Chinese Journal of Chemical Engineering 87 (2025) 103–114

Contents lists available at ScienceDirect

Chinese Journal of Chemical Engineering

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

Full Length Article

MicroFlowSAM: A motion-prompted instance segmentation approach in microfluidics with zero annotation and training

Wenle Xu [1, 2], Lin Sheng [1, 2], Tong Qiu [1, 2, *], Kai Wang [1, 2], Guangsheng Luo [1, 2]

[1] Department of Chemical Engineering, Tsinghua University, Beijing 100084, China
[2] State Key Laboratory of Chemical Engineering and Low-Carbon Technology, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 13 February 2025
Received in revised form
18 May 2025
Accepted 20 May 2025
Available online 30 June 2025

Keywords:
Microfluidics
Microdispersion
Instance segmentation
Large vision model
Prompt engineering

ABSTRACT

Microdispersion technology is crucial for a variety of applications in both the chemical and biomedical fields. The precise and rapid characterization of microdroplets and microbubbles is essential for research as well as for optimizing and controlling industrial processes. Traditional methods often rely on time-consuming manual analysis. Although some deep learning-based computer vision methods have been proposed for automated identification and characterization, these approaches often rely on supervised learning, which requires labeled data for model training. This dependency on labeled data can be time-consuming and expensive, especially when working with large and complex datasets. To address these challenges, we propose MicroFlowSAM, an innovative, motion-prompted, annotation-free, and training-free instance segmentation approach. By utilizing motion of microdroplets and microbubbles as prompts, our method directs large-scale vision models to perform accurate instance segmentation without the need for annotated data or model training. This approach eliminates the need for human intervention in data labeling and reduces computational costs, significantly streamlining the data analysis process. We demonstrate the effectiveness of MicroFlowSAM across 12 diverse datasets, achieving outstanding segmentation results that are competitive with traditional methods. This novel approach not only accelerates the analysis process but also establishes a foundation for efficient process control and optimization in microfluidic applications. MicroFlowSAM represents a breakthrough in reducing the complexities and resource demands of instance segmentation, enabling faster insights and advancements in the microdispersion field.

1. Introduction

Microfluidic technology has gained considerable interest over the past two decades owing to its superior capabilities in mass and heat transfer, controllability, alongside inherent safety [1]. Microdispersion technology is one of the most important techniques, due to its potential applications in various chemical and biomedical fields such as nanoparticle synthesis, single-cell analysis, drug delivery, and carbon dioxide capture [2–5]. Extensive experiments are being conducted to elucidate the mass transfer, heat transfer, and momentum transfer processes, as well as the reaction processes of microdroplets and microbubbles in liquid-liquid and gas-liquid multiphase systems [6,7]. After the experiments, it is essential to obtain detailed information of the microdroplets and microbubbles. Furthermore, as microdispersion technology advances from laboratory settings to industrial applications, the speed and accuracy of microdroplets and microbubbles characterization become critical for the effective control and optimization of production processes [1].

Deep learning-based computer vision technology enables rapid detection and segmentation of microdroplets and microbubbles, offering a powerful tool for the characterization. A convolutional neural network (CNN)-based classification model is developed to distinguish different flow patterns in microchannel liquid/liquid flow, achieving remarkable accuracy exceeding 98% [8]. This model is further integrated with an automatic pump control system, facilitating high-throughput experiments without manual intervention. Segfit [9] combines the Mask R-CNN [10] instance

This article is part of a special issue entitled: Microchemical technology published in Chinese Journal of Chemical Engineering.

* Corresponding author. Department of Chemical Engineering 352, Tsinghua University, Beijing 100084, China.
E-mail address: qiutong@tsinghua.edu.cn (T. Qiu).

https://doi.org/10.1016/j.cjche.2025.05.023

segmentation model with a shape-fitting algorithm. It demonstrates outstanding performance metrics, including a diameter mean absolute error of 0.75 μm, a mean average precision of 93.6%, and a mean average recall of 95.3%. Remarkably, Segfit operates at speeds 1000 times faster than conventional manual methods while maintaining precision comparable to humans. Additionally, transfer learning is applied to enhance image processing in microfluidic research, significantly reducing the need for extensive training datasets and manual annotation efforts by approximately 90% [11]. Despite these advancements, a notable limitation of these methods is their reliance on training specific models tailored to individual datasets, which necessitates substantial data collection and annotation efforts. Moreover, training new models consumes considerable computational resources and time.

Large language models pretrained on extensive web-scale datasets are showcasing impressive zero-shot generalization abilities. By providing these models with adequate task descriptions through natural language prompts, they demonstrate robust performance across diverse applications [12]. This capability extends to computer vision as well, where OpenAI utilizes contrastive learning techniques to enhance models' ability to learn associations between images and text [13]. The pre-trained model shows strong zero-shot generalization capabilities across 30 different downstream computer vision tasks. Meta AI proposes the Segment Anything Model (SAM), a notable advancement in image segmentation [14]. SAM can generate high-quality segmentation masks based on prompts, demonstrating surprising generalization capabilities across various image datasets [14,15].

Motivated by advancements in promptable segmentation models, we propose MicroFlowSAM, a pioneering method tailored for the analysis of microfluidic videos. By tracking the movement of microdroplets or microbubbles, our method automatically determines their coordinates and uses this information as prompts for the state-of-the-art (SOTA) pre-trained segmentation model, SAM. This innovative prompting method enables SAM to accurately segment microdroplets or microbubbles within these videos, thus providing rapid and precise instance segmentation without the need of manual annotation or extensive training. MicroFlowSAM demonstrates robust performance comparable to supervised methods across our 12 datasets. This method eliminates the workload of manual annotation and the computational resource demands of the training process, significantly accelerating the analysis of microfluidic videos.

2. Data Preparation

We collect 12 videos from various experimental conditions and microchannel devices, documenting the generation and flow processes of microdroplets and microbubbles. To comprehensively evaluate the effectiveness and general applicability of our method, we randomly select 10 frames of each video and manually annotated the microdroplets and microbubbles for test. The original frame image is shown in Fig. 1. The resolution of each dataset, along with the average number, area, circularity, and flow velocity of the microdroplets or microbubbles, is detailed in Table 1. Evidently, the microdroplets and microbubbles in the datasets exhibit significant variation in size, shape, color, and speed, which presents a challenge to the generalizability of instance segmentation algorithms. In addition, considering that the overlap between bubbles or droplets in images poses challenges for recognition tasks, we also calculated the overlap ratio as a metric. This metric measures the average proportion of the overlapping area of each bubble or droplet relative to the total area of that bubble or droplet in each dataset.

3. Methods

3.1. Overview

As depicted in Fig. 2, the characterization of microdroplets or microbubbles based on computer vision comprises two primary steps: instance segmentation and characteristics computation.

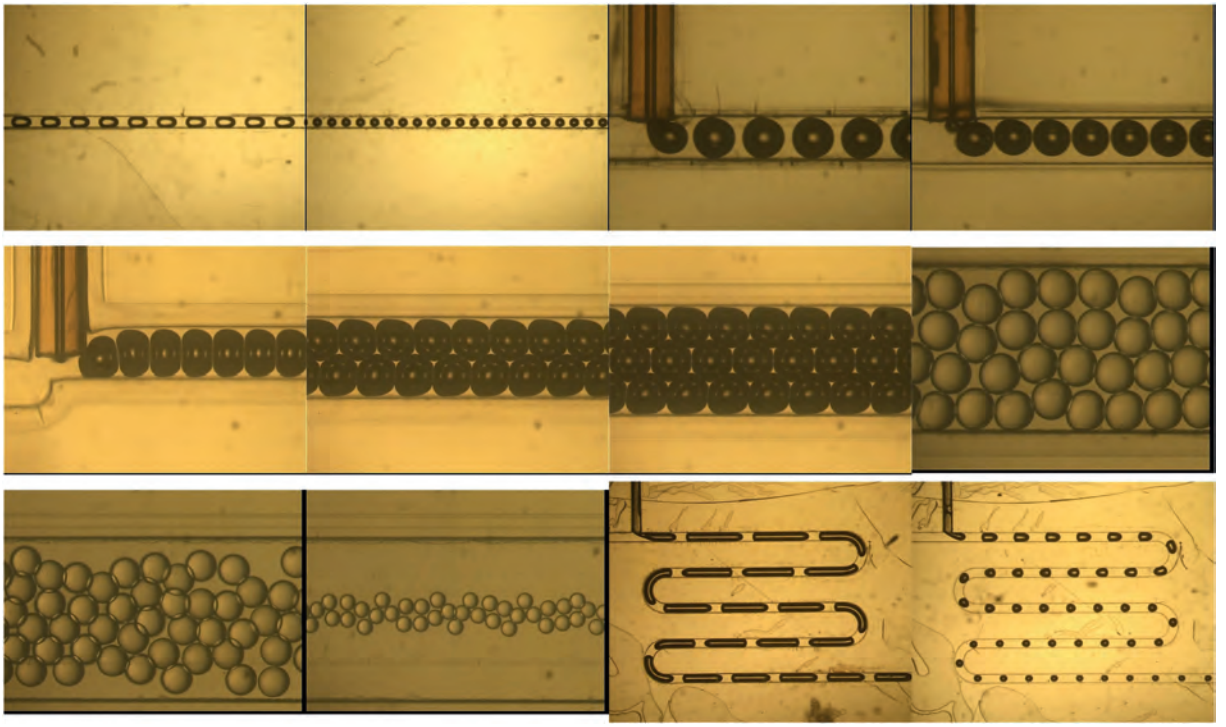

Fig. 1. The original frame image of each dataset (the smaller images are stretched to match the size of the largest one). From left to right and top to bottom, the images correspond to Datasets 1 through 12, respectively.

Table 1
The resolution of each dataset, along with the average number, area, circularity, and flow velocity of the microdroplets or bubbles.

Dataset	Resolution /pixel^2	Average number	Average area /pixel^2	Average circularity	Average velocity /pixel·frame^{-1}	Average overlap ratio
1	1072 × 800	10	2456	0.83	3.96	0
2	1072 × 800	21	857	0.90	7.12	0
3	1072 × 800	4	16907	0.95	30.77	0
4	1072 × 800	5	13517	0.93	34.52	0
5	912 × 688	5	11433	0.85	1.65	0
6	912 × 688	14	11748	0.88	1.67	0
7	912 × 688	19	11633	0.88	1.39	0
8	816 × 688	28	9024	0.90	0.91	0
9	816 × 688	40	5231	0.88	1.27	0.03
10	816 × 688	35	1333	0.89	3.51	0.002
11	1280 × 688	20	7282	0.46	8.38	0
12	1280 × 688	44	1248	0.87	2.05	0

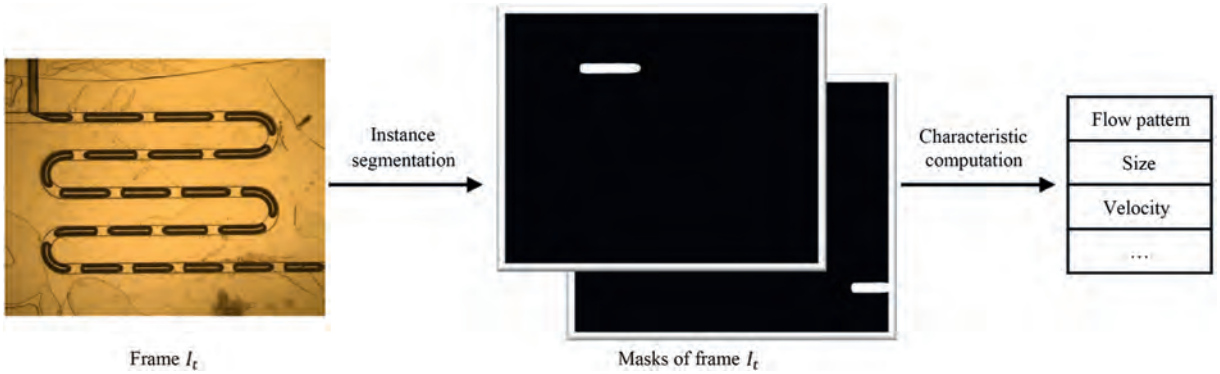

Fig. 2. The pipeline for characterizing microdroplets and microbubbles using instance segmentation, comprising 2 main steps: instance segmentation and characteristic computation.

The key step of this process is instance segmentation, which is a fundamental task in computer vision. When applied in the field of microfluidics, every individual microdroplet or microbubble is treated as a distinct "instance". The target outcome for each instance is represented by a binary matrix referred to as a mask, which precisely delineates the coordinates of every pixel belonging to the microdroplet or microbubble. Based on these masks, straightforward calculations can be performed to determine crucial flow characteristics of interest, such as the sizes, flow patterns, and velocities of the microdroplets or microbubbles.

It is worth noting that while several previous studies have reported on instance segmentation in the field of microfluidics, these methods traditionally rely on supervised methods. They necessitate manual annotation of data, model training, and the retraining of a new model whenever the dataset changes. To address these challenges, we propose MicroFlowSAM, a method that eliminates the need for manual annotation or training. The detailed steps of the method are shown in Fig. 3. MicroFlowSAM consists of three main components: a prompt generator, a pretrained image segmentation model (SAM), and a post-processor. The prompt generator operates by tracking the movement of microdroplets or microbubbles from frame $I_{t-\text{interval}}$ to I_t, generating coordinate points that indicate their positions. These coordinates are then utilized by SAM as input prompts for precise instance segmentation in the current frame. Following segmentation, a post-processor is implemented to handle specific tasks such as removing redundant masks using non-maximum suppression algorithm and filling holes in the masks.

3.2. Segment anything model

SAM is a foundational model in the field of computer vision, specifically focus on prompt-based segmentation tasks. It benefits significantly from pre-training on a large dataset (SA-1B), which includes 11 million high-resolution images paired with 1 billion high-quality segmentation masks. This dataset spans diverse domains, covering various object types, scenes, and visual contexts. This extensive pre-training enables SAM to learn robust and generalizable feature representations, thereby enhancing its capability for zero-shot generalization. Additionally, SAM employs a prompt-based approach to segmenting images. Despite its training data lacking images from the microfluidics domain, this methodology allows SAM to effectively utilize the prompts provided for accurate segmentation of microdroplets and microbubbles.

SAM is structured with three key components, as depicted in Fig. 4: an image encoder, a prompt encoder, and a mask decoder. The image encoder is tasked with mapping the input image into a feature space. The prompt encoder handles input prompts such as points, bounding boxes, low-resolution masks, and text, mapping them into a prompt feature space. Finally, the mask decoder integrates information from both the image and the prompt to generate accurate segmentation masks. Detailed descriptions of the architecture and functionality of each component are provided in the following sections.

3.2.1. Image encoder

The transformer architecture provides significant advantages over traditional CNNs in the field of image processing, particularly for tasks involving large-scale datasets and complex relationship understanding [16]. Transformers excel at modeling long-range dependencies within image and utilize attention mechanisms to flexibly adjust attention weights across different parts of the input. Additionally, transformers offer superior scalability, which is crucial for managing the escalating complexity and scale of contemporary image datasets and models.

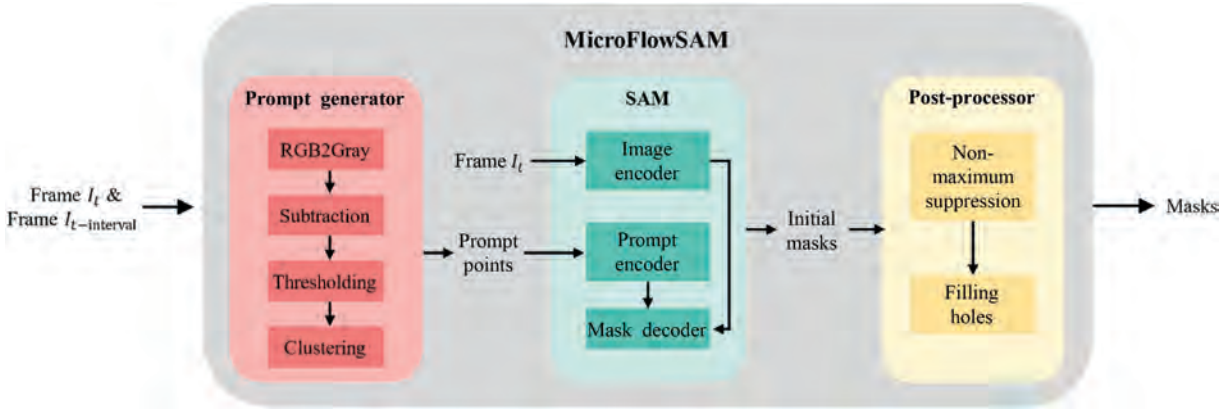

Fig. 3. An overview of MicroFlowSAM. MicroFlowSAM consists of three main components: a prompt generator, a pretrained image segmentation model (SAM), and a post-processor.

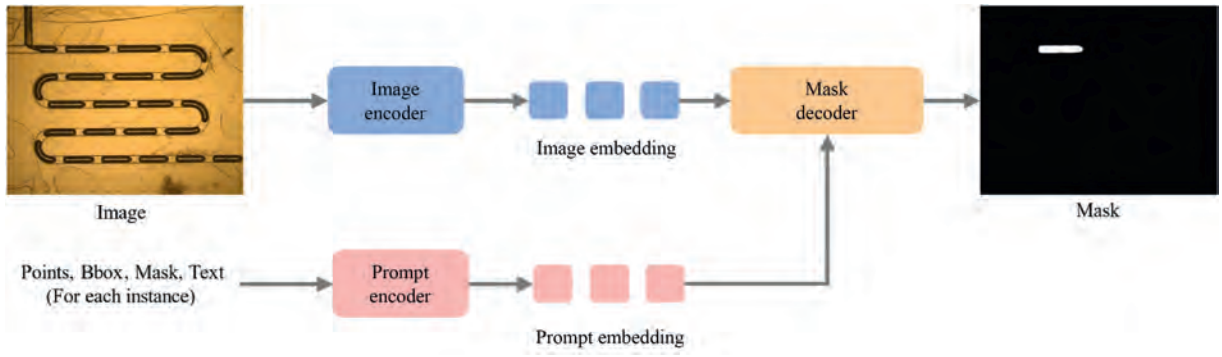

Fig. 4. The Segment Anything Model (SAM) overview. SAM consists of three components: a heavyweight image encoder generates an image embedding, a prompt encoder capable of handling various types of prompts (points, bounding boxes, masks, or text), and a mask decoder that generates the final output.

To achieve these capabilities, SAM' s image encoder also adopts the vision transformer (ViT) architecture [17]. In transformers, attention mechanisms are used to model relationships between tokens, which, in natural language processing, correspond to individual words. However, in the domain of image processing, treating each pixel as a token is impractical due to the vast number of pixels in an image, which would introduce excessive spatial and temporal complexity. Consequently, the ViT divides an image into smaller patches (typically 16 × 16 pixels), treating each patch as a distinct token. The self-attention mechanisms then enable each token to adjust its representation based on information from other patches within the image, effectively capturing global contextual information.

The ViT architecture offers flexibility in model size and computational efficiency by varying the number of layers. Three versions of ViT have been released: ViT-H with 636 million parameters, ViT-L with 308 million parameters, and ViT-B with 91 million parameters. This allows for a trade-off between model performance and computational resources, ensuring adaptability to different application scenarios in image processing.

3.2.2. Prompt encoder

One notable characteristic of SAM is its capability to utilize prompts for guiding segmentation, which is essential for its zero-shot generalization. SAM can process various types of input prompts, including points, bounding boxes, low-resolution masks, and text. The prompt encoder's role is to convert these prompts into embeddings that the model can interpret and use for effective segmentation.

In the prompt encoder, different strategies are employed based on the complexity of the input prompts. For spatial prompts such as points and bounding boxes, which typically consist of lower-dimensional information, a single-layer Gaussian Fourier feature mapping is utilized to convert them into matrices [18]. This processing helps prevent overfitting while efficiently capturing spatial information. For low-resolution masks, which have slightly higher dimensions, SAM utilizes a three-layer CNN for encoding. Additionally, SAM can also incorporate text inputs encoded by models like CLIP [13], extending its capability to interpret textual information for segmentation tasks. This versatility in handling different types of inputs highlights SAM's adaptability and effectiveness in diverse applications in image segmentation tasks.

3.2.3. Mask decoder

The mask decoder incorporates a transformer structure to process embeddings derived from both the image and prompts, facilitating the generation of masks. This component employs three distinct types of attention mechanisms. Self-attention operates among the prompt embeddings, enabling the decoder to focus on relationships within the set of prompts. Cross-attention is used to process interactions between prompts and the image, with prompts serving as queries and the image as keys. Additionally, another variant of cross-attention reverses this relationship, with prompts acting as keys and the image as queries. These mechanisms enable the mask decoder to iteratively integrate information from images and prompts, thereby enhancing its ability to generate accurate masks that precisely delineate objects within the image.

3.3. Prompts generator

Prompts are crucial for fully leveraging SAM's zero-shot generalization capabilities in our microdroplet and microbubble segmentation tasks. When no prompt is provided, SAM operates in Automatic Mask Generation mode. In this mode, a grid of evenly distributed prompt points is generated over the image and fed into the model. The resulting masks are then filtered based on metrics such as confidence and stability scores. This method is tentatively applied to our dataset for instance segmentation, as shown in Fig. 5. The results indicate that instance segmentation is highly inaccurate, with many masks not corresponding to microdroplets or microbubbles. Additionally, the segmentation results for many microdroplets or microbubbles are fragmented into two or more parts, rendering them unsuitable for subsequent characteristics calculations.

While higher-dimensional prompts, such as bounding boxes, have greater potential to enhance the performance of instance segmentation, obtaining such spatial information poses significant challenges [14]. Typically, this requires either manual annotation of objects within images or training of a supervised object detection model to generate bounding boxes for all microdroplets and microbubbles. These methods require human intervention, which contradicts our intention of eliminating manual labor. However, a deeper exploration of the microfluidic dataset reveals a notable distinguishing feature of microdroplets and microbubbles from the background is that they are all in motion. Consequently, this motion information can be leveraged to generate prompts that guide SAM in performing instance segmentation.

To identify the motion of microdroplets and microbubbles, a classical algorithm in computer vision domain is the frame difference method [19,20]. This algorithm leverages the color contrast between the dispersion phase and the continuous phase. As microdroplets or microbubbles move, pixel values between frames change due to this contrast, enabling precise determination of each microdroplet's or microbubble's position. The process of the prompt-generating algorithm is illustrated in Fig. 6.

First, a pair of images of a previous frame $I_{t-\text{interval}}$ and a current frame I_t are converted to grayscale. And the interval being a tunable hyperparameter that should be increased when the velocity of microdroplets or microbubbles is relatively low. The difference between these grayscale images is then calculated and binarized.

As shown in Fig. 7, in the grayscale images, each microdroplet or microbubble is characterized by a smaller, light-colored central area with higher grayscale values and a dark-colored outer region with lower grayscale values. The grayscale value of the background is also higher than that of the dark-colored regions. In the direction of flow, the front region of each microdroplet or microbubble transitions from the background to the outer region of the microdroplet or microbubble. Therefore, when performing Frame $I_{t-\text{interval}}$ - Frame I_t subtraction, the front region exhibits higher grayscale values, which appear as larger white areas in the binarized images. Additionally, parts of the outer dark region of the microdroplet or microbubble may be replaced by the lighter central area, creating smaller white areas in the binarized images. Furthermore, noise in the grayscale images can contribute to the formation of additional small white areas in the binarized images.

Redundant regions are undesirable as they may generate multiple prompts for a single microdroplet or microbubble, potentially affecting the model's inference speed, accuracy, and the subsequent counting of microdroplets or microbubbles. To mitigate this issue, we apply the *K*-means clustering algorithm based on region area to reduce noise and eliminate redundant regions. The algorithm clusters the regions into two groups, with larger regions being identified as potential candidates. The centroid of each potential region is then extracted as the final prompt point. However, in the case shown in Fig. 7(c), each microdroplet or microbubble may correspond to multiple large white areas, posing a challenge for straightforward clustering to eliminate redundant prompts. Given the difficulty in reliably determining if prompts belong to the same microdroplet or microbubble, redundant prompt points are accepted as inputs for SAM. And the issue of redundancy will be addressed in the post-processing stage.

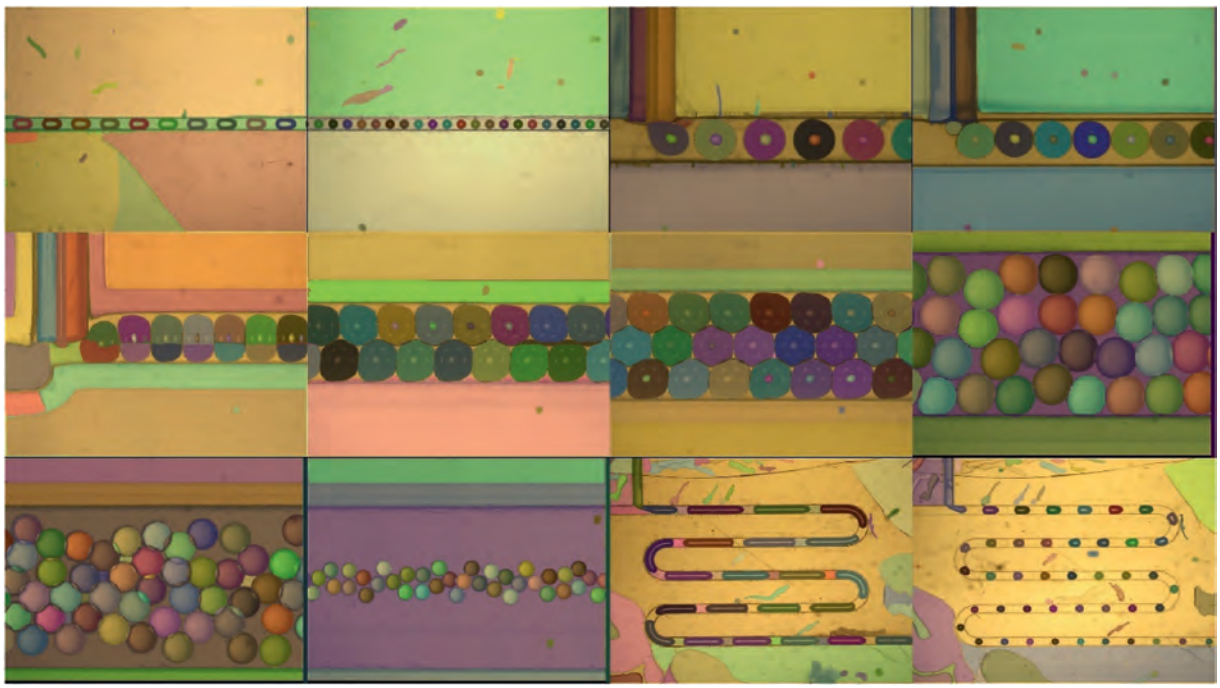

Fig. 5. Instance segmentation results of SAM without prompts.

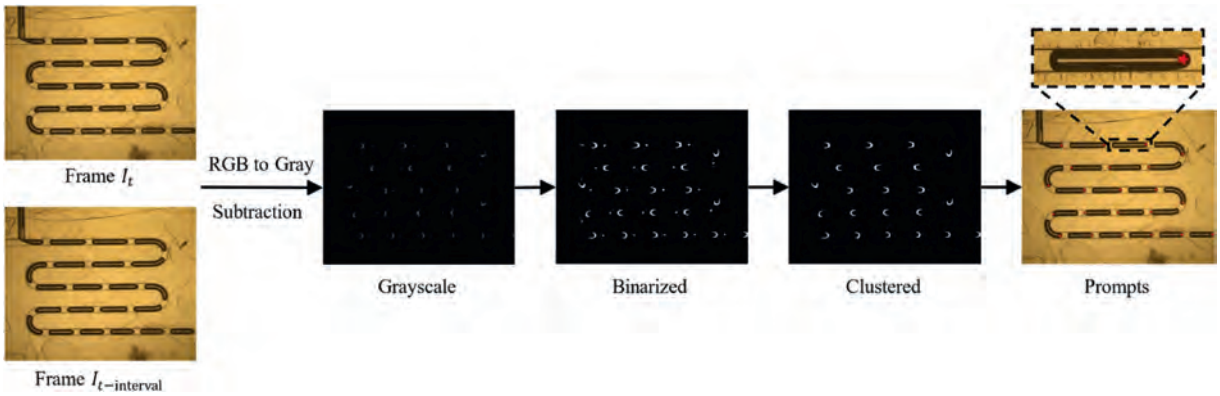

Fig. 6. An illustration of the generation of prompts. By processing two frames through grayscale conversion, subtraction, binarization, and clustering, we extract candidate regions. Finally, the center of the candidate regions is selected as the prompt point.

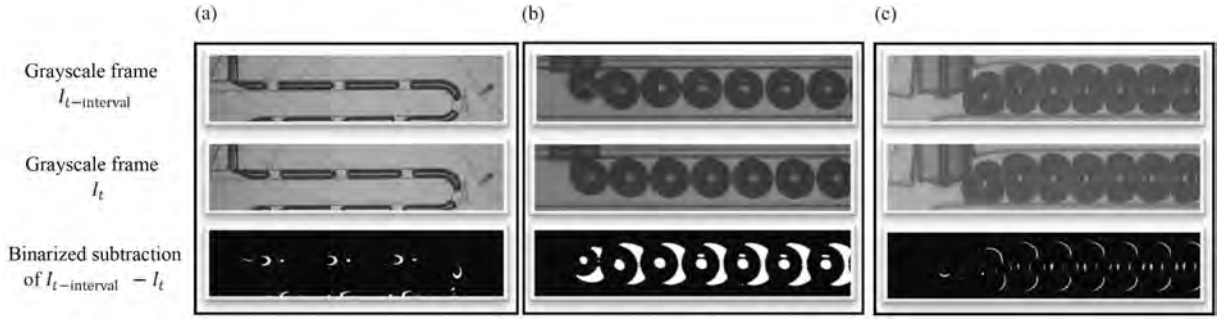

Fig. 7. Examples of the grayscale images for frame $I_{t-\text{interval}}$ and frame I_t, along with the corresponding image obtained through differencing and binarization.

3.4. Post-processor

The post-processor primarily accomplishes two tasks: removing redundant masks of each instance and filling holes in the mask. Once masks are obtained, the non-maximum suppression (NMS) algorithm is employed to address the redundancy issue [21]. The NMS algorithm evaluates the intersection over union (IoU) of different masks to determine if they represent the same instance. If the IoU exceeds a certain threshold, these masks are considered as belonging to a same instance. And only the mask with the highest segmentation confidence is retained, while the others are removed. Additionally, microdroplets and microbubbles located at the image periphery or in the process of formation are filtered out to prevent the subsequent computation of statistics such as size.

In addition to this issue, the inferred masks from MicroFlowSAM occasionally contain holes, as depicted in Fig. 8. The connected component analysis algorithm is used to identify all connected foreground regions, and the holes contained within these connected regions are then filled [22].

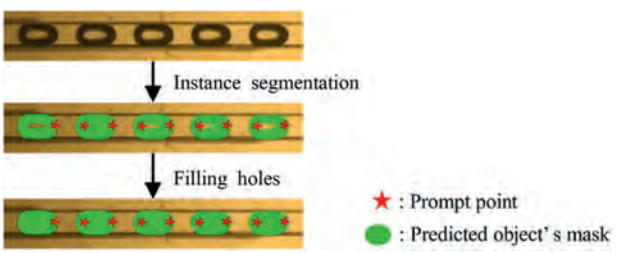

Fig. 8. Examples of the instance segmentation results from Dataset 1, showing masks before and after hole filling.

4. Results and Discussions

4.1. Evaluation criterions

In our algorithm, the quality of the prompts directly influences the subsequent instance segmentation performance. Therefore, we evaluate our prompt generation algorithm using precision (prompt) and recall (prompt) metrics:

$$\text{Precision (prompt)} = \frac{\sum_1^{N_{\text{prompt}}} 1_{\text{if the prompt point is within any GT mask}}}{N_{\text{prompts}}} \tag{1}$$

$$\text{Recall (prompt)} = \frac{\sum_1^{N_{\text{GT masks}}} 1_{\text{if any prompt point is within the GT mask}}}{N_{\text{GT masks}}} \tag{2}$$

The precision (prompt) measures the proportion of prompt points that accurately fall within the ground truth (GT) masks. The recall (prompt) gauges the proportion of all GT masks that are correctly covered by our prompt generation algorithm.

For the instance segmentation task of microdroplets or microbubbles, we evaluate our algorithm using mean average precision (mAP) and mean average recall (mAR), which are based on precision (mask) and recall (mask):

$$\text{Precision (mask)} = \frac{\sum_i^{N_{\text{prediction}}} 1_{\text{if the prediction matches any GT mask}}}{N_{\text{prediction}}} \tag{3}$$

$$\text{Recall (mask)} = \frac{\sum_i^{N_{\text{GT masks}}} 1_{\text{if the GT mask matches any prediction}}}{N_{\text{GT masks}}} \tag{4}$$

IoU is introduced to determine whether a predicted mask matches a GT mask, which is defined as:

$$\text{IoU (predicted mask, GT mask)} = \frac{\text{intersection area}}{\text{union area}} \tag{5}$$

When the IoU of a predicted mask and a GT mask exceeds a specified threshold, the predicted mask is considered to match the GT mask. The final metrics mAP and mAR are defined as:

$$\text{mAP} = \frac{1}{10} \sum_{\text{IoU threshold}=0.50,0.55,\ldots,0.95} \text{precision (mask)} \tag{6}$$

$$\text{mAR} = \frac{1}{10} \sum_{\text{IoU threshold}=0.50,0.55,\ldots,0.95} \text{recall (mask)} \tag{7}$$

Where “mean” represents an average over a series of IoU thresholds, typically ranging from 0.5 to 0.95 in increments of 0.05. And “average” represents an average over different datasets and images.

4.2. Prompts generation

The precision and recall metrics of our prompt generation algorithm across Dataset 1 to 12 are detailed in Table 2. Almost all the recall values approach 1.0, with Dataset 9 achieving 0.997, indicating that the algorithm effectively identifies nearly all moving microdroplets and microbubbles present in our datasets. However, precision exhibits variability, particularly in Dataset 3 and Dataset 4, where values fall below 0.8. By visualizing the results, we attribute the lower precision primarily to the microdroplets or microbubbles located at the image periphery or in the process of formation, which are not annotated during construction datasets, as shown in Fig. 9. This issue is also observed in other datasets, contributing to an overall decrease in precision. However, these incomplete microdroplets or microbubbles will be filtered out by the post-processor.

Table 2
The precision (prompt) and recall (prompt) of our prompt generating algorithm.

Dataset	Precision (prompt)	Recall (prompt)
1	0.924	1.0
2	0.979	1.0
3	0.785	1.0
4	0.719	1.0
5	0.823	1.0
6	0.889	1.0
7	0.800	1.0
8	0.954	1.0
9	0.929	0.997
10	0.978	1.0
11	0.975	1.0
12	0.987	1.0
Average	0.895	1.0

4.3. Instance segmentation

4.3.1. Instance segmentation results

The instance segmentation results of MicroFlowSAM with the ViT-H image encoder are shown in Fig. 10. Our method accurately identifies and segments microdroplets or microbubbles across most datasets, irrespective of their shapes, sizes, or positions. Quantitative evaluation demonstrates that MicroFlowSAM achieves an mAP of 0.937 and an mAR of 0.902. Moreover, the contours of the segmented masks are highly refined, which is beneficial for the accuracy of subsequent statistical analyses of microdroplets or microbubbles.

We evaluate the accuracy and recall of instance segmentation across different datasets and IoU thresholds to gain a deeper understanding of the results for mAP and mAR, as shown in Fig. 11. It is evident that when the IoU threshold is set to 0.5 or 0.7, the precision values are nearly 1 for all datasets. However, when the IoU threshold is set to 0.9, most datasets maintain their previous results, but both precision and recall decrease significantly for Dataset 9 and Dataset 10. This decline is primarily due to overlapping instances within these datasets, where our method encounters challenges in determining which microdroplet or microbubble an overlapping region belongs to. Nonetheless, it is important to note that this inaccuracy does not necessarily impact our goal of measuring microdroplet or microbubble characteristics. This issue can be mitigated by applying circular or elliptical fitting methods before calculating size and other features.

To fully demonstrate the superiority of our method, we select Mask R-CNN [10] and E2EC [23], two models previously reported to perform well in microfluidic image processing research [9,24]. Additionally, we selected the Transformer-based Mask2Former [25] and Mask DINO [26] models, which achieve SOTA performance on datasets like COCO. We use 70% of the previous dataset as the training set and 30% as the test set. Since the training set contains only 84 images, we employ transfer learning techniques. This approach has proven effective, making the dataset size sufficient for smaller models like Mask R-CNN [11]. More specifically, the ResNet50 [27] backbone of Mask R-CNN is pre-trained on ImageNet [28], and E2EC was pre-trained on COCO. Both Mask2-Former and Mask DINO models were also pre-trained on the COCO dataset. MicroFlowSAM is also re-evaluated on the same test set to ensure the comparability of the results. The evaluation results of the three models are shown in Table 3.

MicroFlowSAM achieves a substantially higher mAP of 0.928 than SAM without prompts, albeit with a slightly reduced mAR of 0.915. Although the mAR for SAM is higher in the absence of prompts, this is attributable to its segmentation of numerous masks not corresponding to the dispersed phase, as illustrated in Fig. 5. These non-target masks are extremely difficult to filter out *via* post-processing, rendering the raw output unsuitable for subsequent applications, such as the statistical analysis of dispersed phase characteristics. This discrepancy arises because SAM is a general-purpose pre-trained segmentation model, and its training dataset likely lacks images from the microfluidics domain. However, by leveraging our proposed prompts, we effectively guide the model to the locations of target objects, thereby enabling accurate instance segmentation.

Compared to these other supervised methods, our method has achieved commendable results, with both mAP and mAR exceeding 0.91, performing only slightly inferior to the SOTA Mask2Former. Additionally, mAP and mAR values above 0.9 are considered sufficient for effective application in the microfluidics domain, ensuring that significant errors are not introduced into the statistical analysis of properties such as the particle size of

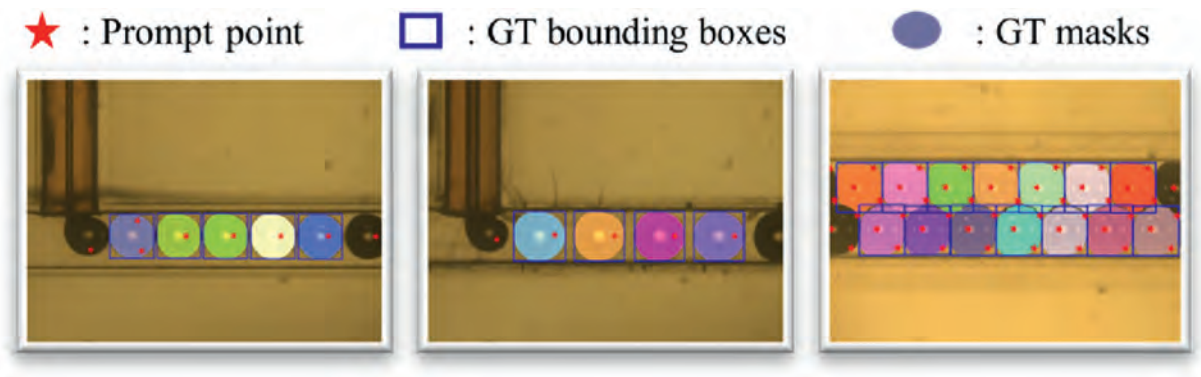

Fig. 9. Examples of the prompts and GT objects.

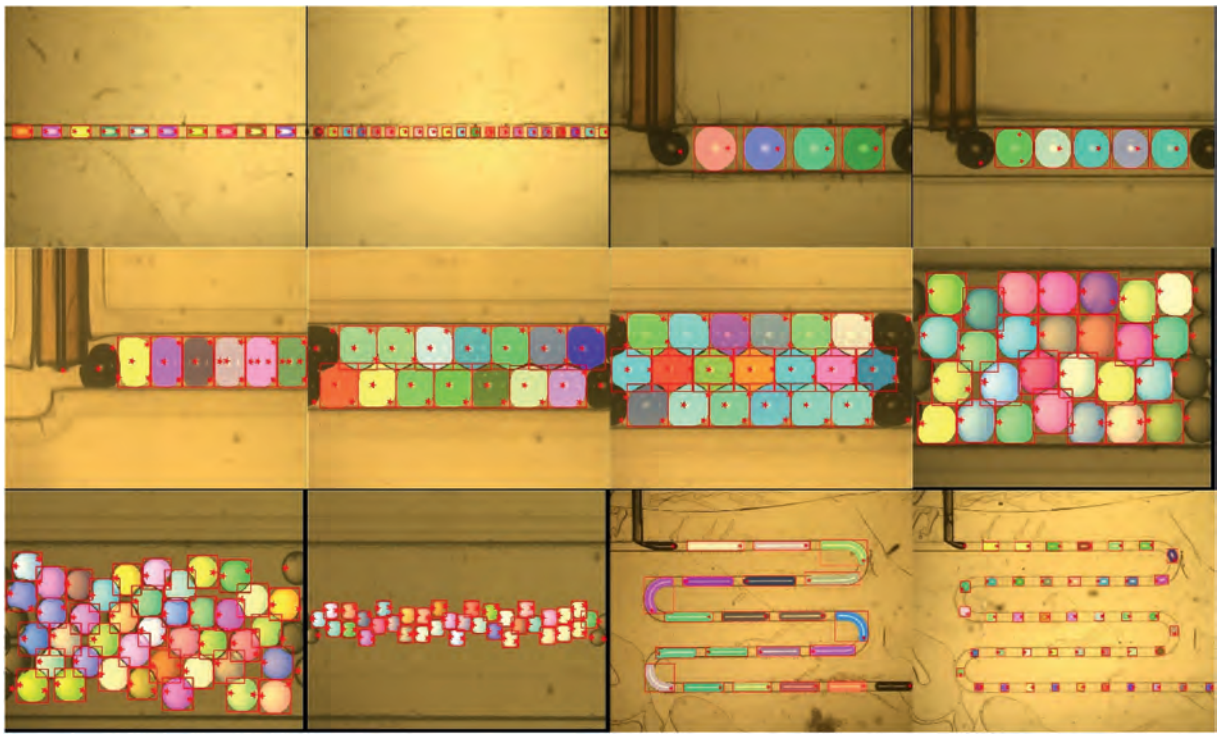

Fig. 10. Instance segmentation results of MicroFlowSAM with the ViT-H image encoder.

bubbles and droplets, a point we would illustrate with a quantitative analysis of size errors in Section 4.4.

4.3.2. Different image encoders

The impact of encoder size on instance segmentation performance is investigated in our study. As mentioned in Section 3.2.1, the official SAM includes three image encoders of different sizes. Additionally, several variants utilizing model compression techniques, such as MobileSAM [29], which uses ViT-T as its image encoder, have been proposed to reduce GPU memory requirements and accelerate the inference process. The performance of MicroFlowSAM with these four image encoders is evaluated and shown in Fig. 12. The *x*-axis represents the computational load of the image encoder's forward computation, measured in giga floating-point operations (GFLOPs). A higher GFLOPs value indicates greater computational resource demands and potentially slower processing speeds. Evidently, the larger image encoders demonstrate better instance segmentation performance in both mAP and mAR metrics, due to their stronger image modeling capabilities.

The processing time for an image of different steps using various image encoders, tested on an RTX A6000 GPU, is presented in Table 4. The primary time-consuming steps include SAM inference processes and post-processing. The inference time of the largest model is 4.8 times longer than that of the smallest model, with the total time being twice as long. Moreover, even for the largest model, its processing time is significantly faster than manual analysis.

4.3.3. Comparison of computational efficiency

The computational efficiency of our proposed method compared to traditional supervised methods is detailed in Table 5. For the training stage, we benchmark supervised methods by measuring their training duration and GPU memory footprint with

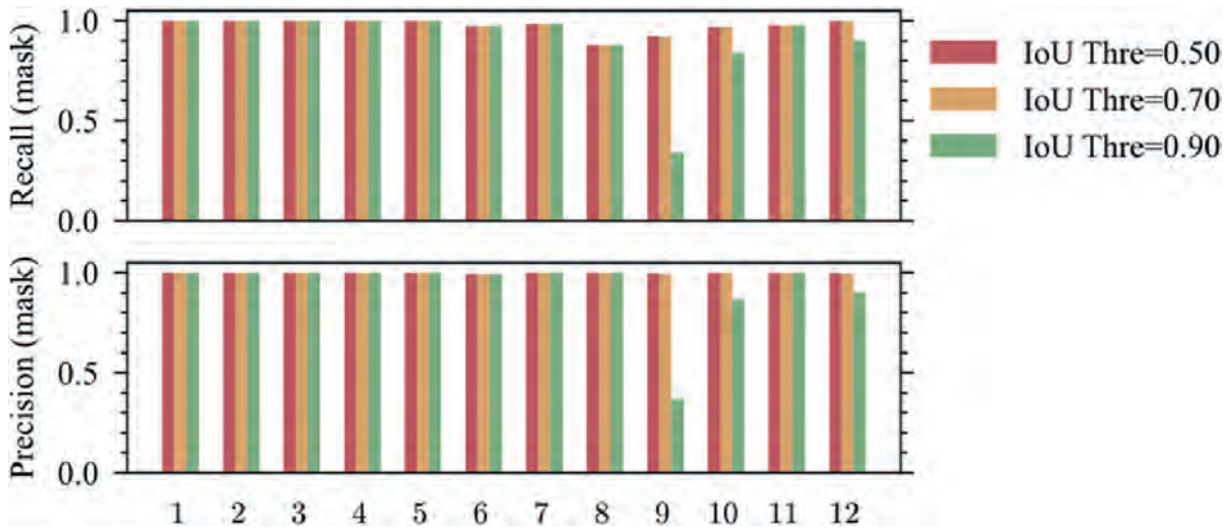

Fig. 11. Evaluation results in different datasets and different IoU thresholds.

Table 3
The comparison of the three methods.

	MicroFlowSAM	SAM	Mask R-CNN	E2EC	Mask2Former	Mask DINO
Annotation free	✓	✓	×	×	×	×
Training free	✓	✓	×	×	×	×
mAP	0.928	0.538	0.826	0.877	0.941	0.909
mAR	0.915	0.949	0.860	0.925	0.955	0.942

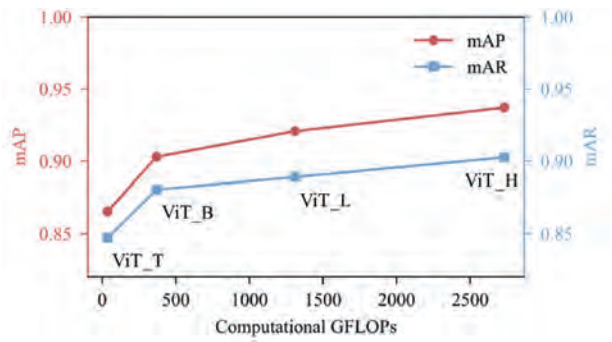

Fig. 12. Performance comparison of MicroFlowSAM with different image encoders.

Table 4
The processing time for an image of different steps using various image encoders, tested on an RTX A6000 GPU.

Time/s	ViT-H	ViT-L	ViT-B	ViT-T
Prompt generating	0.017	0.020	0.018	0.018
SAM inference	0.484	0.338	0.197	0.101
Post-processing	0.242	0.286	0.295	0.245
Total	0.743	0.644	0.510	0.364

a batch size of 4 over 20 epochs. The results in Table 5 reveal that even with the application of transfer learning, these supervised approaches still incur a training time of approximately 10 min. More significantly, Mask2Former, which is the sole model to achieve superior performance over our MicroFlowSAM on both mAP and mAR metrics, imposes a considerable demand on GPU resources, requiring a minimum of 20 GB of VRAM for training. Conversely, our MicroFlowSAM operates without the need for any retraining or fine-tuning, positioning it ahead of other supervised methods during this phase.

For the evaluation of the inference stage, we benchmark the duration each model took to perform inference on a single image, standardized to 1024 × 1024 pixels, and documented the corresponding GPU VRAM consumption. Given that other supervised methodologies often incorporate post-processing operations (such as Non-Maximum Suppression, NMS), we deliberately omit these operations from the timed measurements for both our method and the compared approaches to isolate the core inference performance. As established in prior sections, the inference time and memory requirements of our method scale with the size of the utilized image encoder: smaller encoders yield faster inference and lower memory footprints, typically at the expense of a slight decrease in precision. Notably, when our method is configured with the most compact encoder, ViT-T, its inference time and

Table 5
Comparison of parameter count, consumed time and GPU VRAM of both training and inference stage for different methods.

Methods		MicroFlowSAM				Mask R-CNN	E2EC	Mask2Former	Mask DINO
		ViT-H	ViT-L	ViT-B	ViT-T				
Parameters (M)		641	312	93	10	44	30	44	44
Training (batch size = 4)	Total time/min	\				7	7	10	15
(Epochs = 20)	GPU VRAM/GB	\				6.7	9.2	20.9	45.8
Inference (batch size = 1)	Inference time/s	0.501	0.358	0.215	0.119	0.039	0.052	0.075	0.128
	GPU VRAM/GB	6.2	4.8	3.1	0.3	0.7	0.7	3.0	4.2

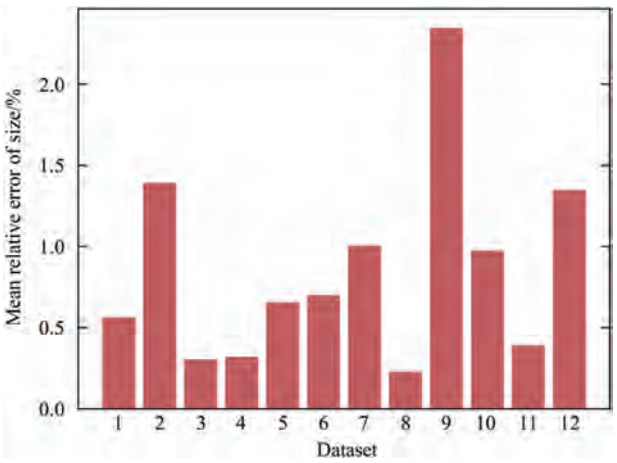

Fig. 13. Mean relative size error across different datasets.

VRAM requirements do not exhibit a significant disparity when compared against models like Mask2Former.

4.4. Applications

Once the mask for each instance is obtained, the characteristics of microdroplets and microbubbles can be easily extracted. For example, obtaining characteristics such as size manually used to be a very time-consuming and labor-intensive task. However, our method automates this process with high precision. Specifically, for microdroplets and microbubbles in Dataset 1 through 7, the length of the bounding box is directly used as the size. For those in Dataset 8, 9, and 10, we first perform circular fitting, and the diameter of the resulting circle is considered the size. In Dataset 11 and 12, where some microbubbles deform within curved channels, we annotate the centerline of the microchannel and calculate the intersection points between each mask and the centerline. By determining the positions of the intersection points along the centerline, we calculated the length of the centerline between these intersection points as the size of each microbubble. These operations are performed on both the predicted and manually annotated masks, and the mean relative size error of the 12 datasets are shown in Fig. 13. For most datasets, the relative size error is below 1%. Even in cases of overlapping instances in Dataset 9, where instance segmentation results may not be flawless, the error in the final feature calculation remains within an acceptable range.

Our method also enables the calculation of dynamic characteristics such as size changes, residence time, and velocity. For example, Dataset 12 is sourced from a study on microflow characteristics and mass transfer performance in viscous solutions [30]. After instance segmentation, microbubble tracking can be directly achieved by calculating the IoU between the masks of consecutive frames (as shown in Movie 1 in the Supplementary Material). This tracking facilitates the computation of dynamic characteristics, with Fig. 14(a) showing the tracking results and Fig. 14(b) illustrating the variation in the projection area of microbubbles on the camera's focal plane with flow distance over one frame. Compared to traditional manual measurement methods, our approach can capture detailed information about the size of each microbubble at every position, which benefits the exploring of microflows characteristics. Additionally, for videos capturing the breakup processes of microdroplets or microbubbles, previously reported methods also required training instance segmentation models [11]. Our approach can replace these instance segmentation models, thereby alleviating the burden of data annotation and model training.

4.5. Discussions

Large-scale pre-trained models are achieving widespread adoption across diverse fields. Instance segmentation models such as the SAM, however, necessitate prompts to designate target regions within an image for subsequent identification and segmentation. Providing these prompts manually for every image in the microfluidics domain is markedly inefficient. We observe that in this domain, the dispersed phase exhibits continuous motion. Leveraging this observation, we propose an automated prompt engineering method that adapts SAM to the microfluidics domain without requiring retraining, yielding performance competitive with supervised approaches. This method enables instance segmentation of the dispersed phase in microfluidic videos, thereby facilitating statistical analysis of its characteristics with high spatiotemporal precision.

While demonstrating this broader applicability, it is important to note that our current work primarily validates instance segmentation in microfluidic images characterized by relatively clear objects and minimal overlap. Previous studies have typically

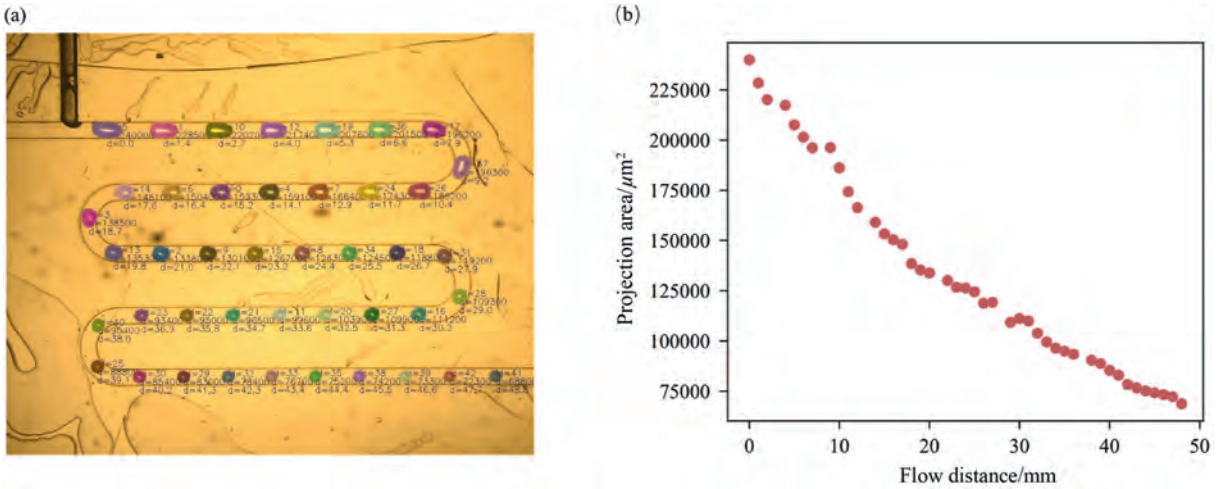

Fig. 14. (a) Visualization of the tracking results, showing each microbubble's ID, projection area (denoted as a), and flow distance (denoted as d). (b) The variation in the projection area of microbubbles with flow distance.

approached this task using supervised methods, which necessitate laborious manual image annotation and model training. This reliance on extensive annotation has consequently restricted the broader application of deep learning computer vision algorithms within the microfluidics field. Our proposed method, MicroFlowSAM, offers a significant advantage by bypassing this entire process, thereby enhancing convenience and lowering the barrier to entry. However, MicroFlowSAM' s performance is suboptimal when faced with challenging scenarios such as low contrast or highly overlapping objects. For these specific situations, supervised fine-tuning of the model presents a more effective solution. Compared to retraining a model from scratch, supervised fine-tuning can achieve commendable results with a substantially smaller set of annotated images.

5. Conclusions

In conclusion, this paper introduces MicroFlowSAM for instance segmentation of microdroplets and microbubbles in microscopic videos. We introduce a promptable large-scale image segmentation model into the field of microfluidics and leverage the motion of microdroplets and microbubbles for automated prompt engineering. This approach enables instance segmentation without the need for training or annotation. It achieves an mAP of 0.937 and an mAR of 0.902, comparable to that of the supervised methods, without requiring training and while also demonstrating its generalizability across our 12 diverse datasets. This method eliminates the need for labor-intensive annotation efforts and reduces the computational resources typically associated with training supervised deep learning models. Moreover, it accelerates the analysis speed of microfluidics videos, underscoring the significant potential of foundational vision-based models in advancing microfluidics research.

CRediT Authorship Contribution Statement

Wenle Xu: Writing – original draft, Software, Methodology, Conceptualization. Lin Sheng: Visualization, Data curation. Tong Qiu: Writing – review & editing, Supervision. Kai Wang: Supervision. Guangsheng Luo: Supervision.

Declaration of Competing Interest

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

The author is an Editorial Board Member/Editor-in-Chief/Associate Editor/Guest Editor for this journal and was not involved in the editorial review or the decision to publish this article.

Acknowledgements

The authors gratefully acknowledge the financial support from National Natural Science Foundation of China (21991104).

Supplementary Material

Supplementary video related to this article can be found at https://doi.org/10.1016/j.cjche.2025.05.023

References

[1] K.F. Jensen, Flow chemistry—Microreaction technology comes of age, *AIChE J.* 63 (2017) 858–869.
[2] I. Lignos, S. Stavrakis, G. Nedelcu, L. Protesescu, A.J. deMello, M.V. Kovalenko, Synthesis of cesium lead halide perovskite nanocrystals in a droplet-based microfluidic platform: fast parametric space mapping, *Nano Lett.* 16 (2016) 1869–1877.
[3] K. Matuła, F. Rivello, W.T.S. Huck, Single-cell analysis using droplet microfluidics, *Adv. Biosyst.* 4 (2020) 1900188.
[4] R.H. Abou-Saleh, F.J. Armistead, D.V.B. Batchelor, B.R.G. Johnson, S.A. Peyman, S.D. Evans, Horizon: microfluidic platform for the production of therapeutic microbubbles and nanobubbles, *Rev. Sci. Instrum.* 92 (2021) 074105.
[5] M. Abolhasani, A. Günther, E. Kumacheva, Microfluidic studies of carbon dioxide, *Angew. Chem. Int. Ed.* 53 (2014) 7992–8002.
[6] S. Battat, D.A. Weitz, G.M. Whitesides, Nonlinear phenomena in microfluidics, *Chem. Rev.* 122 (2022) 6921–6937.
[7] L. Sheng, Y. Chang, J. Wang, J. Deng, G. Luo, Hydrodynamics of gas–liquid microfluidics: a review, *Chem. Eng. Sci.* 285 (2024) 119563.
[8] C. Shen, Q. Zheng, M. Shang, L. Zha, Y. Su, Using deep learning to recognize liquid–liquid flow patterns in microchannels, *AIChE J.* 66 (2020) e16260.
[9] S. Zhang, X. Liang, X. Huang, K. Wang, T. Qiu, Precise and fast microdroplet size distribution measurement using deep learning, *Chem. Eng. Sci.* 247 (2022) 116926.
[10] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, 2980–2988.
[11] S. Zhang, H. Li, K. Wang, T. Qiu, Accelerating intelligent microfluidic image processing with transfer deep learning: a microchannel droplet/bubble breakup case study, *Sep. Purif. Technol.* 315 (2023) 123703.
[12] T.B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D.M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2020, pp. 1877–1901.
[13] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: Proceedings of the 38th International Conference on Machine Learning, PMLR, 2021, 8748–8763.
[14] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, P. Dollár, R. Girshick, Segment anything, in: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, Paris, France, 2023, 3992–4003.
[15] A. Carraro, M. Sozzi, F. Marinello, The segment anything model (SAM) for accelerating the smart farming revolution, *Smart Agric. Technol.* 6 (2023) 100367.
[16] S. Khan, M. Naseer, M. Hayat, S.W. Zamir, F.S. Khan, M. Shah, Transformers in vision: a survey, *ACM Comput. Surv.* 54 (2022) 1–41.
[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: transformers for image recognition at scale, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, 2021. https://openreview.net/forum?id=YicbFdNTTy (Accessed May 10, 2025).
[18] M. Tancik, P.P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J.T. Barron, R. Ng, Fourier features let networks learn high frequency functions in low dimensional domains, in: H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, H.-T. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, Virtual, 2020. https://proceedings.neurips.cc/paper/2020/hash/55053683268957697aa39fba6f231c68-Abstract.html. (Accessed 10 May 2025).
[19] R. Jain, H.-H. Nagel, On the analysis of accumulative difference pictures from image sequences of real world scenes, *IEEE Trans. Pattern Anal. Mach. Intell.* PAMI-1 (1979) 206–214.
[20] J.S. Kulchandani, K.J. Dangarwala, Moving object detection: review of recent research trends, in: 2015 International Conference on Pervasive Computing (ICPC), 2015, 1–5.
[21] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1, MIT Press, Cambridge, MA, USA, 2015, 91–99.
[22] F. Bolelli, S. Allegretti, L. Baraldi, C. Grana, Spaghetti labeling: directed acyclic graphs for block-based connected components labeling, *IEEE Trans. Image Process.* 29 (2020) 1999–2012.
[23] T. Zhang, S. Wei, S. Ji, E2EC: an end-to-end contour-based method for high-quality high-speed instance segmentation, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, 4433–4442.
[24] W. Xu, S. Zhang, K. Wang, T. Qiu, An efficient approach for droplet coalescence videos processing based on instance segmentation and multi-object tracking algorithms, in: Computer Aided Chemical Engineering, Elsevier, 2024, 3001–3006.
[25] B. Cheng, I. Misra, A.G. Schwing, A. Kirillov, R. Girdhar, Masked-attention mask transformer for universal image segmentation, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, New Orleans, LA, USA, 2022, 1280–1289.

[26] F. Li, H. Zhang, H. Xu, S. Liu, L. Zhang, L.M. Ni, H.-Y. Shum, Mask DINO: towards a unified transformer-based framework for object detection and segmentation, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Vancouver, BC, Canada, 2023, 3041–3050.
[27] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 770–778.
[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge, *Int. J. Comput. Vis.* 115 (2015) 211–252.
[29] C. Zhang, D. Han, Y. Qiao, J.U. Kim, S.-H. Bae, S. Lee, C.S. Hong, Faster segment anything: towards lightweight SAM for mobile applications. 2023. http://arxiv.org/abs/2306.14289. (Accessed 17 June 2024).
[30] L. Sheng, Y. Chang, J. Deng, G. Luo, Hydrodynamics and mass transfer performance of gas–liquid microflow in viscous liquids, *Chem. Eng. J.* 454 (2023) 140407.