4AI000 Pendant drop tensiometry

M.J.C.B. van IJsseldijk and Huang, Shao-Kai
(TU Eindhoven)
(Dated: December 20, 2022)

In this paper, a two-step neural network is presented to predict the surface tension of droplets in the pendant drop experiment. To accomplish this, a convolutional neural network(CNN) is used to get a shape prediction from an input picture. The shape prediction is expressed in Chebyshev coefficients, as this always results in a continuous edge representation. The Chebyshev coefficients are then fed to a deep neural network(DNN) which estimates the surface tension from this shape prediction. It is found that the two-step model shows a maximum error in surface tension of 3.5% on experimental pictures, however the performance on synthetic droplets shows that only 56% of the pictures have an error below 5%.

I. INTRODUCTION

Surface tension is one of the key physical properties of fluids. It determines how thick a coating will be, or how well two fluids will mix, for example. For this reason, methods that can accurately determine the surface tension of fluids have attracted great research interest. One method to find the surface tension is the so-called pendant drop test, which is widely used due to its simplicity.

The pendant drop method works by generating drops using a needle of a known size and then capturing the shape of the droplet using a camera. After this picture has been made, the profile of the drop needs to be extracted. The most popular choice for this is a Canny Edge detector [1]. After the edge has been extracted, it is discretized to get points on the interface. Now that the shape of the droplet is represented in points, it is necessary to involve the physical equation that governs what a drop looks like. This is the Young-Laplace equation

$$\sigma\left(\frac{1}{R_1}+\frac{1}{R_2}\right)=\Delta p-\Delta\rho g z. \qquad (1)$$

Here σ represents the surface tension, R_1, R_2 represents the local curvature of a point on the surface, Δp represents the pressure difference across the interface and $\Delta\rho g z$ represent the hydrostatic pressure. In Fig. 1 a pendant drop can be seen with the coordinate system that will be used throughout this paper. Note that in the figure V represents the volume of the droplet, and r_{needle} the radius of the needle. The volume can be used as an alternative to the pressure difference in the Young-Laplace equation for a given r_{needle}. To go from edge points to σ it is only possible to solve it analytically in the case the droplet is a sphere, due to R_1, R_2 being derivates of the shape. This means that it needs to be solved iteratively using an optimization technique of choice. [2]

This method is implemented in an array of software toolboxes, but for this paper the implementation in ImageJ will be used as a baseline. ImageJ is an open-source software package that focusses on picture processing for experimental data. In ImageJ there is a plugin to estimate the surface tension and other parameters of a droplet.[3] It follows the method described before by first

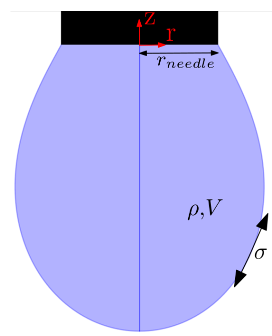


FIG. 1: Schematic of a pendant drop The annotations show the physical parameters and coordinate system

finding the edge points of the droplet and then using an optimization script (4th order Runge-Kutta scheme) to find the parameters of the droplet. The software has been proven to work accurately for an array of droplets, but it is not perfect. Due to it implementing an optimization scheme it can take around two seconds to predict the surface tension for a given image input. If a lot of pictures need to be evaluated, this adds up quickly. Furthermore, in order for the software to work correctly, some inputs are necessary from the user. First the pixel to length ratio needs to be calibrated as this can vary from picture to picture. Then the bounding box where the droplet can be found needs to be drawn. These steps add up in terms of time and are not without error. As an example of this, ImageJ was used to predict the surface tension of a particular drop. In the first example the bounding box was placed a few pixels under the needle, whereas in another picture the top of the bounding box was moved 9 pixels up (Appendix A. Fig. 11). It could be observed that the prediction of the surface tension changed from $\sigma = 27.7mN/m$ to $\sigma = 21.15mN/m$. Which shows that the method is not that robust.

To improve on the long prediction time, efforts have been made to replace the optimization scheme with a machine learning(ML) approach. For example, in ([4]) a DNN is used to estimate the surface tension from droplet edge points. The main conclusion of this paper was that the ML approach outperformed the optimization scheme in terms of accuracy for a large range of generated droplets. However, in this paper both for the training as the validation, generated edge points were used as the input for the DNN. This means that large errors in the edge point location could result in poor performance of the network. Furthermore, the performance of the network has not been validated on experimental pictures.

Nonetheless, it has been shown that applying ML techniques to find the surface tension in images could be beneficial. Therefore, in this paper it will be investigated if a neural network which finds a shape estimate based on a input picture, together with a DNN which estimates the surface tension from this shape estimate, still yields the performance in terms of speed and accuracy as in ([4]). Furthermore, robust performance for real experimental pictures will be investigated. To put this more formally:

Can a neural network architecture, which separately detects the shape of the droplet and the surface tension, result in a fast, robust and accurate pendant drop tension estimation from pictures?

To be more precise about the performance metrics, with *fast* is meant around $50ms$ prediction time per image, with *accurate* and *robust* is meant a maximum surface tension error of 5% on real experimental data.

II. METHOD

The combined neural network architecture will be called the two-step model. The first step will be called PictureNet as it will estimate the droplet shape from an input image. The second step will be called PhysicsNet, as it will predict the surface tension from the shape estimate. Furthermore, to evaluate the performance of the two-step model a one-step model will also be presented in this paper. This one-step model will remove the intermediate shape estimation step, which means it goes from picture to surface tension directly.

A. Data generation

The use of neural networks requires a large set of training data. To supply this, a MATLAB script which outputs a picture of a droplet with a certain σ and V is used. σ and V fully define Eq. 1 for a given ρ and needle radius. For this research, both ρ and the needle radius was kept constant, with $\rho = 1000kg/m^3$ and needle radius equal to $1mm$. To get a large range of droplets, σ is taken between $15mN/m$ to $75mN/m$ and V is taken between $5mm^3$ to $35mm^3$. It was made sure that a uniform distribution was obtained for the data, as this makes sure the model can predict each drop equally well. The distribution of the data can be seen in Fig. 2. To make sure only physically feasible drops were selected from this sweep,

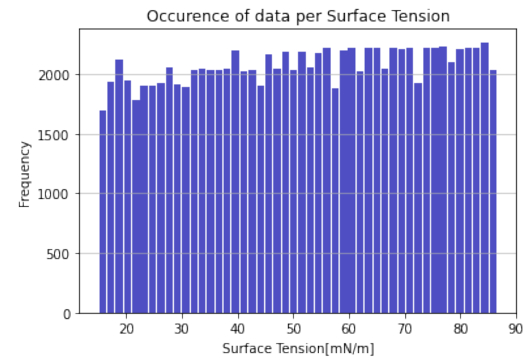


FIG. 2: Data distribution of the surface tension of the generated pictures.

two conditions were applied. Firstly, the pressure difference between the in- and outside of the droplet should be positive. Furthermore, for each point on the edge of the droplet, it must hold that the z-coordinate is decreasing from the needle to the apex of the drop. As an additional constraint, only droplets which had a Worthington number bigger than 0.4 were generated. The Worthington number is defined as

$$Wo = \frac{V}{V_{max}} \tag{2}$$

where V is the volume of the droplet and V_{max} is the maximum volume the droplet can be. For drops with a low Worthington number, it is hard to accurately estimate the surface tension [2]. This can be seen in Fig. 3 for various needle sizes. For a needle with a diameter of $1.65mm$ the surface tension estimate gets close to the real surface tension for $Wo > 0.4$. For this reason, only droplets with a Worthington number above 0.4 were considered.

Initially, 27000 pictures were generated to evaluate the network. However, to make the network more robust,

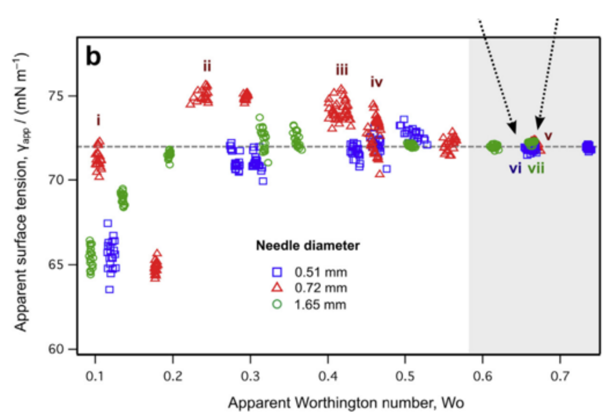


FIG. 3: Measured surface tension for pendant drop experiments plotted against the Worthington number.

data augmentation was applied for translations in the r and z direction. Besides this, also reflection effects were added to the pictures. These additional 80000 pictures, make the network more robust to translations and reflections that occur in real data. The initial dataset that does not use data augmentation will be called the raw dataset, while the dataset that does have data augmentation will be called the augmented dataset.

B. Droplet shape estimation

To estimate the droplet shape, two main methods could be used. The first method would be to estimate points that lay on the edge of the droplet. The second option would be to fit a spline to the edge of the droplet. To fit this spline, the Chebyshev series is used, as it can approximate any Lipschitz continuous function on a fixed interval. The series can be seen in Eq. 3.

$$f(x) = a_0 + a_1 T_1(x) + a_2 T_2(x) + \cdots \tag{3}$$

Here $f(x)$ is the output function, a_i are the Chebyshev coefficients and $T_i(x)$ is the i^{th} Chebyshev polynomial of the first order. To find the Chebyshev coefficients, Eq. 3 can be rewritten to

$$a_i = \frac{p_i}{N} \sum_{n=0}^{N-1} f\left(x_n\right) T_i\left(x_n\right), \tag{4}$$

which for a given function $f(x)$ yields the Chebyshev coefficients. To implement this for a droplet, the z and r position need to be approximated individually, as the Chebyshev polynomial only works for one dimension. The number of Chebyshev coefficients is free to chose, but for this problem $i = 10$ was found to be satisfactory. For the sake of clarity, the method that utilizes the Chebyshev coefficients will be called the spline method from now on.

C. PictureNet: picture to shape estimation

To estimate the shape of a droplet, a CNN will be used due to its proven effectiveness with feature extraction from pictures. This CNN will be referred to as PictureNet from now on. PictureNet has as an input the pictures obtained from the data generation. As an output, either the Chebyshev coefficients, or the edge points are used. Based on the VGG-based concept, the model uses four convolutional layers and three fully connected layers. The convolutional layers use a kernel of 3×3 and a padding of 1 pixel to make the size of the output consistent. Max pooling layers with a 2×2 kernel are applied to prevent over-fitting and to reduce the number of parameters to learn. Furthermore, batch normalization is implemented to increase the learning rate of the network. As an activation function, LeakyRelu is used. Finally, the loss function is defined as the mean-squared error(MSE) between the ground truth and the prediction. The full model structure can be seen graphically on the left-hand part of Fig. 4.

D. PhysicsNet: shape estimate to surface tension

Based on the estimated shape obtained from PictureNet, a surface tension prediction needs to be made. This will be done with a DNN that will be referred to as PhysicsNet from now on. PhysicsNet has as an input the shape estimation obtained from PictureNet, which can be edge points or Chebyshev coefficients. The DNN model has six hidden layers, of which the first two have numerous nodes (512) to give the network a lot of freedom. After these two layers, the network gradually converges the number of nodes to 2 to avoid having information loss between layers. At the first four hidden layers, dropout is applied to help prevent over-fitting. The last layer has two outputs. One for the tension σ and one for the volume V. The volume is used as additional information to compare and pass-through the network during back propagation. The model structure can be seen on the right-hand part of Fig. 4.

E. UnitedNet: picture to surface tension

Instead of having two separate neural networks, it is also possible to go directly from picture to surface tension. In this way, it is not possible to visualize what the network has detected in the image, but it may give more accurate predictions as you do not restrict the network to only find edge points. It could find other representations that are useful to predict the tension, for example. For the architecture of the one-step model, the networks of PictureNet and PhysicsNet are applied back-to-back. For this reason this network will be referred to as UnitedNet from now on.

III. RESULTS

To evaluate the performance of PictureNet, PhysicsNet and UnitedNet, the networks were trained with the Adam optimizer for 200 epochs, with a learning rate of 10^{-4}. After this, the networks were retrained using the SGD optimizer for 200 epochs, with a learning rate of 10^{-4} for the first 100 epochs and 10^{-5} for the last 100 epochs. This extra step was applied to overcome possible saddle points. During the training processes, a model selection function is applied that selects the weights and biasses that yielded the lowest MSE. This model selection helps to prevent over-fitting and to reduce the effect of training oscillations.

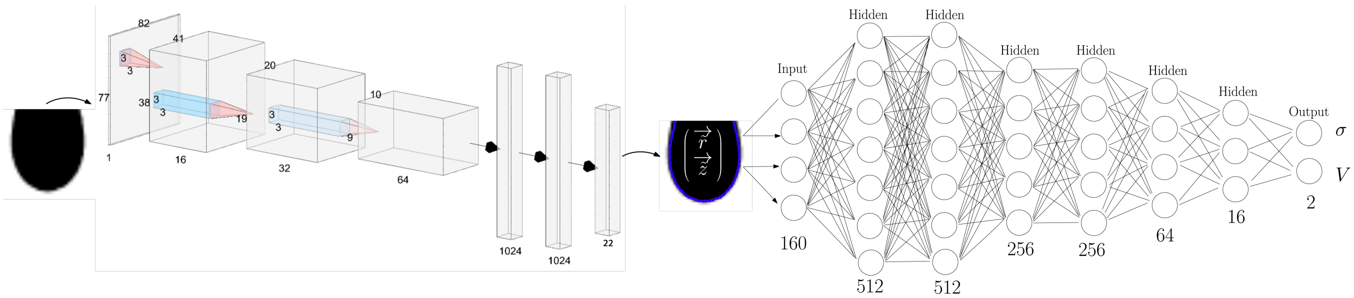


FIG. 4: Network architecture of the two-step model used in this paper. The left shows PictureNet which outputs the shape of the droplet, the right shows PhysicsNet which outputs the surface tension σ and volume V.

A. PictureNet

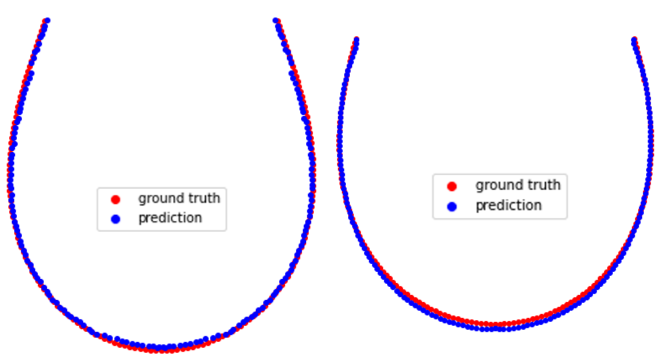
To evaluate the optimal architecture of PictureNet there will be investigated whether using edge points or Chebyshev coefficients returns the best shape estimate. To do this comparison, the model is trained on the raw dataset. In Fig. 5a and Fig. 5b, an estimated droplet shape is shown for either the point method or the spline method. It is observed that the shape returned from the point method is not continuous, while the estimate with the spline method is continuous. Besides the qualitative differences, also the l^2 norm of the prediction and ground truths are checked for both methods. For the point method, an average l^2-norm of 0.26 is obtained, while for the spline method, an average l^2-norm of 0.31 is obtained. This means that the point method has, on average, a better match to the ground truth than the spline method. However, the differences are not very significant in terms of l^2-norm, whereas the lack of continuity of the point method could give problems with the tension estimation. Based on this, it was determined that the spline method will be used as the output of PictureNet.

B. PhysicsNet

To evaluate the optimal architecture of PhysicsNet the effect of using edge points or Chebyshev coefficients as an input will be investigated. As has been discussed in Section A. PictureNet, Chebyshev coefficients will be used as an output. The Chebyshev coefficients can be directly fed into PhysicsNet or converted into edge points. For this analysis, 80 edge points (80 z-coordinates and 80 r-coordinates) were used for the point method.

Firstly, the stability of the MSE loss is investigated for both methods. in Fig. 6 the MSE loss per epoch is plotted for the point method and in Fig. 7 the MSE loss per epoch is plotted for the spline method. It is observed that the MSE loss oscillation is larger for the spline method than for the point method. Furthermore, the average MSE loss is lower for the point method.

Next, the accuracy of the tension prediction is checked for both methods. In Fig. 8 the score for the point and spline method are plotted for the raw dataset. It is observed that the point method returns a better score than the spline method. This observation combined with the lower MSE loss, means that the point method will be used as the input for PhysicsNet.



(a) Shape prediction for point method.

(b) Shape prediction for spline method.

FIG. 5: Shape prediction from PictureNet for either the point method or the spline method.

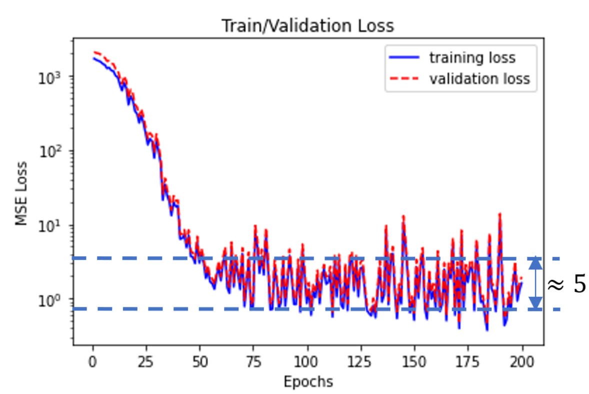


FIG. 6: MSE loss plotted per epoch for the edge point input.

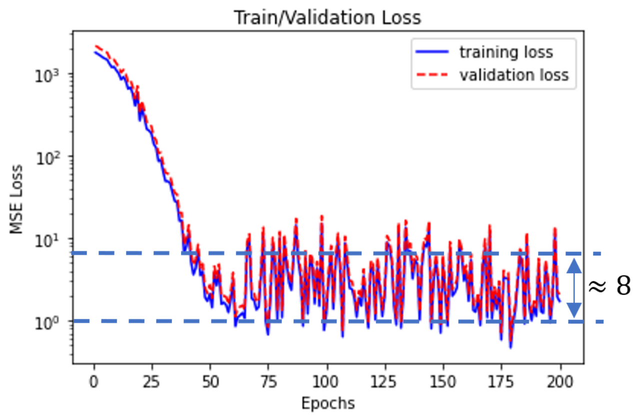


FIG. 7: MSE loss plotted per epoch for the Chebyshev coefficients input.

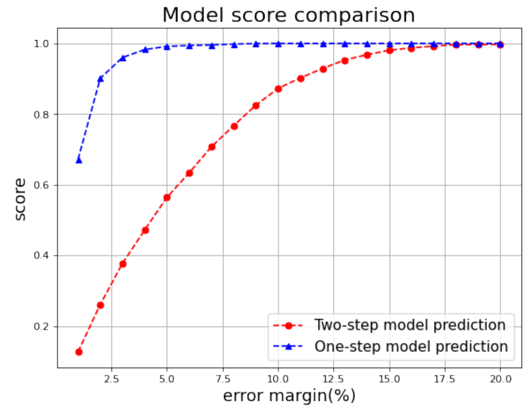


FIG. 9: Error margin for the one-step model and two-step model

C. Two-step model compared to one-step model

Now that the architectures of both PictureNet and PhysicsNet are determined, the sequential performance will be evaluated. This is done by training PhysicsNet and PictureNet on the augmented dataset. The performance is then compared to the UnitedNet which is also trained on the augmented dataset. For clarity, PictureNet and PhyscicsNet applied sequentially is called the two-step model and UnitedNet is called the one-step model. The two-step model was trained in 30 minutes and evaluates new data is 40ms, whereas the one-step model trained for 20min and evaluates new data in 20ms.

1. *Evaluation on test set*

The first step is to compare the one and two-step model on the test set of the augmented dataset. The result of this can be seen in Fig. 9. It can be seen that the one-step model shows a higher score than the two-step model for all error margins.

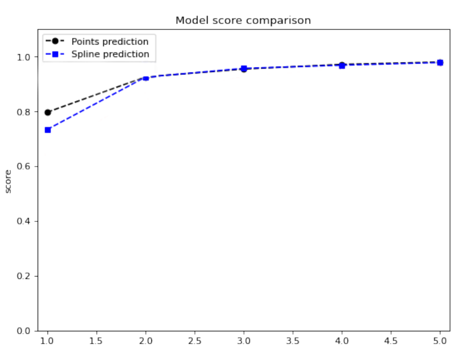
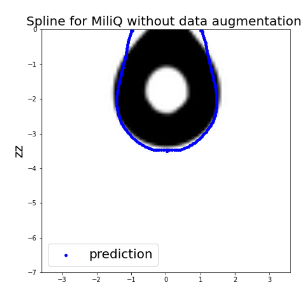


FIG. 8: Evaluation of PhysicsNet with different inputs.

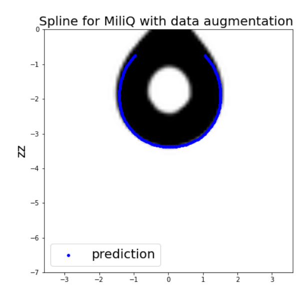
2. *Evaluation on experimental data*

The final step to see the performance of the two-step model is to predict the surface tension for real-world drops. The drops that will be predicted have been produced with a needle of $0.45mm$ radius, and for the fluids defined in Table I. As an example to show what the model is doing, the spline method prediction, both with data augmentation and without, will be shown for the MiliQ droplet. In Fig. 10a the predicted spline without data augmentation is shown, while in Fig. 10b the spline with data augmentation is shown.

It can be observed in Fig. 10a that the model incorrectly predicts the shape of the droplet, whereas in Fig. 10b it can be observed that the spline nicely follows the surface of the fluid, however it also stops following the surface of the droplet when $r = 1$ at the top of the droplet. The estimated surface tension for this, and the other droplets can be seen in Table I. The table contains the results for both the two-step model with and without data augmentation, the one-step model with data aug-



(a) Shape prediction without data augmentation.



(b) Shape prediction with data augmentation.

FIG. 10: Effect of data augmentation on shape prediction.

TABLE I: Tension prediction both with and without data augmentation for the two-step model and one-step model.

Material	Two-step model no augmentation $\sigma[mN/m]$	Two-step model augmentation $\sigma[mN/m]$	One-step model augmentation $\sigma[mN/m]$	Ground Truth $\sigma[mN/m]$
MiliQ	41.19	70.52	97.94	73.04
PDSM1	17.93	23.07	42.25	22.93 ± 0.02
PDSM30	19.98	23.19	43.55	22.12 ± 0.17
AH shampoo	19.35	24.90	44.80	25.7 ± 1.06
Castor Oil	27.8	39.3	66.08	39.05 ± 0.09
Glycerol	20.50	43.89	87.15	45.36 ± 2.06
PEG + 30wt% H20	24.33	51.07	89.60	52.07 ± 0.20

mentation and the ground truth. It can be seen that the maximum error of the two-step model with augmentation is 3.4% .

IV. DISCUSSION & CONCLUSION

It can be concluded that the two-step model that has been presented in this paper is, fast, accurate and robust according to the criteria that were set. Due to the nature of neural networks, once it has been trained, the evaluation of new data is quick. For the specific hardware used here, it takes 30 minutes to train PictureNet and PhysicsNet and 40 milliseconds to evaluate. The two-step model can also be described as robust and accurate if we compare the surface tension estimates obtained for real experimental droplets, as the maximum error is 3.5%. An interesting observation for the shape estimation of these drops is that the two-step model only fits part of the surface, but can still accurately predict the surface tension. This is due to the Young-Laplace equation being a local force balance, which implies that part of the droplet already has enough information to estimate the physical parameters. This good prediction is only possible due to the data augmentation, which makes sure the network does not overfit on the starting location of the droplet. Besides the good performance on the experimental data, it should be noted that the two-step model showed poor results on the test set of the augmented dataset. Only 56% of the pictures in that dataset were accurately predicted within the 5% error margin. This could be due to an error in the conversion from Chebyshev coefficients to edge points, but more likely this poor performance is a result of the error in shape estimation from PictureNet resulting in data that PhysicsNet cannot properly estimate. This could be improved by training PhysicsNet on the output of PictureNet instead of using the ground truth edge points.

The one-step model showed better results on the augmented dataset than the two-step model, but on the experimental dataset it performed much poorer. This likely occurs as the one-step model has only trained on pictures that have a needle size of 1 mm radius. This means that droplets with a different needle size will be estimated poorly. This can be greatly improved upon by training the network on various needle sizes.

Appendix A: ImageJ robustness test

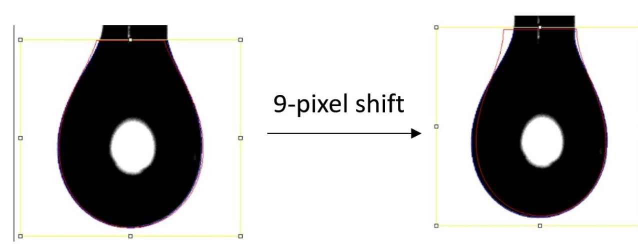


FIG. 11: Shape fit of a droplet in ImageJ. On the left side a correct fit is obtained, while for a nine-pixel shift upwards, the fit is poor.

[1] S. Wan, Z. Wei, X. Chen, and J. Gao, Pendant drop method for interfacial tension measurement based on edge detection, in *2009 2nd International Congress on Image and Signal Processing* (2009) pp. 1–5.

[2] J. Berry, M. Neeson, R. Dagastine, D. Chan, and R. Tabor, Measurement of surface and interfacial tension using pendant drop tensiometry, Journal of Colloid and Interface Science **454** (2015).

[3] A. Daerr and A. Mogne, Pendent_drop: An imagej plugin to measure the surface tension from an image of a pendent drop, Journal of open research software **4** (2016).

[4] F. S. Kratz and J. Kierfeld, Pendant drop tensiometry: A machine learning approach, The Journal of Chemical Physics **153**, 10.1063/5.0018814 (2020).