Department of Electrical Engineering
Faculty of Physical and Mathematical Sciences
University of Chile

SCIAN-Drop & SCIAN-Force
Usage Instructions

The software was tested in Windows 11 with a 12th Gen Intel(R) Core(TM) i5-1240P of 1.70 GHz, 8 GB of RAM DDR4 without a dedicated graphics card. Thank you for using the software.

SCIAN-Drop

For a droplet inmmersed in a given medium, the interfacial tension γ can be obtained from the equation:

$$\gamma = \frac{\Delta\rho g Ro^2}{Bo}$$

$\Delta\rho$ is the mass density difference between the drop and the medium, g is the gravitational acceleration, Ro is the radius of curvature at the drop apex and Bo is the Bond number. For more information, see the SCIAN-Drop documentation and source code in the `GitHub` repository[1]. Its implementation was done entirely in Python, using different libraries detailed in the same repository. Please, provide the required input paramaters and image in the order shown below to avoid mistakes.

1. Input Parameters

The first input section is **"Parameters"** for image-independent parameters:

- droplet density, in $[\mathrm{Kg}/\mathrm{m}^3]$
- medium density, in $[\mathrm{Kg}/\mathrm{m}^3]$
- needle width/thickness, in gauge units

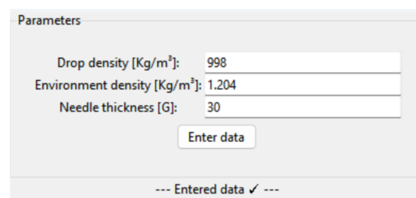


Figure 1: Example of first input data.

Once you entered de values, press the button *Enter data*. After visualizing the message *Entered data*, you can proceed to the next section. For example, under (approximately) normal conditions, a drop of water (density of 998 $[\mathrm{Kg}/\mathrm{m}^3]$) in air (density 1.204 $[\mathrm{Kg}/\mathrm{m}^3]$), hanging from a 30-gauge needle, should correspond to the parameter values shown in Figure 1.

2. Input Image

The following, you can select the image to use in the section **"Image"**. You can select an image that has been pre-processed. (binary format) or use the program processing. Check *Binary image* if you have the first case. For example, we have a binary image of a droplet whose Bond number is equal to 0.21. When loaded, it is displayed as shown in Figure 2.

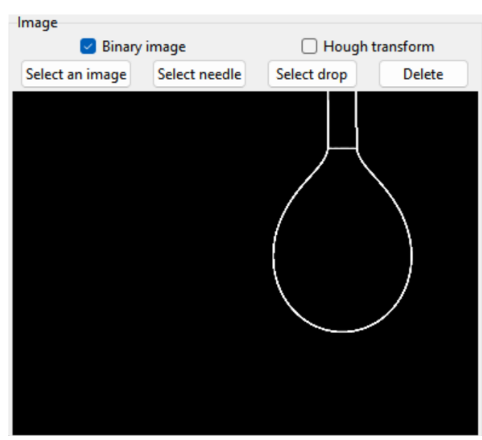
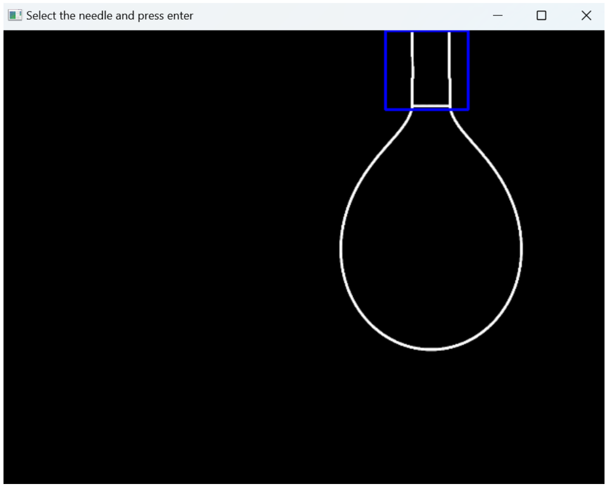


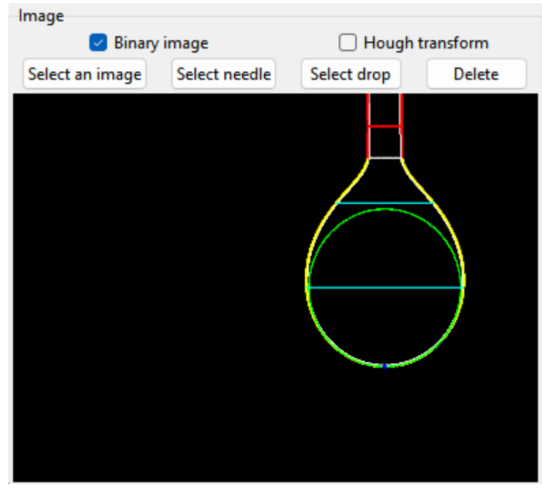
Figure 2: Visualization after loading an image.

Next, you will have to select the area where the needle and the drop will be located, through the respective buttons. The selection window should look like the Figure 3a. With the mouse, choose the region of interest from one end of the rectangle to the opposite end and press `Enter`.Before press *Select drop*, you can enable the *Hough transform* and the circle will be found using this method. It can differ from the classical method, which is based on finding "the larger circle, which passes through the *apex*, content within the droplet" and have a higher computational cost. It does not necessarily present better results. Finally, once you have both areas, it should look like the Figure 3b. In case of any correction, you can press *Delete* and make selections again.

[1] `https://github.com/scianlab/`



(a) Needle selection window. The other one is analogous.



(b) Expected result of the selections.

3. Results

If you completed everything correctly, at the top of the section **"Results"**, a message will be displayed to enable the option *Calculate*, like in Figure 4. Once you press this button, you will have the results of interest with their units of measurement, in the mentioned section before: the radius (`Ro`), the Bond number (`Bo`), the relation `Ds/De` (σ), the interfacial tension (γ), the Worthington number (`Wo`), the Neumann number (`Ne`) and the real volume of the drop (`V`). For more information about this data you can visit the repository. If you want perform a new calculation, you can press the button *New calculation.* y se reiniciará el programa. El resultado final se puede ver en la Figura 5.

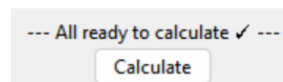


Figure 4: Message that you can perform calculations.

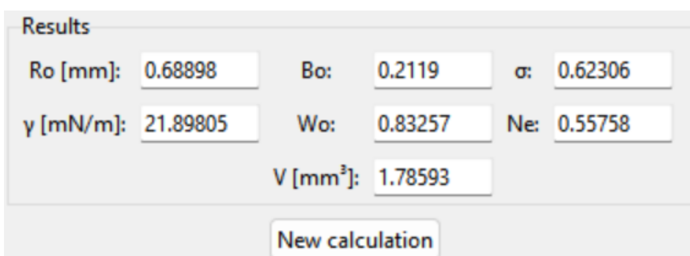


Figure 5: Final results

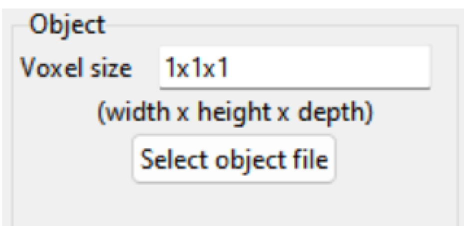
SCIAN-Force

Please, complete the information in the order shown below to avoid mistakes. The visualization of the curvatures is dynamic and interactive. The source code can be found in the repository of `GitHub`. Its implementation was done entirely in Python, using several libraries, which are detailed in the same repository.

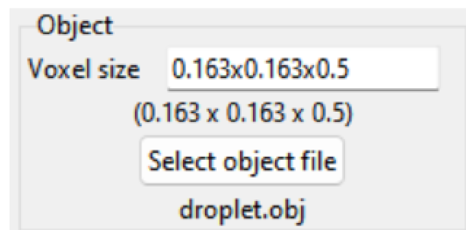
1. Input Object

In the first place, you will have to complete the section **"Object"**, which should look like Figure 6a. In it, you can choose the object to be worked with, whose supported formats are `.OFF` and `.OBJ`, pressing the button *Select object file*.It is possible that the size of the object does not correspond to the real one, therefore, you can modify this parameter through *Voxel size*, by entering the actual size that the voxel should have (equivalent to a pixel in 3D). The format it should have is '`width x height x depth`', can also be seen as the length in `x`, `y` y `z` respectively.

For example, a fluorescence microscope was used to obtain z-axis slices of a droplet. The pixels of each image, with a size of `0.163` μ`m x 0.163` μ`m` and the spacing between each slide is `0.5` μ`m`. With this, a 3D reconstruction was obtained, whose unit of measurement is in pixels, saved in a file '`droplet.obj`'. With this in mind, if the scale of the drop does not match the real one, the voxel size must be entered as a parameter and then the file must be selected, as shown in Figure 6b. Verify that everything has loaded correctly by displaying the voxel name (under the button) and size (in parentheses).



(a) First look of the section.



(b) Overview of the parameters input.

2. Colorbar

Corresponds to an optional section, where you can set the maximum and minimum value of the **"Colorbar"**. If any of the boxes are empty, the corresponding maximum or minimum value of the object's curvatures will be used. It can be useful when comparing two objects with different curvatures values. For example, if you set `1` y `-1` as the maximum and minimum value respectively, the color bar should look as in Figure 7. If this section is to be filled in, it must be done before using one of the methods in the next section. If you enter a range that leaves out data, the graph may not run. In general, warm shades indicate positive average curvatures, cool shades represent negative curvatures, and white represents curvatures close to 0.

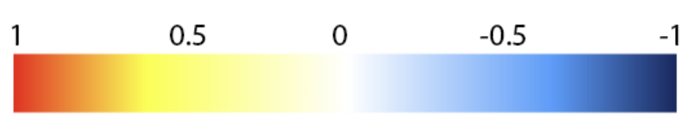


Figure 7: Color bar in a range of `[-1, 1]`.

3. Plot

After setting up the previous sections, you must use some method from **"Plot"** section to obtain the mean curvatures of the object. For this, press any button of interest. Each method has a different effectiveness and computational cost, *Rusinkiewicz Method* is recommended based on the algorithm of the library `trimesh2` in `C++`. Once selected, a window should open in your default browser, displaying the graph of the object like the Figure 8. Depending on your computer, the window may show an error. This is normal, run the method again.

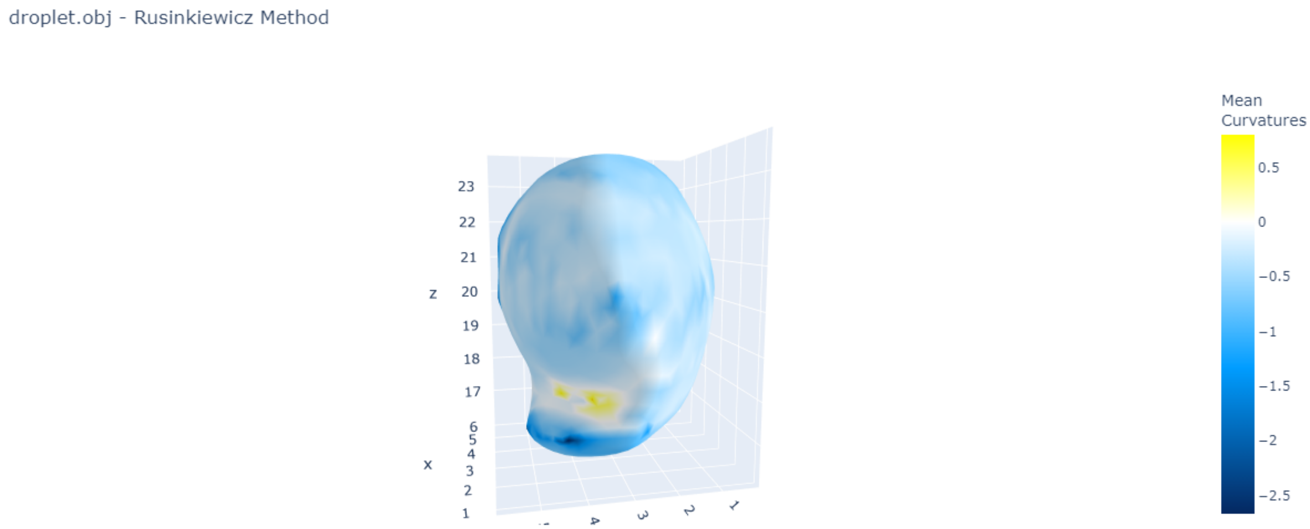
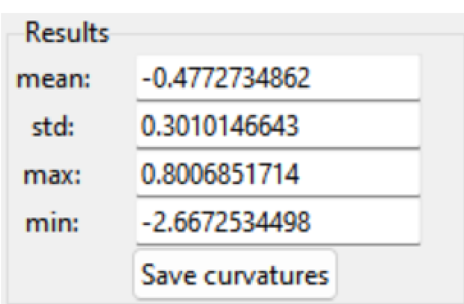


Figure 8: Ventana de la gráfica 3D del objeto.

4. Results

Once you have your graph, the results of the `average`, `maximum`, `minimum` and the `standard deviation` of the average curvature, can be seen in section **"Results"**. In addition, the results of the curvatures can be saved in an Excel sheet in the following format `.xlsx`, pressing the button *Save Curvatures* and selecting the location where you want to save the data. For example, considering the same object as above, your results look like in the Figure 9.



(a) Section with values of interest.

vertex	coord	mean_curvature
0	(1.8640703, 2.617475, 16.908533)	-0.857940698
1	(1.7479194, 2.746778, 17.035107)	-0.890668408
2	(1.7438341, 2.5823011, 17.14509)	-0.709433543
3	(1.8291264, 2.8992753, 16.835342)	-0.891858316
4	(1.7270557, 2.9733243, 17.050777)	-0.899881546
5	(1.8108804, 3.18366, 16.823418)	-0.897773537
6	(1.7290708, 3.271485, 17.056515)	-0.998422536
7	(1.8066688, 3.4850159, 16.830359)	-0.945152348
8	(1.7436371, 3.5775812, 17.058414)	-1.143457129
9	(1.810666, 3.7818148, 16.851383)	-1.078297986
10	(1.7623566, 3.8659396, 17.061516)	-1.450072653
11	(1.8194844, 4.0515995, 16.889565)	-1.340173902
12	(1.7836096, 4.1135063, 17.067543)	-2.129150575
13	(1.8329189, 4.2679462, 16.946222)	-1.452793364
14	(1.8091619, 4.2842627, 17.07038)	-2.66725345
15	(1.8518584, 4.393956, 17.012455)	-1.942827055
16	(2.0170295, 2.330897, 16.96003)	-0.675125371
17	(1.8129697, 2.4319077, 17.129547)	-1.253696028

(b) Excel with the information.

Figure 9: Final results