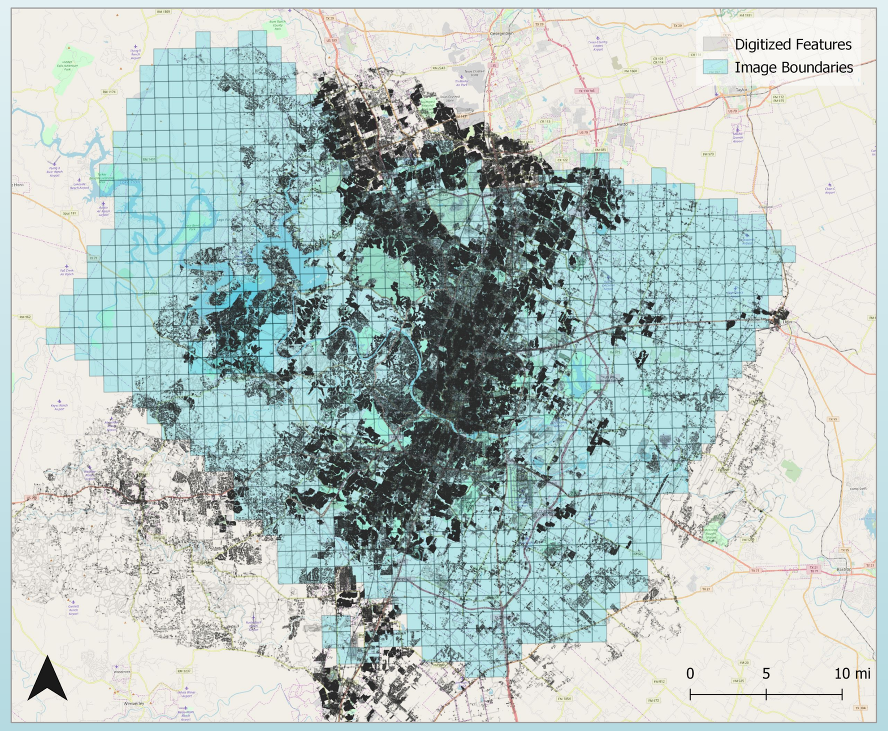
Identifying Structures in Aerial Overhead Imagery with Deep Learning

Brent Redmon, Constance Xu, Jelena Tešić
Computer Science Department, Texas State University

Manual region annotation is time and cost prohibitive. We propose to use a state of the art deep learning-based pipeline to automate region recognition for geospatial analysis. Models are trained and validated on City of Austin impervious coverage dataset.

Motivation

State of the art for object and region labeling in GIS systems is manual labeling and polygon marking, and it does not utilize advances in automated image segmentation.



[Figure A: Geographic jurisdiction of Austin]

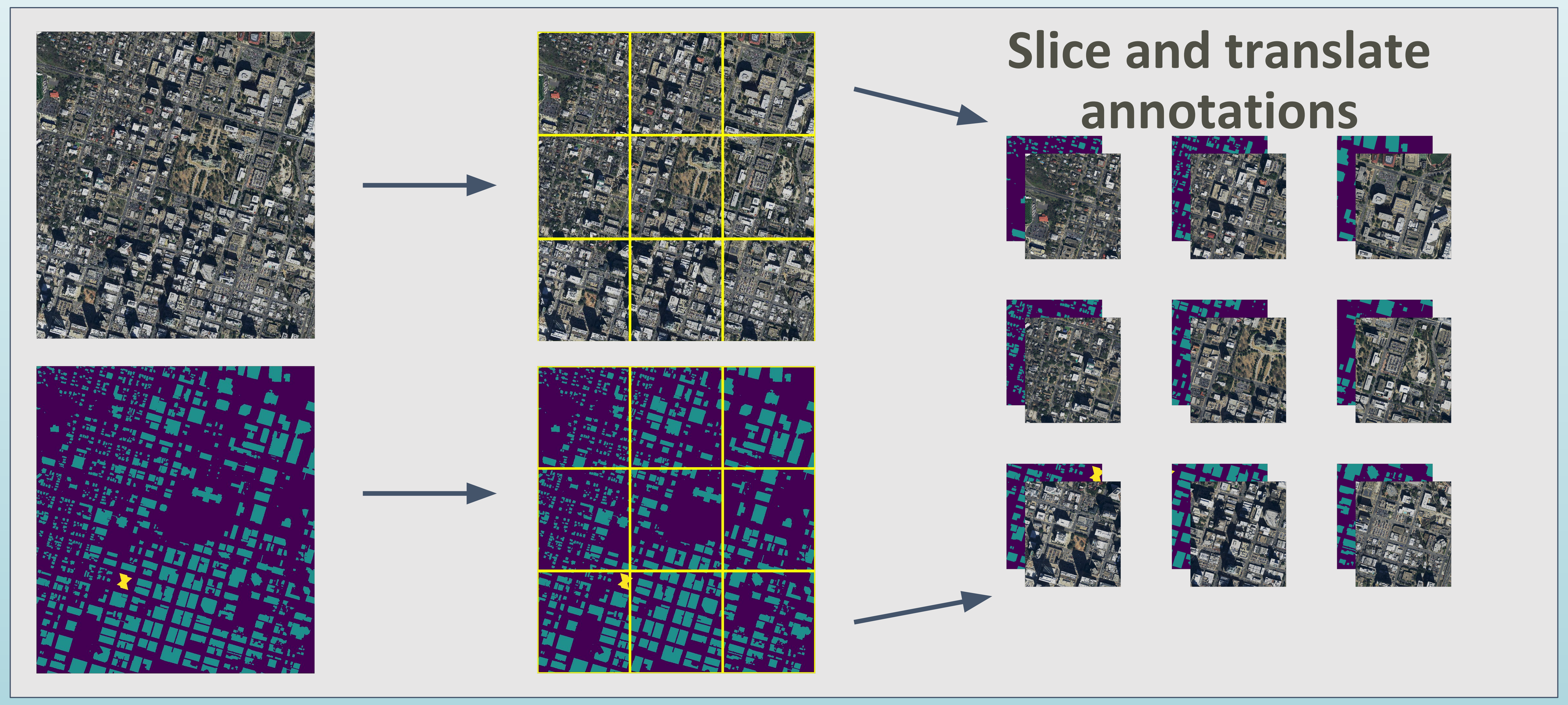
Goals:
1. Evaluate the cost effectiveness of a state of the art deep learning algorithm used for overhead imagery segmentation and object classification
2. Determine viability of porting deep learning platform to mobile architecture

City of Austin dataset:
- *Physical size: **2.5 TB** of data*
- *Originally **66 classes**, reduced to **1***
- *Costs CoA **$500,000** per set*
- *Use existing annotations to train model*

Approach

DATASET - Images from Downtown 6" accuracy
- **Original Size ($\sim$12,000 pix^2)**
  - **Train: 29**
  - **Validation: 12**
  - **Test: 4**
- **Spliced Dataset from original ($\sim$1,200 pix^2)**
  - **Train: 2065**
  - **Validation: 856**
  - **Test: 284**
  - **104,012 annotations total**

[Figure B: Pairing down size of dataset]

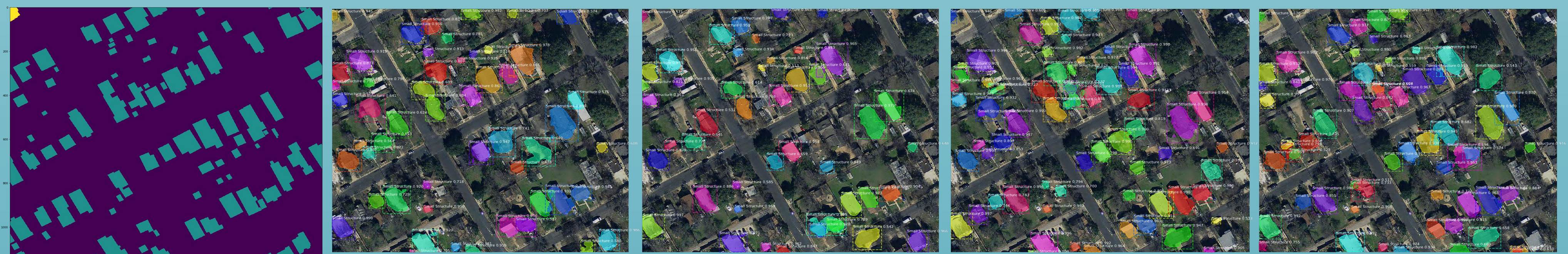


Server
- Linux version NVIDIA DGX Server v4.0.4
- CPU Intel® Xeon® E5-2698CPU @2.20GHz x2
- NVIDIA Tesla P100 GPU x8
- 528 GiB System memory

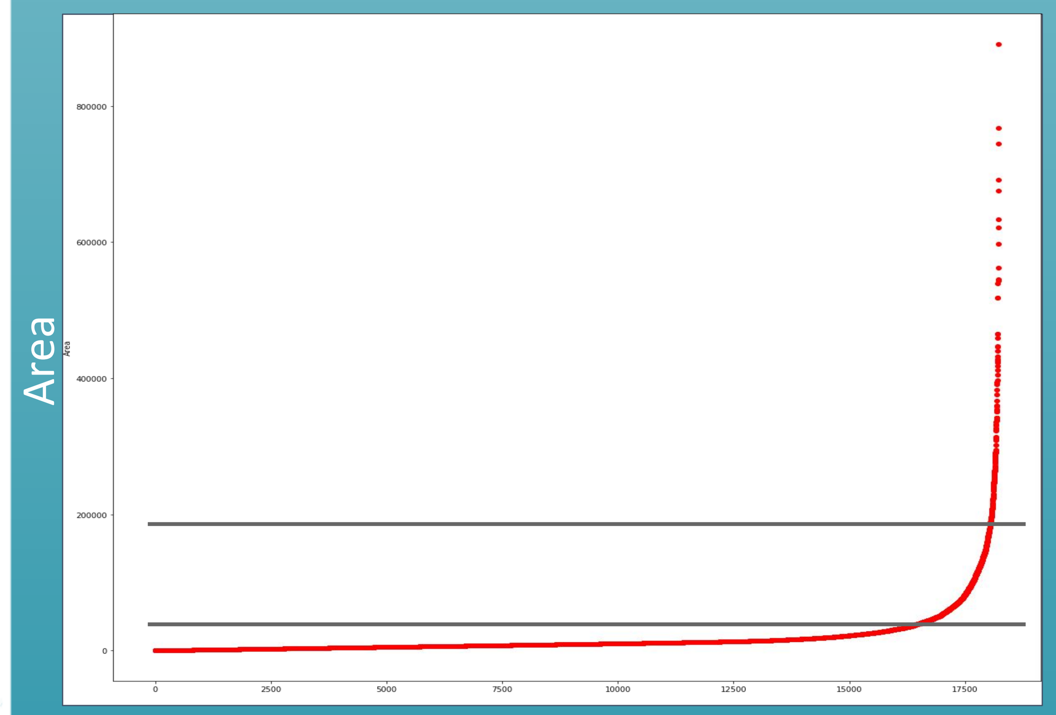
Using state of the art deep learning pipeline from consumer imagery to overhead imagery:
1. Splice images into manageable sizes and transpose existing annotations to spliced images
2. Reduce classes from 66 classes to just a "Structure" class
3. Train custom Mask R-CNN model (modified mask layers)
4. Evaluate mask generation robustness using mean average precision (mAP)
5. Apply a lightweight version of the model onto a single board computer

Findings

[Figure C: Initial model results]



Ground truth masks | Model 1 | Model 2 | Model 3: better segmentations; smaller objects | Model 4



[Figure D: Dataset split by area into 3 categories (Small, Medium, and Other)]

Model	Kernel Size	Epochs	Validation Steps	Steps Per Epoch
1	256	80	50	90
2	128	80	50	90
3	256	80	50	180
4	128	80	50	180

[Table 1: Model configurations]

Deep Learning Mobile Devices

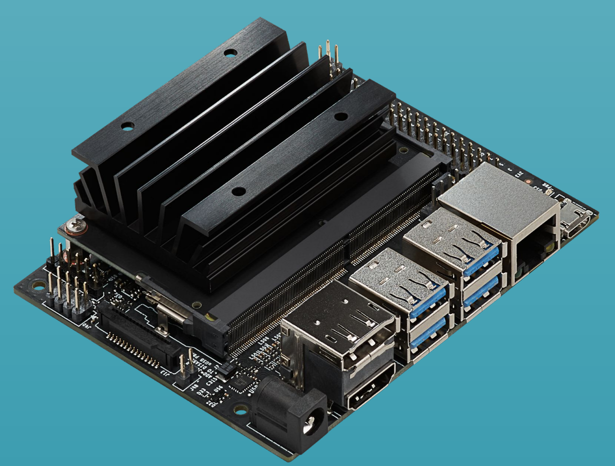
[Table 2: Nano and TX2 Specs]

Jetson Nano	Jetson TX2
ARM Cortex-A57 (quad-core) @ 1.43GHz	ARM Cortex-A57 (quad-core) @ 2GHz + NVIDIA Denver2 (dual-core) @ 2GHz
128-core NVIDIA Maxwell @ 921MHz	256-core NVIDIA Pascal @ 1300MHz
4GB 64-bit LPDDR4 @ 1600MHz | 25.6 GB/s	8GB 128-bit LPDDR4 @ 1866Mhz | 58.3 GB/s

Tensorflow:
- **Requires more memory to train**
- **Smaller sized models when deployed**

PyTorch:
- **Requires less memory to train**
- **Larger sized models when deployed**

[Figure E: Nvidia Jetson TX2]



[Figure F: Nvidia Jetson Nano]

Conclusion

A deep learning model that accurately predicts structures and classifies them by size is feasible, but is not easily done. Packaging this model on a single board computer is possible, but will require more modification to the neural network architecture

Next Steps:
- Successfully run Mobile Mask R-CNN on an Nvidia Jetson TX2
- Continually refining on weights so that the model can be more accurate on large-scale, aerial imagery
- Finding ways to minimize loss of accuracy on Mobile Mask R-CNN while keeping it lightweight

This material is based upon work supported by the National Science Foundation under REU grant #1757893 and by NAVAIR N68335-18-C-0199 and NSF-CRI 1305302 awards. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation and NAVAIR.

Contact Info: btr26@txstate.edu
cxu16@stevens.edu