A Semiautomatic Sperm Cell Data Annotator for Convolutional Neural Network

Priyanto Hidayatullah
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
priyanto@polban.ac.id

Tati Erawati Rajab Mengko
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
tati@stei.itb.ac.id

Rinaldi Munir
School of Electrical Engineering and Informatics,
Institut Teknologi Bandung
Bandung, Indonesia
rinaldi@ informatika.org

Anggraini Barlian
School of Life Sciences and Technology,
Institut Teknologi Bandung
Bandung, Indonesia
aang@sith.itb.ac.id

Abstract— **Convolutional Neural Network (CNN) is one of deep learning methods which is very powerful for solving image processing and computer vision problems. The powerfulness of this method relies on the number and the quality of the training data. The need for numerous and valid annotated training data is crucial. Manual annotation is still the most popular option for obtaining a valid data annotation. There are some manual annotation systems available. Nevertheless, manually annotating a massive amount of data is remarkably tiresome and need plenty of time. This is the problem that we addressed. In this paper, we propose a system to annotate a CNN training data semi-automatically to accelerate the annotation process. We use an extreme case in which the number of objects to annotate inside a video frame is plentiful (334 - 472 objects). We compare the accuracy of its annotation and the speed of the annotation process to the available annotation systems. The result is that our proposed method reduces the average annotation time up to 48.7%.**

Keywords—Convolutional Neural Network, data annotation system, data preparation, deep learning, sperm detection

I. INTRODUCTION

A sequential process of problem-solving using supervised CNN is illustrated in Fig. 1. After understanding the nature of the problem, collecting relevant data is the first step to be carried out. Depending on the nature of the problem, but generally, CNN needs a massive amount of data to reach its best performance.

Data preparation and validation is the next stage. This stage should not be neglected as this gives a significant difference to the result. A vast amount of rough validated training data can give more inferior testing result than a less amount satisfactory validated one. This stage also tells us that the need for CNN to be powerful is, not only numerous data but also the valid one.
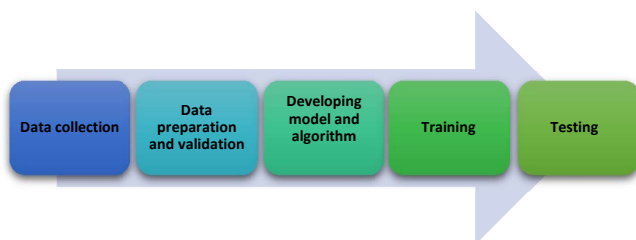

Fig. 1.   The process of problem-solving using CNN

The subsequent stage is developing the model and refining its algorithm. An effective CNN model can be varied depending on its problem characteristics. Refining and improving the model architecture can significantly improve the result and the speed of training and testing.

In the training stage, speed is the primary concern. Most researchers use a better computer to speed up the training process. Even though, there is a researcher who tries to accelerate the training by compressing the model architecture [1]. Some others tune some training parameters to improve speed while maintaining its testing accuracy.

The last stage is testing. Again, speed is the primary concern. Nevertheless, in general, testing is much faster compared to training. There is not much improvement can be done in this stage compared to other stages. The improvement is more to programmatic improvement.

In the sense of workload of a researcher in performing all these stages, data preparation and validation (i.e., annotation in this case) takes a lot of time and effort. Validity is the primary concern of this stage. As a consequence, the stage has to be performed by manual operation. Furthermore, the number of data has an important role as well. It leads to time-consuming, tiredness, and extreme boredom. This stage becomes the hardest part for a researcher to execute and the least enjoyable stage of all. For example, it needs more than 70,000 worker hours to gather, annotate, and organize MS COCO dataset[2], the well-known dataset for object detection, segmentation, and captioning. Nevertheless, this stage result is crucial to the result accuracy of the CNN model.

This research objective is to make data annotation easier and faster. In this paper, we propose a system to help the researcher executing this stage and resolve the annotation problem. What we propose is to perform automatic annotation to most of the objects inside a video frame and do manual annotation to the rest.

In this paper, we handle a real case that is challenging because of two reasons. First, there are numerous objects in one video frame (334 - 472). Second, the object's size is tiny, which is less than 10x10 pixels in 640x480 sample video frames. The object itself has similar shape one another. Based on our experience, performing annotation to these kinds of video frames often hurts our eyes.

We organize the paper in the following structure. The first section explains the background and the proposal of the paper. The second section reviews the associated works previous to this study. The third section describes the proposed method in more detail. The results are explained and analyzed in the fourth section. The last section ends the study and provides perspectives on future work.

II. PREVIOUS WORKS

There are several states of the art of data annotation systems available[3]. As an illustration of their capability, we utilized them for annotating several objects such as a wristwatch, person, pen, laptop, cup, and cellphone.

VGG Image Annotator (VIA) [4] is an offline web-based annotation software for general object and faces. Several researchers from Visual Geometry Group, Department of Engineering Science, University of Oxford, develop this open-source software. The advantage of this software is that it has more marker types, which are dot, rectangle, circle, ellipse, line, polygon, and polyline. The disadvantage of this system is the use of a single annotation color for all objects. We can export the annotation result only to CSV file. Fig. 2 is an example of its annotation result.

LabelMe [5] is a free online web-based annotation system for computer vision purposes. Some researchers from MIT Computer Science and Artificial Intelligence Laboratory develop this system. It has two kinds of markers which are a polygon and a mask. We can export the annotation result into XML format. Fig. 3 is an example of its annotation result.

Labelbox [6] is a commercial online web-based annotation system for classification and segmentation purposes. It has different types of marker, which are a point, line, brush, and superpixel. After finishing the annotation, we can export the annotation result into four different formats such as JSON, CSV, COCO, and, VOC. We can generate the mask which is compatible with some model like Mask R-CNN. Labelbox has the best user experience so far. When we draw a marker on the object, the polygon marker will move to approach the object border, which makes annotation easier. Fig. 4 is an example of its annotation result.
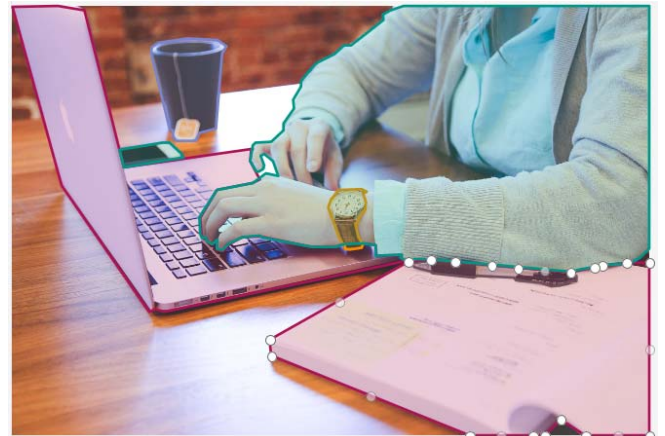
As a short conclusion, there are annotation systems available. The annotation result of these systems can be exported to different kind of formats, which is beneficial. They also have various kind of markers with which they can cover a different kind of object shapes. Unfortunately, these are all manual systems. There are some real cases where the number of objects inside a video frame is numerous. Sperm cell detection and bacteria detection are some of the examples. For these cases, the systems are not suitable as the operator can have exhaustion and extreme boredom.

III. METHODS

In this paper, we performed several stages for semi-automatically annotating the data. This section describes those stages. As an illustration, Fig. 5 is an overview illustration of those stages.

A. Samples

We utilize three bull sperm microscopic observation videos. We extract one frame from each video and annotate them using our proposed method and the currently available

Fig. 2.   VIA annotation result

Fig. 3.   LabelMe annotation result

Fig. 4.   Labelbox annotation result

data annotation system. We employ a phase-contrast microscope for its observation with 400x magnification. The dimension of the video frame is 640x480 pixels. Table I shows the number of objects (sperm cells) inside each frame. Fig. 6 is one of the frame samples.

TABLE I.   SAMPLE SPECIFICATION

	Sample #1	Sample #2	Sample #3	Average
Number of objects	409	472	334	405

B. Implementation Environment

We utilize a Windows 10 Pro (x64) laptop with Intel Core i3 @2.0 GHz, 8 GB RAM, and an NVIDIA GeForce 920MX GPU card for developing and testing the system. The system was built under the .NET Framework environment and coded in C++ using Microsoft Visual Studio 2017. We utilize the powerful OpenCV 3.4.1 library, which contains plentiful useful functions. Besides, we use a 3200 dpi mouse for marking the objects. Using a higher resolution device is crucial for marking small objects.

C. Experiment Setting

The objective of the experiment is to measure the benefit of the proposed system compared to existing systems. For this objective, we performed data annotation on the samples using the proposed system as well as all existing systems. To measure its data annotation accuracy, we employed a system provided by Mask R-CNN Matterport implementation [7] which called data inspection. To measure the speed of annotation, we recorded the time required for data annotation on all system. Afterward, we do a benchmark for all the result.

For the case of this paper, we use annotation for instance segmentation purpose. This case is better to test an annotation system as it is a more complicated case. In instance segmentation, the detection is in pixel level. Therefore the marker is endeavored to be a mask which covers all the object part and left the background unmarked.

D. The proposed method

The objective of the proposed method is to make data annotation faster and easier compared to the manual one with still maintaining its annotation accuracy.
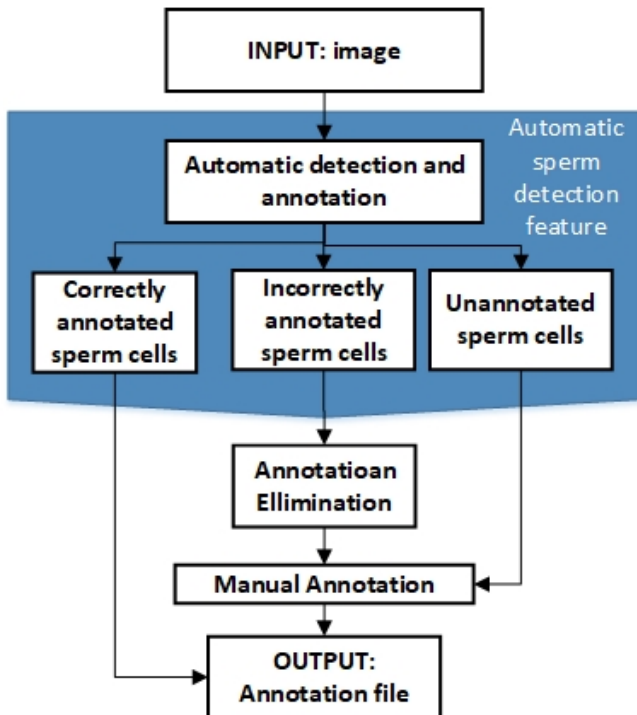

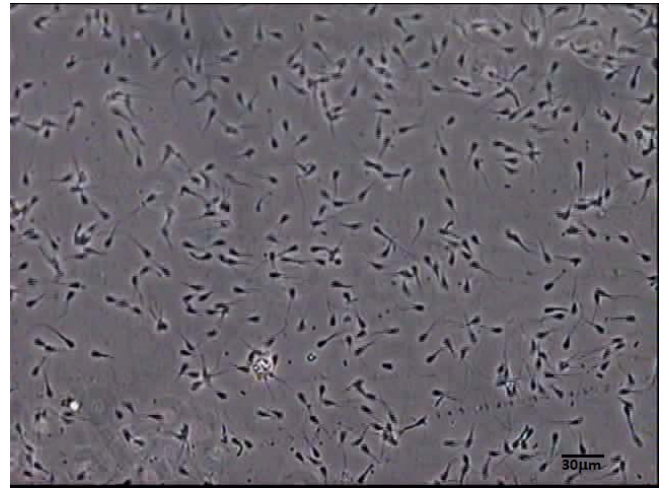Fig. 5.   The structure of the proposed system

Fig. 6.   Example of frame samples

1) The automatic annotation feature

We handle annotating sperm cells in video frames case in this paper. The main idea of the proposed method is to use the existing sperm cell detection method to detect and annotate the cells in the first stage automatically. We utilize the standard image processing based method [8]. The annotations are drawn on the objects as closing curves and listed in a specific list of curves.

To have the maximum number of bull sperm cells detection, we do some parameter tuning to the method. Based on our experiment, we set the minimum and maximum head size of the cells to 3 and 7, which gives the optimum number of cell detection. There are still unannotated cells or even incorrectly annotated ones. The next proposed method features handle these cells.

Using a sperm detection method for annotating data, which is then utilized for performing sperm detection seems to be recursive. Nevertheless, to make it clear once again, the standard image processing method has reached its maximum detection capability to 91%[8] for sperm detection case in its ideal experiment setting. The samples themselves have a small number of objects in a single video frame. While in recent years, the deep learning method has given a more promising result in image processing and computer vision problems. The deep learning-based method, which uses the CNN model, is expected to increase the detection result of the case.

Unfortunately, for this objective, the deep learning-based method needs numerous valid training data, which is time-consuming and tedious if we undertake it manually. The previous standard image processing method role is to speed up this annotation by automatically annotate most of the sperm cells. As the method was not perfect in detecting all the cells, the next features of the proposed method are introduced to complement its imperfection.

One may have an idea to use only the automatic annotation. We do not recommend such an idea as no automatic approach that can annotate all the objects inside a video frame. There will be some unannotated objects. If one pushes to use an annotation result which contains some unannotated objects, the accuracy of the detection result from the trained model will not be optimum.

2) Inaccurate annotation elimination feature

Even though we have carefully chosen the parameter, there is still the possibility that the previous method gives false positives annotations. To eliminate these, we put a mark by clicking the part of the inaccurate annotation curve. From the 1) step, we have the list of annotation curves. Technically, we utilize pointPolygonTest from OpenCV library and iterate it to all existing curve in the list. In the process, we evaluate whether the coordinate of the click is part of an existing curve or not. If it is part of the curve, which pointPolygonTest result is more than 0, then the curve will be superimposed by white color which is then deleted from the list of valid annotation curves. Equation 1 is a more formal notation of it.

$$\text{pointPolygonTest}(\text{Point}(x, y)) \begin{cases} \geq 0, erased \\ else, skip \end{cases} \quad (1)$$

3) Manual annotation feature

The results average of automatic detection and annotation in this paper samples is 85.8%. It is very beneficial as we only have to annotate about 15% sperms left. We add a manual annotation feature in order to annotate sperm that cannot be annotated automatically.

These sperms that cannot be detected automatically are generally sperm with severe cases such as: blurred, overlapping with other sperm so that the shape is not like an ellipse anymore, cut off by the border, and its size is too large or too small due to poor contrast.

The expected annotation result is a mask that covers the entire surface of the sperm cell. Because sperm tends to be elliptical, the marker used is an ellipse. The critical thing to consider when drawing an ellipse is the direction of the ellipse. It depends on the gradient of the marking direction. If the ellipse's direction is upright, it must be dealt with so that it is not divided by zero. In that case, we set the angle to 90º.

Determining the major axis (*maxAxis*) can be done by calculating the distance between the start (*startPoint*) and the end of the tagging (*endPoint*). We set the minor axis to 60% of the major axis, based on the experimental results. We then draw the ellipse on the frame. Fig. 7 illustrates the process. Equation 2 is the formal form of the algorithm.

$$\begin{aligned} maxAxis \\ = euclideanDistance\ (startPoint, endPoint) \end{aligned} \quad (2)$$

$$minAxis = 0.6 * maxAxis$$

$$gradient = \frac{endPoint.y - startPoint.y}{endPoint.x - startPoint.x} \quad (3)$$

$$angle = \begin{cases} gradient \neq\ \sim, \text{atan}(gradient) * \dfrac{180°}{\theta} \\ else, 90° \end{cases} \quad (4)$$

E. Evaluation and Comparison

As the last step, we compare the annotation results of the proposed method to the previous methods'. For comparing the annotation accuracy, we utilize data inspection program from the Mask R-CNN implementation by Matterport [7]. For speed comparison, we record the time used for annotating each sample. We subsequently compare one another to evaluate the improvement made by the proposed method.

IV. RESULT AND DISCUSSION

The objective of the proposed method is to accelerate and simplify the process of object annotation. We have tested the proposed method and all relevant previous works.

After performing experiments and experiencing all the systems, these are some critical findings to note. We eliminate LabelMe as it is not suitable for this paper's case. LabelMe always set a new class for a new marking. This system is troubled to label many objects with the same class.

VIA has some advantages. It is a free system with a variety of marker shape. There are dot, rectangle, circle, ellipse, line, polygon, and polyline for marking. It helps in annotating a different kind of object shapes. Nevertheless, there are some drawbacks. There is no undo and onscreen delete feature. It is quite annoying as one mistake can lead to restarting the whole annotation process. For this paper's case, which uses ellipse as the marker shape, VIA has some limitations. The ellipse can be drawn only in the horizontal and vertical direction. It makes marking inclined objects, which often occur, complicated.

Labelbox gives the best user experience in performing data annotation, despite it is only a manual system. The system provides four marker shapes, which are point, line, brush, and superpixel. It has some features that make annotating more convenient such as hotkeys, undo, delete, and even edit the previous annotation. It also gives a more variety of output types. The drawback is that the researcher has to pay for it as it is a commercial system. It does not have an ellipse shape marker and no automation in the marking. Labelbox as it is an online web-based system, it turns to sluggish when handling many marking inputs (clicks) in a short time. While our proposed method can handle these inputs with ease

Example of automatic annotation result
Example of sperm that is not successfully automatically annotated
Example of sperm with manual annotation
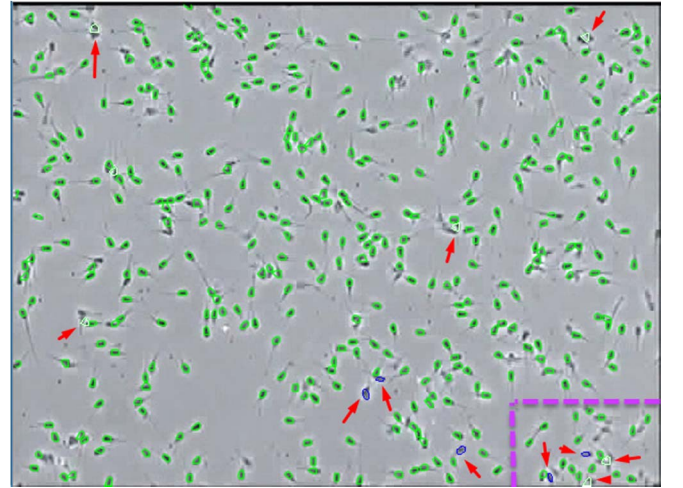Incorrect automatic annotation result

Fig. 7.   Illustration of the proposed method annotation features
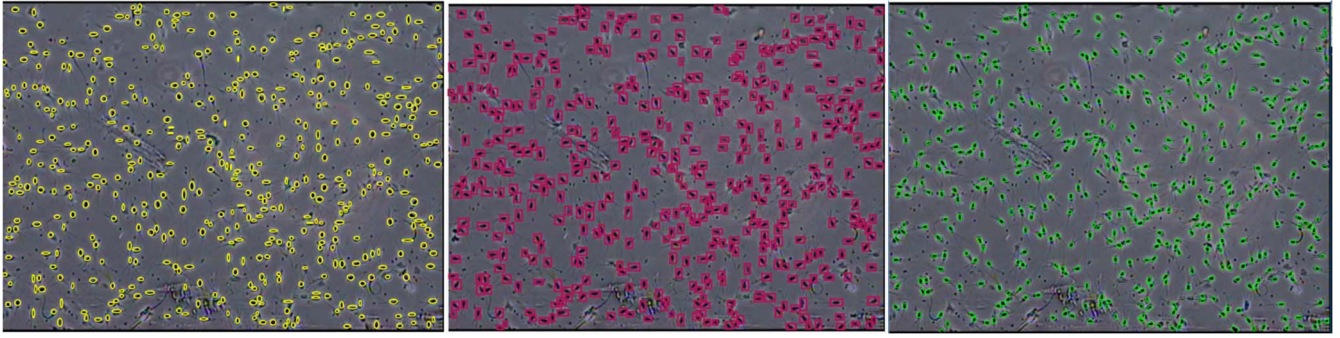
Fig. 8.   Annotation Results Comparison (VIA, Labelbox, Proposed Method)

TABLE II.   ANNOTATION SYSTEMS COMPARISON

| Systems	Platform	Marker	Output Format	Annotation Time			
				Sample #1	*Sample #2*	*Sample #3*	*Average*
VIA	Offline web-based	dot, rectangle, circle, ellipse, line, polygon, and polyline	CSV	32m 03s	34m 39s	25m 00s	30m 34s
Labelbox	Online web-based	point, line, brush, and superpixel	JSON, CSV, COCO, and VOC	30m 02s	33m 44s	24m 22s	29m 23s
Proposed Method	Offline Desktop	ellipse, rectangle	JSON	15m 16s	16m 05s	15m 43s	15m 41s
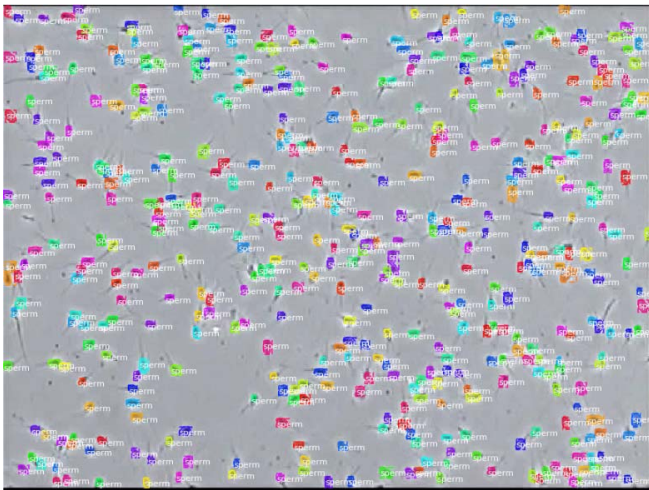
Fig. 9. The result of data inspection

Our proposed method has some significant advantages. The main advantage is the automation for marking most of the objects using the previous sperm detection method. It reduces the average time of annotation 48.7% compared to VIA and 46.6% compared to Labelbox. The system marks most of the sperm cells with an ellipse-like green polygon. If there are some incorrect automatic annotation marker, we can delete it by a right-click which turn the marker into white. Afterward, we can draw a marker to unannotated sperm cells with a blue ellipse marker. Fig .7 is the illustration of this process.

The speed of annotation drops on the third sample. The reason is that there are many artifacts in this sample, which triggers many miss detections on automatic detection. Consequently, we have to delete those miss detection before we do the annotation, which takes more time. It is also critical to note that the previous methods have less time for annotating the third sample compared to the first and second sample

because the third sample has less number of objects. Comparison of the systems is shown in Fig. 8 and Table II.

The other advantage is that even though VIA and the proposed method are using ellipse as the marker, the proposed method marker is better than the ellipse marker of VIA. There is an algorithm difference between them. The proposed method draws ellipse from end to end of the sperm head while the VIA draws from center to the far end of the sperm head. In practice, our method gives more convenience to the annotation operator as the objects are tiny.

In annotation accuracy point of view, as all methods are receiving a manual intervention, all have good accuracy for its final annotation. It means the object annotations are considered to be valid. Nevertheless, as Labelbox use rectangle as the marker, some background part is included inside the marker. It will not be a problem for a general object detection purpose. Nevertheless, this is not suitable for instance segmentation. Fig. 9 gives the illustration of annotation accuracy using data inspection[7].

Besides, the exported annotation result of the proposed method has been tested for training on some popular Mask R-CNN implementations[7], [9] and works very well. Based on the result of this research and comparing to currently available works, we have developed a novel system to annotate a CNN dataset from video frames semi-automatically.

V.   CONCLUSION

The artificial neural network, which CNN is one of the methods, needs a massive amount of data to be powerful enough on solving the classification and object detection problem. In this abundant data era, finding numerous data is not difficult. Nevertheless, annotating the such number of data

is challenging. A proper valid data usually obtained by manual annotation. Unfortunately, performing manual annotation on those vast amounts of data was extremely tedious and tiresome.

In this paper, we have developed a semiautomatic system to annotate sperm cell training data with these features: automatic detection and annotation to most of the objects inside a video frame, elimination the incorrectly annotated objects, manual annotation to unannotated objects as well as the incorrect annotated ones.

This system simplifies the task of the researcher in annotating sperm cell training data for Convolutional Neural Network. The experiment results show that the proposed method increased the speed of the annotation process significantly. It reduces the time up to 48.7% of the manual annotation using other systems. The data inspection also confirms that the annotation result is comparable to the manual ones. It also has been tested on some popular Mask R-CNN implementations[7], [9] and works very well.

In the future, we want to develop the system so that it can have a more variety marker shape (e.g., polygon, lines) and output such as XML and VOC. We also want to utilize the proposed method on other samples.

REFERENCES

[1] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "A Survey of Model Compression and Acceleration for Deep Neural Networks," *arXiv:1710.09282 [cs]*, Oct. 2017.

[2] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," *arXiv:1405.0312 [cs]*, May 2014.

[3] S. Dasiopoulou, E. Giannakidou, G. Litos, P. Malasioti, and Y. Kompatsiaris, "A Survey of Semantic Image and Video Annotation Tools," in *Knowledge-Driven Multimedia Information Extraction and Ontology Evolution*, vol. 6050, G. Paliouras, C. D. Spyropoulos, and G. Tsatsaronis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 196–239.

[4] Abhishek Dutta, Ankush Gupta, and Andrew Zisserman, "VGG Image Annotator (VIA)," 2016. [Online]. Available: http://www.robots.ox.ac.uk/~vgg/software/via/. [Accessed: 12-Oct-2018].

[5] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A Database and Web-Based Tool for Image Annotation," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 157–173, May 2008.

[6] "Labelbox: Create and manage training data: in one place, across teams, at scale." [Online]. Available: https://labelbox.com/. [Accessed: 12-Oct-2018].

[7] W. Abdulla, "Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow," *Matterport Engineering Techblog*, 20-Mar-2018. [Online]. Available: https://engineering.matterport.com/splash-of-color-instance-segmentation-with-mask-r-cnn-and-tensorflow-7c761e238b46. [Accessed: 20-Nov-2018].

[8] A. Akbar, E. Sukmawati, D. Utami, M. Nuriyadi, I. Awaludin, and P. Hidayatullah, "Bull Sperm Motility Measurement Improvement Using Sperm Head Direction Angle," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 4, p. 1642, Aug. 2018.

[9] H.-F. Tsai, J. Gajda, T. F. W. Sloan, A. Rares, and A. Q. Shen, "Usiigaci: Instance-aware cell tracking in stain-free phase contrast microscopy enabled by machine learning," *SoftwareX*, vol. 9, pp. 230–237, Jan. 2019.