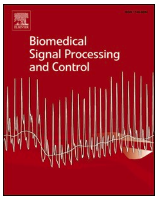
Biomedical Signal Processing and Control 71 (2022) 103120

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc



Classification of Giemsa staining chromosome using input-aware deep convolutional neural network with integrated uncertainty estimates

Hua Wei [a,b], Wen Gao [a], Haitao Nie [a], Jiaqi Sun [a,b], Ming Zhu [a,*]

[a] *Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China*
[b] *University of Chinese Academy of Sciences, Beijing 100049, China*

ARTICLE INFO

Keywords:
Chromosome classification
Convolutional neural network
Deep learning
Scale aware
Probabilistic prediction

ABSTRACT

A fully automated chromosome analysis system can substitute cytogenetic experts for the task of chromosome karyotype analysis, which in turn can substantially increase the efficiency of disease diagnosis. However, the construction of such a system is most crucially restricted by the accuracy of chromosome classification, during karyotype analysis. To facilitate the construction of an automatic chromosome analysis system, an input-aware and probabilistic prediction convolutional neural network (IAPP-CNN) is presented in this paper for high accuracy of chromosome classification. The approach follows three stages and consists of one input-aware module, one feature extractor module and one probabilistic prediction module. In the first stage, the input-aware module develops raw images automatically into the global-scale image, the object-scale image and the part-scale image, by introducing an attention mechanism. In the second stage, the three scale images are input into the feature extraction module through three branches, then the respective feature operators are obtained via their independent CNN feature extractors. In the third stage, the probabilistic prediction module uses three dynamic probabilistic parameters to estimate the prediction of each CNN branch separately, and then combined the three CNN votes for the final decision. The feature expression ability of the key feature was improved and the network was enabled to focus on the recognizable regions in the image. Evaluation results from a large dataset of healthy patients showed that the proposed IAPP-CNN achieved the highest accuracy of 99.2% for the chromosome classification task, surpassing the performance of a competitive baseline created by state-of-the-art methods.

1. Introduction

Karyotyping is an important albeit arduous task in the field of biological research. In addition to being the principal method of detecting genetic diseases, its application is also increasing in other medical diagnoses [1,2]. Traditional karyotype analysis methods rely on time-consuming manual analysis, and the analysis result quality is directly related to the experience of the expert. In recent years, methods based on deep learning to complete automatic karyotype analysis have attracted the attention of researchers [3–5]. Ding et al. [4] proposed a pre-processing model for chromosome segmentation and feature enhancement, and used Convolutional Neural Networks (CNN) for chromosome recognition. Xie et al. [5] divided the automatic chromosome analysis system into three parts: chromosome segmentation, geometric optimization and chromosome classification, as an integrated workflow for automatic karyotype analysis, with chromosome classification being the most the critical part. Although image processing technologies for chromosome classification exist, their accuracy in chromosome classification remains too low for clinical use. Reducing the contribution of experts involved, while ensuring high accuracy of chromosome classification remains a considerable challenge.

Karyotyping is the procedure of visualizing and classifying the chromosomes in a cell, as shown in Fig. 1. In order to better visualize and classify chromosomes, researchers stain chromosomes to increase their contrast and ensure their appearance in cell images. Some researchers use fluorescent staining to dye different types of chromosomes with different colors, as a simple way to help doctors distinguish chromosomes [6,7]. Nevertheless, this method has serious limitations and controversies due to its high cost and unreliable operation. According to the characteristics of chromosomes during the metaphase stage, Giemsa staining has been employed [8–10] to make chromosomes produce lighter and darker gray levels, in order to be classified by cytogenetic experts and image processing technology. Owing to its low cost and strong reliability, Giemsa staining has become the most widely used

---

* Corresponding author.
E-mail address: zhu_mingca@163.com (M. Zhu).

https://doi.org/10.1016/j.bspc.2021.103120
Received 16 April 2021; Received in revised form 12 August 2021; Accepted 30 August 2021
Available online 7 October 2021
1746-8094/© 2021 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

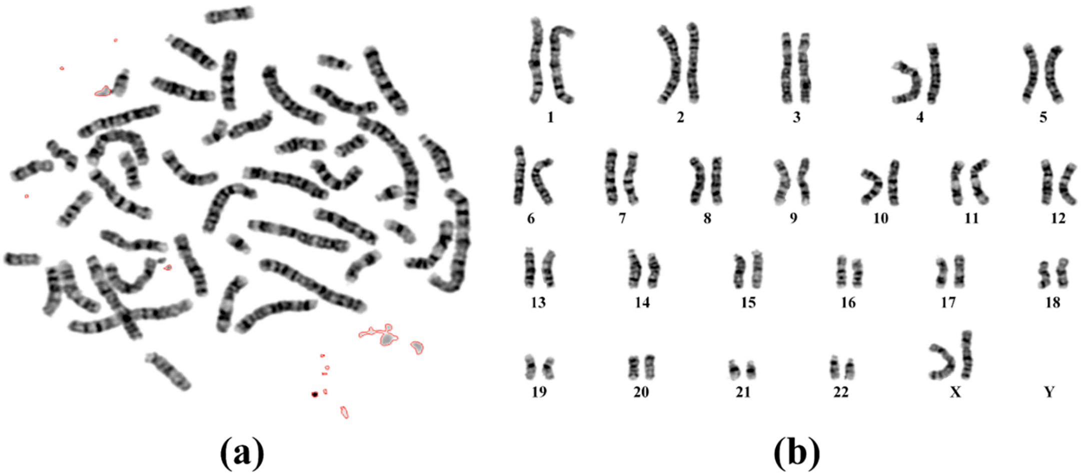


Fig. 1. Metaphase chromosome image using Giemsa staining (a), and the standard karyotyped image (b).

method to obtain stained metaphase chromosomes. Appropriate feature description and extraction are important parts of automatic chromosome classification procedures for Giemsa staining chromosomes, which only have lighter or darker stripes. In the early stage of chromosome classification, researchers have studied the process of feature description and extraction based on medial axis transform (MAT), centromere position, etc. Lerner et al. [11] proposed two approaches about MAT, to examine the MAT-based features [12]. Jahani et al. [13,14] proposed an effective method for centromere locating and length calculation by a linearly varying gray-level mask and an algorithm to identify and straighten the single curved human chromosomes. In addition, the multilayer perceptron (MLP) method has been introduced to improve the accuracy of human chromosome classification [11,15,16]. The above discussed technologies based on traditional classification have improved the efficiency of chromosome classification. However, the performance of these chromosome classification methods deeply depends on the selection of chromosome images, which has been still far from sufficient for wide applications.

The emergence of deep neural networks has brought new opportunities for medical image processing, such as medical image segmentation [17–19] and classification [20–22]. A lot of works based on CNNs have also appeared in the field of chromosome classification [23–32]. Previous works [24,25] only used existing CNNs for fine-tuning on chromosome datasets, and obtained better results than traditional classification methods. Other studies [26,29] used a combination of preprocessing technology and deep learning. They applied traditional methods to straighten bent chromosomes, and then used the processed chromosomes as input into CNN for classification. Using a similar strategy with the above works, Gupta et al. [30] designed a novel CNN named Siamese Networks and achieved 94.6% accuracy in their private dataset. Qin et al. [31] was the first to introduce the part-scale image information for chromosome classification, and utilized the concatenation features from both global-scale and part-scale to predict type and polarity. The introduction of part-scale image information has improved the results of chromosome classification. In addition, Lin et al. [32] proposed an image augmentation method to enlarge the training dataset, to enrich the chromosome data quantity. Fu et al. [28] used Generative Adversarial Networks [33] to generate more chromosome training data. Compared to traditional classification methods, deep learning methods have shown a stronger classification capability for chromosomes. However, there are still challenges in chromosome classification using deep learning to be addressed: (1) performance of fully developed deep convolutional neural network [34–38] on chromosome datasets has room for improvement; (2) it is difficult to recognize chromosomes due to the sub-class categories having a similar appearance, while intra-class images have large differences; (3) chromosome data is relatively scarce, needs to be collected from hospitals by specialists, and labeling is expensive. Therefore, chromosome classification remains a challenging task.

Inspired by the above observations, we propose a practical method to

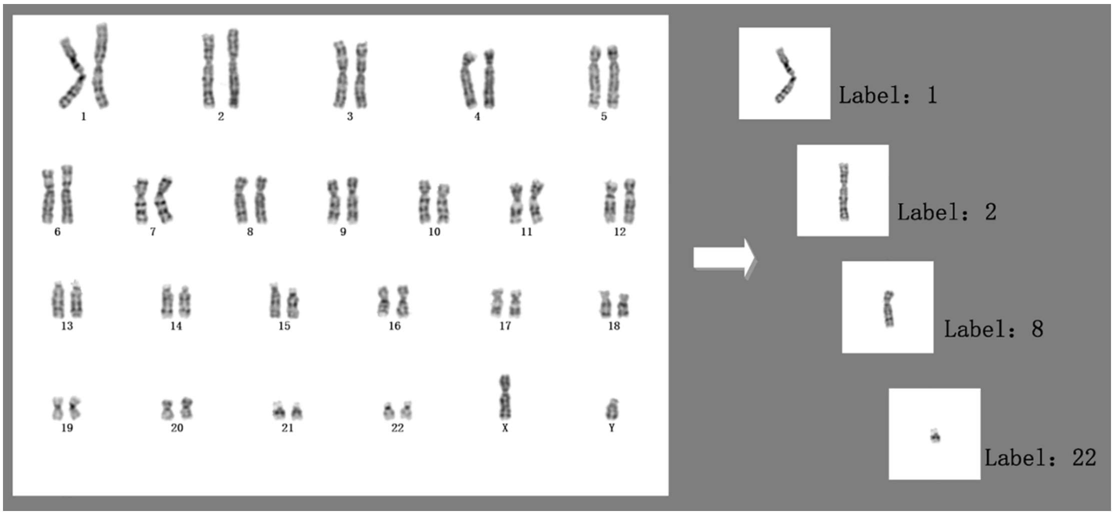


Fig. 2. An example of chromosome image and its corresponding label of chromosome dataset.

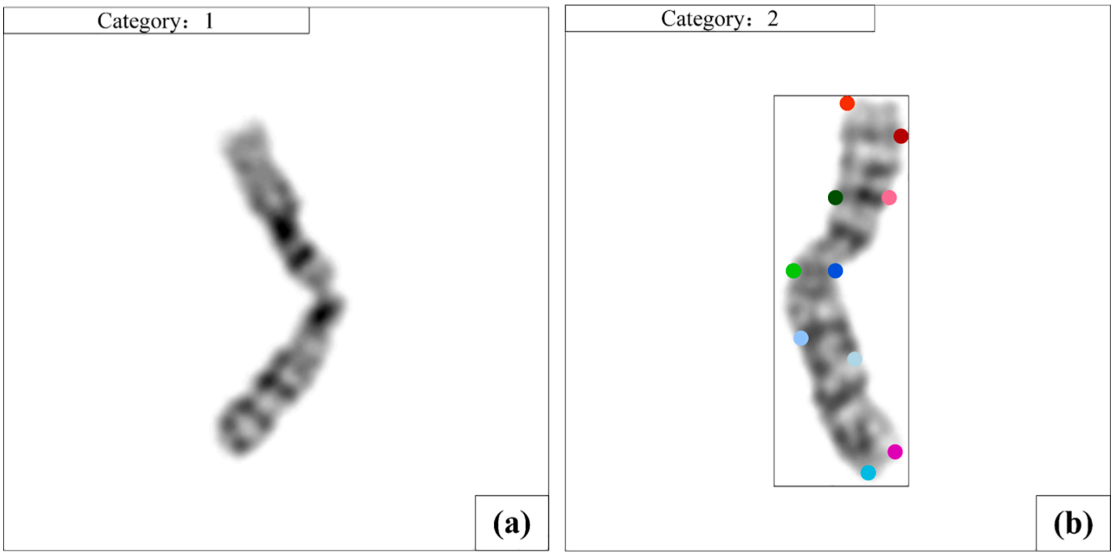


Fig. 3. Dataset annotation for image recognition (images have three-level annotation: image. image with category label annotation (a), image with category label annotation, bounding box annotation (white rectangle), and part annotations (the color dot) (b)).

create a chromosome dataset that effectively reduces manual participation. Based on this, an input-aware and probabilistic prediction convolutional neural network (IAPP-CNN) is proposed. IAPP-CNN highlights the capacity to zoom a raw image into multi-scale images and makes a probabilistic prediction for the category of the chromosome. Object-scale images from original global-scale images and discriminative part-scale images from object-scale images were extracted using an object-aware mechanism and a part-aware mechanism, respectively. Therefore, multi-scale coarse-to-fine images from a single-scale image were developed. To extract more abundant feature information, we innovatively employed three independent state-of-the-art CNNs to learn the three scale images. The above works [11,15,16,30] use one two-layer MLP for the classification of the obtained embedding from CNN or fuse the embedding from two CNN branches before classification [31]. Different from the above works, we added three output logits of MLP directly, to obtain more abundant discriminative prediction information and selected the category with the largest value. Furthermore, inherent ambiguity, such as limited data quality or label noise, leads to overfitting systems with poor generalization on unseen data. To tackle this, we innovatively introduced dynamic probabilistic parameters to capture the confidence score for the predicted output of the three CNN branches. The accuracy and robustness of the model can be further improved without increasing the training. This paper proposed a new approach for the input image and the output prediction of CNNs in the chromosome classification task that has not been explored so far by other researchers [23–30,32]. The experiment demonstrated that the performance of our method was superior to the performance of state-of-the-art CNNs.

2. Introduction of chromosome dataset

The chromosome karyotype images used in this work were collected from the local hospital in February 2020 to create the chromosome classification dataset. The same standard with previous works in the literature was followed in the creation of chromosome dataset for comparability reasons [5,30,31]. As shown in Fig. 2, the annotation information of our dataset includes only the category label. Karyotype chromosome images from 300 patients were obtained and 13,800 chromosome images were produced. Classes 1 to 22, each had 300 images, class X had 200 images, and class Y had 100 images. The chromosome dataset was created by the following proposed methods and its correctness was verified by experts.

Due to the big size of the dataset, manual labeling of each image is a tedious and time-consuming task. Therefore, a method was designed to automatically extract images and annotations from karyotype images, which is convenient for the application of the Giemsa staining chromosome data. A characteristic karyotype image we used for making our dataset is shown on the left of Fig. 2. Each chromosome was labeled into categories according to the existing labeling. The method included the following steps:

(1) Collecting karyotype chromosome pictures of 300 healthy patients and filtering out problematic ones, such as blurred images and blank images;

(2) Generating images and labels: As shown in Fig. 2, each category of the chromosome image has its fixed position. Based on this observation, we can set up automatic chromosome segmentation and obtain the chromosome image and its corresponding label. The karyotype image was binarized and the coarse mask image of each chromosome image was obtained, then morphological open and close operations were applied to the coarse mask image. Each chromosome image corresponding to the binarized mask location was extracted. Finally, each chromosome image was expanded to 224 × 224 pixels with a pixel value of 255 for the background, totaling 13,800 chromosome images.

3. Methods

The main idea behind our IAPP-CNN is to construct a novel deep neural network that can extract more information from chromosome images and make a more robust prediction for chromosome classification. The proposed method is composed of three steps: (1) the input-aware module and feature extraction; (2) model training and loss function; and (3) probabilistic predictions by integrated uncertainty estimates module. In this section, the approach of our method is briefly introduced. Then, the mechanism of multi-scale image generation using the input-aware module is described. Next, the end-to-end training mechanism and loss function are presented. Finally, the prediction mechanism is introduced.

3.1. Approach view

According to the annotations of the datasets [39], works for classification based on CNN can be divided in works based on strong annotations and works based on weak annotations. Works based on strong annotations always employ part annotation or bounding box annotation

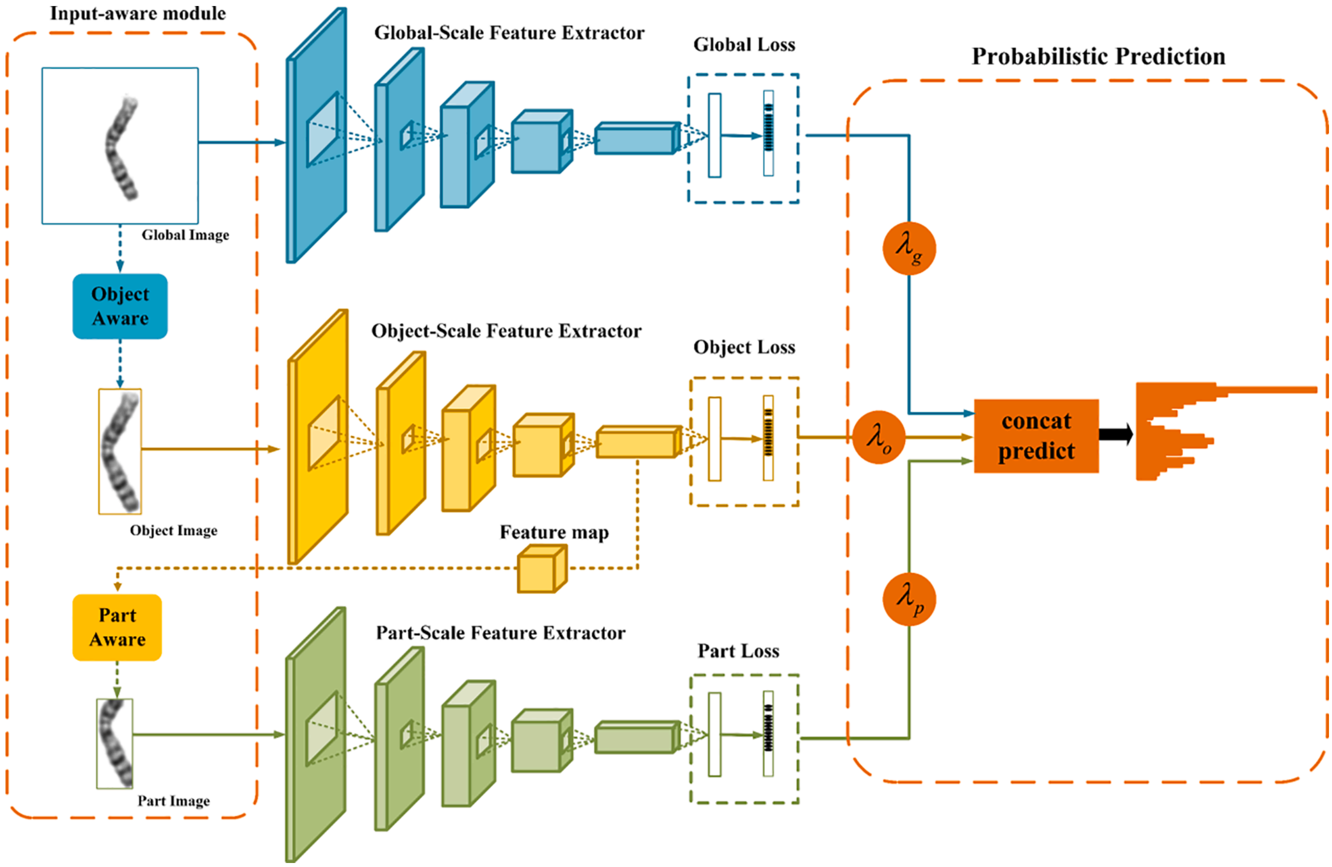


Fig. 4. Overview of IAPP-CNN (the method contains three branches and parameters of branches are independent).

to find the location of the object or discriminative areas, as shown in Fig. 3. Although the accuracy of the classification is improved, hiring experts to label the images is expensive. Thus, methods based on strong annotations are limited. Other works on classification tasks based on weak annotations just used the category label with stronger applicability, but limited annotations always induce a decline of accuracy. To address this problem, a novel IAPP-CNN is proposed here that uses the category label annotations to achieve high accuracy.

The architecture of the IAPP-CNN is presented in Fig. 4. The original input image $I$was regarded as the global-scale image. Using an object-aware mechanism, the object image can be obtained from the global image without using bounding box annotations. To extract the discriminative part area without using part annotations or bounding boxes, we borrowed the last feature map of the object-scale feature extractor. The discriminative part image can be obtained by using the last feature map of the object-scale feature extractor as input to the part-aware mechanism. Three scale images were input to its feature extractor, and three logits were obtained. The three output logits from three CNN branches were added when making the inference prediction, contrary to previous works [11,15,16,30,31]. Furthermore, due to risks from the uncertainty of the classification, an integrated uncertainty estimates module was introduced to evaluate the reliability of each logit, before they were added. The introduction of the integrated uncertainty estimates module further improved the accuracy and robustness of chromosome classification.

3.2. Input-aware module

Chromosome classification differs from generic image classification. It is a fine-grained image classification, aiming to classify the chromosomes into different types. The chromosome images are very similar. Both the global appearance and discriminative part feature play an important role in the chromosome classification task. In this paper, the input-aware module was proposed, forming a strict coarse-grained to fine-grained image process to extract and learn much more information about chromosome images. The module is analysed in the following subsections.

3.2.1. Global-scale image generation and feature learning

As mentioned earlier, most of the methods based on convolutional networks [24–26,29] are fine-tuned using the pre-trained model on large datasets such as ImageNet.

Given a raw image I, the global deep feature representations denoted as $F \in \mathbb{R}^{h \times w \times c}$ were extracted using Formula :

$$F = g(I) \tag{1}$$

where$g(\cdot)$ represents the convolutional layers to map the input image I to a feature representation, and also includes other related operations (i.e., BN, ReLU, and pooling). The high-level representation $F_g \in \mathbb{R}^{h \times w \times c}$ will be used for global-scale classification.

3.2.2. Object-scale image generation

According to the annotations of the dataset, only category label annotations were used in this study. An object-aware mechanism was designed to extract the object-scale image.

The input image of the global-scale feature extractor branch can be written as $I \in \mathbb{R}^{H \times W \times 3}$; where H represents the height of image and W represents the width of image. First, the image was converted to a grayscale image $I_1 \in \mathbb{R}^{H \times W \times 1}$, and then the average value of the gray image was computed by the following Formula :

$$P_{ave} = \frac{\sum_{x=0}^{W-1} \sum_{y=0}^{H-1} Pixel(x, y)}{H \times W} \tag{2}$$

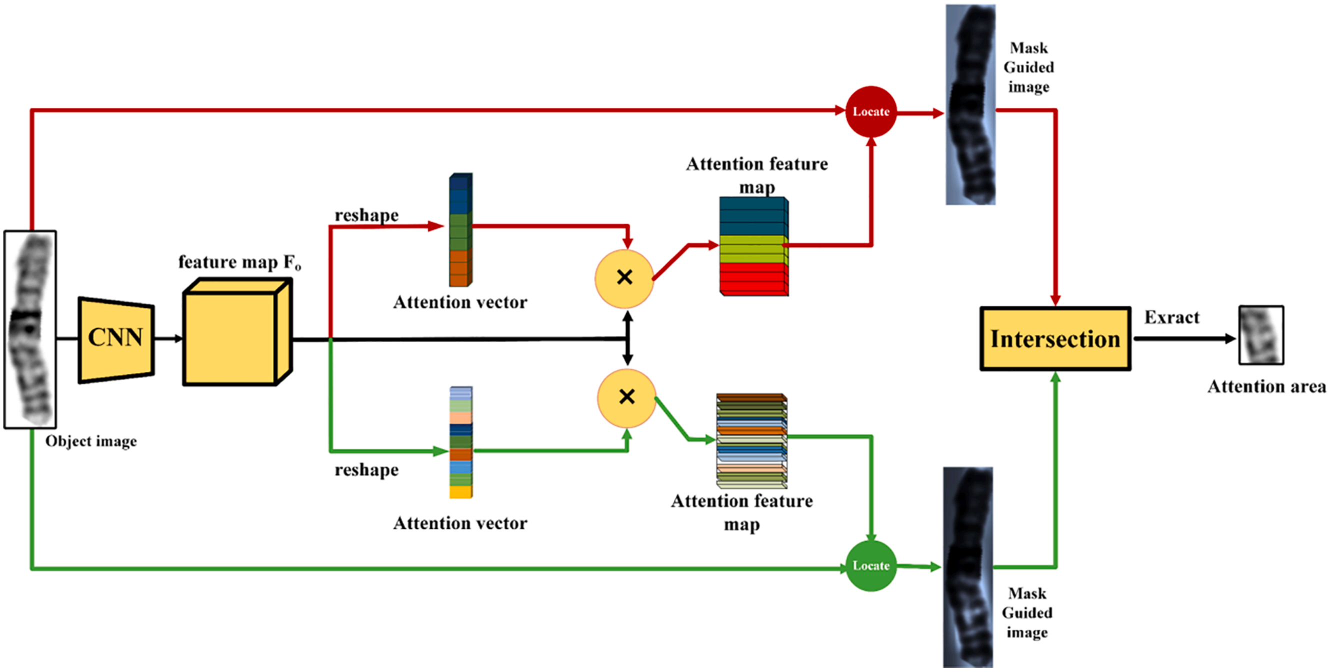


Fig. 5. Discriminative part image obtained module.

where$H$ and W represent the height and the width of the grayscale image, respectively and $Pixel(x,y)$ represents the pixel of the position (x,y).

Here, P_{ave} was employed as the low limit of the threshold to determine whether the element at the position in the grayscale image is part of the object image. The upper limit of the threshold was 254 because the pixel value of the background in a global-scale image was 255.

Then, a coarse binary image was obtained as a mask map initially by the following Formula :

$$B_{(x,y)} = \begin{cases} 255 & \text{if } \mathrm{P}_{ave} < Pixel(x,y) < 254 \\ 0 & others \end{cases} \tag{3}$$

Subsequently, image morphological corrosion was performed to remove noise or other irrelevant details in the binarized image and used morphological expansion to bridge the discontinuous parts of the binarized image. Thus, for the mask image, the locations where the pixel value was 255 were the foreground object image and the locations in which the pixel value was 0 were background images. Finally, we obtained a binary mask by Formula and the object image by inserting the binary mask image $\mathrm{M}^1_{(x,y)}$in the global image.

$$\mathrm{M}^1_{(x,y)} = \begin{cases} 1 & \text{if } \mathrm{P}_{ave} < Pixel(x,y) < 254 \\ 0 & others \end{cases} \tag{4}$$

Then the object image was obtained by:

$$I_o = I_g \odot M^1_{(x,y)} \tag{5}$$

where $\odot$ denotes element-wise multiplication. Image I_o was input into the object-scale feature extractor branch and the deep representation $F_o \in \mathbb{R}^{h \times w \times c}$ that will be used for object-scale classification was extracted.

3.2.3. *Part-scale image generation*

Inspired by works on image retrieval and classification tasks [40–43], a part-aware mechanism was designed to extract discriminative part image information for chromosome classification. As illustrated in Fig. 5, the output feature representation $F_o \in \mathbb{R}^{h \times w \times c}$ was obtained through the object-scale feature extractor branch. Taking it as the basic input feature map, the discriminative part image was extracted by part-aware mechanism.

In the object-scale feature extractor branch, the last feature map written as $F_o \in \mathbb{R}^{H \times W \times C}$ was obtained. Because SE-Net [44] can pay attention to the discriminative area, one squeeze and two excitation mechanisms were proposed by using different reduction ratios r. Two different reduction ratios r were used to encourage and facilitate the last feature map $F_o \in \mathbb{R}^{C \times H \times W}$ to generate two different re-weight feature maps.

Feature map $F_o \in \mathbb{R}^{C \times H \times W}$ can be described as $F_o = [\,x_1 \quad x_2 \quad \dots \quad x_c\,] \in \mathbb{R}^{H \times W \times C}$. Global average pooling was applied to the squeeze feature map F_o into a channel-wise descriptor $z = [\,z_1 \quad z_2 \quad \dots \quad z_c\,] \in \mathbb{R}^c$ by reducing the feature map F_o to its spatial dimension$H \times W$. Thus, the $l-$ th element of channel-wise descriptor z can be computed by:

$$z_l = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_l(i,j) \tag{6}$$

where H and W represent the height and width of the feature map, respectively. Next, different from SE-Net, we designed a two-excitation mechanism by using different reduction ratios. Formula was used to calculate z and get a one-dimensional vector:

$$m = \sigma(W_2 \delta(W_1 z)\,) = [\,m_1 \quad m_2 \quad \dots \quad m_c\,] \in \mathbb{R}^c \tag{7}$$

where $\delta$and σ represent the ReLU function and Sigmoid function, respectively. In $W_1 \in \mathbb{R}^{\frac{c}{r} \times c}$and $W_2 \in \mathbb{R}^{c \times \frac{c}{r}}$, r represents the reduction ratio.

The same design of SE-Net was adopted by forming a dimensionality-reduction layer and a dimensionality-increasing layer with parameters W_1 and W_2, respectively.

The learned parameter W models the correlation and weight between the channels of the feature map. The reduction ratio r, which has an important relationship with the parameter W, can determine the correlation between channels. Using different reduction ratios r, different attention vectors m can be obtained.

In Formula , the attention vector m is obtained. By re-calibrating the input feature map $F_o \in \mathbb{R}^{H \times W \times C}$ with the attention vector m, the re-weighted feature map is obtained by Formula :

$$U = [\,m_1x_1 \quad m_2x_2 \quad \dots \quad m_cx_c\,] \in \mathbb{R}^{c\times h\times w} \tag{8}$$

Two different reduction ratios r were used to generate two different feature maps with different attention vectors m, written, as $U_1 \in \mathbb{R}^{h\times w\times c}$ and $U_2 \in \mathbb{R}^{h\times w\times c}$.

Those feature maps can focus on the same or different discriminative parts. To select the most important discriminative part area, the intersection of two feature maps was chosen.

The feature map $U_1 \in \mathbb{R}^{h\times w\times c}$ was aggregated across the channel dimensions c to produce an average feature map $M \in R^{h\times w}$ by Formula :

$$M = \sum_{i=1}^{c} m_i x_i, M \in \mathbb{R}^{h\times w} \tag{9}$$

Then, the mean value M_{ave} of the feature map is computed as follows:

$$M_{ave} = \frac{\sum_{x=0}^{w-1}\sum_{y=0}^{h-1} M(x,y)}{w\times h} \tag{10}$$

Here, M_{ave} was assigned as the decisive limit of whether the element is chosen or not. Thus, an activation mask map is obtained by Formula :

$$\mathrm{M}^2_{1_{(x,y)}} = \begin{cases} 1 & \text{if } M_{(x,y)} > M_{ave} \\ 0 & \quad others \end{cases} \tag{11}$$

Similar to $U_2 \in \mathbb{R}^{h\times w\times c}$, another activation mask map $\mathrm{M}^2_{2_{(x,y)}}$ can be obtained. The two masks may have the same or different activation areas. To get the discriminative part, the intersection of the M_1^2 and M_2^2 was obtained by Formula :

$$M^2_{(x,y)} = Max(M_1^2 \cap M_2^2) \tag{12}$$

Then, the part image I_p was calculated using Formula :

$$I_p = I_o \odot M \tag{13}$$

The region with the largest connection in M was selected as the binary mask image. Putting the binary mask image M on the object image, results in the discriminative part image. Then, the deep representation $F_p \in \mathbb{R}^{h\times w\times c}$ was obtained by using the part image as input in the part-scale feature extractor.

3.3. *End-to-end training and loss function*

The feature maps in the global-scale, object-scale and part-scale feature extractor branches can be written as $F_g \in \mathbb{R}^{H\times W\times \mathrm{C}}$, $F_o \in \mathbb{R}^{H\times W\times \mathrm{C}}$ and $F_p \in \mathbb{R}^{H\times W\times \mathrm{C}}$, respectively. Taking the output feature representation $F_g \in \mathbb{R}^{H\times W\times \mathrm{C}}$ as an example, a vector that can be written as $z = [\,z_1 \quad z_2 \quad \dots \quad z_c\,] \in \mathbb{R}^c$ is obtained by using global average pooling. Here, z is used as the input feature into a fully connected (FC) layer like $y = fc(x, w, b)$ and the global logits$f = [\,f_0 \quad f_1 \quad \dots \quad f_{n-1}\,] \in \mathbb{R}^N$ are calculated, where$N$ represents the number of the classes.

Taking f as the output logits of the convolution neural networks, the traditional cross-entropy (CE) was employed as the loss function, which is the same as the other two branches, thus, the losses are:

$$Loss_{all} = L_{global} + L_{object} + L_{part} \tag{14}$$

where, L_{global}, L_{object} and L_{part} represent the cross-entropy loss function for each branch, respectively. Each branch uses its loss function for training, and three branches form an end-to-end training. The parameter of each branch is independent, which ensures that each branch can learn its scale image features independently and extract information more efficiently.

3.4. *Probabilistic prediction*

Table 1
The statistics of the chromosome datasets.

Dataset	Picture		Images		total
	Male	Female	Train	Test	
Chromosome	150	150	6900	6900	13,800

In the above section, each branch output logits $y \in \mathbb{R}^N$ was obtained. In order to receive the final prediction, output logits of the three branches were added as shown in Formula :

$$y_{concat} = y_{global} + y_{object} + y_{part} \tag{15}$$

The chosen category was the one with the largesty_{concat} value. Adding the output logits of each branch comprehensively considered the features of different-scale images, ensuring the stability and robustness of the prediction.

Uncertainty quantification is the basic issue yet unsolved for CNNs. For medical images, the model prediction based on CNN requires more risk assessment to achieve high accuracy. To reduce the risks caused by epistemic uncertainty, the integrated uncertainty estimates module was introduced, which is used to estimate the reliability of the model predictions. Before adding the predicted values of the three branches, the uncertainty of the model output was weighed by parameter λ. The formula is as follows:

$$y_{concat} = \lambda_g * y_{global} + \lambda_o * y_{object} + \lambda_p * y_{part} \tag{16}$$

where λ_g, λ_o and λ_p in Formula are used to model the confidence score for the output logits y_{global}, y_{object} and y_{part}, respectively.

To calculate the dynamic probabilistic parameter λ, the existing feature representation was utilized to obtain the uncertainty estimates.

In one of the feature extractor branches, the output logits $y \in \mathbb{R}^N$ were obtained and fed into a SoftMax function to get a confidence score by Formula .

$$p_i = \frac{exp(y_i)}{\sum_{j=1}^{N} exp(y_j)} \tag{17}$$

In $\sum_{i=1}^{m} p_i = 1$ $i \in \{1 \quad 2 \quad \dots \quad N\}$, $P = \{p_1 \quad p_2 \quad \dots \quad p_N\} \in \mathbb{R}^N$ is the prediction score vector and$N$ is the number of object categories. Here, p_i^s was used as the confidence score obtained from the prediction vector p, where i is the corresponding index associated with the maximum score for the image s. Thus, the three output logits y_{global}, y_{object} and y_{part} produced its probabilistic parameters λ_g, λ_o, and λ_p to estimate the confidence for prediction without adding trainable parameters.

4. Experimental results and discussions

4.1. *Dataset*

Based on the method described in Section 2, a large chromosome dataset containing 13,800 chromosome images was extracted from 300 patients. The dataset was split as other typical classification datasets [39,45]. The details of the dataset are displayed in Table 1.

4.2. *Implementation details*

In all the experiments, the input images of the global-scale branch were resized to 224 × 224. The object-scale images obtained from the global-scale branch and part-scale images obtained from the object-scale branch were also resized to 224 × 224 before being fed into the CNNs. When training the model, the images were flipped horizontally with a probability of 0.5. The IAPP-CNN was trained on the chromosome dataset for 50 epochs and the batch size of the train dataset was set to 26. The standard stochastic gradient descent method was used. The learning rate was initially set to 0.02, the weight decay to 1×10^{-5} and the momentum to 0.9. The model was implemented in PyTorch. The same initialization as the IAPP-CNN was used for the other networks.

Table 2
Performances of our method compared with state-of-the-art methods.

Method	F1-score (%)	Accuracy (%)
Sharama *et al.* [29]	96.3 ± 0.2	95.9 ± 0.2
Gupta *et al.* [30]	95.9 ± 0.3	95.9 ± 0.2
Lin *et al.* [32]	97.1 ± 0.3	97.8 ± 0.1
fine-tuning AlexNet [34]	95.8 ± 0.2	96.2 ± 0.1
fine-tuning InceptionV4 [35]	97.5 ± 0.7	97.7 ± 0.5
fine-tuning VGG16bn-Net [36]	97.9 ± 0.9	98.2 ± 0.4
fine-tuning DenseNet [37]	97.8 ± 0.4	98.0 ± 0.3
fine-tuning ResNet [38]	97.3 ± 0.2	97.5 ± 0.3
AlexNet (+IA module)	97.5 ± 0.1	97.6 ± 0.2
InceptionV4 (+IA module)	98.4 ± 0.4	98.5 ± 0.3
VGG16bn-Net (+IA module)	98.5 ± 0.3	98.6 ± 0.4
DenseNet (+IA module)	98.5 ± 0.3	98.6 ± 0.2
ResNet (+IA module)	98.4 ± 0.2	98.5 ± 0.1
IAPP-CNN -based AlexNet	98.3 ± 0.1	98.4 ± 0.2
IAPP-CNN -based Inception V4	99.1 ± 0.2	99.1 ± 0.1
IAPP-CNN -based VGG16bn-Net	99.2 ± 0.2	99.3 ± 0.2
IAPP-CNN -based DenseNet	99.2 ± 0.2	99.3 ± 0.2
IAPP-CNN -based ResNet	99.2 ± 0.1	99.2 ± 0.2

Note: We reported the average performance for 5 random initializations. Here, IA and IUE represent input-aware module and integrated uncertainty estimates module, respectively.

4.3. Evaluation metrics

Performance of chromosome classification in the literature often uses the following four metrics: recall, precision, accuracy, and F1-score. Higher values represent higher accuracy and robustness of the model.

To compute the four metrics, the following four criteria are defined for multi-class classification:

True positives (TP_j): a chromosome image, which belongs to class j, is predicted to be class j.

False positives (FP_j): a chromosome image, which does not belong to class j, is predicted to be class j.

False negatives (FN_j): a chromosome image, which belongs to class j, is predicted to be class $k(\forall k \neq j)$.

True negatives (TN_j): a chromosome image, which does not belong to class j, is predicted to be class $k(\forall k \neq j)$.

Then, recall, precision, accuracy, and F1-score can be computed as following:

$$recall_j = \frac{TP_j}{TP_j + FN_j} \tag{18}$$

$$precision_j = \frac{TP_j}{TP_j + FP_j} \tag{19}$$

$$F1_j = 2\cdot\frac{Re_j \cdot Pr_j}{Re_j + Pr_j} \tag{20}$$

$$F1 = \frac{1}{N_{cls}}\sum_{j=1}^{N_{cls}} F1_j \tag{21}$$

$$accuracy = \frac{1}{M}\sum_{j=1}^{N_{cls}} TP_j \tag{22}$$

where N_{cls} represents the 24 classes of chromosome and M represents the number of all tested chromosome images.

4.4. Quantitative evaluation

4.4.1. Compared with state-of-the-art methods

The performance of the proposed method was compared to state-of-the-art methods by two metrics: the accuracy of all the tested images, and the average F1-score. All experiments were carried out on the same chromosome dataset, as shown in Table 1.

Table 2 provides a comparison of the proposed method with state-of-the-art methods. The methods in Table 2 can be divided into four groups from top to bottom: (1) the first three methods were proposed specifically for classifying images with Giemsa staining chromosomes. The first two methods straightened bent chromosomes, and then fed the processed chromosomes into CNNs, the third method combined the architecture of ResNet and Inception to improve the performance of the original Inception; (2) methods based on fine-tuning the most well-known CNNs. The proposal of AlexNet represented the upsurge in the application of neural networks; VggNet explores the depth of CNNs, which is deeper than AlexNet; Inception increases the width of the network; ResNet develops shorter connections between layers and the model by-pass signal from one layer to the next via identity connections; DenseNet is an architecture that connects all layers directly with each other. Each layer obtains additional input from all preceding layers and passes on its feature-maps to all subsequent layers. Methods based on fine-tuning the model are to train the existing CNNs on the chromosome dataset to obtain the best classification performance. Those methods used the fully developed state-of-the-art CNNs, and represent the best performance on the chromosome classification task; (3) methods adding the input-aware module based on CNNs; and (4) IAPP-CNN based on CNNs.

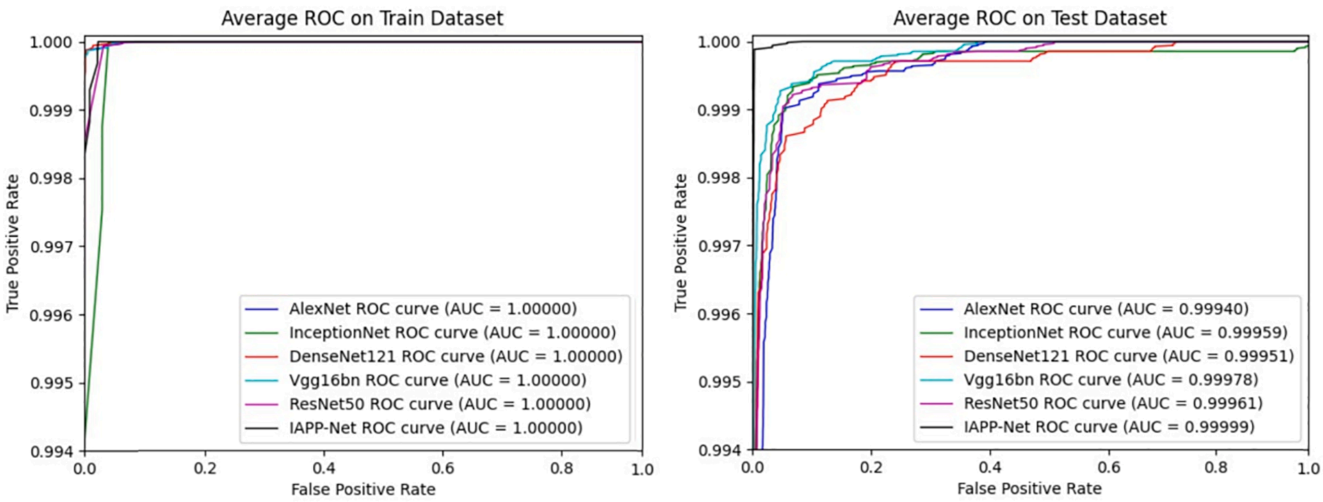


Fig. 6. ROC curve for the IAPP-CNN and other state-of-the-art methods on train dataset and test dataset respectively.

Table 3
The precision, recall, and F1-score of each category of the chromosome.

Class (NO.)	Precision (%)		Recall (%)		F1-score (%)	
1	97.68 ± 1.23	99.67 ± 0.33	97.87 ± 1.13	99.93 ± 0.27	97.77 ± 0.59	99.80 ± 0.30
2	93.70 ± 1.13	98.42 ± 1.1	99.07 ± 0.27	99.60 ± 0.27	96.31 ± 0.75	99.01 ± 0.49
3	97.61 ± 1.28	99.20 ± 0.52	94.87 ± 0.86	99.60 ± 0.27	96.21 ± 0.78	99.40 ± 0.26
4	96.49 ± 2.15	99.13 ± 0.53	96.47 ± 0.20	98.73 ± 0.60	96.47 ± 0.99	98.93 ± 0.43
5	97.58 ± 1.41	99.13 ± 0.20	96.47 ± 1.13	98.80 ± 0.20	97.02 ± 1.13	98.96 ± 0.20
6	97.26 ± 1.06	99.33 ± 0.66	96.60 ± 0.73	99.27 ± 0.40	96.92 ± 0.73	99.30 ± 0.37
7	98.14 ± 0.72	99.80 ± 0.20	98.60 ± 0.60	99.53 ± 0.53	98.37 ± 0.46	99.67 ± 0.33
8	95.27 ± 1.97	98.99 ± 0.33	96.40 ± 1.07	98.73 ± 0.40	95.82 ± 0.84	98.87 ± 0.20
9	97.29 ± 0.73	99.00 ± 0.33	95.87 ± 0.45	98.67 ± 0.67	96.58 ± 0.42	98.83 ± 0.17
10	97.04 ± 0.75	99.14 ± 0.52	98.07 ± 0.60	99.60 ± 0.40	97.55 ± 0.44	99.37 ± 0.36
11	98.99 ± 0.01	99.45 ± 0.21	98.67 ± 0.66	99.60 ± 0.27	98.83 ± 0.34	99.53 ± 0.13
12	99.13 ± 0.60	99.67 ± 0.1	98.53 ± 0.53	99.27 ± 0.26	98.83 ± 0.34	99.46 ± 0.13
13	97.01 ± 0.60	98.94 ± 0.59	97.13 ± 1.47	99.27 ± 0.40	97.07 ± 0.92	99.10 ± 0.43
14	99.05 ± 0.39	99.80 ± 0.47	97.67 ± 0.67	98.87 ± 0.20	98.35 ± 0.37	99.33 ± 0.17
15	97.48 ± 0.84	99.40 ± 0.27	97.80 ± 0.20	99.13 ± 0.53	97.64 ± 0.51	99.26 ± 0.27
16	98.98 ± 0.54	99.40 ± 0.39	96.47 ± 0.87	99.53 ± 0.53	97.70 ± 0.61	99.47 ± 0.37
17	97.89 ± 1.11	99.47 ± 0.53	98.80 ± 0.53	99.67 ± 0.01	98.34 ± 0.66	99.57 ± 0.26
18	99.33 ± 0.81	99.53 ± 0.47	98.13 ± 1.47	99.27 ± 0.27	98.72 ± 0.61	99.40 ± 0.27
19	98.00 ± 0.72	99.53 ± 0.20	98.00 ± 0.33	98.87 ± 0.20	98.00 ± 0.66	99.20 ± 0.13
20	97.43 ± 0.82	98.81 ± 0.52	98.53 ± 0.87	99.40 ± 0.27	97.98 ± 0.63	99.10 ± 0.23
21	99.26 ± 0.53	99.60 ± 0.36	98.67 ± 0.99	99.87 ± 0.20	98.96 ± 0.64	99.73 ± 0.23
22	98.29 ± 0.72	98.94 ± 0.58	99.33 ± 0.33	99.80 ± 0.47	98.81 ± 0.30	99.37 ± 0.30
X	94.58 ± 1.60	98.31 ± 0.79	95.70 ± 2.20	98.40 ± 0.9	95.13 ± 0.87	98.35 ± 0.60
Y	93.64 ± 3.32	98.99 ± 1.01	92.60 ± 1.40	96.80 ± 2.2	93.08 ± 1.34	97.87 ± 0.63

Note: The left of each metric is the result of fine-tuning based on resent50, and the right side is the result of IAPP-CNN. We reported the average performance for 5 random initializations.

For the first group, previous works were trained on our chromosome classification dataset, and reached a similar performance with the fine-tuning CNN groups. In fine-tuning CNN groups, the fine-tuning AlexNet achieved the lowest accuracy of 96.2% and the lowest F1-score of 95.8%. Fine-tuning VGG16bn achieved the highest accuracy of 98.2% and the highest F1-score of 97.9%. All methods based on fine-tuning on well-known CNNs for chromosome image classification were limited to 98% or less in accuracy and F1-score. Therefore, the fully developed classification networks are still lacking to meet the need for high accuracy and high robustness for chromosome classification tasks. Next, the method of adding an input-aware module based on Alexnet significantly improved the accuracy and F1-score. Other methods after adding the input-aware module based on CNNs were also improved, with the accuracy and F1-score reaching more than 98%. Finally, when the integrated uncertainty estimates module was introduced, the accuracy and F1-score reached their highest value. Both the accuracy and F1-score of the method based on Inception V4, VGG16bn, DenseNet, and ResNet all exceeded 99%. It can be seen that although the method based on state-of-the-art CNNs obtained good results relying on its powerful feature extraction ability, it still has limitations.

The introduction of the input-aware module and the integrated uncertainty estimates module greatly improved the performance of the chromosome classification task. Through experimental verification, the proposed IAPP-CNN showed high accuracy and robustness and was able to meet the preliminary clinical requirements.

Furthermore, a receiver operating characteristic (ROC) curve and the corresponding area under curve (AUC) was employed for performance analysis. The ROC curve and AUC are universal evaluation metrics in binary classification problems. However, they must be modified in order to support multi-classification problems. ROC analysis per class using a one-vs-all scheme was used, with higher AUC values denoting better performance. In Fig. 6, all ROC curves are plotted and the AUC for the state-of-the-art methods and our proposed method were calculated on the train and test chromosome dataset. The results show that the proposed IAPP-CNN outperformed the state-of-the-art methods with the least false positive predictions and the highest true positive rates. This demonstrates that in the case of using the state-of-the-art feature extractor, our designed input-aware module and integrated uncertainty estimates module could further improve the performance of chromosome classification.

4.4.2. Evaluation results

To analyze the impact of our method on the classification performance of each class of the chromosomes, adopting resnet50 as the feature extractor, the fine-tuning resnet50 on the chromosome dataset was compared to our proposed IAPP-CNN. The accuracy, recall, and F1-score of each category were reported. The F1-score can reflect the robustness of the model prediction. As shown in Table 3, for the fine-tuning resent50, type 8, type X, and type Y were less than 96%, and other types ranged from 96% to 98%. The IAPP-CNN increased type 8, type X, and type Y by 3%, and other types also increased by about 2%, with most types of chromosomes reaching more than 99%.

To be able to observe the classification results of each category more clearly, we observed the confusion matrix by selected one test result randomly. The confusion matrix of fine-tuning resent50 and the confusion matrix of our IAPP-CNN were calculated on the test dataset. The tested datasets, included 300 images for each type of 1 $\sim$ 22 chromosomes, 200 images of type X chromosome, and 100 images of type Y chromosome. In Fig. 7, the row represents the true label and the column represents the predicted label. The left image is the confusion matrix of the fine-tuning resnet50, the evaluation results of type 4, type 6, type 8 type 9, and type 16 show that there were more than 10 prediction errors in 300 tested images; for type X, there were 10 prediction errors in 200 tested images and for type Y, there were 7 prediction errors in 100 tested images. The right image is the confusion matrix of IAPP-CNN, in which the classification results of all chromosome categories remain relatively stable. For type 4 there were 7 prediction errors in 300 tested images and no prediction errors for the remaining categories or only 1 to 4 incorrect predictions. Therefore, from the classification results of each category, the proposed IAPP-CNN substantially improved the classification effect of each type, lowered the error rate, and proved more robust, in comparison to the fine-tuning resent50.

4.5. Ablation study

4.5.1. Availability of input-aware module

To assess the effectiveness of the proposed input-aware module, the experimental results were observed by choosing different scales. The input-aware module presented in our work can extract global-scale, object-scale and part-scale images using an effective and intuitive method. In this part, one or multiple scales were selected to observe the model performance using two metrics on the test dataset, without adding the integrated uncertainty estimates module.

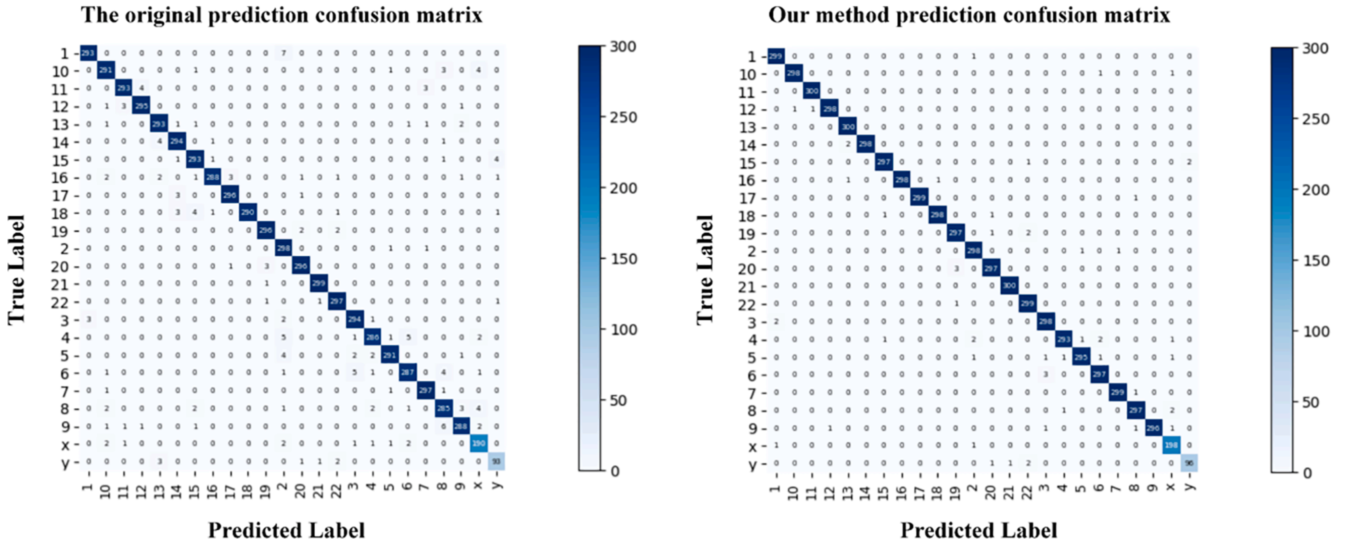


Fig. 7. Confusion matrix ((a) confusion matrix of the fine-tuning resnet50; (b) confusion matrix of IAPP-CNN).

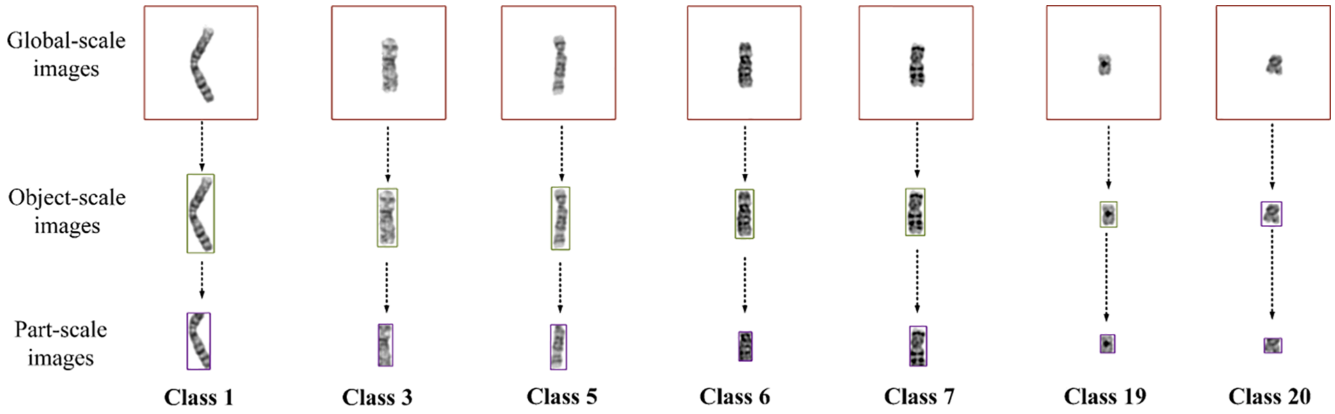


Fig. 8. Example of correctly classified samples. Global-scale, object-scale, and part-scale images are displayed to visually assess the IAPP-CNN.

Table 4
Evaluation of the input-aware module and integrated uncertainty estimates module for average F1-score and accuracy.

parameters	F1-score	Accuracy
Global scale	97.3 ± 0.2	97.5 ± 0.3
Global scale (+IUE module)	97.3 ± 0.2	97.5 ± 0.3
Global scale + Object scale	98.2 ± 0.1	98.3 ± 0.2
Global scale + Object scale + IUE module	98.8 ± 0.1	98.9 ± 0.1
Global scale + Part scale	98.1 ± 0.2	98.3 ± 0.2
Global scale + Part scale + IUE module	98.7 ± 0.2	98.8 ± 0.1
Global scale + Object scale + Part scale	98.4 ± 0.2	98.5 ± 0.1
Global scale + Object scale + Part scale + IUE module	**99.1 ± 0.1**	**99.2 ± 0.2**

Note: We reported the average performance for 5 random initializations. Here, IUE represents integrated uncertainty estimates module.

To observe the classification ability of the model intuitively, we illustrate typical examples of correctly classified chromosomes, as shown in Fig. 8. It can be seen that the IAPP-CNN can precisely zoom from the global-scale image into an object-scale image and capture the discriminable part-scale image automatically.

As shown in the odd rows of Table 4, an F1-score of 97.3% and an accuracy of 97.5% can be achieved, by using global-scale branch only. When the object-scale branch or part-scale branch was introduced, accuracy and F1-score increased by about 0.8%. In addition, the results show that the model showed the same performance when introducing the object-scale or part-scale branch. The combination of all three scales produced the optimal result. Compared with the works which only used the global-scale and part-scale, our designed input-aware module displayed a better performance.

4.5.2. *Availability of integrated uncertainty estimates module*

The integrated uncertainty estimates module employed a dynamic probabilistic parameter λ to assign a confidence score to the prediction result, making the chromosome type prediction more robust. In the above part, the performance of the input-aware module was analyzed. To assess the performance of the integrated uncertainty estimates module, the input-aware module was fixed to make a comparative experiment about probabilistic parameter λ. In Table 4, the even rows provide the results of comparative experiments. According to Formula (15), when the global-scale branch is only used, the use of probabilistic parameter λ presented no effect on the prediction results. When using two scales, whether it was the combination of global-scale and object-scale or part-scale, the accuracy improved by 0.5% and the F1-score improved by 0.6% with the addition of the probabilistic parameter. When three scales are introduced, using the probabilistic parameter made the prediction accuracy reach 99.2% and the F1-score reach 99.1%. Experiments proved that the introduction of the integrated uncertainty estimates module allowed the model to reach optimal performance.

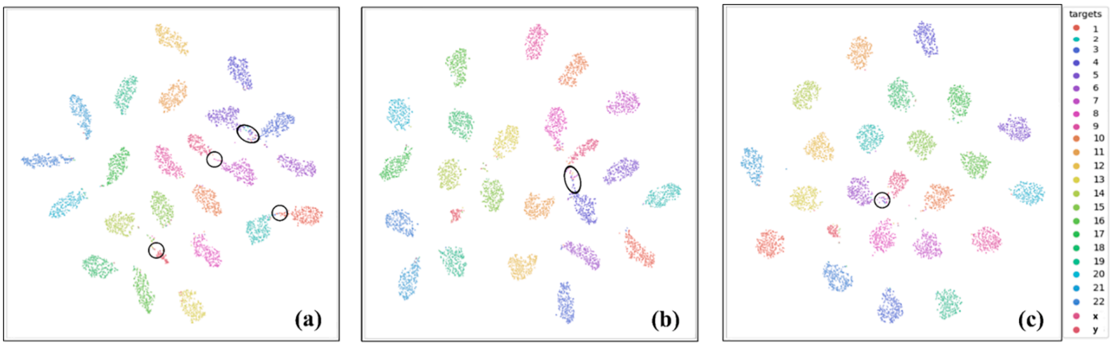


Fig. 9. Visualization of feature embeddings for chromosomes using t-SNE method [46] in the test dataset((a), (b), (c) represent the fine-tuning resnet50, resnet50 with the input-aware module, and IAPP-CNN, respectively).

4.5.3. *Visualization*

In the above part, the proposed IAPP-CNN was quantitatively analyzed. The feature embedding representations learned are visualized in Fig. 9. By using the t-SNE [46] method, the dimension of the feature was reduced to two. Fig. 9(a), (b), and (c) are the t-SNE visualizations of the resnet50, resnet50 with the input-aware module and IAPP-CNN, respectively. It can be seen that compared with fine-tuning ResNet50, IAPP-CNN makes the samples of the same class gather closely and increases the between-class distance.

5. Conclusions

To further improve the performance of the chromosome classification task, a comprehensive study on automatic chromosome classification was performed in this study. First, a novel automated method to create a Giemsa staining chromosome classification dataset that effectively reduces manual participation was proposed. Then, based on this dataset, the current state-of-the-art methods were analyzed and an efficient IAPP-CNN was proposed, using a different approach for the input image and output prediction of the CNNs. The proposed model included an input-aware module, which automatically extracted multi-scale images from coarse to fine, and an integrated uncertainty estimates module, which employed a dynamic probabilistic parameter to evaluate the reliability of prediction. The two designed modules designed are modular and can be applied to other medical image recognition scenarios and other tasks. Extensive experiments on the chromosome classification dataset demonstrated that our proposed IAPP-CNN surpassed the performance of a competitive baseline created by state-of-the-art methods.

In future work, we will attempt to exploit ways to reduce the model parameters, while ensuring the accuracy of the classification.

CRediT authorship contribution statement

Hua Wei: Conceptualization, Methodology, Validation, Writing – original draft. **Wen Gao:** Data curation, Writing – review & editing. **Haitao Nie:** Methodology, Visualization. **Jiaqi Sun:** Investigation, Validation. **Ming Zhu:** Conceptualization, Methodology, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R.J. Wapner, C.L. Martin, B. Levy, et al., Chromosomal microarray versus karyotyping for prenatal diagnosis, Obstetrical & Gynecological Survey. 68 (4) (2013) 276–278, https://doi.org/10.1097/01.OGX.0000429294.57890.B8.

[2] L. Dugoff, M.E. Norton, J.A. Kuller, The use of chromosomal microarray for prenatal diagnosis, Am. J. Obstet. Gynecol. 215 (4) (2016) B2–B9, https://doi.org/10.1016/j.ajog.2016.07.016.

[3] J. Zhang, W. Hu, S. Li, Y. Wen, Y. Bao, H. Huang, C. Xu, D. Qian, Chromosome classification and straightening based on an interleaved and multi-task network, IEEE Journal of Biomedical and Health Informatics. 25 (8) (2021) 3240–3251, https://doi.org/10.1109/JBHI.2021.3062234.

[4] W. Ding, L. Chang, C. Gu, et al., Classification of chromosome karyotype based on faster-RCNN with the segmatation and enhancement preprocessing model, in: 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2019, pp. 1–5, https://doi.org/10.1109/CISP-BMEI48845.2019.8965713.

[5] N. Xie, X.u. Li, K. Li, Y. Yang, H.T. Shen, Statistical karyotype analysis using CNN and geometric optimization, IEEE Access 7 (2019) 179445–179453, https://doi.org/10.1109/Access.628763910.1109/ACCESS.2019.2951723.

[6] E. Schrock, S. du Manoir, T. Veldman, B. Schoell, J. Wienberg, M.A. Ferguson-Smith, Y. Ning, D.H. Ledbetter, I. Bar-Am, D. Soenksen, Y. Garini, T. Ried, Multicolor spectral karyotyping of human chromosomes, Science 273 (5274) (1996) 494–497, https://doi.org/10.1126/science:273.5274.494.

[7] M.R. Speicher, S.G.G. Ballard, D.C. Ward, Karyotyping human chromosomes by combinatorial multi-fluor FISH, Nat. Genet. 12 (4) (1996) 368–375, https://doi.org/10.1038/ng0496-368.

[8] S. Wolff, P. Perry, Differential Giemsa staining of sister chromatids and the study of sister chromatid exchanges without autoradiography, Chromosoma 48 (4) (1974) 341–353, https://doi.org/10.1007/BF00290991.

[9] P. Perry, S. Wolff, New Giemsa method for the differential staining of sister chromatids, Nature 251 (5471) (1974) 156–158, https://doi.org/10.1038/251156a0.

[10] J. Piper, Automated cytogenetics in the study of mutagenesis and cancer, Advances in Mutagenesis Research, Springer, Berlin Heidelberg. 2 (1990) 127–153, https://doi.org/10.1007/978-3-642-75599-6_4.

[11] B. Lerner, H. Guterman, I. Dinstein, Y. Romem, Medial axis transform-based features and a neural network for human chromosome classification, Pattern Recogn. 28 (11) (1995) 1673–1683, https://doi.org/10.1016/0031-3203(95)00042-X.

[12] N. Madian, K.B. Jayanthi, Analysis of human chromosome classification using centromere position, Measurement 47 (2014) 287–295, https://doi.org/10.1016/j.measurement.2013.08.033.

[13] S. Jahani, S.K. Setarehdan, A novel method for centromere and length detection in microscopic images of human chromosomes, in: In 2011 18th Iranian Conference of Biomedical Engineering (ICBME), 2011, pp. 274–277, https://doi.org/10.1109/ICBME.2011.6168571.

[14] S. Jahani, S.K. Setarehdan, An automatic algorithm for identification and straightening images of curved human chromosomes, Biomedical Engineering: Applications, Basis and Communications. 24 (6) (2012) 503–511, https://doi.org/10.4015/S1016237212500469.

[15] M. Javan-Roshtkhari, S.K. Setarehdan, A new approach to automatic classification of the curved chromosomes, in: In 2007 5th International Symposium on Image and Signal Processing and Analysis, 2007, pp. 19–24, https://doi.org/10.1109/ISPA.2007.4383657.

[16] A.M. Jennings, J. Graham, A neural network approach to automatic chromosome classification, Phys. Med. Biol. 38 (7) (1993) 959–970, https://doi.org/10.1088/0031-9155/38/7/006.

[17] W. Gómez-Flores, W. Coelho de Albuquerque Pereira, A comparative study of pre-trained convolutional neural networks for semantic segmentation of breast tumors in ultrasound, Comput. Biol. Med. 126 (2020) 104036, https://doi.org/10.1016/j.compbiomed.2020.104036.

[18] D. Wang, M. Li, N. Ben-Shlomo, C.E. Corrales, Y.u. Cheng, T. Zhang, J. Jayender, A novel dual-network architecture for mixed-supervised medical image

segmentation, Computerized Medical Imaging and Graphics. 89 (2021) 101841, https://doi.org/10.1016/j.compmedimag.2020.101841.

[19] J. Zhang, Y. Xie, Y. Wang, Y. Xia, Inter-slice context residual learning for 3D medical image segmentation, IEEE Trans. Med. Imaging 40 (2) (2021) 661–672, https://doi.org/10.1109/TMI.4210.1109/TMI.2020.3034995.

[20] P. Shan, Y. Wang, C. Fu, W. Song, J. Chen, Automatic skin lesion segmentation based on FC-DPN, Comput. Biol. Med. 123 (2020) 103762, https://doi.org/10.1016/j.compbiomed.2020.103762.

[21] G. S., V.P. Gopi, P. Palanisamy, A lightweight CNN for diabetic retinopathy classification from fundus images, Biomed. Signal Process. Control 62 (2020) 102115, https://doi.org/10.1016/j.bspc.2020.102115.

[22] P. Wang, J. Wang, Y. Li, P. Li, L. Li, M. Jiang, Automatic classification of breast cancer histopathological images based on deep feature fusion and enhanced routing, Biomed. Signal Process. Control 65 (2021) 102341, https://doi.org/10.1016/j.bspc.2020.102341.

[23] H. Jung, B. Park, S. Lee, et al., Deep metric learning network using proxies for chromosome classification and retrieval in karyotyping test, BioRxiv. (2020), https://doi.org/10.1101/2020.05.24.113936.

[24] W. Zhang, S. Song, T. Bai, et al., Chromosome classification with convolutional neural network based deep learning, in: In 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2018, pp. 1–5, https://doi.org/10.1109/CISP-BMEI.2018.8633228.

[25] P. Kiruthika, K.B. Jayanthi, M. Nirmala, Classification of metaphase chromosomes using deep learning neural network, in: In 2018 4th International Conference on Frontiers of Signal Processing (ICFSP), 2018, pp. 110–114, https://doi.org/10.1109/ICFSP.2018.8552042.

[26] T. Arora, R. Dhir, Geometric feature-based classification of segmented human chromosomes, International Journal of Image and Graphics. 20 (1) (2020) 2050006, https://doi.org/10.1142/S0219467820500060.

[27] X.i. Hu, W. Yi, L. Jiang, S. Wu, Y. Zhang, J. Du, T. Ma, T. Wang, X. Wu, Classification of metaphase chromosomes using deep convolutional neural network, J. Comput. Biol. 26 (5) (2019) 473–484, https://doi.org/10.1089/cmb.2018.0212.

[28] Y. Wu, Y. Yue, X. Tan, et al., End-to-end chromosome karyotyping with data augmentation using GAN, in: In 2018 25th IEEE International Conference on Image Processing (ICIP), 2018, pp. 2456–2460, https://doi.org/10.1109/ICIP.2018.8451041.

[29] M. Sharma, O. Saha, A. Sriraman, et al., Crowdsourcing for chromosome segmentation and deep classification, in: In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 786–793, https://doi.org/10.1109/CVPRW.2017.109.

[30] G.G. Swati, M. Yadav, M. Sharma, et al., Siamese networks for chromosome classification, in: In 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), 2017, pp. 72–81, https://doi.org/10.1109/ICCVW.2017.17.

[31] Y. Qin, J. Wen, H. Zheng, X. Huang, J. Yang, N. Song, Y.-M. Zhu, L. Wu, G.-Z. Yang, Varifocal-Net: a chromosome classification approach using deep convolutional networks, IEEE Transactions on Medical Imaging. 38 (11) (2019) 2569–2581, https://doi.org/10.1109/TMI.4210.1109/TMI.2019.2905841.

[32] C. Lin G. Zhao Z. Yang A. Yin X. Wang L.i. Guo H. Chen Z. Ma L. Zhao H. Luo T. Wang B. Ding X. Pang Q. Chen CIR-Net: automatic classification of human chromosome based on Inception-ResNet architecture 1 1 10.1109/TCBB.8857 10.1109/TCBB.2020.3003445.

[33] I. Goodfellow, J. Pouget-Abadie, M. Mirza, et al., Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144. https://dl.acm.org/doi/10.1145/3422622.

[34] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90, https://doi.org/10.1145/3065386.

[35] C. Szegedy, S. Ioffe, V. Vanhoucke, et al., Inception-v4, Inception-ResNet and the impact of residual connections on learning, in: In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2016, pp. 4278–4284.

[36] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. In ICLR 2015: International Conference on Learning Representations, 2015.

[37] G. Huang, Z. Liu, L.V.D. Maaten, et al., Densely connected convolutional networks, in: In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261–2269, https://doi.org/10.1109/CVPR.2017.243.

[38] K. He, X. Zhang, S. Ren, et al., Deep residual learning for image recognition, in: In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[39] C. Wah, S. Branson, P. Welinder, et al., The Caltech-UCSD birds-200-2011 dataset, Computation & Neural Systems Technical Report, California Institute of Technology. https://resolver.caltech.edu/CaltechAUTHORS:20111026-120541847.

[40] X.-S. Wei, J.-H. Luo, J. Wu, Z.-H. Zhou, Selective convolutional descriptor aggregation for fine-grained image retrieval, IEEE Trans. Image Process. 26 (6) (2017) 2868–2881, https://doi.org/10.1109/TIP.2017.2688133.

[41] N. Zhang, J. Donahue, R.B. Girshick, et al., Part-based R-CNNs for fine-grained category detection, European Conference on Computer Vision. (2014) 834–849, https://doi.org/10.1007/978-3-319-10590-1_54.

[42] Z. Yang, T. Luo, D. Wang, et al., Learning to navigate for fine-grained classification, in: In Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 438–454, https://doi.org/10.1007/978-3-030-01264-9_26.

[43] J. Fu, H. Zheng, T. Mei, Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition, in: In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 4476–4484, https://doi.org/10.1109/CVPR.2017.476.

[44] J. Hu, L. Shen, S. Albanie, et al., Squeeze-and-excitation networks, In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 42 (2018) 2011–2023, https://doi.org/10.1109/CVPR.2018.00745.

[45] S. Maji, E. Rahtu, J. Kannala, et al., Fine-grained visual classification of aircraft. https://arxiv.org/abs/1306.5151.

[46] L.V.D. Maaten, G. Hinton, Visualizing data using t-SNE, Journal of Machine Learning Research. 9 (86) (2008) 2579–2605.