Improving Neuropathological Analysis in IHC through Starburst Artifact Detection

Bahena, Alejandra; Capp, Noah; Irwin, David, MD
University of Pennsylvania Digital Neuropathology Lab, Frontotemporal Degeneration Center

Penn Digital Neuropathology Lab

Background

- Semi-Automated Neuropathological Analysis (SANA) is a toolkit currently in development, which allows for the efficient analysis of post-mortem human brain tissue. It provides predictive models for semi-automatic segmentation of objects such as neurons, plaques, and tangles, as well as various modules which measure the degree of pathology or the amount of degeneration in the cytoarchitecture.
- However, artifacts in the tissue afflict the accuracy of these predictive models and cause inaccuracies in the measurements of pathology and/or degeneration.

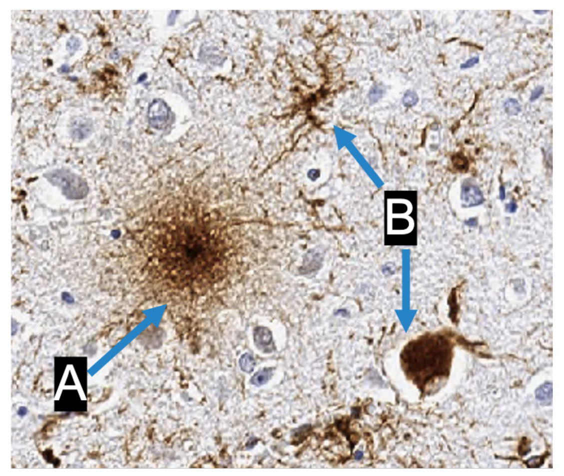

Figure 1. Representative Starburst Artifact in AT8 Stained Tissue
(A) Shows a starburst artifact while (B) show true pathology

- Detecting starburst artifacts in neuropathological tissue is an important step in the development of the SANA toolkit.
- **The aim of this project is to develop methods which detect and remove starburst artifacts from immunohistochemically stained post-mortem human brain tissue images.**

Dataset

- 9 AT8 stained slides from FTLD-Tau patients were selected for analysis. These slides cover various regions from the brain: ANG, HIP, CING, MFC, WERN and OFC.
- From these slides, 89 regions of interest (ROIs) were randomly selected for the dataset.
- The ROIs were manually created as annotations in QuPath 2.0 bioimage analysis software and exported as JSON files.

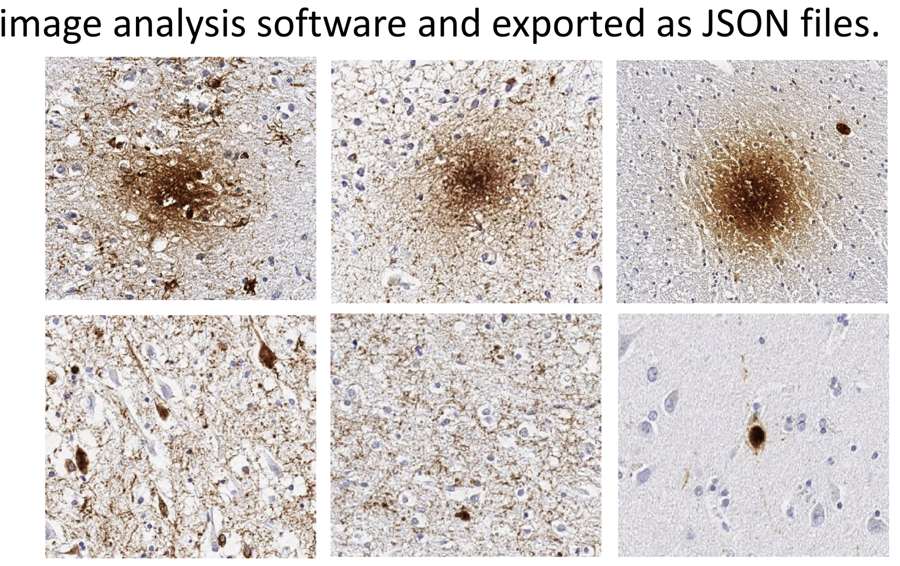

Figure 2. Representative ROIs
ROIs containing starburst artifacts (top row) and ROIs with no starburst artifacts (bottom row)

Image Processing

The Artifact Detection module is developed in Python, along with the rest of the SANA architecture.

Pre-Processing

1. Load ROI Frame into memory.
2. Separate the DAB stain from the Frame through color deconvolution.
3. Apply Anisotropic Diffusion filtering[1] to smooth inconsistencies in the DAB staining.
4. Perform Stain Thresholding to remove insignificant background staining.
5. Apply Morphological Opening filtering[2] to further remove background staining and simplify objects in the stained image.

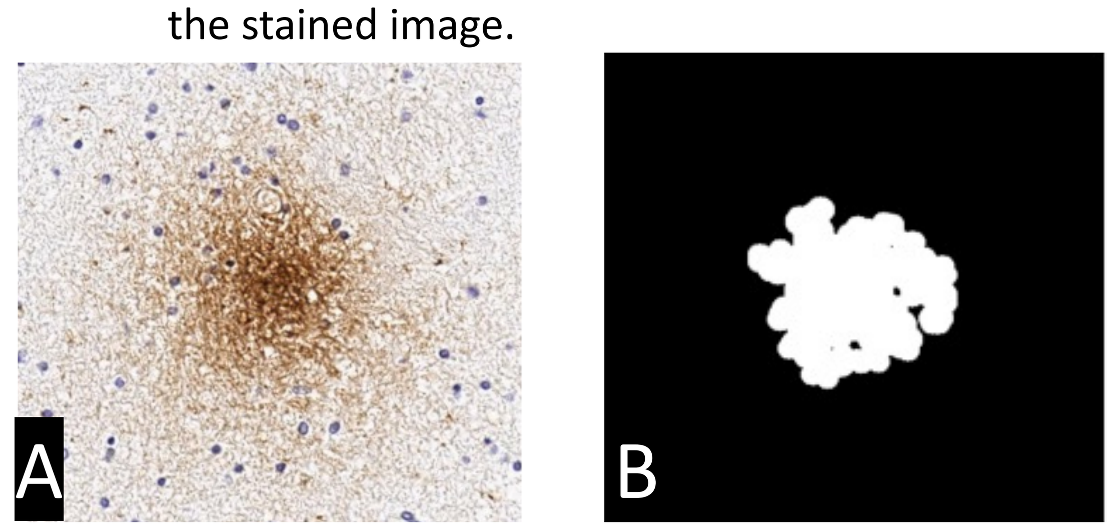

Figure 3. Filtered Image
(A) Shows a starburst artifact while (B) shows the filtered ROI after pre-processing steps.

Object Detection

1. Use DBSCAN algorithm[3] to extract a mask of core cluster points within each object in the stain.
2. Convert each cluster of points into a Polygon annotation using ConvexHull function[4].
3. Export object detections to a JSON file.

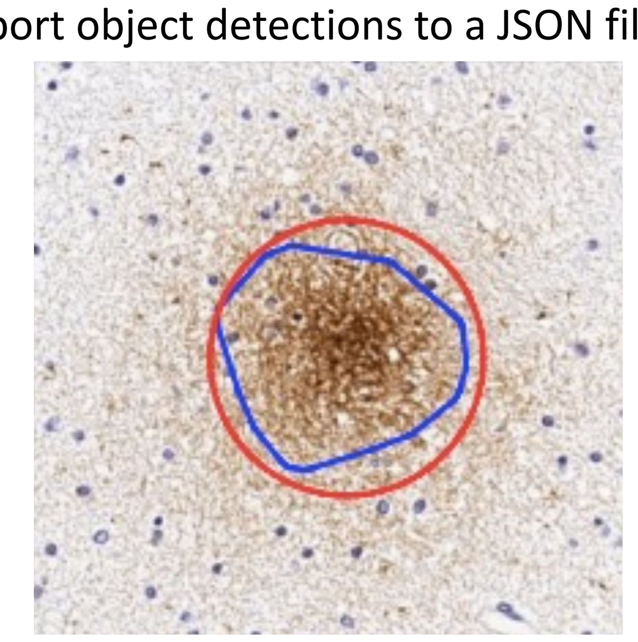

Figure 4. Artifact Detection
Detection outputted by the program after the processing steps (blue) and reference artifact annotation that was manually created (red)

Template Matching

- The Image Processing step provides a segmentation for each object in the stained image.
- A Template Matching algorithm is performed on each segmentation, which yields a confidence value -- The probability that said segmentation is a starburst artifact.
- The Template used is a 2D Gaussian Probability Density Function (PDF) Kernel.

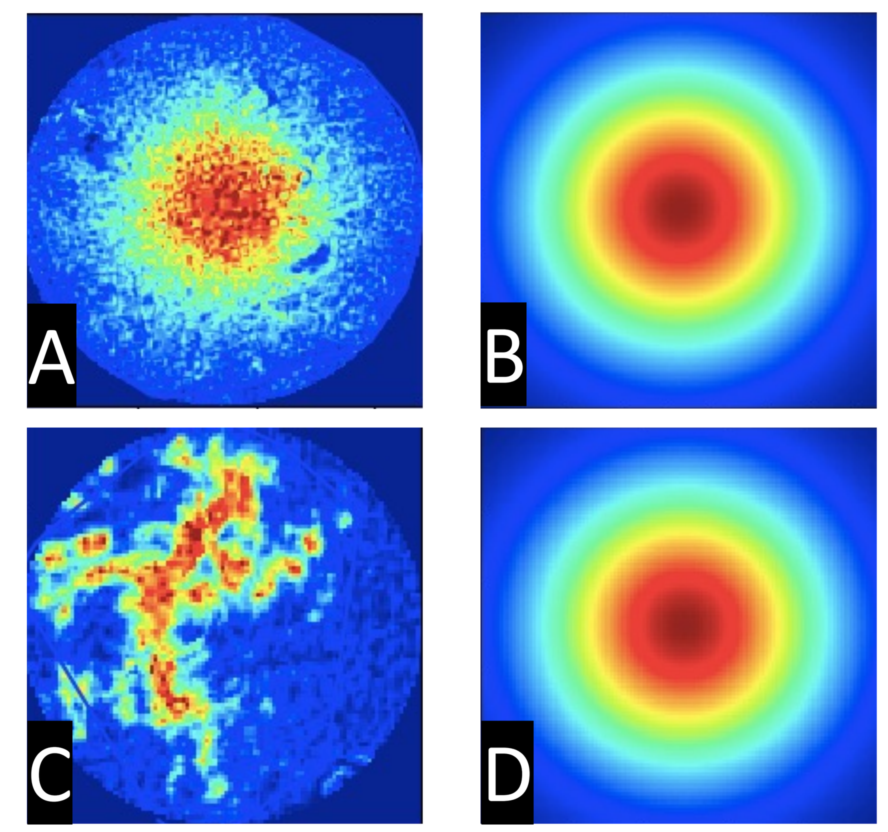

Figure 5. Template Matching
(A) Shows a starburst artifact that matches the artifact template shown (B) with a **confidence value of 0.946.**
(C) Shows a ramified astrocyte that does not match the artifact template shown (D) due to having a **confidence value of 0.554.**

Post-Processing

- Detections are rejected based on minimum surface area criteria.
- After scoring analysis, detections can also be rejected using a minimum confidence threshold.

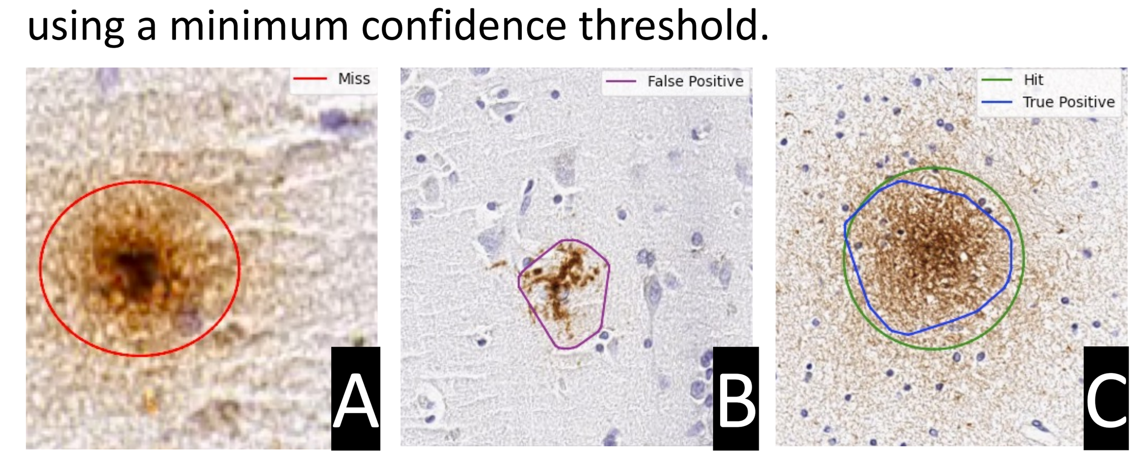

Figure 6. Scoring Analysis
(A) Shows a false negative detection (red) while (B) shows a false positive detection (purple) and (C) shows a true positive detection (blue)

Results

- Positive detections are considered True or False based on the IOU (intersection over union) score between itself and any given manual annotation.

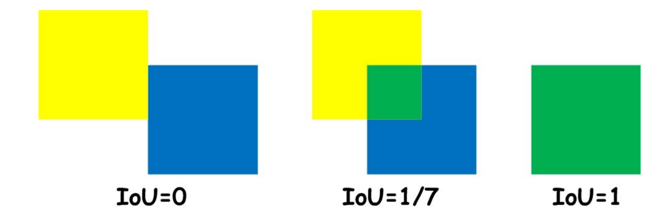

Figure 7. Low and High IOU Values[5]

- Scoring of this object detection software is done via Precision-Recall (PR) Curves, as well as Average Precision (AP) values.

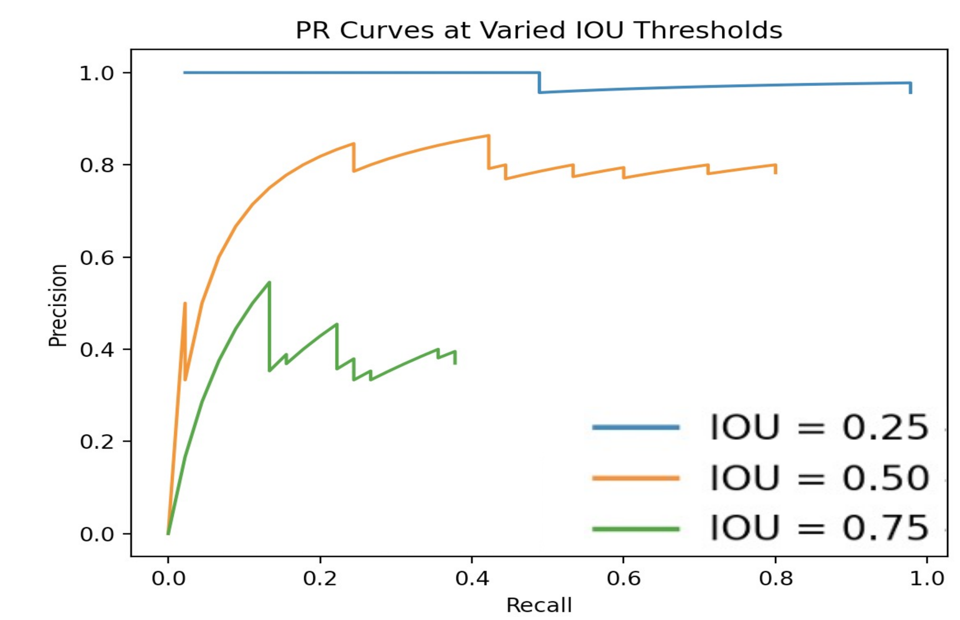

Figure 8. PR Curves at Varied IOU Thresholds
High Precision and Recall values were given at low IOU threshold values

IOU	AP
0.25	94.09%
0.50	62.06%
0.75	14.71%

Table 1. AP Values at Varied IOU Thresholds
High Average Precision was given at low IOU threshold values

Conclusion

- The developed artifact detection software provides high Precision and Recall at low IOU.
- SANA will require this performance at higher IOU's before being implemented in the full processing pipeline.
- Detection of starburst artifacts could improve pathological and degeneration measures.

Future Directions

- Run the system on SANA's large datasets to test performance in more difficult scenarios.
- Optimize image processing parameters and techniques.
- Expand the system to include tissue tears, blood vessels, and other important artifacts.

Contact

Alejandra Bahena
COL 2022
Email: abahena@sas.upenn.edu

References

1. *Andrija Štajduhar, Domagoj Džaja, Miloš Judaš, Sven Lončarič. Automatic detection of neurons in NeuN-stained histological images of human brain. Physica A: Statistical Mechanics and its Applications. Volume 519, 2019. Pages 237-246, ISSN 0378-4371. https://doi.org/10.1016/j.physa.2018.12.027.*
2. *Image filtering*. OpenCV. (2021, April 2). https://docs.opencv.org/4.5.2/d4/d86/group__imgproc__filter.html#ga67493776e3ad1a3df63883829375201f.
3. Scikit-learn developers. (n.d.). *sklearn.cluster.dbscan¶*. Scikit. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html.
4. SciPy v1.7.1 Manual. (n.d.). *scipy.spatial.convexhull¶*. scipy.spatial.ConvexHull . https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.ConvexHull.html.
5. https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/

SCAN ME