Xenopus Image Surface Mapping and Dynamic Co-Expression Analysis

Andrew Montequin, LaBonne Lab

Updated January 2026

1 Introduction

This protocol provides a how-to guide for surface mapping and analyzing ectodermal co-expression patterns in *Xenopus* confocal microscopy data. Example images of *pax3* and *snai2* co-expression at Nieuwkoop & Faber stages 13 and 15 can be downloaded from the Xenbase data repository, found at `https://bigfrog.xenbase.org/pub/xenbase/`.

2 Protocols

Surface Mapping

The surface mapping protocol makes use of the Image Surface Analysis Environment (ImSAnE) developed by Heemskerk and Streichan (2015)[1]. You can download the MATLAB package at `https://github.com/idse/imsane/tree/master`. Note that installation requires a C++ compiler such as Xcode for Macs.

The MATLAB code begins by setting file directories and image metadata, which should be changed for each image. We only used the ImSAnE code for fixed time-point imaging, but it contains options for mapping live-imaging data. ImSAnE also offers several options for the surface detector and fitter types. We used the Planar Edge Detector and Thin Plate Spline Fitter for all of our images, but other detector and fitter combinations may be suitable for different data types. Different surface fitter options are also coded for different mapping techniques. We used x-y mappings that assigned a surface value to every x-y pixel in the original image, but options are available for conformal (angle-preserving) and equidistant (distance-preserving) mappings.

Surface detection uses manually-selected parameters to select one or more image channels for surface detection, blur the channels, apply intensity filters and remove height outliers from the surface represented as a point cloud. The code allows the user to inspect the detection given specific parameters, and later the point cloud representation of the detected surface.

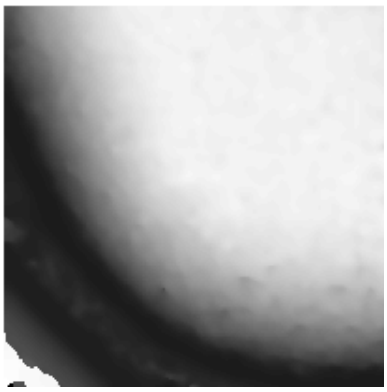

Figure 1: Surface detection example using relatively low intensity thresholds. The real image includes negative space in the bottom left corner, but the low threshold means we spuriously detected a surface there.

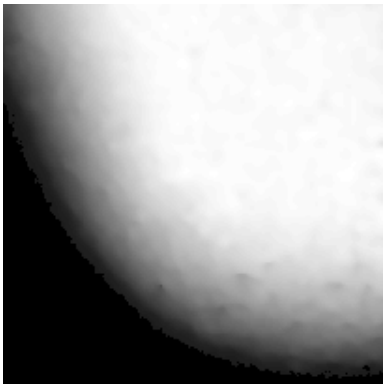

Figure 2: Raising the intensity threshold and re-detecting the surface removes the false positive detection in the bottom left corner.

After fitting a smooth surface to the point cloud, ImSAnE allows the user to inspect the surface fit by slicing the image along a user-defined axis. You can also shift the surface in the normal direction up or down. In our examples, the surface detection assigns the surface location to nuclei in the surface ectoderm. Given the resolution of our images, we found that a shift parameter of -8 centered the final mapped image in between the first and ectodermal cell layers.

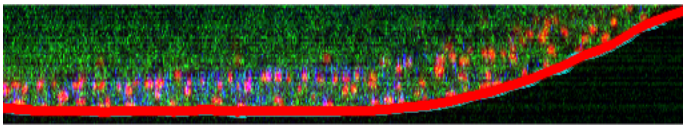

Figure 3: The initial fit surface is overlaid in red on top of an image sliced down the x-axis

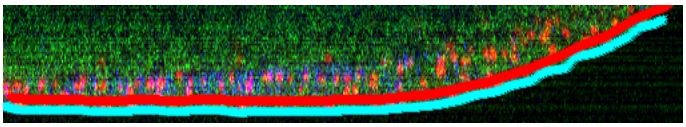

Figure 4: The initial fit surface is overlaid in blue, while the shifted surface is overlaid in red.

Care should be taken at the point of image collection to collect images that can be properly surface mapped. In the example below, the original z-stack image did not have any negative space above the embryo surface. In this case we used the Planar Edge Detector, which assigns the surface location to the point in the stack with the highest intensity gradient. While this typically occurs when the image shifts from negative space to the outer edge of the embryo, this shift was not present and the detector failed. It is possible that another surface detector available through ImSAnE could be adapted for this image. However, since we wanted to keep the same settings for all images, we did not use this particular image for any downstream analysis.

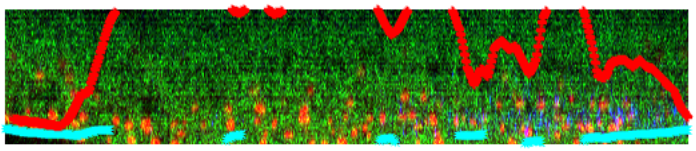

Figure 5: Example of a fit error, when no negative space above the embryo surface is present in the image.

The final step in surface mapping is generating the "pullback," when the detected surface is flattened. This can be done to make one image slice through the surface, or to pull back an entire z-stack. We chose the latter options for our images so that we could analyze both the superficial and deep ectodermal cell layers. We found that pulling back 31 layers of the surface (e.g. the fit surface, 15 layers above that surface, and 15 layers below) captured the appropriate range of the ectoderm for our purposes.

After the pullback, the user can manually inspect distortion introduced into the image. ImSAnE generates a strain map, where blue bars indicate the magnitude of compression and red indicates expansion, each scaled by a user-defined factor. If

desired, ImSAnE also generates the metric tensor which can be used for measuring specific distances or areas within a mapped image.

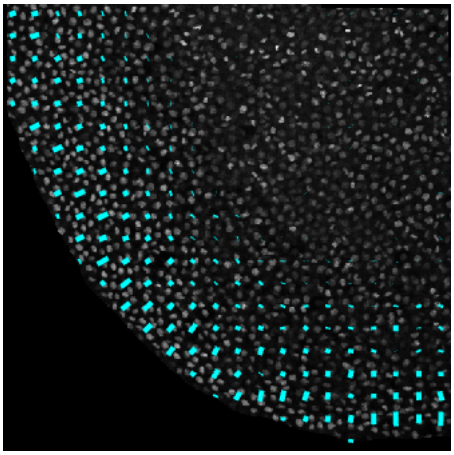

Figure 6: Example of a distortion map. Blue bars indicate direction and magnitude of compression, scaled by a factor of 30.

Co-expression Analysis

Surface mapping allows for comparisons of the same ectodermal cell layer across embryos and images that may vary in surface morphology and stack size. We used mapped images and analysis packages in Python to analyze how intensity measurements varied in selected regions of the neural crest. Our provided code includes functions to import images,[2] measure line intensity profiles, and align/plot these profiles across images. In the example, we manually drew lines through anterior and posterior slices of the neural crest marked by *snai2*.

To measure intensity profiles, we selected points anchor points surrounding the medial and lateral boundaries of the neural crest in both the anterior and posterior. The provided code will display the image in an interactive window and allow the user to click on the image to define medial anterior, lateral anterior, medial posterior and lateral posterior points. After clicking to select the first point, the user should then click on the legend in the top right to advance the prompt before selecting the next point. Using these user-selected points, we then apply the `measure_line_profile` function to measure the fluorescent intensities through the lines connecting those points. We attempted to choose points at the same anterior-posterior level across all images, although this is an area where human error can increase noise in the final analysis.[3]

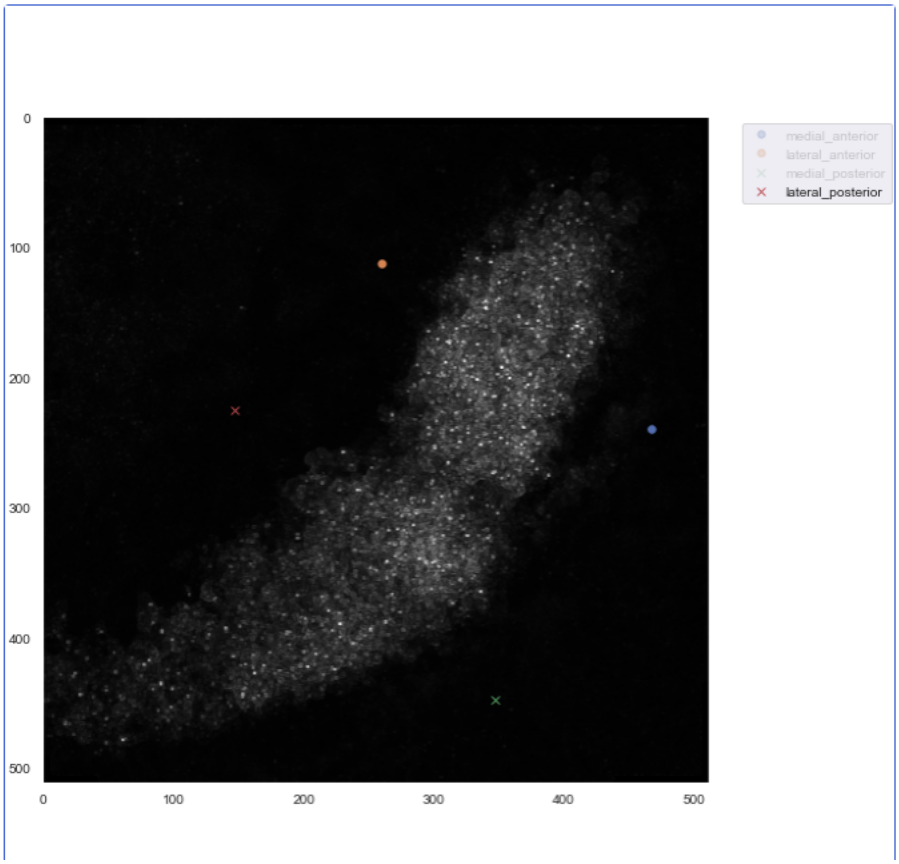

Figure 7: Example of point selection in a *snai2* image.

One important detail to note is that all chosen points should be outside of the *snai2*-positive region, so to have a region of "background" on each side. We did not include images in our analysis where both the medial and lateral boundaries were not visible.

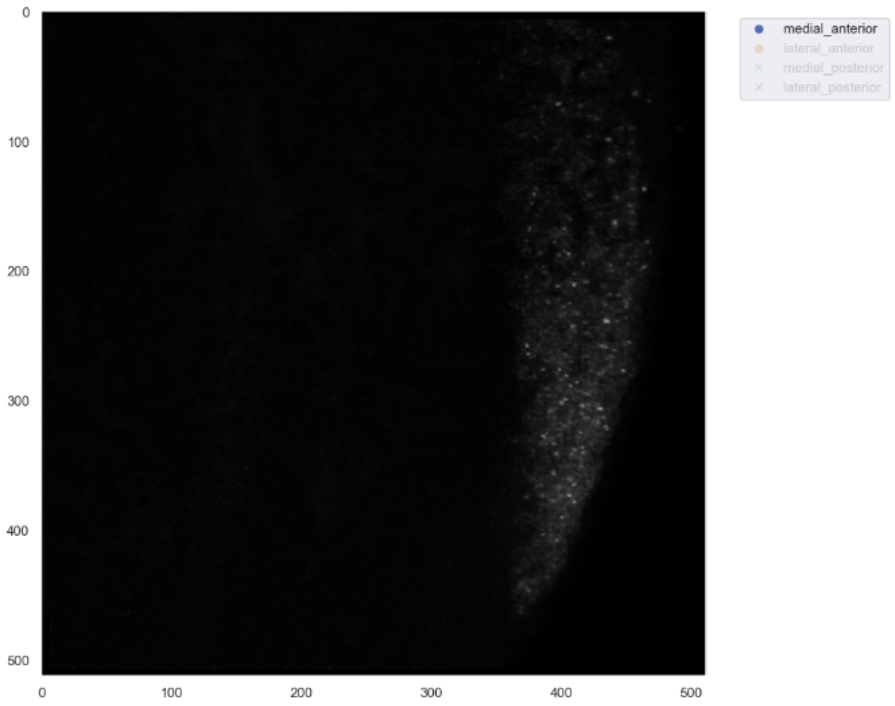

Figure 8: We did not include images like this one, where both the medial and lateral boundaries of the *snai2*-positive region were not visible.

Notes

[1] The Streichan group recently pre-printed details of a new surface analysis package that uses Blender, rather than MATLAB. This may be more user friendly for people with less coding/MATLAB experience. `https://www.biorxiv.org/content/10.1101/2025.02.04.636523v2`

[2] From the 31-slice mapped z-stack, the program treats slices 2-14 as the deep ectodermal cell layer. We use a mean intensity projection of these slices for analysis, although that can be changed for different purposes.

[3] We chose to use a relatively "thick" line (width parameter set to 51) between the chosen points to mitigate slight differences in axial location.

References

Heemskerk, I. and Streichan, S. J. (2015). Tissue cartography: compressing bio-image data by dimensional reduction. *Nat Methods*, 12(12):1139–42.