



An automatic progressive chromosome segmentation approach using deep learning with traditional image processing

Ling Chang¹ · Kaijie Wu¹ · Hao Cheng¹ · Chaocheng Gu¹ · Yudi Zhao¹ · Cailian Chen¹

Received: 1 September 2022 / Accepted: 16 June 2023 / Published online: 28 September 2023
© International Federation for Medical and Biological Engineering 2023

Abstract

The fully automatic chromosome analysis system plays an important role in the detection of genetic diseases, which in turn can reduce the diagnosis burden for cytogenetic experts. Chromosome segmentation is a critical step for such a system. However, due to the non-rigid structure of chromosomes, chromosomes may curve in any direction, and two or more chromosomes may touch or overlap to form unpredictable chromosome clusters in metaphase chromosome images, leading to automatic chromosome segmentation as a challenge. In this paper, we propose an automatic progressive segmentation approach to perform the entire metaphase chromosome image segmentation using deep learning with traditional image processing. It follows three stages. In the first stage, thresholding-based and geometric-based methods are employed to divide all chromosomes as single ones and chromosome clusters. To tackle the segmentation for unpredictable chromosome clusters, we first present a new chromosome cluster identification network named CCI-Net to classify all chromosome clusters into different types in the second stage, and then in the third stage, we combine traditional image processing with deep CNNs to accomplish chromosome instance segmentation from different types of clusters. Evaluation results on a clinical dataset of 1148 metaphase chromosome images show that the proposed automatic progressive segmentation method achieves 94.60% chromosome cluster identification accuracy and 99.15% instance segmentation accuracy. The experimental results exhibit that our proposed approach can effectively identify chromosome clusters and successfully perform fully automatic chromosome segmentation.

Keywords Fully automatic chromosome analysis · Automatic progressive segmentation · Chromosome cluster identification · Chromosome instance segmentation · Deep learning

1 Introduction

Since Tjio and Levan discovered the number of human chromosomes in 1956 [1] and the Denver system of human chromosome classification was established in 1960 [2], human chromosome karyotyping analysis [3, 4] has become an essential procedure for cytogeneticists to diagnose human genetic disorders or cancers in genetic laboratories [5, 6]. In general, a healthy human has 23 pairs of chromosomes, including 22 pairs of autosomes and one pair of sex chromosomes (XX for female and XY for male). Chromosome karyotyping analysis is the process of separating all individual chromosomes from the metaphase chromosome image

firstly, then classifying these chromosomes into 23 types, and finally arranging them in an order to form a karyogram [7], as shown in Fig. 1.

Traditional chromosome karyotyping analysis needs well-trained operators to pay considerable manual effort and time to produce karyotyping results, which are directly related to experts' experience [8, 9]. For cytogeneticists, the analysis process is laborious and time-consuming, so many researchers have been working for years on automated karyotyping using computation techniques. Automatic karyotyping analysis commonly follows two stages: chromosome segmentation and chromosome classification.

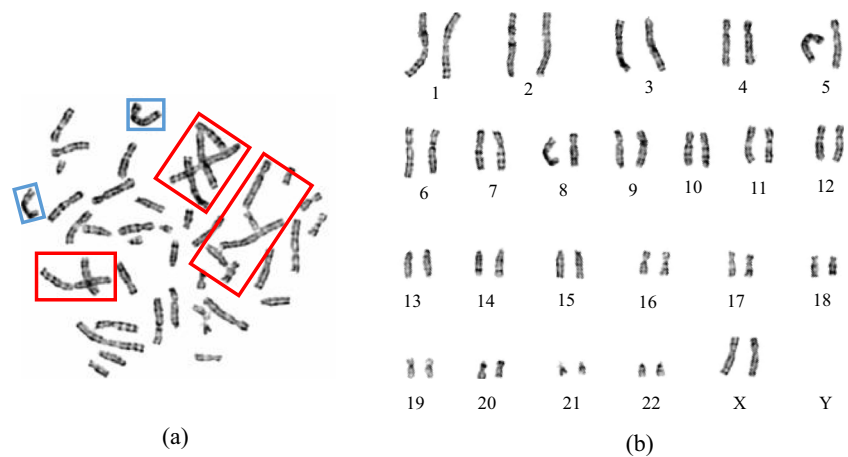
Chromosome segmentation is the critical stage for karyotyping and its results can directly influence classification performance afterward; thus, this paper focused on the segmentation problem.

Nowadays, many chromosome segmentation methods mainly used traditional image processing to extract

✉ Kaijie Wu
kaijiwu@sytu.edu.cn

¹ Department of Automation, Shanghai Jiao Tong University, Shanghai, China

Fig. 1 **a** A Giemsa-stained metaphase chromosome image, in which blue and red rectangles frame curved chromosomes and chromosome clusters, respectively. **b** The corresponding standard karyotyping image of **(a)**



manually tagged features, such as contour, end points, and intersection points to perform automatic chromosome separation [10–12]. However, these features are sensitive to the shape of the chromosome and cannot handle some complicated chromosome clusters. With the prevalence of deep learning, some studies have proposed specialized neural networks to solve chromosome segmentation [13–17]. But these studies only address the separation for overlapped chromosomes. Investigating the above research, they reveal that the challenges of automatic chromosome segmentation mainly lie in the following reasons: (1) As shown in blue rectangles in Fig. 1(a), since chromosomes are non-rigid, they can curve in any direction, making it difficult to accurately extract effective features, such as contour and media axes, for successfully accomplishing the segmentation task. (2) Touching or overlapping chromosome clusters with unpredictable shapes always exist in the metaphase chromosome image, which become a non-ignorable phenomenon and seriously reduce the segmentation accuracy. Red rectangles in Fig. 1(a) show several chromosome clusters.

To address the above challenges, we propose a progressive segmentation approach to achieve the entire metaphase chromosome image separation automatically in this paper. This approach conducts the segmentation with a three-stage style. In the first stage, thresholding-based and geometric-based methods are utilized to divide all chromosome objects into individual ones and chromosome clusters. For chromosome cluster segmentation task, different from other methods like [13–17] which only focus on the segmentation for several shapes of clusters, we first identify all chromosome clusters as three types: touching ones, overlapping ones, and touching and overlapping ones in the second stage, and then we segment chromosome instances from different types of chromosome clusters in the final stage. To identify chromosome clusters, we propose a chromosome cluster identification

network (CCI-Net) which integrates one backbone network, one multi-scale feature fusion (MFF) module, and one identification head. Specifically, the backbone network combined residual network [18] with SE-block [19] extract discriminative multi-scale features, and MFF integrates semantics information from the high-level layer and rich detailed information from all low-level layers; thus, these two modules implement dual feature enhancement. The identification head utilizes the final informative features to classify chromosome clusters. To solve the segmentation of each type of cluster, we introduce different chromosome instance segmentation methods. For touching clusters, we use the least square method to fit a reasonable separation path between a pair of cut-points. For overlapping clusters, we apply a simplified U-Net to extract overlapped regions followed by traditional image processing to stitch individual chromosomes. Combining these two segmentation methods, touching and overlapping clusters can be separated gradually.

To evaluate the performance of our proposed method, we collected 1148 metaphase chromosome images and cropped 6941 chromosome clusters from these images to generate a chromosome cluster dataset. Compared to other state-of-the-art methods, CCI-Net achieves an accuracy of 94.60% for chromosome cluster type identification, and the progressive segmentation obtains an accuracy of 99.15%. The experimental results demonstrate that our proposed method effectively identifies chromosome clusters and successfully performs fully automatic chromosome segmentation.

The main contributions of this paper can be summarized as follows:

1. A new progressive segmentation approach is proposed by using the three-stage style combined traditional image processing with deep learning to achieve the entire metaphase chromosome image segmentation automatically.

2. A novel chromosome cluster identification network named CCI-Net is presented. It consists of one backbone network, one multi-scale feature fusion (MFF) module, and one identification head. With the help of dual feature enhancement implemented by the first two module, the identification head can generate the type of each cluster.
3. Three chromosome instance segmentation methods to separate touching, overlapping, and touching and overlapping chromosome clusters are designed. To our best knowledge, these segmentation methods are the first attempts to solve separation problems for almost all chromosome clusters.
4. Compared to other state-of-the-art methods, experimental results on our clinical dataset demonstrate that our progressive segmentation approach has a superior performance. Meanwhile, CCI-Net makes a high classification accuracy for chromosome cluster identification, and better segmentation results are achieved for all chromosome clusters.

The rest of the paper is structured as follows: Section 2 reviews the related works on chromosome segmentation. Section 3 gives us the details about our proposed approach. In Section 4, we provide experimental results and a discussion. At last, we present the conclusion in Section 5.

2 Related work

High-quality chromosome segmentation is necessary and primarily crucial for chromosome karyotyping analysis development, and the segmentation result directly affects the accuracy of the classification stage. Traditional methods for chromosome segmentation are generally performed manually, yet the segmentation accuracy highly depends on the experiences of the cytogenetics experts. Thus, automatic chromosome segmentation is the tendency in the future. Scholars have dedicated to automatic chromosome segmentation in the metaphase images for many years.

Before the prevalence of deep learning, earlier segmentation approaches mainly employed thresholding-based or geometric-based methods to perform automatic chromosome segmentation [10–12, 20–26]. Minaee et al. [11] proposed a geometric approach to segment chromosomes automatically. The author went through two phases to complete the segmentation. In the first phase, three geometric criteria—surrounding ellipse, convex hull, and skeleton and end points—were designed to detect chromosome clusters gradually. In the second phase, the cut-line method was employed to separate touching or partially overlapped chromosomes. This method achieved 91.9% accuracy on 62 chromosome cluster images. However, its segmentation quality relied heavily on the manually extracted

geometric features, and it cannot separate complicated chromosome clusters.

Altinsoy et al. [20] presented an automatic segmentation method for raw G-band chromosome images. They started from clearing background pixels by thresholding processing. Then, some irrelevant objects, such as staining debris and nucleus, were eliminated using their geometric characteristics. Next, binary watershed transform and several rules were applied to divide all objects into single and cluster chromosomes. After that, they used geodesic distance transform on extracted cut-points to separate touching chromosomes, and for overlapped chromosomes, they mainly considered “+” and “T” shaped overlapped chromosome clusters. Nevertheless, as uneven Giemsa staining chromosome metaphase image, the proposed method may obtain the failed segmentation results.

Nowadays, deep learning has demonstrated its state-of-the-art power, and it brings new opportunities for object segmentation. Hu et al. [13] applied neural network-based segmentation to distinguish partially overlapped chromosomes. To resolve overlapping chromosome segmentation, the author built a dataset including 13,000 semi-synthetically generated overlapping chromosomes. Then, they input the dataset into a customized U-Net to perform separation for overlapped chromosomes and resulted in IOU scores of 94.7%.

To improve the segmentation performance, Saleh et al. [15] combined a modified U-Net and test time augmentation (TTA) to construct an overlapped chromosome segmentation method. Hu et al. [13] and Saleh et al. [15] obtained an improved segmentation accuracy of 99.68%.

Song et al. [16] used the main characteristics of U-Net and SegNet to compose a convolutional neural network named Compact Seg-Uet for overlapping human chromosome segmentation. To address the problem of unrealistic images, they proposed a method to generate more realistic images with opaque overlapped regions, and they achieve an average IOU score of $93.44\% \pm 0.26$.

Lin et al. [14] employed AS-PANet by improving the PANet instance segmentation model to perform the overlapped chromosome segmentation and yielded 85% instance segmentation accuracy.

These four segmentation methods utilized deep CNNs to disengage overlapped chromosomes. However, the first three methods only used for solving partially or simply overlapped chromosome segmentation, and AS-PANet did not reach a clinical segmentation accuracy.

Investigating the above segmentation methods, we can find that they explicitly or implicitly implement chromosome segmentation from different types of chromosome clusters. It indicates that type identification of chromosome clusters is an important premise task for automatic chromosome segmentation.

Recently, Lin et al. [27] proposed a chromosome cluster type identification method using the ResNeXt WSL model. They used ResNeXt weakly supervised learning to construct a backbone network with 101 layers and designed a 9-layer deep neural network as the network header to categorize chromosome features extracted by the backbone network. Meanwhile, the authors presented a fast training methodology to tune the overall framework from coarse-to-fine gradually. Experimental results showed this method achieved 94.09% accuracy. However, this method did not involve chromosome segmentation problem, and the cluster identification performance has room for improvement.

Motivated by the above studies, we propose a progressive automatic chromosome segmentation approach combining deep CNNs and traditional image processing. Especially, a chromosome cluster identification network (CCI-Net) is presented to classify clusters into touching, overlapping, and touching and overlapping chromosomes. Moreover, according to different characteristics of three types of clusters, we design different chromosome instance segmentation methods.

3 Proposed method

This section includes six parts, with the first one introducing a brief overview of our proposed method. The second to fifth parts give the details of each component of the progressive segmentation method sequentially. Then we explain the loss function and training strategy in the fifth part. Finally, evaluation metrics used in experiments are presented.

3.1 Approach view

As shown in Fig. 2, the proposed framework for automatic progressive chromosome segmentation can be described in a step-by-step style with three stages: (1) thresholding-based and geometric-based methods are applied to perform the rough segmentation and divide the whole metaphase

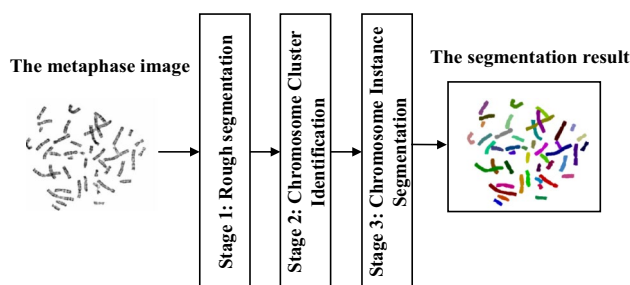


Fig. 2 The process of our automatic progressive segmentation method

chromosome image into single ones and chromosome clusters; (2) CCI-Net is presented to identify chromosome clusters as three different types; (3) using different chromosome instance segmentation methods to separate individual chromosomes from different types of chromosome clusters. Details about the three stages are introduced in the following sections.

3.2 Stage 1: rough segmentation using thresholding-based and geometric-based methods

The thresholding-based method is a popular traditional image segmentation method in practical applications, which can easily separate images into background and foreground. In order to separate chromosome objects from the entire metaphase chromosome images simply, efficiently, and completely, the global thresholding method is utilized to classify the whole image as foreground and background by the following steps:

1. Given a metaphase chromosome image $I \in \mathbb{R}^{h \times w \times c}$, we first calculate the global averaging pixel value by the following formula:

$$\text{Average}_p = \frac{\sum_{x=0}^{w-1} \sum_{y=0}^{h-1} \text{pixel}(x, y)}{h \times w} \quad (1)$$

where h , w , and c represent the height, the width, and the depth of the metaphase image, respectively, and $\text{pixel}(x, y)$ denotes each pixel value of the image I .

2. Due to the noise having been removed from our dataset, we can set $\lambda \cdot \text{Average}_p$ as the threshold value. When the pixel value is less than the threshold, this pixel is considered the background pixel. Otherwise, we save the pixel belonging to foreground objects. For the best segmentation performance, λ is set to 0.9.

After that, a binary image is generated. Before entering the next stage, we compute some chromosome geometry features, such as the convex hull and the minimum bounding rectangle, of each separated objects to identify individual chromosomes and chromosome clusters.

3.3 Stage 2: chromosome cluster identification network (CCI-Net)

As mentioned in Section 2, to facilitate automatic chromosome segmentation, most of studies are devoted to segmenting chromosomes from different types of chromosome clusters, so cluster type identification is a crucial anterior task for individual chromosome segmentation. In this paper, we propose a chromosome

cluster identification network (CCI-Net) to classify all clusters into three types: touching chromosome cluster, overlapping chromosome cluster, and touching and overlapping chromosome cluster. We define these three types of chromosome clusters as follows: if there are no intersection pixels between any two chromosomes, it is a touching cluster; if there are only intersecting pixels at the junction of two chromosomes, it is called the overlapping cluster; a chromosome cluster containing both touching and overlapping chromosomes is called a touching and overlapping cluster.

The structure of CCI-Net is shown in Fig. 3(a). It consists of three components: a backbone network, a multi-scale feature fusion (MFF) module, and an identification head. We combine deep learning network with SE-blocks to construct a CNN-based module to encode the chromosome cluster into corresponding multi-scale feature representations, and since SE-blocks can make these features focus on discriminative features, thus this module performs the first feature enhancement. MFF

module is applied to further enhance multi-scale feature representations by integrating multi-scale features from adjacent high-level and all low-level layers, which makes fusion features aggregate fine-grained details and high-level semantics. And last, the identification head is used to learn the mapping from the fused features to cluster classification probabilities.

3.3.1 A backbone network

On the one hand, ResNet [18] is the most commonly used backbone network for segmentation or classification tasks. On the other hand, as mentioned in Hu et al. [19], SE-block can adaptively recalibrate channel-wise feature responses by explicitly modelling interdependencies between channels and it can be integrated into standard deep CNNs' architectures such as VGG-Net and ResNet, to improve the features learning ability. Considering the above two aspects, we construct a SE-ResNet61 as the backbone network by stacking SE-residual blocks. Each block has four

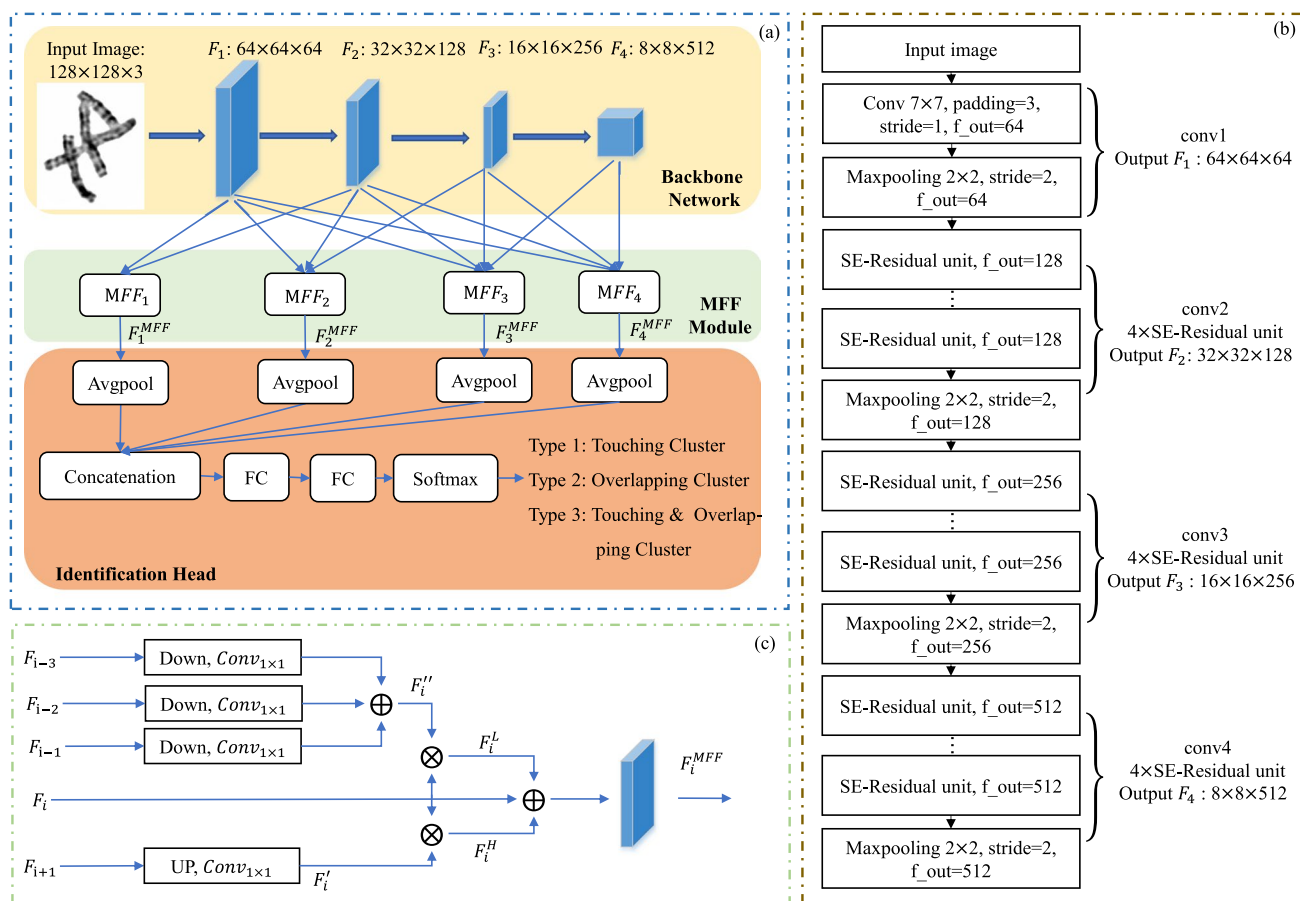
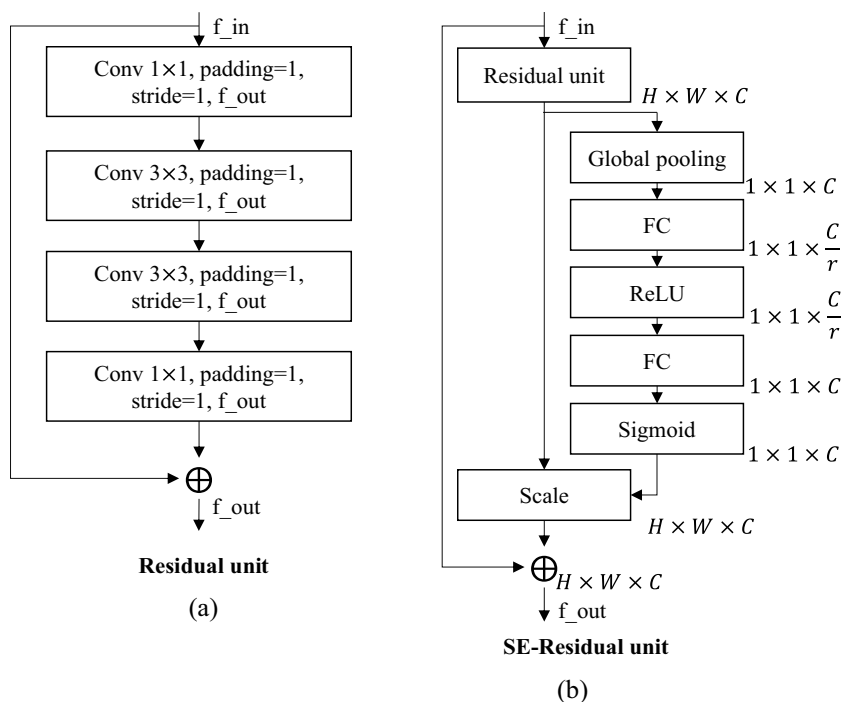


Fig. 3 The architecture of the chromosome cluster identification network (CCI-Net). **a** CCI-Net has three components: one backbone network, one MFF module, and one segmentation head. **b** The detailed

structure of the backbone network. **c** The description of multi-scale feature fusion module

Fig. 4 a, b Schemas of the residual unit and the SE-residual unit, respectively



SE-residual units. As shown in Fig. 4, the residual unit is constructed by four convolutional layers with two 1×1 kernel sizes and two 3×3 kernel sizes, respectively, and the SE-block consisting of one global pooling layer, two fully connected layers, one rectified linear unit (ReLU), and one Softmax layer is integrated into the residual unit to construct a SE-residual unit. As illustrated in Fig. 3(b), the SE-ResNet61, with 61 layers, consists of four submodules. The first submodule conv1 has one convolution layer with 7×7 kernel size and one maxpooling layer with 2×2 kernel size, and its output dimension is $64 \times 64 \times 64$. The second to fourth submodules, termed by conv2 to conv4 sequentially, are all made up of four SE-residual units and one maxpooling layer with 128, 256, and 512 output dimensions, respectively.

When we feed a chromosome cluster image into SE-ResNet61, low-level and high-level features are extracted gradually. We denote these features as $\{F_1, F_2, F_3, F_4\}$, which forms multi-scale feature representations. Compared to the original ResNet, this network extracts more discriminative features; thus, it is considered to perform the first feature enhancement. For a more intuitive understanding of F_i ($i=1, 2, 3, 4$), we use Grad-CAM to visualize each feature layer in F_i , and the visualization images of the four groups of features are given in Figure 5. From Figure 5, we can see that the shallow features of the network, such as F_1 and F_2 , retain almost all the detailed information of the original image, whereas more semantic information can be concentrated on high-level layers

like F_3 and F_4 . As the number of network layers deepens, the output feature map becomes more and more abstract.

3.3.2 Multi-scale feature fusion module

As shown in Fig. 6, we can note that the gray intensity at the junction area of touching chromosomes is very close to the background, while it is much larger than the background for overlapping clusters.

Furthermore, since the small scale of these areas, pooling layers in CCI-Net will make them be ignored in high-level layers. Thus, to facilitate the performance of the chromosome cluster identification network, more discriminative information of junction areas needs to be preserved to further enhance feature representations. He et al. [28] proposed the progressive feature fusion (PFF) module to solve the deviation of high-level features from low-level features. Inspired by this, multi-scale feature fusion (MFF) module is introduced here as shown in Fig. 3(c). Different from PFF module, MFF module enables features of each scale to integrate adjacent high-level and all low-level feature layers. By handling of the MFF module, each scale of the generated fusion feature contains both semantics information from the high-level layer and rich detailed information from all low-level layers.

As illustrated in Fig. 3(a), each scale of features will be handled by a MFF module. Figure 3(c) gives the detailed

Fig. 5 Visualization features extracted from SE-ResNet61. **a–d** Visualization images of F_1 , F_2 , F_3 , and F_4

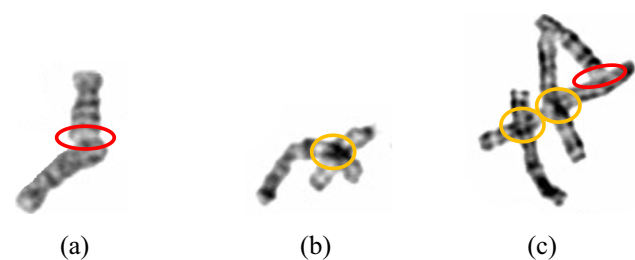
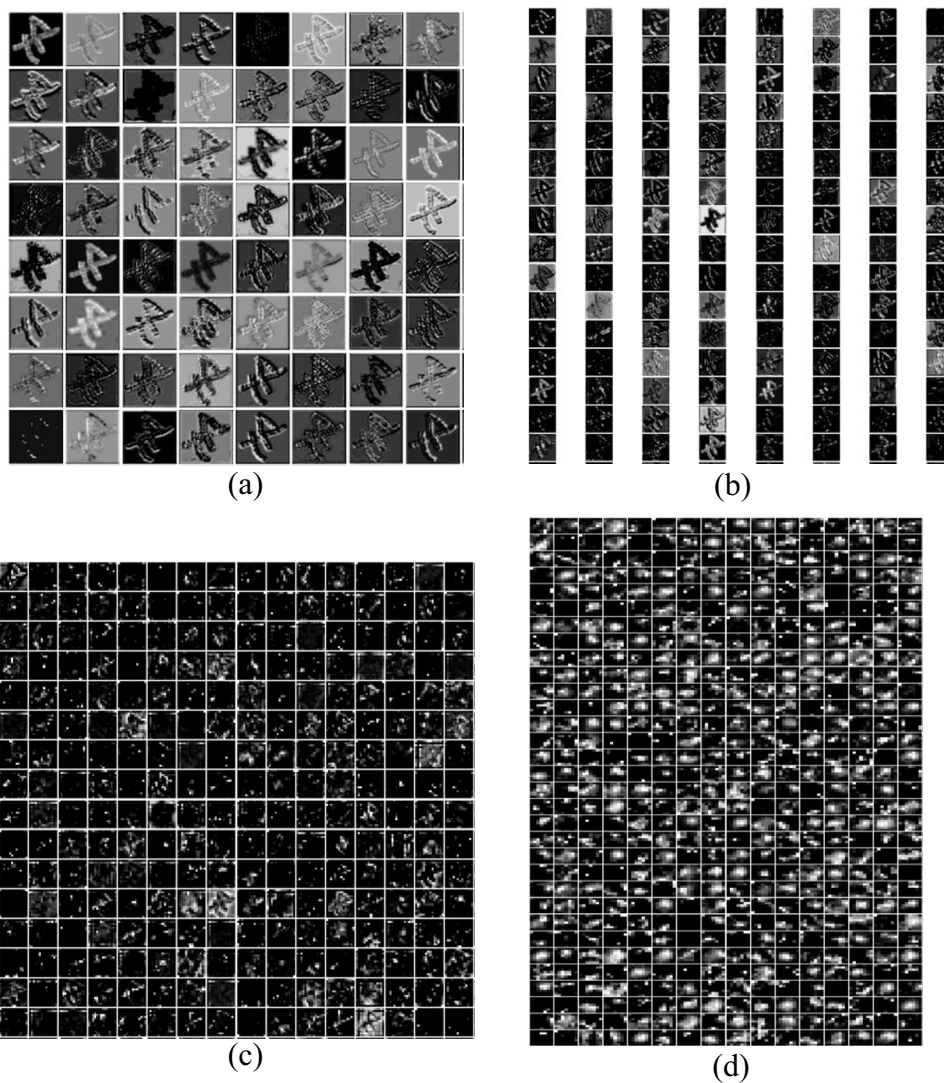


Fig. 6 Examples of different types of chromosome clusters. **a–c** Touching, overlapping, and touching and overlapping clusters, respectively. We use red and orange ellipses to circle touching junction areas and overlapped junction areas, respectively

handling process. For any scale of feature maps F_i , we first apply a $Conv_{1 \times 1}$ on its adjacent high-level feature F_{i+1} to alter the channel number to be

the same as F_i . Then we up-sample the feature maps to generate the transformed high-level feature F'_i which has the same size with F_i . It can be expressed by the following equation:

$$F'_i = Up(Conv_{1 \times 1}(F_{i+1})), (i = 1, 2, 3, 4) \tag{2}$$

where F'_i denotes the transformed high-level feature, $Conv_{1 \times 1}$ represents a 1×1 convolutional layer, and $Up(\)$ denotes $2 \times$ upsample layer. Note that F'_4 equals to 0 because F_5 is not exist. Due to high-level layers F_{i+1} having rich semantic information, the transformed feature F'_i saves the most information from it.

Then, for all low-level features $F_{i-1}, F_{i-2}, F_{i-3}(i=2, 3, 4)$ input into MFF_i , we employ a down-sample with different stride followed by a 1×1 convolutional layer to make all of them having the same resolution and channel number as F_i . The following formula represents the process:

$$F_i'' = \text{Conv}_{1 \times 1}(\text{Down}(F_{i-1})) \oplus \text{Conv}_{1 \times 1}(\text{Down}(F_{i-2})) \oplus \text{Conv}_{1 \times 1}(\text{Down}(F_{i-3})), (i = 2, 3, 4) \quad (3)$$

where F_i'' is the final transformed low-level features, \oplus represents the element-wise addition, and $\text{Down}(\cdot)$ denotes a down-sample layer with 2×2 , 4×4 , and 8×8 kernels for F_{i-1} , F_{i-2} , and F_{i-3} , respectively. F_{-2} , F_{-1} , and F_0 do not exist and set to 0. F_i'' contains the detailed chromosome cluster information as much as possible by integrating all low-level features, which could be beneficial for handling small junction area between clusters.

After obtaining high-level and all low-level information, we perform a parallel two-step element-wise multiply operation which makes F_i' multiply with F_i and F_i'' multiply with F_i simultaneously:

$$F_i^H = F_i' \otimes F_i, i = 1, 2, 3, 4 \quad (4)$$

$$F_i^L = F_i'' \otimes F_i, i = 2, 3, 4 \quad (5)$$

where F_i^H and F_i^L are the generated high-level and low-level enhancement representations, respectively, and \otimes denotes the element-wise multiply operation. The multiplication operation makes F_i fuse with F_i' and F_i'' , respectively, which can save the global information of the F_i' and the detailed information of F_i'' as much as possible. Thus, small cluster details ignored by F_i' can be reactivated by multiplying with low-level features, and chromosome clusters with fewer semantic information in F_i' can be enhanced by multiplying with high-level features.

After enhancing high-level and low-level features with the previous handling, more powerful representations can be obtained by adding two enhanced features to the unprocessed multi-scale features F_i by the following formula:

$$F_i^{MFF} = F_i \oplus F_i^H \oplus F_i^L, (i = 1, 2, 3, 4) \quad (6)$$

where F_i^{MFE} is the output of the MFF_i module.

Through the addition operation, compared with the original multi-scale features extracted from SE-ResNet61, the new generated multi-scale features contain rich detailed information and more contextual information. Thus, MFF modules implemented further enhance the feature representation for chromosome clusters.

3.3.3 An identification head

We use two fully connected layers and one Softmax layer to construct an identification head. Before entering the head, average-pooling layers are applied to conduct features firstly, then we concatenate these processed features to obtain the

final feature representations. The representations are mapped to identification probabilities of cluster types via such head.

3.4 Stage 3: chromosome instance segmentation

The previous stage has divided chromosome cluster into touching, overlapping, and touching and overlapping chromosomes. Next, we will introduce three segmentation methods for three types of clusters according to their own characteristics.

3.4.1 Touching cluster separation

As illustrated in Fig. 7(a), for a cluster image having two touching chromosomes as an example, due to the touching area between two chromosomes having low gray intensity and no pixel intersection, we apply our previously proposed method in [29] to detect two cut-points (m and n) and then compute a separation path p between two cut-points by the following steps:

- (1) Firstly, we connect two cut-points as the guide line l containing a point set $S = \{m, a_1, \dots, a_{z-2}, n\}$.
- (2) Then, after setting the pixel on the guide line l to 1 and the rest of points to 0 in the cluster image, we obtain a binary cluster image, and further we use a 3×3 kernel to dilate the binary image to obtain a touching junction area A .
- (3) After we bring the junction area A back to the original cluster image to compute the average value V_A of A by formula (1), all points of A whose pixel value is less than V_A will be remained. These remained points form candidate points shown in Fig. 7(a).
- (4) Next, if several candidate points form a 4-connected area, we only remain one point with the minimum intensity, and these reserved points constitute the final decision points.
- (5) Finally, we apply the least square method on these decision points to fit the separation path p .

For other touching cluster images having three or more chromosomes, according to each pair of cut-points, we can calculate all separation paths by using previous steps. We can notice that, comparing with connecting two cut-points directly to separate touching chromosomes, our separation approach is more in line with the segmentation style performed manually by experts.

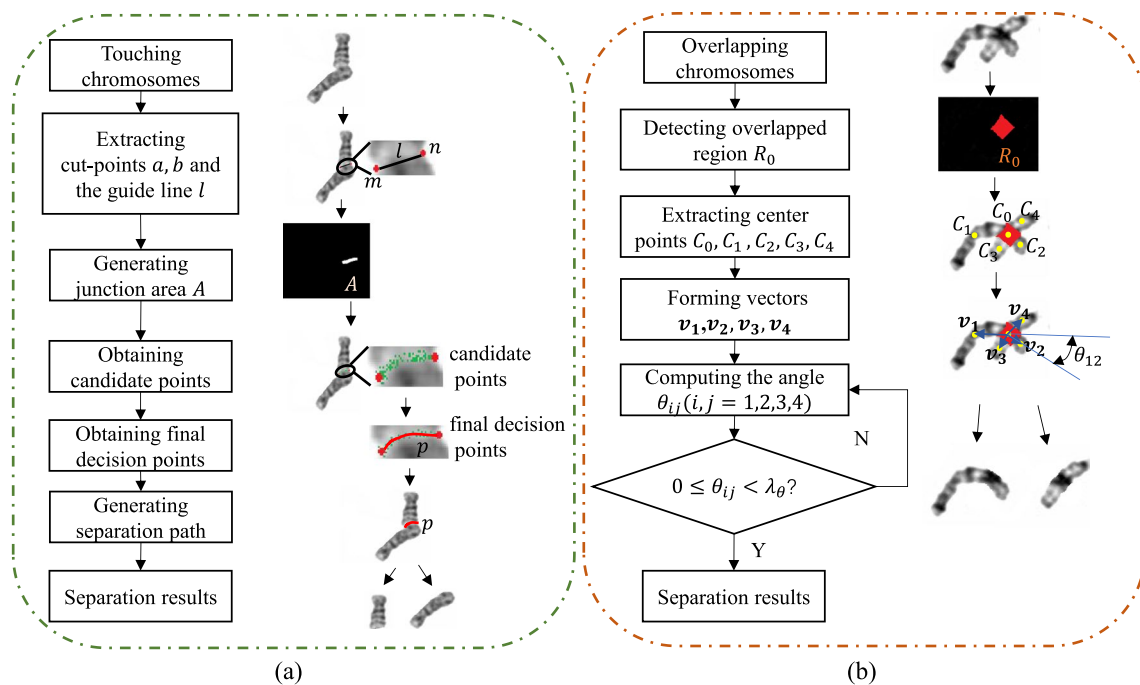


Fig. 7 The process of chromosome instance segmentation. **a, b** Touching clusters and overlapping clusters separation processes, respectively. In **(a)**, we use red points m, n represents a pair of cut-points, meanwhile the white area A and the red curve p as the junction area and separation path, respectively, and green points as the

candidate points. In **(b)**, we use the red area R_0 to denote the detected overlapped region, and yellow points $C_0, C_1, C_2, C_3,$ and C_4 as center point representations, and blue arrows $v_1, v_2, v_3,$ and v_4 to represent vectors connected C_0 with $C_1, C_2, C_3,$ and C_4 , respectively

3.4.2 Overlapping cluster separation

As we all know, U-Net [30] has been widely used in medical image segmentation. Recently, some researchers have applied it for chromosome segmentation, but they just handle simple partial overlapped chromosome using a semi-synthetically generated overlapped chromosome dataset. Meanwhile, traditional methods often use cut-points to separate overlapping chromosomes, but it is highly susceptible to image quality and cannot handle complicated chromosome clusters. Therefore, we present a new segmentation procedure for overlapping chromosomes by combining U-Net and traditional image processing. As shown in Fig. 7(b), the procedure has four steps:

- (1) Firstly, we employ U-Net to detect the overlapped region R_o , and since the resolution of our overlapped chromosome dataset is 128×128 , we simplify the standard U-Net structure, in which the first three down-sampling and the corresponding up-sampling blocks remained.
- (2) After defining the non-overlapping area as $R_1, R_2, R_3,$ and R_4 , we compute center points of these five areas using the following formulas:

$$x_c = \frac{\sum_{i=1}^n x_i}{n} \tag{7}$$

$$y_c = \frac{\sum_{i=1}^n y_i}{n} \tag{8}$$

where n is the pixel number of an area; x_i and y_i represent the abscissa and ordinate values of pixel i , respectively; x_c and y_c represent the abscissa and ordinate values of the center pixel, respectively; and these five center points are defined as $C_1, C_2, C_3, C_4,$ and C_0 corresponding to $R_1, R_2, R_3, R_4,$ and R_0 as shown in Fig. 7(b).

- (III) Thirdly, connecting C_1, C_2, C_3, C_4 to C_0 respectively to form vectors $v_1, v_2, v_3,$ and v_4 , and then we compute the angle θ between two vectors using the following formula:

$$\theta_{ij} = \begin{cases} \arccos \frac{v_i \cdot v_j}{\|v_i\| \cdot \|v_j\|}, & 0 \leq \theta \leq 90^\circ \\ 180^\circ - \arccos \frac{v_i \cdot v_j}{\|v_i\| \cdot \|v_j\|}, & 90^\circ < \theta < 180^\circ \end{cases} \tag{9}$$

where $i, j = 1, 2, 3, 4 (i \neq j); \theta_{ij}$ denotes the angle between v_i and $v_j (\theta_{ij} = \theta_{ji})$.

(D) Finally, if θ_j less than a thresholding value λ_{θ} , we consider R_i , R_j belonging to a same chromosome, and the other non-overlapped area belonging to another individual chromosome.

For overlapping clusters with three or more chromosomes, U-Net is also applied to detect all overlapped areas, then we consider the overlapped area one-by-one to separate overlapping chromosomes gradually.

3.4.3 Touching and overlapping clusters separation

Touching and overlapping clusters contain both overlapping and touching chromosomes. Thus, we can first separate overlapping chromosomes and then segment touching chromosomes using the segmentation methods in (1) and (2), respectively. Up to now, all chromosome clusters are separated as individual chromosomes.

3.5 Loss function and training strategy

In order to make deep neural networks applicable to specific tasks, researchers apply loss functions to optimize network parameters. For our method, we just need to train CCI-Net in stage 2 and simplified U-Net in stage 3 to meet our segmentation purpose. Cross entropy loss is adopted as the loss function for these two networks and its expression is shown in Eq. (10). It is used to judge the gap between the actual output and desired output, and the net uses Softmax function to produce the actual output probability value.

$$\text{CrossEntropyLoss} = -[y * \log(q) + (1 - y) * \log(1 - q)] \quad (10)$$

Then, we train two networks using their own dataset. For CCI-Net, we initialize network parameters via He's method [31], and then use Eq. (10) to train the net until convergence with all chromosome cluster datasets shown in Table 1. For simplified U-Net, we input overlapping cluster images from Table 1 and corresponding mask images to train it until the loss function reaches its minimum value.

Table 1 The statistics of our chromosome datasets

No.	Category	Count
1	The metaphase image	1148
2	Chromosome clusters	6590
3	Touching clusters	4628
4	Overlapping clusters	1387
5	Touching and overlapping clusters	935
6	Individual chromosomes	17,808

3.6 Evaluation metrics

3.6.1 Evaluation metrics for CCI-Net

To evaluate the identification performance of CCI-Net, we adopt recall, precision, accuracy (ACC), and F_1 as evaluation metrics. For the computation of these metrics, we define the following four criteria to fit the context of multi-class classification as other researchers do:

- True positives (TP_j): a single chromosome belonging to class j is identified as class j ;
- False positives (FP_j): a single chromosome not belonging to class j is identified as class j ;
- False negatives (FN_j): a single chromosome belonging to class j is identified as class $k(\forall k \neq j)$;
- True negatives (TN_j): a single chromosome not belonging to class j is identified as class $k(\forall k \neq j)$.

Then, four metrics are calculated by the following formulas:

$$\text{recall}_j = \frac{TP_j}{TP_j + FN_j} \quad (11)$$

$$\text{precision}_j = \frac{TP_j}{TP_j + FP_j} \quad (12)$$

$$\text{accuracy} = \frac{\sum_{j=1}^{N_{cls}} TP_j}{M} \quad (13)$$

$$F_1 = \frac{1}{N_{cls}} \sum_{j=1}^{N_{cls}} \frac{\text{precision}_j \cdot \text{recall}_j}{\text{precision}_j + \text{recall}_j} \quad (14)$$

where N_{cls} equals to 24 which represents the total class number of chromosomes and M denotes the number of all tested single chromosomes.

Precision refers to the ratio of the number of correctly classified positive samples to the number of samples determined to be positive by the classifier. Recall is the ratio of the correctly classified positive samples, which focus on the true positive class of samples. Accuracy is the ratio of correctly predicted chromosome clusters. The F_1 score of type identification for all clusters is calculated by averaging the F_1 score of each type cluster.

3.6.2 Evaluation metrics for segmentation

We adopt three evaluation metrics to validate our proposed segmentation method, including intersection over union (IOU), dice, and accuracy (ACC). IOU is a common metric for object segmentation, which measures the accuracy

of the detected area of corresponding objects in a specific dataset. Dice coefficient is a set similarity measurement function, which represents the similarity of the predicted object to the ground truth. ACC represents the proportion of correctly predicted pixels in the whole image, which concentrates on the accuracy of each pixel. The expressions are as follows:

$$\text{IOU} = \frac{|A \cap B|}{|A \cup B|} \quad (15)$$

$$\text{Dice} = \frac{2|A \cap B|}{|A| + |B|} \quad (16)$$

$$\text{ACC} = \frac{X_{\text{pred}}}{X_{\text{true}}} \quad (17)$$

where A and B denote the predicted object and the ground truth, respectively, and X_{pred} and X_{true} represent the pixel number of predicted object and the ground truth area.

4 Experimental results and discussion

In this section, we first introduce our datasets. Then some experimental details are given. After that we report experimental results to validate the effectiveness of CCI-Net and the automatic progressive segmentation approach, respectively. Finally, we present the discussion at the end of this section.

4.1 Material

To validate our proposed method in this section, the BEIONMED® company from Shanghai, China, provided us 1148 Giemsa-stained microscopic metaphase images with all privacy-removal from healthy people. All chromosomes from the metaphase images were manually annotated by cytologists and provided approximately 300 bands. The mask of each metaphase image was manually labeled by our lab researchers and then verified by cytologists. Meanwhile, we collected 17,808 single chromosomes and 6590 chromosome clusters including 4628 touching clusters, 1387 overlapping clusters, and 935 touching and overlapping clusters from 1148 metaphase chromosome images. Table 1 shows the details of the dataset. These images were randomly split into five subsets to conduct five-fold cross-validation.

4.2 Implementation details

In the experiments, all input images of the CCI-Net or U-Net were resized to 128×128. These two networks were trained on

their own dataset for 50 epochs, and the initial learning rate (LR) was set to 0.001, and LR decayed by 0.5 when the epoch reached 5 or multiples of 5. The momentum is set to 0.9.

We implemented our approach in Python, with the Pytorch framework. All experiments were conducted under an Ubuntu 14.04 LTS workstation with Intel (R) Core (TM) i7-5820K CPU @ 3.30GHz, 15 GB of RAM, and 1 NVIDIA GTX Titan X GPUs.

4.3 Results of CCI-Net

CCI-Net is proposed to identify types of chromosome clusters. To fully illustrate the superiority of CCI-Net, five sets of comparison experiments are given in this section. All methods were evaluated on the chromosome clusters dataset shown in Table 1. We first compare the CCI-Net with state-of-the-art methods (DenseNet121 [32], ResNet-50/101/152 [18], ResNeXt-101 [33], ResNeXt WSL [27]) as shown in Table 2. The first three methods were image classification baseline models, and the fourth method was proposed specifically for identifying chromosome clusters. Then, we add SE-block or MFF module respectively to the above network. In addition, to evaluate the efficiency of SE-blocks and MFF module from CCI-Net, we add these two modules to the above four methods to employ dual feature enhancement. In the first three groups of experiments, we used the classification network that comes with their own network. In order to make a more consistent comparison with CCI-Net, all classification networks in the fifth group of experiments are replaced with the identification head in CCI-Net. We implemented these experiments and evaluated these methods using five-fold cross validation.

As we can see from experimental results of the first group in Table 2, apart from CCI-Net, ResNeXt WSL yielded higher accuracy of 94.09%, but the total time and the number of parameters were the highest. From the detailed description of stage 2 in Section 3, we know that SE-block and the MFF module were applied to enhance multi-scale features twice. We add SE-block or MFF module to networks of the above group, and the second and third groups give experimental results which prove these two modules can improve the classification accuracy by about 0.03–0.1%. Furthermore, to verify the effectiveness of the combination of SE-block and the MFF module, we combined the combination module with feature extraction networks in the above six CNNs-based methods. The evaluation results are shown in the fourth group in Table 2. Obviously, it can be found that the accuracy and F1 scores are improved by about 0.2–0.5% after adding SE-block and MFF module. The results in the fifth group show that adding the identification head can little improve the classification accuracy compared with the fourth group.

Table 2 Comparative experimental results for the CCI-Net

Methods	Recall (%)	Precision (%)	Accuracy (%)	F1 (%)	Parameters	Time (s)
DenseNet-121 [32]	81.68	85.59	87.65	82.23	6.69×10 ⁶	20.10
ResNet-50 [18]	85.68	88.30	90.15	86.08	25.60×10 ⁶	8.34
ResNet-101 [18]	87.92	90.65	91.89	88.32	42.51×10 ⁶	17.40
ResNet-152 [18]	88.79	90.71	91.97	89.09	58.15×10 ⁶	24.30
ResNeXt-101 [33]	89.10	90.79	92.27	89.36	88.79×10 ⁶	26.97
ResNeXt WSL [27]	92.79	93.08	94.09	92.84	88.85×10 ⁶	29.40
DenseNet-121+SE-block	81.71	85.81	87.75	82.31	6.93×10 ⁶	20.09
ResNet-50+SE-block	85.73	88.35	90.19	86.21	25.84×10 ⁶	8.56
ResNet-101+SE-block	88.53	90.78	91.93	88.84	42.75×10 ⁶	17.68
ResNet-152+SE-block	88.82	90.86	92.09	89.23	58.39×10 ⁶	24.68
ResNeXt-101+SE-block	89.27	90.89	92.29	89.41	89.03×10 ⁶	27.37
ResNeXt WSL+SE-block	92.83	93.18	94.20	92.90	89.09×10 ⁶	29.74
DenseNet-121+MFF	81.73	85.79	87.81	82.35	7.08×10 ⁶	21.12
ResNet-50+MFF	85.75	88.41	90.21	86.19	25.99×10 ⁶	8.60
ResNet-101+MFF	88.51	90.81	91.95	88.92	42.90×10 ⁶	17.75
ResNet-152+MFF	88.85	90.83	92.12	89.27	58.54×10 ⁶	24.83
ResNeXt-101+MFF	89.31	90.91	92.33	89.42	89.18×10 ⁶	27.56
ResNeXt WSL+MFF	92.84	93.23	94.25	92.93	89.24×10 ⁶	29.96
DenseNet121+SE-block+MFF	81.79	86.17	87.95	82.55	7.32×10 ⁶	22.31
ResNet-50+SE-block+MFF	85.89	88.49	90.39	86.47	26.23×10 ⁶	10.40
ResNet-101+SE-block+MFF	90.90	90.91	92.00	89.52	43.14×10 ⁶	19.55
ResNet-152+SE-block+MFF	89.02	91.03	92.23	89.40	58.78×10 ⁶	25.42
ResNeXt-101+SE-block+MFF	89.39	91.00	92.58	89.55	89.42×10 ⁶	28.12
ResNeXt WSL+SE-block+MFF	93.07	93.41	94.49	93.01	89.48×10 ⁶	31.58
DenseNet121+SE-block+MFF+Head	81.80	86.18	87.96	82.57	7.08×10 ⁶	22.32
ResNet-50+SE-block+MFF+Head	85.91	88.51	90.43	86.46	25.99×10 ⁶	10.41
ResNet-101+SE-block+MFF+Head	90.93	88.02	92.03	89.54	42.90×10 ⁶	19.56
ResNet-152+SE-block+MFF+Head	89.03	91.04	92.25	89.41	58.54×10 ⁶	25.43
ResNeXt-101+SE-block+MFF+Head	89.42	91.03	92.61	89.57	89.18×10 ⁶	28.12
ResNeXt WSL+SE-block+MFF+Head	93.11	93.42	94.51	93.02	89.24×10 ⁶	31.60
CCI-Net (SE-ResNet61+MFF+Head)	94.32	94.13	94.60	93.57	47.19×10 ⁶	21.79

Bold emphasis words represent our submodules, and bold numbers represent the performance of our approach

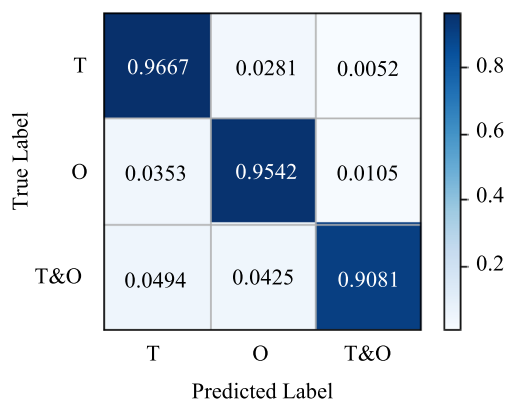


Fig. 8 The confusion matrix of CCI-Net for chromosome cluster classification. T, O, and T and O represent touching, overlapping, and touching and overlapping clusters, respectively

Moreover, our CCI-Net achieved 94.60% identification accuracy, which exceeded the ResNeXt WSL module. In addition, the parameters and time are at a moderate level compared to other methods. Therefore, we can infer that our CCI-Net exhibits excellent performance in the chromosome cluster identification task, and this can directly prove that the combination of SE-block and MFF module and the identification head extract more discriminative features for classification.

To further explore the identification performance of CCI-Net for each cluster type, a confusion matrix is presented in Fig. 8. As illustrated in Fig. 8, the accuracy of the CCI-Net on each type of chromosome cluster in the test dataset is 96.67%, 95.42%, and 90.81%. For touching and overlapping clusters, the significant loss of the confusion matrix was 4.94% of clusters that were misclassified as overlapping ones and the rest were misjudged as touching ones.

Table 3 Comparison experimental results with specifically designed methods. “–” represents that no relevant data is provided

Methods	No. of metaphase images	No. of clusters	ACC (%)	IOU (%)	Dice (%)
Minaee et al. [11]	25	62	91.90	90.72	91.35
Yilmaz et al. [12]	145	–	97.83	98.13	97.74
Altinsoy et al. [20]	508	–	98.94	97.65	98.63
Lin et al. [14]	500	4876	85.00	87.76	86.39
Hu et al. [13]	–	13434	92.72	94.70	93.15
Saleh et al. [15]	–	13434	99.53	99.68	99.21
Song et al. [16]	–	13434	99.16	99.08	99.07
Ours	1148	6950	99.15	99.06	99.01

Bold numbers represent the experimental results of our approach

4.4 Results of automatic progressive segmentation approach

In this section, we present experimental segmentation results in three parts. Firstly, we compare our approach with state-of-the-art methods under three metrics: IOU, Dice, and accuracy. These methods were proposed specifically designed for chromosome segmentation. Table 3 gives the comparison results provided by their own papers. The first three methods were traditional methods and they all applied to separate the entire metaphase chromosome image. Among them, Altinsoy’ algorithm showed better segmentation accuracy of 98.94%. The rest of four methods were grouped as CNN-based methods only used to separate chromosome clusters. We find that Lin’s method obtained the lowest results with 85% accuracy. The other neural networks modified from U-Net architecture were applied to segment semi-synthetically overlapped chromosomes. Although they all performed with high accuracy that exceeded 96%, they just were utilized to handle one type of chromosome clusters, and thus they may not be suitable for the whole metaphase chromosome image segmentation. In contrast, we employ a segmentation scheme combining deep learning with traditional methods to segment the entire metaphase chromosome image progressively. On the basis of the high classification accuracy achieved by the above CCI-Net, we adopt different methods to segment different clusters, thus greatly improving the segmentation accuracy. As we can see in Table 3, compared to the abovementioned algorithms, our

segmentation method outperforms all other models with an accuracy of 99.15%, IOU of 99.06%, and Dice of 99.01%.

Then, to further demonstrate the effectiveness of our segmentation method, we showed the compared results with segmentation baseline models (Mask-RCNN [34], U-Net [30]). All segmentation experiments were implemented on our chromosome datasets, and the results are shown in Table 4. These models were carried out from scratch with two chromosome datasets: the entire metaphase images and chromosome clusters. For simplicity, we used data #1 and data #2 to represent metaphase chromosome images and chromosome clusters, respectively.

As illustrated in Table 4, for data #1, Mask-RCNN and U-Net yielded 81.61% and 81.78%, respectively. But for data #2, these two methods had only obtained the accuracy of 60.57% and 65.81%, respectively. It shows that these two methods cannot perform well on the cluster segmentation task. In contrast, for data #1, our proposed method achieves the accuracy, dice, and IOU of 99.15%, 99.01%, and 99.06%, respectively. Meanwhile the accuracy of 99.32%, IOU of 99.51%, and Dice of 99.17% are obtained from data #2. Furthermore, we compare the results between one-stage and three-stage methods and find that the one-stage method has a similar result with Mask-RCNN. As a side note, the one-stage method applies the rough segmentation using thresholding-based and geometric-based methods. It demonstrates that our segmentation algorithm performs an encouraging segmentation result for either data #1 or data #2.

Table 4 Comparison results with segmentation baseline models on our datasets

Methods	Data #1			Data #2		
	ACC (%)	IOU (%)	Dice (%)	ACC (%)	IOU (%)	Dice (%)
Mask-RCNN	81.62	85.76	81.13	60.57	62.15	60.51
U-Net	81.78	86.12	81.57	65.81	65.39	64.76
One-stage method	81.71	85.80	81.21	60.31	61.68	60.35
Three-stage method (ours)	99.15	99.06	99.01	99.32	99.51	99.17

The last two lines are bolded to represent the segmentation performance of our one-stage and three-stage methods respectively

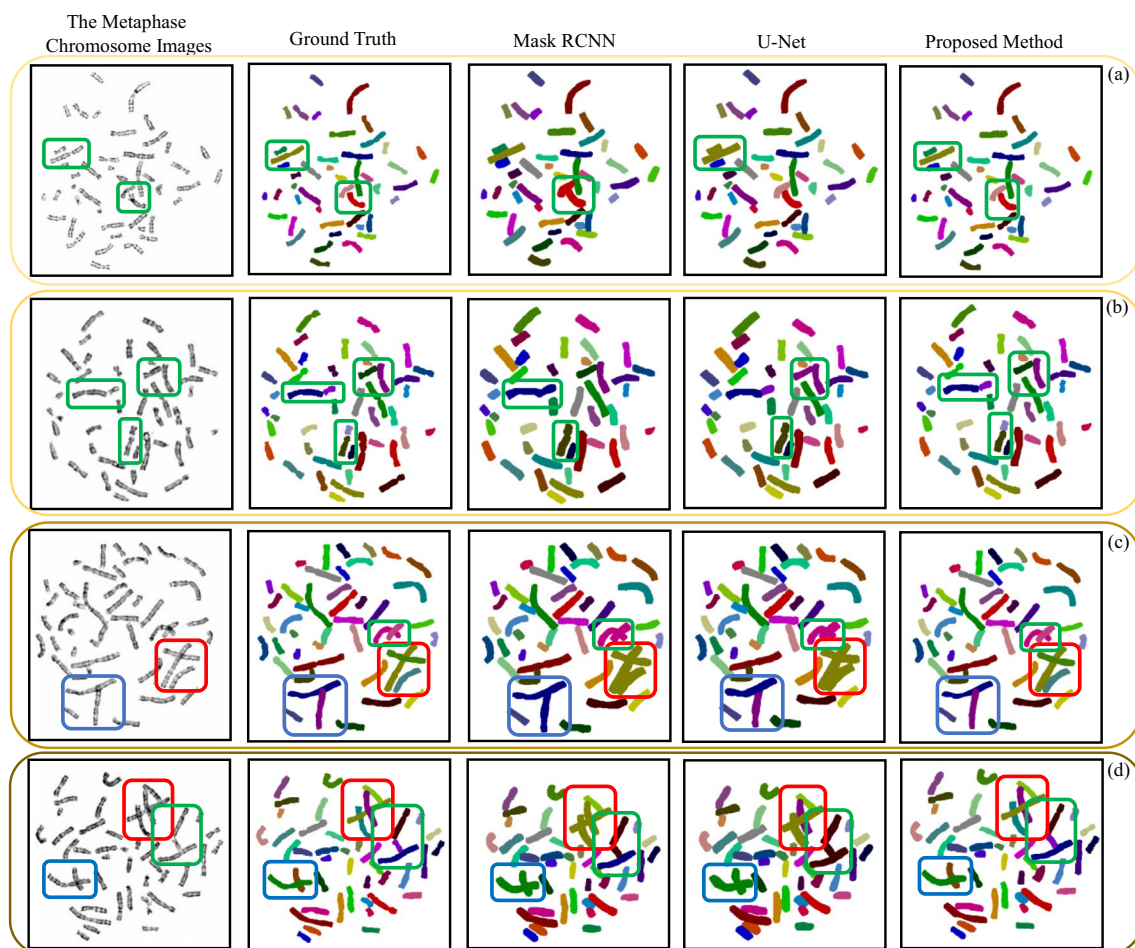


Fig. 9 Visualization of performance of different segmentation approaches. **a–d** Four sets of experimental results. Green, blue, and red boxes in each image delineate touching, overlapping, and touching and overlapping clusters

Finally, to visualize the performance of segmentation methods more intuitively, we show the visualization results of Mask-RCNN, U-Net, and our proposed method in Fig. 9. To better show the segmentation results, we marked 46 chromosomes with different colors. Even though as the popular segmentation baseline methods, Mask-RCNN and U-Net have made many excellent results in natural image processing, they perform poorly at touching or overlapping chromosome separation. Synthesizing four sets of results, we can see that two baseline methods are almost unable to separate three types of chromosomes correctly, as shown by the green, red, and blue boxes in Fig. 9. U-Net sometimes can segment partially overlapping clusters, but it ignores the overlapped area indicated by the blue box in Fig. 9(c). Compared to others, our proposed method enables us to separate each type of clusters correctly and achieves better segmentation performance.

4.5 Discussion

In this section, we further analyze the causes of experimental results. To our knowledge, karyotyping is the gold

standard for diagnosing chromosome disease. Automatic chromosome segmentation is the first challenge problem for karyotyping. This paper proposes a three-stage progressive segmentation approach for chromosome separation automatically. Compared with other methods, the most distinctive highlights of our approach include (1) an automatic progressive chromosome segmentation is adopted to separate the entire metaphase images; (2) CCI-Net is introduced to classify chromosome clusters as three types; and (3) for better performance of chromosome instance segmentation, we design respective separation methods to segment each type of the cluster.

Among all the compared models, for segmentation baseline methods shown in Table 4 and customized CNNs-based methods in Table 3, due to the unpredicted shapes of chromosome clusters, these networks maybe cannot grasp discriminative features to segment all chromosome clusters. Meanwhile, for specifically designed traditional segmentation methods in Table 3, uneven staining and manually handcrafted features contribute to

drawing the segmentation performance down. To address the shortcomings of the above methods and achieve better segmentation results, our approach applies a progressive segmentation style to separate chromosomes gradually. Two main aspects lead to better segmentation results. On the one hand, CCI-Net is employed to identify types of clusters.

It is a vital prerequisite for chromosome segmentation. Compared to other classification approaches, CCI-Net mainly focuses on extracting discriminative features based on SE-ResNet61 and MFF modules to implement dual feature enhancement. Table 2 shows the effectiveness of these two modules and the high classification accuracy of CCI-Net. On the other hand, compared with other chromosome segmentation methods, we design a specific separation method for each type of chromosome cluster according to their characteristics of clusters. Final separation results exhibit superior performance shown in Table 3 and Table 4.

However, for some cases where chromosomes with large bending angles overlap together, our overlapping cluster segmentation method may not be able to handle them. Our future work will address this problem.

5 Conclusion

We have proposed an automatic progressive chromosome segmentation method in this paper. It aims to implement chromosome segmentation on the basis of high accuracy of chromosome cluster identification and consists of three stages. Firstly, a thresholding-based method is applied to divide the whole metaphase chromosome image into foreground and background roughly. After real chromosome clusters are judged by geometric characteristics, CCI-Net is introduced to identify three types of clusters. The core idea of this network is to make full use of multi-scale representations to enrich the final features, so as to achieve high accuracy cluster identification. Finally, we present different separation methods for different types of clusters. For touching clusters, we use cut-points and the feature of junction area to find a reasonable separation path, and for overlapping ones, we combine U-Net with traditional image processing to separate them, and touching and overlapping ones are segmented using the combination of these two separation methods. To verify the effectiveness and robustness of the proposed method, we conduct extensive experiments conducted on chromosome clusters and the entire metaphase images. The results demonstrate that the designed CCI-Net obtains 94.60% identification accuracy, and our segmentation approach outperforms state-of-the-art methods with high segmentation accuracy of 99.15%.

In summary, our network shows excellent adaptability and performance in automatic chromosome segmentation.

Acknowledgements This work is supported by National Major Scientific Research Instrument Development Project (6222780062), Science and Technology Commission of Shanghai Municipality under Grant (20142200240), and the National Key Scientific Instruments and Equipment Development Program of China (2013YQ03065101).

References

1. Tjio JH, Levan A (1956) The chromosome number in man. *Hereditas* 42:1–6
2. Conference D (1960) A proposed standard system of nomenclature of human mitotic chromosomes. *Lancet* 1:1063–1065
3. O'Connor C (2008) Karyotyping for chromosomal abnormalities. *Nature Educ* 1:27
4. Liu X, Fu L, Lin CW et al (2022) SRAS-net: low-resolution chromosome image classification based on deep learning[J]. *IET Syst Biol* 16(3–4):85–97
5. Natarajan AT (2002) Chromosome aberrations: past, present and future. *Mutat Res/Fund Mol Mech Mutagen* 504(1):3–16
6. Patterson D (2009) Molecular genetic analysis of down syndrome. *Hum Genet* 126(1):195–214
7. Arora T, Dhir R (2016) A review of metaphase chromosome image selection techniques for automatic karyotype generation. *Med Biol Eng Comput* 54(8):1147–1157
8. Lerner B (1998) Toward a completely automatic neural-network-based human chromosome analysis. *IEEE Trans Syst Man Cybern B: Cybern* 28(4):544–552
9. Wang X, Zheng B, Wood M et al (2005) Development and evaluation of automated systems for detection and classification of banded chromosomes. *J Phys D Appl Phys* 38(15):2536–2542
10. Grisan E, Poletti E, Ruggeri A (2009) Automatic segmentation and disentangling of chromosomes in Q-band prometaphase images. *IEEE Trans Inf Technol Biomed* 13(4):575–581
11. Minaee S, Fotouhi M, Khalaj BH (2014) A geometric approach for fully automatic chromosome segmentation. 2014 IEEE Signal Processing in Medicine and Biology Symposium (SPMB), pp 1–6
12. Yilmaz IC, Jie Y, Altinsoy E et al (2018) An improved segmentation for raw G-band chromosome images. 2018 5th International Conference on Systems and Informatics (ICSAI)
13. Hu RL, Karnowski J, Fadely R, Pommier JP (2017) Image segmentation to distinguish between overlapping human chromosomes. arXiv preprint arXiv:1712.07639
14. Lin C, Zhao G, Yin A et al (2020) A multi-stages chromosome segmentation and mixed classification method for chromosome automatic karyotyping[C]. In: International Conference on Web Information Systems and Applications. Springer International Publishing, Cham, pp 365–376
15. Saleh HM, Saad NH, Isa NAM (2019) Overlapping chromosome segmentation using U-Net: convolutional networks with test time augmentation - ScienceDirect. *Procedia Comput Sci* 159:524–533
16. Song S, Bai T, Zhao Y et al (2022) A new convolutional neural network architecture for automatic segmentation of overlapping human chromosomes. *Neural Process Lett* 54(1):285–301
17. Liu X, Wang S, Lin JCW, Liu S (2022) An algorithm for overlapping chromosome segmentation based on region selection. *Neural Comput Applic* 1–10

18. He K, Zhang X, Ren S et al (2016) Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
19. Hu J, Shen L, Albanie S et al Squeeze-and-excitation networks. 2018 IEEE/CVF Conf Comput Vis Pattern Recognit 42(2018):2011–2023
20. Altinsoy E, Yang J, Yilmaz C (2020) Fully-automatic raw G-band chromosome image segmentation. IET Image Process 14(9):1920–1928
21. Arachchige AS, Samarabandu J, Knoll J et al (2010) An image processing algorithm for accurate extraction of the centerline from human metaphase chromosomes. Image Processing (ICIP), 2010 17th IEEE International Conference on Image Processing, pp 3613–3616
22. Karvelis PS, Fotiadis DI, Syrrou M et al (2005) Segmentation of chromosome images based on a recursive watershed transform. Third Eur Med Biol Eng Conf 11:1727–1983
23. Liang JI (1989) Intelligent splitting in the chromosome domain - ScienceDirect. Pattern Recognit 22(5):519–532
24. Srisang W, Jaroensutasinee K, Jaroensutasinee M (2006) Segmentation of overlapping chromosome images using computational geometry. Walailak J Sci Technol 3(2):181–194
25. Tanvi T, Dhir R (2014) An efficient segmentation method for overlapping chromosome images. Int J Comput Appl 95(1):29–32
26. Wayalun P, Chomphuwiset P, Laoprasitthachon N et al (2013) Images enhancement of G-band chromosome using histogram equalization, OTSU thresholding, morphological dilation and flood fill techniques. In: Computing and Networking Technology (ICCNT), 2012 8th International Conference on. IEEE, pp 163–168
27. Lin C, Zhao G, Yin A et al (2021) A novel chromosome cluster types identification method using ResNeXt WSL model. Med Image Anal 69:1–9
28. He A, Wang K et al (2022) Progressive multi-scale consistent network for multi-class fundus lesion segmentation. 2022 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1–15 4th International Conference on Big Data and Machine Learning (BDML 2021). 2021: 1-10
29. Chang L, Wu KJ et al (2021) Automatic segmentation of the whole G-band chromosome images based on mask R-CNN and geometric features. 2021 5th International Conference on Advances in Image Processing (ICAIP 2021) and 4th International Conference on Big Data and Machine Learning (BDML 2021), pp 1–10
30. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention, pp 234–241
31. He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: surpassing human-level performance on imagenet classification. Proceedings of the IEEE international conference on computer vision, pp 1026–1034
32. Huang G, Liu Z, Laurens V et al (2017) Densely connected convolutional networks. Proceedings of the IEEE Conference on Computer Vision and Pattern recognition, pp 4700–4708
33. Xie S, Girshick R, Dollár P et al (2017) Aggregated residual transformations for deep neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1492–1500
34. He K, Gkioxari G, Dollár P et al (2017) Mask R-CNN. Proceedings of the IEEE international conference on computer vision, pp 2961–2969

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

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.



Ling Chang received the B.E. degree from the School of Civil Aviation University of China. She is currently pursuing the Ph.D. degree in the Department of Automation, School of Electronic Information and Electrical Engineering, Shanghai Jiaotong University. She focuses on medical image processing, especially digital pathological image and chromosome image.



Kaijie Wu is an Associate Professor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He received his Ph.D. degree in Biomedical Engineering from Tianjin University, Tianjin, China, in 2006. His current research explores biomedical optical imaging, medical information processing, and pattern recognition.



Hao Cheng received the B.E. degree from the School of Shanghai Jiaotong University. He is currently pursuing the Ph.D. degree in the Department of Electronic Engineering, School of Electronic Information and Electrical Engineering, Shanghai Jiaotong University. He focuses on the process of medical image, especially digital pathological image.



Chaocheng Gu is currently an Associate Professor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He received his bachelor degree from Shandong University, Jinan, China, in 2007, and the Ph.D. degree in Mechanical Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013. His current research interests include industry robotics, machine vision, and man-machine interfaces.



Yudi Zhao is currently a post-doctor at School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. She received her Ph.D. degree in Control Science and Engineering from Donghua University, Shanghai, China, in 2022. She has been a Joint Ph.D. Student with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA, from 2017 to

2020. Her scientific interests include deep learning, artificial intelligence, and computer vision.



Cailian Chen (Member, IEEE) received the B.Eng. and M.Eng. degrees in automatic control from Yanshan University, Qinhuangdao, China, in 2000 and 2002, respectively, and the Ph.D. degree in control and systems from the City University of Hong Kong, Hong Kong, in 2006. She was a Senior Research Associate with the City University of Hong Kong, in 2006 and a Postdoctoral Research Associate with the University of Manchester, Manchester, UK, from 2006 to 2008. In 2008, she

joined the Department of Automation, Shanghai Jiao Tong University, Shanghai, China, as an Associate Professor, where she is currently a Full Professor. Her current research interests include wireless sensor networks and industrial applications, computational intelligence and distributed situation awareness, cognitive radio networks and system design, Internet of vehicles and applications in intelligent transportation, and distributed optimization.