# Label-free detection and enumeration of rare circulating tumor cells by bright-field image cytometry and multi-frame image correlation analysis

Ziqiang Du[a#], Ya Li[b#], Bing Chen[b#], Lulu Wang[b], Yu Hu[a], Xu Wang[a], Wenchang Zhang[c*], Xiaonan Yang[a,c*]

[a] School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China

[b]Department of Gastroenterology, The First Affiliated Hospital of Zhengzhou University, Zhengzhou, 450052, China

[c] Key Lab of Microelectronic Devices & Integrated Technology, Institute of Microelectronics, Chinese Academy of Sciences, Beijing, 100029, China

# These authors contributed equally.

Corresponding: zhangwenchang@ime.ac.cn, iexnyang@zzu.edu.cn

## Contents

## Supplementary text

### 1 Data set

The data sets were built by imaging pure cancer cell lines (HT29, A549, and KYSE30), WBCs, and RBCs samples by our system for confirmed ground truth of each image. About 12500 cancer cell lines images, 10500 WBCs images, and 10500 RBCs images were collected and divided as the training set, validation set, and test set. The total set contained 24850 training images (8284 of cancer cell lines, 8426 of WBCs and 8140 of RBCs), 3876 validation images (1245 of cancer cell lines, 1297 of WBCs and 1334 of RBCs) and 5000 test images (1000 of HT29, 1000 of A549, 1000 of KYSE30, 1000 of WBCs and 1000 of RBCs, the ground truth of these types of cells in the test set images were 3591, 3975, 3811, 4356, and 4871, respectively. ). No overlap existed for the three sets to ensure the independence of each set. Before training, cells in the data set were alternatively marked as "CTC", "WBC" or "RBC" according to the ground truth with the aid of a labeling tool called "Colabeler". The image size in the data set was unified to be 416×416. To obtain a better recognition effect, all the raw images were pre-processed by image processing programs, including gray mapping and image sharpening, etc. Since the number of images was large enough to contain a rich variation of training samples, the common data augmentation step was not performed.

## 2 YOLO-V4 and t-SNE algorithm

In this paper, two algorithms, t-SNE and YOLO-V4, were used to make a preliminary evaluation of the separability of the data and the actual decision of the cell types, respectively. The YOLO object detection model was proposed in CVPR2016 and has been developed rapidly in recent years for its highly efficient object detection ability. The network structure and the training environment configuration of the model were shown in Figure S1.

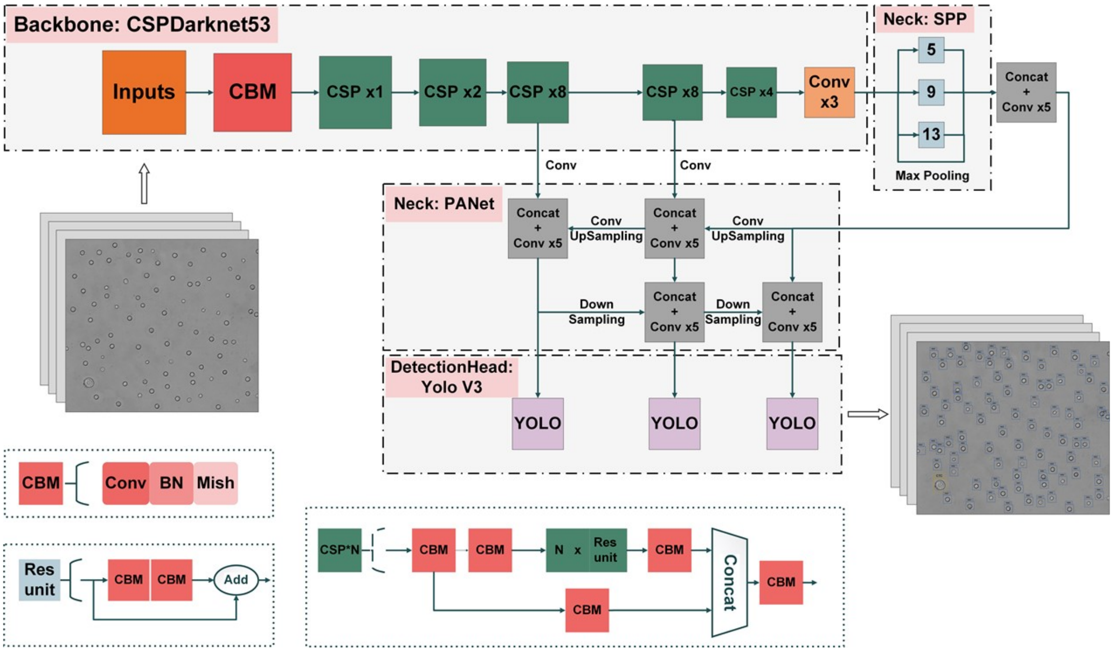

**Figure S1**. The network structure of YOLO-V4

YOLO-V4 network utilized CSPDarknet53, an open-source neural network framework, as the main backbone network to train and extract image features; then SPPNet(Spatial Pyramid Pooling Network) and PANet (Path Aggregation Network) were employed as the neck network to achieve a better fusion of the extracted features, and the head exploited YOLO-V3 to realize classification and positioning of objects. In the CSPDarknet53 module, the feature mapping of the basic layer was divided into two parts and merged again after the cross-stage level, which effectively reduced the amount of calculation. SPP structure was mixed in the convolution of the last feature layer of the backbone network, which improved the detection accuracy of small objects by integrating the high- and low-layer features. Considering the importance of the network's shallow feature information, the PANet introduced the structure of bottom-up path augmentation so that the network could retain more shallow features. In the feature utilization part, YOLO-V3 detection head extracted three feature layers for object detection, and then the final prediction results were given after the score ranking and non-maximum suppression (NMS). Besides, YOLO-V4 introduced a mosaic data augmentation method. By simulating the occlusion of the object, the network was forced to recognize the local unoccluded data. This enhanced the generalization ability of the model. Based on the characteristics of this model, the YOLO-V4 was expected to perform outstandingly to discriminate CTCs from blood cells with high accuracy.

In addition, the t-SNE algorithm was also used in this article for a preliminary assessment of the separability of the data. The t-SNE was a machine learning algorithm for dimensionality reduction, proposed by Laurens van der Maaten and Geoffrey Hinton in 2008. It can reduce the high-dimensional data to 2 or 3 dimensions, to make preliminary judgments on the separability of data in a visual way. In this paper, the images used for data separability verification came from a portion of the training set, totally containing 2400 single-cell images with 800 images from each cell type separately. The size of the images was uniformed to be 80 × 80. The Scikit-Learn machine learning package was used, with the perplexity of 50 and the learning rate of 100. The image data was stored as NumPy and the features were eventually mapped to 3D space.

## 3 Model's training process

For the model training of YOLO-V4, the environment configuration is shown in Table S1, and the training parameters are as follows: the momentum was set to 0.949, the initial learning rate was 0.001, and the weight decay was 0.0005. The batch size was set to 96, and the subdivisions was 48. To better analyze the training process, 60,000 iterations were performed. During the training process, the neural network was fine-tuned step by step to approach the ground truth, followed by the evaluation of the trained model on the test set. The weight file was saved every 1000 iterations, and the mean AP (mAP) of the validation set was calculated every 2 epochs. The confidence-thresh of the validation set was 0.25 and the intersection over union (IOU) thresh was 0.5. It was found that the learning rate dropped to 0.0001 after 48,000 steps and to 0.00001 after 54,000 steps. Then the accuracy of the validation set was significantly saturated and the training process was completed.

**Table S1.** Training environment

| Configuration | Parameter |
|---|---|
| GPU | GeForce RTX3090 |
| CPU | Intel Xeon 8124M |
| Operating system | Ubuntu 20.04 |
| Accelerated environment | CUDA1.5.1 cuDNN8.3.0 |
| Programming language&Library | Python 3.6.8 & Open CV 3.4.5 |

## 4 AP values under multi-frame image correlation

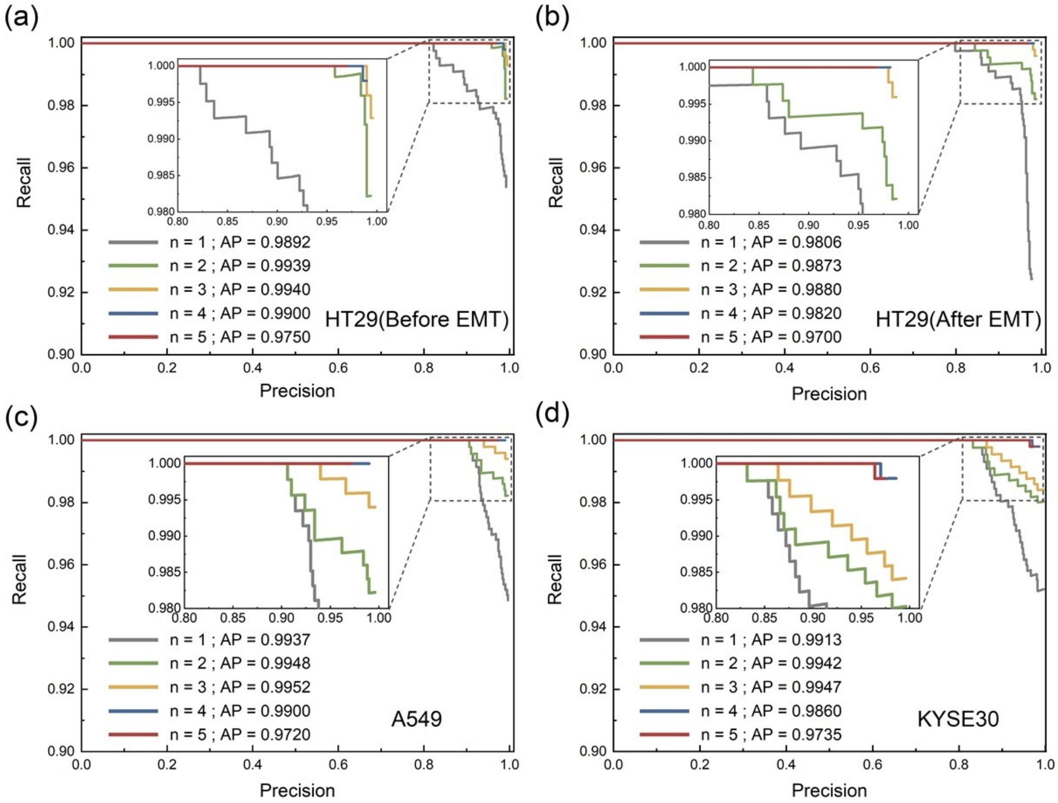

**Fig. S2** AP values at different n values under multi-frame image correlation. AP values for (a) HT29 (Before EMT), (b) HT29 (After EMT), (c) A549, (d) KYSE30.

In the data association of multiple frames, the number of times the tumor cells correctly identified was recorded as n (1≤n≤5). Then, different n as the threshold to make a secondary classification decision on the tracked tumor cells as the multi-frame detection result. PR curves for HT29 (Before EMT or After EMT), A549, KYSE30 from n=1 to n=5 are shown in Figure S2 (a~d), respectively.

## 5 The pathological diagnosis results of the patients.

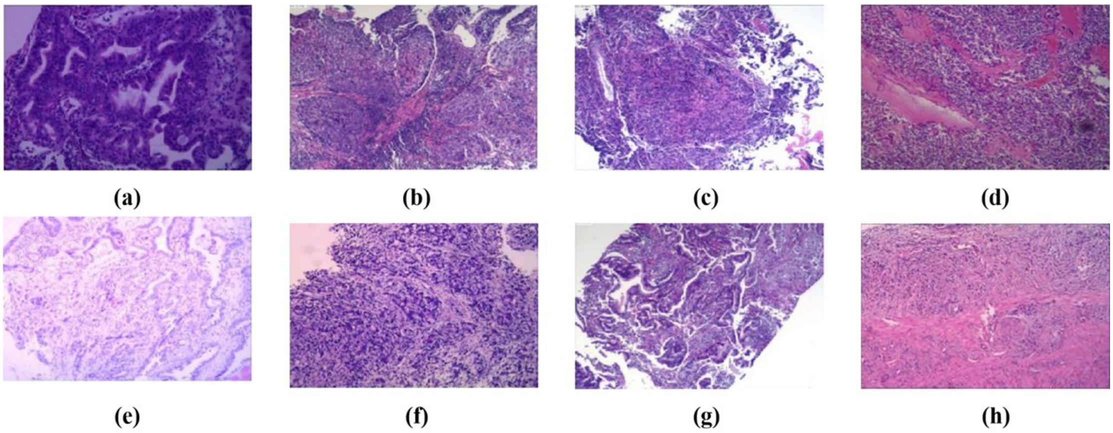

**Fig. S3** The pathological diagnosis results of the 8 patients. (a) (b) colorectal adenocarcinoma, (c) (d) lung cancer, (e)~(h) esophageal carcinoma.

CTCs were successfully detected from patient peripheral blood using our system. To confirm the patient status, the pathological diagnosis results of the 8 patients was exhibited in Figure S3. It included 2 of colorectal adenocarcinoma (Figure S3(a) (b)), 2 of lung cancer (Figure S3(c) (d)), and 4 of esophageal carcinoma (Figure S3(e)~ (h)), with 6 healthy control subjects.