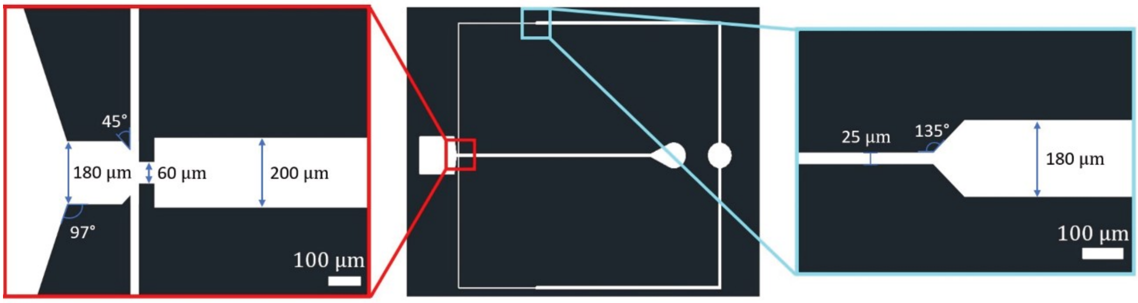
Supplementary Material

Open-source tool for real-time and automated analysis of droplet-based microfluidic

Joana P. Neto,* Ana Mota, Gonçalo Lopes, Beatriz J. Coelho, João Frazão, André T. Moura, Beatriz Oliveira, Bárbara Sieira, José Fernandes, Elvira Fortunato, Rodrigo Martins, Rui Igreja, Pedro V. Baptista, Hugo Águas,*

*** Correspondence:** joana.neto@campus.fct.unl.pt and hma@fct.unl.pt



Supplementary Figure 1. Droplet generator design layout. Insets: flow-focusing region (left) and oil channel (right).

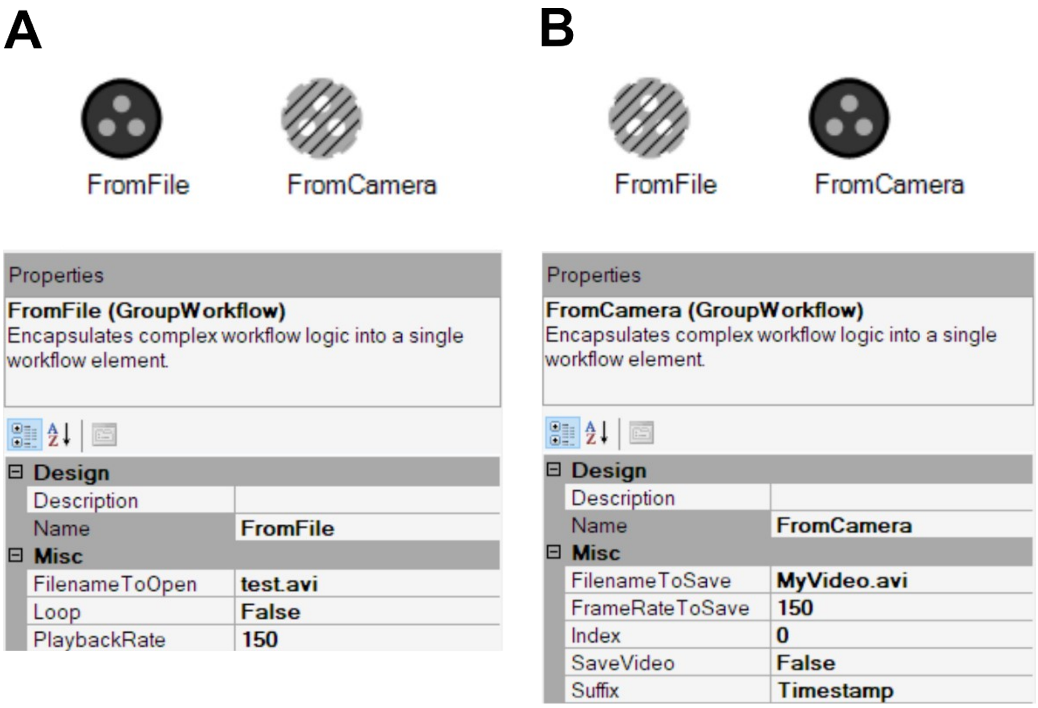
Bonsai workflow for droplet analysis

First, users need to download *Bonsai*, freely available at https://Bonsai-rx.org/docs/articles/installation.html and install *Bonsai - Starter Pack*, from *Manage Packages* (for more detail please consult https://Bonsai-rx.org/docs/articles/packages.html). Moreover, to record from the camera, the user needs to install the adequate camera drivers (e.g., *FlyCapture_2.11.3.425_x64.exe* for the camera used in this work) and *Bonsai – PointGrey Library* package. Then, the workflow used during this work is available online https://github.com/JoanaPNeto/Droplets. The workflow has three main nodes, which represent different tasks within the droplet pipeline analysis: 1) *'Image Acquisition'*, 2) *'Feature Extraction'* and 3) *'Droplet Analysis'*. *Bonsai* uses the OpenCV vision library (https://docs.opencv.org/2.4.13.2/) inside these nodes (e.g., *FromFile*, *FromCamera*, *Crop*, *Grayscale*, *Threshold*, *FindContours, OpticalFlow, Magnitude, Average, Centroid, Buffer*).

Note: The 'Image Acquisition' node needs to be defined before starting the workflow. The *'Feature Extraction'* and *'Droplet Analysis'* are nodes with externalized properties, allowing for the dynamic control of several parameters, while the workflow is running. Moreover, by double-clicking the *'Droplet Analysis'* node, the visualization of the data at different stages of processing can be launched at any time in parallel with the execution of the dataflow.

1) *'Image acquisition'*

In more detail, the node *'Image acquisition'* enables: i) the visualization and acquisition of real-time video data or ii) the visualization of offline video (i.e., pre-recorded video). Therefore, this workflow can be used online to extract measures and make real-time decisions on the experimental conditions, or to analyse video data offline. If users double-click on the *'Image acquisition'* node, a new workflow panel will open and it is possible to enable or disable the node *'FromFile'* or *'FromCamera'* by right-clicking on the respective node. Please note that users must make sure that the workflow is stopped to select the intended option. If the node *'FromFile'* is enabled (**Supplementary Figure 2A**), it will display a pre-recorded video. On *'Properties'*, clicking on *'FileNameToOpen'* allows users to indicate the video directory (the location where the video is saved on the computer). The video file should be avi format (.avi file extension). If the node *'FromCamera'* is enabled (Supplementary Figure 2B), it will display the video being recorded. To save the video, users must choose a folder in the *'Properties'*, as well as the *'FrameRateToSave'*, and select *'True'* for the option *'SaveVideo'*. The video should be saved in avi format, such as *'MyVideo.avi'*. After, the user can initiate the workflow by clicking *'Start'* and the output of the *'Image Acquisition'* node will appear in the pipeline as *'Frames'*.



A

Properties

FromFile (GroupWorkflow)
Encapsulates complex workflow logic into a single workflow element.

Design	
Description	
Name	**FromFile**
Misc	
FilenameToOpen	**test.avi**
Loop	**False**
PlaybackRate	**150**

B

Properties

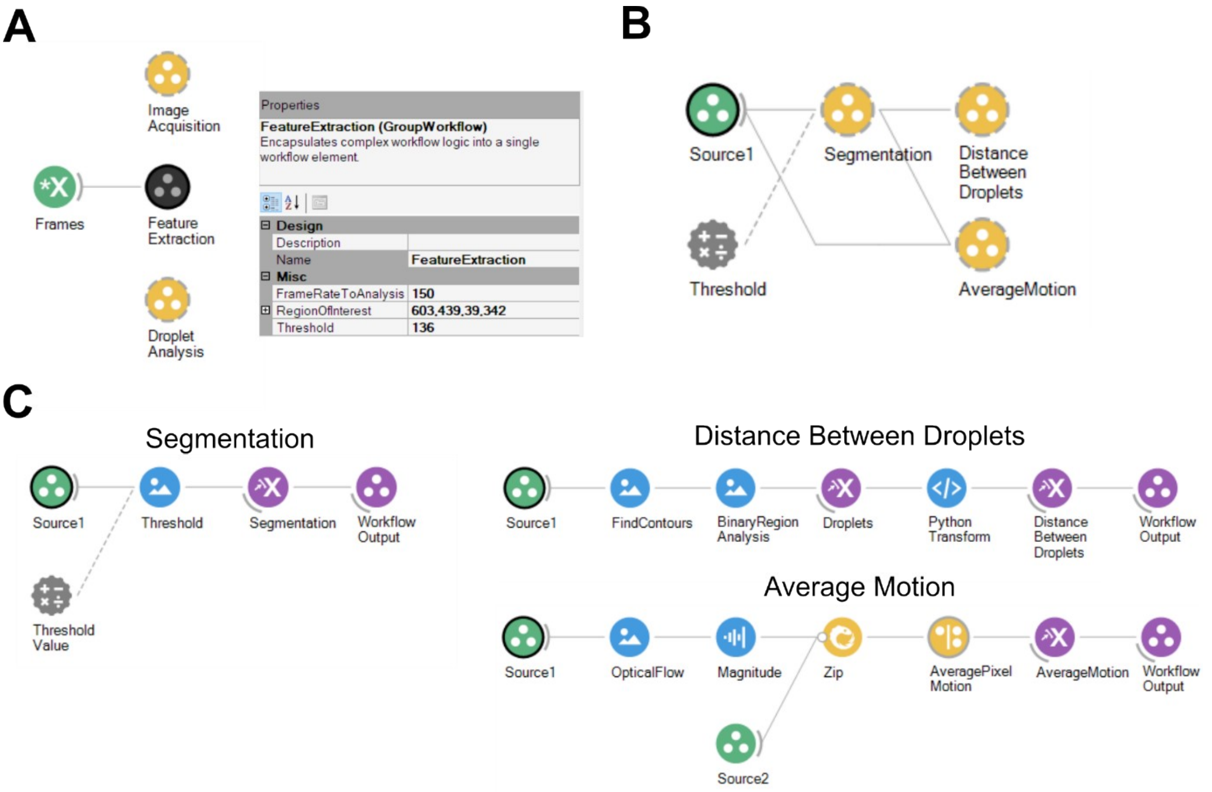
FromCamera (GroupWorkflow)
Encapsulates complex workflow logic into a single workflow element.

Design	
Description	
Name	**FromCamera**
Misc	
FilenameToSave	**MyVideo.avi**
FrameRateToSave	**150**
Index	**0**
SaveVideo	**False**
Suffix	**Timestamp**

Supplementary Figure 2. *'Image Acquisition'* group node. (A) *'FromFile'* node is enabled, (B) *'FromCamera'* node is enabled

2) *'Feature extraction'*

The node *'Feature extraction'* enables the segmentation, detection and extraction of direct measurements related to the droplets. As shown in **Supplementary Figure 3A**, if the user clicks on the node *'Feature extraction'*, the *'Properties panel'* will show three parameters that must be set: i) *'FrameRateToAnalysis'*, ii) *'RegionOfInterest'*, and iii) *'Threshold'*. First, the correct frame rate of the video analyzed must be selected (i.e., camera/video frame rate). Also, the users need to define a region of interest (ROI) around the microfluidic channel. On the other hand, to understand how this node works, by double-clicking on the node, two important group nodes will appear: the node *'Regionofinterest'*, which is used to define the ROI, and the *'DropletProcessing'* node, which is responsible for the segmentation, detection and direct measurements of droplets. The *'Regionofinterest'* crops a rectangular sub-region of each image and converts the image into grayscale, so that a threshold can be applied to create a binary image, depending on the pixel intensity (please see Video Tutorial: Read, crop and save a video file). Users should select their intended ROI (ideally, the region of interest should include droplets and, if possible, avoid adding walls) by clicking on the first corner of the crop rectangle and dragging down to the last corner of the rectangle. Additionally, inside *'DropletProcessing'* (**Supplementary Figure 3B**), the node *'Segmentation'* is used to isolate the droplets by using the *'Threshold'* value for the binarization of the image: all pixels above the threshold become black, and all pixels below the threshold become white (**Supplementary Figure 3C**). Once more, to visualize what is inside each node, users should make sure that the workflow is stopped, and then double-click on the node. By adjusting the *'Threshold'* value on the *'Properties'* panel, the border of the droplets may be tuned for the following step of detection.



Properties

FeatureExtraction (GroupWorkflow)
Encapsulates complex workflow logic into a single workflow element.

Design	
Description	
Name	FeatureExtraction
Misc	
FrameRateToAnalysis	150
RegionOfInterest	603,439,39,342
Threshold	136

Supplementary Figure 3. *'Feature Extraction'* group node. (A) If the node *'FeatureExtraction'* is selected, then the *'Properties'* panel will display all the configuration properties which are available for this node. Most properties can be configured simply by changing the text value in the corresponding row of the property grid, (B) *'DropletProcessing'* node (that is within the *'Feature Extraction'* node) where the users can find three group nodes: *'Segmentation'*, *'DistanceBetweenDroplets'* and *'AverageMotion'*, which unfold as illustrated on (C).

Once the users find appropriate threshold levels for droplets segmentation, droplet detection and droplet measurements can be performed. For the calculation of droplet radius, speed and frequency we will use three measures: i) droplets major axis length, ii) distance between droplets centroid and iii) the average motion of droplets pixels per frame. These values will be computed within the group nodes *'DistanceBetweenDroplets'* and *'AverageMotion'* (**Supplementary Figure 3B, C**). For the droplets major axis length and distance between droplets centroid, firstly the *'FindContours'* node will trace the contours of all the objects in a black-and-white image. An object is defined as a region of connected white pixels. Then, the operator *'BinaryRegionAnalysis'* computes image moments to describe droplets after segmentation. Simple properties of the image which are found via image moments include major and minor axis length, area, center of mass, and orientation for all the detected contours per frame[1]. For instance, the major axis (1) and centroid (2) are calculated using imaging raw (M00, M10, M01) and central moments (Mu20, Mu02, Mu11) (please check code here).

$$MajorAxisLength = 2.75 * \sqrt{\left(\frac{Mu20}{M00}\right) + \left(\frac{Mu02}{M00}\right) + \sqrt{\left(2 * \left(\frac{Mu11}{M00}\right) * 2 * \left(\frac{Mu11}{M00}\right) + \left(\left(\frac{Mu20}{M00}\right) - \left(\frac{Mu02}{M00}\right) * \left(\frac{Mu20}{M00}\right) - \left(\frac{Mu02}{M00}\right)\right)\right)}}$$

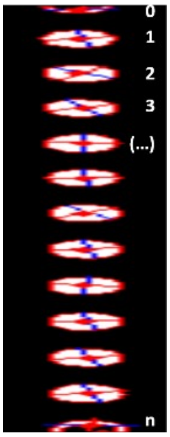
(1)

$$Centroid(x,y) = (\frac{M10}{M00},\frac{M01}{M00}) \tag{2}$$

Moreover, the distance in pixels, between the centroid of consecutive droplets is then computed using the *'PhytonTransform'* node and saved as *'DistanceBetweenDroplets'*. Basically the code encompasses the calculation of the distance (3) between two points in a 2-dimensional space.

$$Distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{3}$$

We observed that droplets entering and exiting (Droplet [0] and Droplet [n]) the frame present a 'instable' centroid, so we decided to use the Droplet [1] and [2] in each frame to calculate the distance. Bellow, an example of *'BinaryRegionAnalysis'* output from video 11, where the red dot represents the centroid and the red axis the major axis, and also the code used to calculate the distance between droplet's centroid in position 1 and 2.



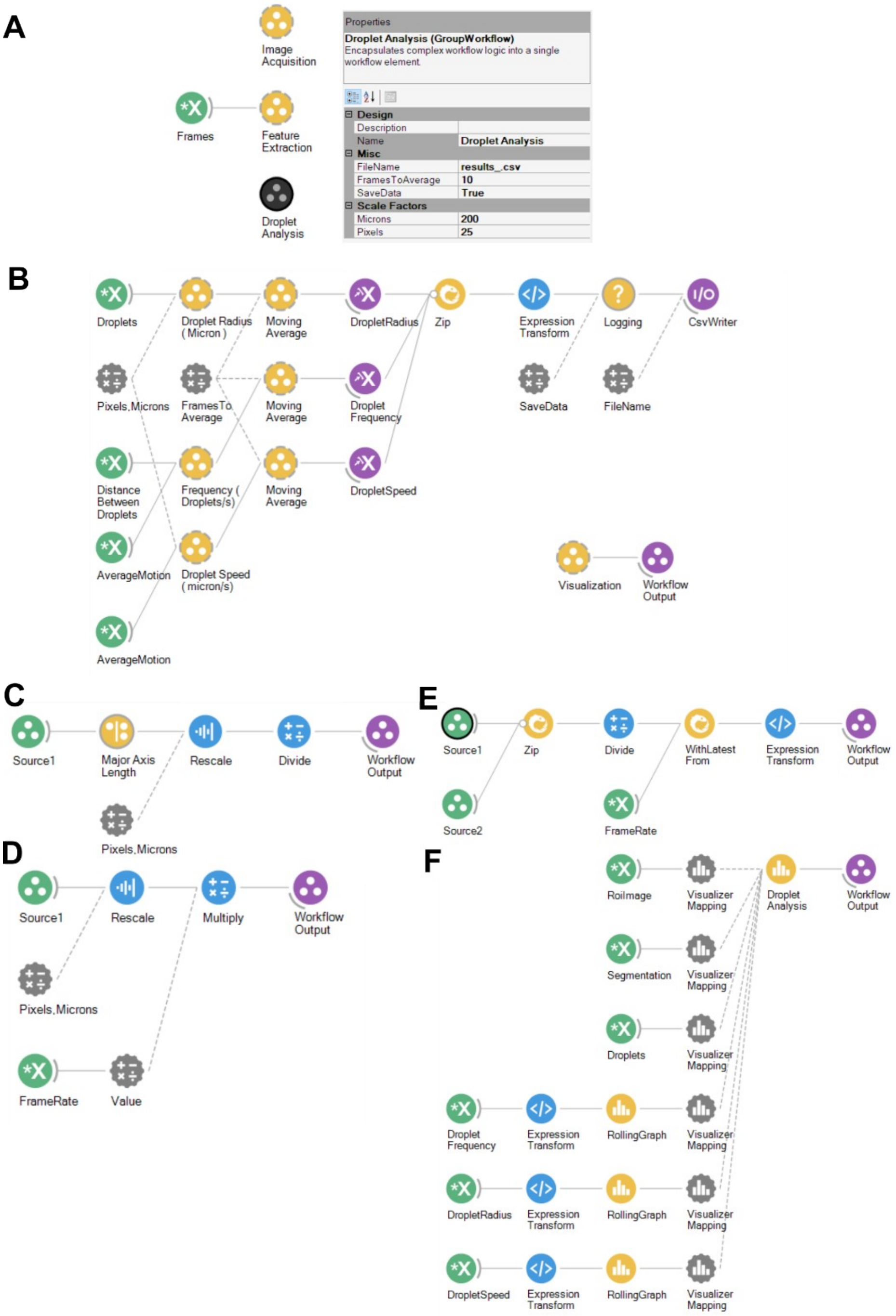
```
#Code used in PythonTransform node

from math import sqrt
@returns(float)
def process(value):
  if value.Count < 4:
    return 0
  else:
    d = value[1].Centroid - value[2].Centroid
    return sqrt(d.X * d.X + d.Y * d.Y)
```

Moreover, for the average motion of droplet pixels the *'Opticalflow'* node (here) computes a dense optical flow through Gunnar Farnerback's algorithm[2] which detects the pixel intensity changes between two consecutive frames. It identifies the motion of objects between every two consecutive frames, providing the flow vectors of all pixels from *'RoiImage'* (represented as *'Source1'*). As shown in **Supplementary Figure 3C** this node is connected to the *'Magnitude'* node, which computes the magnitude (distance covered during pixel motion) from all flow vectors. The vector magnitude is then zipped together with the *'Segmentation'* node (represented as *'Source2'*), which provides the calculation of the motion of all droplet pixels per frame as *'AverageMotion'*. To know more about the *'Zip'* node look here.

3) *'Droplet Analysis'*

Finally, the node *'Droplet Analysis'* computes the radius (µm), production rate (droplets/s) and speed (µm/s) of passing droplets and it is responsible for the visualization and saving of the data. If users click on the node, they will see on *'Properties'* panel the possibility of saving the results in a .csv file (**Supplementary Figure 4A**). To save the file, users need to click on *'FileName'* to choose a folder to save the data and select *'True'* for the option *'SaveData'*. The results (radius, production rate and speed) should be saved in csv format, such as *'results.csv'*. Moreover, users need to define the *'Microns'*/*'Pixels'* aspect ratio by introducing values of channel width (200 µm in this case) in µm and its correspondence in pixels. Additionally, it is possible to reduce the susceptibility of the signal to fast changes, particularly in the occurrence of fast vibrations by introducing *‘FramesToAverage’*. The larger the number of frames used to calculate the moving average, the more fluctuation/instability smoothing occurs, as more droplets are included in each calculated average. Moreover, by double-clicking on *‘Droplet Analysis’*, users will find the main nodes employed to calculate radius (µm), production rate (droplets/s) and speed (µm/s) (**Supplementary Figure 4B**).



Properties

Droplet Analysis (GroupWorkflow)
Encapsulates complex workflow logic into a single workflow element.

Design	
Description	
Name	Droplet Analysis
Misc	
FileName	results_.csv
FramesToAverage	10
SaveData	True
Scale Factors	
Microns	200
Pixels	25

Supplementary Figure 4. *‘Droplet Analysis’* group node. (A) If the node *‘Droplet Analysis’* is selected, the properties panel will display all the configuration properties which are available for this node, which include saving the results in a csv file by enabling *‘SaveData’*, applying a moving average to data and translating pixels’ values to microns. (B) Droplet analysis workflow where droplet radius, droplet generation rate as frequency and droplet speed are calculated and outputted, (C) *'DropletRadius (micron)'* node, (D) *'Droplet Speed (micron/s)'* node, (E) *'Frequency (droplets/s)'* node, (F) *'Visualization'* node.

Droplet radius (4) is calculated within the node *'DropletRadius(micron)'*, where the major axis length from all the droplets per frame (node *'Major Axis Length'*) is rescaled (node *'Rescale'*) for µm and then divide by 2 (node *'Divide'*) (**Supplementary Figure 4C**).

$$Radius\ (\mu m) = \frac{MajorAxisLength(pixel) * (Microns(\mu m)/Pixels(pixel))}{2} \tag{4}$$

Concerning droplet speed (5), by knowing the values of *'Average Motion'* (average motion of all droplet's pixels per frame as *'Source1'*) and *'FrameRate'*, it is possible to calculate the droplet speed, as shown in **Supplementary Figure 4D**, with the node *'Droplet Speed (micron/s)'*. The node *'Rescale'* allows the conversion from pixel/frame to µm/frame and the node *'Multiply'* enables the conversion from µm/frame to µm/s.

$$Speed\left(\frac{\mu m}{s}\right) = AverageMotion\left(\frac{pixel}{frame}\right) * \frac{Microns(\mu m)}{Pixels(pixel)} * FrameRate(\frac{frame}{s}) \tag{5}$$

Lastly in **Supplementary Figure 4E**, the droplet production rate (6) per frame is determined as the distance between the centroid of consecutive droplets divided by their speed in the node *'Frequency (droplets/s)'*. The *'DistanceBetweenDroplets'* and *'AverageMotion'* are zipped together, providing the values for calculating the frequency of droplets per frame. Finally, the *'Frequency (droplets/s)'* node in **Supplementary Figure 4D** computes the production rate (droplets/s) according to the *'FrameRate'* value using the node *'ExpressionTransform'*.

$$Frequency\left(\frac{droplet}{s}\right) = \frac{AverageMotion\left(\frac{pixel}{frame}\right)}{DistanceBetweenDroplets\left(\frac{pixel}{droplet}\right)} * FrameRate\left(\frac{frame}{s}\right) \tag{6}$$

$$= \frac{\frac{FrameRate\left(\frac{frame}{s}\right)}{DistanceBetweenDroplets\left(\frac{pixel}{droplet}\right)}}{AverageMotion\left(\frac{pixel}{frame}\right)} rame/eMotion(nDroplets'd, the\ red\ ax$$

$$XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$$

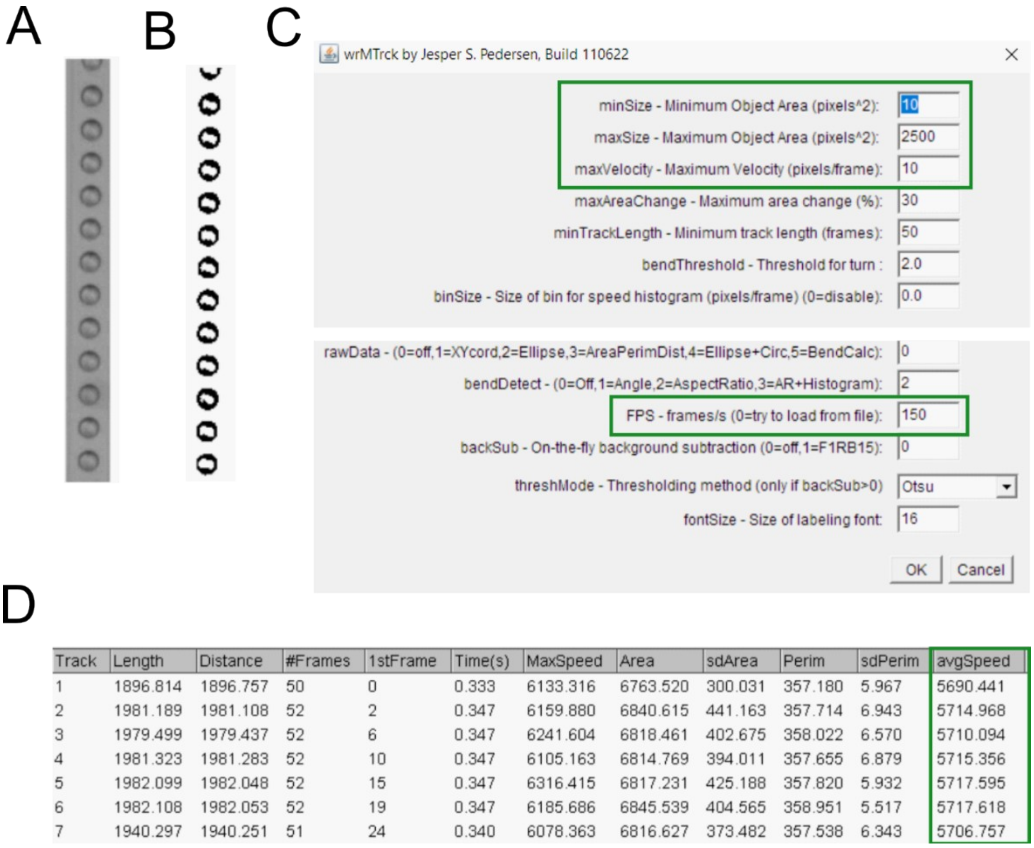
$$XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$$

$$rame/eMotion(nDroplets'd, the\ red\ axis\ the\ major\ axis, the\ blue\ axisXXXXXXXXXXXXXXXXXXXXXXX$$

$$XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX$$

$$XXXXXXXXXXXXXXXXXXXXXXXXXXXXX$$

Finally, the node *'Droplet Analysis'* allows the visualization (**Supplementary Figure 4F**) of a panel where the users can observe the output of: *'RoiImage'* as the result of the crop, *'Segmentation'* as the result of *'Threshold'*, and *'Droplets'* as the result of *'BinaryRegionAnalysis'*, *'Droplet frequency'*, *'Droplet speed'* and *'Droplet Radius'*.

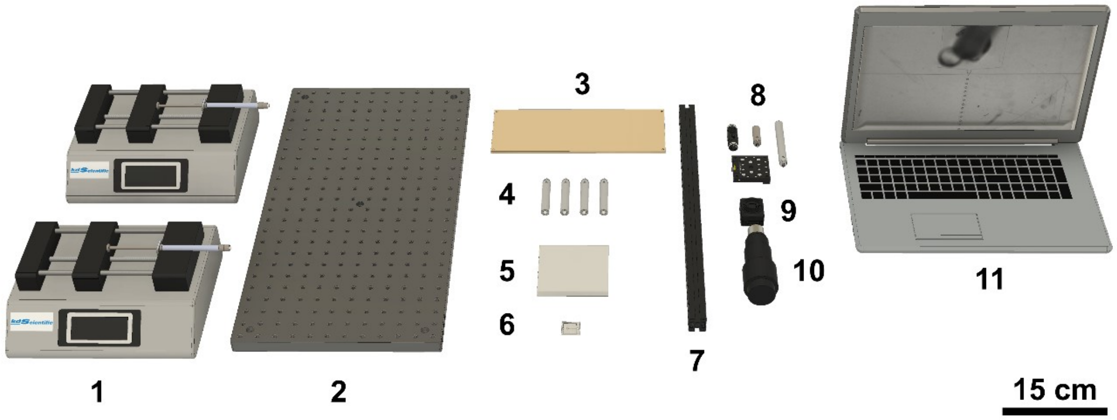


minSize - Minimum Object Area (pixels^2): 10
maxSize - Maximum Object Area (pixels^2): 2500
maxVelocity - Maximum Velocity (pixels/frame): 10
maxAreaChange - Maximum area change (%): 30
minTrackLength - Minimum track length (frames): 50
bendThreshold - Threshold for turn : 2.0
binSize - Size of bin for speed histogram (pixels/frame) (0=disable): 0.0

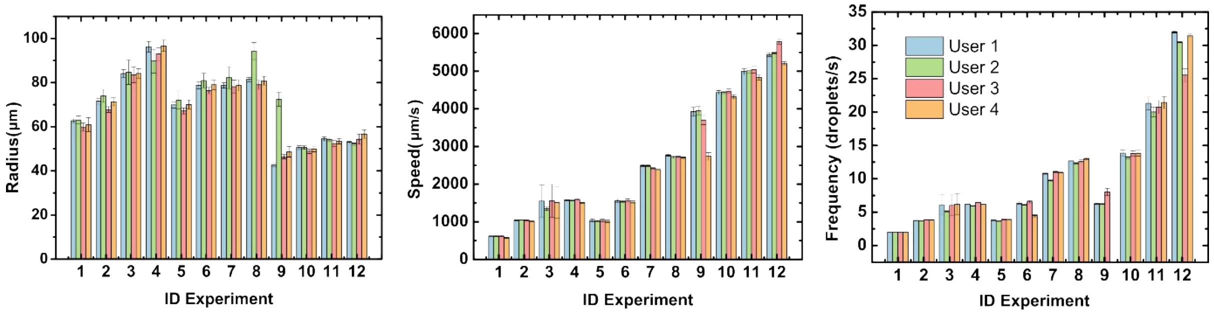
rawData - (0=off,1=XYcord,2=Ellipse,3=AreaPerimDist,4=Ellipse+Circ,5=BendCalc): 0
bendDetect - (0=Off,1=Angle,2=AspectRatio,3=AR+Histogram): 2
FPS - frames/s (0=try to load from file): 150
backSub - On-the-fly background subtraction (0=off,1=F1RB15): 0
threshMode - Thresholding method (only if backSub>0) Otsu
fontSize - Size of labeling font: 16

Track	Length	Distance	#Frames	1stFrame	Time(s)	MaxSpeed	Area	sdArea	Perim	sdPerim	avgSpeed
1	1896.814	1896.757	50	0	0.333	6133.316	6763.520	300.031	357.180	5.967	5690.441
2	1981.189	1981.108	52	2	0.347	6159.880	6840.615	441.163	357.714	6.943	5714.968
3	1979.499	1979.437	52	6	0.347	6241.604	6818.461	402.675	358.022	6.570	5710.094
4	1981.323	1981.283	52	10	0.347	6105.163	6814.769	394.011	357.655	6.879	5715.356
5	1982.099	1982.048	52	15	0.347	6316.415	6817.231	425.188	357.820	5.932	5717.595
6	1982.108	1982.053	52	19	0.347	6185.686	6845.539	404.565	358.951	5.517	5717.618
7	1940.297	1940.251	51	24	0.340	6078.363	6816.627	373.482	357.538	6.343	5706.757

Supplementary Figure 5. Image J software for droplet speed analysis where the wrMTrck plugin was used to obtain the average droplet speed for each video. The video was imported, the crop was adjusted and all frames converted into binary images. Then, the plugin was started and the maximum and minimum size of droplets to detect were adjusted, as well as the maximum expected speed. (A) Video crop section, (B) Binary image after applying 'Make Binary' command, (C) Parameters configuration for droplet speed calculation. Green boxes indicate the parameters to adjust, (D) Summary of results in µm with speed highlighted with a green box.



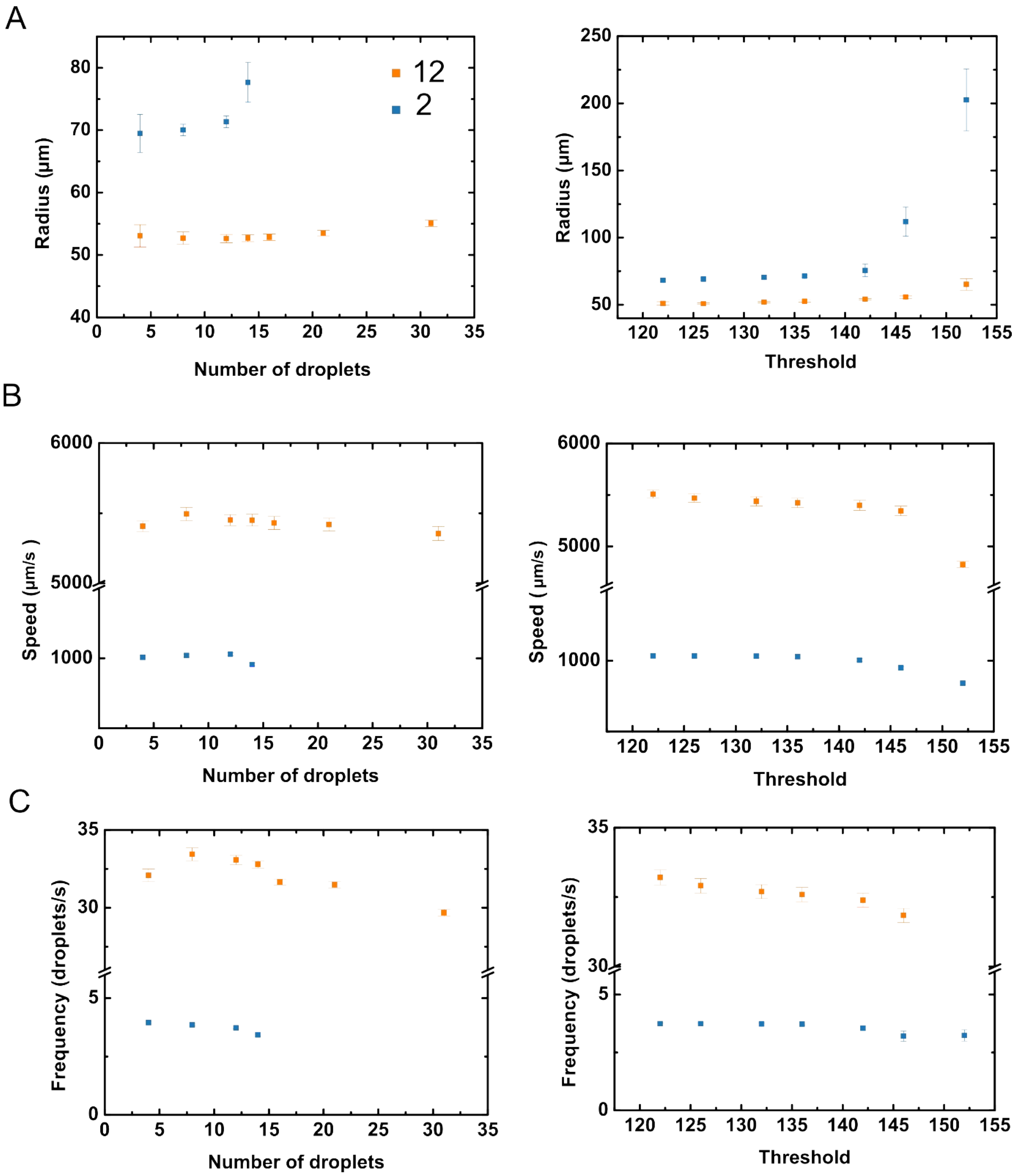
Supplementary Figure 6. Visualization of all the components required for the generation and visualization of nano-liter droplets, disassembled. The following components are numbered: 1 – two Legato 210 P syringe pumps from KdScientific (Holliston, MA, USA); 2 - MBH3060/M aluminum optical breadboard from ThorLabs (Newton, NJ, USA); 3 – standard acrylic base, cut and drilled with laser (dyed in amber in this image for better visualization); 4 – four optical posts (model TR75V/M from ThorLabs); 5 – custom-made LED lighting base; 6 – microfluidic device; 7 – construction rail (model XE25L500/M from ThorLabs); 8 – camera mounting adaptors and screws (several models, from ThorLabs); 9 - Chameleon3 USB3 monochrome camera (model CM3-U3-13Y3M-CS, Python 1300 from Teledyne FLIR, Wilsonville, OR, USA); 10 - MLM3X-MP magnification lens from Computar (Cary, NC, USA); 11 – Laptop for image analysis with the Bonsai software. Briefly, the syringe pumps push the oil and aqueous phases onto the microfluidic chip, and droplets are produced at the flow focusing region. The microfluidic device, in turn, is positioned on top of the transparent acrylic base, and the LED lights below are adjusted to improve the contrast of the droplet border, which in turn facilitates droplet segmentation and analysis by Bonsai. All adjustments are observed via the computer, which receives real-time input from the camera (connected to the magnifying lens) positioned directly above the microfluidic chip.



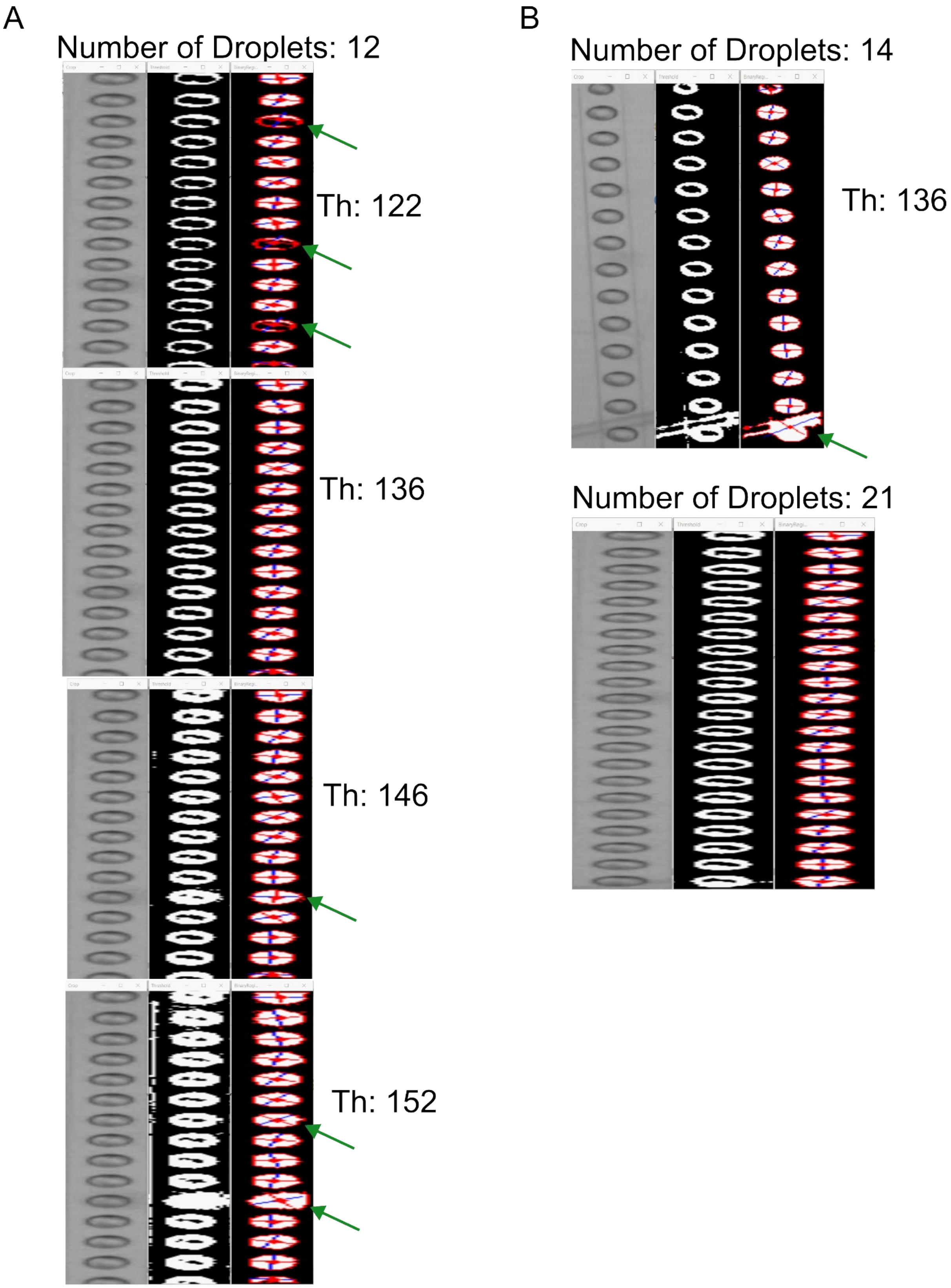
Supplementary Figure 7. Quantification of droplet radius, speed and droplet generation rate by different users using Bonsai.

	Flow (µl/min)		**Image J**						***Bonsai***						
ID Experiment	**Oil**	**Water**	**Radius (µm)**	**stdv**	**Speed (µm/s)**	**stdv**	**Freq (droplets/s)**	**stdv**	**Radius (µm)**	**stdv**	**Speed (µm/s)**	**stdv**	**Freq (droplets/s)**	**stdv**	**Total**
1	1.25	0.25	64.7005	3.50666	614.10	10.55	2.07	0.04	62.63948	0.8175	617.49	14.20	1.99	0.02	32
2	1.25	0.5	69.85876	3.71288	1032.09	10.08	4.00	0.05	71.53448	1.37094	1039.52	11.68	3.76	0.03	75
3	1.25	1	82.0245	2.25788	1536.19	365.37	6.34	1.80	84.07315	1.84121	1551.44	422.66	6.12	1.55	129
4	1.25	1.5	93.59667	7.26481	1604.70	15.52	6.81	0.39	96.26795	2.3841	1577.89	15.71	6.22	0.05	93
5	2.5	0.5	69.61021	4.7483	1053.80	33.94	4.09	0.12	69.87327	1.49203	1036.08	35.36	3.79	0.09	76
6	2.5	1	76.57924	4.62034	1598.65	42.99	6.99	0.26	78.8214	1.61302	1557.83	40.23	6.35	0.18	83
7	2.5	1.5	78.754	6.16459	2401.04	11.21	11.63	0.29	78.72582	1.18537	2490.49	22.89	10.77	0.09	129
8	2.5	2	80.44718	3.58136	2825.46	14.34	14.42	1.54	81.40987	0.96005	2770.36	27.70	12.67	0.14	177
9	5	0.5	43.75581	3.7774	4115.82	140.61	6.56	0.16	42.65912	0.65059	3928.17	110.19	6.26	0.08	81
10	5	1	49.3108	3.02261	4515.95	44.10	14.45	0.67	50.53434	0.89209	4433.86	52.87	13.86	0.43	222
11	5	1.5	55.59898	3.20432	5108.04	119.41	21.88	1.75	54.53234	0.89814	4992.67	76.56	21.31	0.94	426
12	5	2	54.36598	2.86684	5836.80	82.64	37.95	4.37	52.93843	0.54075	5438.83	50.80	32.01	0.15	576

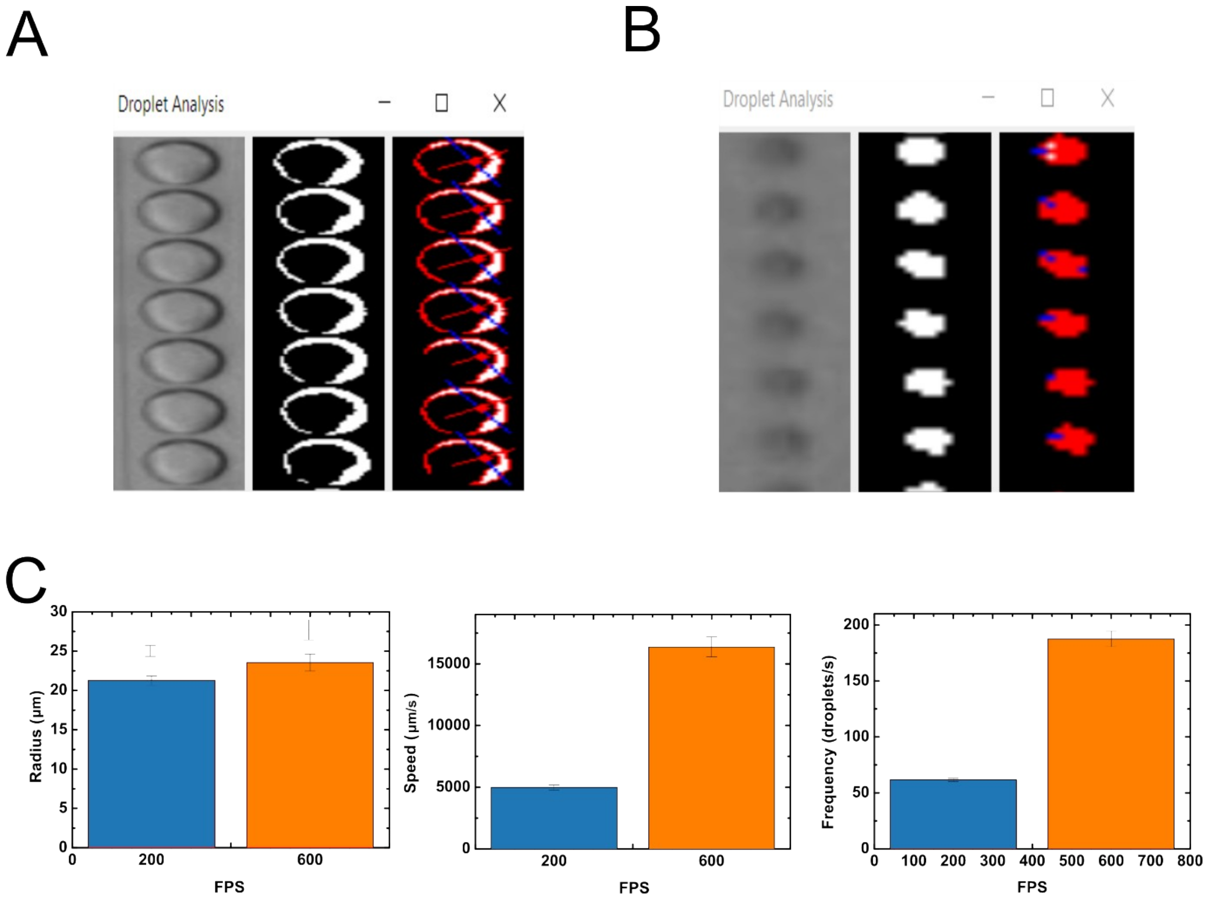
Supplementary Table 1. Quantification of radius, speed and generation rate (as frequency) using Bonsai and their comparison with Image J quantification. The movie ID and the flow rates for each experiment are presented. The average droplet radius, speed and frequency values are calculated with standard deviation (stdv) for each video. With the Bonsai workflow, we measured for the respective flow rates (µL/min) a certain number of droplets (total).



Supplementary Figure 8. Influence of number of droplets (i.e., ROI) and threshold value on *Bonsai* outputs. (A) Droplet radius, (B) speed and (C) generation rate for experiments 2 and 12. For the evaluation of number of droplets, the threshold was set at 136 and for the quantification of threshold, the ROI was defined with ≈ 12 droplets.



Supplementary Figure 9. Visualization of *'Crop'*, *'Threshold'* and *'BinaryRegionAnalysis'* outputs for a range of droplet numbers and threshold values. (A) Different threshold (Th) values with a fixed droplet number of 12, (B) The optimized threshold value for a high number of droplets (14 droplets for experiment 2 and 21 droplets for experiment 12). Green arrows identify artifacts.



Supplementary Figure 10. Challenging experiments for droplet analysis using Bonsai. (A) Experiment where droplets are close to each other (oil and water flow of 5 and 4 µl/min, respectively). Droplets with (B) small radius approximately 20 µm and (C) with high generation rate ≈200 droplets/s (oil and water flow of 20 and 4 µl/min, respectively). The blue bars correspond to a video where the camera frame rate was set to ≈200 fps and the orange bars correspond to a video where the camera frame rate was set to ≈600 fps.

Reference	**Detection Method**	**Label-Free**	**Difficulty to implement and use**	**Cost**	**Processing Speed**	**Comments**
3-4	Optical-Fluorescence	No	Medium	High	Medium	Need of fluorescent dye, expensive
5-8	Electrical	Yes	High	Medium	High	Complex, difficult to implement and expensive
9-12	Optical-Brightfield image	Yes	Low	Low	Medium	Limited processing speed (250 Hz)
13	Optical-Brightfield image	Yes	Low	High	High	Expensive
Our work	Optical-Brightfield image	Yes	Low	Low	High	Low cost, easy to use, fast image processing and analysis (≈7 ms @ 150 fps and ≈2 ms @ 611 fps)

Supplementary Table 2. Summary of existing methods and tools found in literature for real-time and automated droplet analysis.

1. Gonzalez, R. C. & Woods, R. E. *Digital Image Processing*. (Pearson, 2018).
2. Farnebäck, G. Two-frame motion estimation based on polynomial expansion. *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)* **2749**, 363–370 (2003).