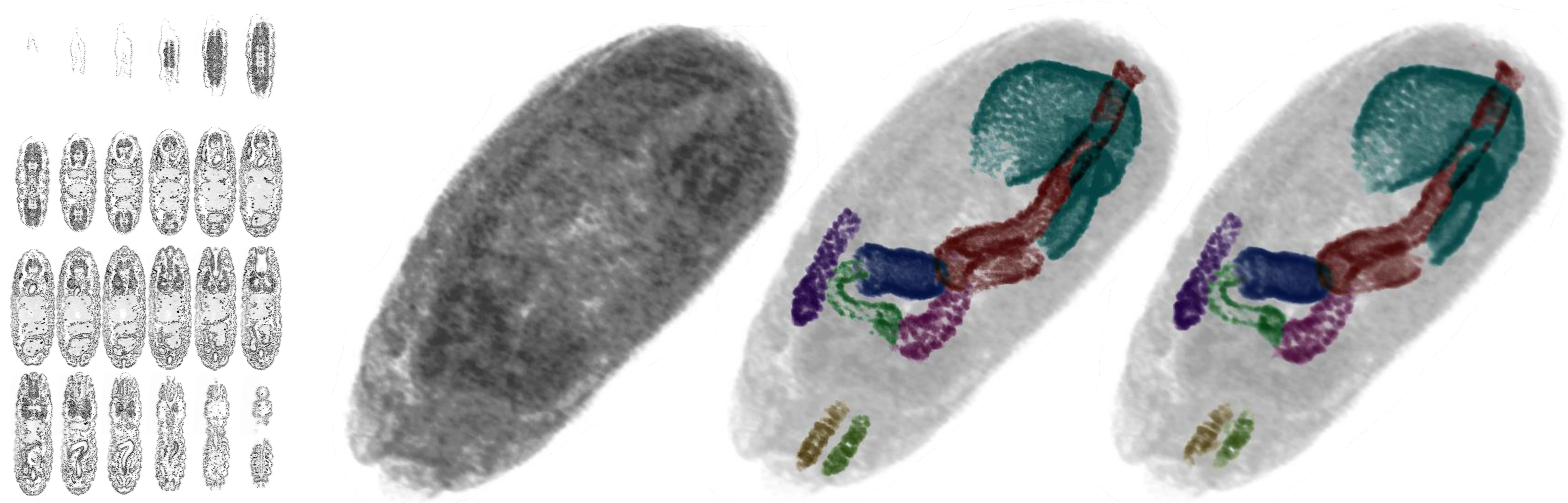
Volumetric Semantic Segmentation using Pyramid Context Features

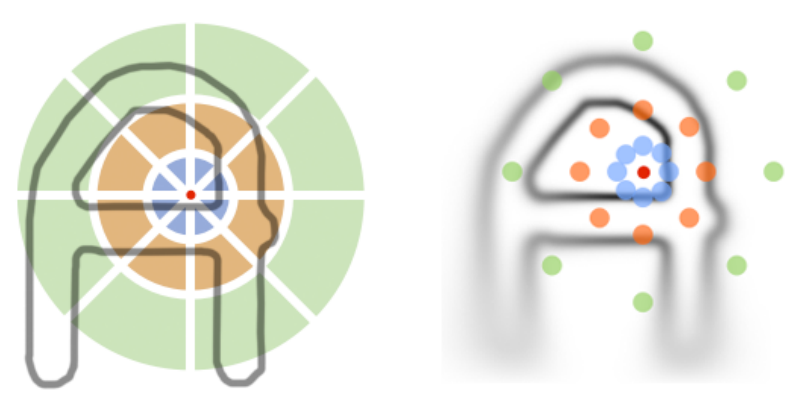
Jonathan T. Barron[1], Pablo Arbeláez[1], Soile V. E. Keränen[2], Mark D. Biggin[2], David W. Knowles[2], Jitendra Malik[1],
[1]UC Berkeley, [2]Lawrence Berkeley National Laboratory

We have large (15-megavoxel) volumetric images of fruit fly embryos acquired with 3D fluorescence microscopy. Biologists have hand-labeled all voxels belonging to different tissue types. **Our goal**: accurately and quickly label all voxels in a new volume.



Scanned Images (subsampled) | Composited Volume | Biologist Annotation | Our Prediction

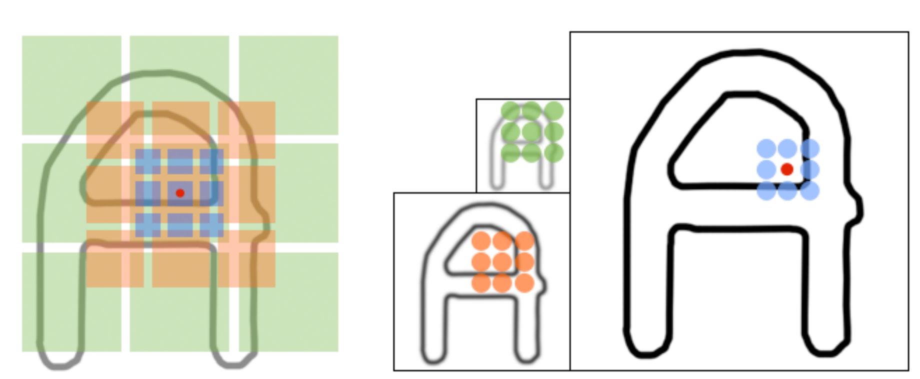
These tissues have very fine detail and vary greatly across subjects. This causes standard segmentation-based or atlas/exemplar-based registration technique to fail. We will perform semantic segmentation by evaluating a classifier on a feature for every voxel in the image. **The size of our volumes makes standard sliding-window classification intractable.**



Shape Context, C-HOG, etc | Geometric Blur, DAISY, etc

We present the **Pyramid Context**: a HOG / Shape context-like feature designed such that exact per-voxel classification is efficient.

Key Idea: if we use average pooling and an axis-aligned, scale-similar, Haar-like histogram arrangement, then our feature has **two equivalent interpretations:**



Pyramid Context (Flat View) | Pyramid Context (Pyramid View)

Key Insight: interpolation, pyramid construction, average-pooling, and linear classifiers are all linear, and therefore **associative**. A linear classifier can therefore be efficiently evaluated on all pyramid context features in a volume as follows:

1) Construct a Gaussian pyramid
2) Filter each level of the pyramid with a 3x3 filter
3) Collapse the pyramid

Per-voxel classification takes **seconds** with this technique, where traditional sliding-window classification takes **hours**.

Architecture

Basic pyramid context features work well for semantic segmentation (see our "shallow" model) but they can also be used as a tool for constructing a "deeper" feature channels. We use three feature types:

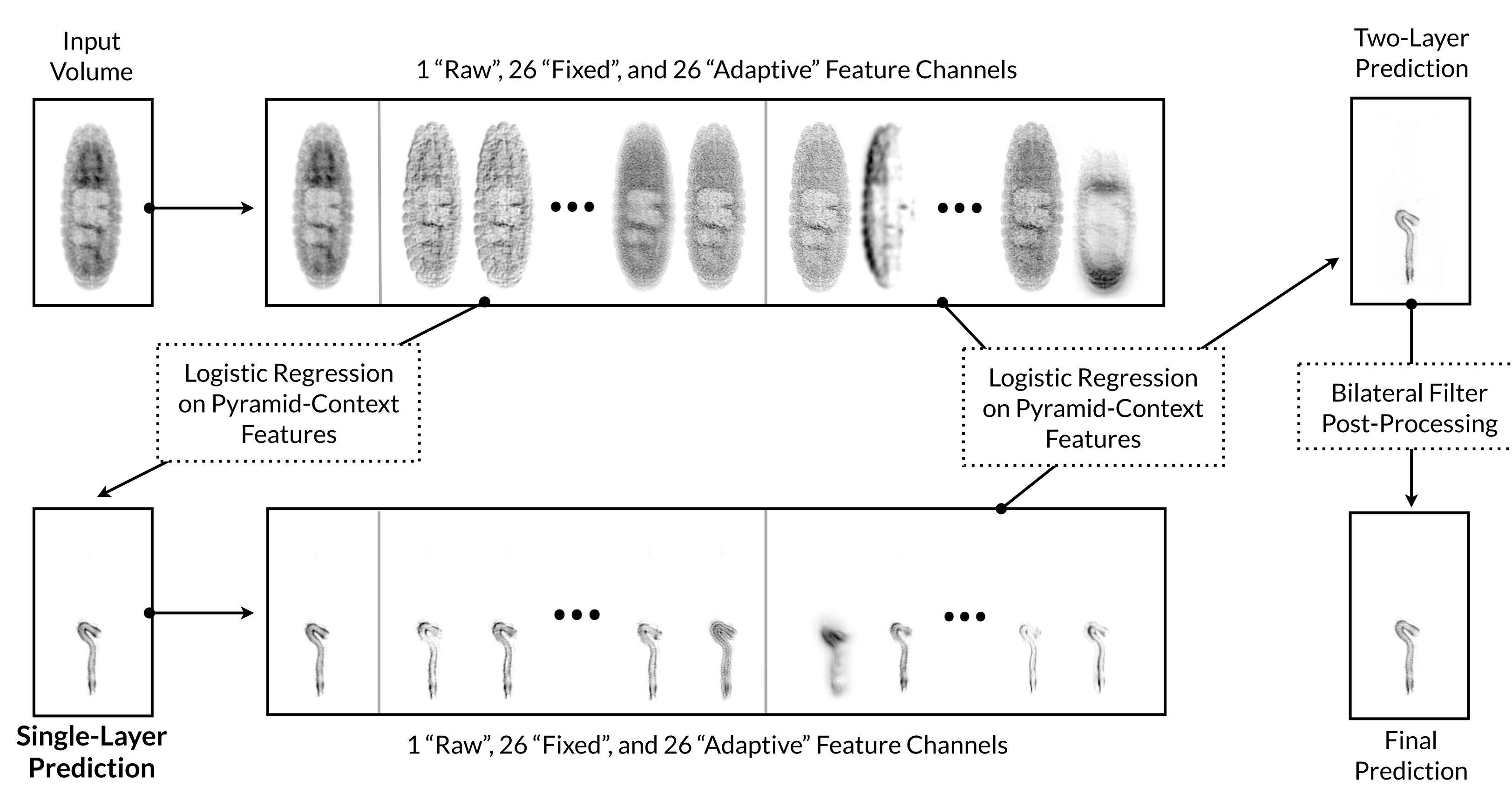
1) The input volume itself
2) Pyramid context features on oriented-edge information
3) Learned pyramid context features on the input volume

Feature learning is done using whitened k-means. Both the oriented-edge channels and the learned channels are half-wave rectified.

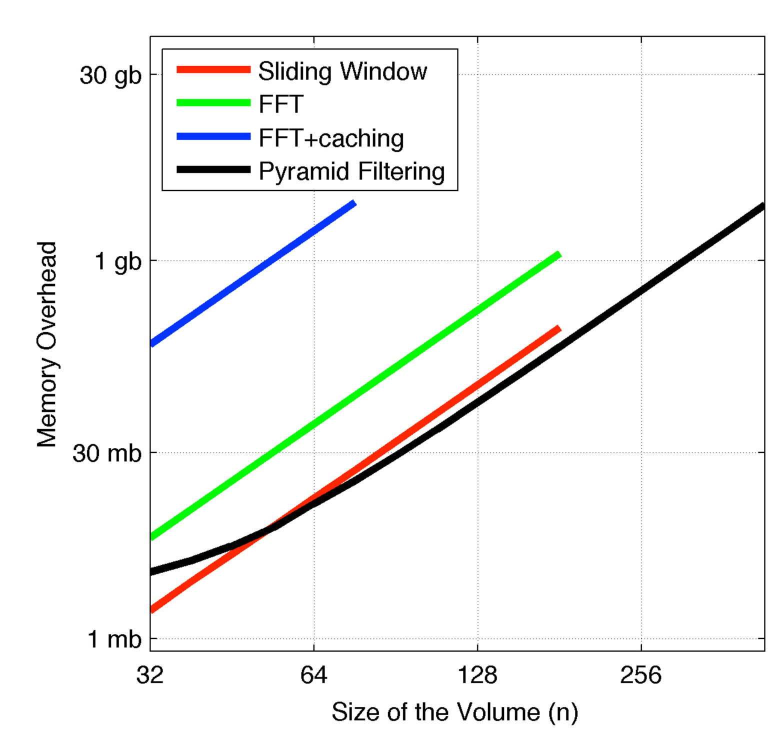
Because our imagery has been aligned in a canonical frame of reference, we also use the absolute position of each voxel as a feature.

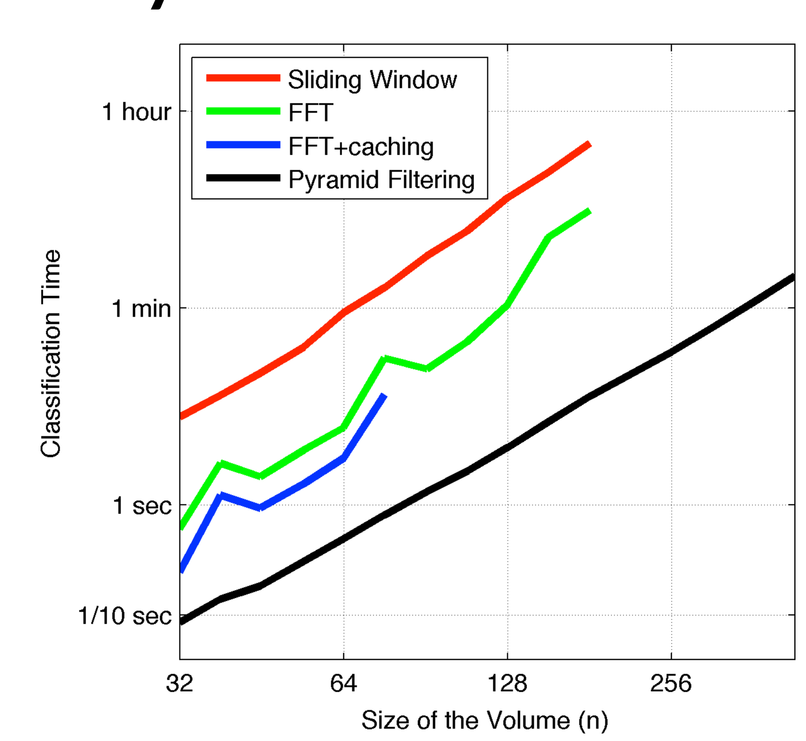
Pyramid context features are also used to model context by training a second classifier on pyramid context features placed on the output of a first classifier.

Our final per-voxel prediction is post-processed using bilateral filtering (which can be viewed as inference in a CRF).



Efficiency



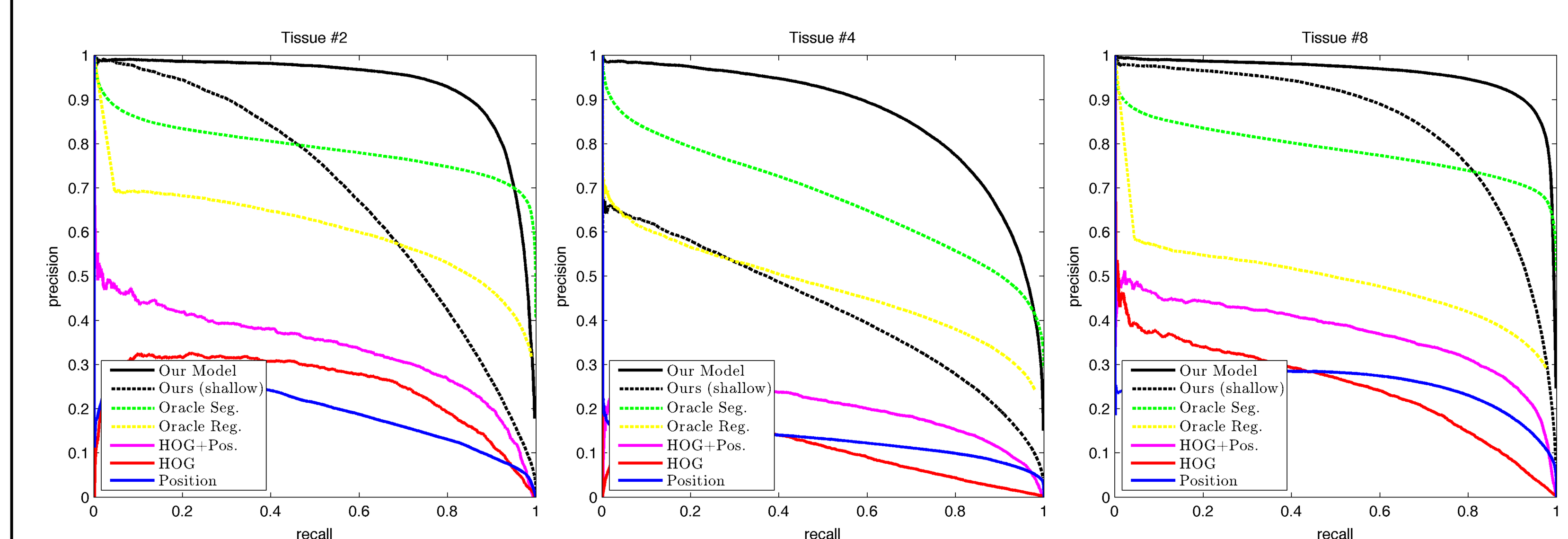


Our technique is 200× faster than sliding window with similar memory requirements, and 5×-20× faster than FFT-filtering while requiring 1/6th-1/160th the memory

Evaluation

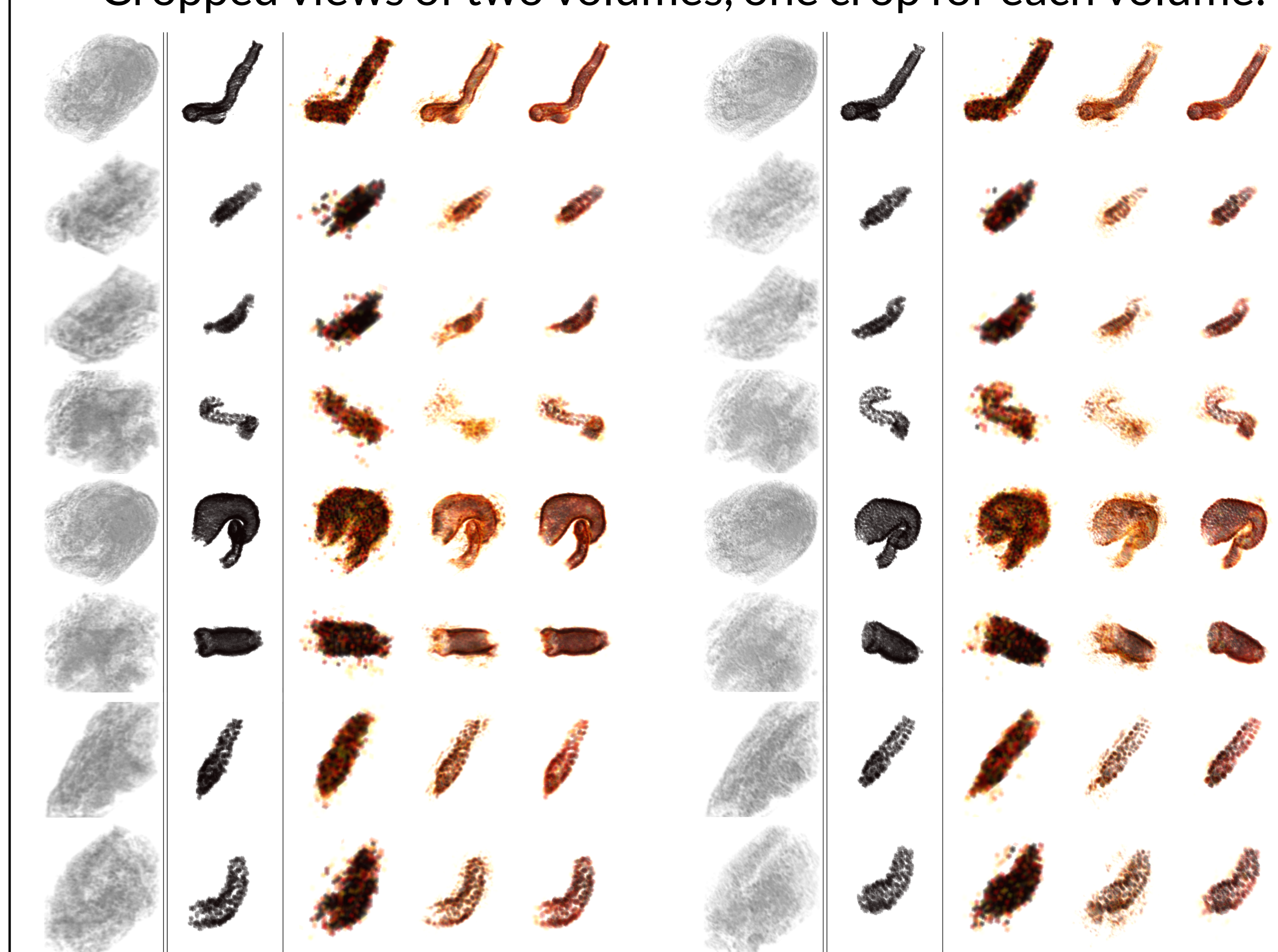
Test-set APs (28 15-megavoxel scans, 14 train, 14 test):

Model	Tissue 1	Tissue 2	Tissue 3	Tissue 4	Tissue 5	Tissue 6	Tissue 7	Tissue 8	Avg.
Oracle Segmentation Algorithm.	0.745	0.791	0.781	0.677	0.762	0.818	0.783	0.787	0.768
Oracle Registration Algorithm	0.485	0.597	0.592	0.468	0.464	0.746	0.443	0.476	0.534
HOG	0.249	0.252	0.256	0.101	0.173	0.470	0.157	0.250	0.239
Position	0.227	0.200	0.247	0.127	0.237	0.261	0.212	0.249	0.220
HOG + Position	0.417	0.339	0.345	0.204	0.431	0.545	0.337	0.371	0.374
Our Model (shallow)	0.705	0.688	0.691	0.425	0.679	0.848	0.818	0.846	0.712
Our Model	0.914	0.934	0.933	0.865	0.945	0.975	0.947	0.958	0.934



Evaluation is difficult as few techniques exist, so we compare against oracle segmentation and registration techniques, which we outperform significantly.

Cropped views of two volumes, one crop for each volume:



Slices of two volumes, for three different tissue types