Adversarial Multiscale Feature Learning for Overlapping Chromosome Segmentation

Liye Mei, Student Member, IEEE, Yalan Yu, Yueyun Weng, Xiaopeng Guo, Student Member, IEEE, Yan Liu, Du Wang, Sheng Liu, Fellow, IEEE, Fuling Zhou, and Cheng Lei

Abstract—Chromosome karyotype analysis is of great clinical importance in the diagnosis and treatment of diseases, especially for genetic diseases. Since manual analysis is highly time and effort consuming, computer-assisted automatic chromosome karyotype analysis based on images is routinely used to improve the efficiency and accuracy of the analysis. Due to the strip shape of the chromosomes, they easily get overlapped with each other when imaged, significantly affecting the accuracy of the analysis afterward. Conventional overlapping chromosome segmentation methods are usually based on manually tagged features, hence the performance of which is easily affected by the quality, such as resolution and brightness, of the images. To address the problem, in this paper, we present an adversarial multiscale feature learning framework to improve the accuracy and adaptability of overlapping chromosome segmentation. Specifically, we first adopt the nested U-shape network with dense skip connections as the generator to explore the optimal representation of the chromosomes images by exploiting multiscale features. Then we use the conditional generative adversarial network (cGAN) to generate images similar to the original ones, the training stability of which is enhanced by applying the least-square GAN objective. Finally, we employ Lovász-Softmax to help the model converge in a continuous optimization setting. Comparing with the established algorithms, the performance of our framework is proven superior by using public datasets in eight evaluation criteria, showing its great potential in overlapping chromosome segmentation.

Index Terms—Overlapping Chromosome Segmentation; Conditional Generative Adversarial Network; Nested U-shape Network; Multiscale Feature Learning; Lovász-Softmax.

I. INTRODUCTION

HUMAN chromosome karyotype analysis is of great diagnostic and prognostic value in diseases. It is usually performed in clinical diagnosis, cancer cytogenetic, and genetic abnormalities detecting such as Edwards syndrome and Down syndrome [1, 2]. The morphology of chromosomes, such as the excess or missing, or the structural defects of specific chromosomes can be directly linked to corresponding diseases, hence chromosome karyotype analysis based on images plays a critical role in routine disease diagnosis and treatment [3]. Fig. 1 illustrates the process of chromosome karyotype analysis, which consists of two main steps: segmentation and classification. The performance of the segmentation can directly influence the accuracy of the classification afterward. As shown in the red circles in Fig. 1(a) and Fig. 1(b), the overlapping chromosome segmentation is even more challenging due to the ambiguity in the overlapping regions, which can greatly influence the accuracy of the chromosome karyotype analysis. Hence, segmenting overlapping chromosomes with high efficiency and accuracy can significantly enhance the overall performance of the chromosome karyotype analysis.

Since manual segmentation is both time and effort consuming, and the accuracy highly depends on the experiences of the analyst, over the past few decades, many algorithms have been proposed to automatically segment the chromosome from the images on computers. According to their principles, these segmentation methods can be roughly classified into two categories: heuristic methods and learning-based methods. For heuristic methods, they utilize manually tagged features, such as contour, pixels, geometric features, to perform segmentation [4-6]. The representative examples include the global context and variant analysis method [7], contour analysis method [8], edge detection method [9], thresholding strategies [10, 11], and watershed and clustering method [12]. These methods can achieve impressive segmentation results when features are properly tagged. However, they are very sensitive to the shape and overlapping regions of the chromosomes. Moreover, since they do not consider the untagged features, the performance and applicability of these methods are limited. While, for learning-based methods, they usually apply machine learning techniques to mine potential information from the images to perform chromosome segmentation and medical image analysis[13, 14]. Two representative examples are fully convolution network (FCN) method [15, 16], and U-shape Network (UNet) [17, 18].
These methods can independently conduct chromosome segmentation when being trained. However, limited by the architecture of the network, current learning-based methods only utilize several layers’ features and they do not take advantage of multiscale features to adapt different chromosome scales. Hence, they do not perform well in overlapping chromosome segmentation.

In this paper, considered the various scales and overlapping regions of chromosome images, we demonstrate an adversarial multiscale feature learning (AMFL) framework which employs nested U-shape convolutional neural network (NestedUNet) [19], conditional generative adversarial network (cGAN) [20], and Lovász-Softmax [21] for overlapping chromosome segmentation. Specifically, NestedUNet consists of UNets [22] of varying depths and owns dense skip connectivity, making it capable of synthesizing multiscale feature maps for segmentation. Hence, our AMFL framework utilizes NestedUNet to explore the optimal representation of chromosome images by exploiting multiscale features and features fused. Meanwhile, we use cGAN to push the output distributions close to the ground truths for its success in computer vision tasks, such as blind motion deblurring[23], image fusion [24], and image inpainting [25]. Finally, to optimize the performance of the discriminatively trained overlapping chromosome segmentation, we apply Lovász-Softmax, which is based on the convex lovász extension of the submodular loss, as the segmentation loss to achieve superior segmentation. Hence, our AMFL framework utilizes NestedUNet to explore the optimal representation of chromosome images by exploiting multiscale features and features fused. Meanwhile, we use cGAN to push the output distributions close to the ground truths for its success in computer vision tasks, such as blind motion deblurring[23], image fusion [24], and image inpainting [25]. Finally, to optimize the performance of the discriminatively trained overlapping chromosome segmentation, we apply Lovász-Softmax, which is based on the convex lovász extension of the submodular loss, as the segmentation loss to achieve superior chromosome segmentation performance and higher index scores to the traditional cross-entropy (CE) loss. Additionally, we utilize the least-square GAN objective [26] to replace the original GAN loss in the overlapping chromosome segmentation task to stabilize the training and avoid model collapsing. To verify the feasibility of our method, we carry out extensive experiments to compare the performance of our AMFL framework with others. Results show the superiority of our AMFL framework in this work in terms of visual perception analysis and quantitative score comparison.

II. METHODS

In this section, we introduce the architecture of our AMFL framework, the quantitative criteria to evaluate the performance of the chromosome segmentation, and the references which are used to validate the feasibility of our method.

A. Network architecture

As depicted in Fig. 2, our AMFL framework consists of two modules: a generator and a discriminator. The generator is responsible for exploiting multiscale features for segmentation by producing “fake” chromosome images. While the discriminator is to distinguish the “fake” images from the “real” ones by adversarial learning. Once the discriminator is “fooled” by the generator, the network is ready to segment chromosomes with high accuracy.

1) Generator

As shown in Fig. 2(a), we adopt the advanced NestedUNet [19] as the generator, which consists of an encoder and a decoder. It takes a source chromosome image as the input and outputs a multiclass one-hot map. Specifically, each node in the graph represents a nested convolution block, the downward arrows, upward arrows, and dotted arrows indicate 2 × 2 max-pooling, 2×2 up-sampling, and skip connections, respectively. The skip connections merge the encoding and decoding features in the channel dimension by tensor concatenation, enabling dense feature propagation. To better comprehend the network connectivity, we denote \( x^{i,j} \) the output of the node \( X^{i,j} \). It can be formulated as:

\[
x^{i,j} = \begin{cases} 
N(D(x^{i,j-1})), & j = 0 \\
N([x^{i,j}]_1^{l_{i,j}}, U(x^{i,j-1} - 1)), & j > 0 
\end{cases}
\]

where function \( N(\cdot) \) denotes a Nested convolution block operation, \( D(\cdot) \) and \( U(\cdot) \) denote a down-sampling layer and an up-sampling layer respectively, and \([\cdot]\) denotes the concatenation layer. Intuitively, we can see that the nodes at the level of \( j = 0 \) receive only one input from the previous layer of the encoder, whereas the nodes at the level of \( j \geq 1 \) receive the up-sampled output of \( j + 1 \) nodes from the lower skip connection and all the outputs of the previous \( j \) nodes in the same skip connection. Therefore, a dense skip connection is constructed and multiscale features are integrated to provide better feature representation for the segmentation of overlapping chromosome regions with different scales. Meanwhile, in order to better describe network parameters, the number of filters is defined as: \( f = \{64, 128, 256, 512, 1024\} \), and the number of input channels, middle channels, and output channels of a nested convolution block is defined as follows:

\[
I_y = \begin{cases} 
f(i), & j = 0 \\
f(i) \cdot j + f(i), & j > 0 
\end{cases}, \quad M_y = f(i), \quad O_y = f(i)
\]

where \( I_y \), \( M_y \), \( O_y \) are the input channels, middle channels, and output channels of the \( y \)th node, respectively. Note that

---

**Fig.1.** Chromosome karyotype analysis process. (a) The image of meta chromosome. (b) Segmentation map of the chromosome image. (c) The sorted karyotype. The red circles make the overlapping regions.
middle channels are the output of the first convolution layer, and also the input of the second convolution layer in the nested convolution block. To describe the network structure in detail, we denote the convolution layer, Batch Norm layer, and Rectified Linear Unit [27] as Conv, BN, and ReLu, respectively. The Nested Conv block is a Conv-BN-ReLu with a filter size of $3 \times 3$, padding of 1 and stride of 1, which aims to keep the size of the feature map consistent after each convolution operation. The last node of the feature map is mapped into a confidence map using Softmax operation for producing one hot map.

3) Discriminator As shown in Fig. 2(b), we leverage a simplified fully convolutional neural network [16, 24] as the discriminator, which is able to push the outputs distribution closer to the ground truths, making the generator produce high-confident segmentation maps. Specifically, it consists of five convolution layers with a filter size of $4 \times 4$ kernel and {64,128,256,512,1} channels in the stride of 2. Each convolution layer is followed by a Leaky-ReLu parameterized by 0.2 except for the last layer, which is followed by a sigmoid operation for discriminating “real” or “fake” images. Finally, it’s worth noting that the input of the discriminator is multi-channel images created by concatenating the source images and the segmented images (generator produced) in the channel dimension, aiming to provide prior information for better-discriminating features. The generator and discriminator are alternately trained using the objective function represented as follows.

### B. Objective Function

1) Lovász-Softmax It can optimize the Jaccard index in the continuous optimization framework [21]. Specifically, this method can substantially improve the accuracies of semantic segmentation by optimizing the correct loss during training. Therefore, we choose Lovász-Softmax as the loss of the generator, which can be simplistically defined as:

$$L_{Lovász-Softmax} = \frac{1}{C} \sum_{c \in C} \frac{1}{\Delta J_{c}} \sum_{i} \Delta J_{c}(m_{i}) \Delta_{c}$$

(3)

where $m_{c}$ is a vector of pixel errors for class $c \in C$ aiming to construct the loss surrogate to $\Delta J_{c}$, it’s defined by:

$$m_{c} = \begin{cases} 1 - f_{i}(c) & \text{if } c = y_{i} \\ f_{i}(c) & \text{otherwise} \end{cases}$$

(4)

In the equation, $y$ is ground truth and $f_{i}(c)$ is the predicted scores of the model that is mapped to probabilities through a Softmax unit. $\Delta J_{c}$ is the set function encoding a submodular Jaccard loss for class $c$, indicating a set of mispredictions. Specially, $\Delta$ is the surrogate for the minimization of $\Delta$ with first-order continuous optimization, and the elementary operations involved in the calculation of $\Delta$ (sort) are differentiable.

2) GAN loss First and foremost, we need to choose an appropriate loss function for training our AMFL framework. It is well known that the regular GAN [28] loss is always difficult to convergence and can suffer from model collapsing. We therefore adopt the least-squares generative adversarial
networks (LSGAN) [26] as the loss function in our work, which is more stable and can achieve better segmentation results. It is defined as:

\[ L_{\text{LSGAN}}(D) = \mathbb{E}_{x \sim P_{\text{data}(\cdot)}} [(D(i, y) - 1)^2] + \mathbb{E}_{y \sim P_{\text{data}(\cdot)}} [(D(i, G(i)) - 1)^2] \]  

(5)

The adversarial learning process is also optimized through the LSGAN, which is formulated as:

\[ L_{\text{LSGAN}}(G) = \mathbb{E}_{x \sim P_{\text{data}(\cdot)}} [(D(i, G(i)) - 1)^2] \]  

(6)

Furthermore, in order to make the segmentation map as close as possible to the ground truth, we adopt Lovász-Softmax loss for supervised segmentation. The objective function for AMFL therefore can be defined as:

\[
\begin{align*}
\min_D L(D) &= L_{\text{LSGAN}}(D) \\
\min_G L(G) &= L_{\text{LSGAN}}(G) + \lambda L_{\text{Lovász-Softmax}}
\end{align*}
\]  

(7)

where \( \lambda \) controls the relative importance of the two objective functions. Empirically, we set \( \lambda \) to 10 in our work.

C. Evaluation metrics

To quantitatively evaluate the performance of our method, we select eight evaluation metrics including the pixels accuracy (Acc) [29], dice coefficient (Dice) [30], intersection over union (IoU) [29], precision, recall, false-negative rate (FNR), false-positive rate (FPR) [29] and Hausdorff distance (Hausdorff) [29], which are briefly introduced below. For convenience, we use \( O \) to denote the output segmentation image, and \( G \) to indicate the ground truth. Moreover, the index ranges over the interval \([0,1]\) except Hausdorff.

1) Acc It indicates the pixel accuracy of the predicted results in the segmentation. In other words, it represents the proportion of pixels in an image that is correctly predicted. The Acc is calculate as:

\[ \text{Acc} = \frac{\sum_{i=0}^{c} \sum_{j=0}^{e} P_{ij}}{\sum_{i=0}^{c} \sum_{j=0}^{e} P_{ij}} \]  

(8)

Here \( P_{ij} \) means the numbers of pixels that are classified as pixel \( j \) but actually belongs to pixel \( i \), and \( c \) is the categories.

2) Dice This metric represents the similarity of predicted image \( O \) to the ground truth \( G \). The Dice is calculated as:

\[ \text{Dice}_c = 2 \frac{|G_c \cap O_c|}{|G_c| + |O_c|} \]  

(9)

where \( |G| \) and \( |O| \) represent the numbers of elements in the arrays.

3) IoU This metric represents the intersection area between the predicted image \( O \) and the ground truth \( G \), the IoU can be calculated as follows:

\[ \text{IoU}_c = \frac{|G_c \cap O_c|}{|G_c| + |O_c| - |G_c \cap O_c|} \]  

(10)

4) Precision It indicates how reliable the prediction is. This metric can be calculated as:

\[ \text{Precision}_c = \frac{TP_c}{TP_c + FP_c} \]  

(11)

where \( TP_c \) represents the true positives which means the pixels correctly predicted to belong to class \( c \), while \( FP_c \) represents the false positives, indicating the pixels predicted as class \( c \) but do not actually belong to class \( c \).

5) Recall It indicates how sensitive the prediction is. So, it is also called sensitivity, which can be calculated as follows:

\[ \text{Recall}_c = \frac{TP_c}{TP_c + FN_c} \]  

(12)

where the \( FN_c \) represents the false negatives, meaning the pixels predicted as not class \( c \) but actually belong to class \( c \).

6) FNR It’s also called the under-segmentation rate, which measures the proportion of the positive classes that is predicted to be negative. It is defined by:

\[ \text{FNR}_c = \frac{FN_c}{TP_c + FN_c} \]  

(13)

7) FPR It’s also called the over-segmentation rate, measuring the proportion of the negative classes that is predicted to be positive. This metric is calculated as:

\[ \text{FPR}_c = \frac{FP_c}{FP_c + TN_c} \]  

(14)

where \( TN_c \) represent the true negatives which mean the pixels that are correctly predicted not to belong to class \( c \).

8) Hausdorff It represents the shape similarity between the predicted images \( O \) and the ground truths \( G \). It is calculated by:

\[ \text{Hausdorff}_c = \max \left\{ \sup_{x \in G_c} \inf_{y \in O_c} d(x, y), \sup_{y \in O_c} \inf_{x \in G_c} d(x, y) \right\} \]  

(15)

where \( d(\cdot) \) represents the Euclidean distance between the pixel points \( x \) and \( y \). The smaller the Hausdorff distance is, the greater the similarity between the predicted segmentation maps and the ground truth is.

Note that for each metric, a higher value indicates better performance, except for FNR, Hausdorff and FPR, where a lower score gives a better segmentation result.

D. Baselines and implementation

We validate the effectiveness of our method by comparing it with ten recently state-of-the-art algorithms, including efficient neural network (ENet)[31], bilateral segmentation network
version 1 (BiSeNetV1) [32], BiSeNetV2 [33], DeepLabV3+ [34], faster fully convolutional network (FastFCN) [35], U-shape network (UNet) [22], recurrent residual UNet (R2UNet) [36], attention UNet (AttUNet) [37], recurrent residual attention UNet (R2AttUNet) [38], and nested UNet (NestedUNet) [19]. Among them, ENet, BiSeNetV1, and BiSeNetV2 are small-scale models, which usually have smaller network scales and higher inference speed. While the others are large-scale models, which usually have more complex network structures and can learn more potential semantic features. The above methods are used as baselines to evaluate the performance of our method comprehensively.

E. Selection of the objective function and generator

1) Selection of the objective function In order to show the superiority of using Lovász-Softmax as the loss for overlapping chromosome segmentation, as shown in Fig.3, we draw average metric graphs for prevalent losses on all testing sets with all the methods. Intuitively, it can be seen that our method with Lovász-Softmax outperforms all baseline methods in all the metrics. Moreover, it is clear that all baseline methods with Lovász-Softmax also show a leading scoring trend against other losses, indicating that Lovász-Softmax is effective and optimal for overlapping chromosome segmentation feature extraction.

2) Selection of the generator In Table I, we evaluate the performance of our method with different generator networks in terms of all the quantitative indicators. Obviously, we can clearly see that the framework with NestedUNet as the generator network is better than other configurations.

### III. EXPERIMENTS AND RESULTS

A. Preliminary Preparation

1) Data Preparation and Preprocessing use the Pommier’s overlapping chromosome datasets [39, 40] to verify the effectiveness of the present method. The dataset contains a total number of 13,434 grayscale images with a resolution of 94 × 93. For each image, there is a corresponding ground truth, in which each pixel represents an object class. In the segmentation map, class labels of 0, 1, 2 and 3 are denoted as the background (shown as black), non-overlapping regions of the first chromosome (shown as red), non-overlapping regions of the second chromosome (shown as green) and overlapping regions of chromosome (shown as blue), respectively. To match the images with our network, we pad the images to 128 × 128. The padding value of input images and ground truths are set as 255 and 0 to be consistent with the background of the original images, respectively. We divide the datasets into two subsets: 80% for training (a total number of 10747 images) and the rest 20% for testing (a total of 2867 images).

2) Implementation In the training stage, all input images are normalized to the range of 0-1, and the batch size is set to 64. We optimize the generator and the discriminator alternately, both applying the Adam solver with a fixed learning rate of 0.0002, and momentum parameters $\beta_1=0.5, \beta_2=0.999$. We train our framework from scratch with the training sets to produce the “optimized” model after 135 epochs. For the inference stage,
we use the well-trained framework to segment the images. All the experiments are conducted in Pytorch [41] under an Ubuntu OS cloud server with an Intel Xeon(R) CPU E5-2680 v4 @ 2.40GHz, 40 GB of RAM, and a NVIDIA Tesla P40 GPU with 24 GB of memory.

B. Performance

Fig. 4 exhibits some examples of the segmentation results of our method, from which we can see that our method achieves an excellent visual perception result. Moreover, we can also see that the various scales of chromosome individuals and overlapping regions are correctly segmented in all the images, indicating that our methods performs well in the multiscale segmentation task. In order to further highlight the superior performance of our algorithm, we show confusion matrices of average IoU scores on all the testing sets in Fig. 5. We can see that our method shows better results than other state-of-the-art methods. Through carefully comparison, these quantitative results are consistent with the quantitative results in Section II.E, demonstrating the significant superiority of our method, not only for visual perception but also for quantitative analysis.

C. Performance evaluation

1) Visual evaluation

In this section, we visually compare the performance of our method with baseline methods. Fig.6 exhibits the results including difference images using pseudo-color map. Here, the difference images are generated through logical multiplication of the inversed ground truth and corresponding predicted result. Fig.6(a)-(j) are the results acquired using baseline methods with CE loss, while Fig.6(k)-(o) are acquired using the presented method with various loss.
functions. We can see from Fig.6 that our method with Lovász-Softmax or weight-dice loss achieves excellent segmentation results, while the performance of other methods is obviously poor, meaning that these methods do not learn effective features for the overlapping chromosome segmentation. NestedUNet performs the segmentation better than other large-scale models, indicating that multiscale feature learning is helpful for overlapping chromosome segmentation. Furthermore, we can see the difference images acquired with our method is obviously clean than those acquired with other methods, indicating that the cGAN applied in our methods is effective to distinguish the segmented images and ground truths, so as to better learn the features of the chromosomes. Additionally, it is clear that our method with Lovász-Softmax loss segments the images more accurately, where almost every chromosomal region is correctly segmented, compared with other methods. This indicates that the Lovász-Softmax loss helps improve the discrimination ability of our method. In a word, the images shown in Fig. 6 visually show that our AMFL framework can segment overlapping chromosomes with better performance than the baseline methods.

2) Quantitative evaluation In this part, we quantitatively compare the performance of our method with others and show the results in Table II. Here, we use CE loss for baseline methods. For comparison, we use different loss functions for our methods. We can see that our AMFL always show superior results no matter what kind of loss function is used. This indicates that using cGAN to discriminate features can push the output distribution closer to the ground truth so that our method outperforms others in overlapping chromosome segmentation tasks. It is also clear that the small-scale models present almost the worst scoring in terms of Dice, IoU and Hausdorff, while large-scale models reach better scores, suggesting that overlapping chromosome segmentation requires a more complex network structure. Again, NestedUNet achieves the top performance than other methods, quantitatively verified the importance of multiscale feature analysis. Our method with Lovász-Softmax loss achieves the best performance in most of the metrics. In some cases, it gives the second-best performance, demonstrating the effectiveness of using Lovász-Softmax to improve the discrimination ability. It is worth emphasizing that our method has a lower Hausdorff distance score, indicating it retains the shape and structure of the chromosome in the output images. The quantitative results, which are consistent with what we can see from Fig. 6, prove the effectiveness of our method in overlapping chromosome segmentation.

3) Computational Efficiency To evaluate the computation efficiency, we calculate and list the average running time of CPU and GPU when using different methods on all the testing sets in Table III. The methods with small-scale networks consume the least resources. For the methods with large-scale networks, our method takes about 0.03s to segment an image on GPU, which ranks second only behind UNet. Our method spends 0.57s on CPU, ranking second behind only UNet and
AttUNet. The results show that, in addition to the outstanding segmentation performance, our method also performs well in the computational efficiency, suggesting its great potential in real applications.

IV. CONCLUSION AND FUTURE WORK

In this paper, we propose and demonstrate the AMFL framework for overlapping chromosome segmentation. In the network, we adopt NestedUNet to learn multiscale features to detect overlapping regions, utilize cGAN to provide prior information for better discriminating features, employ Lovász-Softmax loss to optimize the segmentation task, and use least square GAN objective to enhance the training stability. To verify the performance of our framework, we compare our method with ten state-of-the-art semantic segmentation methods. The results show that our AMFL is superior to others both visually and quantitatively. In the future research, we intend to collect annotated clinical data and design a fully automatic system for the segmentation, classification and karyotype analysis of chromosomes.

REFERENCES

[1] A. Basu, H. Alder, A. Khiyami, P. Leahy, C. M. Croce, and S. Haldar, “Down syndrome and microRNAs,” *Biomed. Rep.*, vol. 8, no. 1, pp. 11-16, Jan. 2018.
[2] A. Tramontana, B. Hartmann, and E. Hafner, “DiGeorge syndrome chromosome region deletion and duplication: Prenatal genotype - phenotype variability in fetal ultrasound and MRI,” *Prenat. Diagn.*, vol. 39, no. 13, pp. 1225-1234, Dec. 2019.
[3] H. Fang, S. a. Liu, Y. Wang, C. i. Chiang, and C. i. Lin, “Phenotypic features of a microdeletion in chromosome band 20p13: A case report and review of the literature,” *Mol. Genet. Genom. Med.*, vol. 7, no. 7, pp.e00739, Jul. 2019.
[4] S. Minaee, M. Fotouhi, and B. H. Khalaj, "A Geometric Approach For Fully Automatic Chromosome Segmentation." in *Proc. SPIM*, Philadelphia, PA, USA, 2014, pp. 1-6.
[5] P. S. Karvelis, A. T. Tzallas, D. I. Fotiadis, and I. Georgiou, “A Multichannel Watershed-Based Segmentation Method for Multispectral Chromosome Classification,” *IEEE Trans. Med. Imaging*, vol. 27, no. 5, pp. 697-708, May. 2008.
[6] N. Madian, and K. B. Jayanthi, “Analysis of human chromosome classification using centromere position," *Measurement*, vol. 47, no. 1, pp. 287-295, Jan. 2014.
[7] G. Ritter, and L. Gao, "Automatic segmentation of metaphase cells based on global context and variant analysis," *Pattern Recognit.*, vol. 41, no. 1, pp. 38-55, Jan. 2014.
[8] N. Madian, and K. B. Jayanthi, “Overlapped chromosome segmentation and separation of touching chromosome for automated chromosome classification," in *Proc. EMBC*, San Diego, CA, USA, 2012, pp. 5392-5395.
[9] Saiyod, Saiyan & Wayalan, Pichet. "A New Technique for Edge Detection of Chromosome G-BAND Images for Segmentation," in *Proc. SPMB*, Philadelphia, PA, USA, 2014, pp. 1-6.
[10] E. Poletti, F. Zappelli, A. Ruggeri, and E. Grisan, “A review of differential diagnosis of Down syndrome and microRNAs,” *Biomed. Rep.*, vol. 8, no. 1, pp. 11-16, Jan. 2018.
X. Guo, R. Nie, J. Cao, D. Zhou, L. Mei, and K. He, “FuseGAN: A Local adaptive threshold approach to assist automatic chromosome image segmentation,” Lat. Am. Appl. Res., vol. 44, no. 3, pp. 277-282, Jul. 2014.

J. P. Gawande, R. Manohar, J. P. Gawande, R. Manohar, J. P. Gawande, and R. Manohar, “Watershed and Clustering Based Segmentation of Chromosome Images,” in Proc. IJCC, Hyderabad, IND, 2017, pp. 697-700.

Y. Chang, L. Yan, M. Chen, H. Fang, and S. Zhong, “Two-Stage Convolutional Neural Network for Medical Noise Removal via Image Decompensation,” IEEE Trans. Instrum. Meas., vol. 69, no. 6, pp. 2707-2721, Jun. 2020.

S. Singh, and R. S. Anand, “Multimodal Medical Image Fusion using Hybrid Layer Decomposition with CNN-based Feature Mapping and Structural Clustering,” IEEE Trans. Instrum. Meas., vol. 69, no. 6, pp. 3855-3865, Jun. 2020.

Pardo, Esteban, Morgado, Jose, Mario, T., Malpica, and Norberto, “Semantic segmentation of mFISH images using convolutional networks,” Cytom. Part A, vol. 93, no. 6, pp. 620-627, Apr. 2018.

J. Long, E. Shelhamer, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 4, pp. 640-651, Apr. 2015.

H. M. Saleh, N. H. Saad, and N. A. M. Is, “Overlapping Chromosome Segmentation using U-Net: Convolutional Networks with Test Time Augmentation,” Procedia Computer Science, vol. 159, pp. 524-533, Oct. 2019.

R. L. Hu, J. Karnowski, R. Fadely, and J. Pommier, “Image Segmentation to Distinguish Between Overlapping Human Chromosomes,” in Proc. NIPS, Long Beach, CA, USA, 2017.

Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, “UNet++: Redesigned Skip Connections to Exploit Multiscale Features in Image Segmentation,” in IEEE Trans. Med. Imaging, vol. 39, no. 6, pp. 1856-1867, Jun. 2020.

P. Isola, J. Zhu, T. Zhou, and A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks,” in Proc. CVPR, Berkeley, CA, USA, 2017, pp. 5967-5976.

M. Berman, A. R. Triki, and M. Blaschko, “The Lovasz-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks,” in Proc. CVPR, Salt Lake City, UT, USA, 2018, pp. 4413-4421.

O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation.” in Proc. MICCAI, Munich, DEU, 2015, pp. 234-241.

O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, “DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks.” in Proc. CVPR, Salt Lake City, UT, USA, 2018, pp. 8183-8192.

X. Guo, R. Nie, J. Cao, D. Zhou, L. Mei, and K. He, “FuseGAN: Learning to Fuse Multi-Focus Image via Conditional Generative Adversarial Network,” IEEE Trans. Multimedia, vol. 21, no. 8, pp. 1982-1996, Aug. 2019.

Z. Yuan, H. Li, J. Liu, and J. Luo, “Multiview Scene Image Inpainting for Intelligent Vehicles Based on Conditional Generative Adversarial Networks,” IEEE Trans. Veh. Technol., vol. 5, no. 2, pp. 314-323, June, 2020.

X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, “Least Squares Generative Adversarial Networks,” in Proc. ICCV, Venice, ITA, 2017, pp. 2813-2821.

A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models.” in Proc. ICML, Atlanta, GA, USA, 2013, pp. 3.

A. Radford, L. Metz, and S. Chintala, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks,” in Proc. ICLR, San Diego, CA, USA, 2015.

A. A. Taha, and A. Hanbury, “Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool,” BMC Med. Imag., vol. 15, no. 29, pp. 15-29, Aug. 2015.

K. Srivatsanwattana, D. Sneed, and N. Rajpoot, “A Stochastic Polygons Model for Glndular Structures in Colon Histology Images,” IEEE Trans. Med. Imaging, vol. 34, no. 11, pp. 2366-2378, Nov. 2015.

A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, “E2Net: A Deep Neural Network Architecture for Real-Time Semantic Segmentation,” in Proc. CVPR, Las Vegas, NE, USA, 2016.

C. Yu, J. Wang, C. Peng, C. Guo, G. Yu, and N. Sang, “BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation,” in Proc. ECCV, Munich, DEU, 2018, pp. 325-341.