects, thereby enhancing the model's overall performance. Alternatively, we could consider modeling the contour correlation between unoccluded and occluded objects within the image. Leveraging the results of this modeling could assist the model in accurately predicting the contours of occluded objects.

V. CONCLUSION

The occlusion problem in segmentation has long been challenging. Most existing methods require additional annotations and struggle to capture occluded contour information. This paper introduces an innovative occlusion segmentation framework for particle objects that relies solely on instance-level segmentation labels to obtain complete contour information, including occluded regions. The approach uses self-supervised learning to acquire comprehensive contour information from homogeneous, complete particle objects. However, if the contour information lacks similarity between unoccluded and occluded objects, the effectiveness of the self-supervised learning process may be compromised. Future work could incorporate a small number of labeled samples to provide additional contour information about occluded objects, enhancing the model's overall performance. Additionally, modeling the contour correlation between unoccluded and occluded objects within the image could further assist the model in accurately predicting the contours of occluded objects.

REFERENCES

[1] J. Liu, Z. Jiang, W. Gui, and Z. Chen, "A novel particle size detection system based on rgb-laser fusion segmentation with feature dual-recalibration for blast furnace materials," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 10, pp. 10 690–10 699, 2022.

[2] V. Petukhov, R. J. Xu, R. A. Soldatov, P. Cadinu, K. Khodosevich, J. R. Moffitt, and P. V. Kharchenko, "Cell segmentation in imaging-based spatial transcriptomics," *Nature biotechnology*, vol. 40, no. 3, pp. 345–354, 2022.

[3] J. Liu, Z. Jiang, W. Gui, Z. Chen, W. Luo, and C. Zhang, "Hierarchical packing model: Estimating the overall particle size distribution from surface images and permeability properties," *IEEE Transactions on Instrumentation and Measurement*, 2023.

[4] Z. Jiang, J. Liu, Z. Chen, W. Luo, C. Zhang, and W. Gui, "Overall particle size distribution estimation method based on kinetic modeling and transformer prediction," *Engineering Applications of Artificial Intelligence*, vol. 128, p. 107517, 2024.

[5] L. Ke, Y.-W. Tai, and C.-K. Tang, "Deep occlusion-aware instance segmentation with overlapping bilayers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4019–4028.

[6] P. Follmann, R. König, P. Härtinger, M. Klostermann, and T. Böttger, "Learning to see the invisible: End-to-end trainable amodal instance segmentation," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 1328–1336.

[7] J. Zhou, D. Yang, Z. Cui, S. Wang, and H. Sheng, "Lrfnet: An occlusion robust fusion network for semantic segmentation with light field," in *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2021, pp. 1178–1168.

[8] J. Liu, Z. Jiang, T. Cao, Z. Chen, C. Zhang, and W. Gui, "Generated pseudo-labels guided by background skeletons for overcoming under-segmentation in overlapping particle objects," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[9] X. Liang, L. Lin, Y. Wei, X. Shen, J. Yang, and S. Yan, "Proposal-free network for instance-level object segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2978–2991, 2017.

[10] Q. Xie, O. Remil, Y. Guo, M. Wang, M. Wei, and J. Wang, "Object detection and tracking under occlusion for object-level rgb-d video segmentation," *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 580–592, 2018.

[11] Z. Chen, D. Ting, R. Newbury, and C. Chen, "Semantic segmentation for partially occluded apple trees based on deep learning," *Computers and Electronics in Agriculture*, vol. 181, p. 105952, 2021.

[12] K. Wada, K. Okada, and M. Inaba, "Joint learning of instance and semantic segmentation for robotic pick-and-place with heavy occlusions in clutter," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9558–9564.

[13] J. Lazarow, K. Lee, K. Shi, and Z. Tu, "Learning instance occlusion for panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 720–10 729.

[14] Y. Chen, G. Lin, S. Li, O. Bourahla, Y. Wu, F. Wang, J. Feng, M. Xu, and X. Li, "Banet: Bidirectional aggregation network with occlusion handling for panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3793–3802.

[15] P. Purkait, C. Zach, and I. Reid, "Seeing behind things: Extending semantic segmentation to occluded regions," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1998–2005.

[16] J. Winn and J. Shotton, "The layout consistent random field for recognizing and segmenting partially occluded objects," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1. IEEE, 2006, pp. 37–44.

[17] X. Yuan, A. Kortylewski, Y. Sun, and A. Yuille, "Robust instance segmentation through reasoning about multi-object occlusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 141–11 150.

[18] S. Kim, T. Ahn, Y. Lee, J. Kim, M. Y. Wang, and F. C. Park, "Dsqnet: A deformable model-based supervised learning algorithm for grasping unknown occluded objects," *IEEE Transactions on Automation Science and Engineering*, pp. 1–14, 2022.

[19] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 857–876, 2021.

[20] S.-H. Huang, T.-Y. Chao, B. A. Wibisono, M. P.-H. Lin, and C.-C. Huang, "Ssioe: Self-supervised indoor occupancy estimation for intelligent building management," *IEEE Transactions on Automation Science and Engineering*, pp. 1–14, 2023.

[21] G. Peng, Z. Ren, H. Wang, X. Li, and M. O. Khyam, "A self-supervised learning-based 6-dof grasp planning method for manipulator," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3639–3648, 2022.

[22] D. Li, J. Lu, T. Zhang, and J. Ding, "Self-supervised learning and multisource heterogeneous information fusion based quality anomaly detection for heavy-plate shape," *IEEE Transactions on Automation Science and Engineering*, pp. 1–12, 2023.

[23] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020.

[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[25] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[27] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[28] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[29] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[31] V. Tereshchenko, Y. Tereshchenko, and D. Kotsur, "Point triangulation using graham's scan," in *Fifth International Conference on the Innovative Computing Technology (INTECH 2015)*. IEEE, 2015, pp. 148–151.

[32] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1486–1494.

[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[34] J. Žunić, K. Hirota, and P. L. Rosin, "A hu moment invariant as a shape circularity measure," *Pattern Recognition*, vol. 43, no. 1, pp. 47–57, 2010.

[35] C. Edlund, T. R. Jackson, N. Khalid, N. Bevan, T. Dale, A. Dengel, S. Ahmed, J. Trygg, and R. Sjögren, "Livecell—a large-scale dataset for label-free live cell segmentation," *Nature methods*, vol. 18, no. 9, pp. 1038–1045, 2021.

[36] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

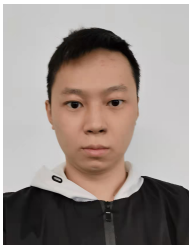
[37] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[38] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1438–1447.

[39] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 461–11 471.

[40] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 85–100.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.



Jinshi Liu received the B.S., M.S., and Ph.D. degrees in automation from Central South University in 2017, 2020, and 2024, respectively. Currently, he is working as a postdoctoral fellow at Central South University. His research interests include deep learning, computer vision, and digital twin.

Zhaohui Jiang received a M. Eng. degree in Automatic Control Engineering and a Ph.D. degree in Control Science and Engineering from Central South University, China in 2006 and 2011, respectively. He is currently a professor at Central South University. His research interests include Detection Technology and Automatic Equipment, image processing, industrial VR, modeling and optimal control of complex industrial processes.

Weihua Gui received the degree of the B.Eng. in Electrical Engineering and the M.Eng. in Automatic Control Engineering from Central South University, China in 1976 and 1981 respectively. From 1986 to 1988 he was a visiting scholar at Universität-GH-Duisburg, Germany. He has been a full professor in Central South University since 1991. He was elected as a member of the Chinese Academy of Engineering in 2013. His main research interests are in modeling and optimal control of complex industrial process, distributed robust control, and fault diagnoses.

Zhiwen Chen (M'17) received his B.S. degree in electronic information science and technology and M.S. degree in electronic information and technology from CSU, China, in 2008 and in 2012, respectively, Ph.D. degree in electrical engineering and information tech nology from University of Duisburg-Essen, Germany in 2016. He is currently an associated pro fessor at CSU. His research interests are model-based and data-driven fault diagnosis and health monitoring, data analytics.

Chaobo Zhang (Member, IEEE) received the master's and Ph.D. degrees in civil engineering from Hong Kong University of Science and Technology, Hong Kong, in 2017 and 2020, respectively. He is currently an assistant researcher in Peng Cheng Laboratory, Shenzhen, China. His current research interests include intelligent sensing, big data analysis, deep learning and computer vision for industrial applications.