# TweezerTracker Reference

## Tracker Settings

The tracker settings directly influence how much accuracy you can get from your images, but also how fast the localization algorithm can compute positions. The internal details of the algorithms are beyond the point of this document (see [link](http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3353059/)) [[[1]](#footnote-1)]). However to use the tracking software it is important to know how the ROI size and settings influence accuracy and speed of the tracking algorithms.

Both QI and the Z Lookup table (ZLUT) algorithms will compute interpolated pixel values in a circle within the given images. The size and sampling density of this circle is defined by the settings that you assign, as shown in the image below. Sampling in polar coordinates could be easily defined by having number of radial steps and a number of angular steps, but this would typically have to be changed every time the camera magnification is modified (in which case the region of interest also needs to change). To prevent this, all tracking parameters are computed as a function of the ROI size.

D = ROI coverage \* ROI size

ROI (pixels)

Rmin

#Rbins

Rmin = MinRadius (pixels)

#Abins

ROI (pixels)

Sampling circle:

* MinRadius: Start radius of the sampling circle.
* ROI coverage: Converts the ROI size into the diameter of the sampling circle. Typically 0.8
* Radial sample density: Samples per pixel length in radial direction. Typically 2.5
* Angular sample density: Samples per pixel length in angular direction at the perimeter. Logically, sample density increases as you get closer to the center. Typically 0.7.

Other settings:

* QI iterations: Number of iterations of the QI algorithm. 4 gives good results.
* QI Angular Step Iteration factor: This is a trick to use lower angular sample densities in the first few iterations of QI. Every iteration, the angular sample density is multiplied with this value until it ends up being the value assigned by the user. It starts at (AngularStepDensity / AngularStepIterationFactor#Iterations). It gives a significant speed benefit without accuracy loss if set to 2.
* CUDA Device:
  + >= 0, select device with given index
  + -1: select device with the highest score (score = #Processor \* Clock frequency)
  + -2: use all CUDA capable devices, and distribute the tasks evenly. Dual-GPU cards like GTX 690 need this.
* Number of threads: Depending on CPU or CUDA tracker this means different things:
  + CPU: Number of threads (each running a tracker)
  + GPU: Number of streams

## Algorithm schematic

Center-of-mass

Quadrant Interpolation

Build ZLUT

Compute Z position from ZLUT

Center of mass provides a fast initial estimate

QI is ran for multiple iterations, typically 3

Z position is computed by looking up the radial profile in a precomputed lookup table

1. Non-Bias-Limited Tracking of Spherical Particles, Enabling Nanometer Resolution at Low Magnification (http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3353059/) [↑](#footnote-ref-1)