AI-Based Detection of Droplets and Bubbles in Digital Microfluidic Biochips

Jianan Xu, Wenjie Fan, Jan Madsen, Georgi Plamenov Tanev, and Luca Pezzarossa
Department of Applied Mathematics and Computer Science
Technical University of Denmark
Kgs. Lyngby, Denmark
Email: [s204698, s210310]@student.dtu.dk, [jama, geta, lpez]@dtu.dk

***Abstract*—Digital microfluidic biochips exploit the electrowetting on dielectric effect to move and manipulate microliter-sized liquid droplets on a planar surface. This technology has the potential to automate and miniaturize biochemical processes, but reliability is often an issue. The droplets may get temporarily stuck or gas bubbles may impede their movement leading to a disruption of the process being executed. However, if the position and size of the droplets and bubbles are known at run-time, these undesired effects can be easily mitigated by the biochip control system. This paper presents an AI-based computer vision solution for real-time detection of droplets and bubbles in DMF biochips and its implementation that supports cloud-based deployment. The detection is based on the YOLOv5 framework in combination with custom pre and post-processing techniques. The YOLOv5 neural network is trained using our own data set consisting of 5115 images. The solution is able to detect droplets and bubbles with real-time speed and high accuracy and to differentiate between them even in the extreme case where bubbles coexist with transparent droplets.**

I. Introduction

Digital microfluidic (DMF) biochips are devices that allow to move and manipulate microliter-sized liquid droplets on a planar surface patterned with individually controllable electrodes. The movement of the droplets, which serve the function of fluidic vehicles and reaction chambers, can be controlled to perform fundamental operations, such as mixing, merging, splitting, dispensing, and disposing, needed to carry out biochemical processes. This technology has the potential to automate and miniaturize traditional laboratory processes and possibly enable advanced biochemical and medical tests to be carried out directly at home or in the field. The droplet movement is achieved by applying an electric potential to the electrodes along the droplet path. Droplets tend to follow this path due to electrowetting on dielectric effect [1].

The actuation and physical process that causes the droplets to move is in general quite reliable. However, sporadic missing-movement events can happen, leading to droplets getting temporarily stuck in one place. The cause for these events can be found in small impurities, imperfections in the surface fabrication, and transient charge build-up in the droplet. Missing-movement events are easily recoverable by just repeating one or more times the same electrodes actuation sequence that was missed. However, in order to perform this correction, the controlling system needs to know that a missing-movement event has happened. In addition to missing-movement events, gas bubbles might form when biochemical processes require heating up a sample. These bubbles get trapped in the region of space where the droplets move and, depending on their dimensions, they might interfere with the movement of the droplets. This effect can be mitigated by the controlling system by repeating one or more times the same actuation sequence to push the bubble out of the path of the droplet, or by re-routing the droplet around the bubble. To achieve this, the controlling system needs to know the position and size of the bubbles.

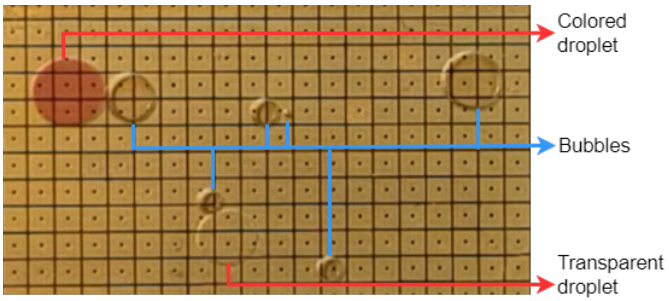


Fig. 1. A top view of a DMF biochip including colored and transparent droplets and bubbles.

This paper presents an AI-based computer vision solution for real-time detection of droplets and bubbles in DMF biochips. The solution is based on the YOLOv5 framework, in combination with pre and post-processing techniques for image preparation and result computation. The YOLOv5 neural network is trained using our own data set consisting of 5115 images. The proposed solution is able to detect droplets and bubbles with real-time recognition speed and high accuracy. One of the most important contributions is the ability to differentiate bubbles from droplets even in extreme cases. For example, the solution is able to properly differentiate a bubble from a transparent droplet, even when they are almost indistinguishable to the human eye.

In addition to the AI-based solution for the detection, the paper also presents an implementation that supports cloud-based deployment for running the computationally-heavy part of the solution on a high-performance remote server. Thus, increasing detection performance and allowing for better maintainability and deployment of new detection models. The entire solution can also be deployed locally.

978-3-9819263-7-8/DATE23/© 2023 EDAA

The rest of this paper is structured in four sections. Section II provides an overview of the DMF and AI technologies used in this work, and includes a summary of related works. Section III presents the developed AI-based detection solution and our data set. Section IV describes the experiment-driven evaluation and discusses the results. Finally, Section V concludes the paper.

II. Background and Related Work

A. DMF Biochip Technology

The target DMF biochip used in this work is the Bioware platform [2]–[4]. However, the proposed solution is also applicable to other state-of-the-art DMF devices such as DropBot [5], OpenDrop [6], and Puddle [7]. The Bioware platform consists of a 32 by 20 regular array of individually controllable square electrodes, which allow to move any droplet along any path on the surface. The platform is designed to support modularity, and sensors and actuators can be integrated into the DMF biochip. For example, color, pH, impedance, etc. can be measured to verify that the biochemical process execution is proceeding as expected or as final process. Actions such as heating, cooling, magnetic field, etc. can be applied to promote transformations as part of the biochemical process.

Figure 2 shows the section drawing of a DMF biochip. The droplets are confined between the substrate consisting of a PCB patterned with the electrodes and an ITO-coated glass top plate. This space is filled with silicon oil and, as previously mentioned, droplets are moved by activating electrodes along the desired path. The figure also shows a heating actuator used to warm up the droplets, the gas bubbles formed as a consequence of the heating process, and a surface defect and an impurity, which are possible causes of missing-movement events.

Experiments have shown that real-life biochemical processes can successfully be carried out on the platform. The fundamental steps of two traditional bioassays have been performed on the platform: PCR-based full cell cloning and magnetic beads-based ELISA with both MRSA and SARS-CoV2 proteins [8]. For these real-life experiments, human supervision is currently necessary to correct sporadic missing-movement events and to avoid bubbles. Thus, remarking the need for an automated feedback system based on the detection solution presented in this paper.

B. The YOLO Framework

The proposed AI-based solution is based on the YOLO framework. YOLO, an acronym for 'you only look once', is a state-of-the-art real-time framework based on convolutional neural networks (CNN) for object detection and classification in images [9]. In YOLO, a single CNN is applied to the entire image. The image is divided into cells and each cell is responsible for detecting objects within itself [10]. The object detection based on the CNN from YOLO [11] delivers low recognition time and high accuracy, making it a suitable choice for our droplet and bubble detection task. In this work, we use YOLOv5 [12].

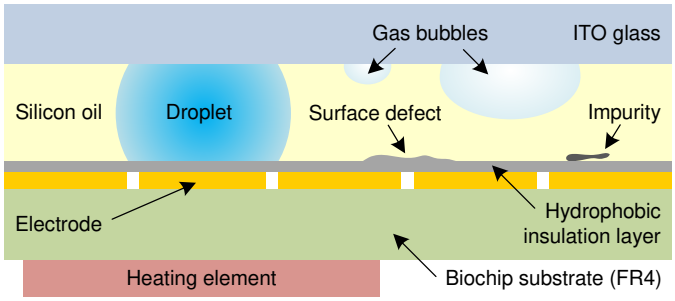


Fig. 2. Section drawing of a DMF biochip showing a droplet and bubbles confined between the substrate and the ITO-coated glass. A surface defect or an impurity (enlarged) are possible causes of missing-movement events.

C. Related Work

An existing method for detecting droplets is to use capacitive sensing techniques to detect the change of the relative dielectric constant between the electrodes and the top plate caused by the presence of a droplet [5], [13]. This method is rather precise, but it struggles to detect bubbles and distinguish them from droplets.

Solutions based on applied computer vision are also used to detect droplets. The work presented in [14] tries to detect droplet motion using real-time background subtraction and color plane extraction methods. The solution is quite fast, taking around 200 ms to detect one droplet. The work presented in [15] combines median filtering, Gauss filtering, the Canny algorithm, and the Hough transform (line and circle detection) for identifying droplets. Their solution also takes around 200 ms to locate a droplet. The work presented in [16] also uses computer vision to monitor droplet movement and calculate the corresponding volume by identifying droplet geometry and performing velocity measurements. None of these works is able to detect bubbles.

The work presented in [17] uses YOLOv3 to track moving droplets within images from fluid dynamic simulations of multi-core emulsions and soft flowing crystals. It proves that the YOLO framework is able to detect droplet-like structures with low error levels and speeds exceeding 30 frames per second when using commonly available desktop GPUs.

III. Detection Solution

A. Overview

The full detection process consists of several phases. At first, a camera produces a video stream of the biochip, from which images are extracted at periodic intervals. These images are then pre-processed and used as input to the YOLOv5 CNN. The YOLOv5 framework produces a list of detected droplets and bubbles including their position and size relative to the image coordinate system. Detected droplets and bubbles characterized by a confidence score lower than a certain threshold are then filtered out and their location and size on the physical biochip are computed using fiducial markers as reference points in the input images. In the following, we explain the details of these phases, the characteristic of the training data set, and we discuss the implemented cloud-based solution.

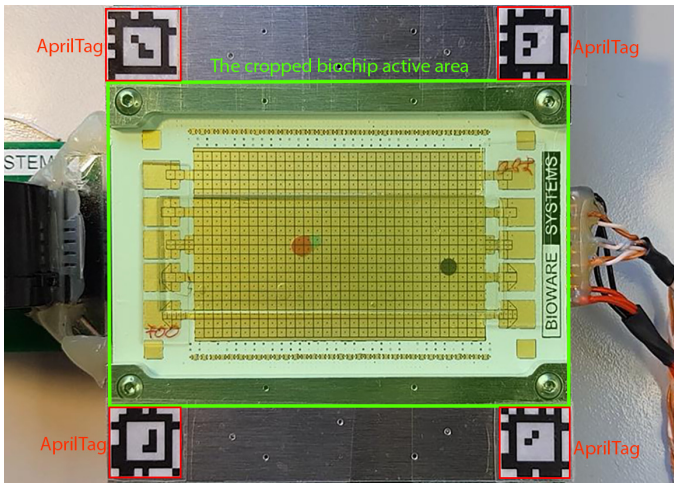


Fig. 3. During the pre-processing phase of the detection process, Apriltags are used to extract (de-warp and crop) the biochip region from the full image.

B. Image Collection and Pre-Processing

The input video stream is recorded using a camera placed directly above the DMF biochip. The camera used in our solution records 30 frames per second with a resolution of 3840 by 2160 pixels and a field-of-view of 67.2° horizontal and 53.0° vertical. Depending on the rate of droplet activity, ten to twenty frames per second are extracted from the videos and saved as images in JPEG format.

A visual fiducial system based on Apritags [18] is used to locate and identify the biochip in images or video streams, as shown in Figure 3. Four numbered Apriltags are placed in known positions on the four corners of the stainless steel frame surrounding the biochip and are used to extract physical reference points in the image. These points are used to de-warp and crop the original image in order to produce images that consistently contain only the biochip active area. In this way, the pixel coordinates of each electrode in a set of images remain the same independently of external disturbances such as camera shack, tilt, or rotation. In addition, it also eliminates the possibility of erroneously identifying objects outside the biochip as droplets. If needed, the images are then scaled to the desired resolution, such as 640 by 360 pixels.

For testing purposes, we trained and used models for RGB images, as well as models for grayscale images. For the latter, an additional grayscale conversion step is carried out during the pre-processing phase. Converting RGB images to grayscale reduces the input dimension leading to faster training and detection [19]. However, this introduces a trade-off between speed and detection accuracy, as shown later in the experimental evaluation.

C. Post-Processing

The results produced by the YOLOv5 framework consist of a list of descriptors of the detected droplets and bubbles. Each descriptor includes the detected class, the coordinates of the center of a bounding box enveloping the detected object, the width and height of the box, and the detection confidence score. The box coordinates and sizes are provided as normalized values against the image width (for x-values) and height (for y-values) of the input image.

At first, the detections are filtered depending on the confidence score to discard false detections. Then, an output image is generated by overlay boxes enveloping the detected objects. Using the AprilTag-based fiducial system and the topology information of the biochip electrodes, we can compute the electrode coordinates on which the center of the droplets lies. For droplets, this information can be used to detect missed-movements events by comparing the detected position with the expected ones. For bubbles, it can be used to divert the path of a droplet or to attempt to push the bubble away.

TABLE I
DATA SET DIVIDED INTO THREE GROUPS. FOR EACH GROUP, THE AMOUNTS OF IMAGES AND LABELED OBJECTS ARE REPORTED.

Data set	Images	Bubbles	Droplets	Hands
D_{close_up}	1069	0	2383	0
D_{full}	4272	0	19642	267
$D_{full\_bubbles}$	843	11397	4860	0

D. Training Data Set

To train the CNN, we have generated a large data set consisting of images created with the pre-processing methodology previously presented. YOLOv5 expects annotations in a text file according to a format similar to the one used in the results, where each line of the file describes a bounding box. We used Roboflow[1], a popular image annotation tool, to label the objects of interest in the images. We labeled every droplet that occupies from half to 22 electrodes (approx. 1 - 44 µl) as "droplet" and every bubble that occupies from a quarter to seven electrodes as "bubble". To test the extensibility of our models, we also added the label "hand" in a subset of images to mark user hands entering the frame to perform operations on the biochip.

Overall, we have divided our data set into three groups, as shown in Table I. The data set D_{close_up} contains close-up images without any pre-processing. This data set was used to train a preliminary test model. The data sets D_{full} and $D_{full\_bubbles}$ contain full-view images with and without bubbles, respectively. These two data sets were used to train the main models. The table also shows the size of the data set groups and the number of labeled objects in each group. Each data set group can be used to train models using RGB or grayscale images since color does not affect the position of droplets and bubbles.

The images of every data set group were randomly divided into the training set, validation set, and test set with the proportion of 7:2:1, respectively. The training set was used for the development of the YOLOv5 model, the validation set was used for evaluating the performance of the model while tuning

[1]Roboflow: https://roboflow.com/

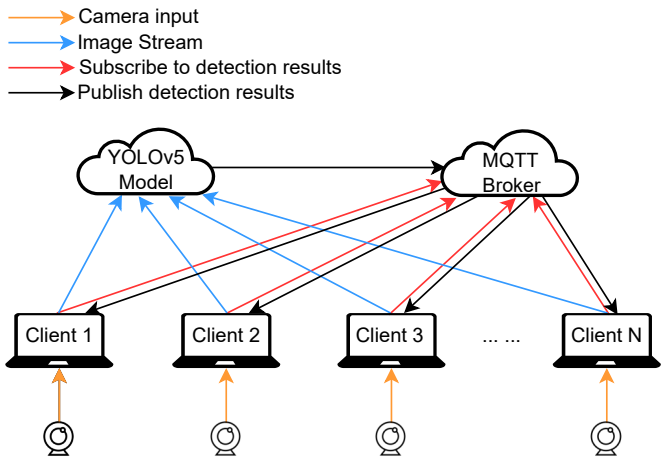


Fig. 4. Diagram of the cloud-based solution with multiple clients. The detection application and the MQTT broker are deployed on two different remote machines.

the hyperparameters of the model, and the test set was used for testing the performance of the final model.

E. Implementation

We structured the implementation of the detection application in three parts: a client, a detection server, and a MQTT broker server. The client contains the logic close to the DMF platform. It manages the camera input and forwards the video stream to the detection server. The detection server performs the image pre-processing, the detection using the YOLOv5 framework, and the post-processing of the result. The results are then sent to an MQTT broker server to which the client can subscribe to access the detection information.

The three parts can be deployed in the same machine or in different ones depending on the use-case requirements. For example, for point-of-care diagnostic applications in regions with limited connectivity, the local deployment could be preferable. On the contrary, for use in research or high-throughput diagnostic, the detection server can be deployed in a remote machine with high-performance GPUs. Remote deployment can also help reduce the cost of the DMF platform due to reduced local computational requirements. In addition, it allows for centralized detection model updates and data collection for further training. Figure 4 shows a diagram of the cloud-based solution where multiple clients use the detection application and the MQTT broker deployed on two different remote machines.

IV. Experimental Evaluation

A. Overview

The proposed solution is evaluated in terms of detection speed, mean average precision (mAP), extensibility, and robustness. In total, eight models were trained and evaluated using the three data sets presented in Table I. Table II reports the results for all the models. The subscript in the models' names are defined as follows:

c : the close-up data set ($\mathrm{D}_{\mathrm{close\_up}}$) is used;
f : the full-view data sets ($\mathrm{D}_{\mathrm{full}}$ and/or $\mathrm{D}_{\mathrm{full\_bubbles}}$) are used;
b : bubbles are included;
h : hands are included;
g : grayscale images are used;
r : RGB images are used.

For example, $\mathrm{M}_{\mathrm{fgb}}$ represents a model trained with full-view grayscale images that contain droplets and bubbles.

At first, we trained the models M_{cg} and M_{cr} as a preliminary test to verify the feasibility of using the YOLOv5 framework for droplet and bubble detection before carrying out the labeling of a large image set. The data set used to train these models is $\mathrm{D}_{\mathrm{close\_up}}$. The two models were trained using the same annotations on grayscale and RGB versions of the images. Results are reported in the first two rows of Table II. The model achieved a high level of accuracy, which justified advancing with further experiments.

The other six rows of the table are dedicated to models trained upon the full-view data sets $\mathrm{D}_{\mathrm{full}}$ and $\mathrm{D}_{\mathrm{full\_bubbles}}$. Each model was trained with the grayscale and RGB versions of the images. M_{fg} and M_{fr} were trained using the data set $\mathrm{D}_{\mathrm{full}}$, but excluding the labeled hands, while $\mathrm{M}_{\mathrm{fgh}}$ and $\mathrm{M}_{\mathrm{frh}}$ were trained using the same data set including hands to test extensibility. Finally, $\mathrm{M}_{\mathrm{fgb}}$ and $\mathrm{M}_{\mathrm{frb}}$ include bubbles and were trained using the $\mathrm{D}_{\mathrm{full}}$ and $\mathrm{D}_{\mathrm{full\_bubbles}}$ data sets.

B. Detection Speed

To evaluate the detection speed, we executed tests for all the models on the corresponding validation set targeting a 'Tesla V100 PCIE 16GB' GPU and a '12$^{\mathrm{th}}$ Gen. Intel Core i7-12650H 2.30 GHz' CPU. Table II shows the detection time for all the models.

By running trained models with the GPU and comparing every grayscale model and its corresponding RGB model, we can notice that the grayscale versions tend to have a smaller detection time due to the smaller data size associated with each pixel. The speed of the proposed solution is overall high, with a total detection time smaller than 2.7 ms for models containing a single class and smaller than 8.1 ms for models containing multiple classes.

When running the models on the CPU (personal computer without GPU support), the total detection time is smaller than 76 ms for models containing a single class and smaller than 83 ms for models containing multiple classes. These detection speeds are more than sufficient for practical applications. In this case, grayscale models do not always perform faster than RGB models, even if they appear to perform better when deployed on a GPU.

C. Accuracy (mAP)

To evaluate the detection accuracy, we use the average precision (AP) and mean average precision (mAP) metrics.Average precision (AP) estimates the area under the curve (AUC) of the precision vs. recall relationship [20]. The AP@0.5 means that the AP is calculated using an intersection over union (IoU) threshold of 0.5, where IoU indicates the overlap of the predicted and the ground truth bounding boxes. The AP@.5:.95

TABLE II
PERFORMANCE RESULTS FOR ALL TRAINED MODELS.

						Detecting time (ms)	
Model name	**Color mode**	**Class name**	**AP@.5**	**AP@.5:.95**	**mAP@.5:.95**	Intel Core i7	Tesla v100
$\mathrm{M_{cg}}$	Grayscale	Droplet	0.995	0.938	0.938	72.4	2.5
$\mathrm{M_{cr}}$	RGB	Droplet	0.993	0.874	0.874	74.3	2.7
$\mathrm{M_{fg}}$	Grayscale	Droplet	0.995	0.921	0.921	73.2	2.4
$\mathrm{M_{fr}}$	RGB	Droplet	0.995	0.920	0.920	76.0	2.7
$\mathrm{M_{fgh}}$	Grayscale	Droplet	0.995	0.921	0.838	74.1	2.5
		Hand	0.935	0.755			
$\mathrm{M_{frh}}$	RGB	Droplet	0.995	0.921	0.861	74.4	2.5
		Hand	0.960	0.801			
$\mathrm{M_{fgb}}$	Grayscale	Droplet	0.995	0.783	0.706	83.0	6.5
		Bubble	0.974	0.630			
$\mathrm{M_{frb}}$	RGB	Droplet	0.994	0.811	0.724	81.4	8.1
		Bubble	0.923	0.637			

is the average AP over ten IoU thresholds from 0.5 to 0.95 with a step of 0.05. The mAP for object detection is the mean of the APs over all object classes.

Table II reports the AP@.5 and AP@.5:.95 metrics for all the classes in each model. All the AP@.5 are higher than 0.923 and, for the droplet class, they are higher than 0.993. The AP@.5:.95 is higher than 0.874 for the models detecting the single droplet class. However, for models containing two classes, the AP@.5:.95 is lower, reaching a minimum of 0.630. This means that the models are overall very good at detecting the presence of droplets and bubbles, but the accuracy decreases as the IoU threshold approaches 0.95. This is particularly true for the $\mathrm{M_{fgb}}$ model. In other words, if the IoU threshold turns higher, detections with lower IoU values will be considered as false. Thus, reducing the AP value. Nevertheless, the mPAs for all models are higher than 0.706 and experiments have shown that the achieved accuracy is more than sufficient for practical applications in the DMF context.

Comparing the mAP@.5:.95 of RGB models and grayscale models, we can observe that grayscale models can detect droplets better when only the droplet class needs to be detected. However, if we need to recognize both droplets and bubbles, or droplets and hands, the RGB model performs better. This shows that the information on distinguishing features of bubbles is partially contained in the color channels.

D. Extensibility

To verify that our model can be extended when more classes need to be detected, we introduced hand detection in the existing models $\mathrm{M_{fg}}$ and $\mathrm{M_{fr}}$ to get $\mathrm{M_{fgh}}$ and $\mathrm{M_{frh}}$, respectively. The mAP@.5:.95 of both extended models is higher than 0.838. This confirms that it is possible to extend our models to introduce more objects, such as pipettes and tweezers.

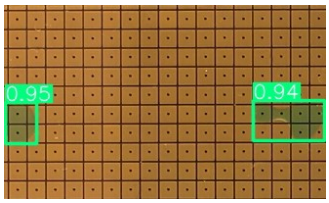
E. Robustness

As previously stated, external disturbances from the camera and objects outside the biochip can be effectively eradicated by pre-processing. An ideal detector should be insusceptible to the intra-class variation such as different shapes, sizes, and colors of droplets, while maintaining sensitivity to the inter-class variations (droplets and bubbles). The robustness of our system is measured by the ability to accurately detect and distinguish droplets from bubbles, especially for transparent droplets which are almost invisible and indistinguishable from bubbles to the human eye.

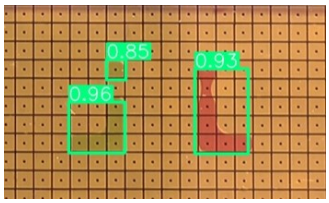
To demonstrate robustness, we tested the $\mathrm{M_{frb}}$ model with images containing extreme cases. Figure 5 shows a selection of detection results. Detection videos are also available[2,3]. Droplets are enclosed in green rectangles, while bubbles are enclosed in red rectangles. In the figure, we can observe that the model can detect droplets regardless of their colors, also including transparent ones.

The model also succeeded in detecting merging droplets as shown in Figures 5e and 5h, even when the two droplets do not have the same color. Moreover, transparent droplets in

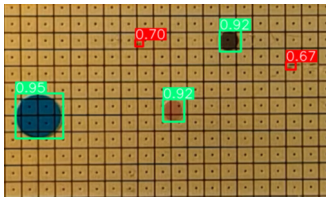
[2]Droplets and bubbles: https://tinyurl.com/3acfa723
[3]Droplets with different colors and shapes: https://tinyurl.com/bdcnrw2c



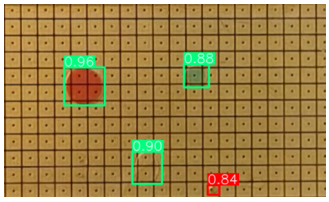
(a) Green droplets. Left one is splitting.



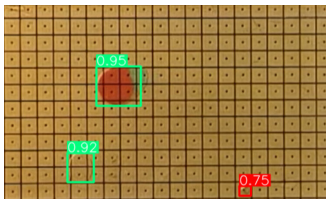
(b) Green droplet has split. Red droplet is splitting.



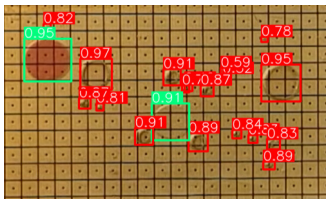
(c) Colored droplets. Few small bubbles.



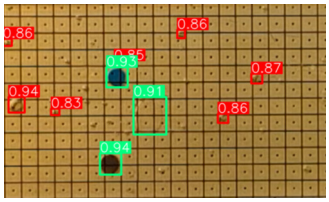
(d) Colored and transparent droplets.



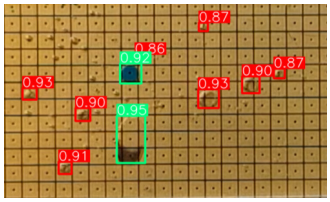
(e) Red and green droplets are merging.



(f) Red and transparent droplets. large and small bubbles.



(g) Blue, black and transparent droplets. Small bubbles.



(h) Blue droplet. Black and transparent droplets are merging. Small bubbles.

Fig. 5. A selection of detection results.

Figures 5d–5g were distinguished from bubbles as expected. Figures 5f–5h show a complex biochip configuration with numerous droplets and bubbles. Also in this case, the model can detect droplets and bubbles with almost all confidence scores above 0.8. In Figures 5g and 5h, some minuscule bubbles (approx. less than 0.5 µl) were not identified. This is an expected behavior since they were not labeled in our data set. These minuscule bubbles do not pose a threat to droplets and can be easily pushed away by droplets without interfering with their motion.

V. Conclusion

In this paper, we presented a solution for real-time detection of droplets and bubbles in DMF biochips based on the YOLOv5 framework in combination with specialized pre and post-processing techniques. Our own data set consisting of 5115 images containing a total of 24502 labeled droplets and 11397 labeled bubbles was used to train six main models. We also presented an implementation to support cloud-based and local deployment of the solution. The solution was evaluated in terms of speed, accuracy, extensibility, and robustness, showing excellent results. In summary, the contributions of this paper are: (1) the AI-based solution able to detect and distinguish bubbles and droplets, (2) the versatile solution for cloud-based and local deployment, and (3) the large labeled data set.

References

[1] F. Mugele and J.-C. Baret, "Electrowetting: From basics to applications," *Journal of Physics: Condensed Matter*, vol. 17, no. 28, pp. 705–774, 2005.

[2] G. P. Tanev, L. Pezzarossa, and J. Madsen, "The bioware cyber-fluidic platform: A holistic approach to digital microfluidics," in *Proc. of the 13th International Workshop on Bio-Design Automation.* ACM, 2021.

[3] G. P. Tanev, L. Pezzarossa, W. E. Svendsen, and J. Madsen, "A reconfigurable digital microfluidics platform," in *Proc. of the 11th International Workshop on Bio-Design Automation.* ACM, 2019.

[4] ——, "A modular reconfigurable digital microfluidics platform," in *Proc. of the 19th Symposium on Design, Test, Integration & Packaging of MEMS and MOEMS.* IEEE, 2018, pp. 1–6.

[5] R. Fobel, C. Fobel, and A. R. Wheeler, "DropBot: An open-source digital microfluidic control system with precise control of electrostatic driving force and instantaneous drop velocity measurement," *Applied Physics Letters*, vol. 102, no. 19, pp. 1129–1132, 2013.

[6] M. Alistar and U. Gaudenz, "OpenDrop: An integrated do-it-yourself platform for personal use of biochips," *Journal of Bioengineering*, vol. 4, no. 4, p. 45, 2017.

[7] M. Willsey, A. P. Stephenson, C. Takahashi, P. Vaid, B. H. Nguyen, M. Piszczek, C. Betts, S. Newman, S. Joshi, K. Strauss, and L. Ceze, "Puddle: A dynamic, error-correcting, full-stack microfluidics platform," in *Proc. of the 24th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2019, pp. 183–197.

[8] G. P. Tanev, "A modular design approach for programmable cyber-fluidic systems," Ph.D. dissertation, Technical University of Denmark, 2021.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. of Advances in Neural Information Processing Systems*, vol. 25. Curran Associates, Inc., 2012.

[10] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[12] G. Jocher et al., "ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple M1, Reproducibility, ClearML and Deci.ai integrations," Aug. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7002879

[13] J. Gong and C. J. Kim, "All-electronic droplet generation on-chip with real-time feedback control for EWOD digital microfluidics," *Lab on a Chip*, vol. 8, no. 6, pp. 898–906, 2008.

[14] Y.-J. Shin and J.-B. Lee, "Machine vision for digital microfluidics," *Review of Scientific Instruments*, vol. 81, no. 1, p. 014302, 2010.

[15] Z. Luo, B. Huang, J. Xu, L. Wang, Z. Huang, L. Cao, and S. Liu, "Machine vision-based driving and feedback scheme for digital microfluidics system," *Open Chemistry*, vol. 19, no. 1, pp. 665–677, 2021.

[16] P. Q. Vo, M. C. Husser, F. Ahmadi, H. Sinha, and S. C. Shih, "Image-based feedback and analysis system for digital microfluidics," *Lab on a Chip*, vol. 17, no. 20, pp. 3437–3446, 2017.

[17] M. Durve, F. Bonaccorso, A. Montessori, M. Lauricella, A. Tiribocchi, and S. Succi, "Tracking droplets in soft granular flows with deep learning techniques," *The European Physical Journal Plus*, vol. 136, no. 8, p. 864, 2021.

[18] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2011, pp. 3400–3407.

[19] P. Kaler, "Study of grayscale image in image processing," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 4, no. 11, pp. 309–311, Nov.2016.

[20] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *Proc. of the International Conference on Systems, Signals and Image Processing*, 2020.