MicroscoPi Autofocus Algorithm Report:

This report is to highlight the progress that I made whilst working on developing and implementing an autofocus algorithm for the MicroscoPi unit over the 4 weeks that I spent working in the Micron research group. It is also intended to help succeeding contributors to pick up from the state that the project has arrived at.

Metric Details:

I have developed two metrics. A 'detail' metric, using edge feature analysis and a 'blur' metric, using Fourier spectra analysis.

The detail metric made use of the 'Canny' edge detection algorithm and assessed global detail in terms of local details, by considering rectangular sub arrays of pixels.

The blur metric isolated the Discrete Fast Fourier Transform in 2 dimensions. This worked by isolating the different levels/intensities of high/ and low spatial frequencies, contained within a series of concentric rings emerging radially from the centre of the image. More detail on this technique can be found in my PowerPoint presentation.

The derivation of the formulas that I used for these metrics, arise from the normalisation of the metrics to the interval [0,1] via a sigmoidal scaling function, accounting for the relative importance of either parameter on the apparent perceived blur. The power laws and constants used were just empirically derived to achieve a nicest fit to variation in the metric range that is most realistic for any sample slide images, i.e. the metric in reality is only realised outside of the extremities of the interval [0,1], despite the theoretical possibility of being at the extremities e.g. for a completely blank image. Images like these are not going to be observed in microscopy, hence the use of this scaling was invoked. More details of the parameters are explained in my PowerPoint presentation.

The metrics can be interpreted as providing a measure of the detail and blur at the current image focal. A measure closer to 1 indicates a high level of detail and a low level of blur whereas, a measure closer to 0 indicates a low level of detail and a high level of blur.

It appears that the two functions are somewhat complementary. The detail metric is very sensitive to being within a local discrepancy from absolute focus of about +-300 steps, where outside this area the metric extremely small. However, this metric falls short when within a narrower range from absolute focus at around +-100 steps.

In this narrower range from absolute metric, the blur metric performs much better. It follows that a hybrid approach of the two metrics would seem to perform well, using the detail metric to inform of local region of discrepancy from absolute focus and allowing the blur metric to take over to 'hone in' on the absolute focus.

Metric Performances and Runtimes:

For the detail metric, the size of the rectangular subarrays that is inputted (p, q) is the main factor effecting both the runtime and performance of this metric. It also needs to match up with the resolution of the input image, i.e. (p,q) should divide the resolution of the image (x,y). This could be something to alter, to compromise between runtime and accuracy. Although, accuracy may be less important if the proposed hybrid approach is used, since this metric would only be used as a distinguisher between whether we are sufficiently close to absolute tolerance to begin the switch to the blur metric.

A further improvement for the runtime of the both metrics would be to apply an effective crop, getting rid of all of the 'black space' outside of the PiCamera's circle of view. This could increase run time substantially since, there could be significantly less data to process, however, the camera would maybe have to be in a fixed position for this.

For the blur metric, this is particularly important, since any 'black space' outside of the PiCamera's circle of view will distort the frequency domain in the Fourier Transform. This is due to the contrast from the sample image to the 'black space' being associated with high spatial frequencies, when in fact this is not a feature of the sample but of the hardware setup.

Furthermore, if the image could be cropped to a size where its resolution dimensions are a power of 2, runtime will greatly be increased by an order of magnitude, since one can switch to the 'Fast Fourier Transform' regime for powers of 2. I assume that this feature is supported by the NumPy implementation.

Finally, an empirical decision on the number of rings to be used will affect both runtime and effectiveness, I have found that 25 rings seemed to perform well. It should be noted that the rings are set up to produce the input number of concentric rings each of equal area, emerging radially from the centre of the image until touching the smallest dimension of the image.

Autofocus Algorithm:

I didn't get much time to actually play with this, but the general idea was to drive the motor down to 'absolute zero' at the microswitch (which has been setup) and to then drive up to an underestimate of a reference position associated with the specific lens being used. Then, run the hybrid approach as detailed above.

One problem is false positives that may occur e.g. when there is dirt on the slide. one way to check for this is to once hit this (false) local maxima, check locally either side to determine if we are at absolute focus. If not, continue algorithm, until one reaches the true focal point.

William Manley
Email: william.manley01@gmail.com
Do get in touch if anything needs clarification, thanks.