Per-pixel sequencing

Kellen Dye

July 13, 2014

1 Background

We compare a per-pixel sequnecing method for in situ base calling in preserved tissue to a blob-based approch described in [1].

We attempt to isolate and sequence rolling circle products (RCPs) for four hybridization cycles. For each of the four hybridization cycles, the sample is imaged at each position of a 4×4 grid once for each base and once for a general stain.

A sample base image is given in figure 1.

2 Pre-processing

Input data is a set of Zeiss `zvi` files produced by a microscope imaging device.

ImageJ was used to extract 16-bit `tiff` images from the input files. Each 16-bit image was then converted to an 8-bit `tiff` using 14 bits of the original 16 in order to match the data set used in [1].

At each position, we produce a maximum intensity projection (MIP) for each hybridization cycle. We also transform each base image with a morphological top-hat filter with a kernel size of 4. A before/after comparison of the images enhanced with the top-hat transform is given in figure 2.

3 Registration

At each position, the MIP of the first hybridization cycle serves as a basis for registration and the MIPs of the other hybridization cycles are registered against it. The registration transform is then applied to the enhanced versions of the individual base images. We have used Random Sample Consensus (RANSAC) with Speeded-up Robust Features (SURF) as the registration method. A before/after comparision is given in figure 3.

Using RANSAC on the general stain, however, produced poor results. We therefore use adaptive thresholding to find bright points in the general stain, then

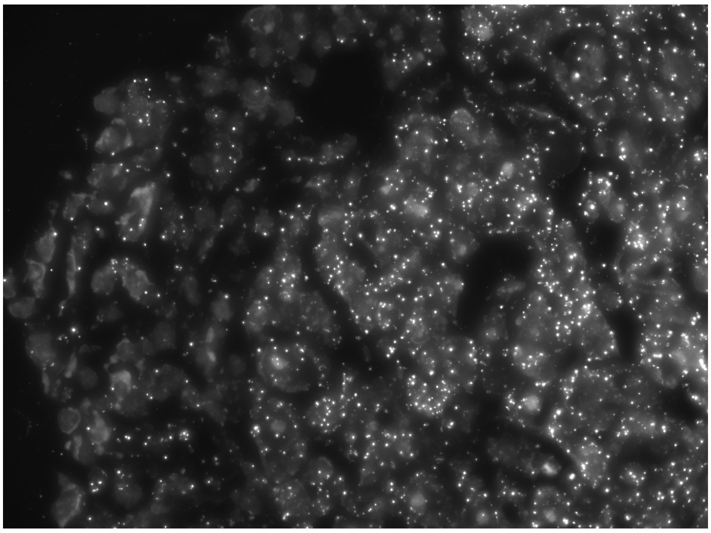

Figure 1: A sample base image

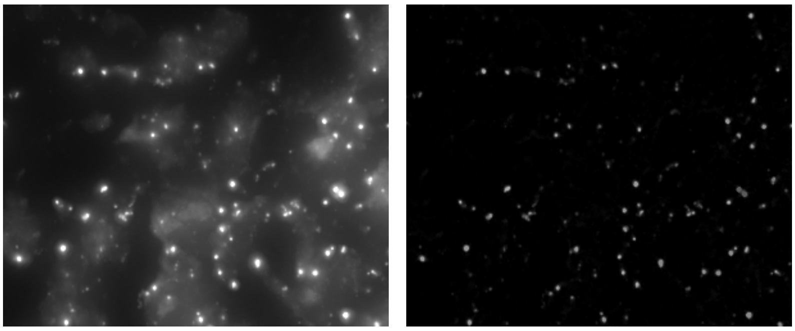

Figure 2: Before/after top-hat transformation

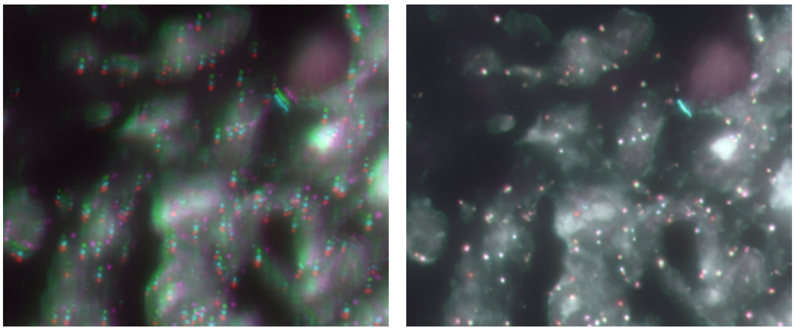

Figure 3: RANSAC registration

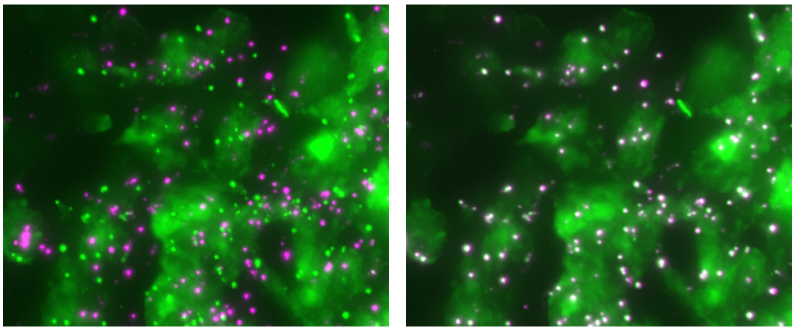

Figure 4: CPD registration

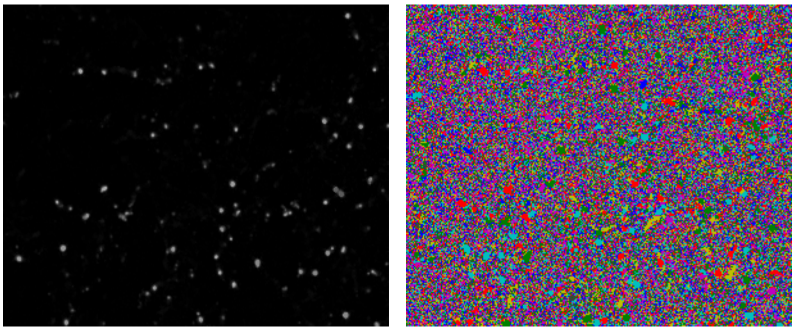

Figure 5: Sequenceing result

apply Coherent Point Drift (CPD) [2], a point-based method, for registration of each hybridization cycle's general stain. A before/after comparison is given in figure 4. We were not successful in registering the general stain for some positions of our sample (positions 5, 12, 13, 14); we discuss some potential improvements to registration of the general stain in section 9.

4 Per-pixel sequencing

We sequence the entire image on a per-pixel basis, where the sequence is defined as the concatenation of the base which gives the highest intensity response in each hybridization cycle:

$$seq(x,y) = \mathop{\Big|\Big|}\limits_{\substack{c\,\in\\ \{1,2,3,4\}}} \mathop{argmax}\limits_{\substack{b_c\,\in\\ \{A,C,T,G\}}} I_{b_c}(x,y)$$

Figure 5 shows the result of sequencing compared with one of the enhanced base images.

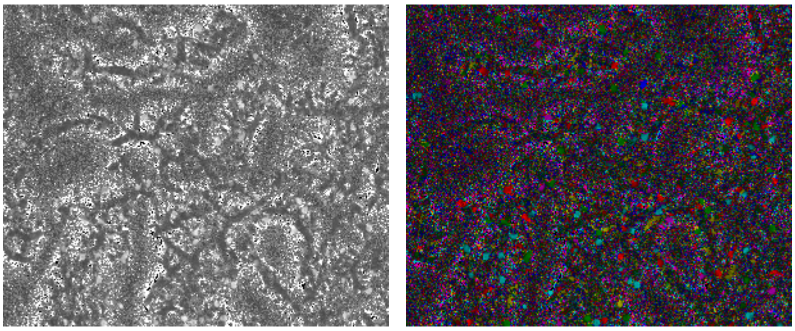

Figure 6: Quality

5 Object detection

For each sequence label we perform watershed segmentation to split touching blobs. We then locate the region centroids and use this as the object location.

Before object detection, we apply thresholding to some pixel-based features and to some region-based features.

6 Features

In order to identify objects in the sequence images, we define some per-pixel and some region-based features which can be used to exclude background pixels from our result.

Each pixel is evaluated with a quality measure. For each hybridization cycle, the proportion of the total intensity at that pixel position given by the "best" base is calculated. The quality is then defined as the minimum of these values over all hybridization cycles.

$$quality(x,y) = \underset{\substack{c\,\in \\ \{1,2,3,4\}}}{min} \frac{\underset{\substack{b_c\,\in \\ \{A,C,T,G\}}}{max} \quad I_{b_c}(x,y)}{\sum\limits_{\substack{b_c\,\in \\ \{A,C,T,G\}}} \quad I_{b_c}(x,y)}$$

Figure 6 shows the overall quality and the quality overlaid on the sequence image. Figure 7 shows the number of objects and the method precision versus a quality threshold. Precision is determined using a list of known sequences; any objects with a sequence not on this list are considered errors.

We calculate the average intensity for the decided sequence on the premise that if the sequence as a whole has very low intensity it may well be background noise.

$$avg(x,y) = \frac{1}{|c|} \sum\limits_{\substack{b_c\,\in \\ seq(x,y)}} \quad I_{b_c}(x,y)$$

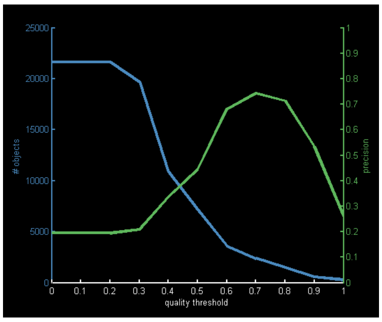

Figure 7: Quality

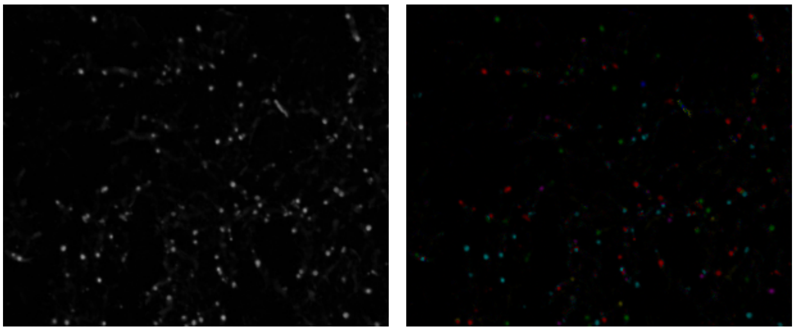

Figure 8: Average intensity

Figure 8 shows the average intensity for each pixel sequence and this overlaid on the sequence image. Figure 9 shows the number of objects and the method precision versus a threshold on average intensity.

The maximum intensity for the sequence is also calculated.

$$maxIntensity(x,y) = \underset{\substack{b_c \in \\ seq(x,y)}}{max} \quad I_{b_c}(x,y)$$

Figure 10 shows the maximum intensity for each pixel sequence and this overlaid on the sequence image. Figure 11 shows the number of objects and the method

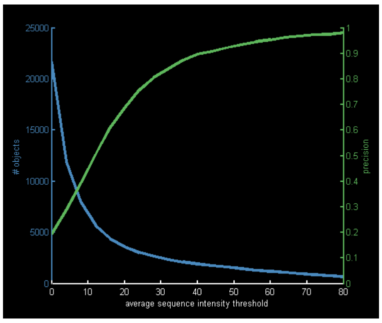

Figure 9: Average intensity

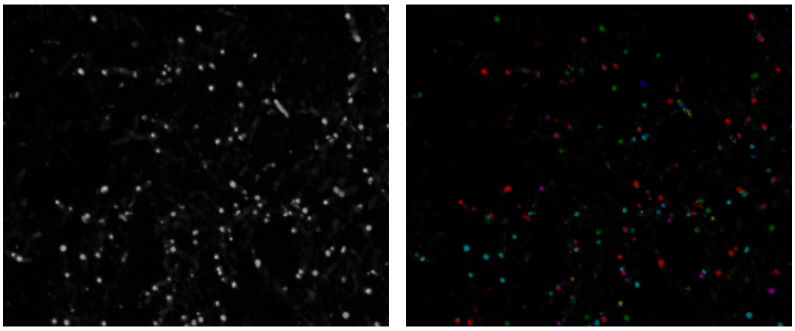

Figure 10: Maximum intensity

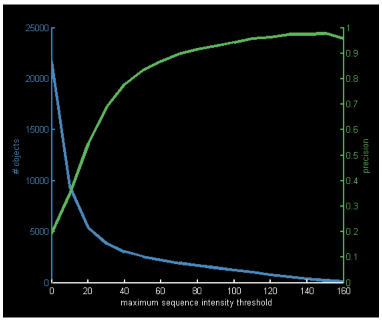

Figure 11: Maximum intensity

precision versus a threshold on maximum intensity.

The size of a connected region is also considered with both very large and very small regions potentially being background regions. Figure 12 shows the number of objects and the method precision versus a lower threshold on region size. Figure 13 shows the number of objects and the method precision versus a upper threshold on region size.

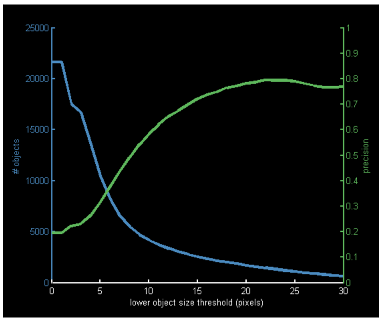

Figure 12: Region size: lower threshold

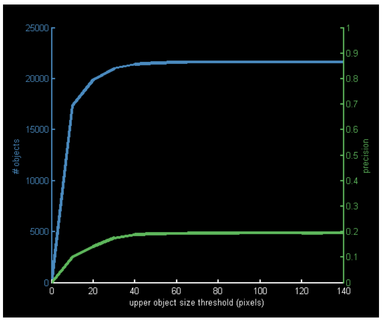

Figure 13: Region size: upper threshold

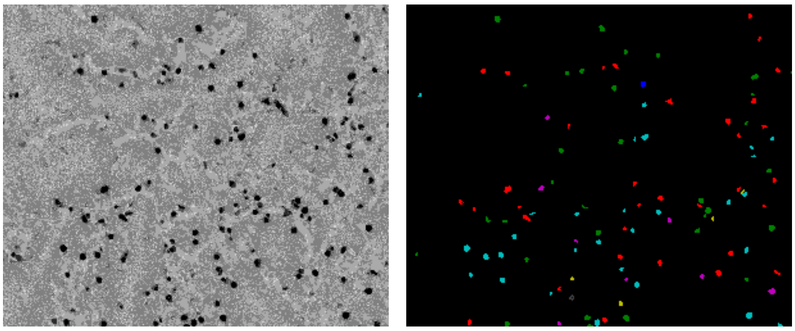

Figure 14: Exclusions, average contributions

7 Method

At each pixel position, the sequence is then determined and the additional features described above (quality, average intensity, maximum intensity) are calculated.

To find the RCPs, we exclude:

- a-priori known error sequences (AAAA, CCCC, GGGG, TTTT)
- regions outside of an inclusive binary threshold of the general stain
- pixels whose average intensity is < 25
- pixels whose maximum intensity is < 40
- pixels whose quality is < 0.475
- regions whose size is > 80
- regions whose size is < 5

Figure 14 shows (left) for each pixel the number of these exclusions which apply where lighter pixels are excluded by more of these thresholds; this figure also shows (right) the remaining pixels and their sequences after all of the thresholds are applied.

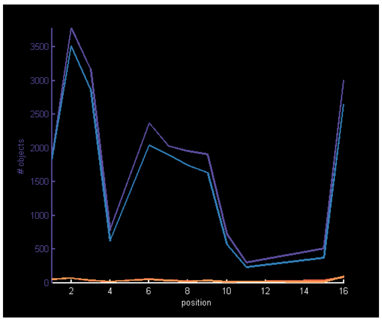

Figure 15: Comparision: number of objects

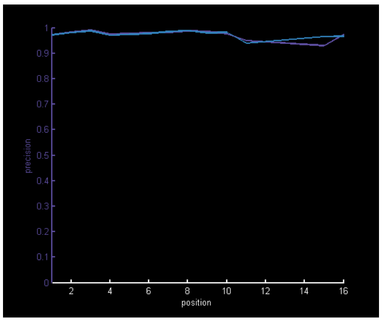

Figure 16: Comparision: precision

8 Comparison with existing blob-based approach

Our approach could function without using the general stain at all, but the blob-based approach relies on it in order to define object and background regions. This comparison therefore ignores positions 5, 12, 13, and 14.

Our registered and top-hat transformed images were fed into the portion of the CellProfiler pipeline responsible for sequencing and the results compared with our method.

Figure 15 shows the number of correct versus the number of incorrect objects identified for the two methods. The dark blue and dark orange show the blob-based approach while the lighter blue and lighter orange show the per-pixel approach.

Figure 16 shows the precision for the two methods. Again, The dark blue shows the blob-based approach while the lighter blue shows the per-pixel approach.

Figure 17 shows position 9. Crosses are object positions for the blob-based approach while circle are object positions for the pixel-based approach. Blue objects are valid sequences while magenta objects are errors. The first closest pairs of objects between the blob and pixel-based approaches are connected by lines. Green lines are matching sequences while red lines are sequences which do not match. If an object identified by the pixel-based approach was not paired with a blob-based object, it is colored green if it is a valid object and red if it is

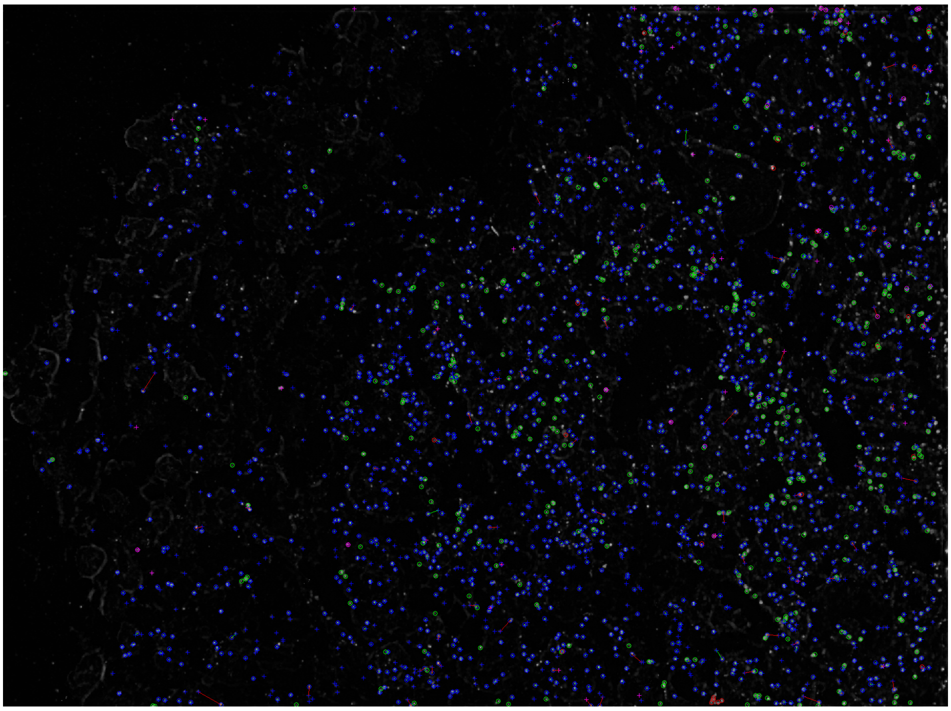

Figure 17: Comparision: points

an error.

9 Potential improvements

9.1 Touching blobs

One of the computationally expensive parts of our method is using watersheds to split blobs of the same class since the watershedding is run once per class. An alternative method could consider blobs as being the same as a connected components. In many cases, there are few touching blobs of the same class, so the results given by omitting the watershed procedure would be substantially the same.

9.2 General stain

The registration of the general stain presents several issues. The general stain can differ greatly from the base images and from the maximum intensity projection for a hybridization cycle; in these cases it is not possible to use RANSAC to match points.

The general stain may have very few points. In this case we have not been able to get correct results with gradient descent, RANSAC, or coherent point drift.

The general stains can differ between hybridization cycles, generally getting more intense with each cycle and including more background areas with high(er) intensity. In order to use CPD, relatively consistent point sets are required. Our method for isolating these points is to use an adaptive threshold, but this thresholding is affected by the intensity changes and therefore does not necessarily produce consistent sets of points.

A potential solution to these issues is to register the adjacent general stains from the *same* hybridization cycle, since their intensity characteristics are more uniform. Once adjacent general stains are registered with each other, a point set for the the entire sample in a single hybridization cycle could be produced, which could then be registered against other hybridization cycles. In order to do this, we would need to configure the imaging process to produce a larger area of overlap between imaging positions than our current data set; the current data set has too few points in common between adjacent imaging positions to produce consistent results. There is still the potential for these overlapping areas to be incorrectly registered if they do not contain enough information.

9.3 Rigid registration

The implementations of RANSAC and CPD used for Matlab are affine and can therefore introduce shear or scaling during registration. Though this can help in some cases, the positions which were not successfully registered were very sheared and/or scaled, so we speculate that the inclusion of shearing/scaling may have resulted in worse registration than a fully rigid method would have.

References

[1] R. Ke et al. “In situ sequencing for RNA analysis in preserved tissue and cells.” In: *Nature methods* 10.9 (2013), pp. 857–860.

[2] A. Myronenko and X. Song. “Point Set Registration: Coherent Point Drift.” In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 32.12 (Dec. 2010), pp. 2262–2275.