Why Is It So Hard to Scale Development Data Projects? (*Answer: Build a Data Lake First*)

Jonathan Hersh (Chapman)

Private Sector Development Research Network Workshop on Firm-level Data

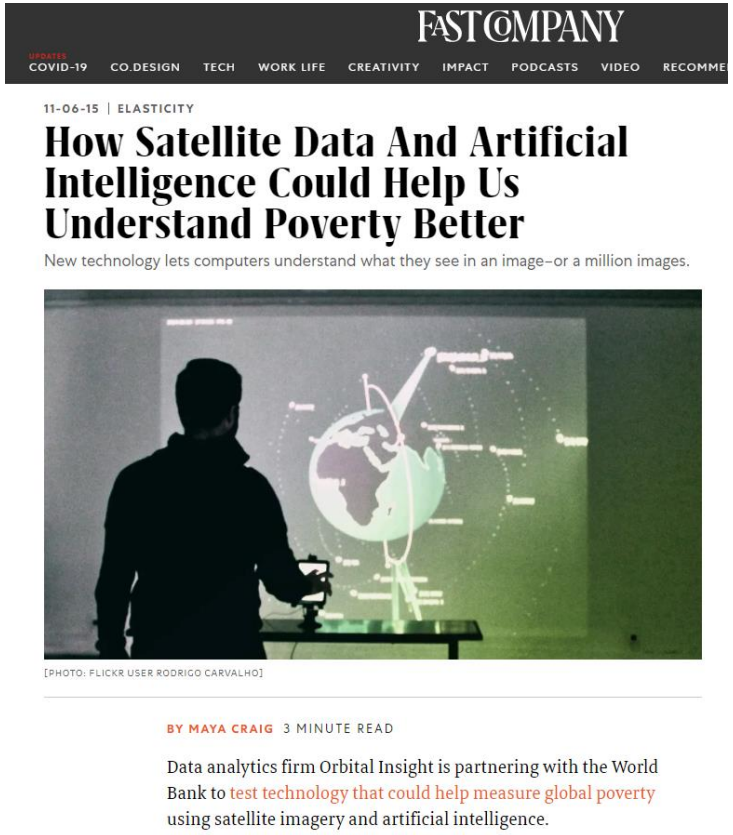
4/6/2021

About: Jonathan Hersh, PhD

- Assistant Professor Economics and Management Science Chapman University
- PhD in economics, Boston University
- **Research Fields**:
  - Applications of artificial intelligence (computer vision)
  - Economics of information systems
  - Development economics
  - Digitization strategy
- **Teaching Fields**:
  - Machine learning
  - Applications of artificial intelligence

Formerly: Data Scientist @ World Bank

- Satellite Imagery + Computer Vision + Machine Learning
- Advised World Bank/IDB on COVID poverty transfers in Belize, Togo, Guinea



FAST COMPANY

UPDATES COVID-19 CO.DESIGN TECH WORK LIFE CREATIVITY IMPACT PODCASTS VIDEO RECOMME

11-06-15 | ELASTICITY

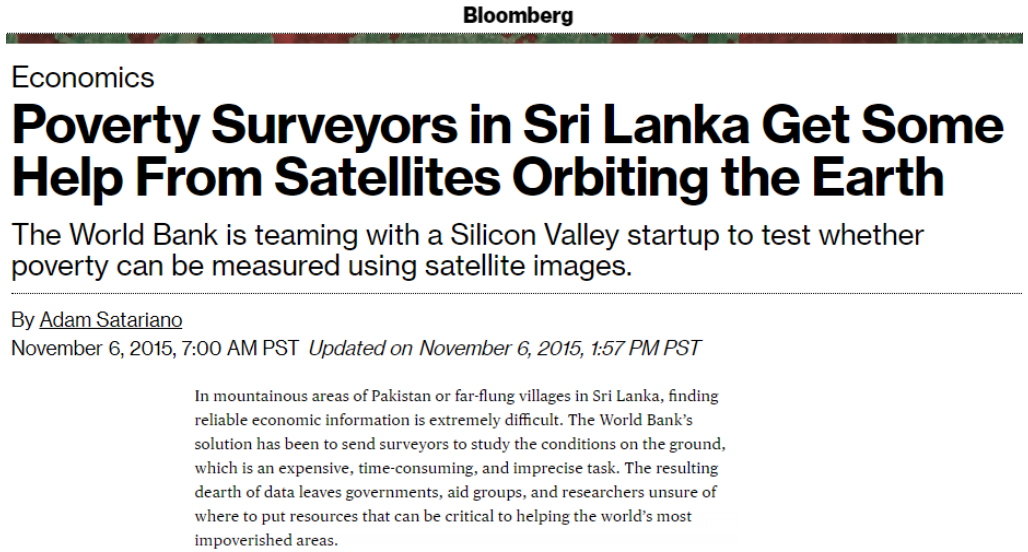
How Satellite Data And Artificial Intelligence Could Help Us Understand Poverty Better

New technology lets computers understand what they see in an image–or a million images.

[PHOTO: FLICKR USER RODRIGO CARVALHO]

BY MAYA CRAIG 3 MINUTE READ

Data analytics firm Orbital Insight is partnering with the World Bank to test technology that could help measure global poverty using satellite imagery and artificial intelligence.



Bloomberg

Economics

Poverty Surveyors in Sri Lanka Get Some Help From Satellites Orbiting the Earth

The World Bank is teaming with a Silicon Valley startup to test whether poverty can be measured using satellite images.

By Adam Satariano

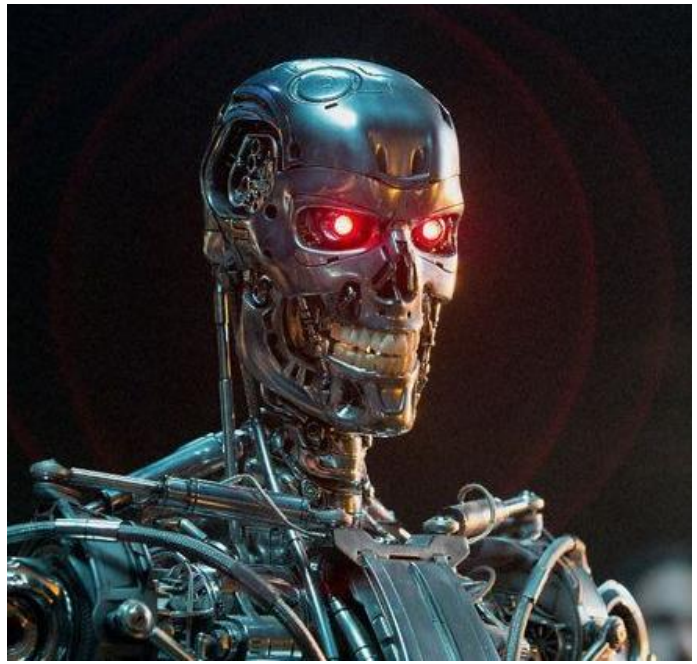
November 6, 2015, 7:00 AM PST *Updated on November 6, 2015, 1:57 PM PST*

In mountainous areas of Pakistan or far-flung villages in Sri Lanka, finding reliable economic information is extremely difficult. The World Bank's solution has been to send surveyors to study the conditions on the ground, which is an expensive, time-consuming, and imprecise task. The resulting dearth of data leaves governments, aid groups, and researchers unsure of where to put resources that can be critical to helping the world's most impoverished areas.

Why Do Development Data Projects Fail?

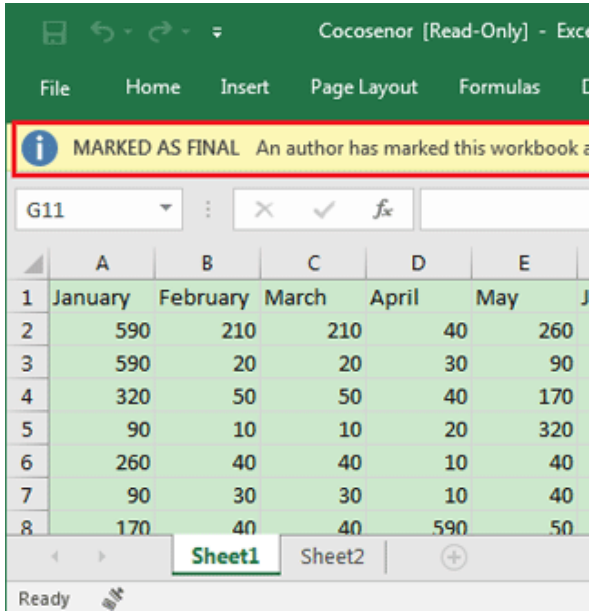
1. Models can only be as good as your data pipeline (Bad data = bad model)
2. Build a consistent data lake first (excel files are bad)
3. All data and models should be an API (i.e. Application Programming Interface or structured way to access data)

Perceptions of capabilities of Artificial Intelligence

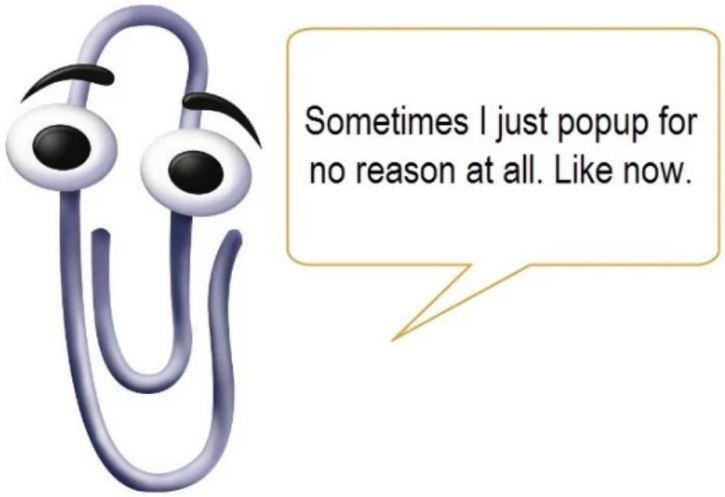


+

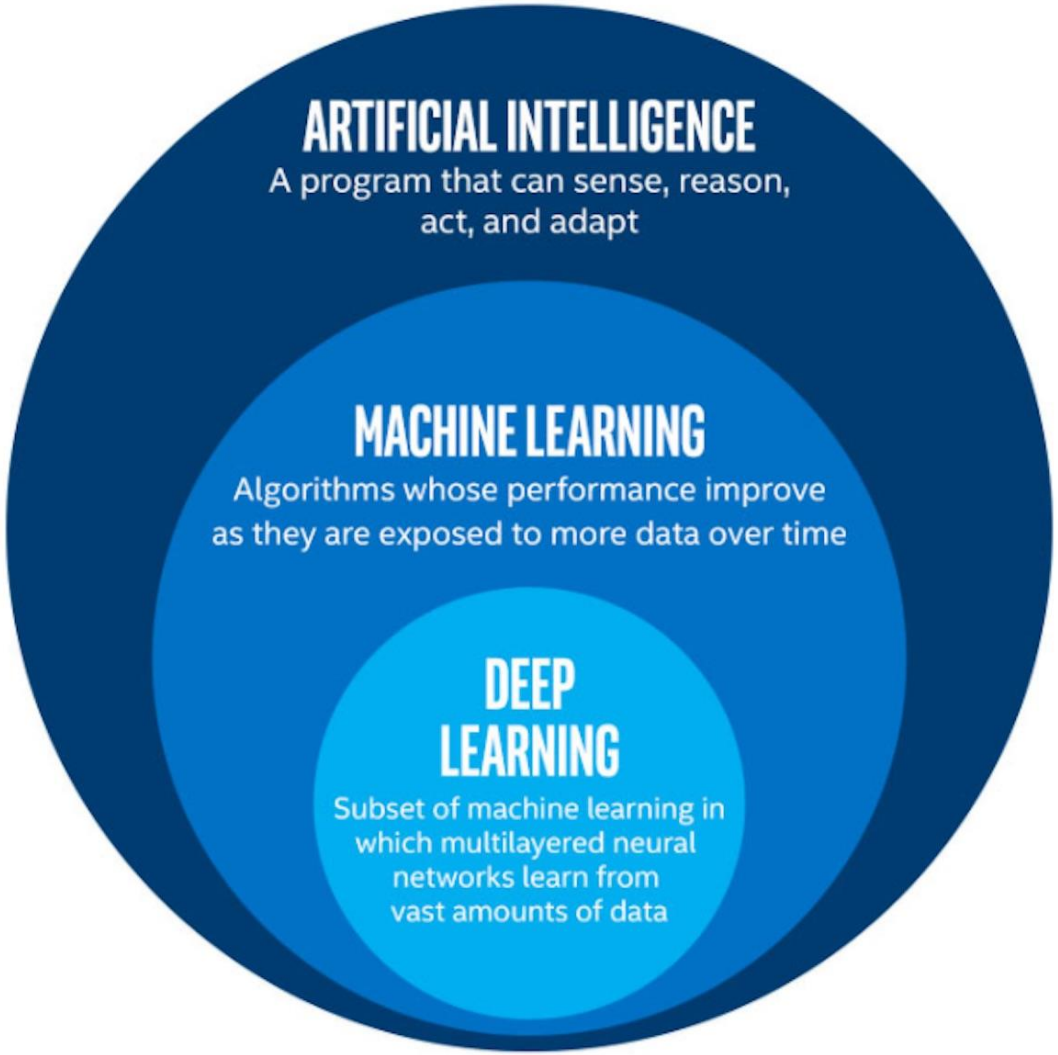
Existing data infrastructure



=

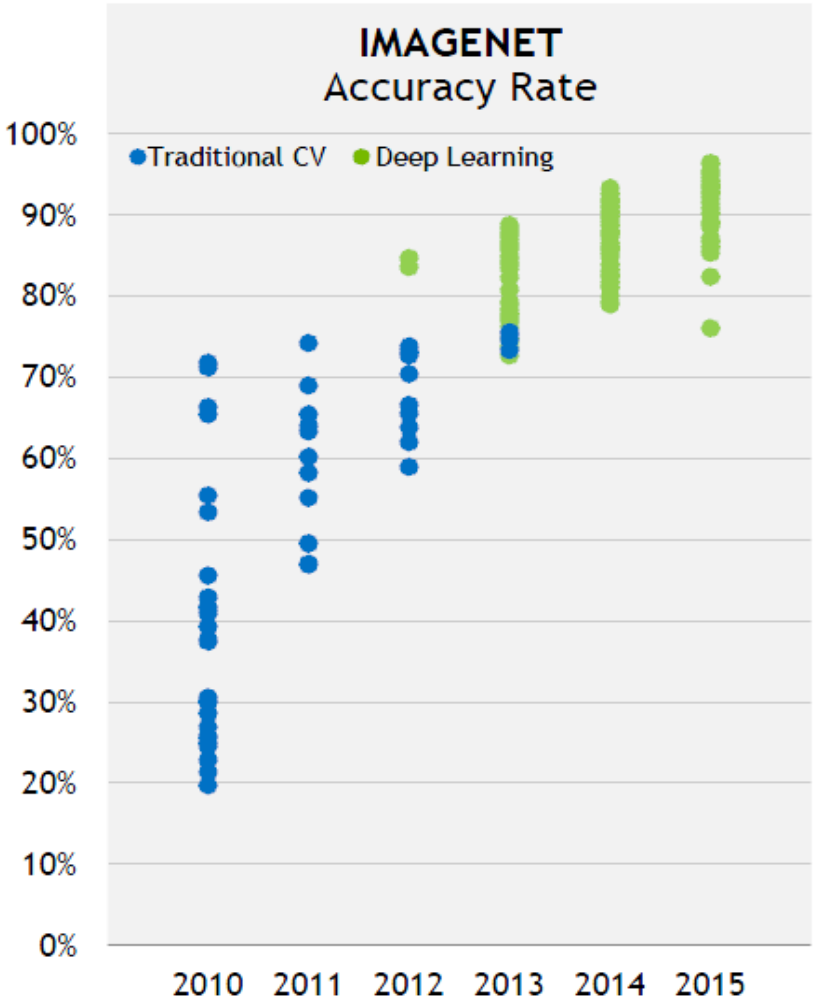


Artificial Intelligence vs Machine Learning vs Deep Learning



"**The deep in deep learning isn't a reference to any kind of deeper understanding achieved by the approach**; rather, it stands for this idea of successive layers of representations" – Francois Chollet

One Reason for The Deep Learning Hype: Images



- Before deep learning, computers couldn't recognize objects from images.
- In 2012 researchers first used a deep learning model on the ImageNet competition, which tests an AI to recognize objects in images
- But ImageNet had 1M+ images!
- Translation: models and data are close complements

Modern Data Pipelines

Raw Data

amazon S3

Raw Data Transformation

python

R

Analysis Dataset

amazon S3

Model Output Dataset

amazon S3

API = structured way to access your model

API

Data Viz Dashboard

tableau

Shiny

plotly | Dash

Report Few Will Read

Data Lake: a centralized repository that stores all your structured and unstructured data on demand and accessible by everyone

python

R

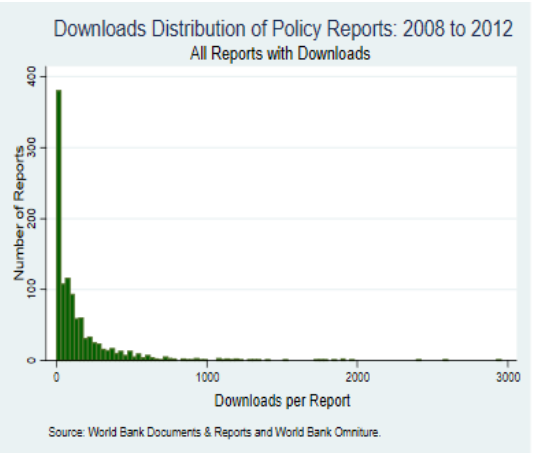
Variable Creation

python

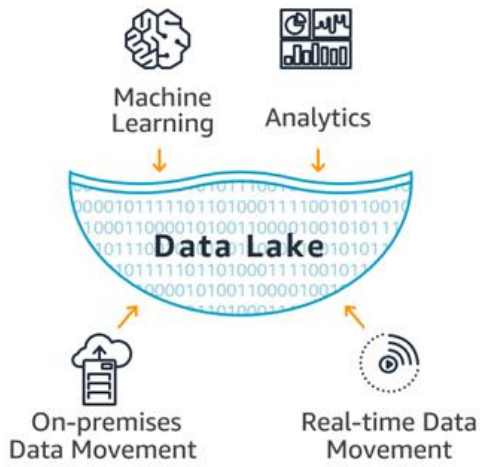
R

Modeling & Exploratory Data Analysis

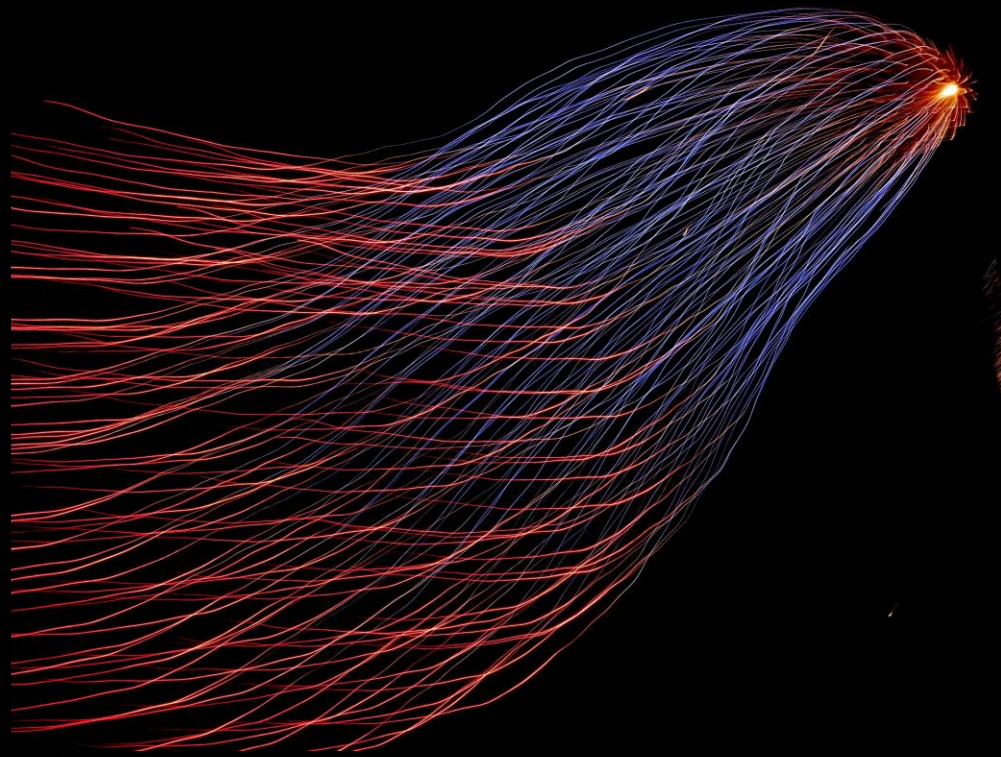
Most economists only consider this portion of the data pipeline!



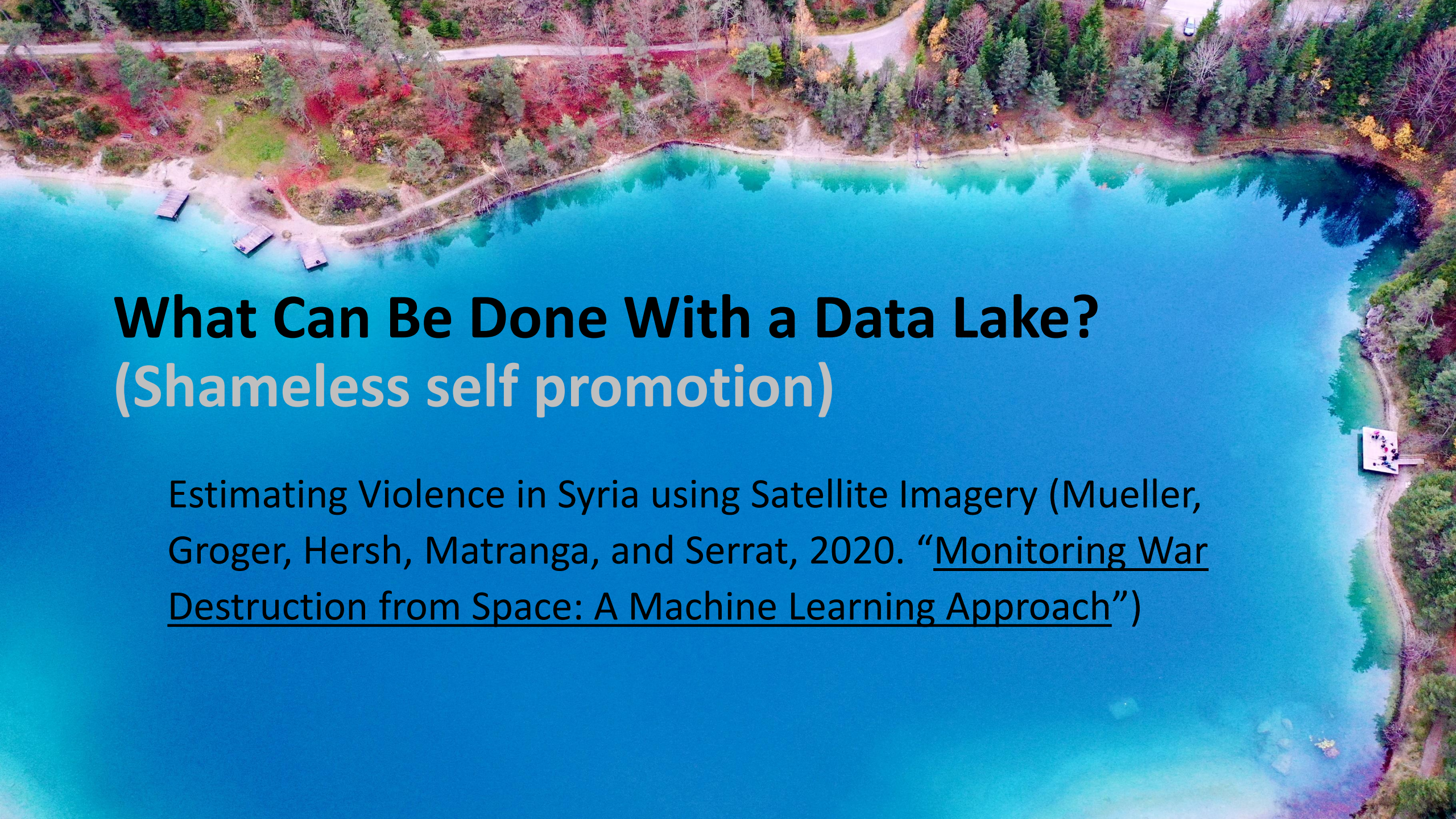
Many Options of Data Lakes to Choose From



Questions at This Point



- What's best data lake?
  - What's the best food when you're starving?
  - Cloud for sure, open source if you can
- What programming languages do I need to learn?
  - None really. Version control (git). R and Python will make your life easier. SQL variants (or just dbplyr)
- **Theme: Have a consistent data strategy and ensure everyone abides by it**

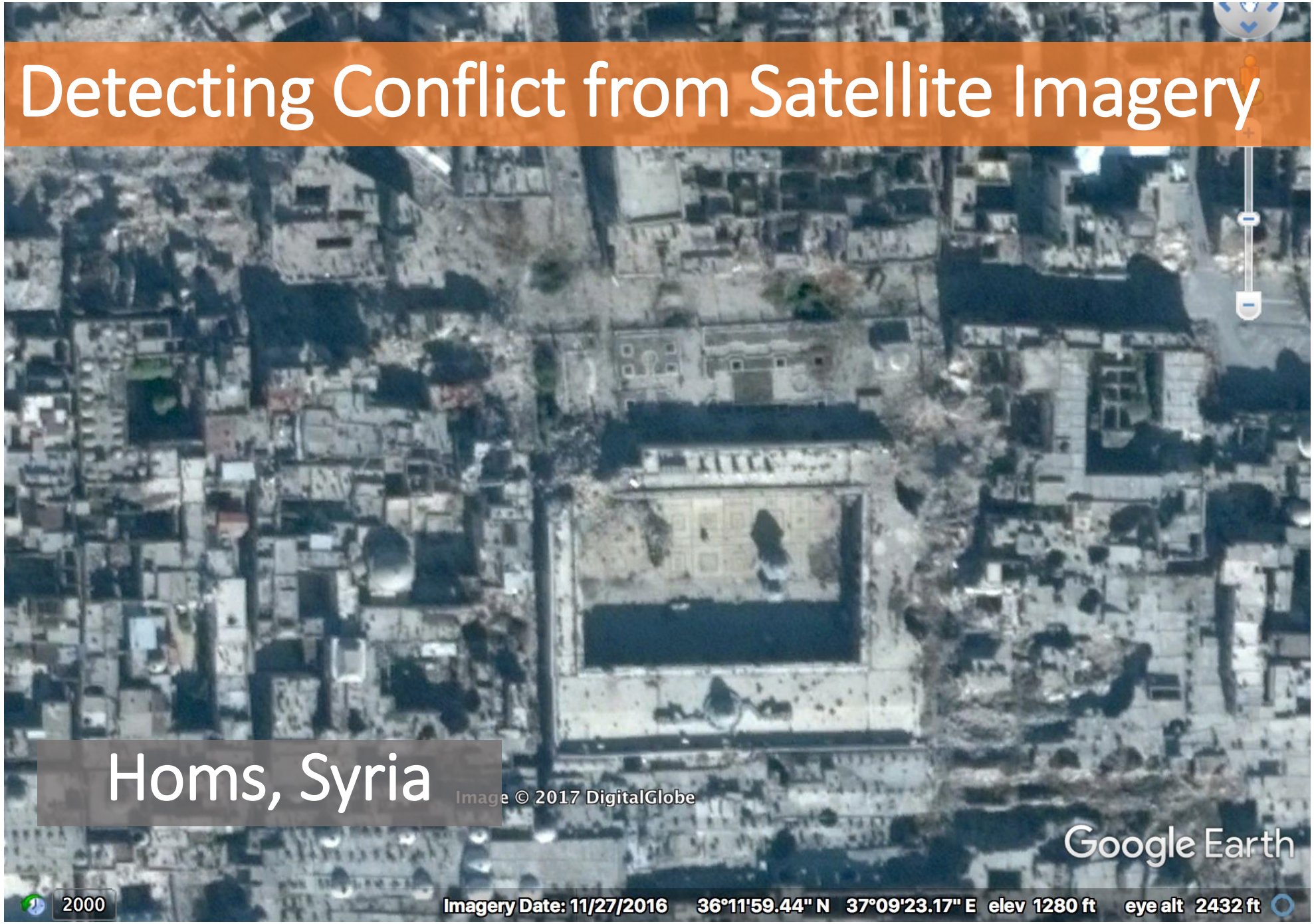


What Can Be Done With a Data Lake?
(Shameless self promotion)

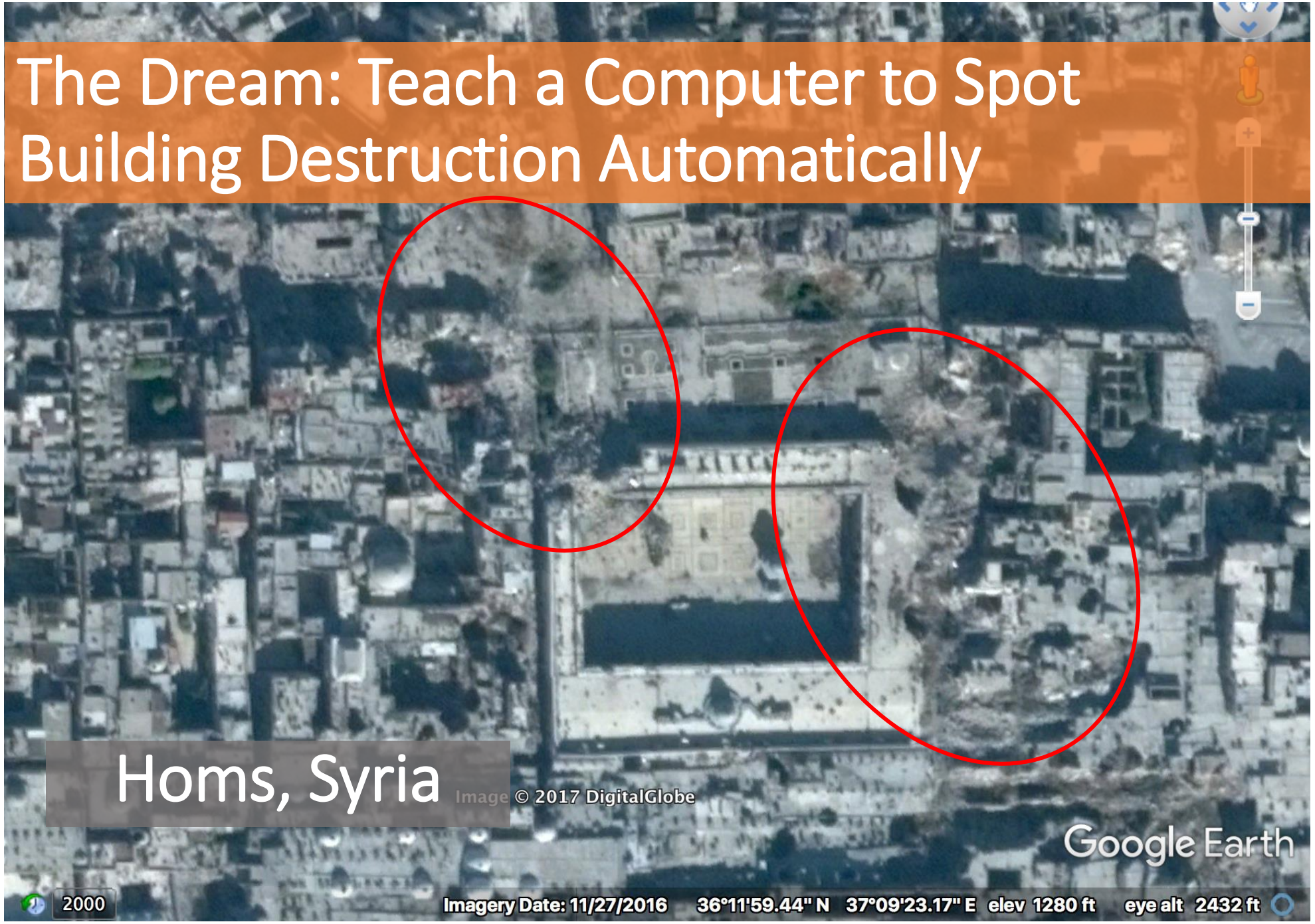
Estimating Violence in Syria using Satellite Imagery (Mueller, Groger, Hersh, Matranga, and Serrat, 2020. “Monitoring War Destruction from Space: A Machine Learning Approach”)

Information on ongoing violence may be biased, outdated, and imprecise

Detecting Conflict from Satellite Imagery



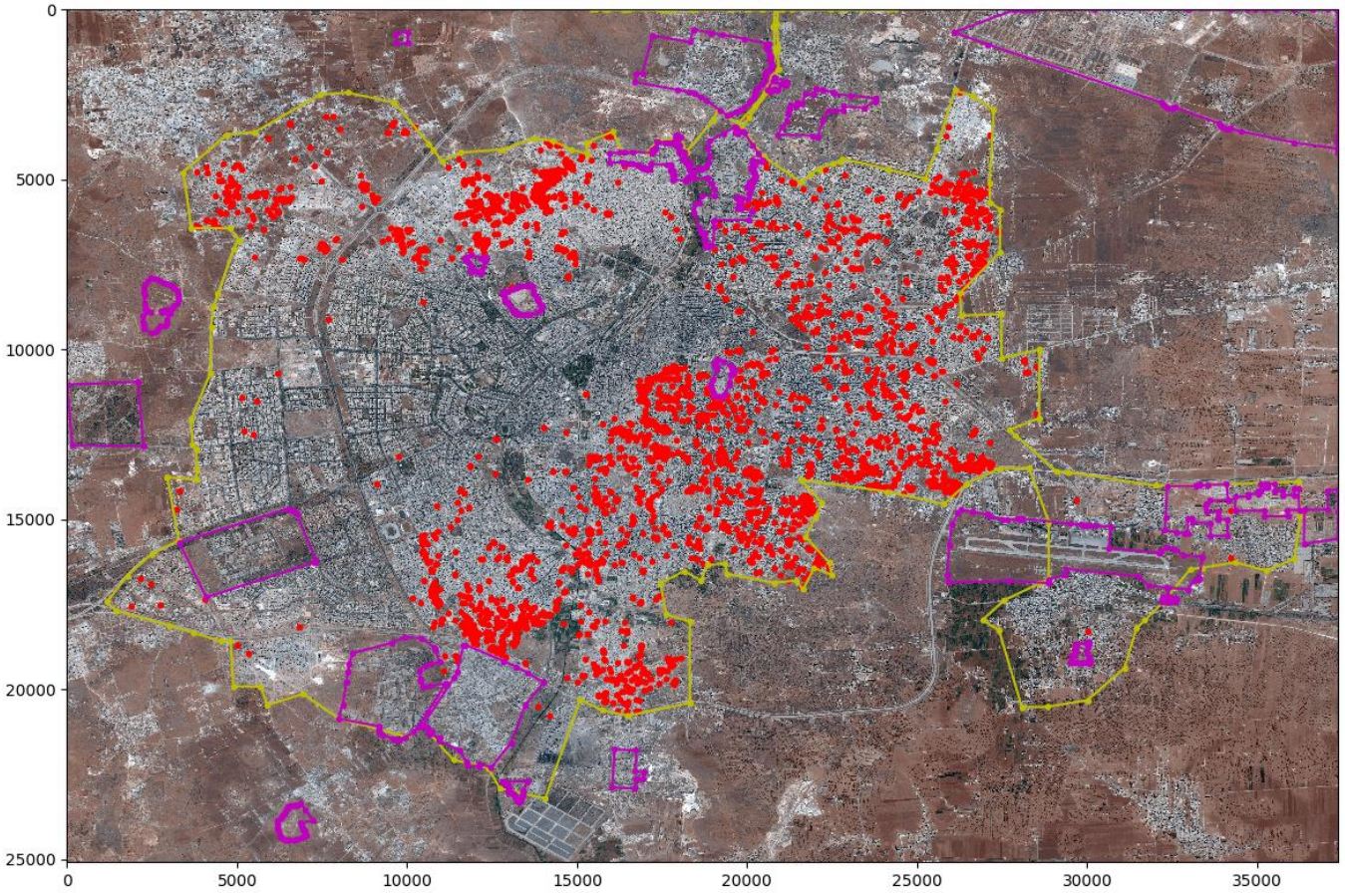
The Dream: Teach a Computer to Spot Building Destruction Automatically

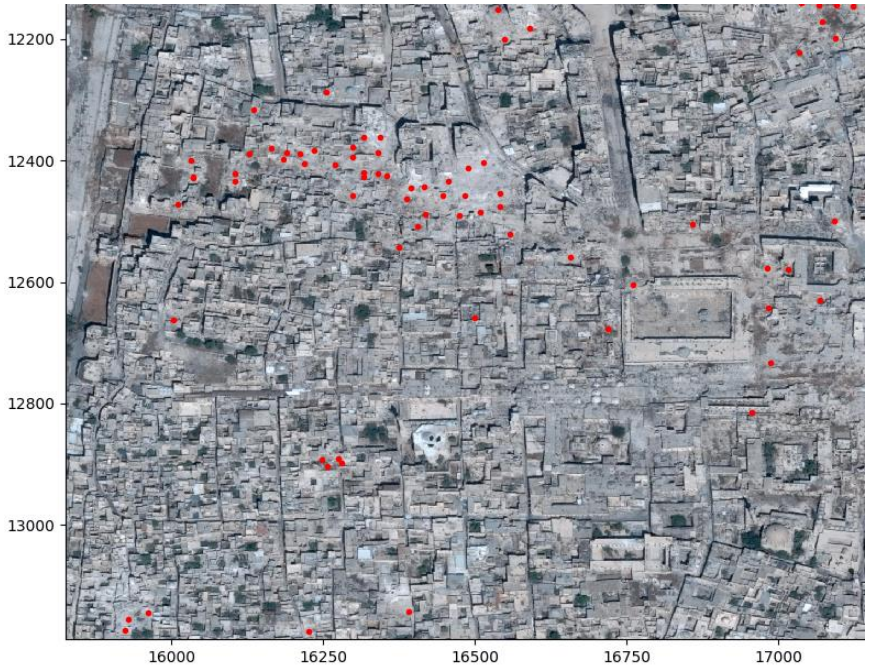


This Paper:

- Train CNN deep learning architecture to recognize building destruction from Google Maps imagery
- Training data: UNOSAT hand-labeled building destruction
  - Novel data augmentation approach to expand training labels to 2.2m
- Novel second machine learning stage that uses temporal and contextual information to increase precision due to imbalanced data
- Validate using external bombing event data

Why is this hard?
Destruction is Sparse Relative to Non-Destruction





Only 2.3% of images show any destruction in our sample of six Syrian cities

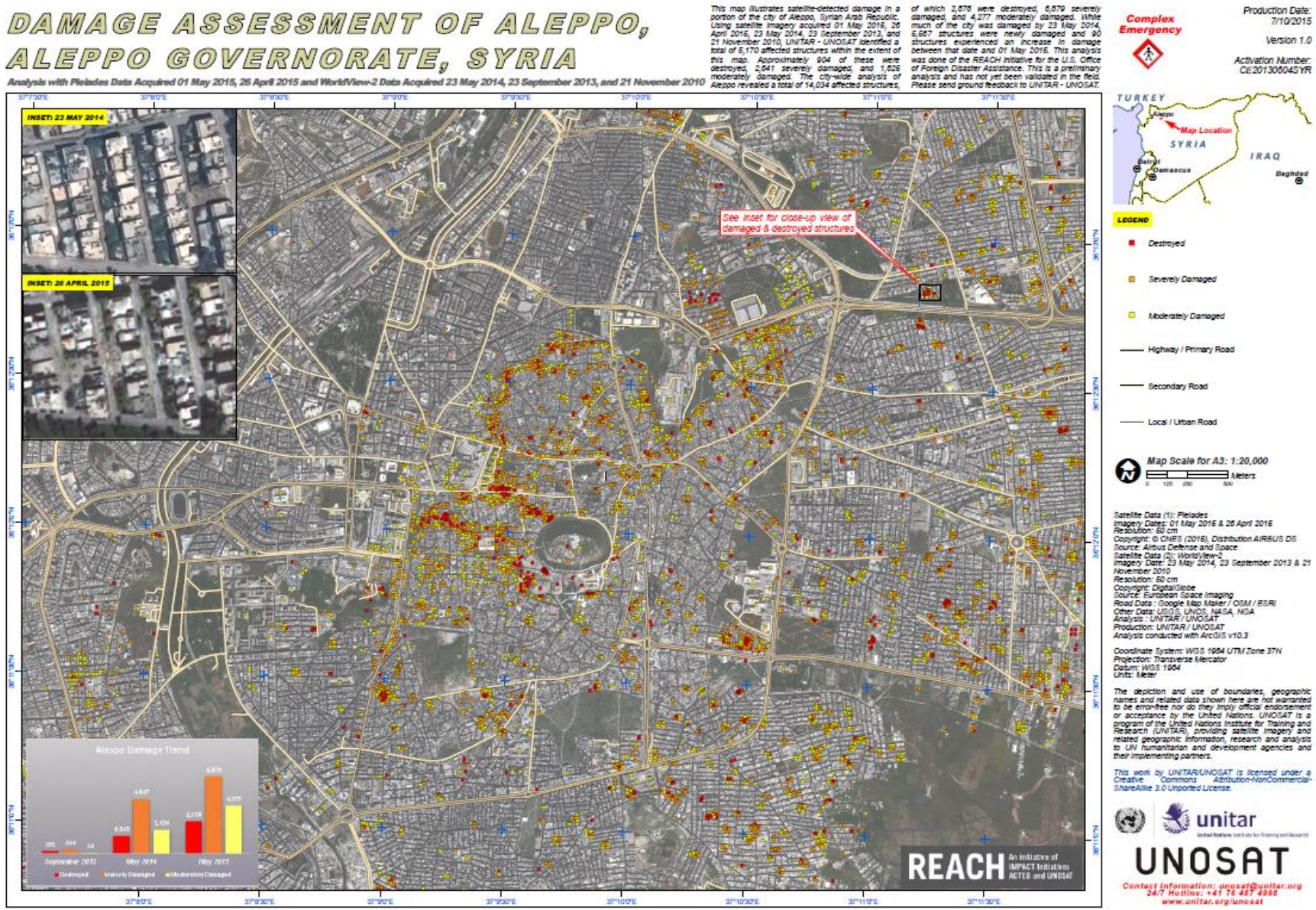
Illustration of Unbalanced Data Problem

- **Accuracy**: $\frac{\mathrm{TP}+\mathrm{TN}}{TP+FN+FP+TN}$
- **Precision** (share of positives predicted correctly): $\frac{\mathrm{TP}}{TP+FP}$
- **True Positive Rate / Recall** (share of actual positives predicted that are correct): $\frac{\mathrm{TP}}{TP+FN}$

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

- Suppose you have 100,000 images, but only 1000 (1%) are destroyed
- 12% FRP means your model produces: 99,000 [FP + TN] * 0.12[FPR] = 11,880 FPs
- 80% TPR/recall means your model produces 1000*0.8 TP = 800 TPs
- Precision for that model is: $\frac{\mathrm{TP}}{TP+FP} = \frac{800}{11880+800} = 0.063$
- Probability of being correct if you find destruction is only 6.3%!

Ground Truth: Building Destruction Annotations from UNOSAT



Label Augmentation Method

- To increase the number of training labels, we apply a novel data augmentation method based on assumptions of the data generating process
- **No reconstruction** => label destroyed at time t, remain destroyed t + 1
- **No reconstruction** => label not destroyed at t, is also not destroyed t - 1

T=1	T = 2	T = 3	T=4
			
			

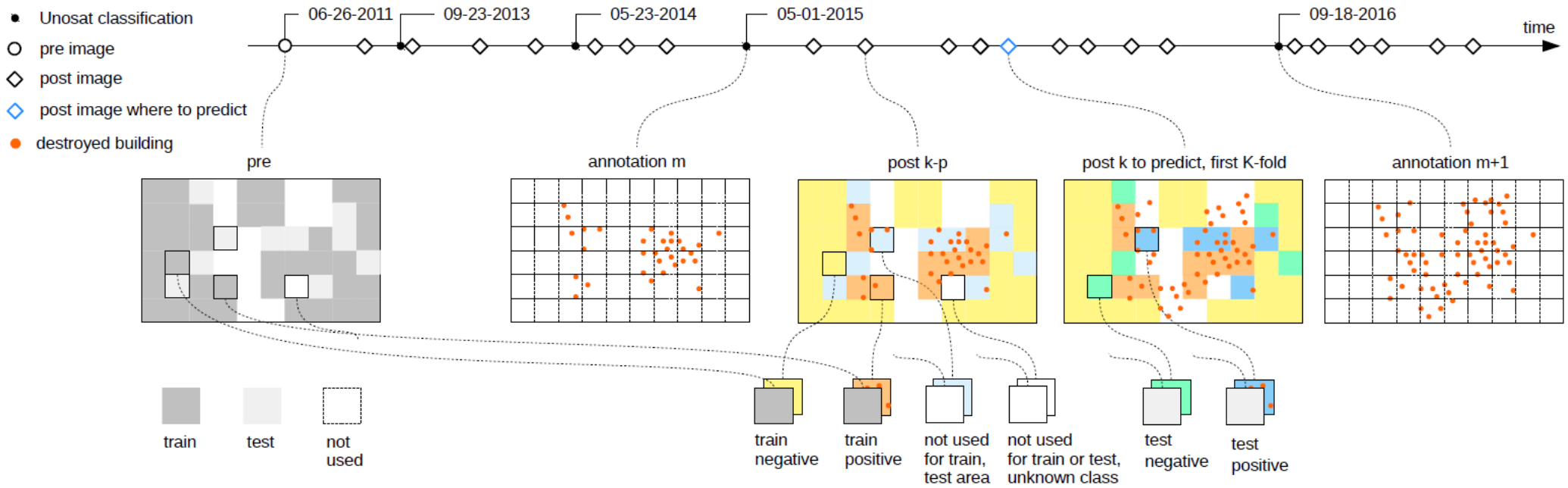

Label


Augmented label

Images By City Over Time

City	Total images	Temporal periods (post period)	Labeled images (with augmentation)	Share of images with destruction
Aleppo	2,106,412	22	1,626,920	1.82%
Daraa	202,462	13	125,231	1.00%
Deir-Ez-Zor	98,602	7	84,723	2.86%
Hama	285,057	9	224,365	3.73%
Homs	200,035	5	83,941	8.26%
Raqqa	180,184	8	112,481	2.14%
All	3,072,752	64	2,257,661	2.37%

Sampling for Testing and Training



- Four temporal periods with annotations (at patch level)
- Training and testing samples are from distinct spatial areas

Neural Network Architecture

- We use a Convolutional Neural Network (CNN), which is a series of neural network filters, where the filters have been optimized for the prediction task
- We tried many architectures. Standard (ResNet, VGG16), and boutique (U-Net)
- In the end we use a simple but flexible 2 convolutional layer network
- We use a random search algorithm to sample hyperparameters:
  - Number of convolutional layers
  - Number of neurons of the fully connected layer
  - Filters and kernels of the convolutional layer and activation functions (relu vs sigmoid)
  - Pooling size and max pooling layer
  - Varying dropout
  - Different epochs, batch sizes, class weight

Second Stage Machine Learning Smoothing

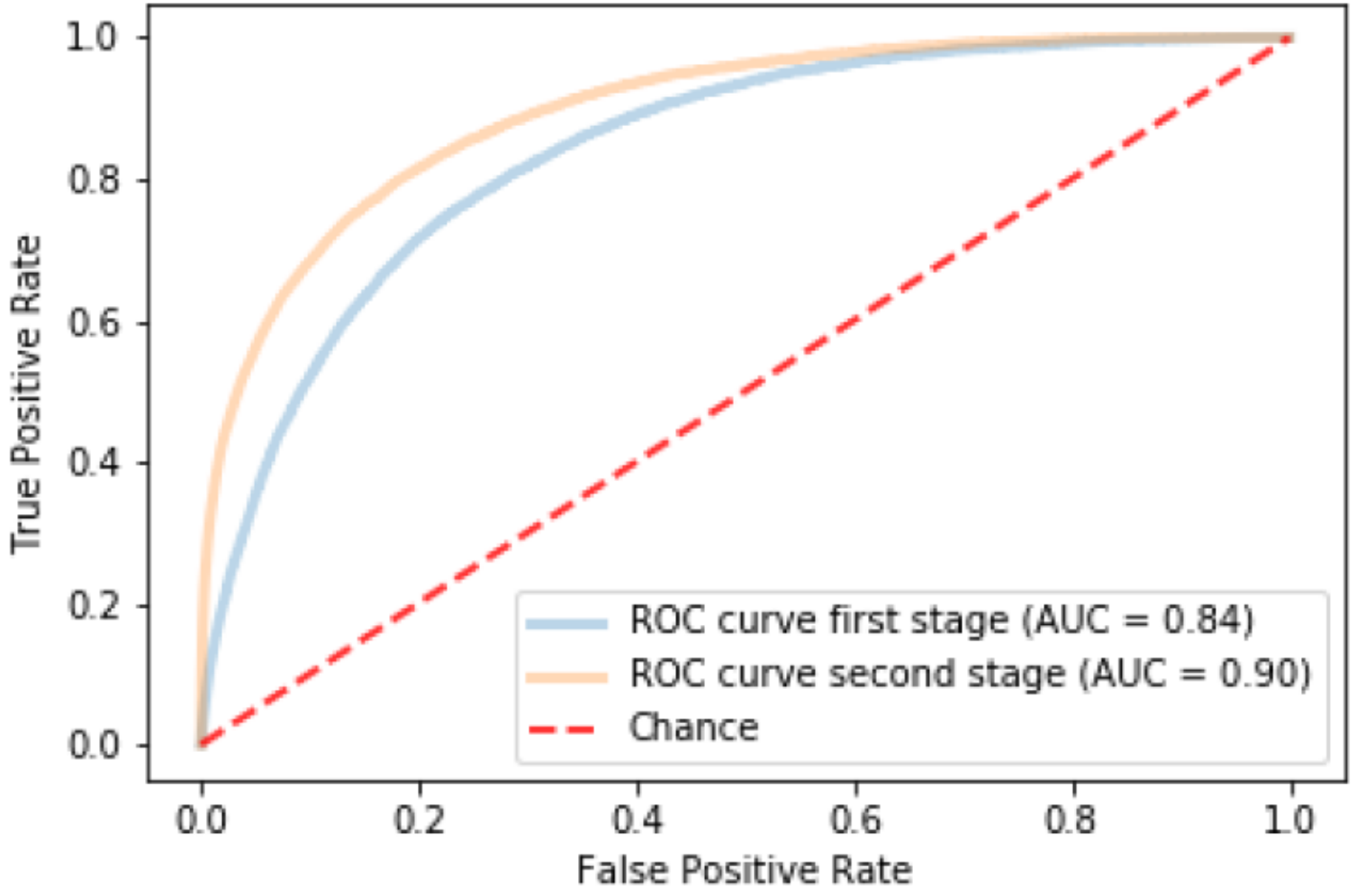
- Idea: destruction is correlated in space and time
- We train a random forest model on the CNN predictions using two spatial lags and two temporal lags
- This stage separated from CNN stage for maximum flexibility and modularity

Table 2. Model precision varying second stage

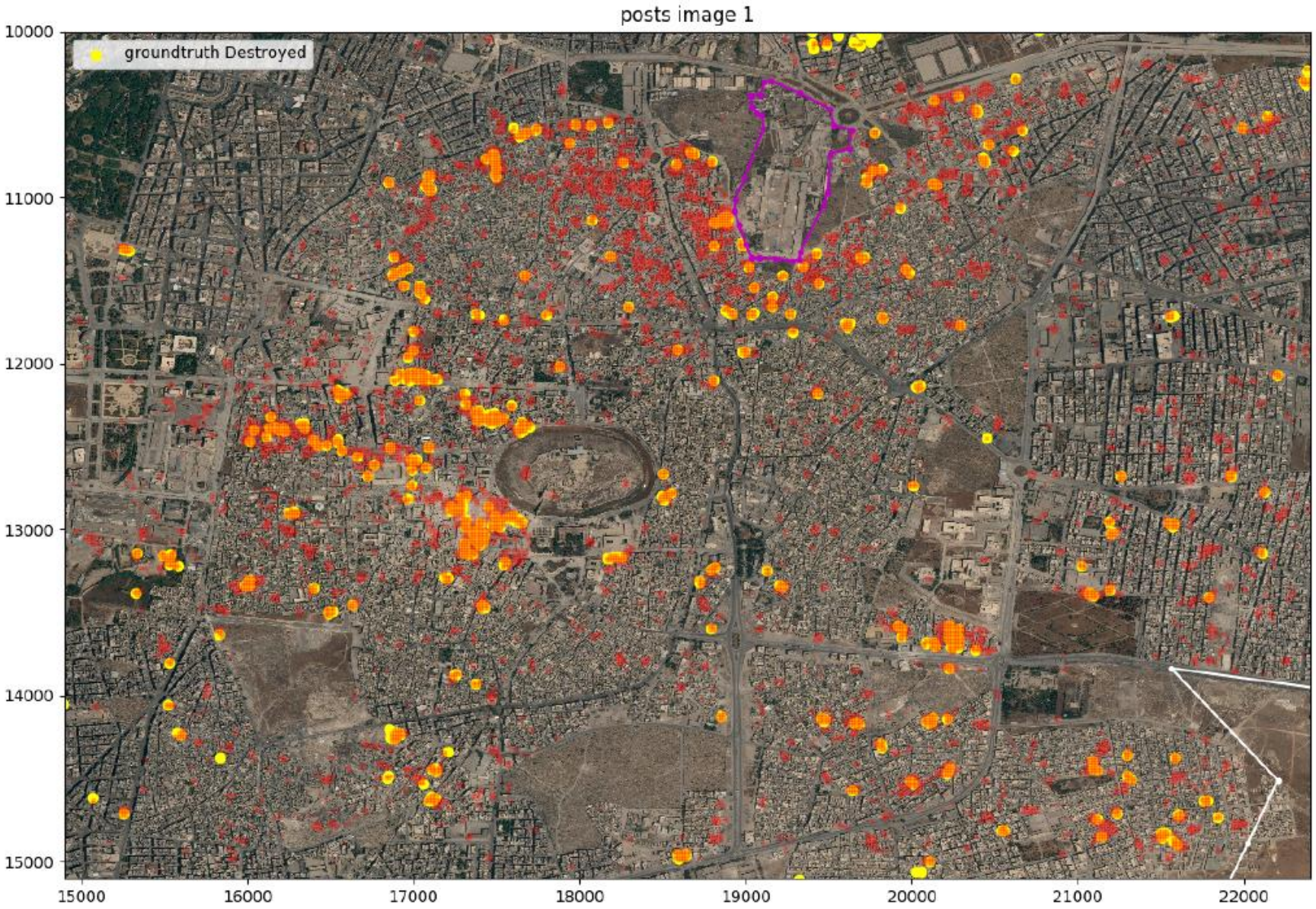
City	(1) First stage	(2)	(3) Second stage	(4)
	CNN	CNN + Spatial	CNN + Spatial & 1 temporal lag/lead	CNN + Spatial & 2 temporal lag/leads
Aleppo	16.1%	16.8%	28.2%	35.7%
Daraa	4.2%	4.6%	9.5%	12.1%
Deir-Ez-Zor	11.0%	12.0%	18.6%	21.9%
Hama	54.5%	65.3%	67.5%	68.0%
Homs	25.8%	35.3%	44.6%	56.1%
Raqqa	12.8%	17.8%	20.9%	31.8%
All	24.5%	28.7%	37.4%	42.7%

Sources: Author calculations, UNITAR/UNOSAT damage annotations for Syria.

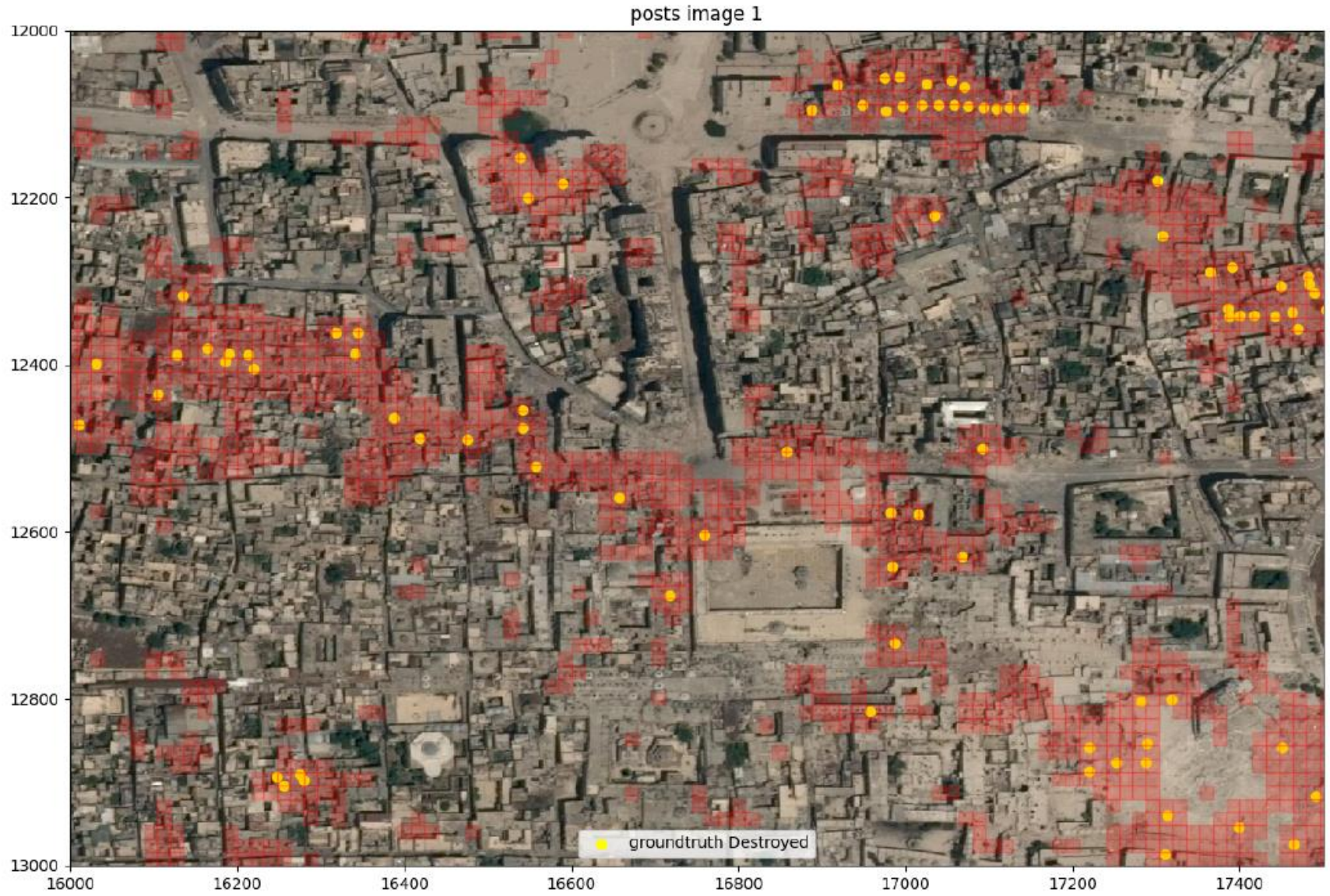
Aleppo ROC Curve (Test Sample)



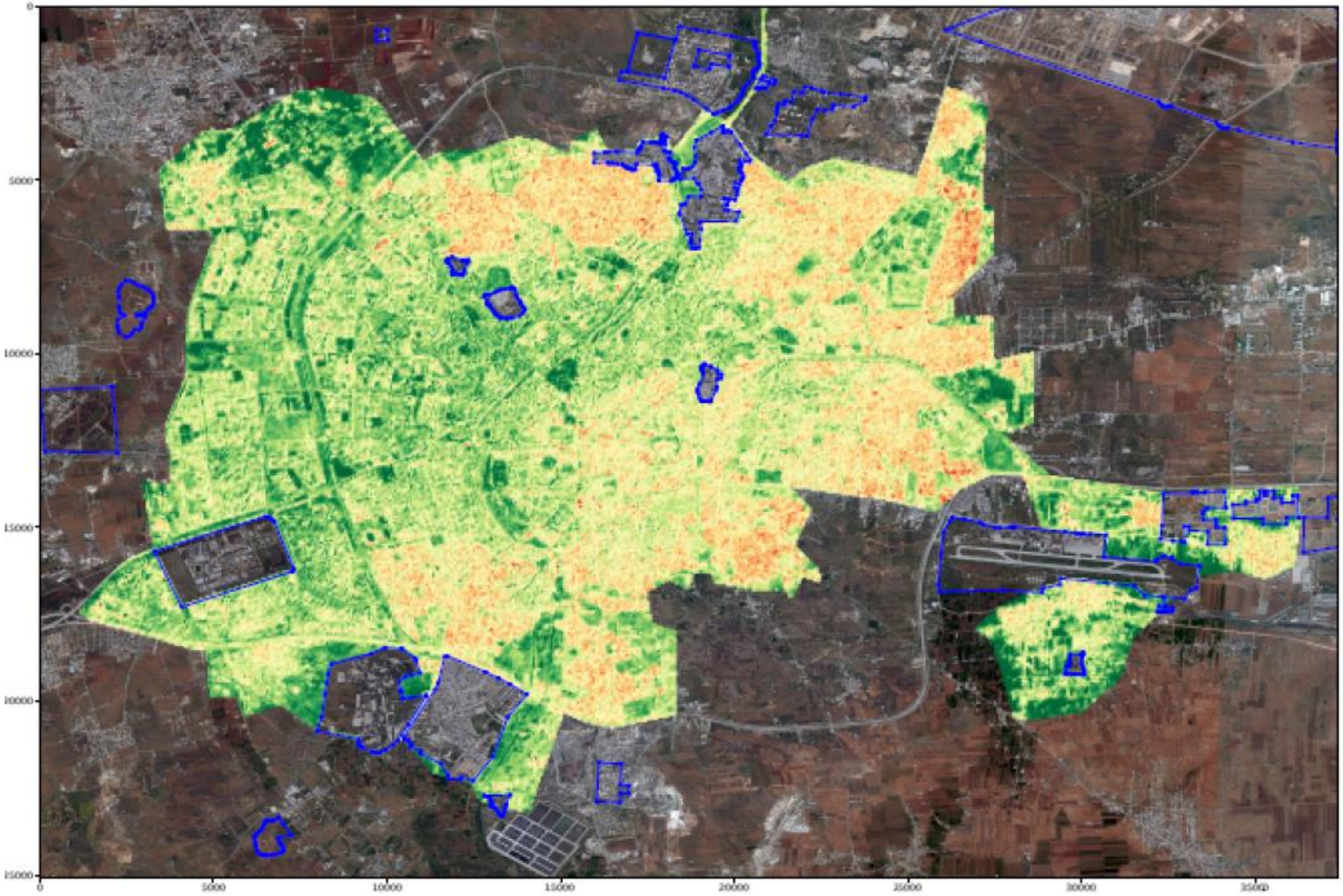
Destruction Predictions Aleppo



Destruction Predictions Aleppo



Destruction Predictions Aleppo



Conclusion

- We propose a novel data augmentation and second stage machine learning stage to increase precision of building destruction
- 2nd ML stage doubles precision, to 42%, good enough for automated detection
- Predictions of building destruction are validated using external bombing event data
- Next steps:
  - Other cities?
  - Severity of damage?

Comments/suggestions appreciated!

Jonathan Hersh

Assistant Professor

Argyros School of Business, Chapman University

hersh@chapman.edu