*

Chromosome Detection in Metaphase Cell Images Using Morphological Priors

Jun Wang, Chengfeng Zhou, Songchang Chen, Jianwu Hu, Minghui Wu, Xudong Jiang *Fellow*, *IEEE*, Chenming Xu, and Dahong Qian, *Senior Member, IEEE*

***Abstract*—Reliable chromosome detection in metaphase cell (MC) images can greatly alleviate the workload of cytogeneticists for karyotype analysis and the diagnosis of chromosomal disorders. However, it is still an extremely challenging task due to the complicated characteristics of chromosomes, e.g., dense distributions, arbitrary orientations, and various morphologies. In this paper, we propose a novel rotated-anchor-based detection framework, named DeepCHM, for fast and accurate chromosome detection in MC images. Our framework has three main innovations: 1) A deep saliency map representing chromosomal morphological features is learned end-to-end with semantic features. This *not only* enhances the feature representations for anchor classification and regression *but also* guides the anchor setting to significantly reduce redundant anchors. This accelerates the detection and improves the performance; 2) A hardness-aware loss weights the contribution of positive anchors, which effectively reinforces the model to identify hard chromosomes; 3) A model-driven sampling strategy addresses the anchor imbalance issue by adaptively selecting hard negative anchors for model training. In addition, a large-scale benchmark dataset with a total of 624 images and 27,763 chromosome instances was built for chromosome detection and segmentation. Extensive experimental results demonstrate that our method outperforms most state-of-the-art (SOTA) approaches and successfully handles chromosome detection, with an *AP* score of 93.53%. Code and dataset are available at: https://github.com/wangjuncongyu/DeepCHM.**

***Index Terms*— Chromosome, rotated object detection, deep learning, karyotyping.**

I. Introduction

Numerical and structural abnormalities of chromosomes (i.e., chromosomal disorders) are the main causes of natural abortion, birth defects (e.g., congenital disability), and genetic diseases (e.g., down syndrome)[1]. Early diagnosis is clinically significant for the prevention of the abovementioned chromosomal diseases. In clinical practice, karyotype analysis[2] is a routine procedure for examining chromosomal disorders. It consists of two main stages, as demonstrated in Fig. 1: 1) metaphase cell (MC) image acquisition and 2) karyotype (KT) image acquisition. The first stage normally produces dozens of MC images for a patient, and

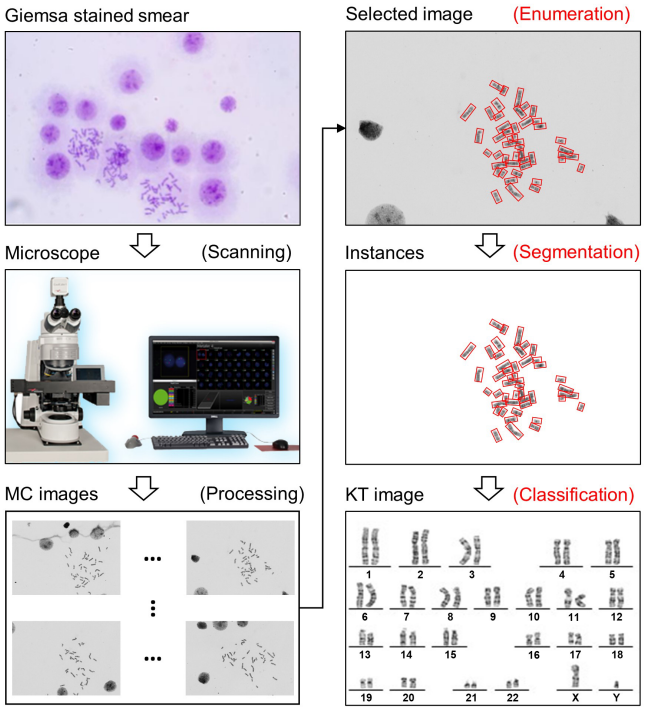


Fig. 1. Illustration of karyotype analysis, which generally comprises two main stages: (a) metaphase cell (MC) image acquisition and (b) karyotype (KT) image acquisition. Chromosome enumeration, segmentation, and classification are the main steps for KT image acquisition, all of which should rely on chromosome detection.

* This work was supported in part by National Natural Science Foundation of China under Grant 81974276 and 62101318, and Key Research and Development Program of Jiangsu Province under Grant BE2020762. *(Corresponding author: X. Jiang, C. Xu, and D. Qian).* J. Wang and C. Zhou are contributed equally in this work as co-first authors.

J. Wang and M. Wu are with the School of Computer and Computational Science, Hangzhou City University, Hangzhou, China (e-mail: wjcy19870122@163.com, mhwu@zucc.edu.cn).

C. Zhou and D. Qian are with the School of Biomedical Engineering, Shanghai Jiao Tong University, Shanghai, China (e-mail: joe1chief1993@gmail.com; dahong.qian@sjtu.edu.cn).

J. Hu is with the College of Computer Science and Technology, Zhejiang University, Hangzhou, China (e-mail: jianwuhu@icloud.com).

S. Chen and C. Xu are with the Obstetrics and Gynecology Hospital, Institute of Reproduction and Development, Fudan University, Shanghai, China (e-mail: 26775172@qq.com; xuchenm@163.com).

X. Jiang is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore (e-mail: exdjiang@ntu.edu.sg).

Mentions of supplemental materials and animal/human rights statements can be included here.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org

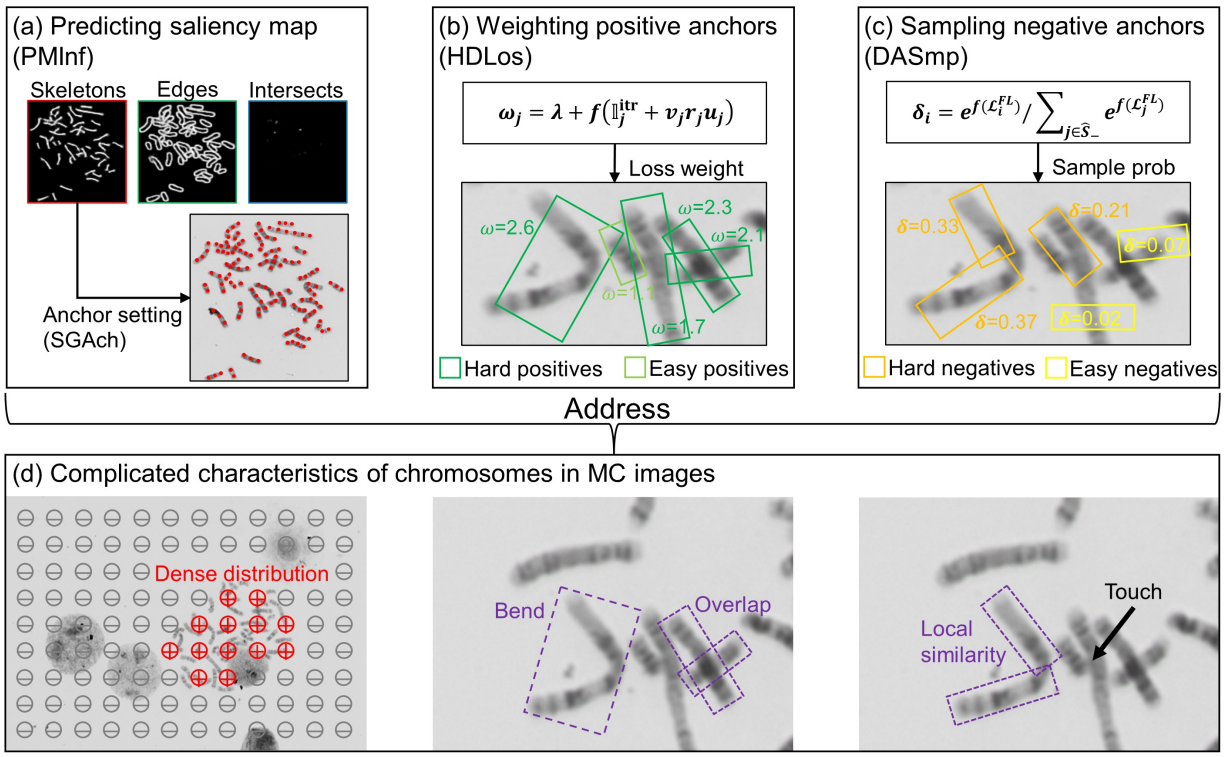


Fig. 2. The proposed three strategies to enhance the detection performance. (a) The saliency map with three channels of skeletons, edges, and intersections is predicted for two purposes: 1) as the Prior Morphological Information (PMInf) for enhancing the feature representations of chromosomes and 2) reducing redundant anchors by setting anchors on the skeleton regions instead of on the whole image space (i.e., the SGAch). (b) The Hardness-aware Loss (HDLos) adjusts the loss contribution of positive anchors according to their hardness scores, which reinforces the model's ability to detect hard chromosomes. (c) The Dynamic Adaptive Sampling (DASmp) strategy selects hard negative anchors using sampling probabilities that are calculated from training losses, aiming to further address the local-similarity issue that normally appears on the bend chromosomes. (d) demonstrates that chromosome detection is a challenging task due to the complicated morphological characteristics of chromosomes. In particular, the dense distribution will lead to severe anchor imbalance issues between the foreground and background. Note that, for clarity, we only show the partial region of the image in the above cases.

cytogeneticists must count the number of chromosomes over at least 20 images for the diagnosis of numerical abnormality (i.e., chromosome enumeration)[3]. In addition, cytogeneticists need to select several high-quality MC images for subsequent processing, including chromosome segmentation and classification for the layout of KT images, and finally diagnose structural abnormalities. Considering that every normal person has 23 pairs of chromosomes, manual karyotype analysis is extremely labor intensive and time-consuming even for experienced cytogeneticists.

Consequently, there is an urgent need to develop reliable automated frameworks for fast chromosome examination. Many prior studies have devoted significant efforts to this area, mostly with the focus on chromosome enumeration[3], segmentation[4]–[10], and classification[11]–[13]. Notably, fast and accurate chromosome detection in MC images is a vital prerequisite for effectively addressing the abovementioned problems. In recent years, many deep convolutional neural network (DCNN[14])-based detectors have been developed for object detection in natural images[15]–[17]. Inspired by these works, some chromosome detectors have been developed for specific tasks. For example, Xiao et al.[3] designed a horizontal-anchor-based detector named DeepACEV2 based on the FasterRCNN[15] for chromosome enumeration. Wang et al.[8] modified the MaskRCNN[16] for chromosome instance segmentation.

However, existing detectors are still inappropriate for chromosome detection, mainly due to two reasons: 1) Unlike objects in other images, chromosomes in MC images have complicated morphological features such as dense distributions, arbitrary orientations, touching, overlapping, local similarities, wide variations in lengths, and diverse G-band patterns (see Fig. 2(d)). As a result, it is challenging to achieve satisfactory results even when using SOTA detectors (see Section II for details). 2) Apart from the accuracy, the detection speed is also important for automated karyotyping. As above mentioned, the karyotyping of each patient is always performed on dozens of MC images to avoid misdiagnosis. However, top-performing detectors still rely on the two-stage FasterRCNN structure, which locates objects by classifying and regressing millions of anchors that are densely set in the entire image space. This pipeline involves several complex steps such as RoI-Proposal and RoI-Alignment. Unfortunately, this approach suffers from a relatively slow detection speed and an anchor-imbalance issue, where most anchors are distributed in the background regions rather than the foregrounds. To the best of our knowledge, the research community still lacks a robust framework that can handle the abovementioned chromosome characteristics well and perform fast and accurate chromosome detection.

In this paper, we propose a novel DCNN-based detection

framework, named DeepCHM, for chromosome detection in MC images. The DeepCHM is a one-stage detector with rotated anchors, making it more suitable for automated karyotype analysis than existing approaches. This feature enables faster and more accurate localization of chromosomes with arbitrary orientations. To enhance the model's performance, three strategies are proposed as follows:

Firstly, an end-to-end deep learning approach is employed to learn a three-channel saliency map, which represents the chromosomal skeletons, edges, and intersections (as shown in Fig. 2a). This saliency map serves as Prior Morphological Information (PMInf) to enhance the feature representations of chromosomes. Furthermore, the predicted skeletons are utilized to guide anchor setting through a strategy named Skeleton-Guided Anchor (SGAch), where anchors are placed on the chromosome skeletons instead of the entire image. This strategy effectively eliminates redundant negative anchors from the background regions that do not contribute to useful learning signals, resulting in faster detection and significant improvement in model performance.

Secondly, prior research has shown that chromosomes with a dense distribution, extremely long, bend, and overlapped structures are more difficult to identify than the straight and independent chromosomes[2]. In light of of these observations, we propose a Hardness-aware Loss function (HDLos) based on the focal loss[18] and Kullback-Leibler divergence loss[19]. This loss function weights the contribution of positive anchors according to their hardness scores, which are calculated based on the density, length, and bend degree of chromosomes. The HDLos enhances the detector's ability to handle these challenging hard chromosomes as seen in Fig. 2b.

Thirdly, we propose a Dynamic Adaptive Sampling (DASmp) strategy to select hard negative anchors. This approach utilizes resampling probabilities calculated from the training loss to effectively address the local-similarity issue that commonly occurs on bend chromosomes (as shown in Fig. 2c).

To develop and validate the effectiveness of our method, we established a large-scale dataset named AutoKary2022. Our extensive experimental results demonstrate that DeepCHM outperforms the majority of existing detectors in terms of both chromosome detection performance and efficiency. Overall, our main contributions can be summarized as follows:

1) We develop a framework named DeepCHM for single-stage chromosome detection in MC images, which utilizes rotated bounding boxes to accurately locate chromosomes;
2) We propose a SGAch strategy to significantly boost the model's performance and accelerate detection by guiding anchor placement on the chromosome skeletons instead of the entire image;
3) We design a penalized loss function (i.e., the HDLos) for handling hard samples and a negative anchor sampling strategy (i.e., the DASmp) for addressing the anchor-imbalance issue, which further improve the detection performance.
4) To the best of our knowledge, we have built the first large-scale dataset (namely, the AutoKary2022) for chromosome detection and segmentation. It contains 624 metaphase cell images of resolution 2200×3200, with a total of 27,763 chromosome instances were densely annotated by experienced cytogeneticists.

II. Related Work

In this section, we first present an overview of general DCNN-based object detectors and analyze their pros and cons. Then, we review related studies on automated chromosome detection in MC images.

A. Two-stage Detectors

Most existing detectors such as Faster-RCNN[15] and FPN[17] contain two stages: a Region Proposal Network (RPN) for candidates detection followed by a Region Classification Network (RCN) for simultaneous false positive reduction and category classification. These detectors can achieve SOTA performance in many tasks by adopting the RCN to refine the results of RPN. However, their computational cost is relatively high because proposal generation and subsequent feature resampling are required in the RPN and the RCN stages, respectively which makes them unsuitable for time-sensitive tasks.

B. One-stage Detectors

One-stage detectors[18], [20], [21] were developed for speeding the detection. For example, Redmon and Farhadi[21] developed the YOLOv3 framework for ultrafast object detection, which is approximately 4 times faster than the FPN. However, without the refinement stage, the performance of the one-stage detectors is generally inferior to that of the multistage detectors due to the anchor imbalance[18]. The root cause of anchor imbalance is that anchors are densely set on the image. Since most image regions are background, the number of negative anchors may be as large as 10, 000 times that of positive anchors. This degenerates the model and wastes computational resources (e.g., in terms of intersection-over-union (IoU)), as the easy negatives dominate the model training. To address this, the Online Hard Example Mining (OHEM)[22] samples top-k hard negatives to maintain a balance between the positives and negatives (e.g., $1:3$). A more efficient alternative is the focal loss strategy used in[18], which calculates a hard-example-biased cross entropy for model training thereby avoiding the anchor sampling operation. Although the effectiveness of focal loss has been proven via a one-stage detector named RetinaNet, it might not be efficient enough for chromosome detection in MC images.

C. Rotated Detectors

In practice, objects may have arbitrary orientations and dense distributions, such as aerial images[23][24], text images[25], and especially MC images[3]. In these applications, a single horizontal bounding box often contains several intact or

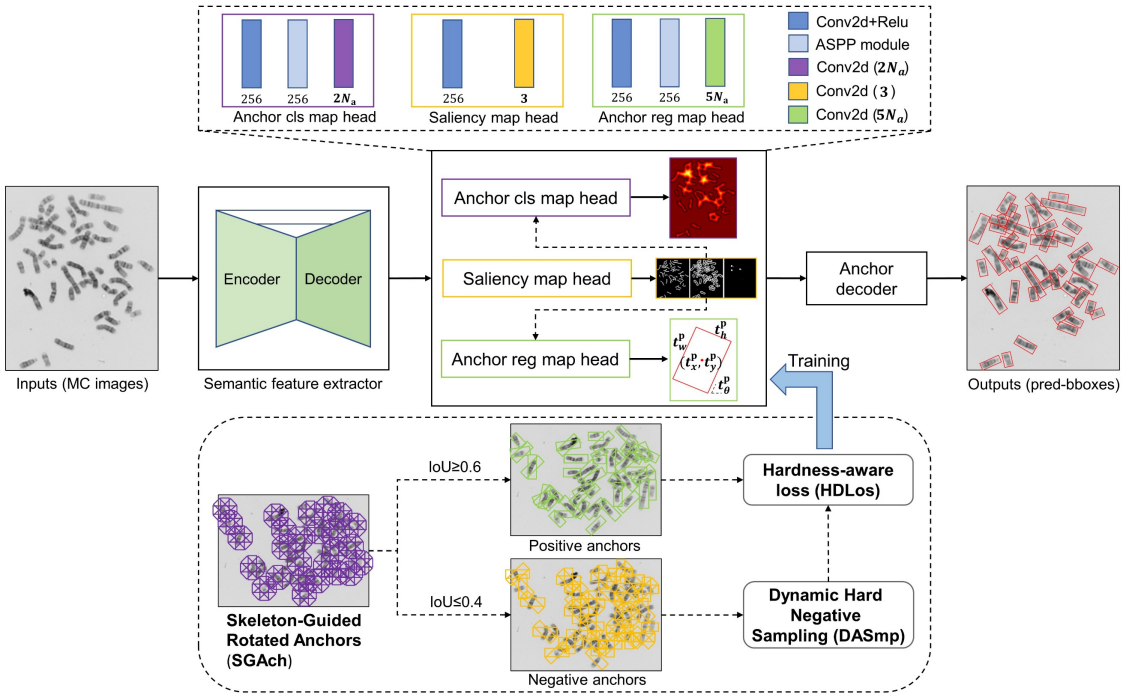


Fig. 3. Overview of the proposed DeepCHM. A backbone network with an encoder and decoder structure serves as a feature extractor for the resulting three head subnetworks. The saliency map enhances feature representations (i.e., the PMInf) and reduces redundant anchors (i.e., the SGAch) for the anchor classification and regression tasks. In addition, the hardness-aware loss (i.e., the HDLos) along with the anchor sampling strategy (i.e., DASmp) optimize the model's learnable parameters, making the model focus on hard examples. For clarity, we only show the partial region of the image in the above case.

fragmentary instances, which may impede the subsequent procedures (e.g., segmentation and classification). Therefore, researchers have resorted to detectors using rotated anchors for more accurate detection. For example, Ming et al.[24] added an angle parameter to each bounding box to locate objects. Recently, Yang et al.[19] proposed the Kullback-Leibler divergence loss for the regression of rotated anchors to address the flaw of L1-smooth loss[26]. However, compared to the horizontal anchor strategy, the rotated anchor strategy has much higher computational complexity and more severe anchor imbalance issues. To address these issues, most SOTA rotated detectors are still based on horizontal anchors[27]. They apply spatial transformations on horizontal anchors to obtain rotated boxes. Although these detectors are effective in specific applications, their performance may deteriorate in the case of densely distributed objects that are difficult to horizontal anchors.

Inspired by the abovementioned works, we designed the DeepCHM that inherits their advantages, including one-stage detection and rotated anchors. At the same time, as mentioned above, the proposed DeepCHM has distinctive advantages in overcoming the anchor imbalance issue (i.e., SGAch along with DASmp) and the specific challenges in chromosome detection (i.e., PMInf and HDLos).

D. Automatic Chromosome Detection

Many methods have been developed for chromosome classification[11]–[13][28] and segmentation[2][4]–[8][29]. However, very few studies have tried to establish a robust method for fast and accurate chromosome detection. Recently, Xiao et al.[3] resorted to FasterRCNN for chromosome enumeration. Although their method can effectively count the number of chromosomes, it cannot handle chromosome detection well because horizontal bounding boxes are used. In addition, to address the anchor imbalance issue, they proposed an IoU-based Hard Negative Anchor Sampling (HNAS) strategy to build minibatches for model training. This sampling strategy is efficient and effective. However, it cannot guarantee the selection of real hard negative anchors since it is not model-driven. In contrast, the rotated MaskRCNN developed by Wang et al.[8] is a better solution, which locates chromosomes more accurately via rotated bounding boxes. However, similar to most existing rotated detectors, the rotated MaskRCNN is also based on horizontal anchors, which may confuse the detector in distinguishing the touching and overlapping chromosomes.

III. METHODS

The goal is to train the proposed DeepCHM model via classification and regression of a set of rotated anchors that can be denoted by $\boldsymbol{S}_{\mathrm{a}} = \{A_m | m = 1, \ldots, M\}$, with the supervision of a set of ground-truth bounding boxes (gt-bboxes) $\boldsymbol{S}_{\mathrm{g}} = \{G_n | n = 1, \ldots, N\}$. Once the model is trained, it can locate the chromosomes in an MC image with a set of predicted bounding boxes (pred-bboxes) $\boldsymbol{S}_{\mathrm{p}} = \{B_k | k = 1, \ldots, K\}$. Fig. 3 shows the overall framework of DeepCHM, which is a single, unified network consisting of 1) a DCNN-based backbone network with an encoder-decoder structure for extracting semantic features, 2) three head subnetworks for saliency map prediction, anchor classification, and regression, and 3) an anchor decoder for finally determining the pred-bboxes. The saliency map head

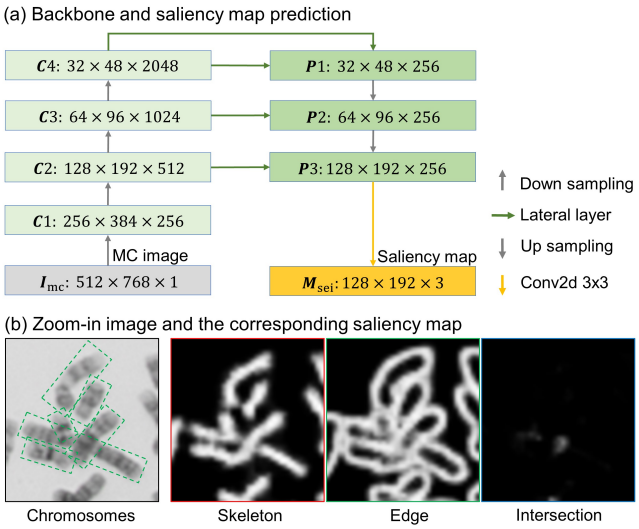


Fig. 4. (a) The backbone network for the saliency map prediction. (b) Zoom-in image of a cluster of chromosomes, and the corresponding saliency map demonstrates that the skeleton and the edge predictions can isolate densely distributed instances, while the intersection prediction can enlighten the model to handle the intersection regions.

predicts skeletons, edges, and intersections of chromosomes, which serves as the prior morphological information (i.e., the PMInf) to enrich the semantic features for more accurate anchor classification and regression. More importantly, the predicted skeletons guide the anchor setting (i.e., the SGAch) to significantly reduce the number of negative anchors. Furthermore, the hardness-aware loss (i.e., the HDLos) along with the anchor sampling strategy (i.e., DASmp) optimize the model, making it focus on hard examples. Details are presented in the following subsections.

A. Backbone and Saliency Map Prediction

Fig. 4 (a) illustrates the backbone network for the saliency map prediction, which is an encoder-decoder structure following the design of FPN[17]. In this study, ResNet-50[30] is adopted as the encoder to extract multiscale CNN feature maps, i.e., $(\boldsymbol{C}_1, \boldsymbol{C}_2, \boldsymbol{C}_3, \boldsymbol{C}_4)$, from the MC image of size 512×768. Then, a parallel "top-down" pathway merges the feature maps of the same spatial size from the encoder and the decoder via lateral connections to obtain an enhanced feature hierarchy $(\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3)$ with a scaling step of 2, which can represent chromosomes at serval scales. Since the deepest feature map $\boldsymbol{P}_3$ of size 128×192 covers the chromosomes well, we only adopt it for the subsequent process for simplicity.

Then, the saliency map, namely the skeletons, edges, and intersections, is obtained from the feature map $\boldsymbol{P}_3$ by a 3×3 convolutional layer with three channels and a sigmoid activation. The convolutional layer parameters are optimized with the focal loss to minimize the regression errors between the predicted map and the ground truth. Let the salient map and the ground truth be $\boldsymbol{M}_{\text{sei}} \in R^{H \times W \times 3}$ and $\boldsymbol{M}^*_{\text{sei}} \in R^{H \times W \times 3}$, respectively, the loss function is defined as follows:

$$\mathcal{L}_{\text{sei}} = -\sum\big[\boldsymbol{M}^*_{\text{sei}}(1-\boldsymbol{M}_{\text{sei}})^{\alpha}\log(\boldsymbol{M}_{\text{sei}}) + (1-\boldsymbol{M}^*_{\text{sei}})^{\beta}\boldsymbol{M}_{\text{sei}}{}^{\alpha}\log(1-\boldsymbol{M}_{\text{sei}})\big], \quad (1)$$

where $H = 128$ and $W = 192$. The subscript "sei" indicates the three channels corresponding to the skeletons, edges, and intersections. α and β are two hyperparameters that have the default values of 2.0 and 4.0 as suggested by Law et al.[31]. The ground truth skeletons and edges with one-pixel width are obtained according to the annotated masks of chromosomes, while the intersections are the cross points of skeletons (see Section IV.A for details). Notably, to make the model more robust to inaccurate annotations, we regress the Gaussian distributions instead of the one-pixel wide points, which can be formulated as follows:

$$\boldsymbol{M}^*_{c\in(\text{s,e,i})}(x,y) = \max\left[\exp\left(-\frac{(x-x_c)^2(y-y_c)^2}{2\sigma_c^2}\right)\middle|(x_c,y_c) \in \boldsymbol{S}_c\right], \quad (2)$$

where $\boldsymbol{S}_c$ is the points forming the ground truth skeletons, edges, or intersections. The variance σ_c^2 is empirically set to 1.0, 1.0, and 2.0 for the skeletons, edges, and intersections, respectively.

The prediction of the saliency map aids chromosome detection in three aspects: 1) the saliency map provides the prior morphological information for the anchor classification and regression (see Section III.B for details). Fig. 4(b) shows the saliency map predictions of a representative example, which demonstrates that the skeleton and the edge predictions can isolate chromosomes well, even for the touching cluster, while the intersection predictions can enlighten the model to handle the intersection regions; 2) To reduce the redundant anchors, skeleton prediction is used to guide the setting of rotated anchors (see Section III.B for details); 3) Similar to a multitask framework, it can be recognized as a deep supervision strategy to refine the deepest feature map $\boldsymbol{P}_3$.

B. Skeleton-Guided Anchor Setting (SGAch)

Then, the saliency map $\boldsymbol{M}_{\text{sei}}$, which serves as the prior morphological features, is concatenated with the semantic feature map $\boldsymbol{P}_3$ and fed to the anchor classification and regression head subnetworks for predicting the pred-bboxes. In both subnetworks, a 1×1 convolutional layer is applied to fuse the maps, followed by the Atrous Spatial Pyramid Pooling (ASPP) module[32] to encode multiscale local information. The encoded results are utilized to predict a classification map $\boldsymbol{M}_{\text{cls}}$ of size $H \times W \times N_{\text{a}} \times 2$ and a regression map $\boldsymbol{M}_{\text{reg}}$ of size $H \times W \times 5N_a$, where N_{a} is the number of base anchors at each spatial location. In this study, we empirically set a total of 48 rotated anchors (i.e., $N_{\text{a}} = 48$) at each location with the following hyperparameters: base size $l = 5.0$, size scales $\boldsymbol{s} = [2.0, 5.0]$, aspect ratios $\boldsymbol{r} = [2.0, 4.0]$, and an angle interval of 15° in the range of $[0^{\circ}, 180^{\circ})$. $\boldsymbol{M}_{\text{cls}}$ predicts a softmax probability for each anchor, determining whether the anchor is positive or not, while $\boldsymbol{M}_{\text{reg}}$ predicts five offset items that are used to move, scale, and rotate positive anchors for more accurate localization of objects.

As mentioned above, existing detectors densely set anchors on the whole image space, which will produce a great number of anchors and suffer from high computational costs and severe

Algorithm 1 Skeleton-Guided Anchor Generation

Inputs: the skeleton map $\boldsymbol{M}_{\mathrm{s}}$ and threshold T_{loc}; the feature stride Δ; the anchor setting: base size l; scales $\boldsymbol{s}$; aspect ratios $\boldsymbol{r}$; and rotated angles $\boldsymbol{\theta}$;
Output: the set of rotated anchors $\boldsymbol{S}_{\mathrm{a}}$.

Initialize $\boldsymbol{S}_{\mathrm{a}} = [\,]$;
Get anchor locations: $\boldsymbol{S}_{\mathrm{loc}} = [(x,y)|\boldsymbol{M}_{\mathrm{s}}(x,y) \geq T_{\mathrm{loc}}]$
For each anchor location (x,y) in $\boldsymbol{S}_{\mathrm{loc}}$ **do**:
  For each size scale s in $\boldsymbol{s}$ **do**:
    For each aspect ratio r in $\boldsymbol{r}$ **do**:
      $w = l \times s \times \mathrm{sqrt}(r)$; $h = l \times s \times \mathrm{sqrt}(1/r)$
      For each rotated angle θ in $\boldsymbol{\theta}$ **do**:
        $\boldsymbol{S}_{\mathrm{a}}.\mathrm{append}([x \times \Delta,\ y \times \Delta,\ w,\ h,\ \theta])$
      End
    End
  End
End

anchor imbalance issues. For example, if we conventionally set anchors on the whole spatial space of 128×192, the number of rotated anchors per image is up to 1,179,648 (i.e., 128×192 anchor locations multiply to 48 base anchors per location), most of which are redundant to the chromosomes. To this end, we propose the skeleton-guided anchor setting strategy (i.e., SGAch) to reduce the redundant anchors based on the predicted skeletons. Given the skeleton map $\boldsymbol{M}_{\mathrm{s}}$ (the ground truth map in the training stage or the prediction map in the inference stage), we first obtain the anchor locations by a thresholding operation as follows:

$$\boldsymbol{S}_{\mathrm{loc}} = [(x,y)|\boldsymbol{M}_{\mathrm{s}}(x,y) \geq T_{\mathrm{loc}}], \tag{3}$$

where (x,y) is the pixel coordination, and T_{loc} in the range of $[0,1)$ is a user-defined threshold that empirically set to 0.2 in this study. Then, we set the 48 base anchors at each location in $\boldsymbol{S}_{\mathrm{loc}}$. Empirically, when SGAch is applied, the number of anchors can be reduced by more than 40 times. The pseudocode of skeleton-guided anchor generation is summarized in Algorithm 1.

C. Hardness-Aware Loss (HDLos)

During the training stage, anchors are assigned to the following two sets: the positive anchor set $\boldsymbol{S}_{+}$ and the negative anchor set $\boldsymbol{S}_{-}$ (see Section III.D for details). Then, a classification loss is applied to optimize the anchor classification head subnetwork as follows:

$$\mathcal{L}_{\mathrm{cls}} = \mathcal{L}_{-}^{\mathrm{FL}} + \mathcal{L}_{+}^{\mathrm{PFL}}, \tag{4}$$

where FL is the abbreviation of focal loss, and $\mathcal{L}_{-}^{\mathrm{FL}}$ denotes the focal loss for the negative set:

$$\mathcal{L}_{-}^{\mathrm{FL}} = \frac{1}{|\boldsymbol{S}_{-}|}\sum_{i\in\boldsymbol{S}_{-}} \mathrm{FL}(\boldsymbol{M}_{\mathrm{cls}}^{i}, \boldsymbol{Y}_{i}), \tag{5}$$

where $|\boldsymbol{S}_{-}|$ is the number of anchors in the negative set. $\boldsymbol{M}_{\mathrm{cls}}^{i}$ is the prediction score of the i_{th} negative anchor, and $\boldsymbol{Y}_{i} = 0$ is the ground truth for negative anchors. The second term $\mathcal{L}_{+}^{\mathrm{PFL}}$ in Eq. (4) is a penalized focal loss (PLF) for the positive set, which

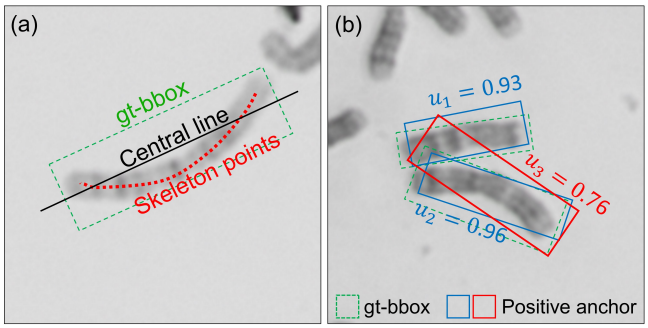


Fig. 5. (a) An example for calculating the bend score of the chromosome. The bend score is defined as the average distance of all skeleton points to the central line. (b) An example demonstrates that a positive anchor with a smaller value of u may confuse the model because it overlaps with multiple densely distributed instances.

is defined as follows:

$$\mathcal{L}_{+}^{\mathrm{PFL}} = \frac{1}{|\boldsymbol{S}_{+}|}\sum_{j\in\boldsymbol{S}_{+}} \omega_{j}\mathrm{FL}(\boldsymbol{M}_{\mathrm{cls}}^{j}, \boldsymbol{Y}_{j}), \tag{6}$$

where $\boldsymbol{M}_{\mathrm{cls}}^{j}$ is the prediction score of the j_{th} positive anchor, and $\boldsymbol{Y}_{j} = 1$ is the ground truth for the anchor. ω_{j} is a weight term adjusting the bias on the hard-positive anchors that belong to cross-overlapped, bend, long, and densely distributed chromosomes. ω_{j} is calculated by:

$$\omega_{j} = \lambda + f(\mathbb{I}_{j}^{\mathrm{itr}} + v_{j}r_{j}u_{j}), \tag{7}$$

where $\lambda \geq 0$ is a constant that is set to 1.0 in this study. f is a monotonically increasing function, and the following exponential function is adopted:

$$f(x) = 1 - \frac{1}{\mathrm{e}^{\rho x}}. \tag{8}$$

where $\rho > 0$ is a scale parameter which is set to 1.0 by default. The term $\mathbb{I}_{j}^{\mathrm{itr}}$ in Eq. (7) is an indicator function for whether the j_{th} anchor belongs to a cross-overlapped chromosome ($\mathbb{I}_{j}^{\mathrm{itr}} = 1$) or not ($\mathbb{I}_{j}^{\mathrm{itr}} = 0$). v_{j}, r_{j}, and u_{j} measure the bend degree, length, and distribution density of the chromosome that is represented by the j_{th} anchor. They are calculated as follows:

$$v = \frac{1}{|\boldsymbol{S}_{\mathrm{skel}}|}\sum_{(x_i,y_i)\in\boldsymbol{S}_{\mathrm{skel}}} \frac{|kx_i - y_i + b|}{\sqrt{k^2+1}}, \tag{9}$$

where $\boldsymbol{S}_{\mathrm{skel}}$ is the skeleton points of the chromosome, and $y = kx + b$ is the central line of the gt-bbox (see Fig. 5(a) for an example). The value of v is the average distance of all skeleton points to the central line, which is larger for bend chromosomes. r_{j} is the aspect ratio of the chromosome's gt-bbox:

$$r = \frac{\max(w,h)}{\min(w,h)}, \tag{10}$$

where w and h are the width and height of the gt-bbox, respectively. A longer chromosome normally has a larger value of r. u_{j} is calculated as follows:

$$u_{j} = \frac{\max[\mathrm{IoU}(A_{j}, G_{n})|n=1,2,\ldots,N]}{\sum_{n=1}^{N}\mathrm{IoU}(A_{j}, G_{n})}, \tag{11}$$

where $\mathrm{IoU}(A_{j}, G_{n})$ indicates the skew-IoU[25] between the j_{th} anchor and the n_{th} gt-bbox. As demonstrated in Fig. 5(b), the

positive anchor with a smaller value of u may confuse the model because it overlaps with multiple densely distributed instances. To alleviate this issue, an anchor with a smaller value of u should have a smaller weight in the loss calculation.

Accordingly, the following penalized Kullback-Leibler divergence loss is proposed to optimize the regression subnetwork:

$$\mathcal{L}_{\text{reg}} = \frac{1}{|\boldsymbol{S}_+|}\sum_{j\in\boldsymbol{S}_+}\omega_j\left[1.0 - \frac{1.0}{\tau+\ln(D_j+1.0)}\right], \tag{12}$$

where $\tau \geq 1$ is a hyperparameter used to modulate the entire loss (τ is empirically set to 1.0 in this study). D_j is the Kullback-Leibler divergence between the Gaussian distribution of the j_{th} pred-bbox and the Gaussian distribution of its target gt-bbox. More details of the Kullback-Leibler divergence loss can be found at [19].

Finally, the model training can be formally expressed as minimizing the following multitask loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{sei}} + \lambda_1\mathcal{L}_{\text{cls}} + \lambda_2\mathcal{L}_{\text{reg}}, \tag{13}$$

where λ_1 and λ_2 are two hyperparameters that control the contribution of the anchor classification term and the anchor regression loss term, respectively. Both hyperparameters are empirically set to 2.0 in this study.

D. Dynamic Hard Negative Sampling (DASmp)

In the training stage, the anchor set $\boldsymbol{S}_{\text{a}}$ is split into two subsets, i.e., the positive set $\boldsymbol{S}_+$ and the negative set $\widehat{\boldsymbol{S}}_-$, according to the following criterion: 1) the anchors with a skew-IoU$\geq$ 0.6 to any gt-bboxes and the anchors with the maximum skew-IoU to each gt-bbox are assigned to the positive set; 2) the anchors with skew-IoU$\leq$ 0.4 to all gt-bboxes are assigned to the negative set; 3) similar to other anchor-based detectors, the remaining anchors are directly ignored.

Although the redundant anchors have been significantly reduced via the SGAch strategy, the anchor-imbalance issue still exist. To address this, we propose the DASmp strategy for sampling negative anchors from the negative set, which can be expressed by $\boldsymbol{S}_- = \Psi(\widehat{\boldsymbol{S}}_-,\ N_{\text{samp}}, \boldsymbol{\delta})$, where Ψ represents the reservoir-based weighted sampling algorithm[33], N_{samp} is the sampling number that is set to enforce a 1:2 ratio between positive and negative anchors, and $\boldsymbol{\delta}$ is a vector as the sampling weights that reflects the hardness of the anchors. $\boldsymbol{\delta}$ is calculated as follows: For each negative anchor A_i in $\widehat{\boldsymbol{S}}_-$, we first calculate their focal loss $\mathcal{L}_i^{\text{FL}} = \text{FL}(\boldsymbol{M}_{\text{cls}}^i, 0)$ as defined in Eq. (5), which can measure the hardness of the anchor, i.e., the larger $\mathcal{L}_i^{\text{FL}}$ is, the harder the anchor. Then, we normalize the loss via the softmax function and obtain the final hardness score:

$$\delta_i = \frac{e^{f(\mathcal{L}_i^{\text{FL}})}}{\sum_{j\in\widehat{\boldsymbol{S}}_-} e^{f(\mathcal{L}_j^{\text{FL}})}}, \tag{14}$$

where f is an exponential function as defined in Eq. (8) with the scale parameter $\rho = 4.0$ to smooth the hardness scores. Since the hardness scores are calculated based on the classification loss, the anchor sampling is biased on the model-aware hard anchors that are dynamically changed with the training iterations.

E. Anchor Decoder

Finally, the pred-bboxes $\boldsymbol{S}_{\text{p}}$ are obtained from the skeleton-guided rotated anchors $\boldsymbol{S}_{\text{a}}$ based on the anchor classification map $\boldsymbol{M}_{\text{cls}}$ and the regression map $\boldsymbol{M}_{\text{reg}}$. The process consists of the following three steps:

1) *Highly Confident Anchors*: The predicted softmax scores and the offset items corresponding to the anchor locations (i.e., $\boldsymbol{S}_{\text{loc}}$) are first extracted and reshaped to $M \times 1$, and $M \times 5$, respectively, where M is the total number of anchors in the anchor set $\boldsymbol{S}_{\text{a}}$. Then, highly confident anchors are obtained by thresholding with a user-defined threshold of T_{score}.

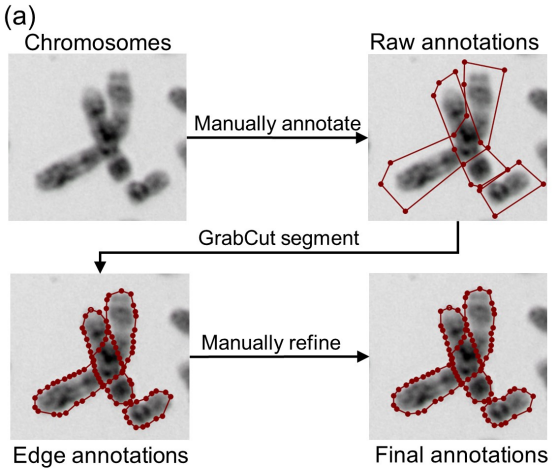
2) *Box Decoding*: The anchors obtained in the above step are converted to the pred-bboxes based on their offset items. Each pred-bbox comprises six parameters, i.e., the center point $(x_{\text{p}}, y_{\text{p}})$, the width w_{p}, the height h_{p}, the angle θ_{p}, and the softmax score.

3) *Non-Maximum Suppression*: The pred-bboxes set $\boldsymbol{S}_{\text{p}}$ may contain redundant boxes that are highly overlapped. To reduce the redundant boxes, we apply the skew-NMS[25] on $\boldsymbol{S}_{\text{p}}$ with a threshold T_{nms} (0.2 in this study), and the remaining boxes serves as the final predictions.

IV. Experiments

A. Dataset

We built a large-scale dataset named AutoKary2022. We collected 624 Giemsa-stained microscopic metaphase cell images of 50 volunteers (containing a total of 27, 763 chromosome instances) from the Obstetrics & Gynecology Hospital of Fudan University. All images with a resolution of 2200 × 3200 were obtained using the AMS5 Scanner with a 60x NA1.4 objective lens. The images were annotated by two experienced cytologists using the LabelMe annotation tool[34]. The region of each chromosome (i.e., the ground-truth mask) was outlined with a polygon. Then, we obtained the rotated gt-bbox, skeleton, edge, and intersection points of each instance based on their annotations. Notably, to alleviate the workload of cytologists, we proposed a semi-automatic method for chromosome annotation. As demonstrated in Fig. 6a, cytologists only need to annotate a raw polygon for each instance. Then, a GrabCut method[33] was used to segment the instances automatically based on their raw polygons, which aimed to obtain more precise edges of the instances. Finally, the edge polygons were manually refined to obtain the final annotations. In this study, we randomly selected 40 patients with 518 images for training and the remaining 10 patients with 106 images for testing (see Fig. 6b).



(b)

Subsets	Patients	Images	Chromosomes
Train	40	518	23,070
Test	10	106	4,693

Fig. 6. (a) The procedure for data annotation. The region of each chromosome was firstly outlined with a polygon. Then, a GrabCut method was applied to segment instances based on their polygons, which aimed to obtain more precise edges of the instances. (b) Number of samples in the training and test datasets.

B. Experimental Design

We conducted a comparative study between our detector and six SOTA rotated detectors, namely R3Det[35], Rotated RetinaNet (rRetinaNet)[18], Rotated FasterRCNN (rFasterRCNN)[15], Oriented RCNN (rRCNN)[36], RoI Transformer (RoI-Trans)[23], and Oriented RepPoints (rRepP)[37]. These methods all predict rotated pred-bboxes from horizontal anchors instead of rotated anchors. To demonstrate the superiority of the rotated detectors in chromosome detection, we also compared our method to seven horizontal detectors: YOLOv3[21], RetinaNet[18], FasterRCNN[15], TOOD[38], DINO[39], DDOD[40], and VFNet[41].

Given that our dataset contains ground-truth masks and the proposed detector incorporates segmentation information (i.e., the saliency maps) into training, we trained two instance segmentation methods, MaskRCNN[16] and SCNet[42], for fair comparison. The above rotated and horizontal detectors were implemented in the MMRotate[27] and the MMDetection[43] frameworks, respectively.

To validate the effectiveness of each proposed strategy, i.e., the saliency map as the prior morphological information (PMInf), the skeleton-guided anchors (SGAch), the hardness-aware loss (HDLoss), and the anchor sampling strategy (DASmp), we conducted an ablation study. We first trained a baseline detector without any proposed strategy, then additional detectors were trained by adding each strategy at a time to the baseline detector. Furthermore, experiments were also performed to verify the impact of some hyperparameters that might be vital for the model performance.

C. Implementation Details

The proposed models were trained using Google TensorFlow (version 2.8 with Keras API) on NVIDIA RTX 3090Ti GPUs. During the training stage, the multitask loss $\mathcal{L}_{\text{total}}$ was minimized using the Adam optimizer with a learning rate of 0.0001, decaying every 518 iterations using an exponential rate of 0.96. The total number of iterations was 62,160 (120 epochs times 518 iterations), and the mini-batch size was 1. During the training, we resized each image and its ground-truth masks to size of 512×768 using bilinear interpolation, and conducted random flipping and rotation of images as data augmentation to enlarge the training set.

For the ablation study, the baseline detector was trained using the above configurations but without implementing the proposed strategies. The main differences between the baseline and the DeepCHM are as follows: 1) The saliency maps were not used in the heads of the detector; 2) The rotated anchors were set within the whole image space instead of being adjusted based on the skeletons; 3) The penalized term (i.e., the ω_j in Eq. 6 and Eq. 12) was not included in the loss functions; 4) Negative anchors were selected randomly rather than being adaptively sampled.

When training models of existing methods, we utilized their default configurations within the MMRotate and the MMDetection frameworks. To ensure fairness, all detectors were trained with identical input size of 512×768, a ResNet-50-based backbone network, and the same batch size.

D. Evaluation Metrics

To comprehensively validate the detection performance, we adopted the following five metrics: *Precision*, *Recall*, *F1-score*, Mean False Positives (*mFPs*) per image, and Mean Overlap Degree (*mOD*). The calculation of the abovementioned metrics can be formally expressed as follows:

$$Precision = \frac{TP}{TP+FP}, \tag{15}$$

$$Recall = \frac{TP}{TP+FN}, \tag{16}$$

$$F1-score = \frac{2\times Precision\times Recall}{Precision+Recall}, \tag{17}$$

$$mFPs = \frac{FP}{N_{\text{im}}}, \tag{18}$$

$$mOD = \frac{1}{TP}\sum_{n=1}^{TP} u_n, \tag{19}$$

where $N_{\text{im}} = 106$ is the number of testing images. TP, FP, and FN are the number of True Positives, False Positives, and False Negatives, which are determined as follows: 1) True Positives (TP): the pred-bboxes with skew-IoU$\geq$ 0.5 to any gt-bboxes; 2) False Positives (FP): the pred-bboxes with skew-IoU$<$ 0.5 to all gt-bboxes; 3) False Negatives (FN): the gt-bboxes without skew-IoU$\geq$ 0.5 to any pred-bboxes. u_n measures the overlap degree (see Eq. 11) between each true positive pred-bbox and

TABLE I
COMPARISON WITH THE STATE-OF-THE-ART DETECTORS

Detectors	*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)	Speed (fps)
One-stage detectors							
YOLOv3#	63.84 ± 0.43	83.95 ± 0.71	71.84 ± 0.71	77.18 ± 0.33	6.01 ± 0.35	91.02 ± 0.31	28.76
RetinaNet#	73.12 ± 0.65	83.11 ± 0.88	88.11 ± 2.50	85.34 ± 1.14	9.04 ± 0.65	86.44 ± 0.28	21.77
TOOD#	82.70 ± 0.54	87.17 ± 0.73	85.93 ± 0.16	86.35 ± 0.44	6.02 ± 0.42	90.26 ± 0.96	18.62
DINO#	88.36 ± 0.64	92.43 ± 0.50	93.51 ± 0.33	92.93 ± 0.24	3.76 ± 0.38	90.67 ± 0.11	10.60
DDOD#	89.41 ± 0.11	92.82 ± 0.73	88.25 ± 0.81	90.35 ± 0.16	2.99 ± 0.41	90.82 ± 0.18	19.94
VFNet#	86.59 ± 0.19	92.54 ± 0.12	89.16 ± 0.11	90.74 ± 0.08	3.18 ± 0.06	90.81 ± 0.05	17.80
rRetinaNet*	78.73 ± 0.58	89.23 ± 0.34	82.75 ± 1.20	85.74 ± 0.69	4.30 ± 0.16	96.74 ± 0.07	27.97
R3Det*	85.09 ± 0.98	93.71 ± 0.27	86.47 ± 0.87	89.79 ± 0.48	2.44 ± 0.13	96.77 ± 0.07	23.90
rRepP*	90.51 ± 0.18	**94.79 ± 0.36**	92.97 ± 0.37	93.83 ± 0.24	2.26 ± 0.17	96.18 ± 0.02	**36.98**
DeepCHM (Ours)*	**93.53 ± 0.28**	94.71 ± 0.17	**93.51 ± 0.28**	**94.11 ± 0.08**	2.31 ± 0.08	**97.13 ± 0.03**	32.57
Multistage detectors							
FasterRCNN#	73.22 ± 0.34	91.32 ± 0.23	77.62 ± 0.33	83.70 ± 0.22	3.31 ± 0.66	90.49 ± 0.44	20.37
MaskRCNN#	73.62 ± 0.42	91.53 ± 0.59	77.97 ± 0.58	84.44 ± 0.25	3.20 ± 0.29	90.26 ± 0.12	6.45
SCNet#	78.79 ± 0.38	91.87 ± 0.30	82.34 ± 0.48	86.72 ± 0.21	3.21 ± 0.16	90.45 ± 0.08	3.99
rFasterRCNN*	85.86 ± 0.77	95.43 ± 0.09	89.25 ± 0.69	92.14 ± 0.38	1.83 ± 0.04	96.85 ± 0.03	21.81
rRCNN*	87.28 ± 0.98	96.56 ± 0.19	89.46 ± 0.84	92.76 ± 0.45	**1.35 ± 0.08**	96.99 ± 0.06	18.76
RoI-Trans*	89.31 ± 0.92	94.95 ± 0.16	89.75 ± 0.72	92.18 ± 0.37	2.06 ± 0.08	96.11 ± 0.02	17.39

The highest score in each column is shown in bold. The hashtag '#' and the asterisk '*' indicates horizontal and rotated detectors, respectively. Each value was given as the average ± standard deviation of the results of 5-fold validation.

their surrounding gt-bboxes. A larger value of *mOD* means more precise localization of the chromosomes.

In addition to the above metrics, we also employed the Average Precision (*AP*) score[42], which is commonly used in object detection tasks to evaluate the overall model performance. To ensure the reliability of our results, we performed 5-fold validation, where we trained five independent models for each detector, and reported the average *AP* score across all five folds.

V. RESULTS

A. Comparison with Existing Detectors

Since different values of the T_{score} for box proposal will lead to different metric values, the value of T_{score} should be first determined for each detector. Considering that *F1-score* is a metric balancing *Precision* and *Recall*, we determined the value of T_{score} that maximized *F1-score* for each detector.

Table I presents the results of all detectors across five folds, from which four main conclusions can be drawn. Firstly, it is evident that rotated detectors outperformed the horizontal detectors in terms of performance. Even though some of the SOTA horizontal detectors, such as the DDOD[40], also achieved relatively high *AP* and *F1-scores*, their *mOD* scores were significantly lower than those of the rotated detectors.

Secondly, among the rotated methods, our method achieved the highest performance with *AP*, *Precision*, and *Recall* as high as 93.53 ± 0.28 %, 94.71 ± 0.17 %, and 93.51 ± 0.28%, respectively, at a low 2.31 ± 0.08 *mFPs*. The SOTA rotated method, rRepP[37], was also highly effective with an *AP* score of 90.51 ± 0.18%. While existing rotated detectors are more accurate in locating chromosomes than the horizontal detectors, they rely on predicting rotated pred-bboxes from horizontal anchors, which can hinder their ability to learn discriminative features, especially in cluster regions. To address this, the rRepP method uses a set of sampled points instead of anchors to locate instances.

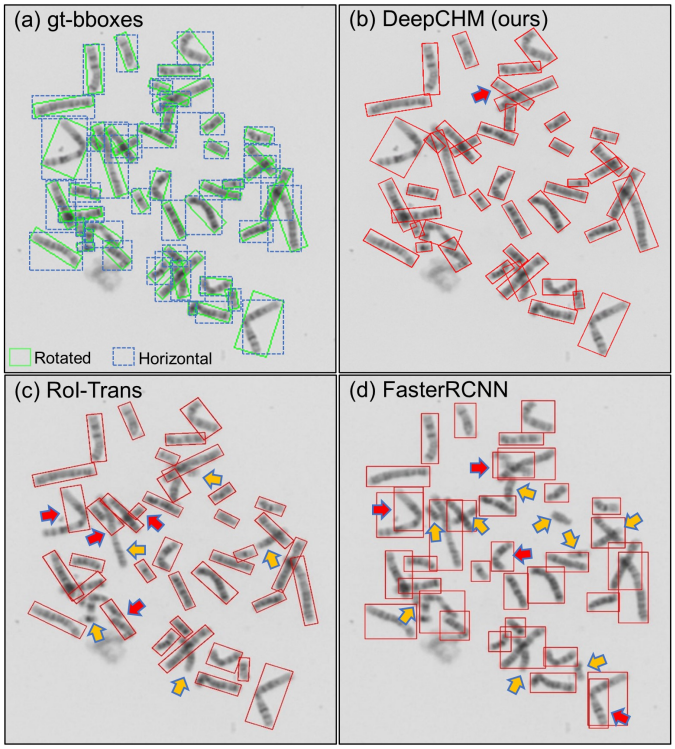


Fig. 7. Detection results on a representative example. (a) shows the gt-bboxes for the rotated and horizontal detections. (b)-(d) show the pred-bboxes of the proposed DeepCHM, RoI-Trans, and FasterRCNN, respectively. Existing detectors missed some chromosome instances, especially the cross-overlapped ones (see the yellow arrows). In addition, existing detectors produced more false positives, especially for the bend chromosomes, as indicated by the red arrows.

Thirdly, the performance of some SOTA single-stage methods even surpassed multistage methods. For instance, the *AP* score of the DDOD method was 89.41 ± 0.11%, which was higher than that of RoI-Trans[23] with an *AP* score of 89.31 ± 0.92 %. However, it has no significant advantage in inference speed. In contrast, our method

TABLE II
THE NUMBER OF IMAGES WITH *RECALL*=100% IN THE TEST DATASET

Detectors	Fold #1	Fold #2	Fold #3	Fold #4	Fold #5
YOLOv3#	0	0	0	0	0
RetinaNet#	1	2	4	0	1
TOOD#	1	4	5	5	2
DINO#	19	15	17	16	17
DDOD#	4	3	3	5	5
VFNet#	5	3	3	4	6
rRetinaNet*	4	3	1	4	3
R3Det*	4	4	3	4	5
rRepP*	**21**	18	**25**	26	17
DeepCHM*	18	**25**	19	**27**	**27**
FasterRCNN#	0	1	1	0	0
MaskRCNN#	0	1	0	0	0
SCNet#	1	2	2	1	2
rFasterRCNN*	11	5	4	5	7
rRCNN*	8	4	2	5	3
RoI-Trans*	6	3	3	2	5

The highest value in each column is shown in bold. The hashtag '#' and the asterisk '*' indicate horizontal and rotated detectors, respectively.

achieves a speed of 32.57 frames per second (fps), which surpasses the DDOD and other multistage methods significantly.

Finally, it was observed that the MaskRCNN[16] and SCNet[42] methods indeed improved the detection performance by incorporating a segmentation task in their models to learn masks. However, the improvement achieved by the MaskRCNN was minor when compared to the FasterRCNN[15], with an *AP* improvement of only about 0.40%. This could be attributed to the fact that these methods predicted masks based on the proposal of horizontal boxes, which did not accurately locate chromosomes.

Fig. 7 visualizes the detection results of a representative case corresponding to the proposed DeepCHM, the RoI-Trans[23], and the FasterRCNN[15]. As indicated by the yellow arrows in Fig. 7(c) and (d), the RoI-Trans and the Faster-RCNN missed some chromosomes, especially for the touching and the overlapped chromosomes, while our method successfully detected them. In addition, RoI-Trans and FasterRCNN produced more false positives, as indicated by the red arrows in Fig. 7(c) and (d). In particular, false positives usually appeared on long bend chromosomes due to the local similarity issue. Considering that both the RoI-Trans and the FasterRCNN were based on horizontal anchors, the reason might be that a horizontal anchor contains multiple instances, which will be much harder for instance separation. However, the rotated detector RoI-Trans still outperformed the horizontal detector FasterRCNN, in particular, with fewer false negatives and more accurate localization of objects.

Notably, whether all instances in an MC image are correctly identified (i.e., *Recall*=100%) is also vital for clinical applications. Consequently, we counted the number of images with *Recall*=100% out of the total 106 test images. The results from Table II further confirmed the superiority of our method to existing detectors. In all five folds, our method achieved more than 18 images with *Recall*=100%, while the other methods (except the SOTA rRepP and DINO methods) had much smaller numbers.

B. Ablation Study

1) Validation of the proposed strategies: The results of the proposed strategies (i.e., the PMInf, SGAch, HDLos, and DASmp) are presented in Table III. It is evident that each strategy brought gains. The baseline detector only achieved an *AP* score of 88.68 ± 0.38 %, while applying each strategy led to higher *AP* scores. For instance, the DASmp improved the model with an *AP* score of 91.88 ± 0.37 %. Among these

TABLE III
EFFECTIVENESS OF THE PROPOSED STRATEGIES ON THE CHROMOSOME DETECTION

Detectors	*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)
baseline	88.68 ± 0.38	91.66 ± 0.39	91.21 ± 0.18	91.43 ± 0.28	3.70 ± 0.18	96.97 ± 0.02
baseline+PMInf	89.42 ± 0.44	92.66 ± 0.37	91.19 ± 0.24	91.92 ± 0.19	3.21 ± 0.18	97.02 ± 0.04
baseline+HDLos	89.60 ± 0.27	91.21 ± 0.22	92.38 ± 0.26	91.80 ± 0.11	3.27 ± 0.12	97.00 ± 0.06
baseline+DASmp	91.88 ± 0.37	91.88 ± 0.37	92.93 ± 0.31	92.40 ± 0.17	3.35 ± 0.19	97.00 ± 0.05
baseline+SGAch	92.57 ± 0.53	94.74 ± 0.34	92.56 ± 0.21	93.63 ± 0.22	2.16 ± 0.24	97.17 ± 0.05
baseline+SGAch+PMInf	92.62 ± 0.51	94.53 ± 0.32	93.01 ± 0.34	93.75 ± 0.19	2.39 ± 0.25	$\mathbf{97.21 \pm 0.03}$
baseline+SGAch+HDLos	92.96 ± 0.64	95.00 ± 0.33	93.07 ± 0.33	94.02 ± 0.26	2.17 ± 0.24	97.16 ± 0.02
baseline+SGAch+DASmp	92.94 ± 0.28	$\mathbf{95.11 \pm 0.31}$	93.01 ± 0.16	94.05 ± 0.14	$\mathbf{2.12 \pm 0.17}$	97.18 ± 0.08
baseline+all (DeepCHM)	$\mathbf{93.53 \pm 0.28}$	94.71 ± 0.17	$\mathbf{93.51 \pm 0.28}$	$\mathbf{94.11 \pm 0.08}$	2.31 ± 0.08	97.13 ± 0.03

The highest score in each column is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation. PMInf: prior morphological information; HDLos: Hardness-aware loss (i.e., the Penalized Focal Loss and the Penalized Kullback-Leibler Divergence Loss); DASmp: dynamically adaptive anchor sampling; SGAch: skeleton-guided rotated anchors.

TABLE IV
IMPACT OF ANCHOR'S ANGLE INTERVAL ON THE MODEL PERFORMANCE

Angle interval	Anchors	*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)
5°	144	92.99 ± 0.11	94.83 ± 0.23	93.01 ± 0.17	93.86 ± 0.07	2.23 ± 0.11	97.03 ± 0.05
10°	72	93.36 ± 0.20	94.63 ± 0.16	93.47 ± 0.25	93.99 ± 0.18	2.34 ± 0.07	97.08 ± 0.03
15°	48	$\mathbf{93.53 \pm 0.28}$	94.71 ± 0.17	$\mathbf{93.51 \pm 0.28}$	$\mathbf{94.11 \pm 0.08}$	2.31 ± 0.08	$\mathbf{97.13 \pm 0.03}$
20°	36	93.52 ± 0.19	$\mathbf{94.96 \pm 0.18}$	93.09 ± 0.34	93.96 ± 0.10	2.16 ± 0.08	97.11 ± 0.04
30°	24	93.41 ± 0.18	94.84 ± 0.07	92.94 ± 0.18	93.82 ± 0.08	2.21 ± 0.04	97.09 ± 0.05
45°	16	92.40 ± 0.17	92.89 ± 0.04	92.65 ± 0.28	92.70 ± 0.15	$\mathbf{2.16 \pm 0.01}$	97.06 ± 0.01
60°	12	92.20 ± 0.16	92.29 ± 0.61	92.01 ± 0.64	92.14 ± 0.05	2.67 ± 0.34	97.03 ± 0.04

The highest score in each column is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation. The anchor size setting: base size $l = 5.0$, size scales $\boldsymbol{s} = [2.0, 5.0]$, and aspect ratios $\boldsymbol{r} = [2.0, 4.0]$.

TABLE V
IMPACT OF THE THRESHOLD FOR DETERMINING THE ANCHOR'S LOCATION

Threshold	*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)
$T_{\text{loc}} = 0.1$	93.42 ± 0.23	$\mathbf{94.89 \pm 0.21}$	93.45 ± 0.11	$\mathbf{94.12 \pm 0.14}$	$\mathbf{2.21 \pm 0.09}$	97.09 ± 0.03
$T_{\text{loc}} = 0.2$	$\mathbf{93.53 \pm 0.28}$	94.71 ± 0.17	$\mathbf{93.51 \pm 0.28}$	94.11 ± 0.08	2.31 ± 0.08	97.13 ± 0.03
$T_{\text{loc}} = 0.3$	93.36 ± 0.47	94.59 ± 0.24	93.32 ± 0.30	93.90 ± 0.27	2.35 ± 0.12	$\mathbf{97.16 \pm 0.05}$
$T_{\text{loc}} = 0.4$	93.03 ± 0.23	94.39 ± 0.44	93.27 ± 0.58	93.77 ± 0.19	2.43 ± 0.24	97.09 ± 0.04
$T_{\text{loc}} = 0.5$	92.83 ± 0.14	94.36 ± 0.49	93.16 ± 0.43	93.70 ± 0.05	2.45 ± 0.28	97.14 ± 0.06
$T_{\text{loc}} = 0.6$	92.10 ± 0.12	93.89 ± 0.61	93.37 ± 0.52	93.58 ± 0.10	2.70 ± 0.33	97.09 ± 0.04

The highest score in each column is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation.

TABLE VI
SIGNIFICANCE OF THE PRIOR MORPHOLOGICAL INFORMATION

The PMInf		*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)
☒Edge	☒Intersection	91.82 ± 0.35	94.39 ± 0.30	92.50 ± 0.34	93.37 ± 0.08	2.43 ± 0.15	97.06 ± 0.05
☑Edge	☒Intersection	92.66 ± 0.33	94.58 ± 0.23	93.33 ± 0.23	94.06 ± 0.23	$\mathbf{2.15 \pm 0.11}$	97.11 ± 0.03
☒Edge	☑Intersection	92.24 ± 0.43	94.25 ± 0.27	93.19 ± 0.38	93.52 ± 0.28	2.54 ± 0.12	97.07 ± 0.03
☑Edge	☑Intersection	$\mathbf{93.53 \pm 0.28}$	$\mathbf{94.71 \pm 0.17}$	$\mathbf{93.51 \pm 0.28}$	$\mathbf{94.11 \pm 0.08}$	2.31 ± 0.08	$\mathbf{97.13 \pm 0.03}$

The highest score in each column is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation.

strategies, SGAch was the most effective, achieving an *AP* score as high as $92.57 \pm 0.53\%$ with an improvement of 3.89%. Besides, the baseline detector only achieved a maximum *F1-score* of 91.43 ± 0.28 %, with an average false positives per image (i.e., the *mFPs*) of 3.70 ± 0.18 counts. The SGAch strategy increased the maximum *F1-score* to $93.63 \pm 0.22\%$, while reducing the *mFPs* to 2.16 ± 0.24 counts. Moreover, the SGAch strategy can significantly accelerate the training process by reducing the number of skew-IoU calculations for anchors. An epoch of a detector without the SGAch strategy required approximately 900 seconds on an RTX 3090Ti GPU, which was greatly reduced to only about 260 seconds when the SGAch strategy was applied. Furthermore, when PMInf, HDLos, or DASmp were applied with SGAch, the detection performance was further improved, demonstrating that the proposed strategies can complement each other well.

2) Hyperparameters for anchor setting: The above results demonstrate that the SGAch strategy contributes the most to results. This strategy involves several hyperparameters that can impact the model's performance, including the base size, size scales, aspect ratios, and the angle interval of anchors. The former three parameters can be empirically adjusted based on the size distribution of gt-boxes, while the angle interval should be carefully determined through experiments. Table IV shows the results corresponding to the following angle intervals: 5°, 10°, 15°, 20°, 30°, 45°, and 60°. It reveals that 10°, 15°, 20°, and 30° were all acceptable with high *AP* scores. Among them, the angle interval of 15° achieved the highest *AP* score of 93.53 ± 0.28 %. Besides, the detection speed was also considered. A smaller angle interval *not only* leads to higher computational costs (more anchors per location) *but also* exacerbates the anchor-imbalance issue. On the contrary, a too large angle interval results in too few anchors that may not be sufficient to overcome the arbitrary orientation characteristic of chromosomes. Therefore, the angle interval of 15° was chosen as a compromise in this study.

Apart from the above anchor parameters, the threshold for determining the anchor locations (i.e., the T_{loc} in Eq. 3) might also impact the model performance. To investigate this, we trained 5-fold models with different threshold values of 0.1, 0.2, 0.3, 0.4, 0.5, and 0.6, respectively. The results are presented in Table V. It can be observed that as the threshold value increases, the performance of the model deteriorates, particularly when $T_{\text{loc}} > 0.3$. However, the thresholds of 0.1, 0.2, and 0.3 are all optional as they brought similar *AP* scores.

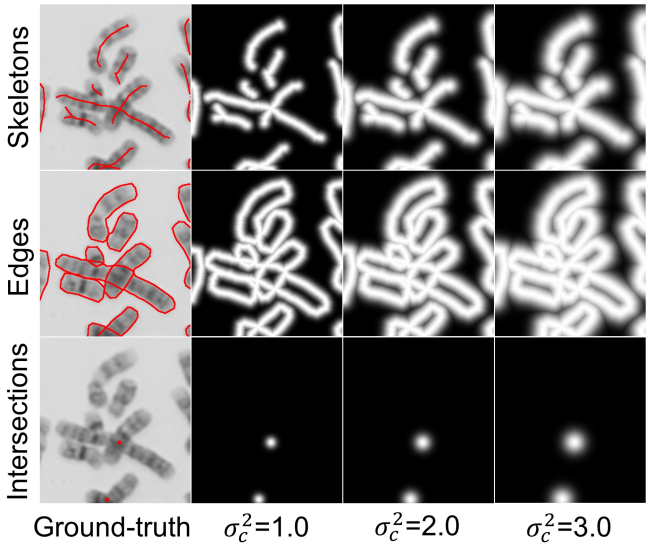


Fig. 8. Visualization of Gaussian maps corresponding to skeletons, edges, and intersections of the chromosomes. The Gaussian map with the variance of 1.0 is better for the skeletons and edges which can still isolate each instance well, while 2.0 is better for the intersections.

3) The significance of edges and intersections: We also conducted experiments to validate the effectiveness of edge and intersection prediction. In these experiments, the prediction of the edge map or the intersection map was removed from the DeepCHM model. Table VI tabulates the results, which demonstrates that the prediction of edge and intersection maps can indeed improve the model performance. When both the edge and intersection predictions were removed, the model only achieved an *AP* score of $91.82 \pm 0.35\%$ and a maximum *F1-score* of $93.37 \pm 0.08\%$. The results also reveal that edge prediction is more important than that of intersection. The *AP* score of the former was $92.66 \pm 0.33\%$, whereas the latter was

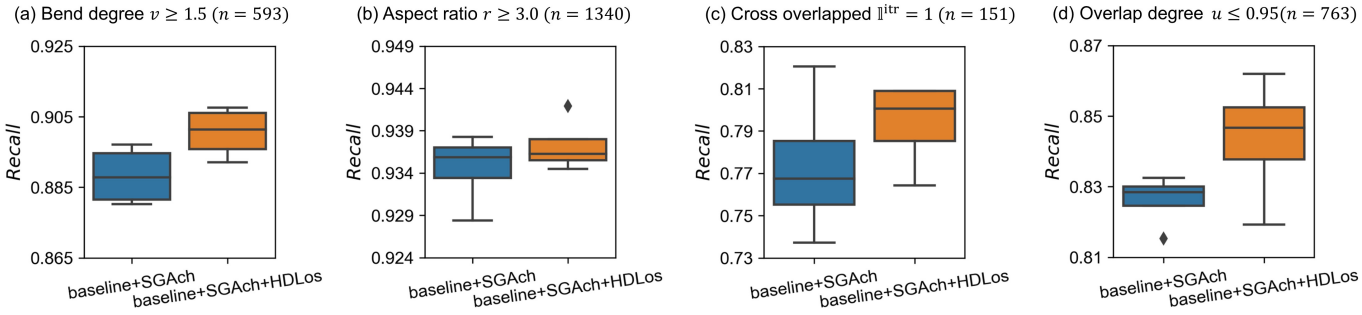


Fig. 9. We calculated the Recalls from hard chromosomes in the testing dataset and drew box-plots of the 5-fold *Recalls* of the baseline+SGAch and the baseline+SGAch+HDLos. (a)-(d) correspond to the chromosome groups with bend degree $v \geq 1.5$, aspect ratio $r \geq 3.0$, cross overlapped (i.e., $\mathbb{I}_j^{itr} = 1$), or overlap degree $u \leq 0.95$, respectively. The overlap degree was calculated between each gt-bbox. n is the number of chromosome instances in each group. The plots demonstrate that the proposed HDLos can indeed improve the model's ability to detect hard chromosomes.

TABLE VII
COMPARISON WITH HARD NEGATIVE ANCHOR SAMPLING STRATEGIES

Detectors	*AP* (%)	*Precision* (%)	*Recall* (%)	*F1-score* (%)	*mFPs* (count)	*mOD* (%)
baseline+SGAch+DASmp	$\mathbf{92.94 \pm 0.28}$	95.11 ± 0.31	$\mathbf{93.01 \pm 0.16}$	$\mathbf{94.05 \pm 0.14}$	2.12 ± 0.17	97.21 ± 0.03
baseline+SGAch+OHEM	91.72 ± 0.61	$\mathbf{95.62 \pm 0.19}$	90.48 ± 0.62	92.98 ± 0.40	$\mathbf{1.83 \pm 0.07}$	$\mathbf{97.24 \pm 0.04}$
baseline+SGAch+HNAS	92.14 ± 0.48	94.27 ± 0.17	92.73 ± 0.26	93.48 ± 0.23	2.04 ± 0.14	97.16 ± 0.05

The highest score in each column is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation. DASmp (ours): dynamically adaptive anchor sampling (loss-based); OHEM: online hard example mining (loss-based); HNAS: hard negative anchor sampling (IoU-based).

TABLE VIII
AVERAGE PRECISION (*AP*) ACHIEVED WITH DIFFERENT SETTINGS OF THE SCALE PARAMETER ρ

DeepCHM with	$\rho = 0.25$	$\rho = 0.50$	$\rho = 1.00$	$\rho = 2.00$	$\rho = 4.00$
HDLos	93.60 ± 0.18	$\mathbf{93.61 \pm 0.26}$	93.53 ± 0.28	93.56 ± 0.25	93.44 ± 0.24
DASmp	93.45 ± 0.31	93.52 ± 0.16	93.51 ± 0.32	$\mathbf{93.59 \pm 0.26}$	93.53 ± 0.28

The highest score in each row is shown in bold. Each value was given as the average ± standard deviation of the results of 5-fold validation.

$92.24 \pm 0.43\%$.

In our implementation, the skeletons, edges, and intersections were learned from Gaussian regression of their ground-truth maps. The ground-truth skeletons, edges, and intersections are formed with positive points, i.e., the pixels with a value of 1 as illustrated in Fig. 8 by the red points. The ratio between the positives and the negatives is severely imbalanced, making the saliency maps hard to learn. Gaussian regression with the Focal loss is an effective solution to address the above issue[31]. However, the Gaussian variance (see Eq. 2) should be set properly to generate the Gaussian maps. As illustrated in Fig. 8, larger variances will lead to wider saliency distributions. We chose the variance of 1.0 for the skeletons and edges, because they can still isolate each instance well, which benefits addressing the dense distribution issue. For the intersections, we used the variance of 2.0 as it can generate maps that accurately cover the intersection regions.

4) Effectiveness of the HDLos on hard chromosomes: The HDLos was proposed for model training to assist the hard chromosomes detection, i.e., the bend, long, densely distributed, and overlapped instances. To verify its effectiveness, we calculated the 5-fold *Recall* scores for the hard chromosomes in the testing dataset. Fig. 9 shows the box plot of the Recalls of the baseline+SGAch and the baseline+SGAch+HDLos detectors. The plots demonstrate that the proposed HDLos can enhance the model's ability to detect hard chromosomes with a large bend degree (Fig. 9(a)) and a large aspect ratio (Fig. 9(b)). Meanwhile, the performance improvement in detecting cross-overlapped and densely distributed chromosomes is more obvious, as demonstrated in Fig.9(c)-(d). The cross-overlapped instances are typically long chromosomes that are densely distributed. Consequently, they had higher loss weights (see Eq. 7) during the training stage, which strengthen the model's ability to identify them.

5) Comparison with exiting anchor sampling methods: As aforementioned, several anchor sampling methods have been proposed to improve detection performance, such as the OHEM method[22] and the HNAS method[3]. To demonstrate the superiority of our DASmp method, we also conducted 5-fold validation on the baseline+SGAch detectors that were trained using the OHEM method[22] or the HNAS method. The results are tabulated in Table VII. Our method achieved the best *AP* score of $92.94 \pm 0.28\%$ and *F1-score* of $94.05 \pm 0.14\%$, which was higher than the OHEM method ($91.72 \pm 0.61\%$ and $92.98 \pm 0.40\%$) and the HNAS method ($92.14 \pm 0.48\%$ and $93.48 \pm 0.23\ \%$). The OHEM method achieved the worst *Recall* score, only $90.48 \pm 0.62\ \%$, whereas the proposed DASmp and the HNAS method achieved *Recall* scores of $93.01 \pm 0.16\ \%$ and $92.73 \pm 0.26\ \%$, respectively. For this phenomenon, we conjecture that the OHEM method only adopted top-k hard negatives and completely discarded easy examples.

6) The impact of the scale parameter ρ on the performance: A scale parameter ρ was defined in Eq. 8 that was used to adjust the loss weights of positive anchors (see Eq. 7) and the sampling weights of negative anchors (see Eq. 14). We also verified its impact on the model performance. Table

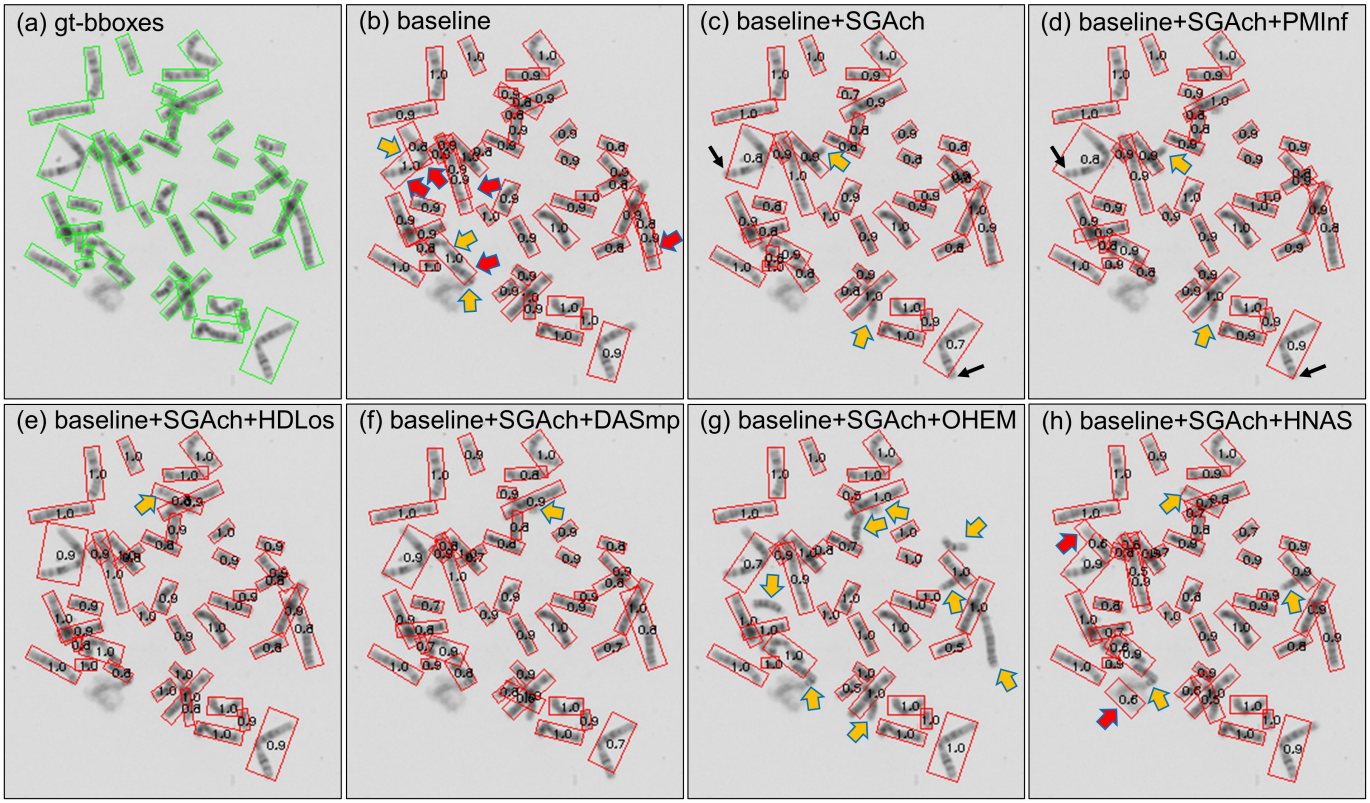


Fig. 11. Detection results of a representative example. (a) shows the ground-truth bounding boxes. (b)-(d) show the predicted bounding boxes and their probabilities of different detectors. The yellow arrows and red arrows indicate false negatives and false positives, respectively. The baseline detector produced more false positives. When the SGAch strategy was applied, the number of false positives was reduced, but some cross-overlapping chromosomes were missed. The PMInf provided more accurate localization, especially for the long bend instances, as indicated by the black arrows in (c) and (d). The HDLos improved the detector with more accurate localization and fewer false negatives of cross-overlapped instances. The proposed DASmp strategy outperformed the OHEM and the HNAS methods with fewer false positives and false negatives.

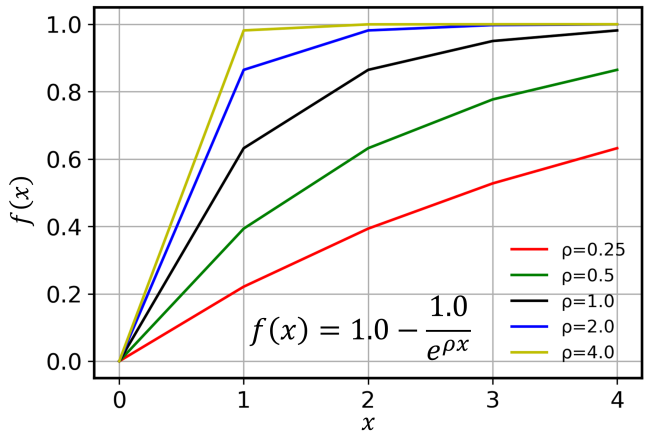


Fig. 10. Plots of the function $f(x)$ that used to calculate the loss weights of positive anchors and the sampling weights of negative anchors.

VIII tabulates the results of the DeepCHM models that trained using the penalized loss (i.e., the HDLos) and the anchor sampling method (i.e., the DASmp) in different ρ, including 0.25, 0.5, 1.0, 2.0, and 4.0. An interesting phenomenon can be observed as follows: the DeepCHM can achieve satisfactory performance, despite the variations of the ρ. However, setting a smaller value of ρ, e.g., 0.5, for the HDLos was better than setting a large value, e.g., 4.0, but it was on the contrary for the DASmp. To analyze this phenomenon, we plotted the curves of Eq.8 with the above settings in Fig. 10. The slope of the function is much steeper for larger values of ρ, especially when the input augment $x \leq 1$. When $x > 1$, the slope becomes flatter with the increased values of ρ. According to the definition in Eq. 7 for the calculation of the loss weights, the input augment (i.e., $\mathbb{I}_j^{\text{itr}} + v_j r_j u_j$) is typically greater than 1.0 in most cases. Therefore, a smaller value of ρ (e.g., 0.5 or 1.0) can produce more discriminative weights. For the DASmp strategy, the focal loss $\mathcal{L}_i^{\text{FL}}$ of negative anchors becomes increasingly smaller during training. Thus, a larger value of ρ (e.g., 2.0 or 4.0) may be more appropriate to guarantee more distinguished sampling weights.

7) Visualization analysis of the improvements: To qualitatively understand the improvement of the proposed strategies, we drew the pred-bboxes and their predicted scores on the testing images. Fig. 11 visualizes the results of a representative case corresponding to the baseline, the baseline+SGAch, the baseline+SGAch+PMInf, the baseline+SGAch+HDLos, and the baseline+SGAch with different sampling methods. Red arrows and yellow arrows indicate false positives and false negatives, respectively. Fig. 11(b) demonstrates that the baseline detector suffered from several false positives and false negatives, most of which appeared on the long and bend chromosomes. When the SGAch strategy was applied, false positives and false negatives were reduced. The bend chromosomes were detected successfully, but the bounding boxes were not precisely regressed (see black

TABLE IX
REVIEW OF STATE-OF-THE-ART STUDIES ON CHROMOSOME DETECTION AND SEGMENTATION

Detectors	Basic method	Task	Rotated anchor	Materials (images/instances)	Raw data	Publicly available	Performance
DeepCHM (ours)	RPN	Detection	Yes	624/27,763	☑	☑	93.53% *AP*
Xiao et al.[3]	FasterRCNN	Detection	No	1375/63,026	☑	☒	99.60% m*AP*
Wang et al.[8]	MaskRCNN	Instance seg	No	1378/Unknown	☒	☒	69.96% *mAP*
Chang et al.[7]	MaskRCNN	Instance seg	No	1148/4568	☒	☒	99.30% *accuracy*
Sun et al.[9]	UNet	Overlap seg	N/A	Unknown/13,434	☒	☒	99.88% *accuracy*
Saleh et al.[6]	UNet	Overlap seg	N/A	Unknown/13,434	☒	☒	99.68% *accuracy*
Arora et al.[4]	Active contours	Overlap seg	N/A	200/1367	☒	☒	81.00% *accuracy*
Minaee et al.[29]	Geometric method	Touch seg	N/A	25/1150	☒	☒	91.90% *accuracy*

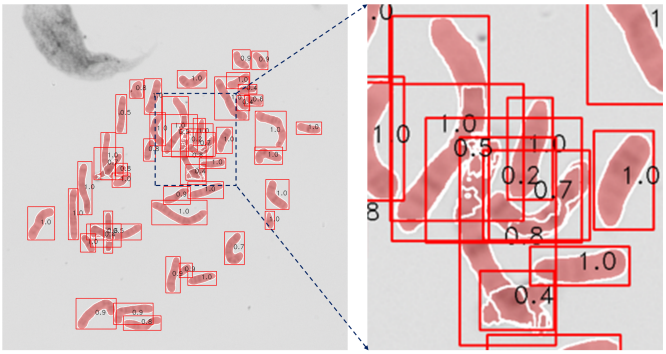


Fig. 12. Result visualization of the MaskRCNN from a representative case. The horizontal pred-bboxes cannot locate chromosomes well and it has adverse impact on the segmentation task.

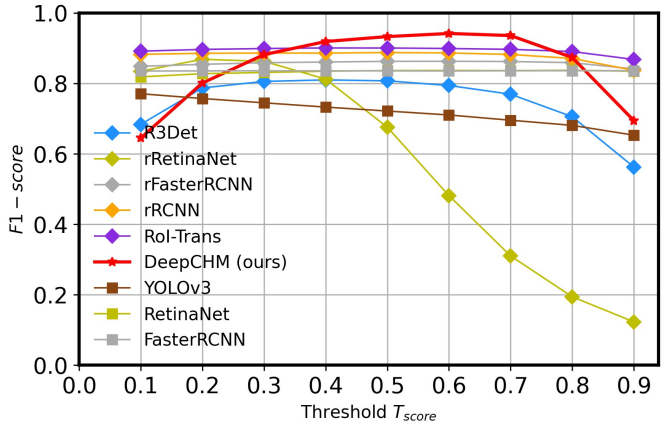


Fig. 13. The *F1-scores* achieved by the detectors varied with the threshold value of T_{score}. Compared to the one-stage detectors (i.e., YOLOv3, R3Det, RetinaNet, and the proposed DeepCHM), the change in T_{score} had less impact on the multistage detectors, i.e., FasterRCNN, rRCNN, and RoI-Trans. However, the proposed DeepCHM outperformed the multistage detectors, especially when T_{score} was set to the range of [0.4,0.8].

arrows in Fig. 11(c)). After the PMInf was applied, the pred-boxes became more accurate (see Fig. 11(d)). Compared to PMInf, the HDLos strategy was more effective. This not only led to more precise pred-boxes but also improved the detection performance in the cross-overlapped instances (see Fig. 11(e)). Finally, it can be observed from Fig. 11(f)-(h) that the proposed anchor sampling strategy DASmp outperformed the OHEM and the HNAS methods, with fewer false positives and false negatives. Notably, the OHEM method missed many chromosomes, as indicated by the yellow arrows in Fig. 11(g). This phenomenon is consistent with the statistical results in Table VII.

VI. DISCUSSION

Fast and accurate chromosome detection in metaphase cell images is vital for automatic karyotype analysis because it is the key preprocessing step for the layout of karyotype images[2]. Although some prior studies have tried to address this problem[3], [8], there is still no widely accepted solution. Table IX summarizes the SOTA studies on chromosome detection and segmentation. It can be observed that most researchers focused on developing conventional geometric-based segmentation methods[29] or deep-learning-based segmentation methods[6] for only separating the touching and overlapping chromosomes. Recently, Xiao et al.[3] directly improved the Faster-RCNN for end-to-end chromosome enumeration. However, as proven in our study, the horizontal detector is not a good choice for a more general chromosome detection task due to the arbitrary orientation and dense distribution of chromosomes. Fig. 12 visualizes a representative case from the results of MaskRCNN. Obviously, each horizontal bounding box may cover several objects in the cluster regions, which is inconvenient for subsequent instance segmentation and classification tasks.

Although many excellent detectors have been developed for oriented object detection in natural images[27], they are still inappropriate for the chromosome detection task. Unlike the objects in the natural images, the chromosomes in metaphase cell images have much more complicated characteristics, i.e., wide variation of lengths, bend, touching, cross-overlapping, local similarity, and diverse G-band patterns (see Fig. 2 (d)). Existing methods may be vulnerable to these issues, which has also been proven by our experiments, as demonstrated in Table I.

Consequently, in this study, we developed the DeepCHM detector and three novel strategies, i.e., the prediction of a saliency map, the hardness-aware loss (HDLos), and the anchor sampling strategy (DASmp), which are specifically designed to address the abovementioned challenges. The saliency map prediction is the most important part of DeepCHM, as it provides hints (i.e., the PMInf) for the detector to locate chromosomes more precisely (see Fig. 4). Like an attention mechanism, it enables the detector to focus on the foreground regions instead of the background (i.e., the SGAch). Taking advantage of saliency map prediction, the other two strategies can further improve the detector in different aspects. Specifically, the HDLos pays more attention to hard positive

anchors, while the DASmp strategy mines model-aware hard negative anchors, and thus the model can learn more discriminative features to improve the detection (see Fig. 8 and Table III).

More importantly, most existing rotated detectors are based on horizontal anchors and have complicated multistage network architectures, e.g., including the RPN, the RCN, and even submodules for transforming the horizontal anchors to rotated bounding boxes, which inevitably impede the detection performance and efficiency. In contrast, the proposed DeepCHM is a fast rotated-anchor-based detector with a much simpler network structure that is similar to the RPN. It can be easily adopted to other similar applications such as the nuclei detection in histopathological images[44].

Notably, some limitations need to be addressed in future work. First, the threshold value T_{score} should be carefully set. As shown in Fig. 13, the *F1-score* of all detectors varied with T_{score}. The proposed DeepCHM, as a one-stage detector, inferior to the multistage detectors (i.e., FasterRCNN[15], rFasterRCNN[27], rRCNN[27], and RoI-Trans[23]) except for the $T_{\text{score}} \in [0.4,0.8]$. But it outperforms other one-stage detectors (i.e., YOLOv3[21], R3Det[35], RetinaNet[18], rRetinaNet[27]) at most cases.

Second, although DeepCHM can achieve promising performance for chromosome detection, it may fail to detect some exceptional cases. For example, when two touching chromosomes distribute along the same direction, they may be recognized as a single instance, as indicated by the red arrow in Fig. 7 (b).

Third, training the model of the proposed method requires access to ground-truth saliency maps obtained from object masks. However, in most object detection datasets, masks are not available, which might hamper the extension of the method to other applications. A potential solution is GrabCut[45] which can roughly segment the objects from gt-bboxes.

Finally, it is recognized that a large-scale dataset is a prerequisite for developing robust deep-learning-based methods. To the best of our knowledge, the research community still lacks a large-scale dataset that is publicly available for the study of chromosomes. Therefore, we built and released the AutoKary2022 dataset, hoping it can facilitate the research and application of automatic karyotype analysis. Notably, the dataset contains densely annotated information (i.e., the masks), and it can be used to train instance segmentation methods such as the MaskRCNN. Considering object detection is the key to ensuring top-performing instance segmentation, we focused on developing a high-performance detector in this study. The proposed method can be adopted as the RoI proposer in two-stage instance segmentation methods, e.g., the MaskRCNN and its variants. Besides, the current version of our dataset was built only from 50 volunteers, and its annotations lack chromosome type information. In the following, we will continue to enlarge the dataset with more samples and improve the annotations.

VII. Conclusion

In this paper, we proposed a one-stage detector named DeepCHM for accurate and fast chromosome detection in metaphase cell images. To tackle the challenges posed by dense distribution, arbitrary orientations, and diverse chromosome shapes, we integrated prior morphological knowledge (i.e., skeletons, edges, and intersections) into the model, which *not only* enhanced feature learning *but also* facilitated rotated anchor setting. To further boost the performance by handling the hard positives and negatives, we designed the hardness-aware loss (i.e., the HDLos) and the loss-based dynamic anchor sampling strategy (i.e., the DASmp) to train the model. On top of a large-scale densely annotated dataset named AutoKary2022, extensive experiments were conducted to demonstrate the superiority of DeepCHM to most SOTA detectors. These studies yielded some attractive findings and clearly highlighted the main challenges of automated karyotyping, which is highly beneficial for both current applications and future research.

Acknowledgment

This research was also supported by the advanced computing resources provided by the Supercomputing Center of Hangzhou City University.

References

[1] M. A. Hultén, S. Dhanjal, and B. Pertl, “Rapid and simple prenatal diagnosis of common chromosome disorders: Advantages and disadvantages of the molecular methods FISH and QF-PCR,” *Reproduction*, vol. 126, no. 3, pp. 279–297, 2003.

[2] R. Remani Sathyan, G. Chandrasekhara Menon, S. Hariharan, R. Thampi, and J. H. Duraisamy, “Traditional and deep-based techniques for end-to-end automated karyotyping: A review,” *Expert Syst.*, vol. 39, no. 3, p. e12799, 2022.

[3] L. Xiao, C. Luo, T. Yu, Y. Luo, M. Wang, F. Yu, Y. Li, C. Tian, and J. Qiao, “DeepACEv2: Automated Chromosome Enumeration in Metaphase Cell Images Using Deep Convolutional Neural Networks,” *IEEE Trans. Med. Imaging*, vol. 39, no. 12, pp. 3920–3932, 2020.

[4] T. Arora, “A novel approach for segmentation of human metaphase chromosome images using region based active contours,” *Int. Arab J. Inf. Technol.*, vol. 16, no. 1, pp. 132–137, 2019.

[5] N. Madian and K. B. Jayanthi, “Overlapped chromosome segmentation and separation of touching chromosome for automated chromosome classification,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2012, pp. 5392–5395.

[6] H. M. Saleh, N. H. Saad, and N. A. M. Isa, “Overlapping chromosome segmentation using U-Net: Convolutional networks with test time augmentation,” in *Procedia Computer Science*, 2019, pp. 524–533.

[7] L. Chang, K. Wu, C. Gu, and C. Chen, “Automatic Segmentation of the Whole G-band Chromosome Images Based on Mask R-CNN and Geometric Features,” in *ACM International Conference Proceeding Series*, 2021, pp. 56–61.

[8] P. Wang, W. Hu, J. Zhang, Y. Wen, C. Xu, and D. Qian, “Enhanced Rotated Mask R-CNN for Chromosome Segmentation,” in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2021, pp. 2769–2772.

[9] X. Sun, J. Li, J. Ma, H. Xu, B. Chen, Y. Zhang, and T. Feng, “Segmentation of overlapping chromosome images using U-Net with improved dilated convolutions,” *J. Intell. Fuzzy Syst.*, vol. 40, pp. 5653–5668, 2021.

[10] G. Wang, H. Liu, X. Yi, J. Zhou, and L. Zhang, “ARMS Net: Overlapping chromosome segmentation based on Adaptive Receptive field Multi-Scale network,” *Biomed. Signal Process. Control*, vol. 68, p. 102811, 2021.

[11] Y. Qin, J. Wen, H. Zheng, X. Huang, J. Yang, N. Song, Y. M. Zhu, L. Wu, and G. Z. Yang, "Varifocal-Net: A Chromosome Classification Approach Using Deep Convolutional Networks," *IEEE Trans. Med. Imaging*, vol. 38, no. 11, pp. 2569–2581, 2019.

[12] J. Zhang, W. Hu, S. Li, Y. Wen, Y. Bao, H. Huang, C. Xu, and D. Qian, "Chromosome Classification and Straightening Based on an Interleaved and Multi-Task Network," *IEEE J. Biomed. Heal. Informatics*, vol. 25, no. 8, pp. 3240–3251, 2021.

[13] C. Lin, G. Zhao, A. Yin, Z. Yang, L. Guo, H. Chen, L. Zhao, S. Li, H. Luo, and Z. Ma, "A novel chromosome cluster types identification method using ResNeXt WSL model," *Med. Image Anal.*, vol. 69, no. 101943, 2021.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017.

[16] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, 2020.

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

[18] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020.

[19] X. Yang, X. Yang, J. Yang, Q. Ming, and J. Yan, "Learning High-Precision Bounding Box for Rotated Object Detection via Kullback-Leibler Divergence," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 18381–18394, 2021.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *In Proceedings of the European Conference on Computer Vision*, 2016, pp. 21–37.

[21] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv Prepr. arXiv1804.02767*, 2018.

[22] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 761–769.

[23] J. Ding, N. Xue, Y. Long, G. S. Xia, and Q. Lu, "Learning roi transformer for oriented object detection in aerial images," in *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2849–2858.

[24] Q. Ming, L. Miao, Z. Zhou, J. Song, and X. Yang, "Sparse label assignment for oriented object detection in aerial images," *Remote Sens.*, vol. 13, no. 14, p. 2664, 2021.

[25] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimed.*, vol. 20, no. 11, pp. 3111–3122, 2018.

[26] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[27] Y. Zhou, X. Yang, G. Zhang, J. Wang, Y. Liu, L. Hou, X. Jiang, X. Liu, J. Yan, and C. Lyu, "MMRotate: A Rotated Object Detection Benchmark using Pytorch," *arXiv Prepr. arXiv2204.13317*, 2022.

[28] M. F. S. Andrade, L. V Dias, V. Macario, F. F. Lima, S. F. Hwang, J. C. G. Silva, and F. R. Cordeiro, "A study of deep learning approaches for classification and detection chromosomes in metaphase images," *Mach. Vis. Appl.*, vol. 31, no. 7, pp. 1–18, 2020.

[29] S. Minaee, M. Fotouhi, and B. H. Khalaj, "A Geometric Approach For Fully Automatic Chromosome Segmentation," *IEEE Signal Process. Med. Biol. Symp.*, pp. 1–6, 2014.

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

[31] H. Law and J. Deng, "CornerNet: Detecting Objects as Paired Keypoints," *Int. J. Comput. Vis.*, vol. 128, no. 3, pp. 642–656, 2020.

[32] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2017.

[33] P. S. Efraimidis and P. G. Spirakis, "Weighted random sampling with a reservoir," *Inf. Process. Lett.*, vol. 97, no. 5, pp. 181–185, 2006.

[34] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, no. 1, pp. 157–173, 2008.

[35] X. Yang, Q. Liu, J. Yan, A. Li, Z. Zhang, and G. Yu, "R3Det: Refined Single-Stage Detector with Feature Refinement for Rotating Object," *arXiv Prepr. arXiv1908.05612*, 2019.

[36] X. Xie, G. Cheng, J. Wang, X. Yao, and J. Han, "Oriented R-CNN for Object Detection," in *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2022, pp. 3520–2529.

[37] J. Li, Wentong and Chen, Yijie and Hu, Kaixuan and Zhu, "Oriented RepPoints for Aerial Object Detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 1829–1838.

[38] W. Feng, C., Zhong, Y., Gao, Y., Scott, M. R., & Huang, "Tood: Task-aligned one-stage object detection," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 3490–3499.

[39] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv Prepr. arXiv2203.03605*, 2022.

[40] Z. Chen, C. Yang, Q. Li, F. Zhao, Z.-J. Zha, and F. Wu, "Disentangle your dense object detector," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 4939–4948.

[41] H. Zhang, Y. Wang, F. Dayoub, and N. Sunderhauf, "Varifocalnet: An iou-aware dense object detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8514–8523.

[42] T. Vu, H. Kang, and C. D. Yoo, "Scnet: Training inference sample consistency for instance segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, vol. 35, no. 3, pp. 2701–2709.

[43] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, and J. Xu, "MMDetection: Open MMLab Detection Toolbox and Benchmark," *arXiv Prepr. arXiv1906.07155*, 2019.

[44] S. Chen, C. Ding, M. Liu, J. Cheng, and D. Tao, "CPP-net: Context-aware polygon proposal network for nucleus segmentation," *IEEE Trans. Image Process.*, vol. 32, pp. 980–994, 2023.

[45] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut - Interactive foreground extraction using iterated graph cuts," in *ACM SIGGRAPH 2004 Papers, SIGGRAPH 2004*, 2004, pp. 309–314.