



An End-to-End System for Automatic Urinary Particle Recognition with Convolutional Neural Network

Yixiong Liang¹ · Rui Kang¹ · Chunyan Lian¹ · Yuan Mao¹

Received: 2 July 2017 / Accepted: 11 July 2018 / Published online: 27 July 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

The urine sediment analysis of particles in microscopic images can assist physicians in evaluating patients with renal and urinary tract diseases. Manual urine sediment examination is labor-intensive, subjective and time-consuming, and the traditional automatic algorithms often extract the hand-crafted features for recognition. Instead of using the hand-crafted features, in this paper we propose to exploit convolutional neural network (CNN) to learn features in an end-to-end manner to recognize the urinary particle. We treat the urinary particle recognition as object detection and exploit two state-of-the-art CNN-based object detection methods, Faster R-CNN and single shot multibox detector (SSD), along with their variants for urinary particle recognition. We further investigate different factors involving these CNN-based methods to improve the performance of urinary particle recognition. We comprehensively evaluate these methods on a dataset consisting of 5,376 annotated images corresponding to 7 categories of urinary particle, i.e., erythrocyte, leukocyte, epithelial cell, crystal, cast, mycete, epithelial nuclei, and obtain a best mean average precision (mAP) of 84.1% while taking only 72 ms per image on a NVIDIA Titan X GPU.

Keywords Urinary particle recognition · CNN · Faster R-CNN · SSD

Introduction

The urine sediment examination of biological particles in microscopic images is one of the most commonly performed *vitro* diagnostic screening tests in clinical laboratories and it plays an important role in evaluating the kidney and genitourinary system and monitoring body state. General indications for urinalysis include: the possibility of urinary tract infection or urinary stone formation; non-infectious renal or post-renal diseases; in pregnant women and patients with diabetes mellitus or metabolic states who may have proteinuria, glycosuria, ketosis or acidosis/alkalosis [15, 18].

Traditionally, the trained technicians count the number of each kind of particles of urinary sediment by visual

inspection. The manual urine sediment examination works but is labor-intensive, time-consuming, subjective, and operator-dependent in high-volume laboratories. All these issues have motivated lots of automated methods for the analysis of urine microscope images (e.g. [1, 2, 19, 21, 25, 33]). As shown in Fig. 1a, almost all of them follow the multi-stage pipeline, i.e., first generating candidate regions based on segmentation and then extracting hand-crafted features over regions for classification. Therefore, the performance of these methods heavily depends on the accuracy of the segmentation and the effectiveness of the hand-crafted features. However, due to the complicated characteristics of urinary images, the precise segmentation of the interested particles is quite difficult, or even impossible, and the resulting hand-crafted region features are often less discriminatory.

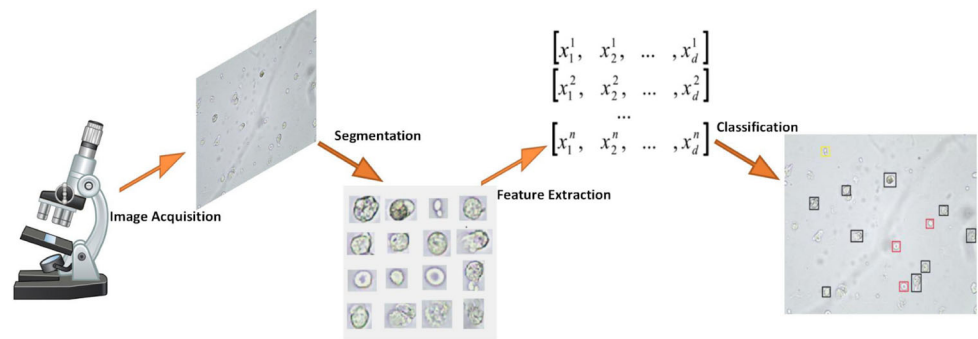
To avoid the segmentation stage and improve the discriminability of features, as shown in Fig. 1b, we propose to exploit CNN to automatically learn task-specific features and perform the urinary particle recognition in an end-to-end manner. Specifically, we treat the urinary particle recognition as object detection and exploit two well-known CNN-based object detection methods, Faster

This article is part of the Topical Collection on *Image & Signal Processing*

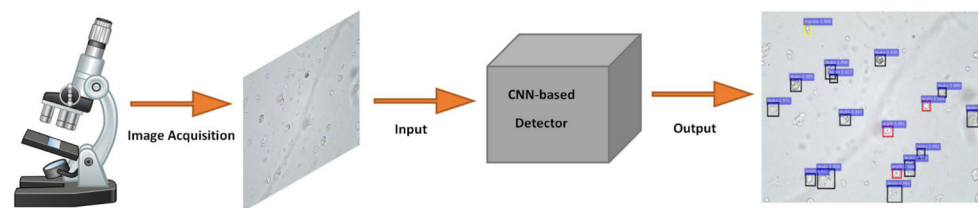
✉ Yixiong Liang
yxtliang@csu.edu.cn

¹ School of Information Science and Engineering,
Central South University, Changsha 410083, China

Fig. 1 The pipelines for urinary particles recognition



(a) The traditional multi-stage pipeline



(b) The proposed end-to-end pipeline

R-CNN [29] and SSD [23], along with their variants including Multiple Scale Faster R-CNN (MS-FRCNN) [13], Faster R-CNN with online hard example mining [34] (OHEM-FRCNN) and the proposed Trimmed SSD to locate and recognise the urinary particles. These end-to-end methods do not perform explicitly segmentation and hand-crafted features extraction, but can automatically learn more discriminative features from the annotated images. We also investigate different factors such as training strategies, network structures, fine-tuning tricks, data augmentation etc, to make these methods more appropriate for urinary particle recognition.

We summarize our contributions as follows:

- We exploit Faster R-CNN [29] and SSD [23] for urine particle recognition. It is segmentation free and can learn task-specific features in an end-to-end manner.
- We investigate various factors to improve the performance of Faster R-CNN [29] and its variants [13, 34] for urine particle recognition.
- We propose a scheme, Trimmed SSD, to prune the network structure adopted in SSD [23] to achieve better performance for urine particle recognition.
- We obtain a best mAP of 84.1% while taking only 72 ms per image for 7 categories recognition of urine sediment particles. Importantly, we also get a best AP of 77.2% for cast particles, the most valuable but most difficult to detect ingredients [4, 6, 40].

The remainder of this paper is organized as follows: Section “[Related work](#)” reviews the related works of urine

particle recognition and the CNN-based object detection methods. Section “[Meta-architectures](#)” describes the detection architectures for urinary particle recognition, including Faster R-CNN, SSD and their variants. Section “[Experiments](#)” details the urinalysis database organization and provides extensive experiments analysis. Section “[Adding bells & whistles](#)” shows more experimental comparisons intuitively. Section “[Conclusions](#)” concludes the paper.

Related work

Urine particle recognition

The recognition of urinary sediment particles has been extensively studied following the traditional multi-stage pipeline (Fig 1a) and a variety of approaches can be adopted in each stage.

Rabznto et al.[25] first obtained patches of interest by a detection algorithm, and then extracted invariant features based on “local jets” [31]. Although the system presents reliable recognition results on a pollen dataset, more accurate location for interest patches needed to be improved. In [2], a new technique based on the adaptive discrete wavelet entropy energy was proposed for feature extraction, which follows by some image preprocessing stage including noise reduction, contrast enhancement and segmentation. In classification, the artificial neural network (ANN) classifier was selected to achieve the best performance. Liang et al. [21] adopted a two-step process (the first location step

and the second tuning step) to segment particles' contour. They proposed a two-tier classification strategy to better reduce the false positive rate caused by impurity and poor focused regions. Shen et al. [33] used AdaBoost to select a little part typical Haar features for support vector machine (SVM) classification, and improved system speed via cascade accelerating algorithm. Zhou et al. [40] demonstrated an easy-implemented automatic urinalysis system employing a SVM classifier to distinguish casts from other particles. After the adhesive particles separation by watershed algorithm, Li et al. [19] proposed to combine the Gabor filter with the scattering transform for robust feature description.

The above-mentioned conventional recognition model works for automated urinalysis, but importantly all stages (i.e. segmentation, feature extraction and classification) need to be carefully designed. In addition, the complicated characteristics of urine microscopical images also bring more challenges to this task. Therefore, there is an increasing demand for better solutions relying more on automatic learning and less on hand-designed heuristics.

CNN-based object detection

The Overfeat [32] made the earliest efforts to apply deep CNNs to learn highly discriminative yet invariant feature for object detection and has achieved a significant improvement of more than 50% mAP when compared to the best methods at that time which were based on the hand-crafted features. Since then, a lot of advanced CNN-based methods (e.g. [7, 8, 23, 26–29]) have been proposed for high-quality object detection, which can be roughly classified into two categories: object proposal-based and regression-based.

The object proposal-based method first generates a series of proposals by applying region proposal methods and then classifies each proposal as background or category-specific objects. The notable R-CNN [8] generates about 2,000 region proposals by selective search [36] and repeatedly resize each proposal box to a fixed size to extract CNN features for SVM classification. The SPP-net [11] introduces a spatial pyramid pooling layer that can flexibly handle variable-size inputs, which avoids repeatedly computing the convolutional features (compute only once per image) and therefore accelerates R-CNN significantly. Instead of a spatial pyramid pooling layer, the Fast R-CNN [7] extends SPP-net by introducing a ROI pooling layer and a joint classification loss and bounding box regression loss. It can fine-tune all layers in an end-to-end manner, which significantly speeds up the stages of training and testing.

The handcrafted region proposal methods such as selective search [36] or Edgeboxes [41] is often time-consuming which immediately becomes the bottleneck of object detection systems. The Faster R-CNN [29] proposes

a region proposal network (RPN) for generating region proposals and combines RPN and Fast R-CNN into a single network by sharing their full-image convolutional features, thus it enables nearly cost-free region proposal generation. Faster R-CNN is flexible and robust to many follow-up improvements (e.g., [13, 14, 16, 20, 22, 38, 39]), and has been achieving top performances in several benchmarks [10].

Regression-based method reformulates the object detection as a regression problem with separated bounding boxes and class-specific probabilities and detects objects by regular and dense sampling over locations, scales and aspect ratios [9, 23, 26–28]. It does not require proposal generation stage and therefore is much simpler than proposal-based methods. SSD [23] and YOLO [26–28] are two representative regression-based methods: YOLO [26] opens the door to achieve real-time CNN-based object detection and SSD [23] is proposed for improving YOLO's performance of small-sized objects detection and localization accuracy. Generally, regression-based methods are much faster than proposal-based methods but the detection accuracy is usually behind that of the proposal-based methods [39].

Meta-architectures

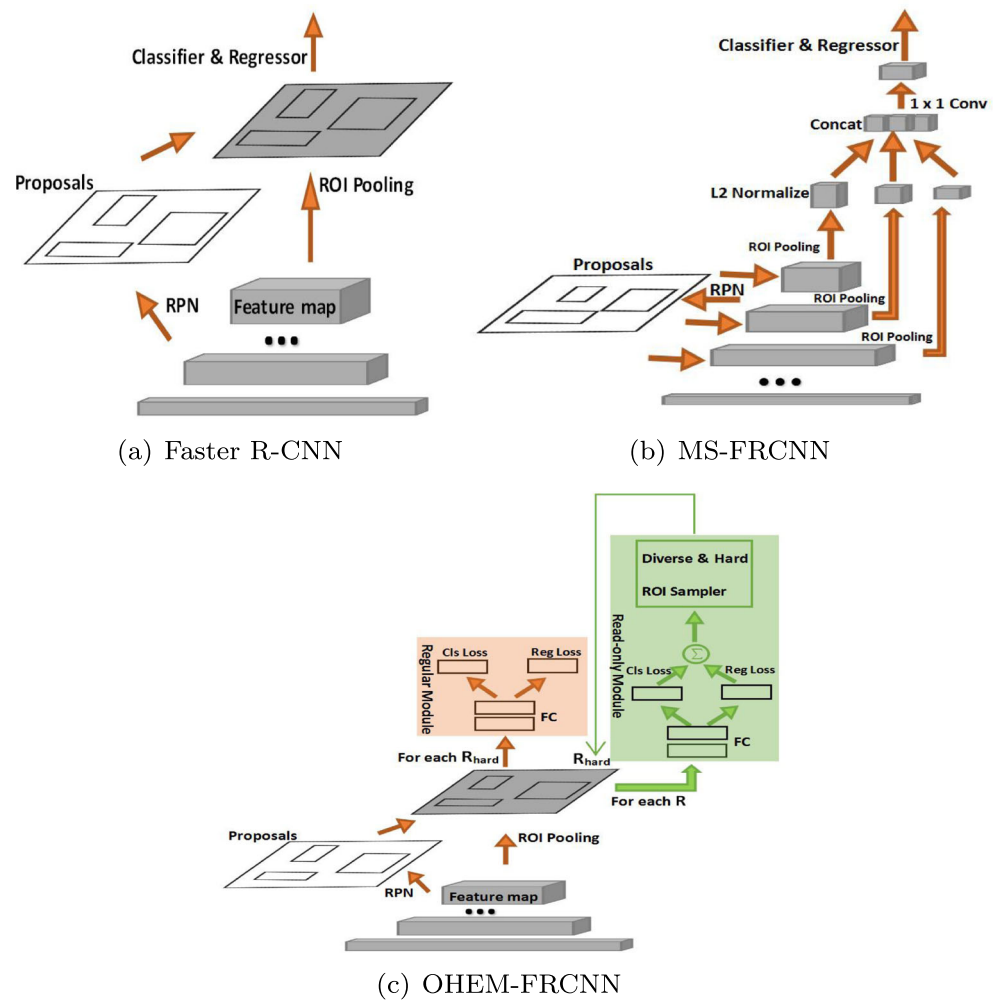
In this paper, we focus primarily on Faster R-CNN [29] and its structural variants, i.e. multiple scale Faster R-CNN (MS-FRCNN) [13], Faster R-CNN with online hard example mining (OHEM-FRCNN) [34] for urinary particle recognition. We also investigate the performance of SSD [23] on urinary particle recognition and propose a named Trimmed SSD.

Faster R-CNN and its variants

Faster R-CNN [29] is a single unified network which integrates a fully convolutional region proposal generator (RPN) with a fast region-based object detector (Fast R-CNN) [7]. As shown in Fig. 2a, the deep detection framework also can be described as the pipeline of "shareable CNN feature extraction + region proposal generation + region classification and regression". Moreover, to predict objects across multiple scales and aspect ratios, Faster R-CNN [29] adopts a pyramid of anchors with different aspect ratios, which is a key component for sharing features without extra cost.

MS-FRCNN [13] is a follow-up improvement and it keeps RPN unchanged and builds a more sophisticated network for Fast R-CNN detector by a combination of both global context and local appearance features. As Fig. 2b shows, each object proposal receives three feature tensors through

Fig. 2 The architectures of Faster R-CNN, MS-FRCNN and OHEM-FRCNN



ROI pooling from the last three convolutional layers. After L2 normalization to each tensor, outputs are concatenated and compressed to maintain the same size as the original architecture.

OHEM-FRCNN is a combination of online hard example mining (OHEM) [34] and Faster R-CNN [29]. OHEM [34] is a novel bootstrapping for modern CNN-based object detectors trained purely online with SGD. Instead of a sampled mini-batch [29], it eliminates several heuristics and hyperparameters in common use and selects automatically hard examples by loss. As Fig. 2c shows, for each iteration, given the feature map from shareable convolutional network and ROIs from RPN, the read-only ROI network performs a forward pass and computes loss for all input ROIs. Then the regular ROI network computes forward and backward passes only for hard examples selected by hard ROI sampling module according to a distribution that favors diverse, high loss candidates.

SSD and the Proposed Trimmed SSD

SSD [23] can be decomposed into a truncated base network (usually a VGG-16 net [35]) and several auxiliary convolutional layers used as feature maps and predictors. Unlike Faster R-CNN [29], SSD increases detection speed by removing the region proposal generation and the subsequent pixel or feature resampling stages. Unlike YOLO [26], it improves detection quality by applying a set of small convolutional filters to multiple feature maps to predict confidences and boxes offsets for various-size categories, as shown in Fig. 3.

Trimmed SSD is the proposed method which a tailored version of the original SSD model [23] for urinary particle recognition. As Fig. 3 shows, from bottom to top, original SSD selects conv4_3, fc7 (convolutional layer), conv6_2, conv7_2, conv8_2, conv9_2 and pool6 as feature maps to produce confidences and locations. If we directly transfer it to urinary particle recognition with only 7 categories,

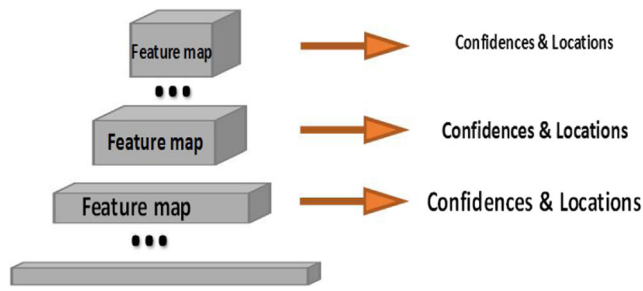


Fig. 3 The architecture of SSD

it may produce a large number of redundant prediction results interfering with the final detection performance. And the framework is too complicated to perfectly fit our dataset. For simplification, we attempt to remove several top convolutional layers from the auxiliary network of SSD, which leads to the trimmed SSD.

Experiments

As there is no standard benchmarks available, we first establish the database consisting of 6,804 urinary microscopical images with ground truth boxes marked by clinical experts. All annotated images have a size of 800 × 600, containing objects from 7 categories of urinary sediment particles, i.e.,

erythrocyte (eryth), leukocyte (leuko), epithelial cell (epith), crystal (cryst), cast, mycete, epithelial nuclei (epithn) and one background class. Figure 4 shows 7 categories of urinary sediment particles from our database, each of which includes many subcategories with various shapes.

In fact, our 6,804 annotated images have a total of 273,718 ground truths, where meaningless background occupies 230,919 annotations, up to eight-four percent. We remove images only including noise and finally get 5,376 useful images, which contain (ground truth boxes) 21,815 for eryth, 6,169 for leuko, 6,175 for epith, 1,644 for cryst, 3,663 for cast, 2,083 for mycete and 687 for epithn. From the final 5,376 images, we randomly select 268 images making up 1/20 as test set, and the others as trainval set, where train set makes up 5/6. Figure 5 demonstrates the details of dataset organization and categories distribution. The top pie chart shows how 5,376 images are organized into train/val/test sets. The bottom bar graphs display detailed objects distribution for the imbalanced dataset.

By default, we still use PASCAL-style Average Precision (AP) at a single IoU threshold of 0.5 and the mAP as metric to evaluate different detection architectures. Due to the limited data, we adopt the well-adopted transfer learning mechanism, i.e. first initialize with pre-trained models on ImageNet dataset [30] and then fine tune them using own dataset.

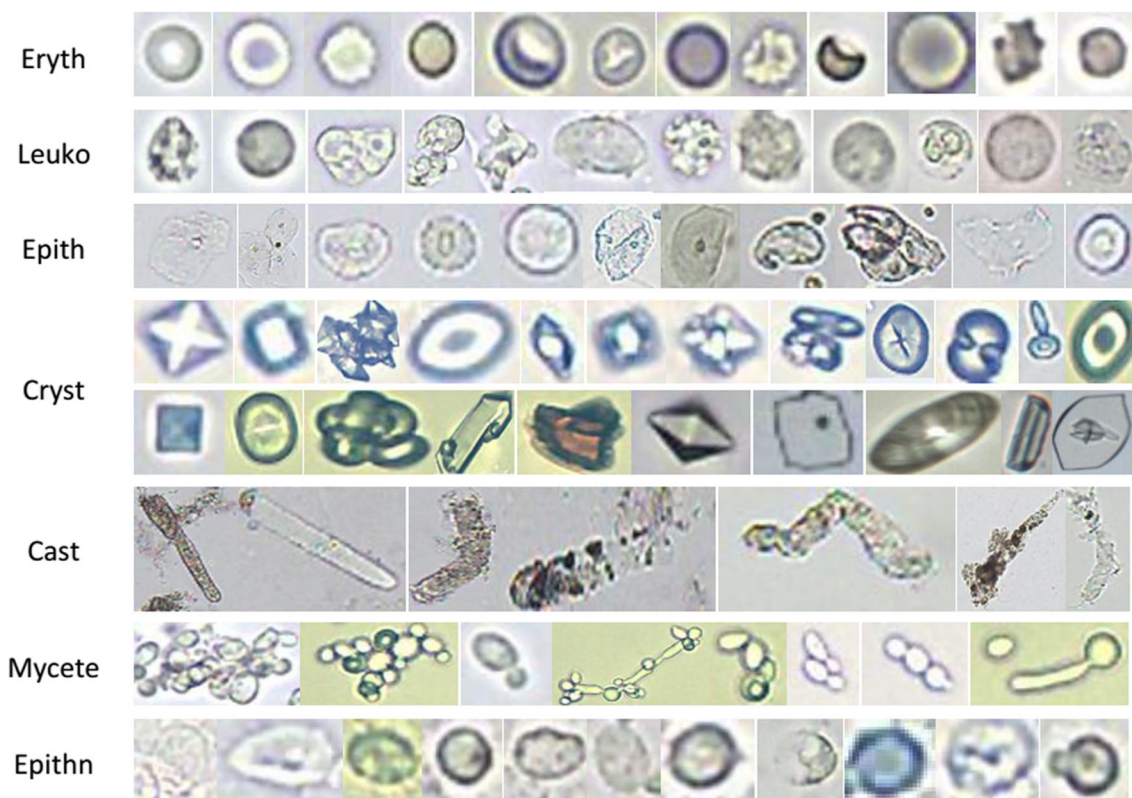


Fig. 4 Selected samples of urinary sediment particle

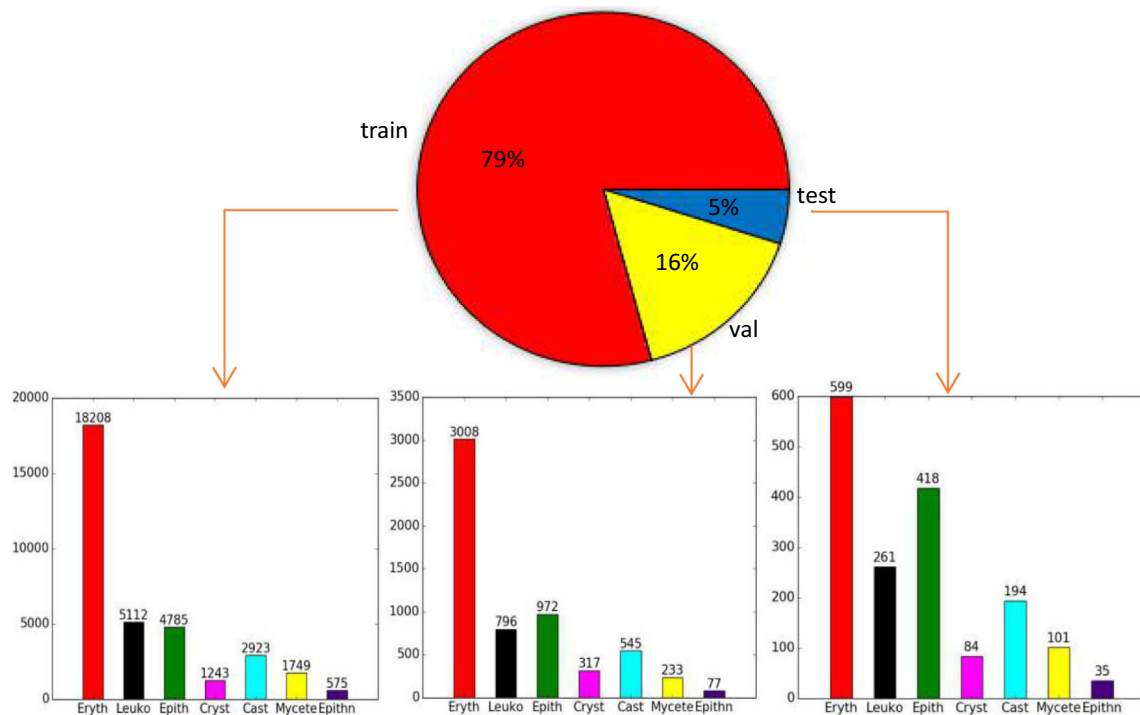


Fig. 5 Dataset organization and categories distribution

Urinary particle recognition based on Faster R-CNN

Feature extractors We first apply a convolutional feature extractor to the input image to obtain high-level features. We mainly focus four feature extractors: ZF net [37], VGG-16 net [35], the ResNet [12] (including ResNet-50 and ResNet-101), and the PVANet [16]. We fine tune all convolutional layers of ZF net and PVANet, but only the conv3_1 and up of VGG-16 net and ResNet.

Training strategies When training Faster R-CNN, we fine-tune pre-trained models with SGD for 70k mini-batch iterations (unless specified otherwise), with a mini-batch size of 128 on 1 NVIDIA Titan X GPU, a momentum of 0.9 and a weight decay of 0.0005. We start from a learning rate 0.001, and decrease it by 1/10 after 50k iterations. But fine-tuning PVANet [16] adopts a learning rate policy of plateau: 0.003 base learning rate, 0.3165 gamma and a different weight decay of 0.0002. As all know, there are two training solutions, 4-step alternating training and approximate joint training (also called as end2end training). In order to select one more effective and efficient solution for the following networks training, we design this experiment based on ZF net [37] and VGG-16 net [35]. Table 1 shows that adopting the strategy of approximate joint training takes less time, but yields higher mAP (nearly the same accuracy on VGG-16 net), so the next series of experiments all adopt the end2end training solution.

Anchor scales Unlike generic objects in natural images, the particles of urinary sediment vary very widely in their shapes, sizes and numbers. Moreover, some urinary microscopical images contain a lot of small objects (like erythrocyte and leukocyte), so as many anchors as possible should be covered in our experiment, especially for small scales.

We compare the detection results under varying anchor scales. First, for networks of ZF, VGG-16 and ResNet we all choose the default settings (the anchor scales of $\{128^2, 256^2, 512^2\}$ and the aspect ratios of $\{1:1, 1:2, 2:1\}$) as benchmarks. Then, keep aspect ratios unchanged and gradually increase anchors with smaller scales (i.e., 64^2 and 32^2). The comparative results are listed in Table 2, which shows that more anchors yield higher mAP in general. However, increasing anchor scales $\{64^2, 128^2, 256^2, 512^2\}$ to $\{32^2, 64^2, 128^2, 256^2, 512^2\}$ can not achieve

Table 1 Training time and mAP by different training solutions

Net	Training strategy	Training time(h)	mAP
ZF	4-step alternating training	5.33	0.694
	end2end training	4.6	0.723
VGG-16	4-step alternating training	12.23	0.756
	end2end training	11.68	0.757

Experiments perform on the ZF and VGG-16 networks, in which the iteration parameters of 4-step alternating training is [80,000 40,000 80,000 40,000]

Bold numbers mean the best performance amongst all the methods

better performance on both ZF net and VGG-16 net. It mainly due to the capacity of networks becoming saturated because we do get an accuracy boost when using ResNet-50 and ResNet-101. Further, we delete the scale of 512^2 as comparison only using ZF and VGG-16 nets. On ZF net, the scales of $\{64^2, 128^2, 256^2\}$ has the same 9 anchors with $\{128^2, 256^2, 512^2\}$, but outperforms by 3.4% mAP. Similarly, on VGG-16 net, the performance is improved by 0.5% mAP. It indicates that most particles in our dataset are small objects and the small anchor scales are indispensable. In addition, we note that deeper networks take more test time, but anchor scales have little impact on detection cost. Finally, it's worth mentioning that the PVANet with best performance takes less test time despite deeper layers, partly because of more anchor scales (5×5) but thin structure.

Data augmentation Commonly, adopting data augmentation in deep learning can expand training samples, avoid over-fitting and improve test accuracy, especially for small-scale training sets. Faster R-CNN also adopts a horizontal flip to augment training set. Empirically, we append a vertical flip to further expand training data. As comparison, we remove all data augmentations and only use original data in training-stage. Table 3 shows us that adopting horizontal flip or vertical flip alone does increase mAPs. However, there is no benefit to further append vertical flip after a horizontal flip.

MS-FRCNN Generally, Faster R-CNN could obtain excellent performance on several natural image benchmarks that hold objects almost occupying the majority of an image. But as mentioned in the previous section, most objects in urine sediment microscopical images are small and low-resolution. Faster R-CNN only uses one higher convolutional layer as feature map, which hardly detects some small objects because of bigger stride and larger receptive field size. Inspired by [24] that combines semantic information from a deep, coarse layer with appearance information from a shallow, fine layer for accurate and detailed segmentations, there have been proposed lots of multi-scale approaches [3, 5, 13, 16, 17], so that the size of receptive field could match various-size objects, especially for small instances.

In order to validate the effectiveness of multi-scale methods for urinary particle recognition, we conduct a series of experiments based on the MS-FRCNN architecture [13]. The final results are shown in Table 4. Overall, MS-FRCNN takes more test time per image and the mAPs are worse a bit than original Faster R-CNN (FRCNN). But we can get an interesting observation from the table, as the number of anchors increases, the final gap between precisions becomes smaller (a difference of 0.4%). In addition, the accuracy of small objects (i.e., eryth, leuko, and epithn) is more superior than no multi-scale. It should

be mentioned that the PVANet also contains a multi-scale structure [17]. We argue that the excellent performance of PVNet partly benefits from it.

OHEM-FRCNN We choose Faster R-CNN as a base object detector and embed the novel bootstrapping technique, online hard example mining (OHEM). As reported in Table 5, OHEM improves the mAP of Faster R-CNN (FRCNN) from 79.5% to 81% while taking approximately the same test time. Specifically, all categories except leukocyte yield better APs, where erythrocyte, cast and epithelial nuclei benefit more. In addition, the gains from OHEM can be increased by enlarging and complicating training set.

Urinary particle recognition based on SSD

When training SSD, we fine-tune a pre-trained model with SGD for 120k mini-batch iterations, with a mini-batch size of 32 on 1 NVIDIA Titan X GPU (a mini-batch size of 16 on 2 GPU during SSD500 training), a momentum of 0.9 and a weight decay of 0.0005. By default, we adopt the multistep learning rate policy with a base learning rate of 0.001 (0.01 when use batch normalization for all newly added layers), a stepvalue of [80,000, 10,000, 120,000] and a gamma of 0.1.

Scales of the default boxes We have known that SSD discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales at each feature map cell. In order to relate these default boxes from different feature maps to corresponding receptive fields, Liu et al. [23] designed a scale strategy that regularly but roughly responses specific boxes to specific areas of the image, where the lowest feature map has a minimum scale of S_{min} and the highest feature map has a maximum scale of S_{max} , and all other feature maps in between are regularly scattered (more details, please refer to the paper [23]).

Considering lots of small particles in urine sediment images, we adjust empirically the scales of default boxes when training SSD300 and the experimental results are listed in Table 6, from which we can see that decreasing the minimum scale of 0.2 to 0.1 (the maximum scale of 0.9 remains unchanged.) increases mAP by 2% in which the AP of cast increases by 14.1%.

Input sizes Generally, increasing the size of input images can improve detection accuracy, especially to small objects. We also increase the input size from 300×300 to 500×500 . Here we train SSD500 only once, with a minimum scale of 0.1 and a maximum scale of 0.9. Unfortunately, we just obtain a poor 65.8% mAP (the last row in Table 6), because of decreasing the batch size setting from 32 to 16 to run this model in limited GPU resources. We argue that better results can be achieved if increase the setting of batch size.

Table 2 Comparisons of detection results using different networks and different anchor scales

Net	Anchor scales	mAP	eryth	leuko	epith	cryst	cast	mycete	epithn	Test time (sec/img)
ZF	{128 ² , 256 ² , 512 ² }	0.723	0.607	0.749	0.845	0.856	0.658	0.781	0.566	0.044
	{64 ² , 128 ² , 256 ² , 512 ² }	0.796	0.853	0.809	0.855	0.858	0.671	0.861	0.665	0.045
	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.779	0.859	0.805	0.854	0.847	0.657	0.863	0.57	0.046
VGG-16	{64 ² , 128 ² , 256 ² }	0.757	0.748	0.823	0.846	0.85	0.642	0.82	0.568	0.044
	{128 ² , 256 ² , 512 ² }	0.757	0.599	0.772	0.874	0.794	0.708	0.874	0.679	0.102
	{64 ² , 128 ² , 256 ² , 512 ² }	0.802	0.842	0.818	0.868	0.873	0.716	0.877	0.621	0.104
ResNet-50	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.795	0.854	0.825	0.857	0.851	0.724	0.876	0.576	0.104
	{64 ² , 128 ² , 256 ² }	0.762	0.743	0.822	0.863	0.759	0.712	0.88	0.558	0.104
	{128 ² , 256 ² , 512 ² }	0.77	0.613	0.831	0.853	0.852	0.757	0.873	0.615	0.219
ResNet-101	{64 ² , 128 ² , 256 ² , 512 ² }	0.784	0.761	0.824	0.86	0.822	0.768	0.859	0.595	0.219
	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.804	0.876	0.812	0.86	0.854	0.747	0.874	0.605	0.22
	{128 ² , 256 ² , 512 ² }	0.761	0.606	0.83	0.864	0.802	0.769	0.875	0.578	0.268
PVANet	{64 ² , 128 ² , 256 ² , 512 ² }	0.773	0.841	0.814	0.848	0.852	0.749	0.863	0.446	0.267
	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.801	0.872	0.809	0.839	0.852	0.772	0.883	0.581	0.268
	{48 ² , 96 ² , 144 ² , 256 ² , 512 ² }	0.841	0.884	0.843	0.871	0.877	0.765	0.890	0.760	0.072

By default, we keep anchor ratios unchanged: only PVANet sets it to {0.5, 0.667, 1.0, 1.5, 2.0}, all the others set {2:1, 1:1, 1:2}. The last column is an approximate time of net forward-propagating when test an image. And the last two rows are test results after 60k mini-batch iterations

Bold numbers mean the best performance amongst all the methods

Table 3 The effect of data augmentation on test precision

Flip types	mAP	eryth	leuko	epith	cryst	cast	mycete	epithn
No flip	0.748	0.865	0.819	0.826	0.764	0.582	0.844	0.533
Only horizontal flip	0.779	0.859	0.805	0.854	0.847	0.657	0.863	0.57
Only verticle flip	0.767	0.853	0.827	0.855	0.879	0.647	0.858	0.448
Horizontal and verticle flip	0.742	0.756	0.795	0.836	0.771	0.677	0.763	0.599

The network is ZF using a anchor scales of {32², 64², 128², 256², 512²} and a aspect ratios of {1:1, 1:2, 2:1}
 Bold numbers mean the best performance amongst all the methods

Table 4 Comparisons on ZF net using different anchor scales when adding a multi-scale structure from MS-FRCNN

Method	Anchor scales	mAP	eryth	leuko	epith	cryst	cast	mycete	epithn	Test time (sec/img)
FRCNN	{128 ² , 256 ² , 512 ² }	0.723	0.607	0.749	0.845	0.856	0.658	0.781	0.566	0.044
MS-FRCNN	{128 ² , 256 ² , 512 ² }	0.712	0.601	0.747	0.817	0.822	0.61	0.781	0.607	0.075
FRCNN	{64 ² , 128 ² , 256 ² , 512 ² }	0.796	0.853	0.809	0.855	0.858	0.671	0.861	0.665	0.045
MS-FRCNN	{64 ² , 128 ² , 256 ² , 512 ² }	0.756	0.845	0.811	0.824	0.835	0.639	0.815	0.512	0.08
FRCNN	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.779	0.859	0.805	0.854	0.847	0.657	0.863	0.57	0.046
MS-FRCNN	{32 ² , 64 ² , 128 ² , 256 ² , 512 ² }	0.775	0.867	0.81	0.836	0.809	0.646	0.871	0.589	0.077

Bold numbers mean the best performance amongst all the methods

Table 5 Comparisons between FRCNN and OHEM-FRCNN on VGG-16 net using the same anchor scales of {32², 64², 128², 256², 512²}

Method	mAP	eryth	leuko	epith	cryst	cast	mycete	epithn	Test time (sec/img)
FRCNN	0.795	0.854	0.825	0.857	0.851	0.724	0.876	0.576	0.104
OHEM-FRCNN	0.810	0.871	0.807	0.866	0.859	0.755	0.877	0.633	0.115

Bold numbers mean the best performance amongst all the methods

Table 6 Detection results using SSD model, where SSD300 has an input size of 300 x 300, SSD500 increases it to 500 x 500, and the penultimate row, SSD300*, represents a Trimmed SSD removing conv7, conv8, and conv9 layers

SSD model	S _{min}	S _{max}	mAP	eryth	leuko	epith	cryst	cast	mycete	epithn	Test time (sec/img)
SSD300	0.2	0.9	0.732	0.841	0.764	0.828	0.745	0.559	0.797	0.587	0.021
	0.1	0.9	0.752	0.766	0.741	0.838	0.782	0.7	0.839	0.596	0.021
SSD300*	0.2	0.9	0.773	0.846	0.748	0.837	0.772	0.721	0.85	0.638	0.021
SSD500	0.1	0.9	0.658	0.557	0.609	0.834	0.632	0.669	0.792	0.512	0.047

Bold numbers mean the best performance amongst all the methods

Trimmed SSD In order to reduce the complexity of the original SSD method and avert over-fitting on the small-scale urinalysis database, we propose a Trimmed SSD by removing conv7, conv8, and conv9 layers of SSD300. Surprisingly, as shown in the penultimate row of Table 6, such a simple pruning does yield a better mAP (a boost of 4.1%).

Adding bells & whistles

As mentioned, Faster R-CNN methods exceed SSD methods in detection accuracy, but are often slower due to the region proposal generation stage. In this section, we study the impact of several factors to region proposal

generation while combining above experiments in details. Further, we also compare PVANet against VGG-16 on specific detection performances more intuitively.

Region proposal generation

Anchor scales We further provide analysis of anchor scales influence on object proposals on VGG-16 net. The curve of recall for anchor scales at different proposal numbers is plotted in Fig. 6a. Correspondingly, the related detection performances are shown in Table 2 (the VGG-16 module). As Fig. 6a shows, from the default anchor scales $\{128^2, 256^2, 512^2\}$, the proposals recall increases gradually while adding smaller scales (i.e., 64^2 and 32^2), and the scales of $\{64^2, 128^2, 256^2\}$ outperforms the scales of $\{128^2, 256^2,$

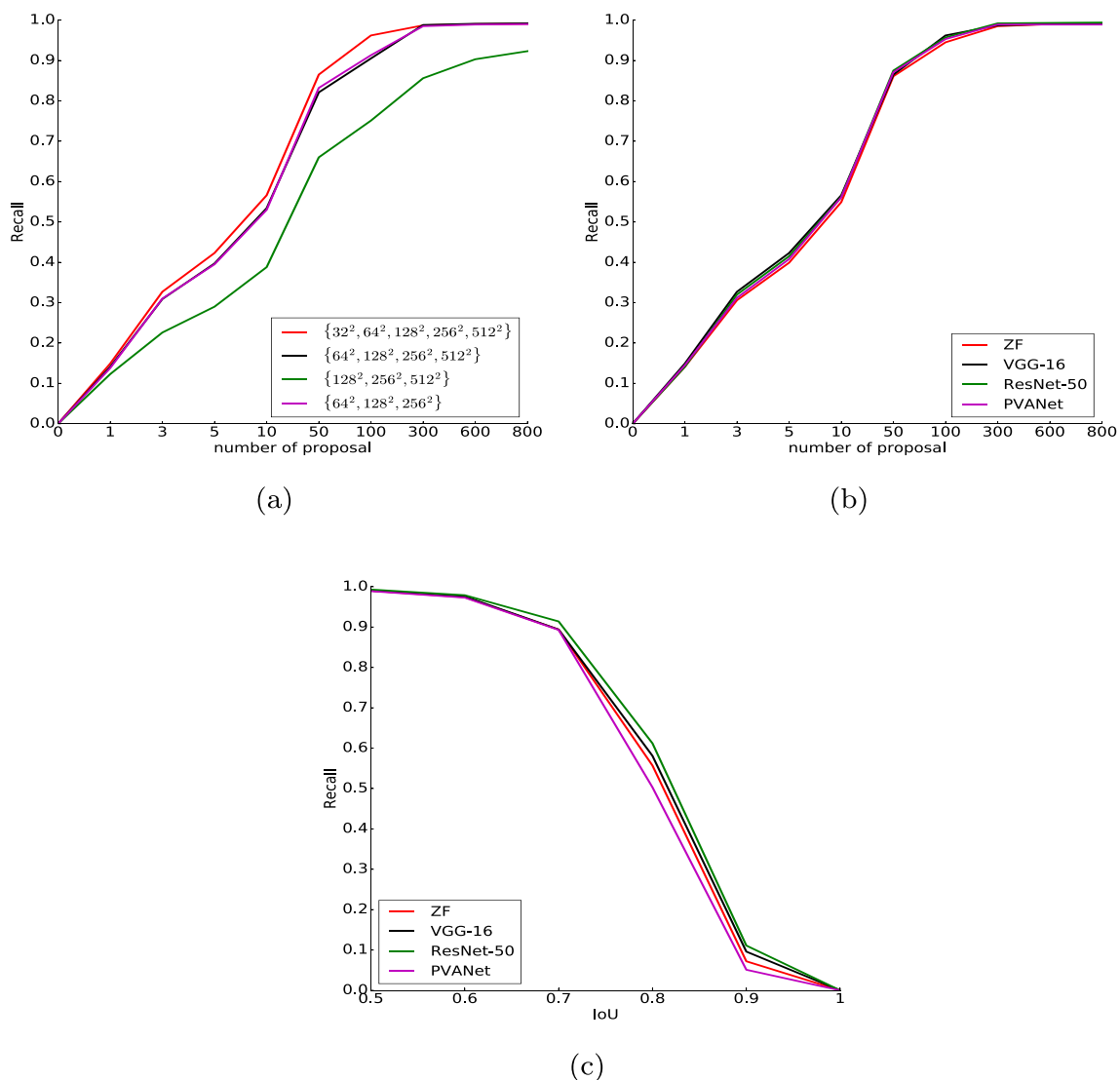
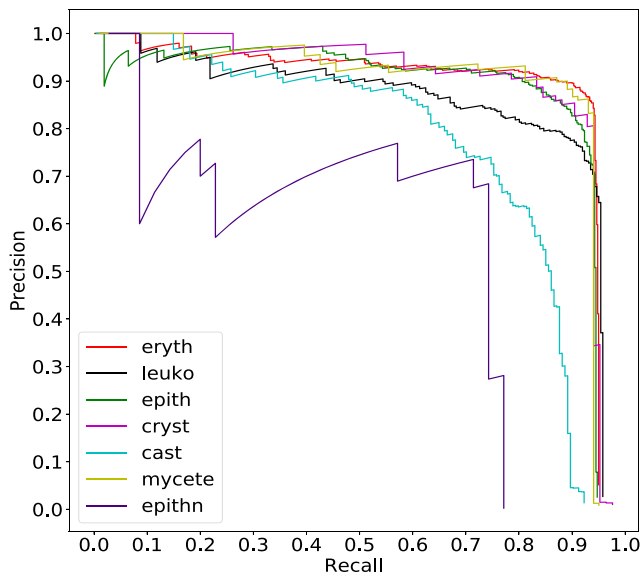
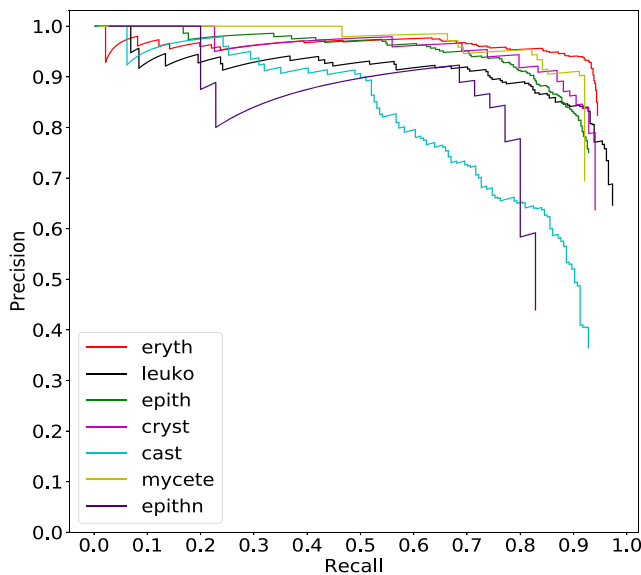


Fig. 6 Analysis for region proposal generation on urinalysis database. **a** Recall versus number of proposal for different anchor scales using VGG-16 with a fixed IoU of 0.5. **b** Recall versus number of proposal

for different networks with a fixed IoU of 0.5. **c** Recall versus IoU threshold for different networks with a fixed number of proposal (600)



(a) VGG-16



(b) PVANet

Fig. 7 Precision versus recall. **a** VGG-16 net with a anchor scales of { 32^2 , 64^2 , 128^2 , 256^2 , 512^2 } and a mAP of 79.5%. **b** PVANet with a mAP of 84.1%

512^2 } by a significant margin, closed to the scales of { 64^2 , 128^2 , 256^2 , 512^2 }. The results are consistent with the detection accuracy with respect to mAP & APs, and indicates two keys about anchor scales: (1) the more the better, and (2) the smaller the superior. Reasonable design of anchor scales benefits both proposals generation and final detection.

Feature extractors Table 2 has already illustrated the detection performances with respect to accuracy and speed using different networks. Treating RPN as a class-agnostic object detector, here we further investigate performances of different networks in terms of proposals quality. Figure 6b, plotting recall versus number of proposals with a loose IoU of 0.5, shows little differences between several networks when adopting the best anchor scales of { 32^2 , 64^2 , 128^2 , 256^2 , 512^2 }. For higher IoU thresholds, shown in Fig. 6c, the recall of PVANet drops faster than other networks.

PVANet vs. VGG-16

Although PVANet achieves worse performances for region proposal generation, eventually it achieves the highest detection results, a mAP of 84.1% shown in Table 2. We conjecture that it owes to the sophisticated Fast R-CNN detector stage. To verify the conjecture, we compare PVANet against VGG-16 and Fig. 7 shows curves of precision-to-recall separately on PVANet and VGG-16 networks. In contrast, PVANet maintains higher precisions stably, as recalls increases. The precisions of VGG-16 net drop sharply in the end. For detections of cast and epithelial nuclei, PVANet also performs better than VGG-16 net.

Conclusions

In this paper, we treat the urinary particle recognition as object detection and propose to exploit modern CNN-based object detectors, Faster R-CNN and SSD, for automatic urinary particle recognition. They are segmentation free and can learn task-specific features in an end-to-end manner. When applying Faster R-CNN, SSD, and their variants to urinary particle recognition, we effectively adopt the mechanism of deep transfer learning. Moreover, we conduct extensive experimental analysis to demonstrate the impact of various factors, including training strategies, network structures, anchor scales, and so on. After conducting comprehensive experiments, we obtain a best mAP of 84.1% with a test time of 70 ms per image while using Faster R-CNN on PVANet.

Acknowledgements This research was partially supported by the Natural Science Foundation of Hunan Province, China (No. 14JJ2008) and the National Natural Science Foundation of China under Grant No. 61602522, No. 61573380, No. 61672542.

Compliance with Ethical Standards

Conflict of interests The authors declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

Appendix

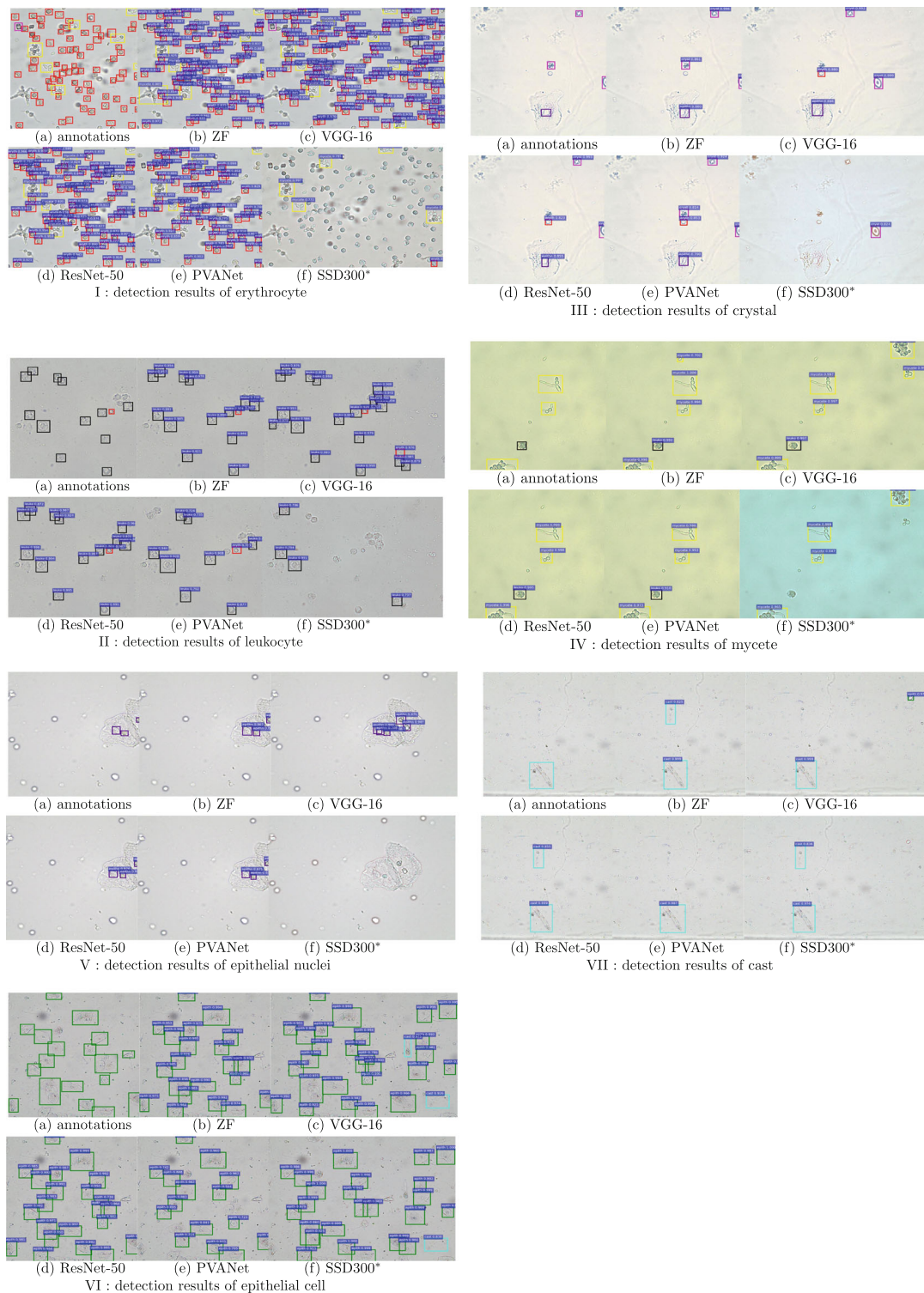


Fig. 8 Selected detection examples of urinary particle on urinalysis test set. We show detections with scores higher than 0.7. All examples are divided into 7 groups, where 5 groups are at high-power field (i.e., erythrocyte, leukocyte, crystal, mycete, epithelial nuclei) and the other 2 groups at low-power field (i.e., epithelial cell, cast). In each group: **a** shows original image with ground truth boxes; **b-d** are Faster R-CNN detections separately on ZF, VGG-16 and ResNet-50 networks

with a anchor scales of $\{32^2, 64^2, 128^2, 256^2, 512^2\}$; **e** shows detection results on PVANet; **f** shows detection results on SSD300* model. For the ground truths and detection boxes, different categories use only different colors: eryth (red), leuko (black), epith (green), crystal (magenta), cast (cyan), mycete (yellow). As shown in this figure, the performance of SSD is inferior to Faster R-CNN, and it misses a lot of small objects

References

- Almadhoun, M. D., and El-Halees, A., Automated recognition of urinary microscopic solid particles. *Journal of medical engineering & technology* 38(2):104–110, 2014.
- Avci, D., Leblebicioglu, M. K., Poyraz, M., and Dogantekin, E., A new method based on adaptive discrete wavelet entropy energy and neural network classifier (ADWEENN) for recognition of urine cells from microscopic images independent of rotation and scaling. *Journal of medical systems* 38(2):7, 2014.
- Bell, S., Lawrence Zitnick, C., Bala, K., and Girshick, R., Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2874–2883, 2016.
- Budak, Y. U., and Huysal, K., Comparison of three automated systems for urine chemistry and sediment analysis in routine laboratory practice. *Clinical laboratory* 57(1):47, 2011.
- Cai, Z., Fan, Q., Feris, R. S., and Vasconcelos, N., A unified multi-scale deep convolutional neural network for fast object detection. *European conference on computer vision*, pp. 354–370: Springer, 2016.
- Chien, T. I., Kao, J. T., Liu, H. L., Lin, P. C., Hong, J. S., Hsieh, H. P., and Chien, M. J., Urine sediment examination: a comparison of automated urinalysis systems and manual microscopy. *Clinica Chimica Acta* 384(1):28–34, 2007.
- Girshick, R., Fast r-CNN. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, 2015.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J., Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.
- Han, J., Zhang, D., Cheng, G., Liu, N., and Xu, D., Advanced deep-learning techniques for salient and category-specific object detection: a survey. *IEEE Signal Processing Magazine* 35(1):84–100, 2018.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R., Mask r-CNN. In: *IEEE International conference on computer vision (ICCV)*, pp. 2980–2988: IEEE, 2017.
- He, K., Zhang, X., Ren, S., and Sun, J., Spatial pyramid pooling in deep convolutional networks for visual recognition. In: *European conference on computer vision*, pp. 346–361: Springer, 2014.
- He, K., Zhang, X., Ren, S., and Sun, J., Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
- Hoang Ngan Le, T., Zheng, Y., Zhu, C., Luu, K., and Savvides, M., Multiple scale faster-RCNN approach to driver's cell-phone usage and hands on steering wheel detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 46–53, 2016.
- Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al., Speed/accuracy trade-offs for modern convolutional object detectors. In: *IEEE CVPR*, Vol. 4, 2017.
- İnce, F. D., Ellidağ, H. Y., Koseoğlu, M., Şimşek, N., Yalçın, H., and Zengin, M. O., The comparison of automated urine analyzers with manual microscopic examination for urinalysis automated urine analyzers and manual urinalysis. *Practical Laboratory Medicine* 5:14–20, 2016.
- Kim, K. H., Hong, S., Roh, B., Cheon, Y., and Park, M., Pvanet: Deep but lightweight neural networks for real-time object detection. arXiv:1608.08021, 2016.
- Kong, T., Yao, A., Chen, Y., and Sun, F., Hypernet: towards accurate region proposal generation and joint object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 845–853, 2016.
- Kouri, T., Fogazzi, G., Gant, V., Hallander, H., Hofmann, W., and Guder, W., European urinalysis guidelines. *Scandinavian Journal of Clinical and Laboratory Investigation-Supplement* 60(231), 2000.
- Li, C., Tang, Y. Y., Luo, H., and Zheng, X., Join gabor and scattering transform for urine sediment particle texture analysis. In: *2nd international conference on Cybernetics (CYBCONF), 2015 IEEE*, pp. 410–415: IEEE, 2015.
- Li, Y., and He, K., Sun, J., others: r-FCN: Object detection via region-based fully convolutional networks. In: *Advances in neural information processing systems*, pp. 379–387, 2016.
- Liang, Y., Fang, B., Qian, J., Chen, L., Li, C., and Liu, Y., False positive reduction in urinary particle recognition. *Expert Systems with Applications* 36(9):11,429–11,438, 2009.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S., Feature pyramid networks for object detection. In: *CVPR*, Vol. 1, p. 4, 2017.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C., SSD: Single shot multibox detector. In: *European conference on computer vision*, pp. 21–37: Springer, 2016.
- Long, J., Shelhamer, E., and Darrell, T., Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.
- Ranzato, M., Taylor, P., House, J., Flagan, R., LeCun, Y., and Perona, P., Automatic recognition of biological particles in microscopic images. *Pattern recognition letters* 28(1):31–39, 2007.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, 2016.
- Redmon, J., and Farhadi, A., Yolo9000: better, faster, stronger. In: *2017 IEEE Conference on computer vision and pattern recognition (CVPR)*, pp. 6517–6525: IEEE, 2017.
- Redmon, J., and Farhadi, A., 2018.
- Ren, S., He, K., Girshick, R., and Sun, J., Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, pp. 91–99, 2015.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. et al., Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115(3):211–252, 2015.
- Schmid, C., and Mohr, R., Local grayvalue invariants for image retrieval. *IEEE transactions on pattern analysis and machine intelligence* 19(5):530–535, 1997.
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y., Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv:1312.6229, 2013.
- Shen, M. I., and Zhang, R., Urine sediment recognition method based on svm and adaboost. In: *International conference on Computational intelligence and software engineering, 2009. ciSE 2009*, pp. 1–4: IEEE, 2009.
- Shrivastava, A., Gupta, A., and Girshick, R., Training region-based object detectors with online hard example mining. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 761–769, 2016.
- Simonyan, K., and Zisserman, A., Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

36. Uijlings, J. R., Van De Sande, K. E., Gevers, T., and Smeulders, A. W., Selective search for object recognition. *International journal of computer vision* 104(2):154–171, 2013.
37. Zeiler, M. D., and Fergus, R., Visualizing and understanding convolutional networks. In: *European conference on computer vision*, pp. 818–833: Springer, 2014.
38. Zhang, L., Lin, L., Liang, X., and He, K., Is Faster r-CNN doing well for pedestrian detection? In: *European conference on computer vision*, pp. 443–457: Springer, 2016.
39. Zhang, S., Wen, L., Bian, X., Lei, Z., and Li, S. Z., Single-shot refinement neural network for object detection. In: *IEEE CVPR*, 2018.
40. Zhou, Y., and Zhou, H., Automatic classification and recognition of particles in urinary sediment images. In: *Computer, informatics, cybernetics and applications*, pp. 1071–1078: Springer, 2012.
41. Zitnick, C. L., and Dollár, P., Edge boxes: Locating object proposals from edges. In: *European conference on computer vision*, pp. 391–405: Springer, 2014.