Supplementary Information

December 16, 2013

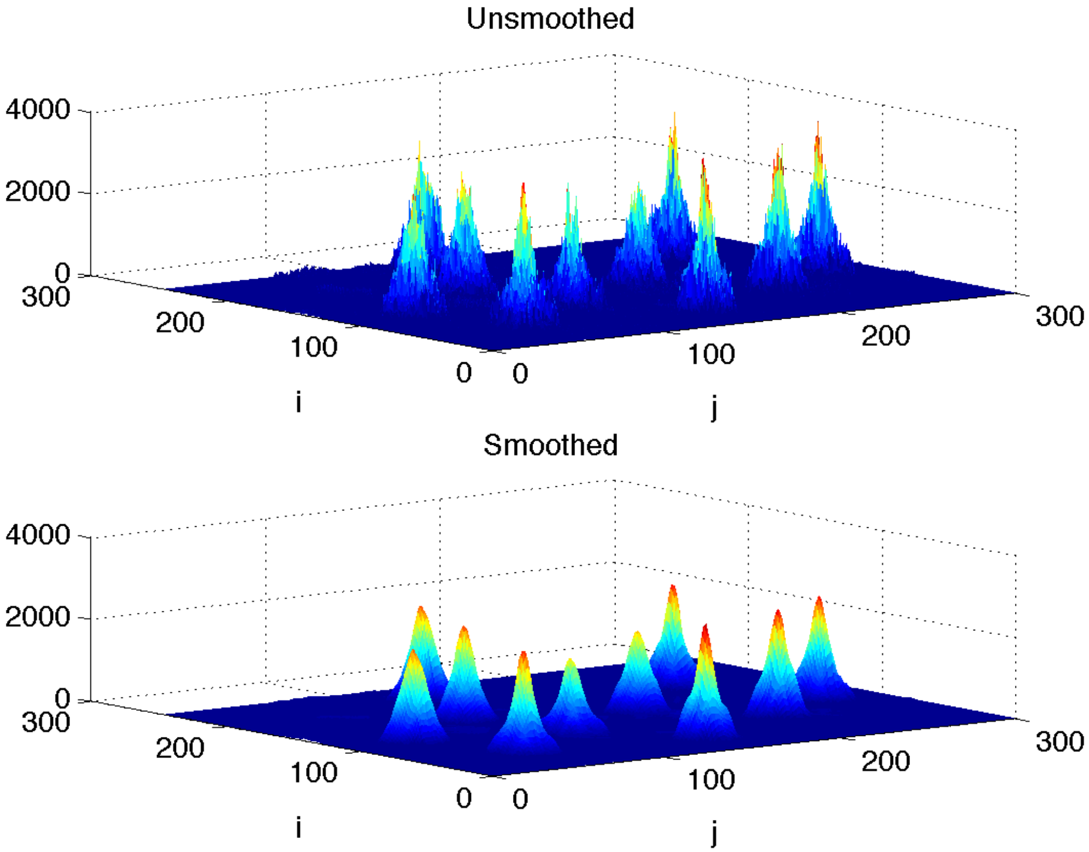


Figure S1: Top, the original accummulation array values. Bottom, smoothed accummulation array values. Smoothing is performed using a 5 × 5 averaging matrix

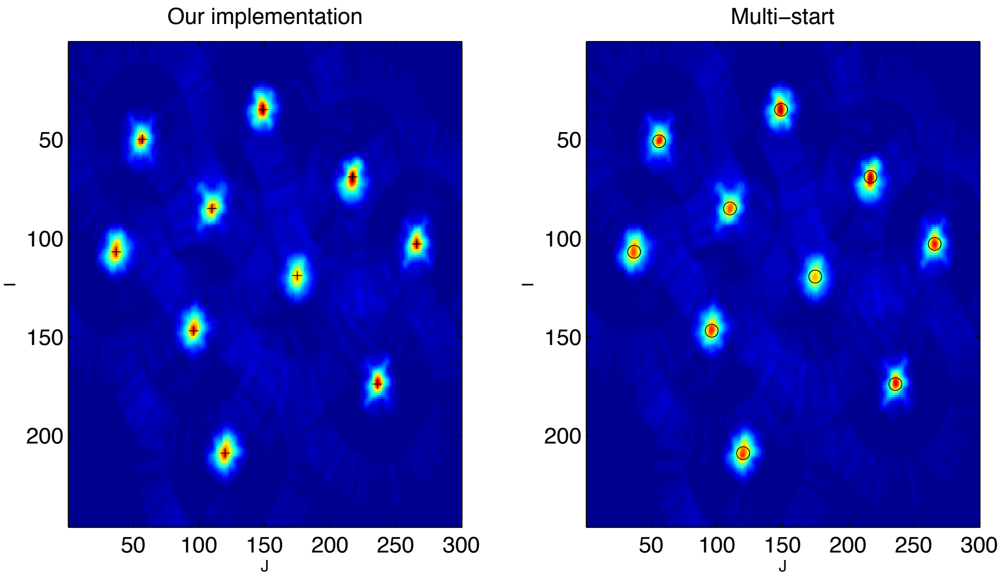


Figure S2: Left, black crosses mark locations of maxima obtained using Aguilera's code. Right, black circles mark locations of maxima obtained using MATLAB's multistart "fmincon"

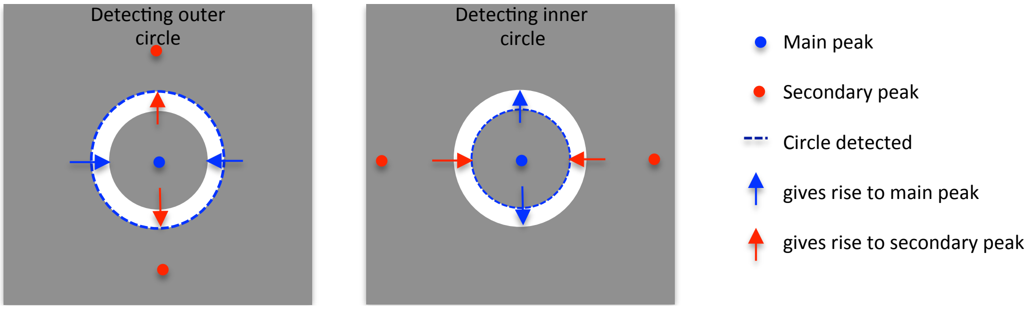
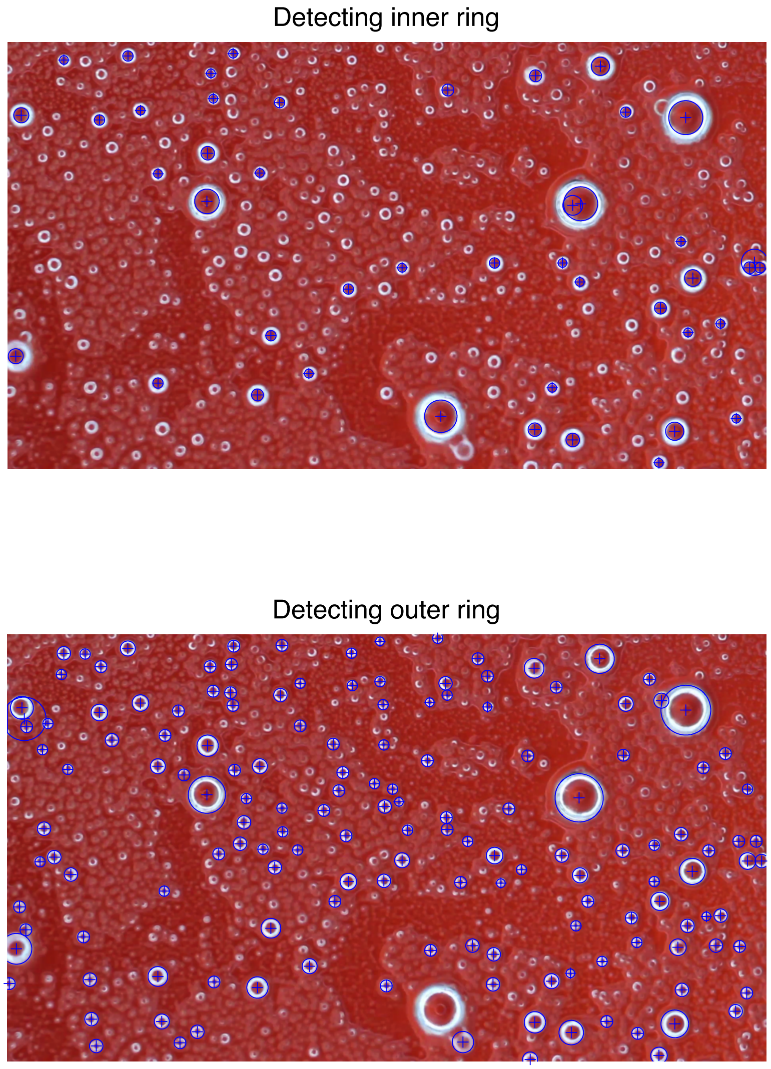


Figure S3: The gradient ∇I points inward at the outer edge, and outward at the inner edge. This allows us to detect either the outer or inner circles by incrementing only in the direction either parallel or anti-parallel to ∇I



Detecting inner ring

Detecting outer ring

Figure S4: Results of CHT for detecting either the inner ring (Top) or the outer ring (Bottom)

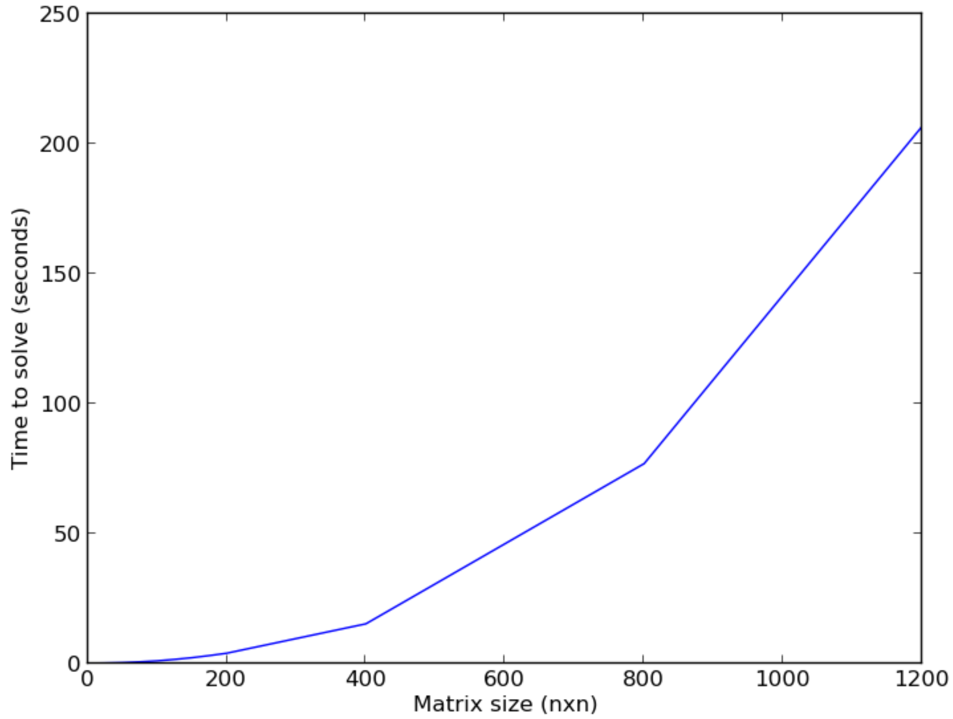


Figure S5: The figure above shows the running times for the GLPK solver on matrices as large as 1200 by 1200. While the GLPK solver, which is implemented in C, runs faster than our implementation in Python, it too runs in polynomial time.

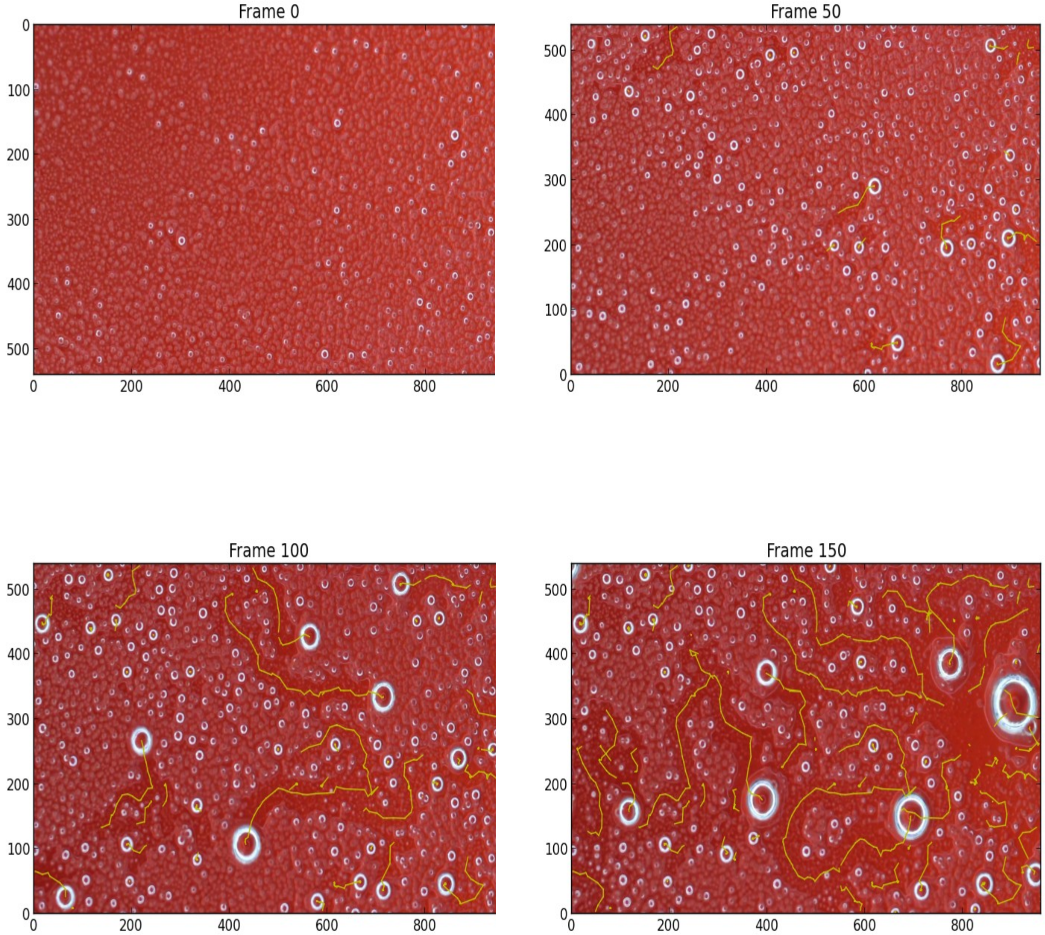


Figure S6: The figures above show the tracking program's results at various stages during the first 150 frames. At frames 100 and 150, you can see serveral lines with no droplets at the end which are the results of different droplets merging together. To see the animation, run the scipt track_droplets.py.