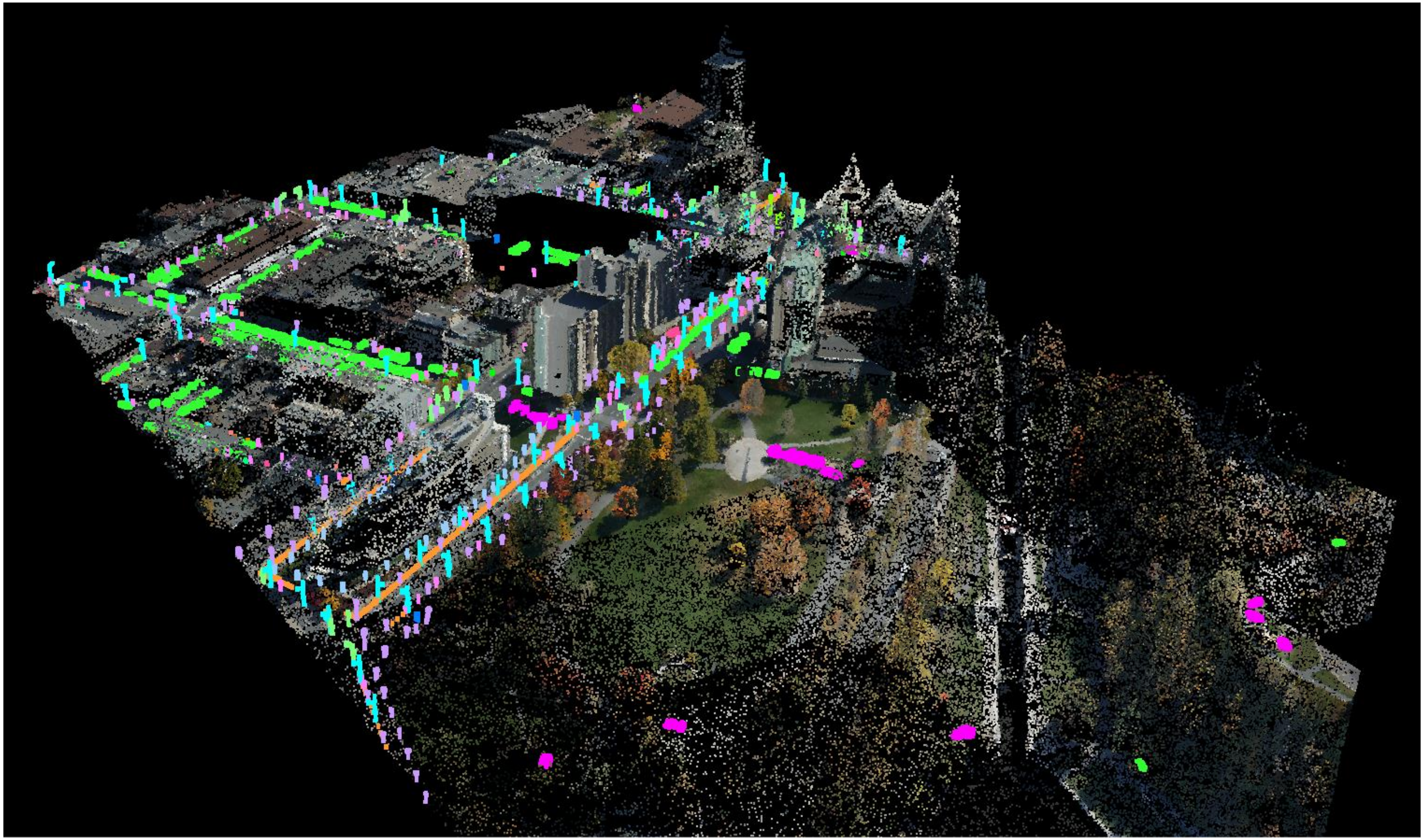
# Efficient Interactive Labeling of Small Objects in Urban LIDAR Scans

Thomas Funkhouser
Princeton University

# Motivation

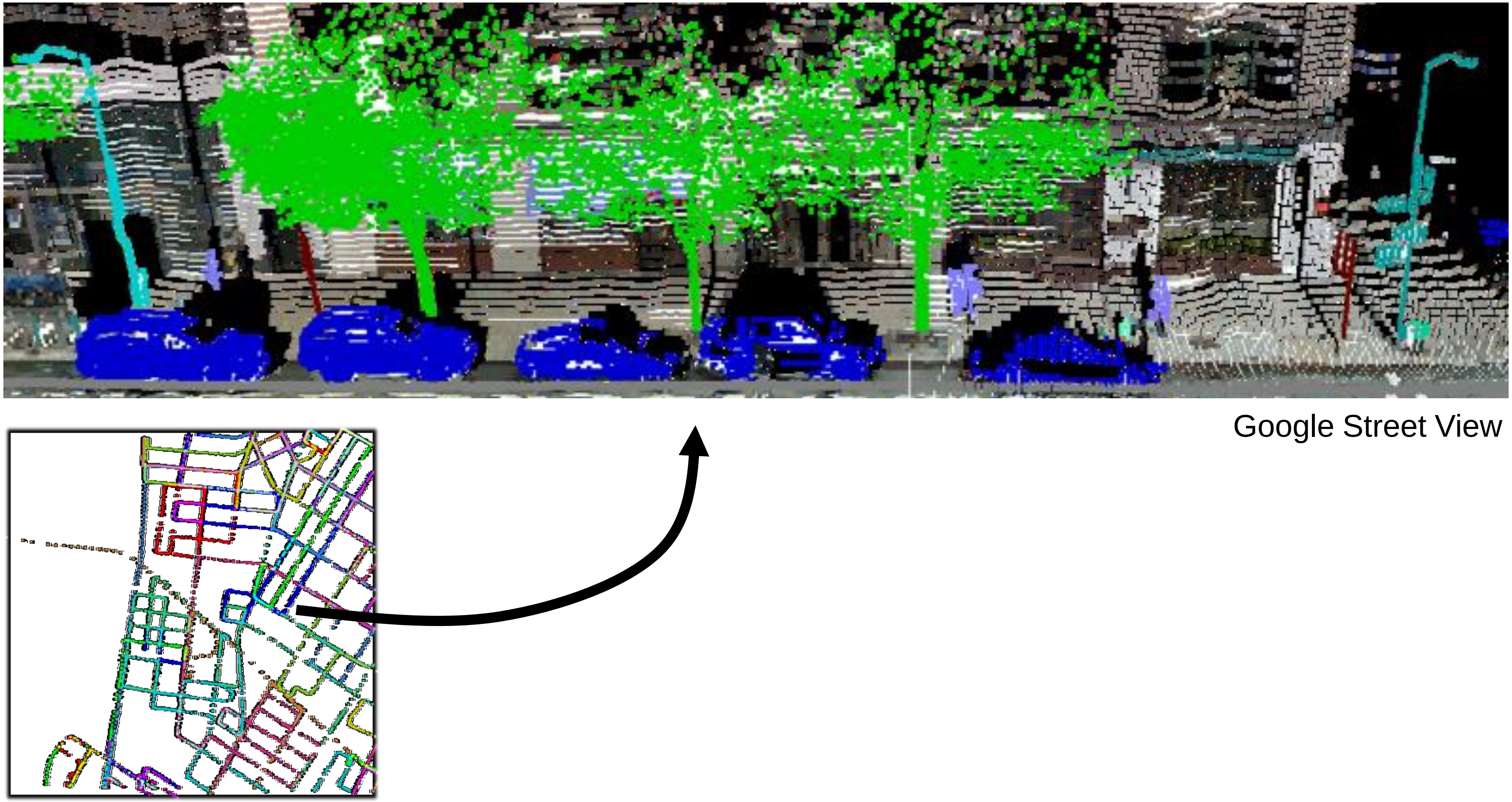
Semantic modeling of cities with labeled small objects



Neptec

# Motivation

Semantic modeling of cities with labeled small objects

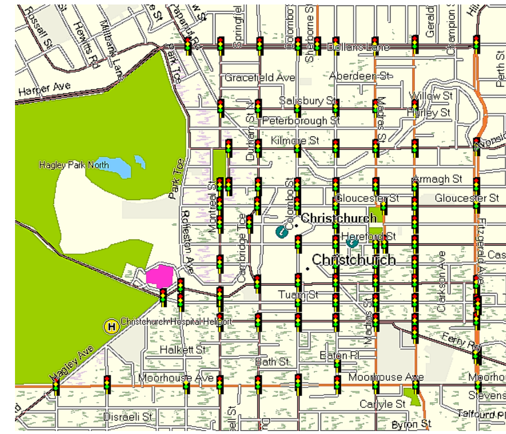


Google Street View

# Applications

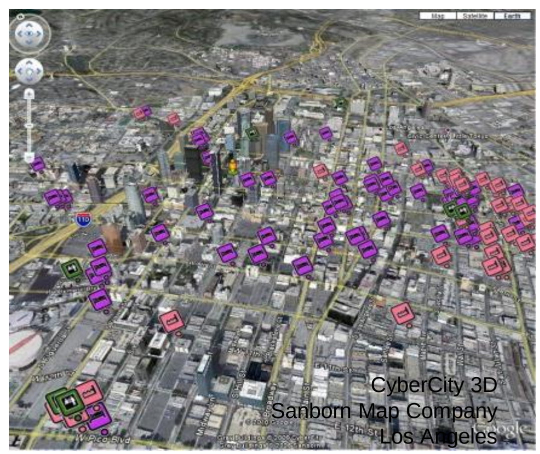

Mobile augmented reality


Simulation




Mapping

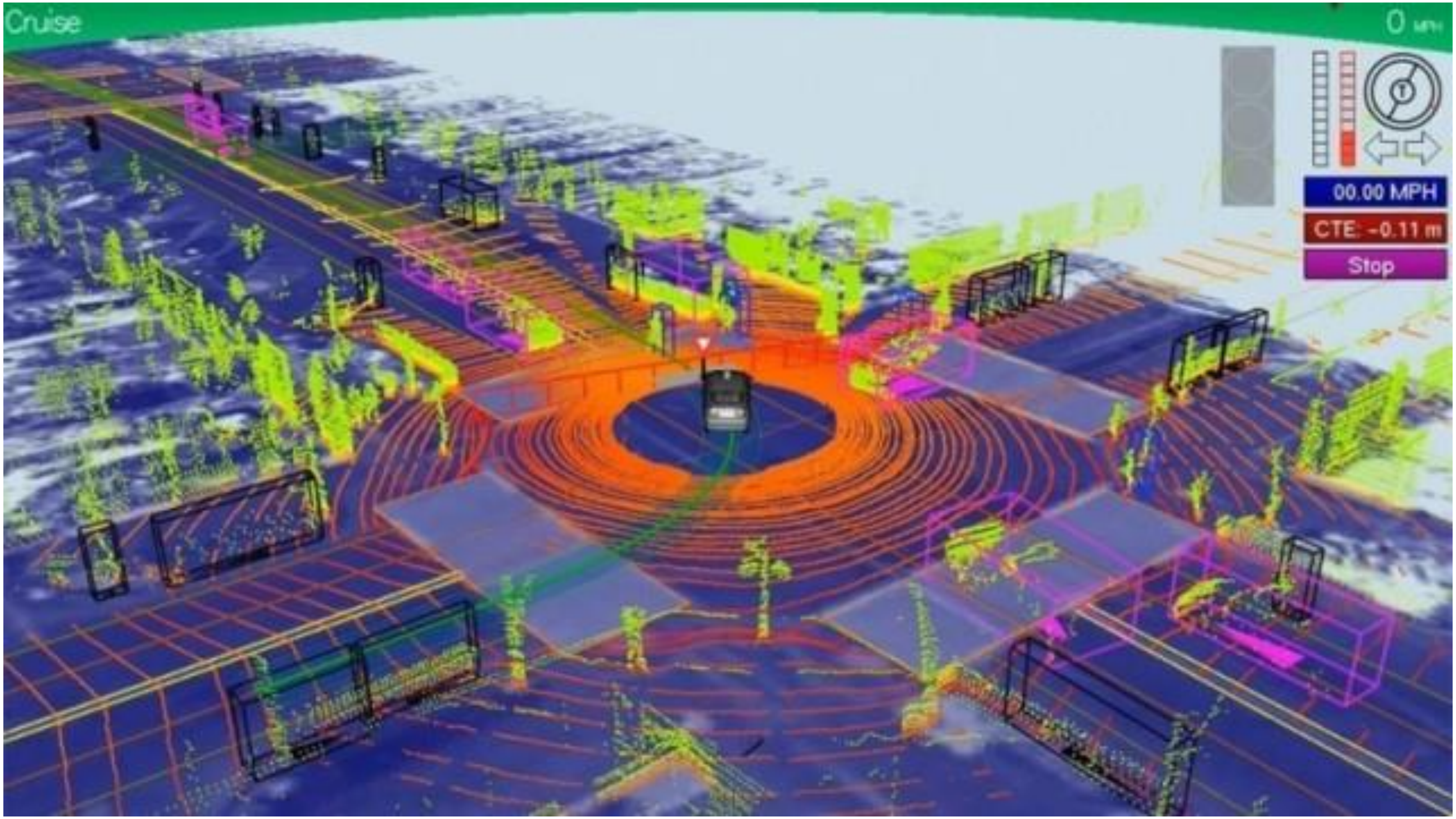

Urban planning




Anthropology


Driving Simulation

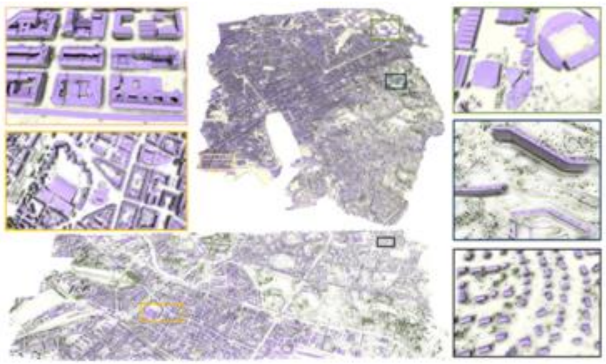
# Applications




Semantic maps for self-driving cars

# Automatic Semantic Segmentation?

Large-scale structures (roads, buildings, etc.):



Lafarge et al, 2011

Lafarge et al, 2010

Poullis et al, 2009

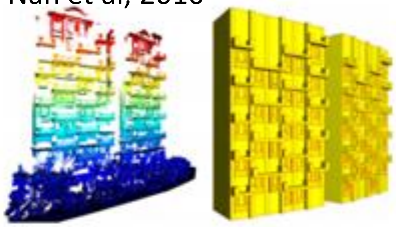
Pauly et al, 2008

Zhou et al, 2010

Boyko et al, 2011

Musialski et al, 2012

Nan et al, 2010

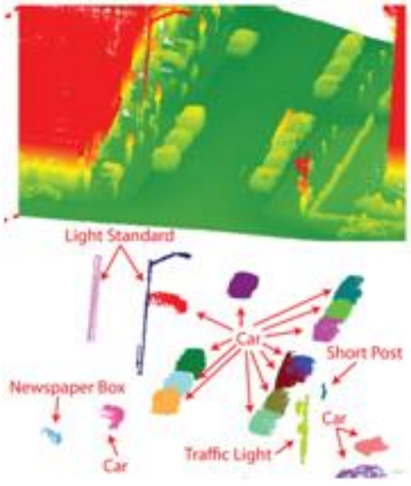


# Automatic Semantic Segmentation?

Roadside objects (cars, signs, lights, pedestrians, etc.)

Linet et al, 2013

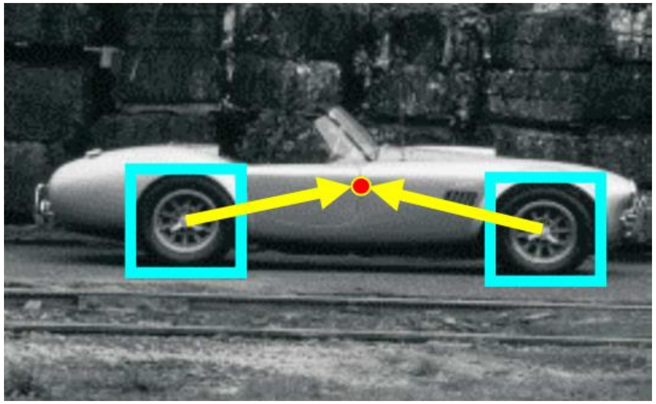
Velizhev et al, 2012



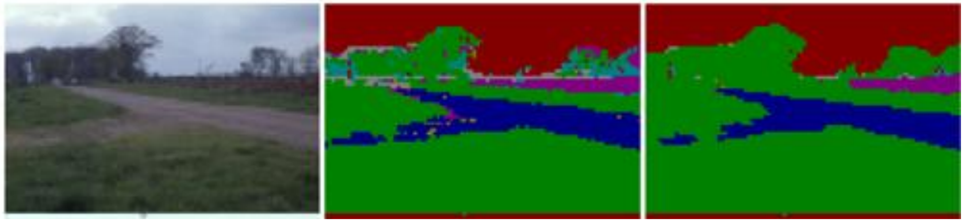

Patterson et al, 2008

# Automatic Semantic Segmentation?

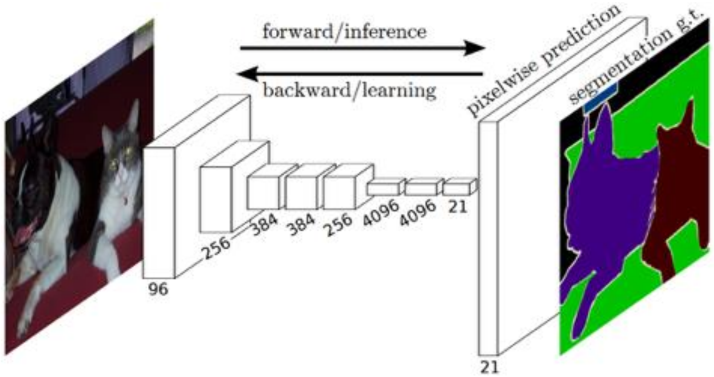
Roadside objects (cars, signs, lights, pedestrians, etc.)



Implicit Shape Model
(e.g., Liebe et al., 2008)



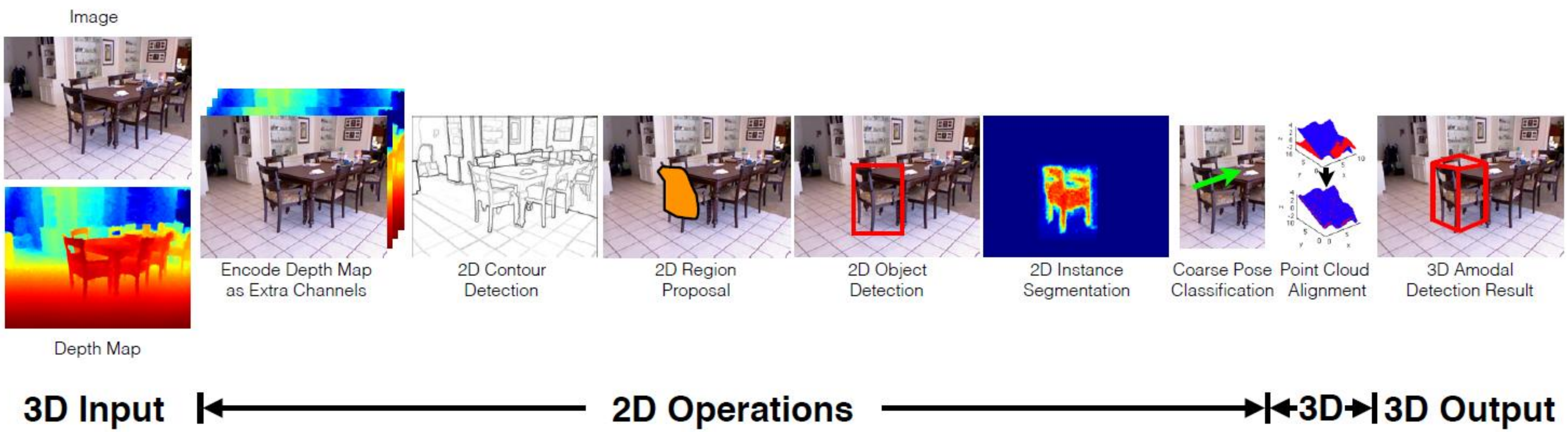
Conditional Random Field
(e.g., Wojek et al., 2008)




Deep Learning
(e.g., Long et al., 2015)

# Automatic Semantic Segmentation?

Indoor objects (chairs, tables, desks, etc.)

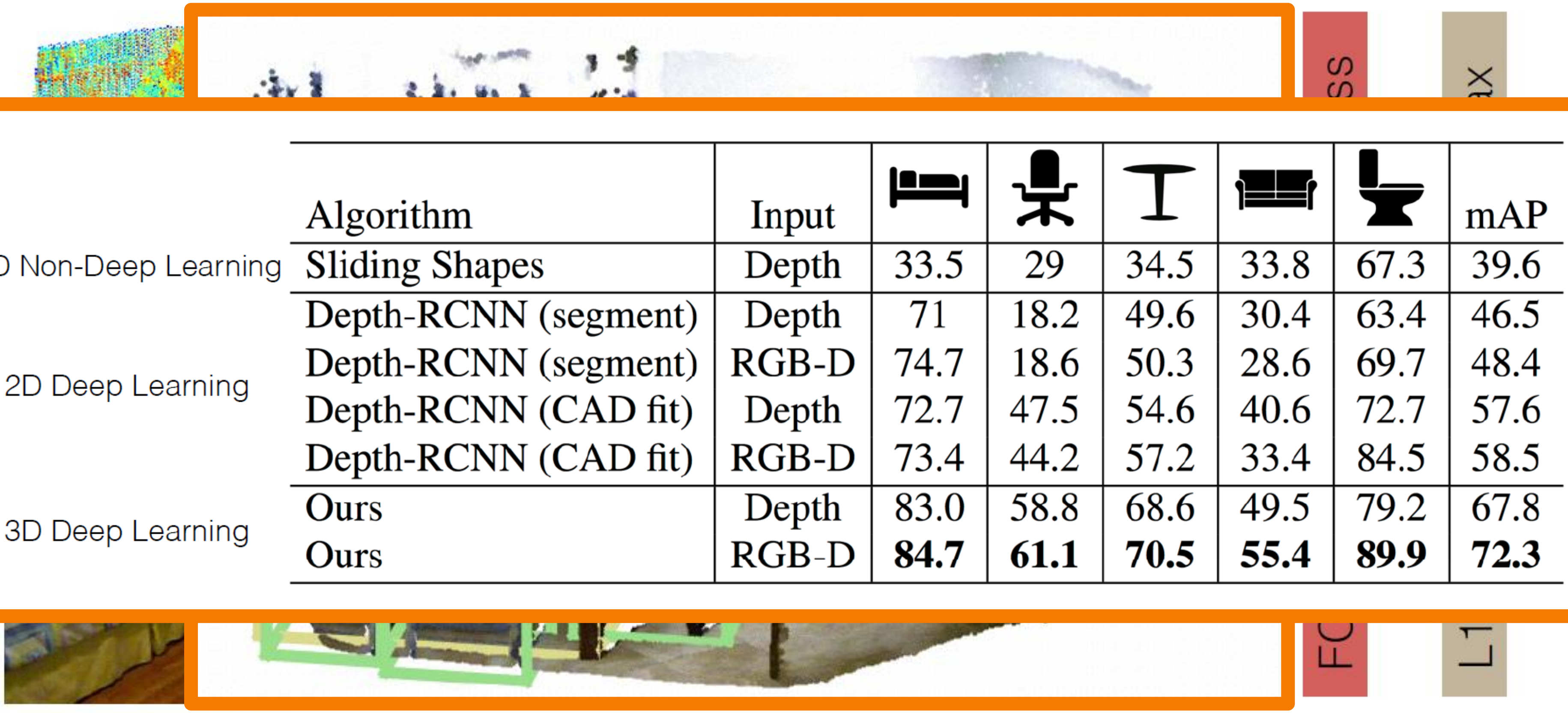



[CVPR13] Perceptual Organization and Recognition of Indoor Scenes from RGB-D Images
[IJCV14] Indoor Scene Understanding with RGB-D Images: Bottom-up Segmentation, Object Detection and semantic segmentation
[ECCV14] Object Detection and Segmentation using Semantically Rich Image and Depth Features
[CVPR15] Aligning 3D Models to RGB-D Images of Cluttered Scenes
[CVPR16] Cross Modal Distillation for Supervision Transfer

2D Deep Learning
(e.g., Gupta et al., 2016)

# Automatic Semantic Segmentation?

Indoor objects (chairs, tables, desks, etc.)

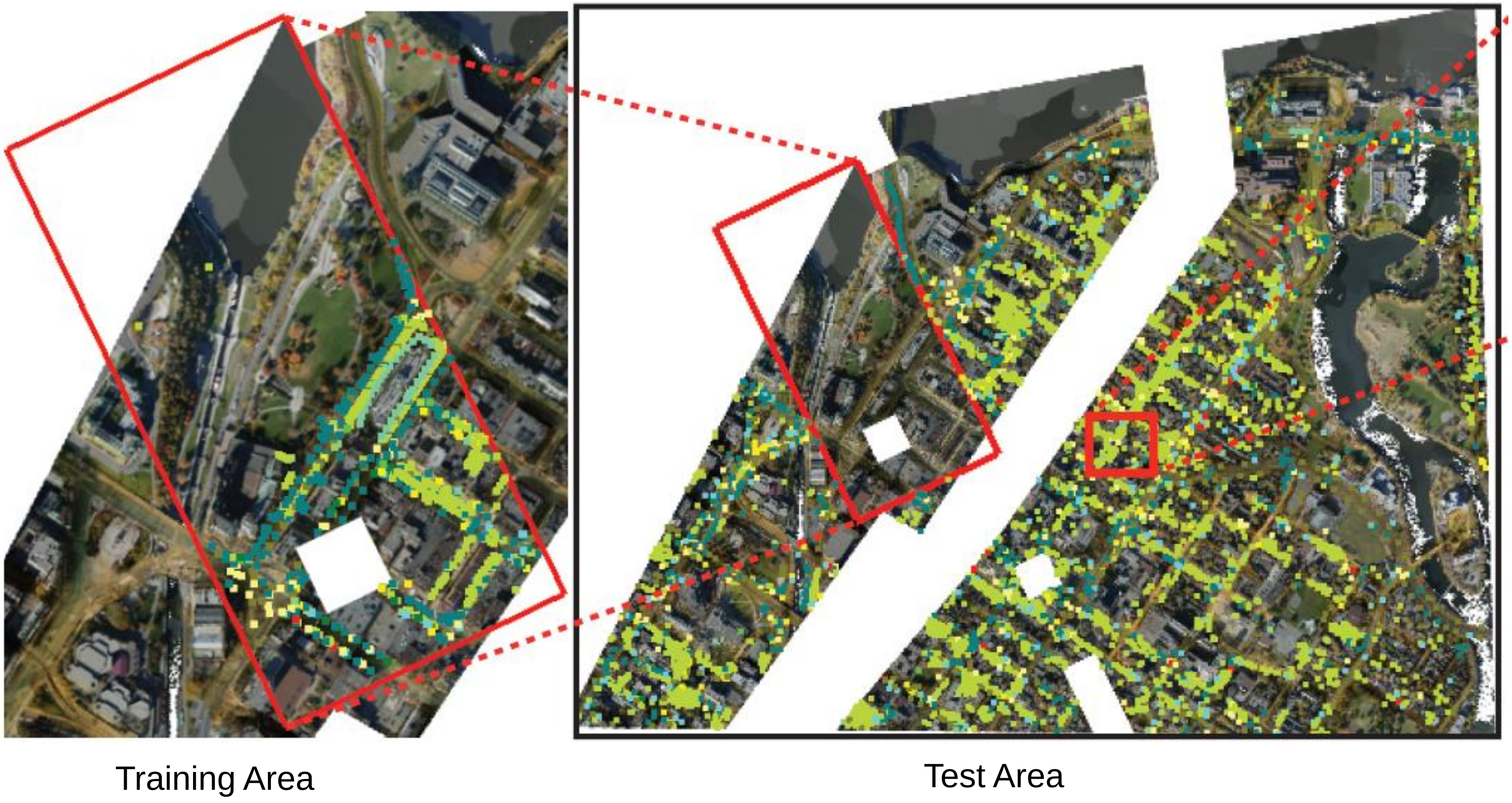


| | Algorithm | Input | | | | | | mAP |
|---|---|---|---|---|---|---|---|---|
| 3D Non-Deep Learning | Sliding Shapes | Depth | 33.5 | 29 | 34.5 | 33.8 | 67.3 | 39.6 |
| 2D Deep Learning | Depth-RCNN (segment) | Depth | 71 | 18.2 | 49.6 | 30.4 | 63.4 | 46.5 |
| | Depth-RCNN (segment) | RGB-D | 74.7 | 18.6 | 50.3 | 28.6 | 69.7 | 48.4 |
| | Depth-RCNN (CAD fit) | Depth | 72.7 | 47.5 | 54.6 | 40.6 | 72.7 | 57.6 |
| | Depth-RCNN (CAD fit) | RGB-D | 73.4 | 44.2 | 57.2 | 33.4 | 84.5 | 58.5 |
| 3D Deep Learning | Ours | Depth | 83.0 | 58.8 | 68.6 | 49.5 | 79.2 | 67.8 |
| | Ours | RGB-D | **84.7** | **61.1** | **70.5** | **55.4** | **89.9** | **72.3** |

3D Deep Learning
(e.g., Song et al., 2016)

# Automatic Semantic Segmentation?

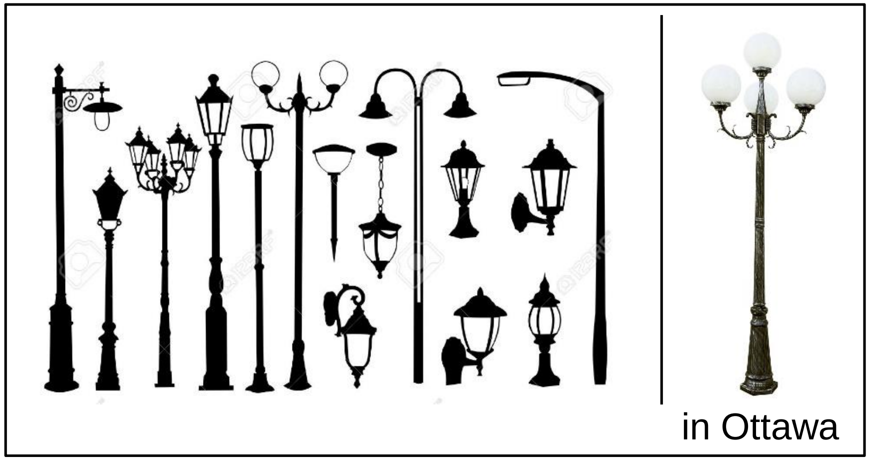
Automatic supervised algorithms require training sets



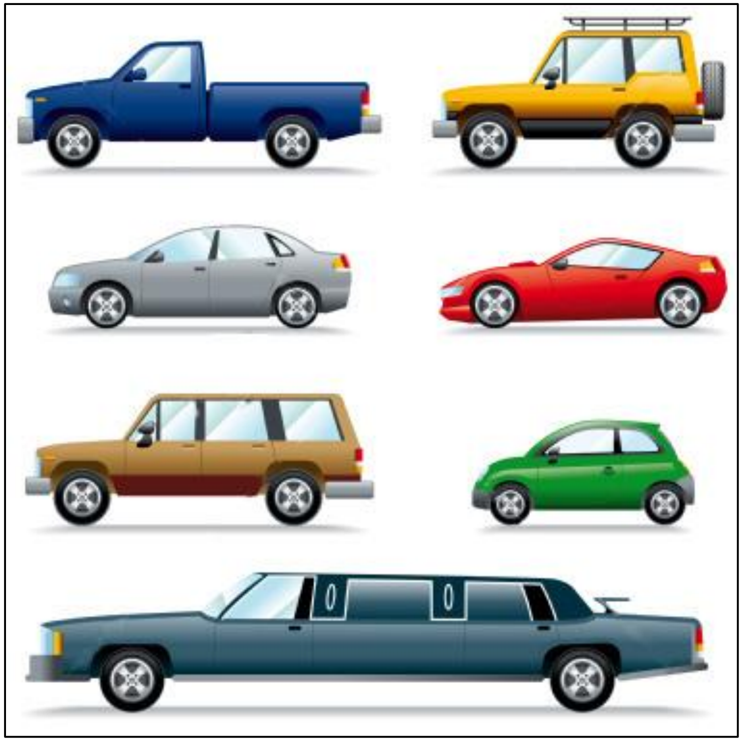

Creating training sets requires manual annotation

# Automatic Semantic Segmentation?

"Automatic" supervised algorithms require training sets **and fine-tuning for every test set**




Different types of sidewalk lamps



Vehicle? Car? Honda? Accord?

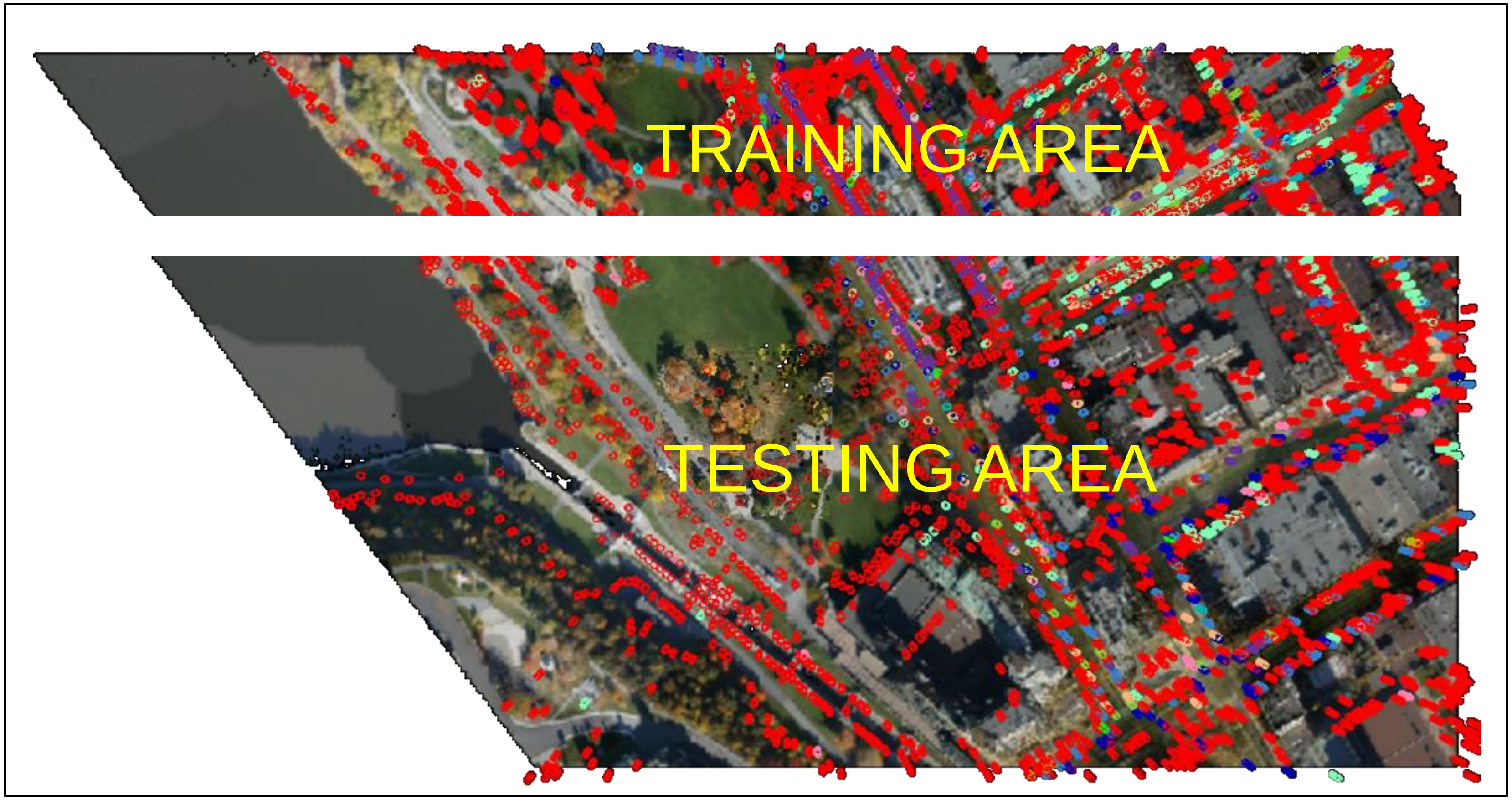
Creating fine-tuning training sets requires manual annotation

# Manual Annotation is Necessary

What manual annotation method is best?

# What Manual Annotation Method is Best?

Typical method: manually annotate training set, learn model, and apply model to test set



# What Manual Annotation Method is Best?

Typical method: manually annotate separate training set, learn model, apply model to test set, **and then fix errors**

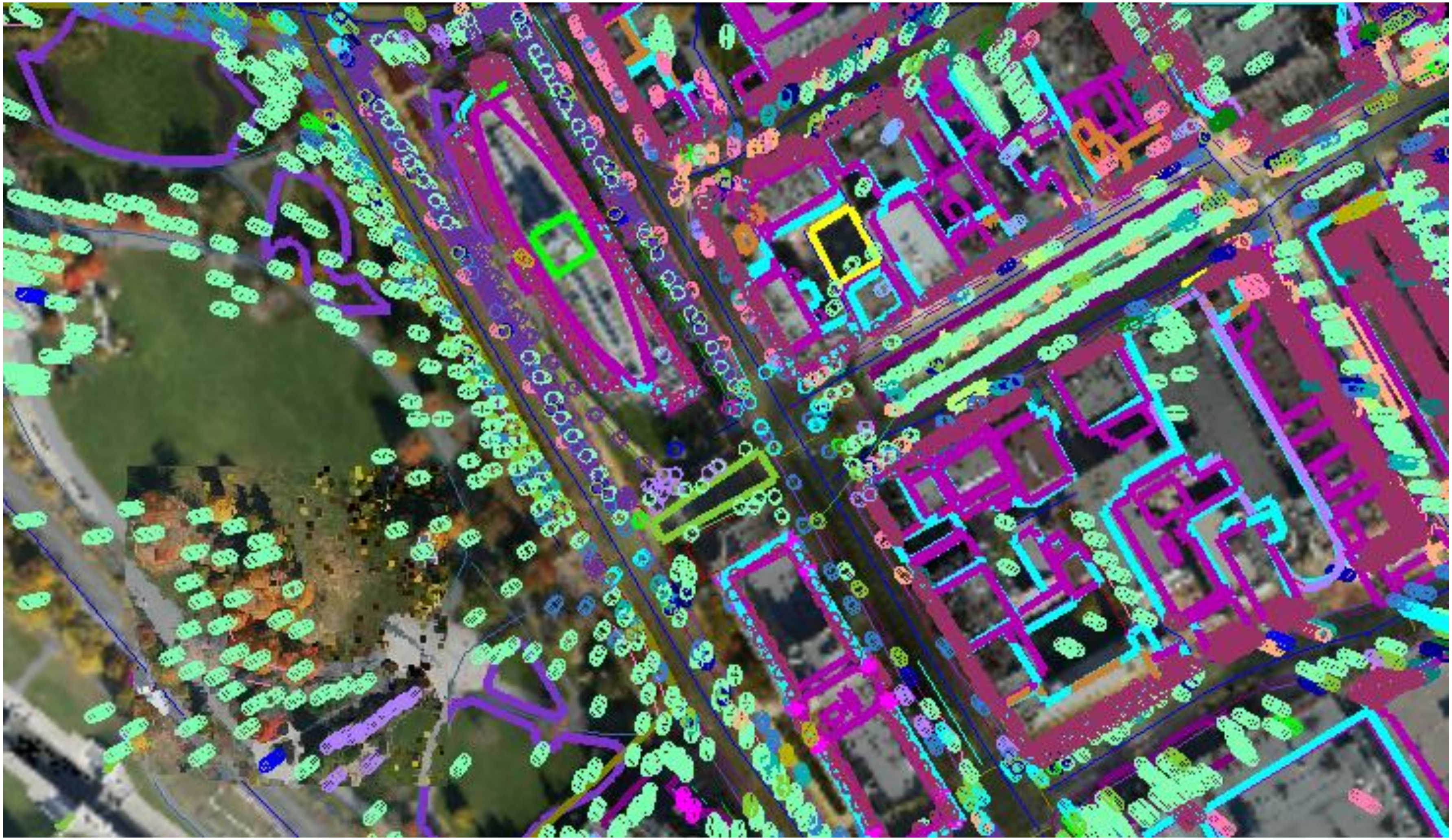
| Class | # in Truth Area | Location Found | ( % ) | Segmentation Precision | Segmentation Recall | # in Test Area | Recognition # Predicted | Recognition Precision | Recognition Recall |
|---|---|---|---|---|---|---|---|---|---|
| Short Post | 338 | 328 | ( 97 ) | 92 | 99 | 116 | 131 | 79 | 91 |
| Car | 238 | 179 | ( 75 ) | 92 | 77 | 112 | 218 | 50 | 62 |
| Lamp Post | 146 | 146 | (100) | 89 | 98 | 98 | 132 | 70 | 86 |
| Sign | 96 | 96 | (100) | 83 | 100 | 60 | 71 | 58 | 65 |
| Light Standard | 58 | 57 | ( 98 ) | 91 | 92 | 37 | 51 | 45 | 62 |
| Traffic Light | 42 | 39 | ( 93 ) | 84 | 86 | 36 | 33 | 52 | 47 |
| Newspaper Box | 37 | 34 | ( 92 ) | 38 | 93 | 29 | 14 | 0 | 0 |
| Tall Post | 34 | 33 | ( 97 ) | 58 | 96 | 10 | 6 | 67 | 40 |
| Fire Hydrant | 20 | 17 | ( 85 ) | 88 | 100 | 14 | 10 | 30 | 21 |
| Trash Can | 19 | 18 | ( 95 ) | 60 | 100 | 15 | 14 | 57 | 40 |
| Parking Meters | 10 | 9 | ( 90 ) | 100 | 100 | 0 | 4 | 0 | 0 |
| Traffic Control Box | 7 | 7 | (100) | 80 | 100 | 5 | 0 | 0 | 0 |
| Recycle Bins | 7 | 7 | (100) | 92 | 100 | 3 | 1 | 0 | 0 |
| Advertising Cylinder | 6 | 6 | (100) | 96 | 100 | 3 | 0 | 0 | 0 |
| Mailing Box | 3 | 3 | (100) | 98 | 100 | 1 | 2 | 0 | 0 |
| "A" - frame | 2 | 2 | (100) | 86 | 100 | 0 | 0 | 0 | 0 |
| All | 1063 | 976 | ( 92 ) | 86 | 93 | 539 | 687 | 58 | 65 |

Accuracy of model's predictions

*[Golovinskiy et al., ICCV 2009]*

Fixing errors requires more manual annotation

# What Manual Annotation Method is Best?



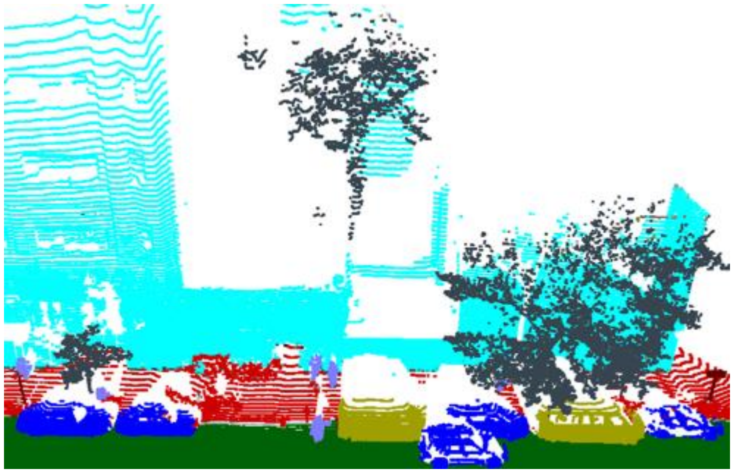
Given a new data set, how label everything perfectly?

# Goal

Interactive system for manual labeling of small objects in LiDAR scans of urban environments

- Handle city-scale LIDAR datasets
- Achieve production-level accuracy (~100%)
- Require minimal user interaction

LIDAR Data



Instance-level Semantic Segmentation

# Outline of Talk

Introduction

Experiences with different interactive labeling systems

1. One-by-one labeling
2. Interactive learning
3. Active learning
4. Group active learning

Summary and conclusion

# Outline of Talk

Introduction

Experiences with different interactive labeling systems

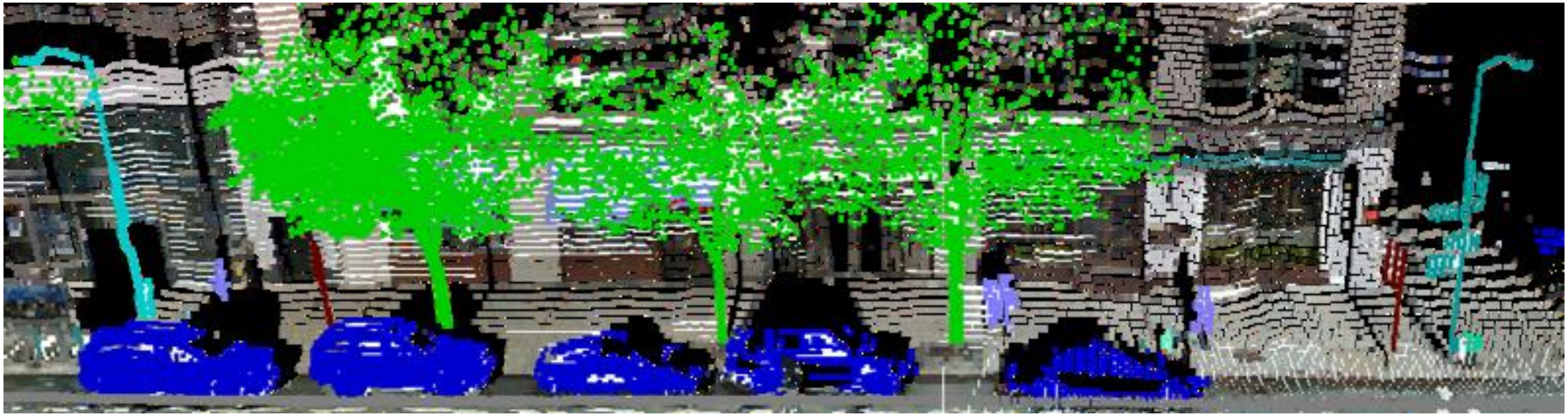
1. One-by-one labeling
2. Interactive learning
3. Active learning
4. Group active learning

Summary and conclusion

# One-by-One Labeling

Approach:

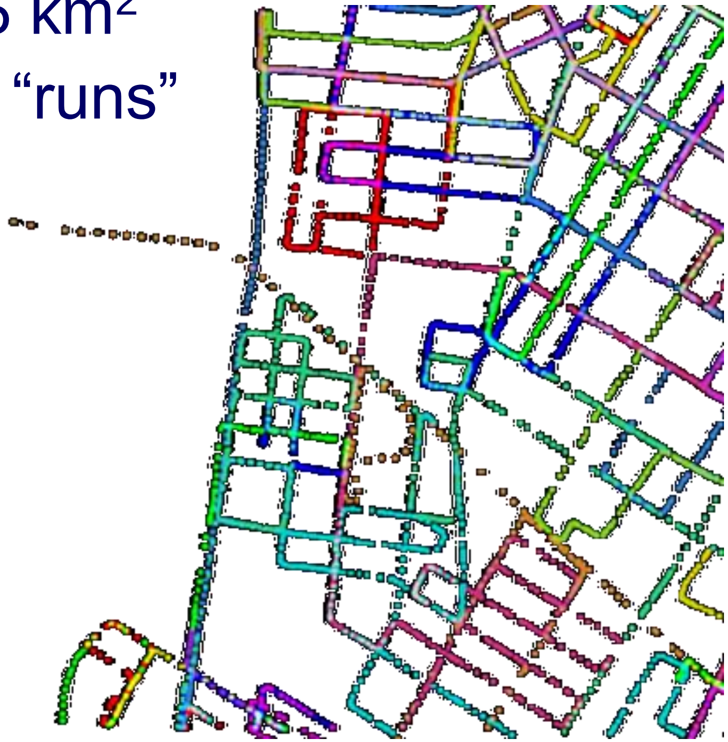
1. Computer provides initial segmentation
2. User finds objects, merges/splits segments, and assigns semantic labels with a keyboard key



# One-by-One Labeling

Data: Manhattan (R5 Google Street View)

- Push-broom LIDAR images from side-facing scanners
- 390M points
- 100 city blocks
- 3.5 km$^2$
- 20 “runs”



Demo

# One-by-One Labeling Result

Result:

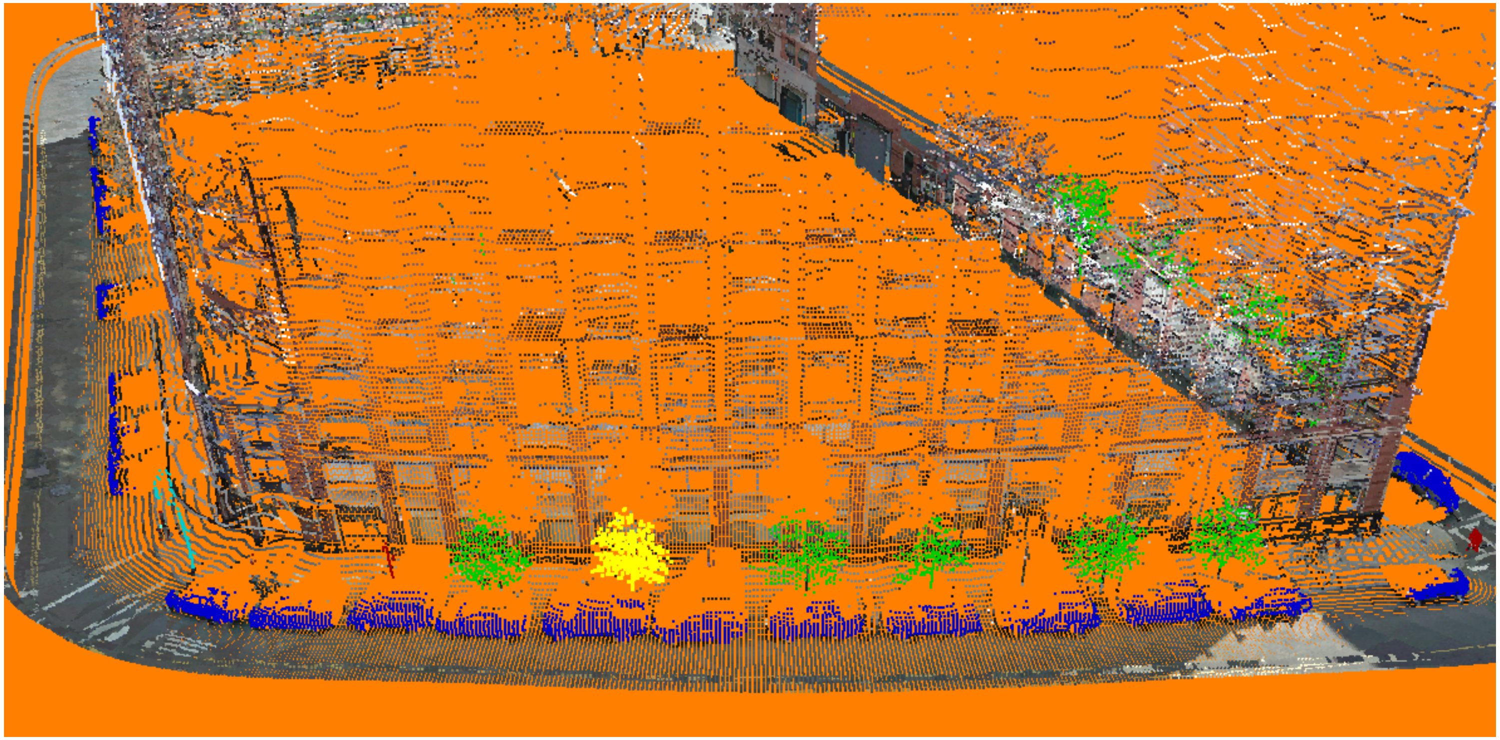
- Manually segmented and labeled 6,533 objects in around 20 hours

| Run | Car | Van | Truck | Tree | Street Light | Traffic Light | Stop Sign |
|---|---|---|---|---|---|---|---|
| NYC 0 | 291 | 36 | 71 | 109 | 54 | 41 | 16 |
| NYC 0, side 2 | 194 | 44 | 17 | 91 | 34 | 28 | 38 |
| NYC 11 | 50 | 2 | 12 | 67 | 21 | 14 | 5 |
| NYC 11, side 2 | 35 | 6 | 5 | 161 | 26 | 28 | 2 |
| NYC 12 | 324 | 52 | 40 | 131 | 61 | 55 | 12 |
| NYC 14 | 82 | 12 | 4 | 107 | 17 | 15 | 6 |

# One-by-One Labeling Conclusion

One-by-one labeling of objects is possible, but tedious

- Domain-specific tools for visualization, camera control, etc. make a big difference on interactive labeling efficiency



# Outline of Talk

Introduction

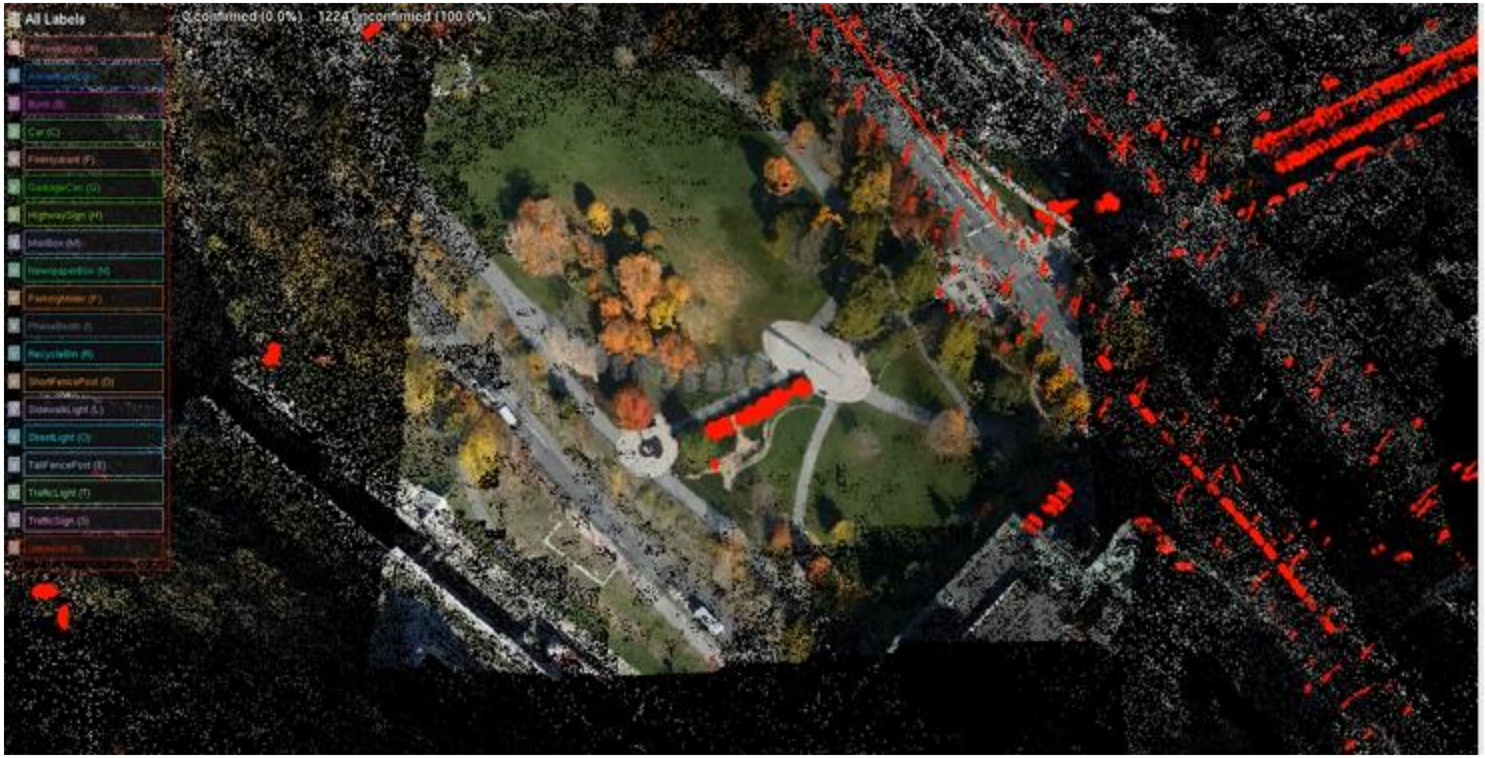
Experiences with different interactive labeling systems

1. One-by-one labeling
2. Interactive learning
3. Active learning
4. Group active learning

Summary and conclusion

# Interactive Learning

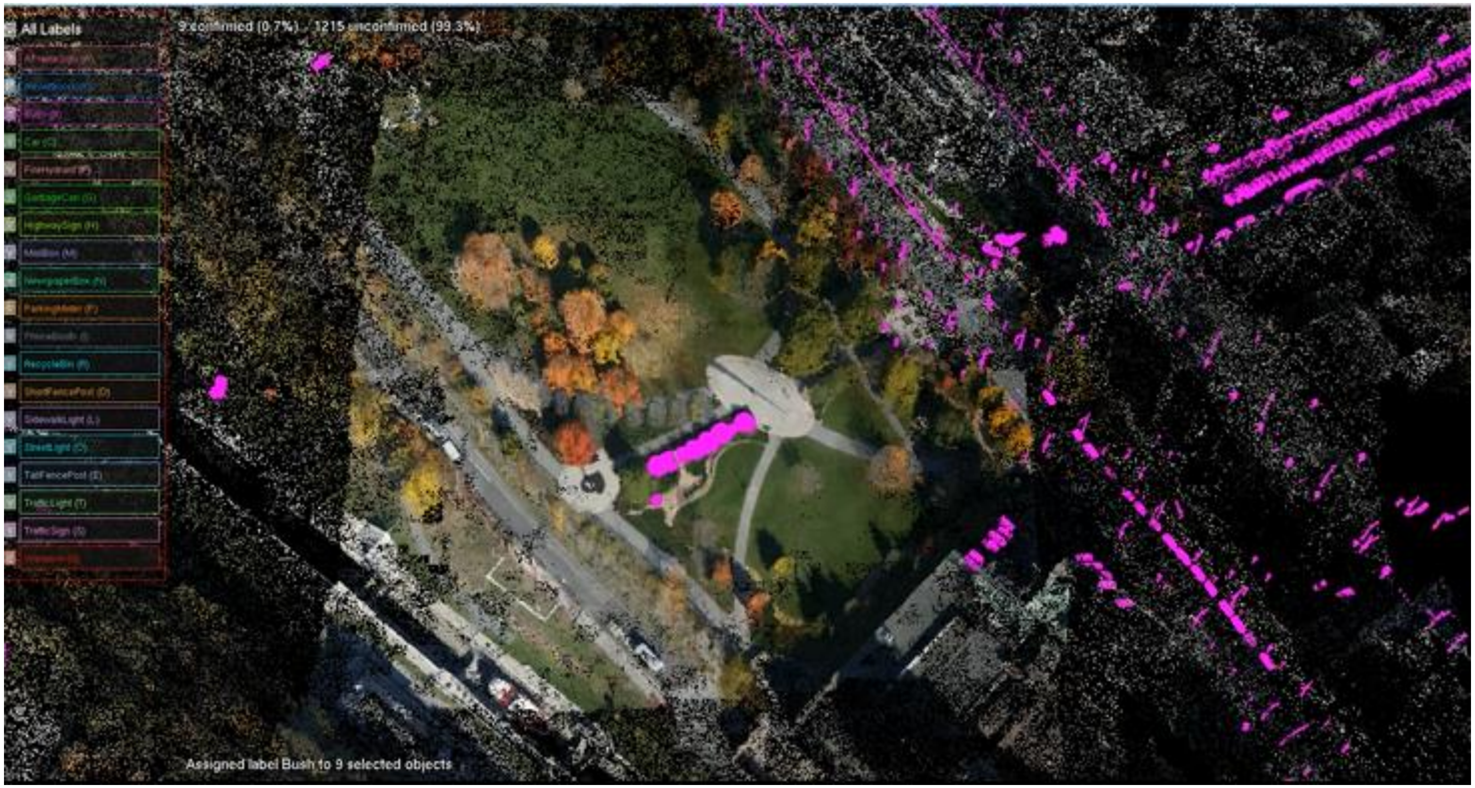
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




Input

# Interactive Learning

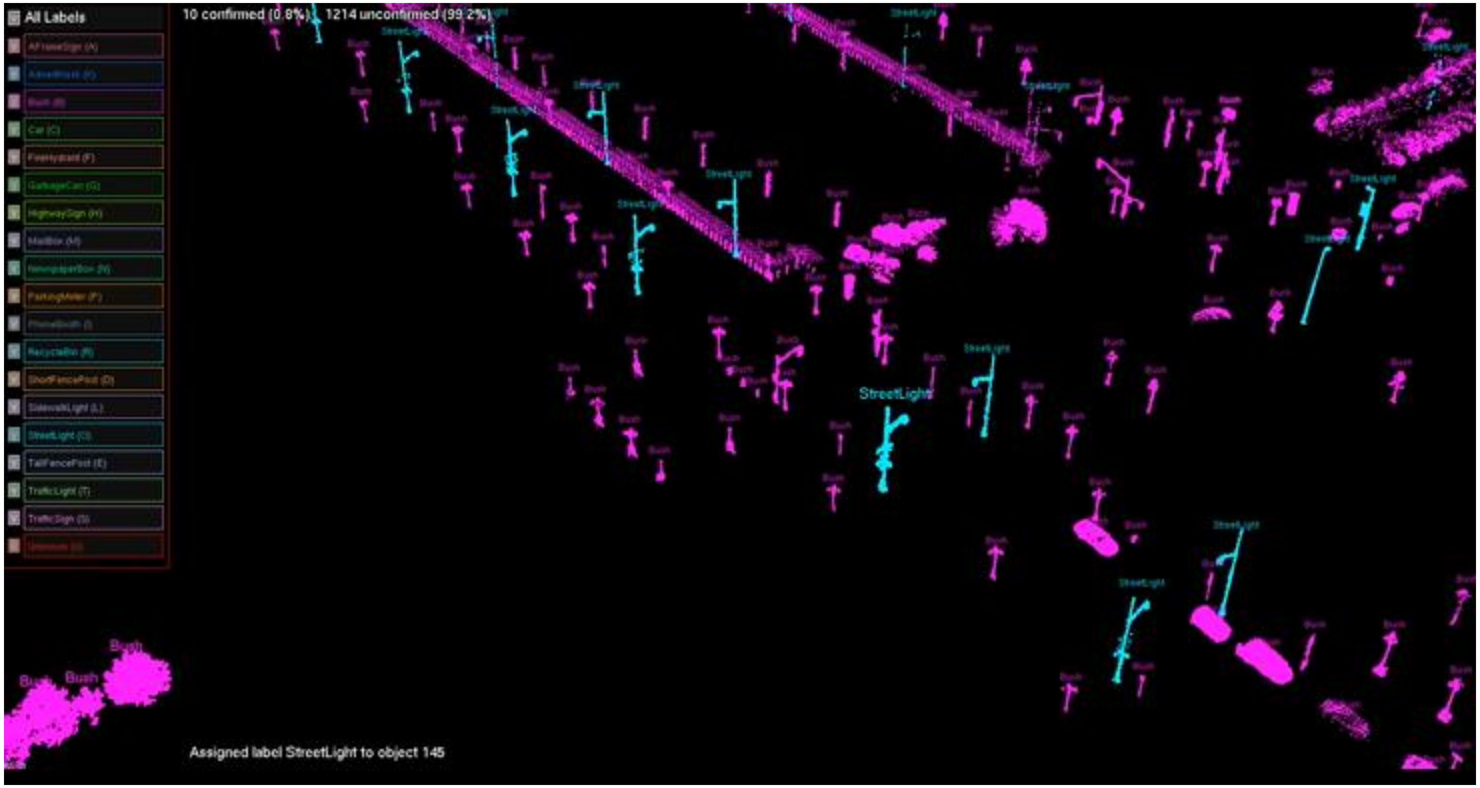
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




After 1 object label

# Interactive Learning

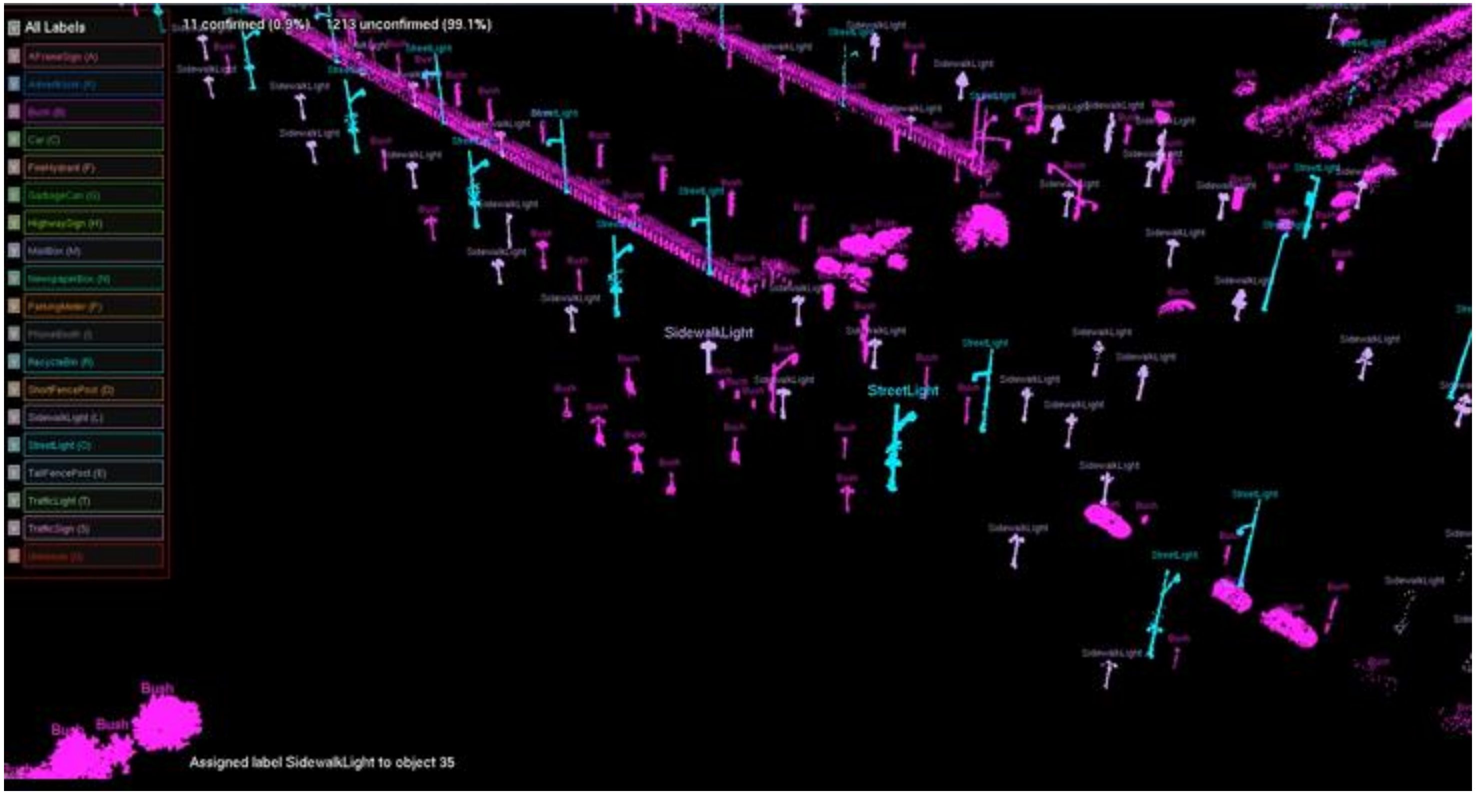
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




After 2 object label

# Interactive Learning

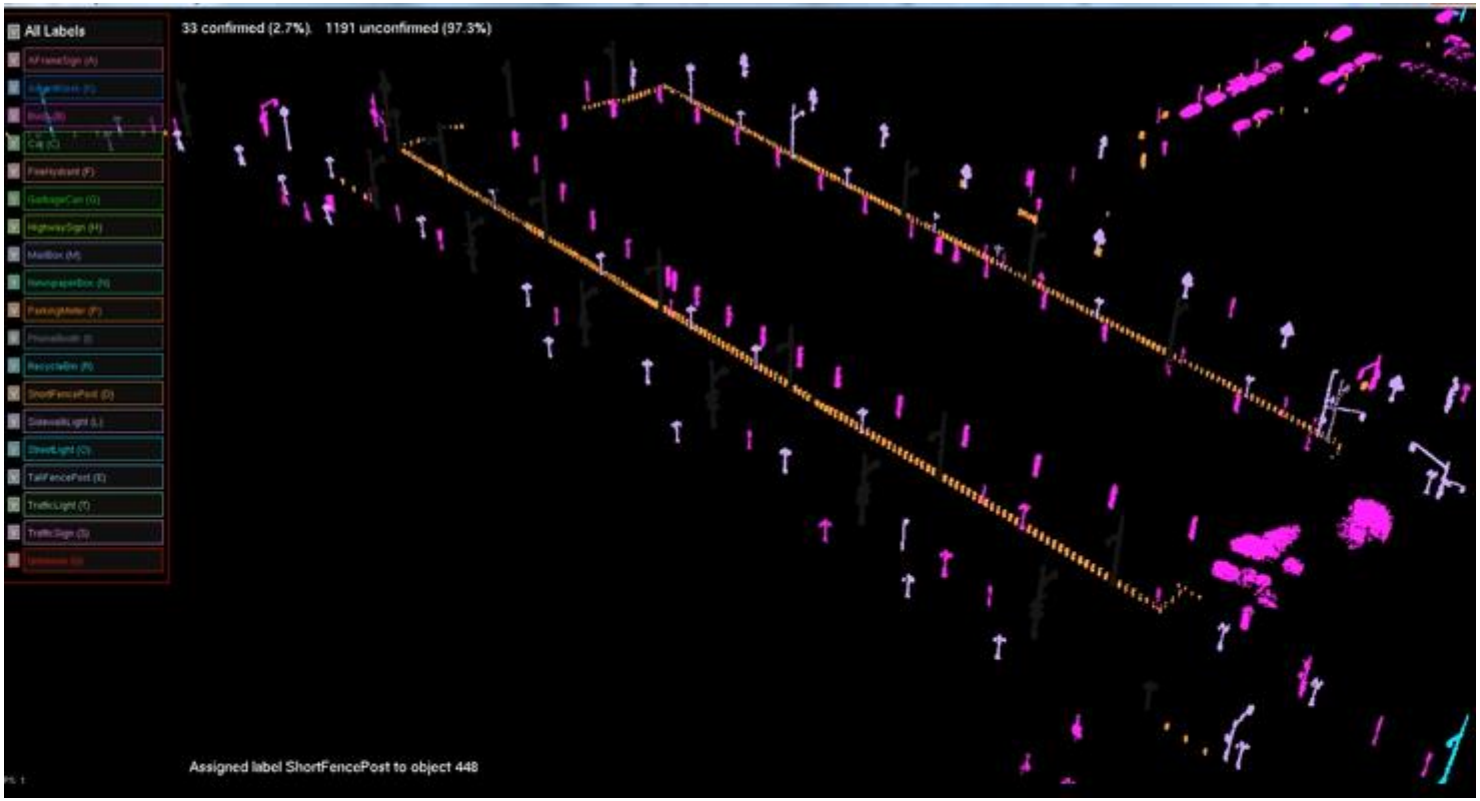
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




After 3 object labels

# Interactive Learning

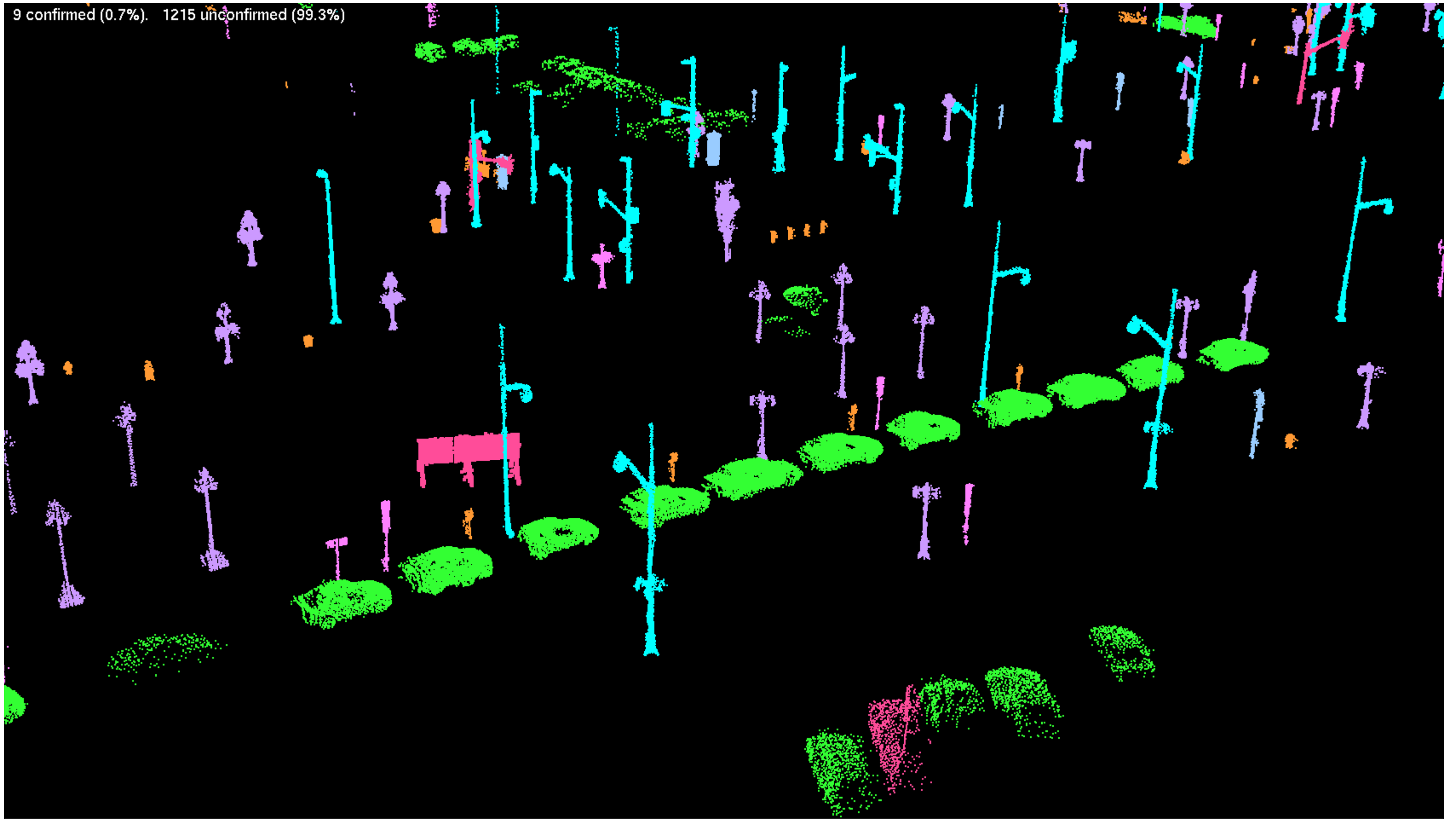
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




After 4 object labels

# Interactive Learning

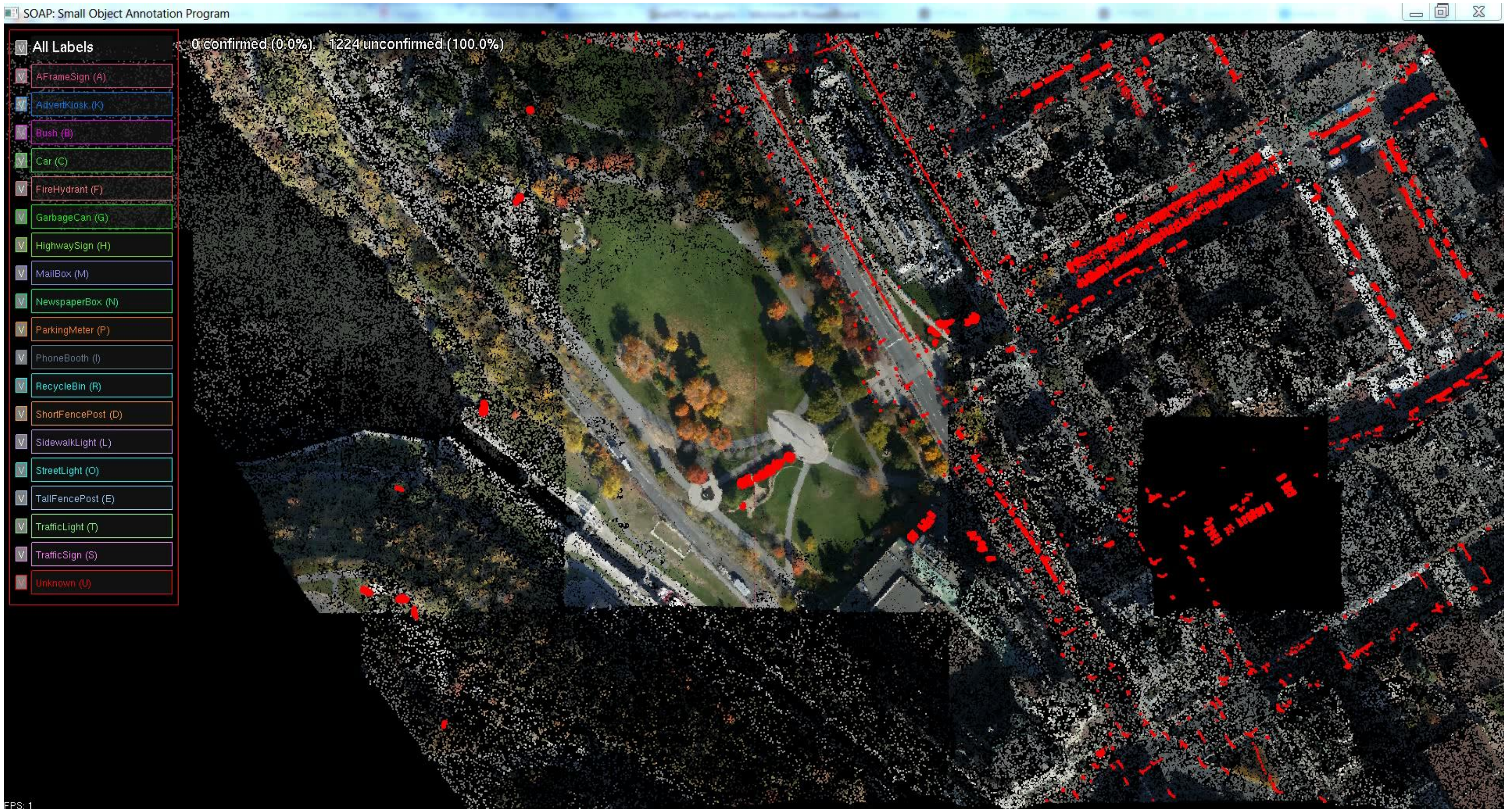
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately




After 9 object labels

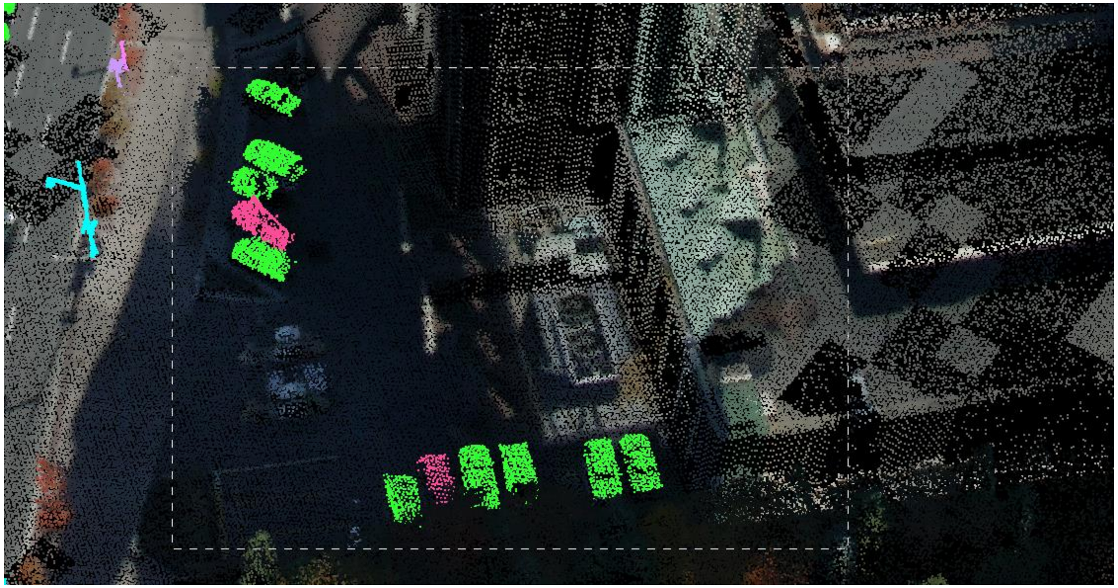
# Interactive Learning

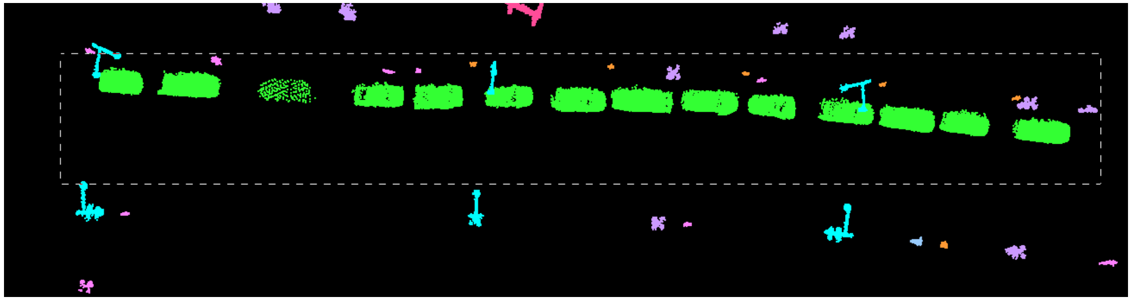
Approach: each time user labels an object, computer updates classifier and re-predicts labels for other objects immediately



# Interactive Learning

Key feature: “class-aware” group selection tools

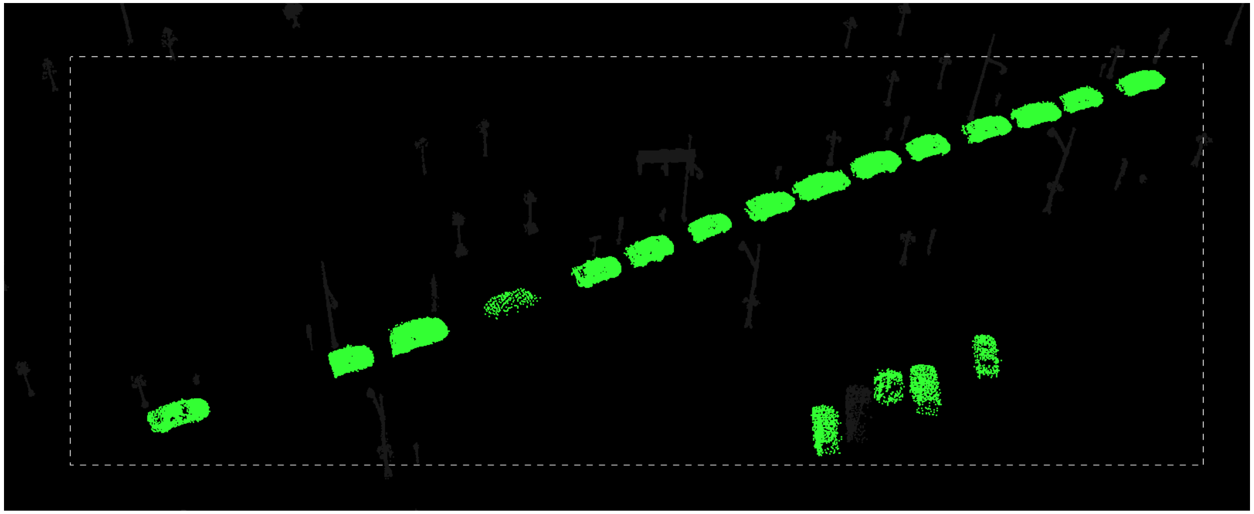




It is difficult to select groups of objects of the same class
with typical bounding box selection tools

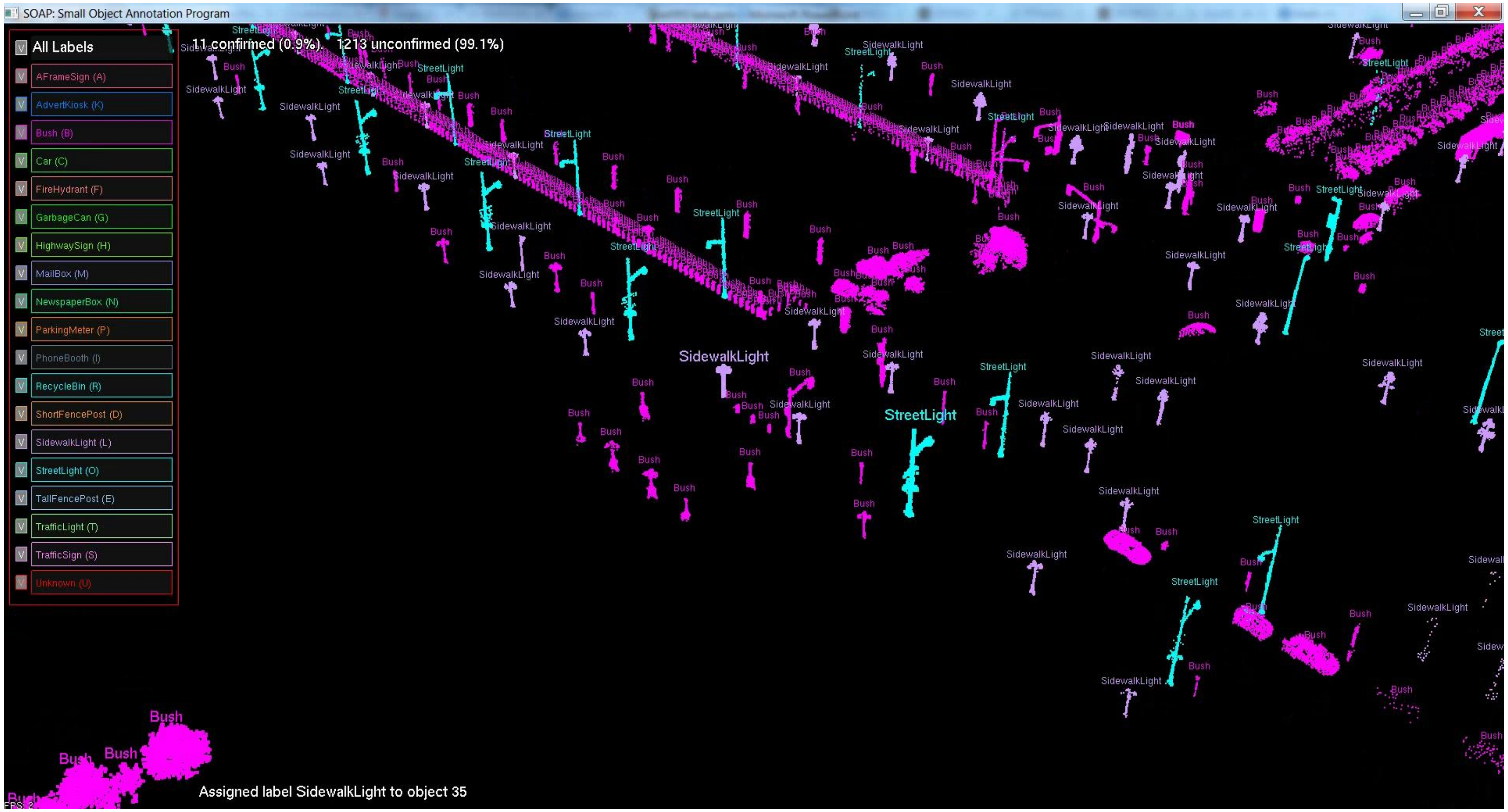
# Interactive Learning

Key feature: “class-aware” group selection tools

# Interactive Learning

Key feature: “class-aware” group selection tools

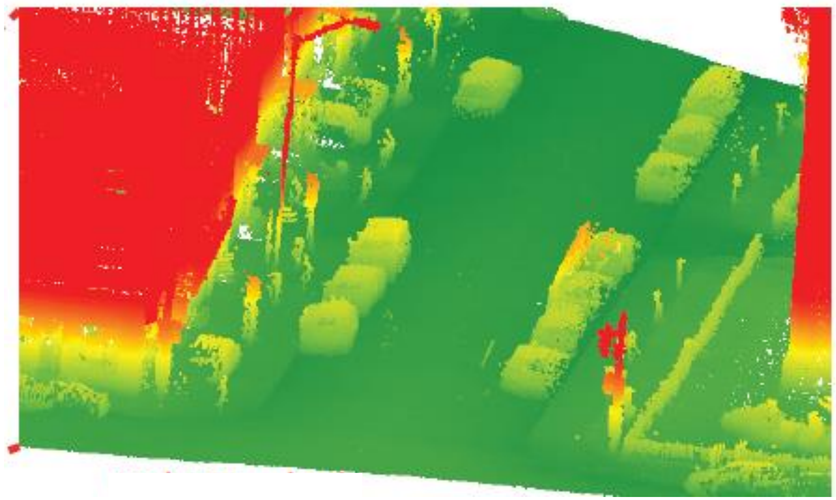


# Interactive Learning Experiment

# Interactive Learning Experiment

Data: Ottawa (Neptec)

- 1 aerial and 4 car-mounted LIDAR scanners
- Point cloud (no viewpoints)
- 6 km$^2$, 954M points



# Interactive Learning Experiment

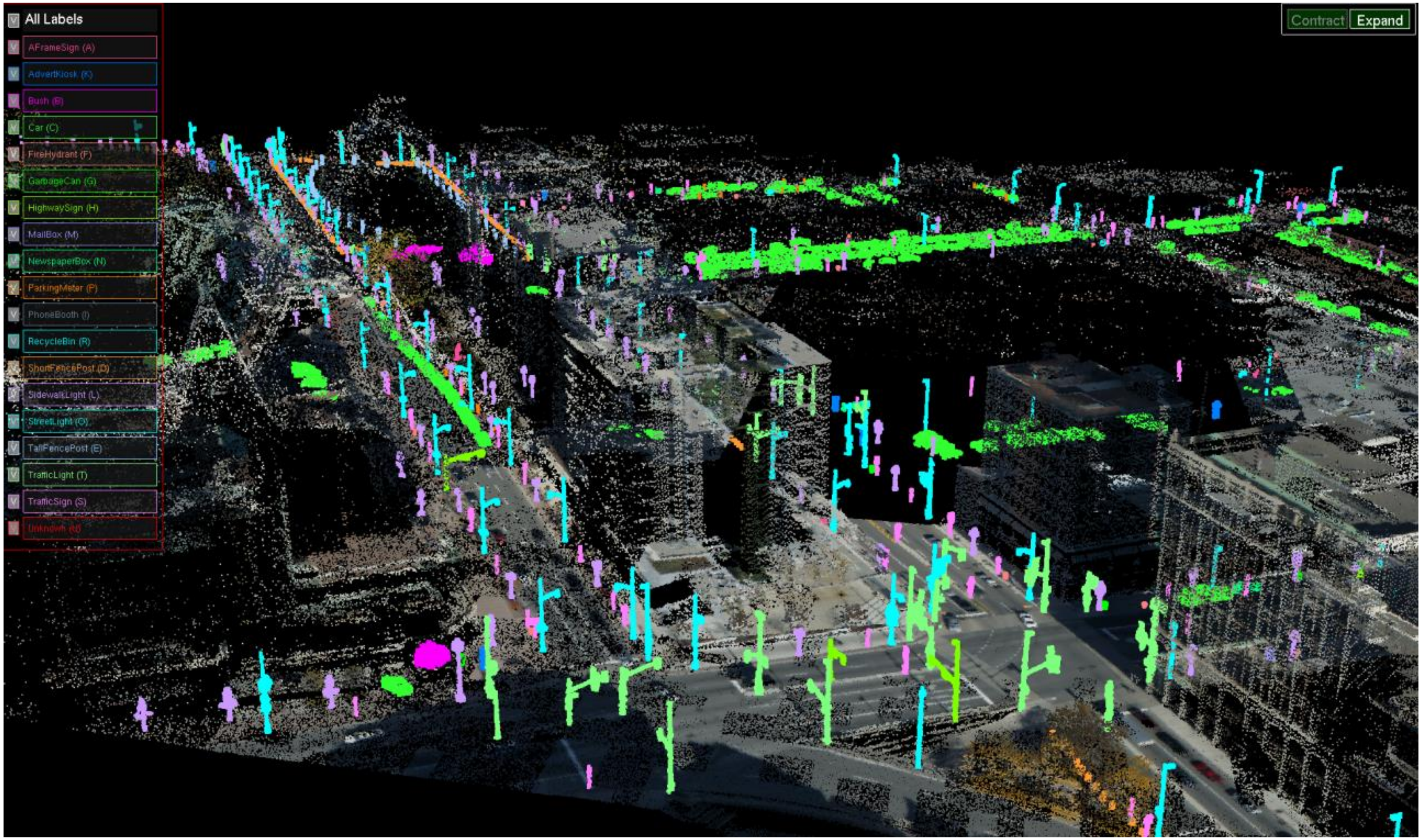
Ground truth:

- $0.3\ km^2$, 100M points
- 1224 manually segmented and labeled objects in 18 classes

  bush, fire hydrant, mailbox,
  newspaper box, parking meter,
  advertising kiosk, garbage can,
  recycle bin, phone booth, traffic sign,
  highway sign, A-frame sign,
  sidewalk light, street light, traffic light,
  short fence post, tall fence post, and car

# Interactive Learning Experiment

Ground truth:



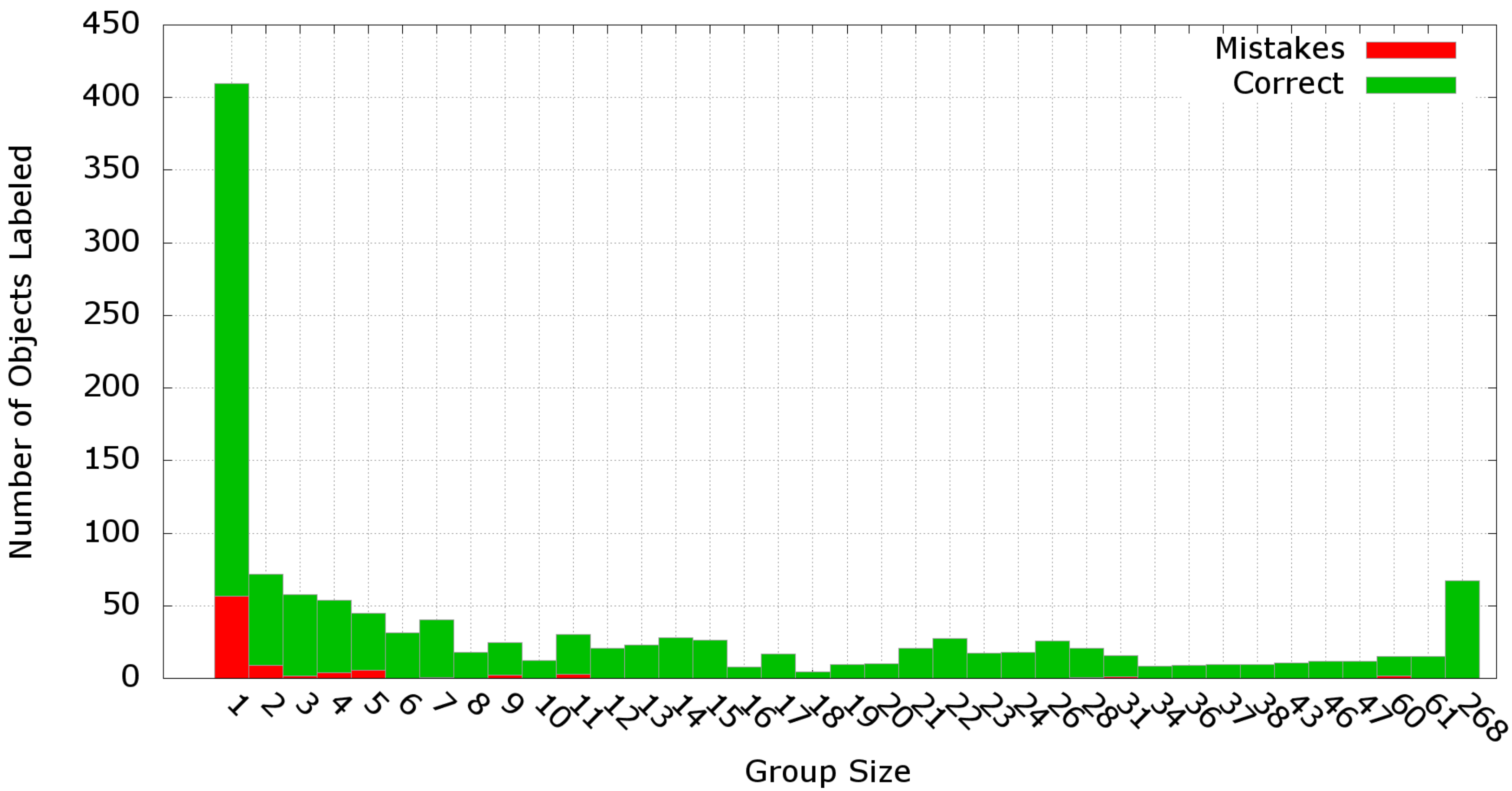
# Interactive Learning Experiment

Protocol:

- Subjects: 4 students (no experience at all)
- Instructions: 5 minutes of instruction
- Training: 15 minutes of practice labeling 163 objects in a different area of city
- Task: “Provide/confirm label for every object with 100% accuracy as quickly as possible”

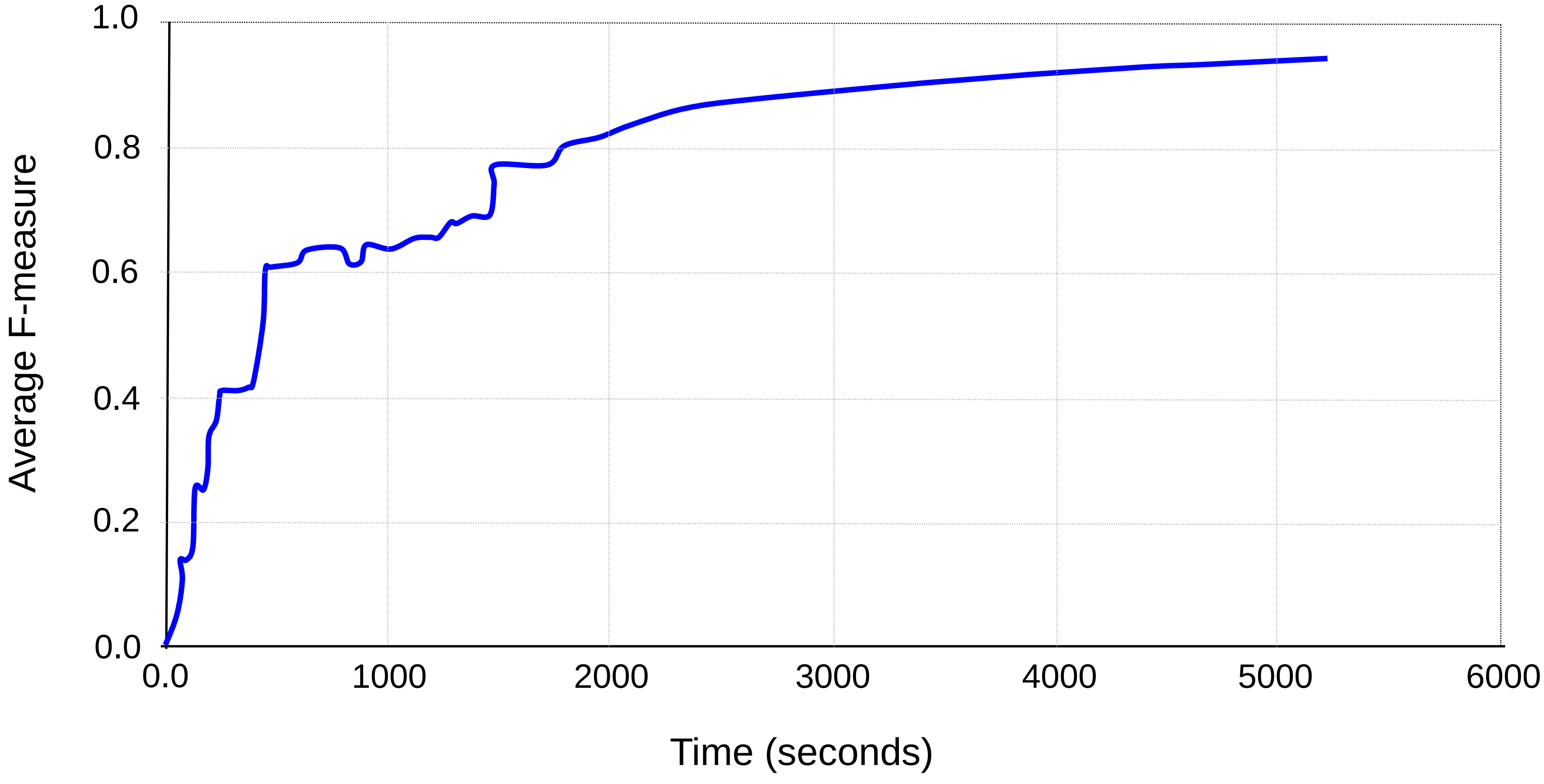
# Interactive Learning Results

Subjects are able to select groups effectively ☺



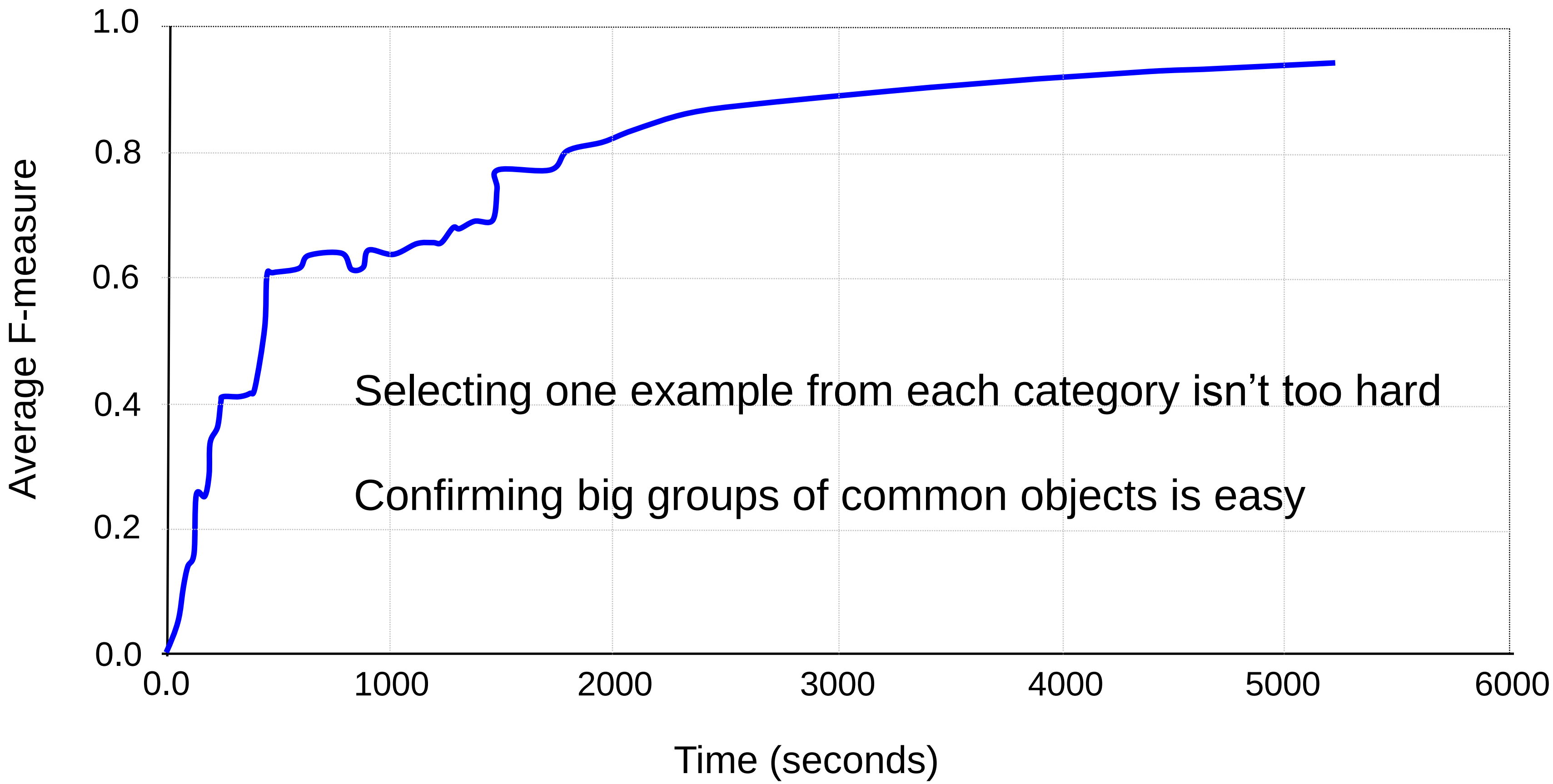
# Interactive Learning Results

Subjects make super-linear progress ☺



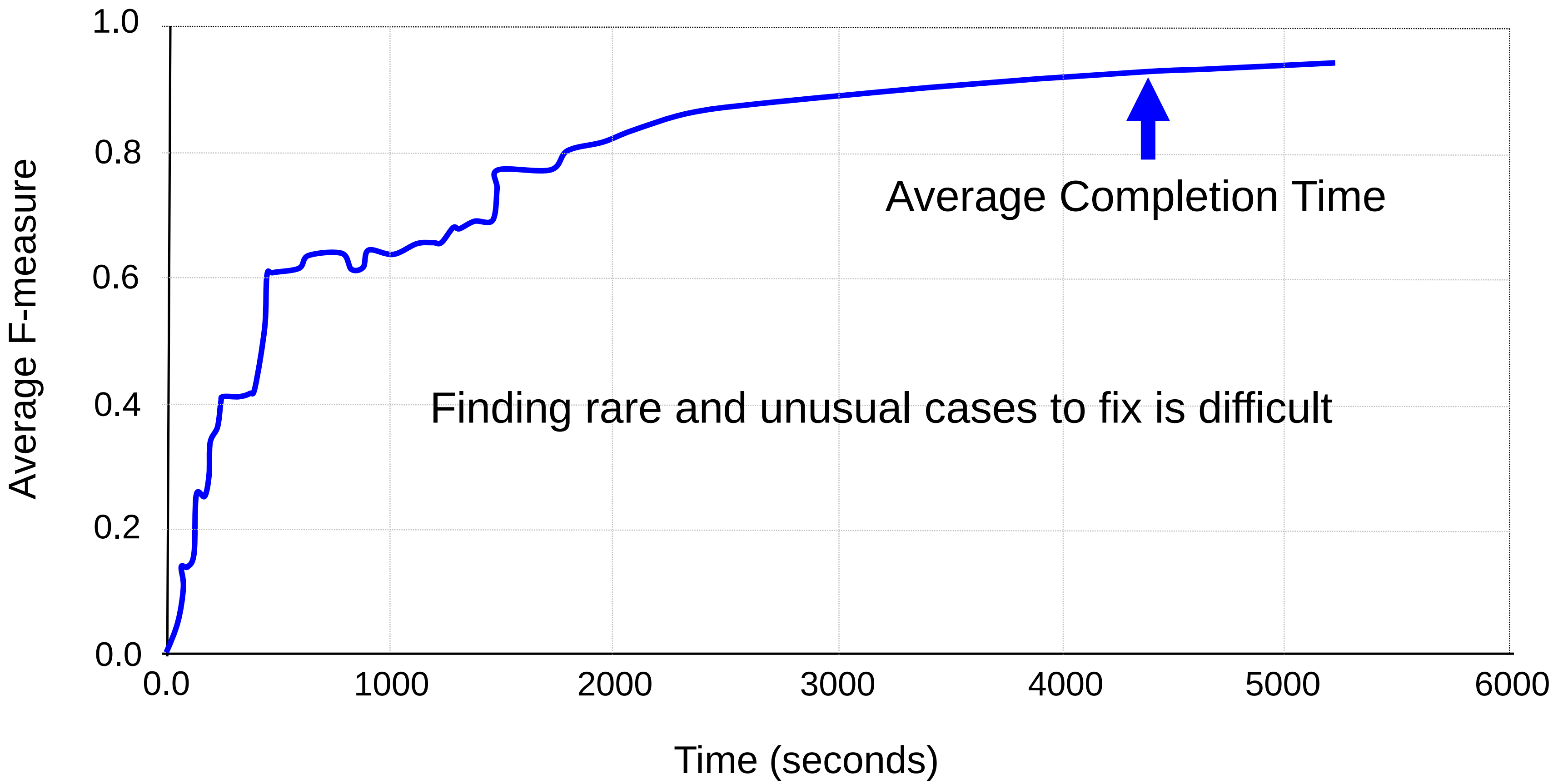
# Interactive Learning Results

Subjects make rapid progress at the beginning ☺



# Interactive Learning Results

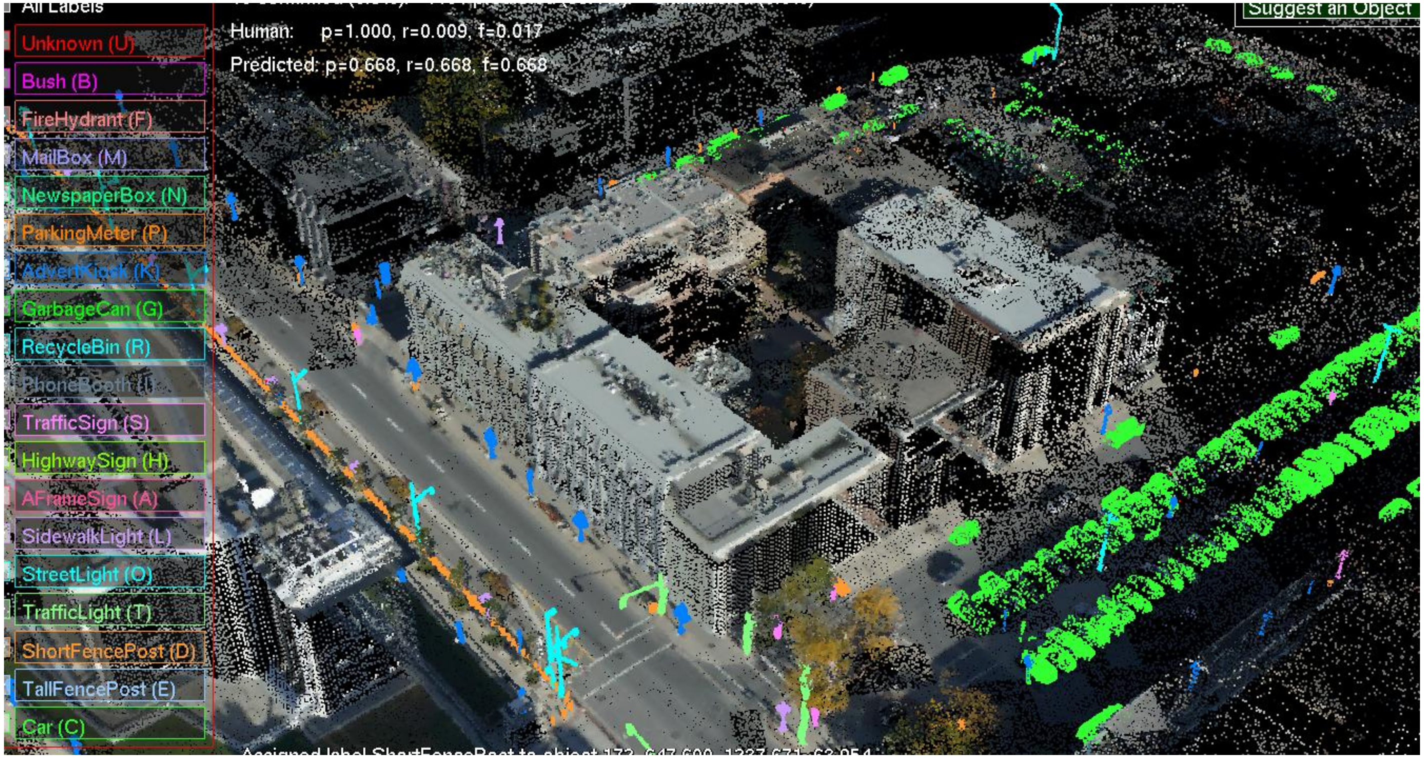
Subjects make slow progress at end ☹



# Interactive Learning Conclusion

Interactive learning is still time-consuming

- Decisions on navigation and selection take time
- Finding objects to label/fix is particularly difficult at the end



# Outline of Talk

Introduction

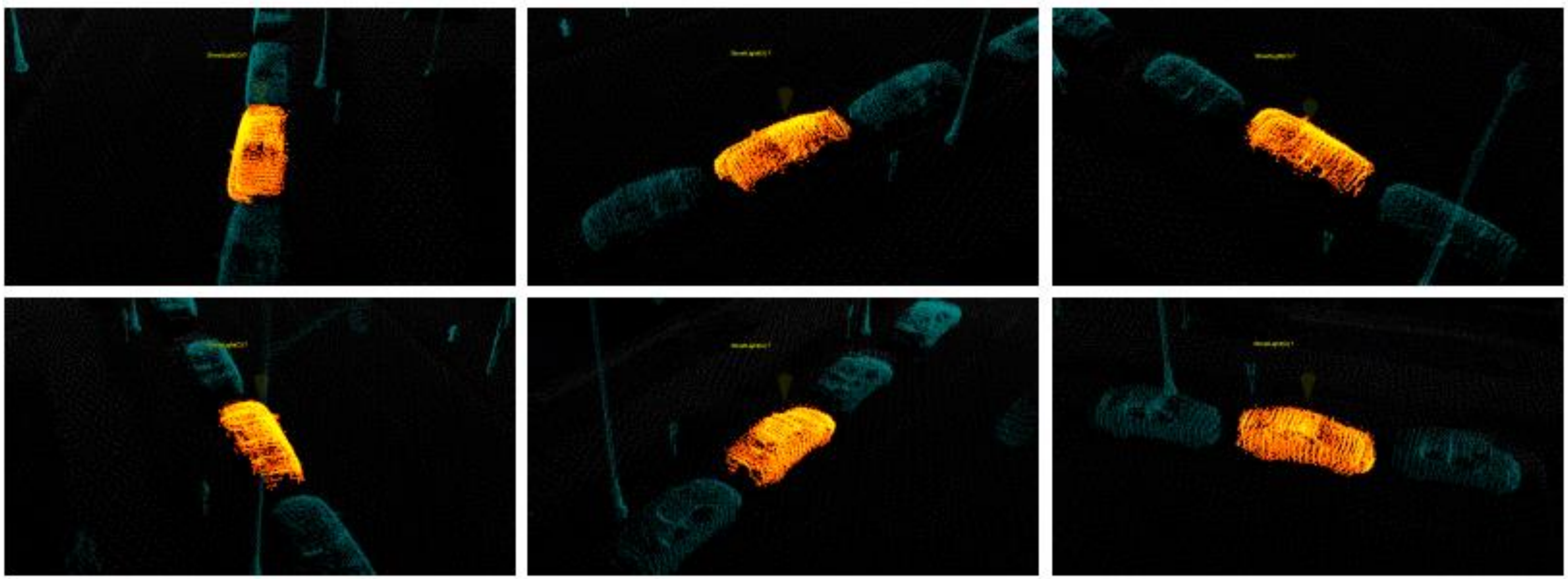
Experiences with different interactive labeling systems

1. One-by-one labeling
2. Interactive learning
3. Active learning
4. Group active learning

Summary and conclusion

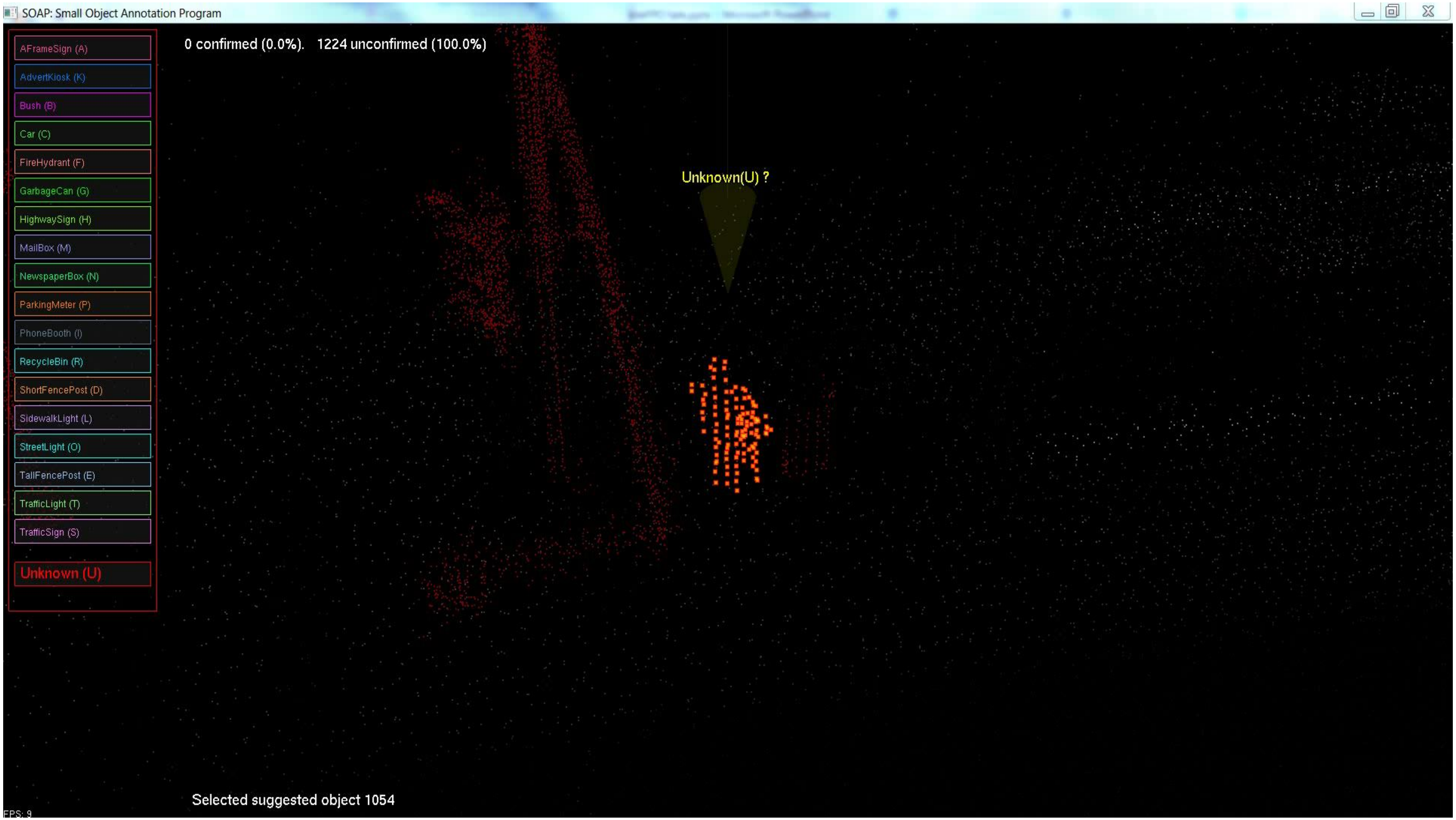
# Active Learning

Approach: computer selects next object to label, controls camera and highlighting, and asks user only to provide label



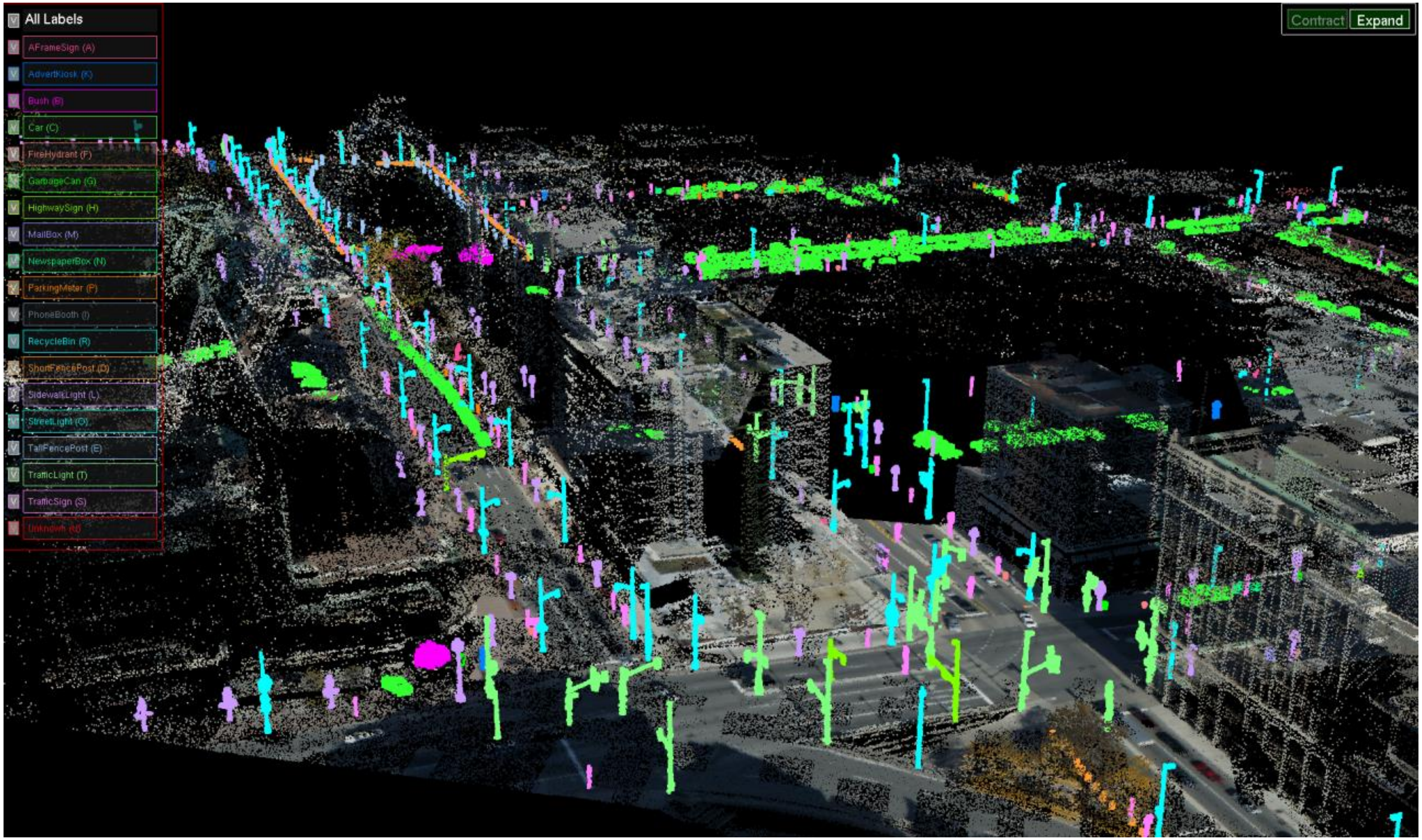
Rotating camera view around selected object to label

# Active Learning



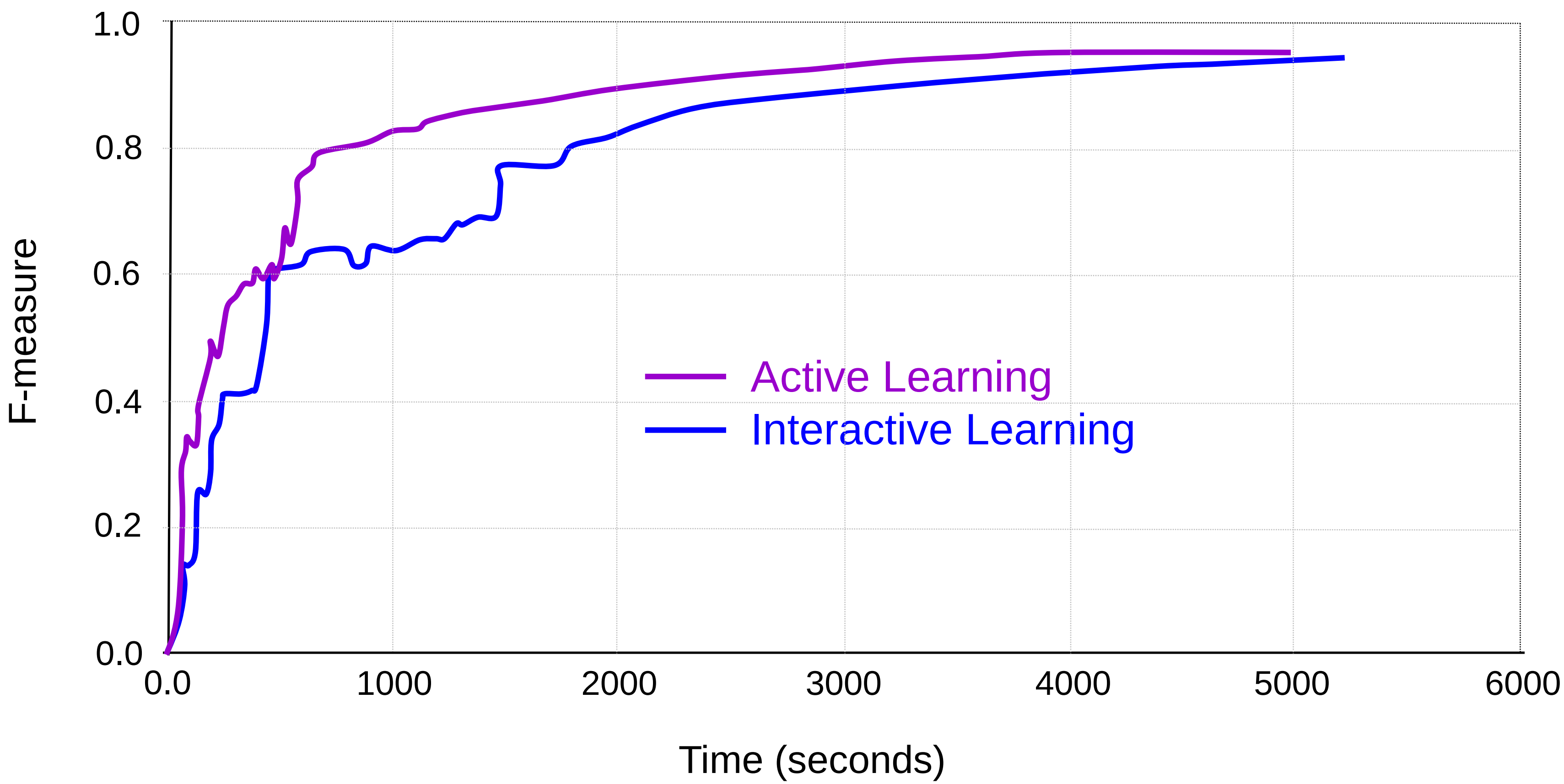
# Active Learning Experiment

Same protocol as before …



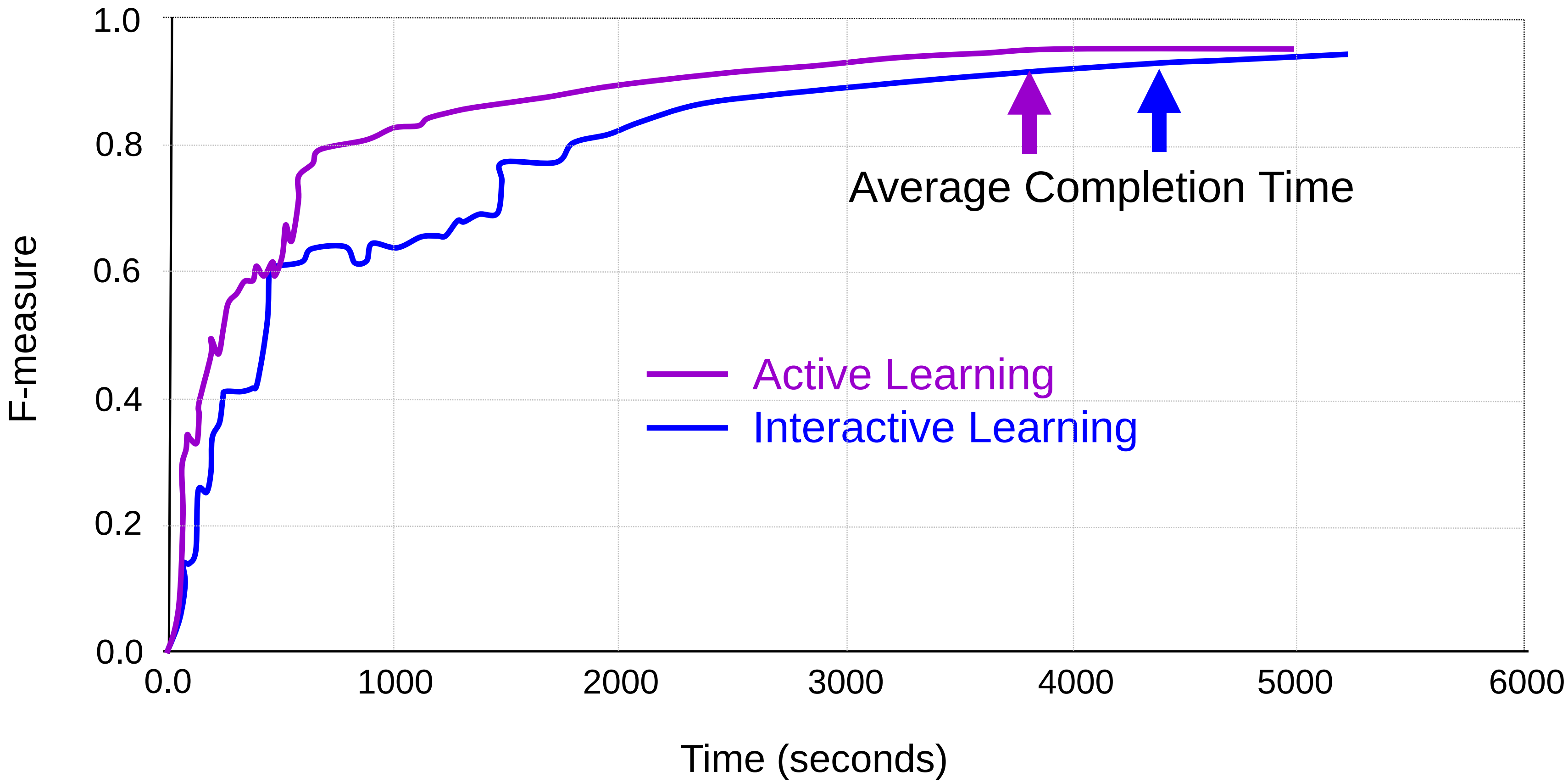
# Active Learning Results

Subjects make faster progress with active learning ☺



# Active Learning Results

Overall time to complete task is still pretty slow ☹



# Active Learning Conclusions

People provide labels more quickly

- Don’t have to worry about camera control or visualization parameters

However, progress is slow because each label is for only one object

- Have to label or confirm every object

# Outline of Talk

Introduction

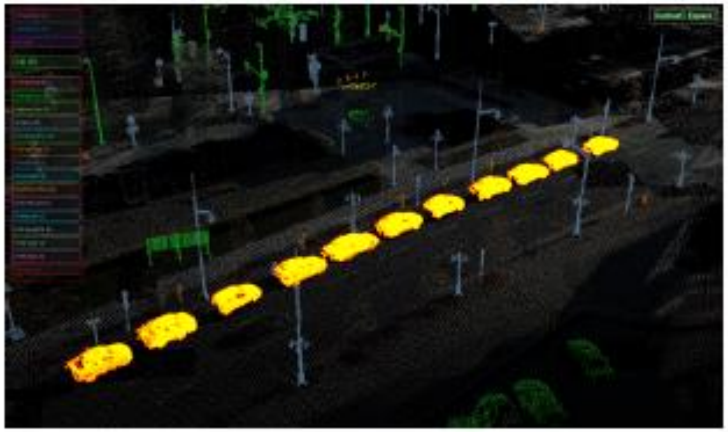
Experiences with different interactive labeling systems

1. One-by-one labeling
2. Interactive learning
3. Active learning
4. Group active learning

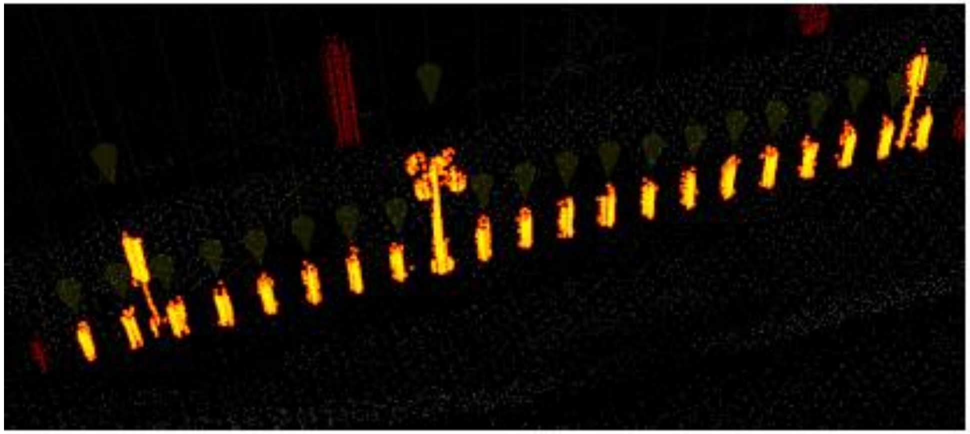
Summary and conclusion

# Group Active Learning

Approach: computer selects group of objects to label, controls camera and highlighting, and asks user to provide label or contract group



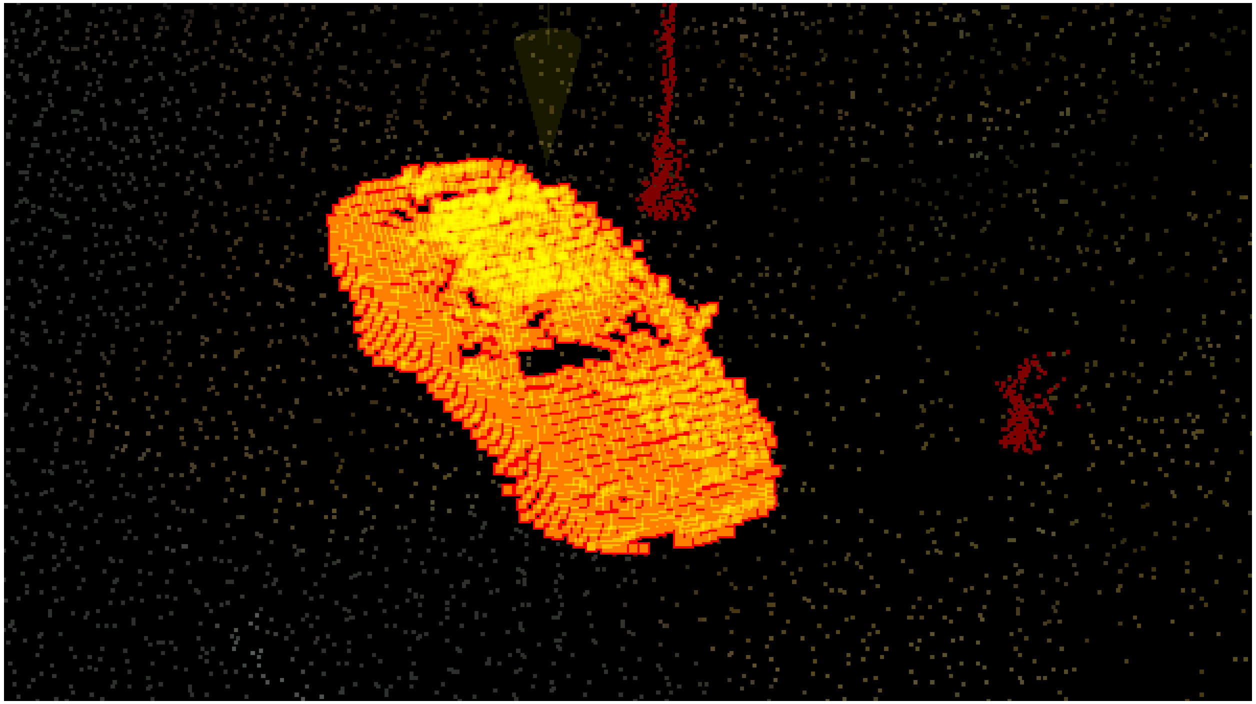
"Car"



"Please contract group"

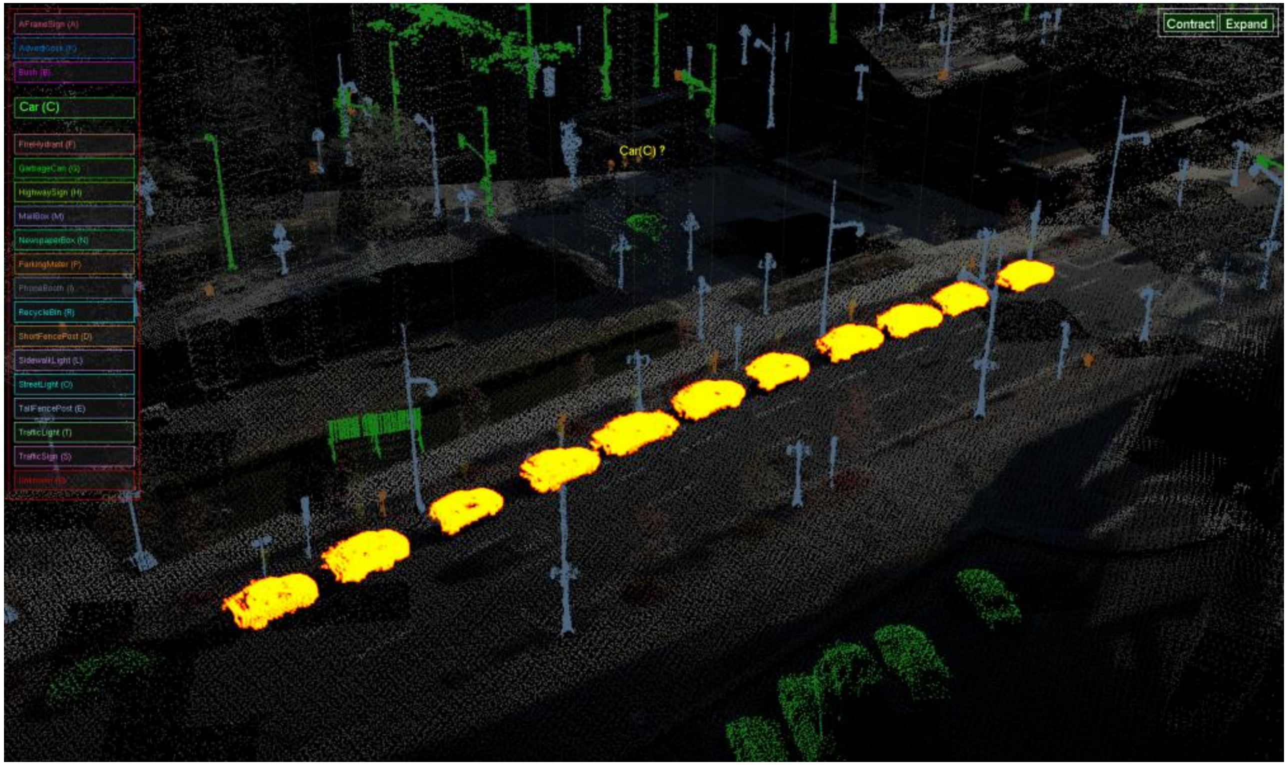
# Motivation for Group Active Learning

How fast can you recognize/label this object?



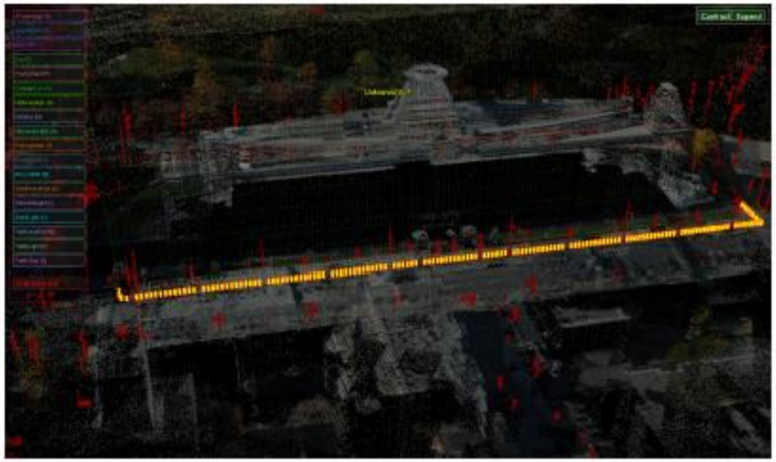
# Motivation for Group Active Learning

How about these?

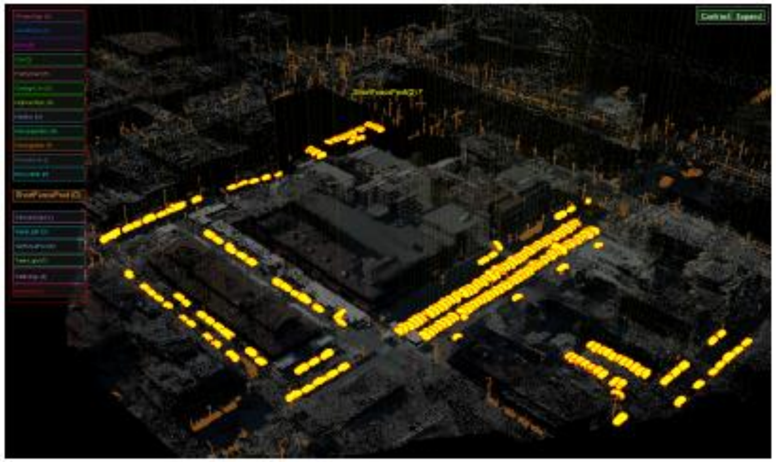


# Motivation for Group Active Learning

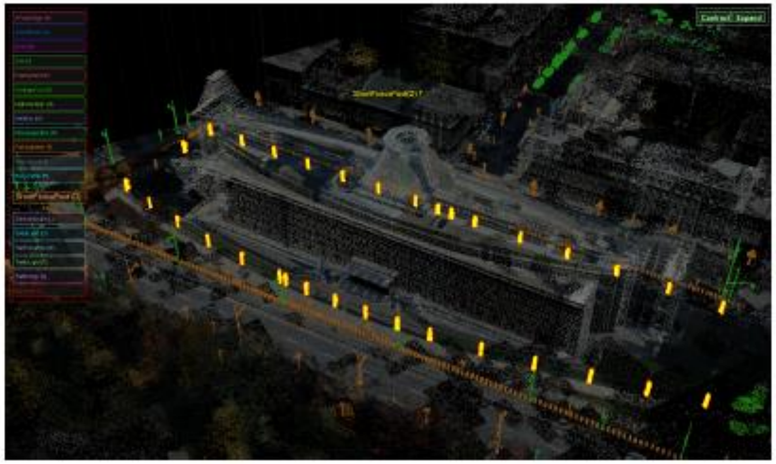
People are fast at recognizing groups of objects



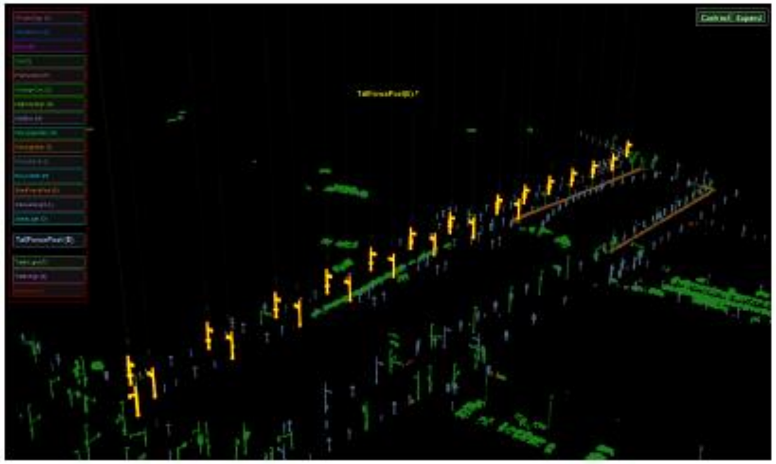
(a) 149 short posts



(b) 210 cars



(c) 33 tall fence posts



(d) 23 street lights

# Motivation for Group Active Learning

Studies from perceptual psychology show …

People grasp gyst of images in 100ms [Rensink et al. 2000, Sanocki et al. 1997]

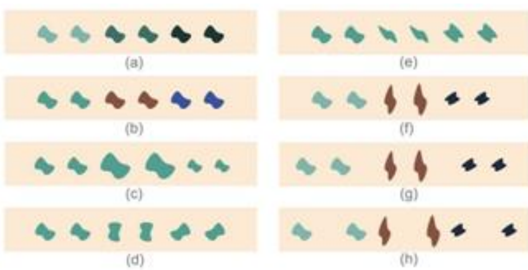
- Can answer specific questions about gyst
  [Delorme et al. 2002, Thorpe et al. 1996]
- Even when distracted
  [Li et al. 2002]

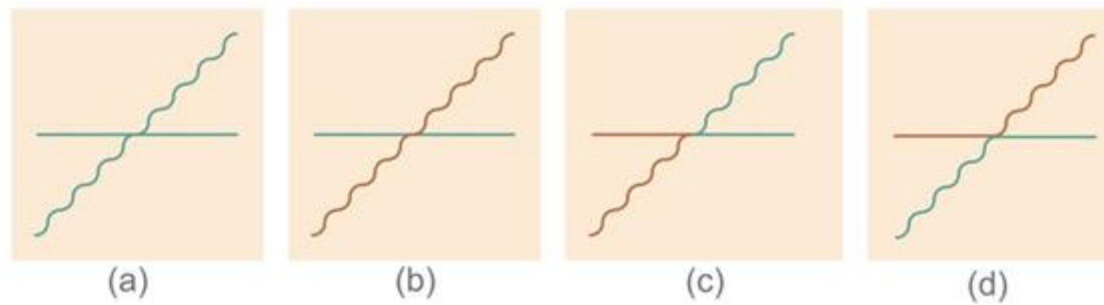
People understand images of groups
with regular patterns of similar items [Koffka 1922]

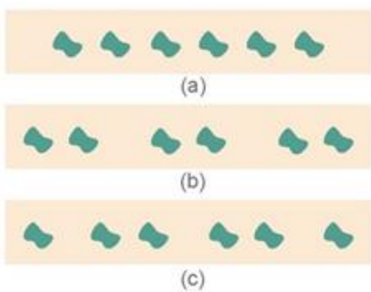
- Maintain only summary representations about groups
  [Ariely 2001]
- Do it rapidly and robustly
  [Chong et al. 2003, Chong et al. 2005, Haberman 2010]

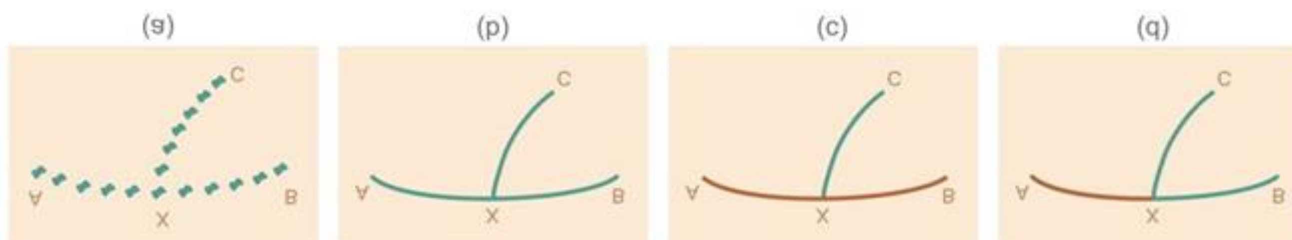
# Motivation for Group Active Learning

Gestault rules for visual grouping suggest which patterns enable rapid recognition of shapes



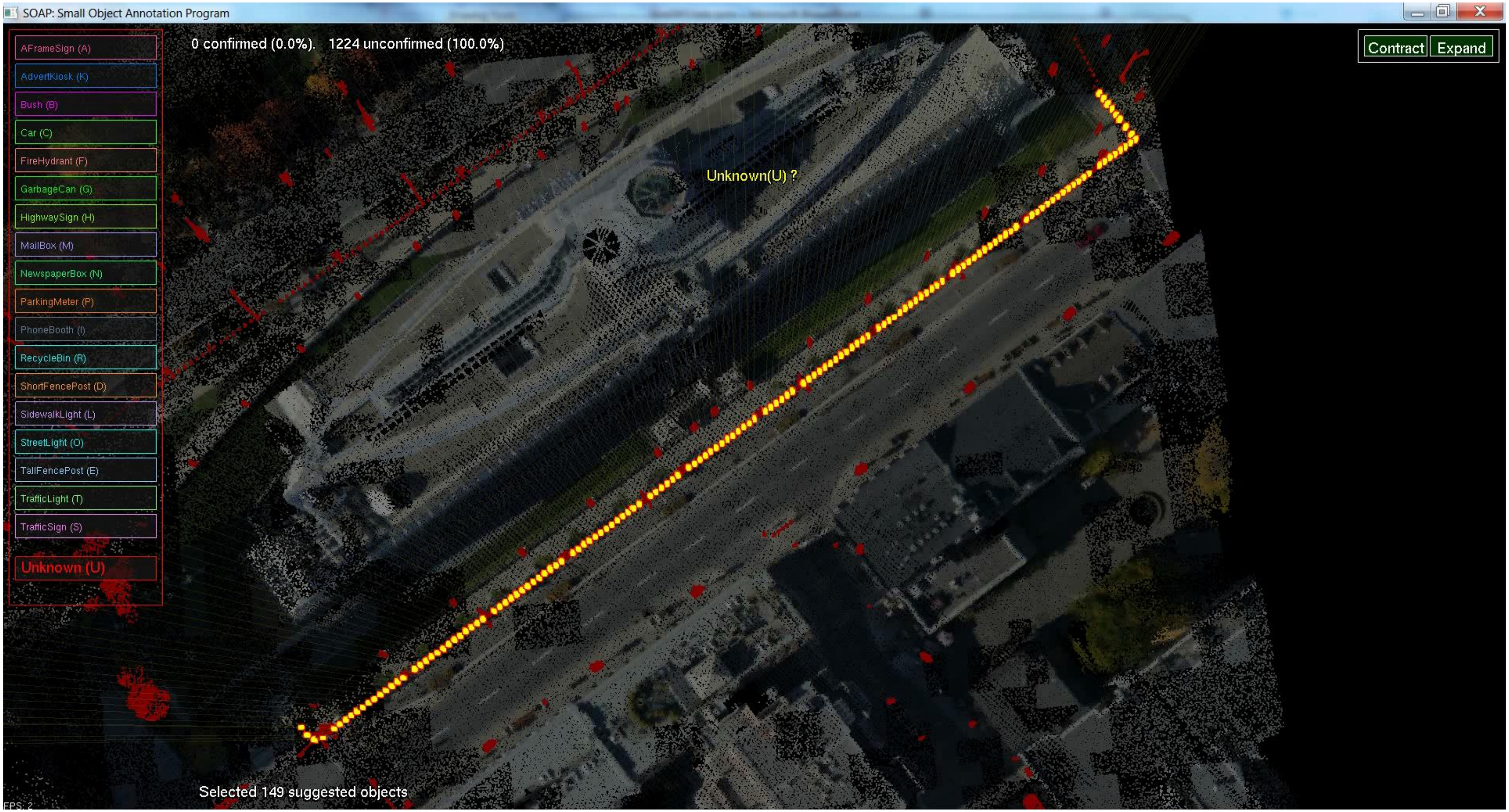




[“Laws of Seeing”. Metzger, 1936]
http://www.scholarpedia.org/article/Gestalt_principles

# Group Active Learning Interface



# Group Active Learning

Challenges: choosing good groups to show user

1. Model the benefit of showing a group
2. Provide real-time algorithm to construct next group

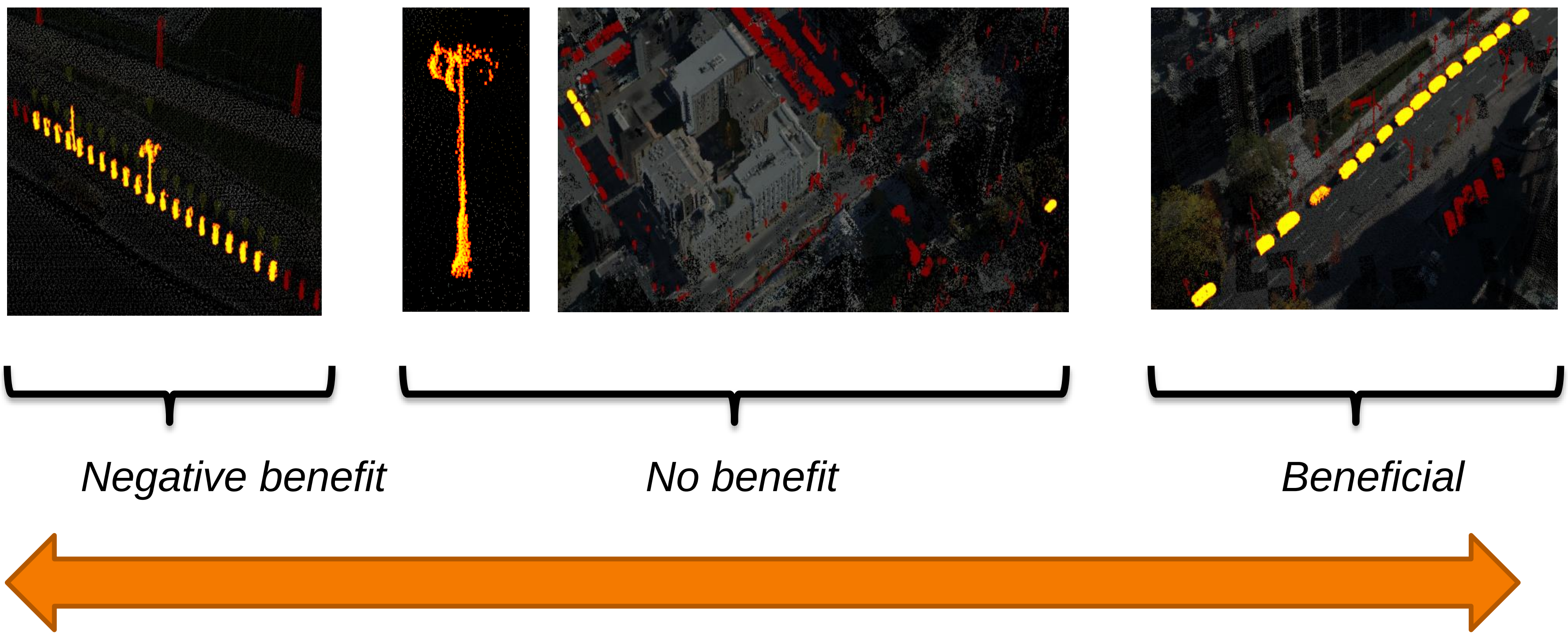
# Group Active Learning

Challenges: choosing good groups to show user

1. Model the benefit of showing a group
2. Provide real-time algorithm to construct next group

# Group Active Learning

Benefit of a group: expected time savings if group is shown to user (compared to 1-by-1 labeling)



# Group Active Learning

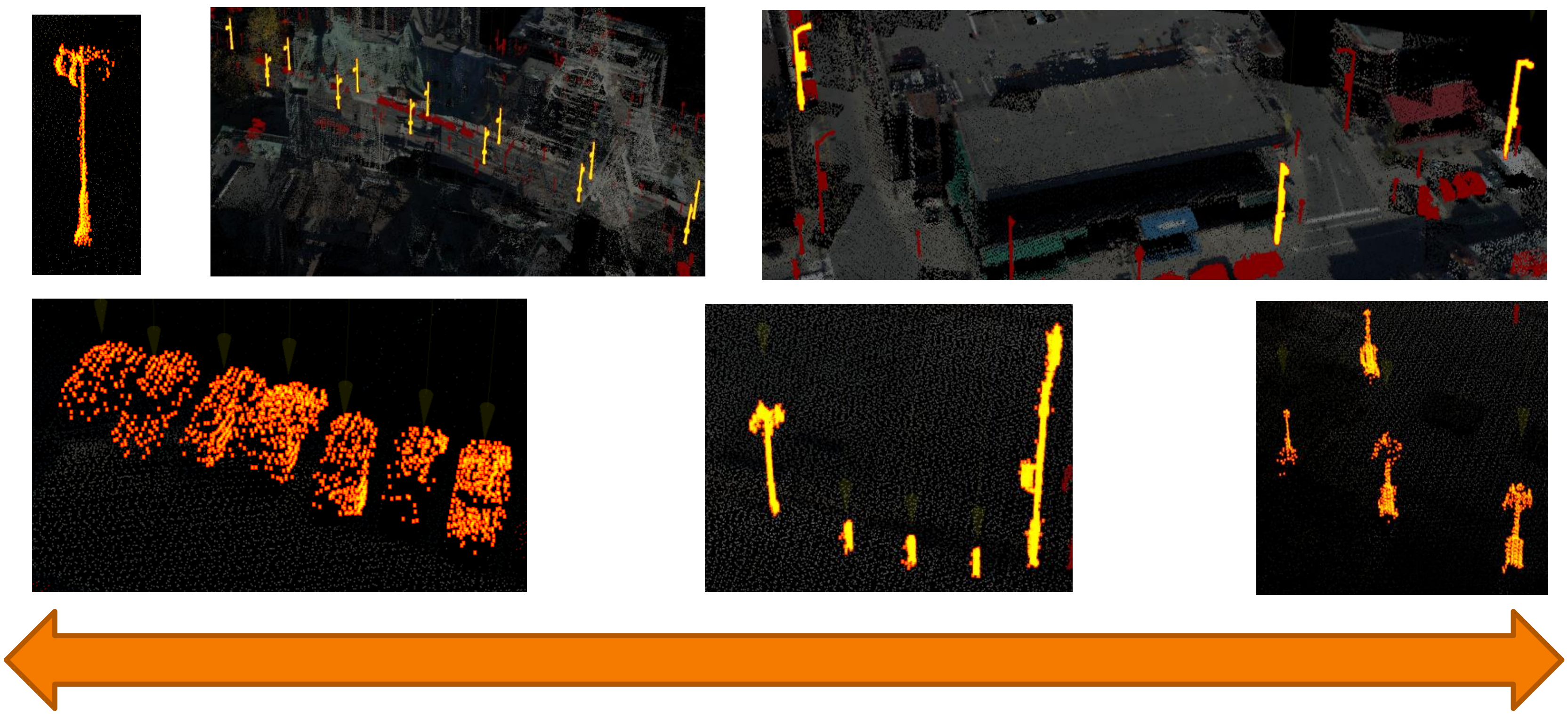
Benefit of a group:

$$Benefit = p_{Label} \cdot T_{1-by-1} - T_{group}$$

# Group Active Learning

Benefit of a group:

$$Benefit = p_{Label} \cdot T_{1\text{–}by\text{–}1} - \boxed{T_{group}}$$

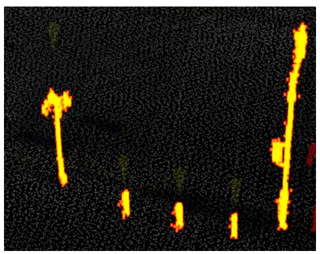
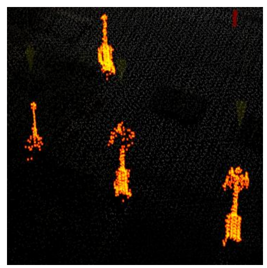
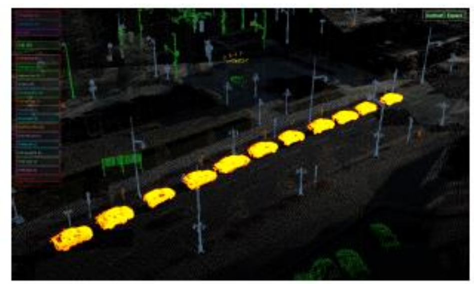
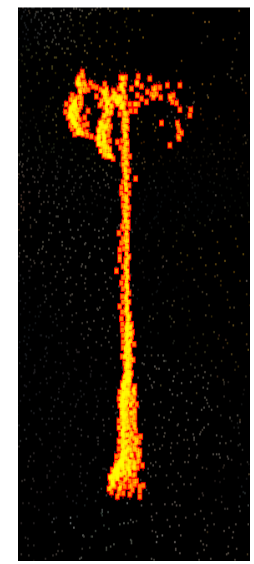
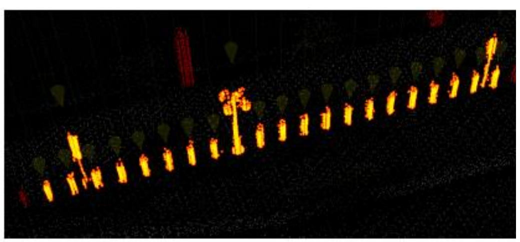
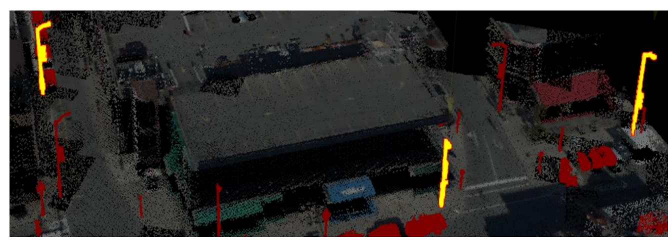


*Lower* Time to recognize and label group *Higher*

# Group Active Learning

Benefit of a group:

$$Benefit = \boxed{p_{Label}} \cdot T_{1-by-1} - T_{group}$$

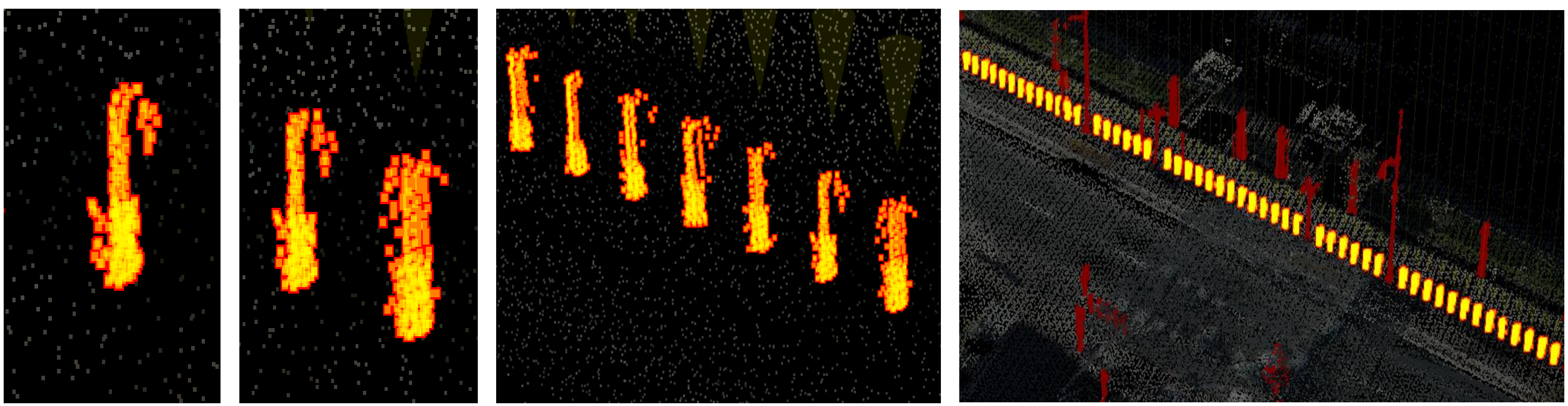
*Lower* *Higher*

Probability of user providing label for group

# Group Active Learning

Benefit of a group:

$$Benefit = p_{Label} \cdot \boxed{T_{1-by-1}} - T_{group}$$



*Lower* *Higher*

Time to recognize and label objects 1-by-1

# Group Active Learning

Time to recognize and label objects in group 1-by-1:

$$Benefit = p_{Label} \cdot \boxed{T_{1-by-1}} - T_{group}$$

$$\boxed{T_{1-by-1}} = (T_{id} + T_{label}) \cdot |group|$$

# Group Active Learning

Time to recognize and label group of objects:

$$Benefit = p_{Label} \cdot T_{1-by-1} - \boxed{T_{group}}$$

$$T_{1-by-1} = (T_{id} + T_{label}) \cdot |group|$$

$$\boxed{T_{group}} = T_{id} + \sum T_{ver} + T_{label}$$

# Group Active Learning

Model recognition and label selection with Hick's Law:

$$Benefit = p_{Label} \cdot T_{1-by-1} - T_{group}$$

$$T_{1-by-1} = (T_{id} + T_{label}) \cdot |group|$$

$$T_{group} = \boxed{T_{id}} + \sum T_{ver} + \boxed{T_{label}}$$

$$\boxed{T_{id}} \approx a_{id} \log_2(n+1)$$

$$\boxed{T_{label}} \approx a_{label} \log_2(n+1)$$

Hick's Law:

$$T_{CRT} = aH = a\sum_{i}^{n} p_i \log_2\left( 1/p_i + 1 \right)$$

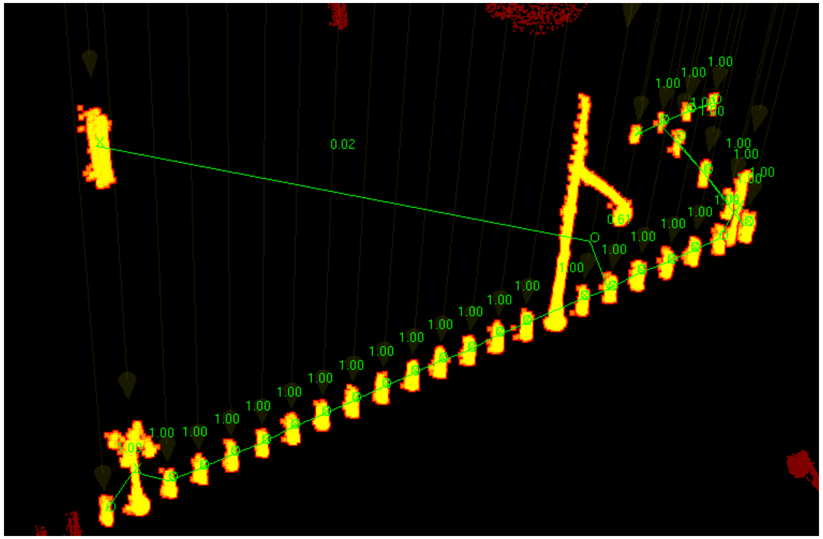
# Group Active Learning

Model group verification time based on similarity of adjacent objects:

$$Benefit = p_{Label} \cdot T_{1-by-1} - T_{group}$$

$$T_{1-by-1} = (T_{id} + T_{label}) \cdot |group|$$

$$T_{group} = T_{id} + \boxed{\sum T_{ver}} + T_{label}$$



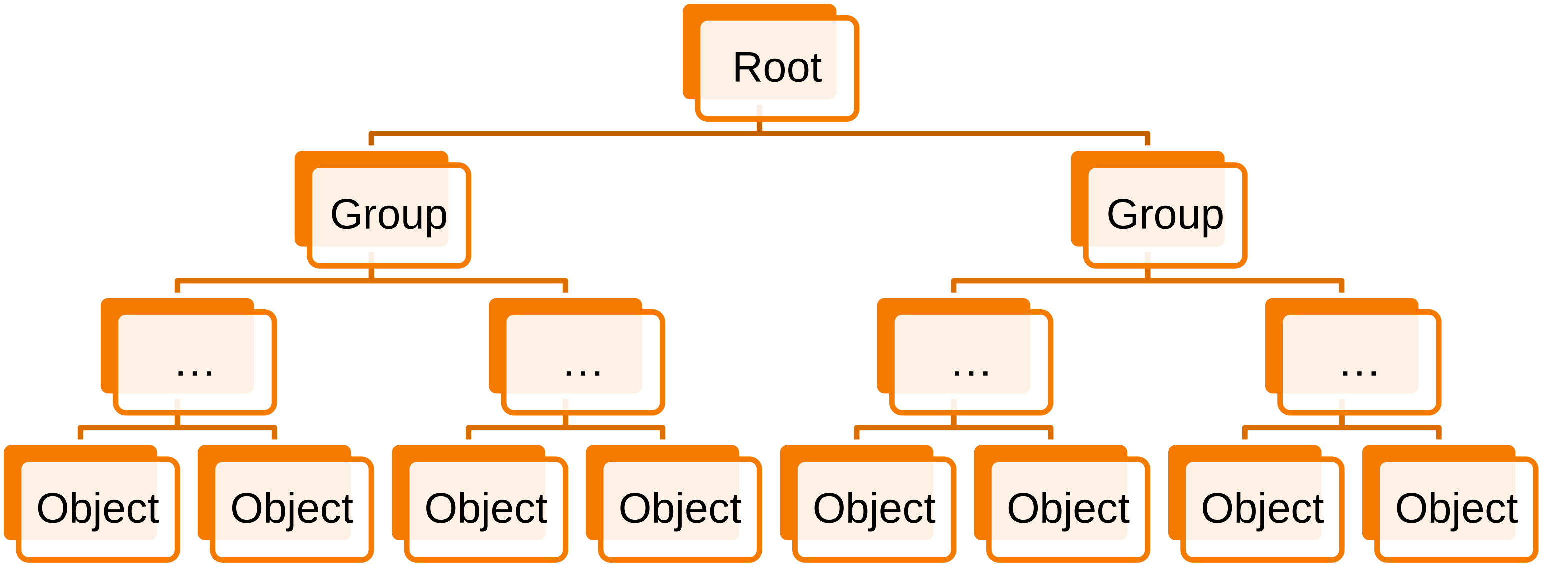
# Group Active Learning

Challenges: choosing good groups to show user

1. Model the benefit of showing a group
2. Provide real-time algorithm to construct next group

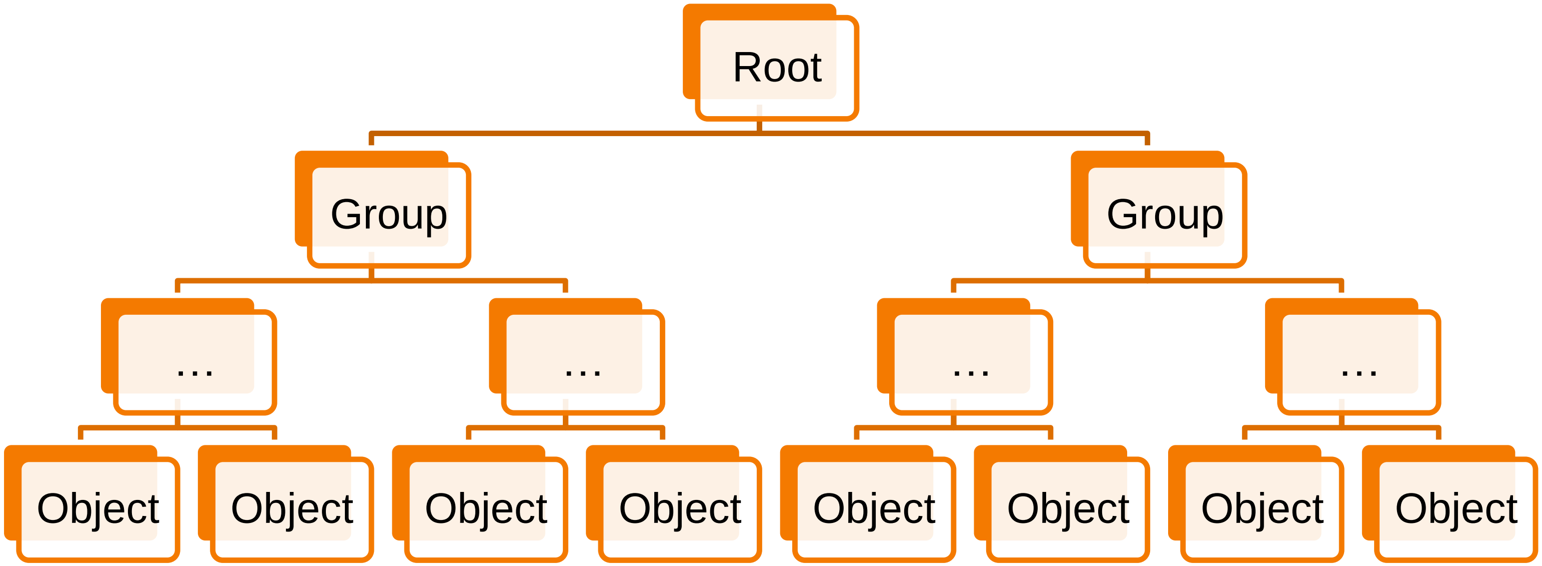
# Group Active Learning

Hierarchical clustering algorithm to construct candidate groups



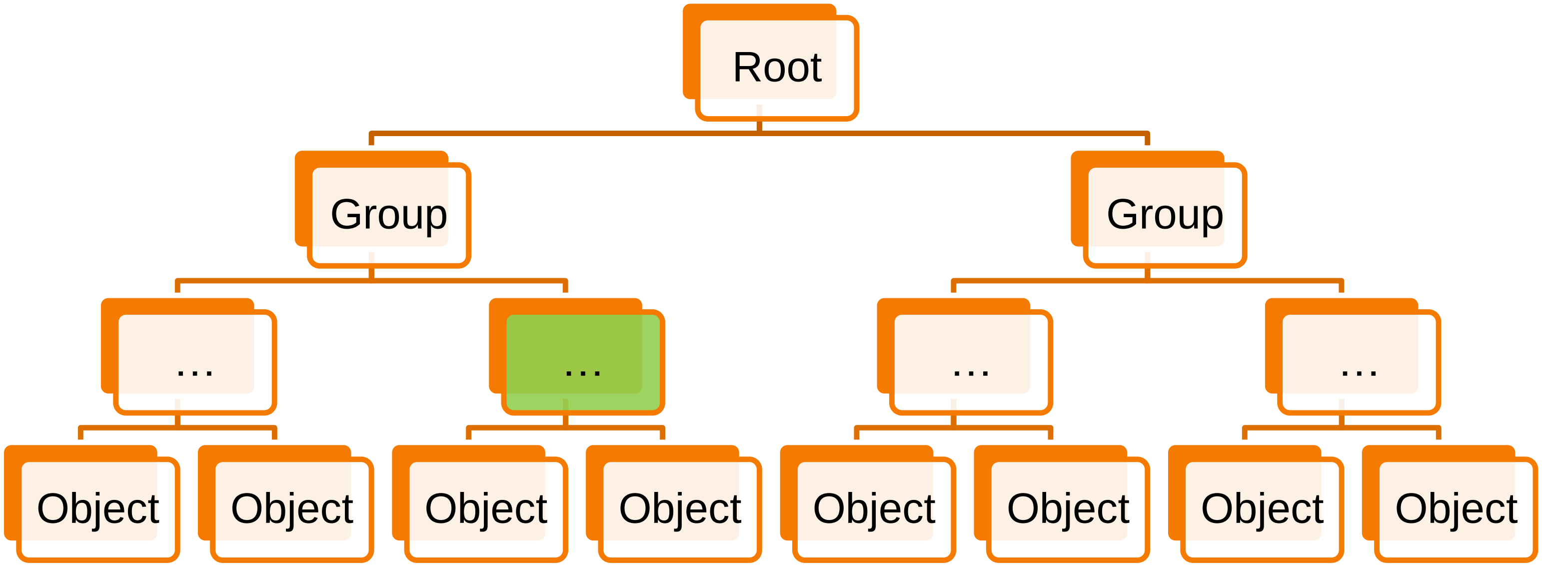
# Group Active Learning

Hierarchical clustering algorithm to construct candidate groups



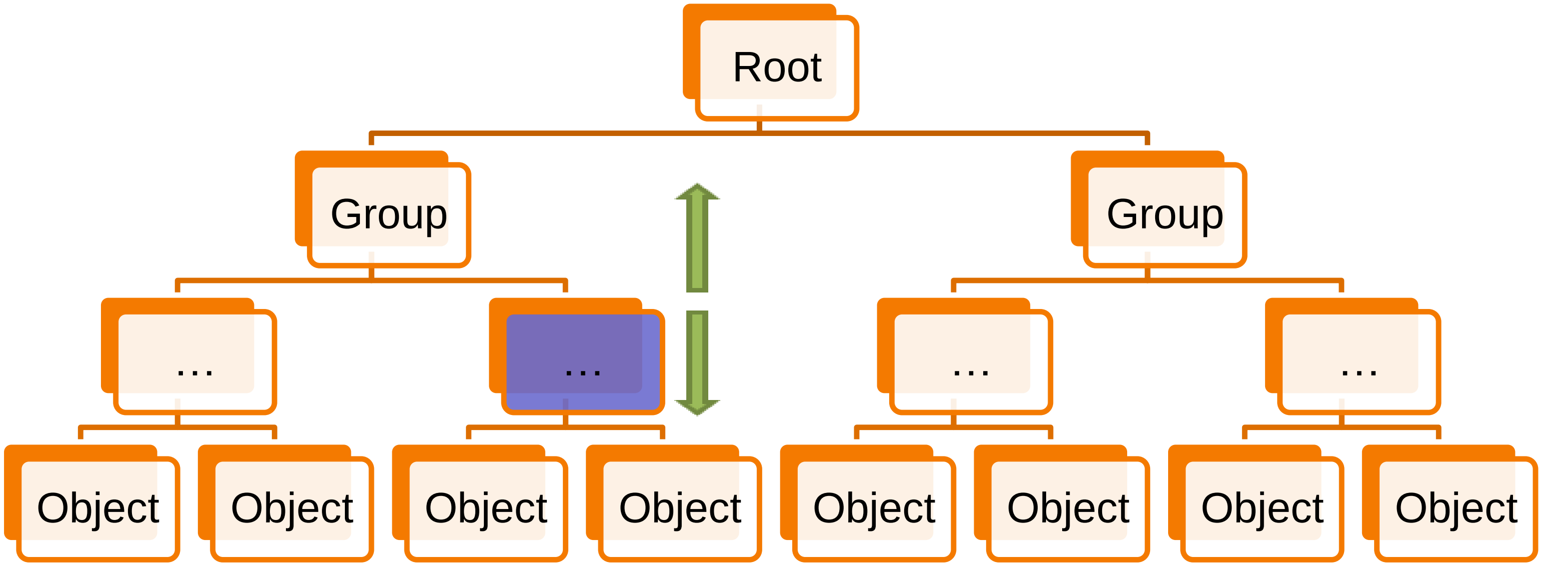
# Group Active Learning

Select the most beneficial group to show user …



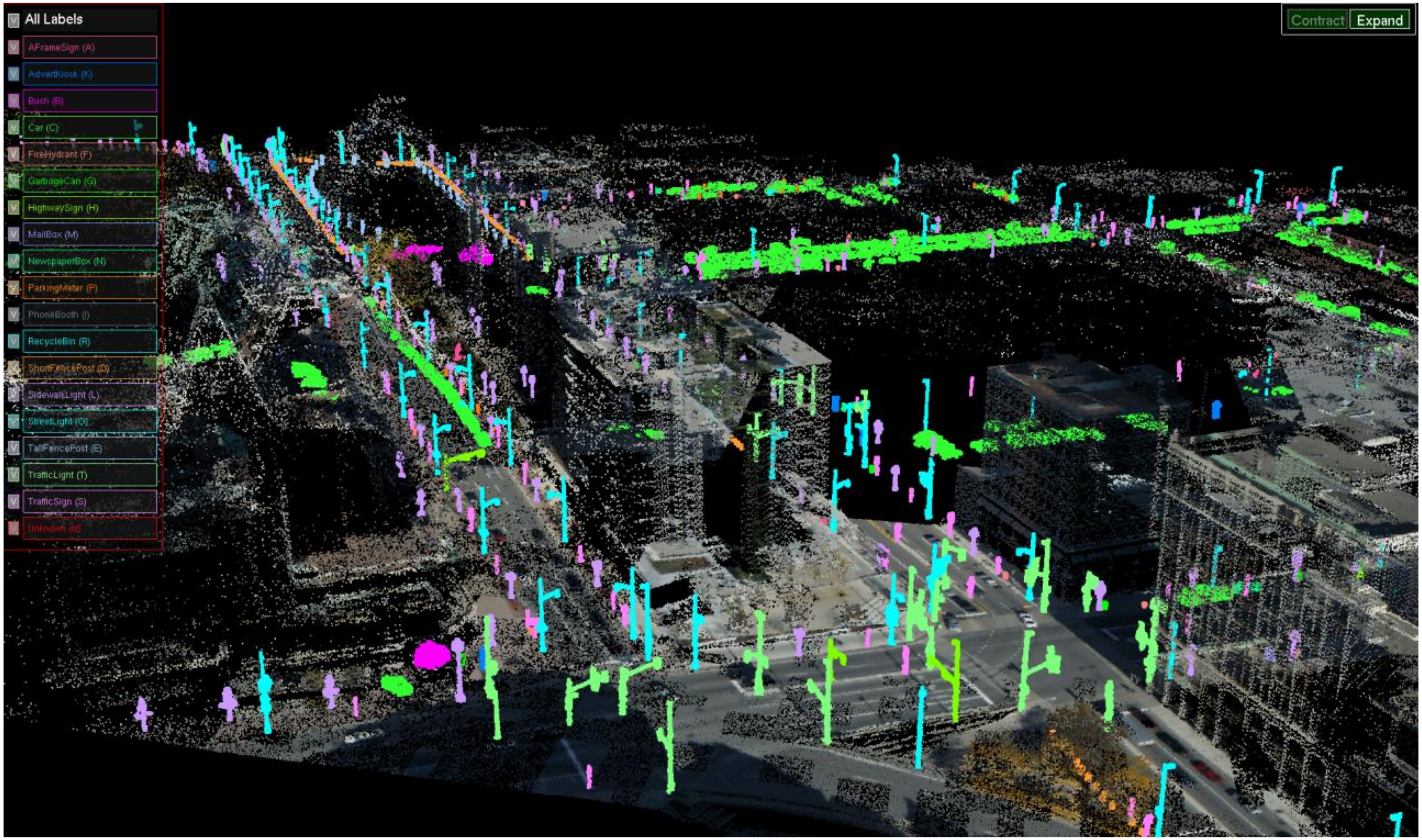
# Group Active Learning

Contract (or expand) group if user requests



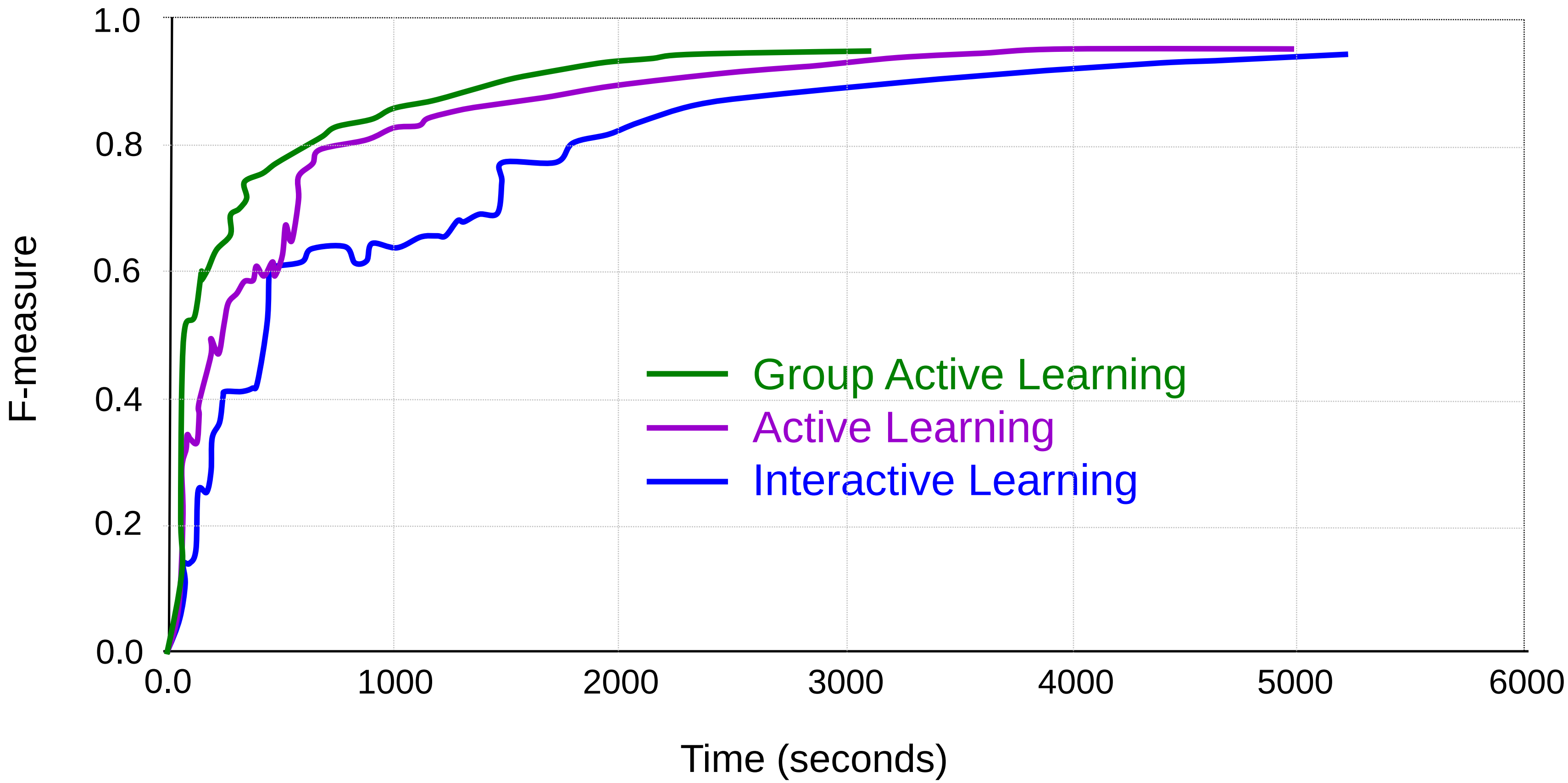
# Group Active Learning Experiment

Same protocol as before …



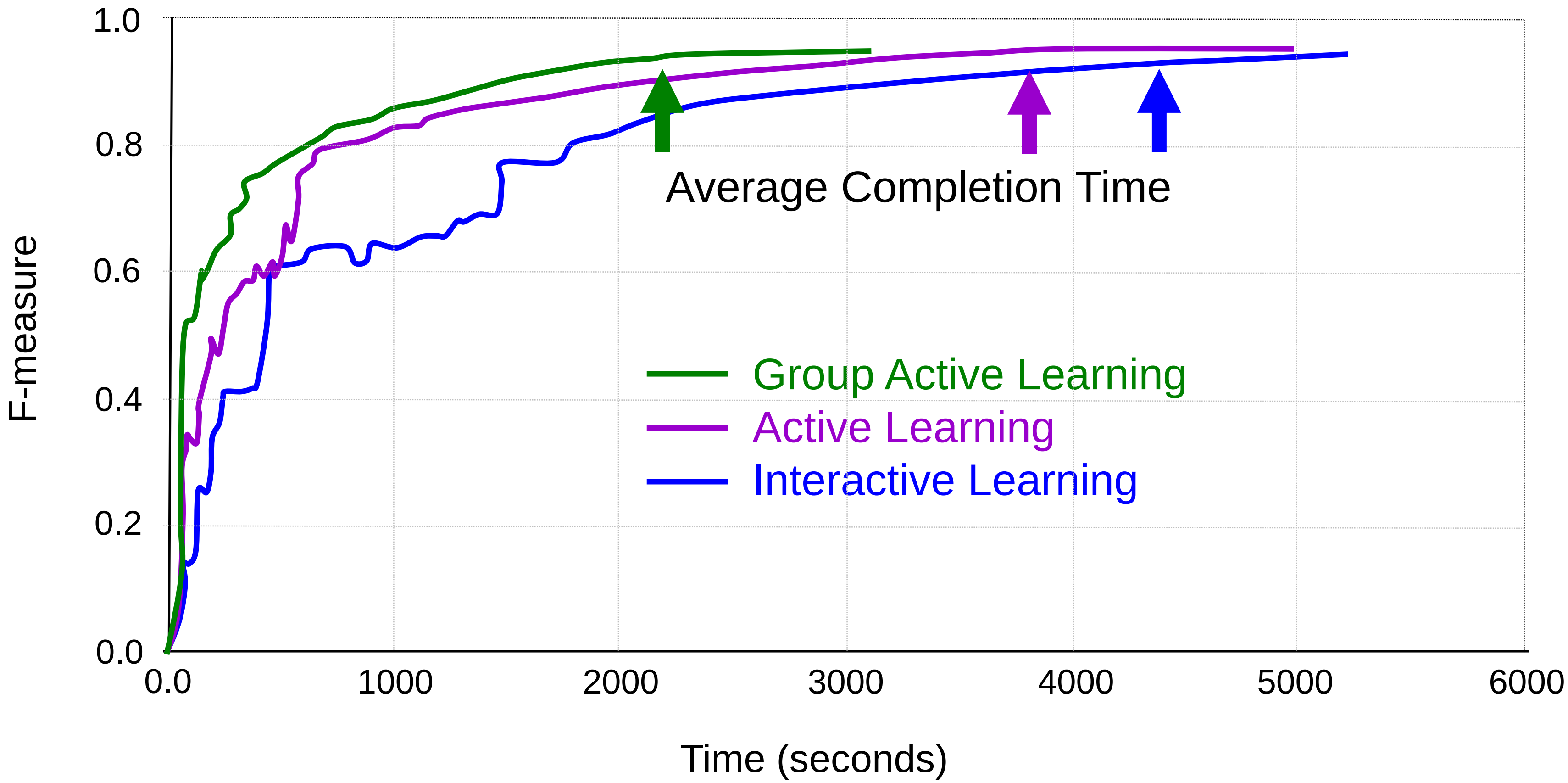
# Group Active Learning Results

Group active learning required less time ☺



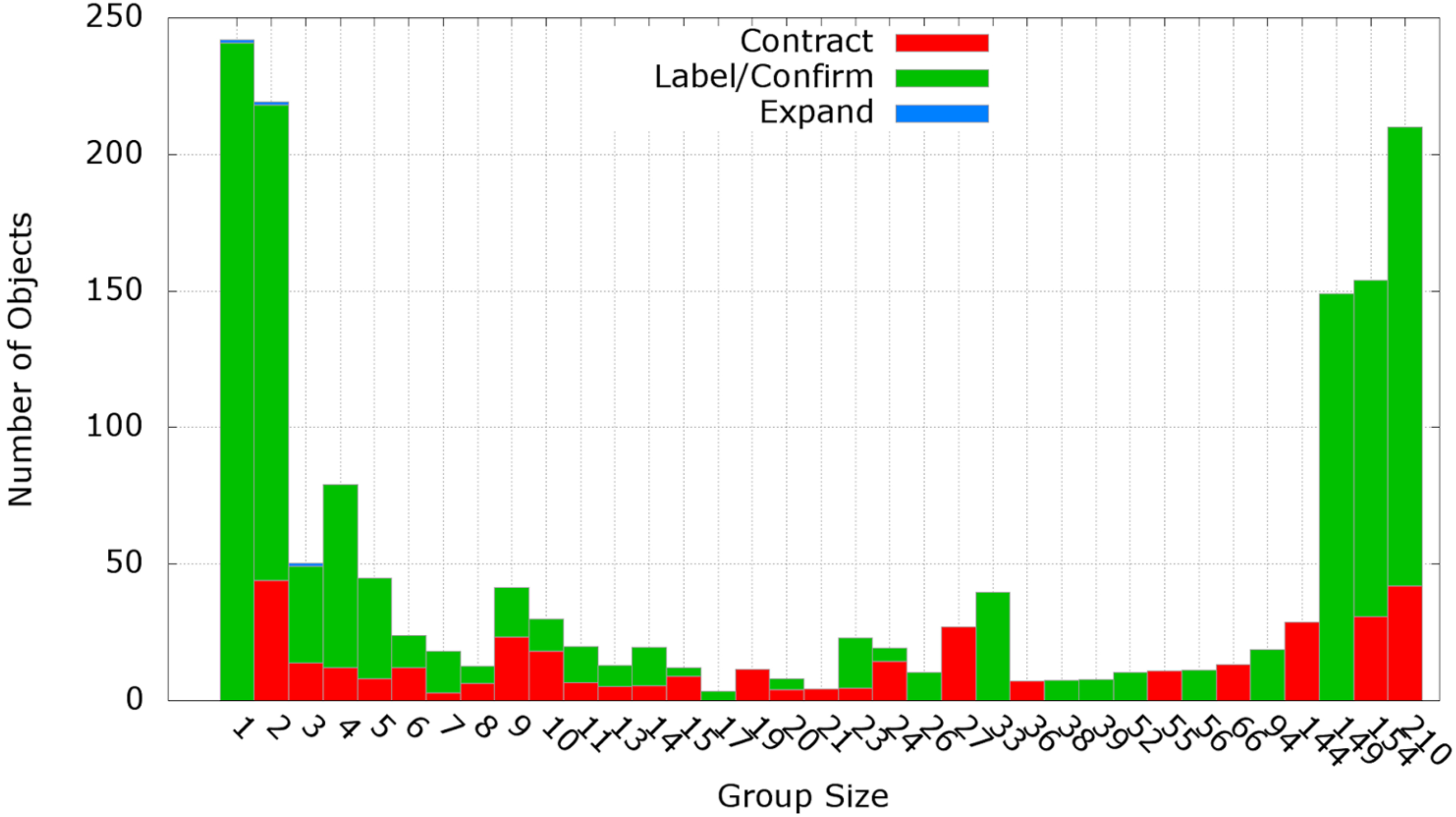
# Group Active Learning Results

Group active learning required less time ☺



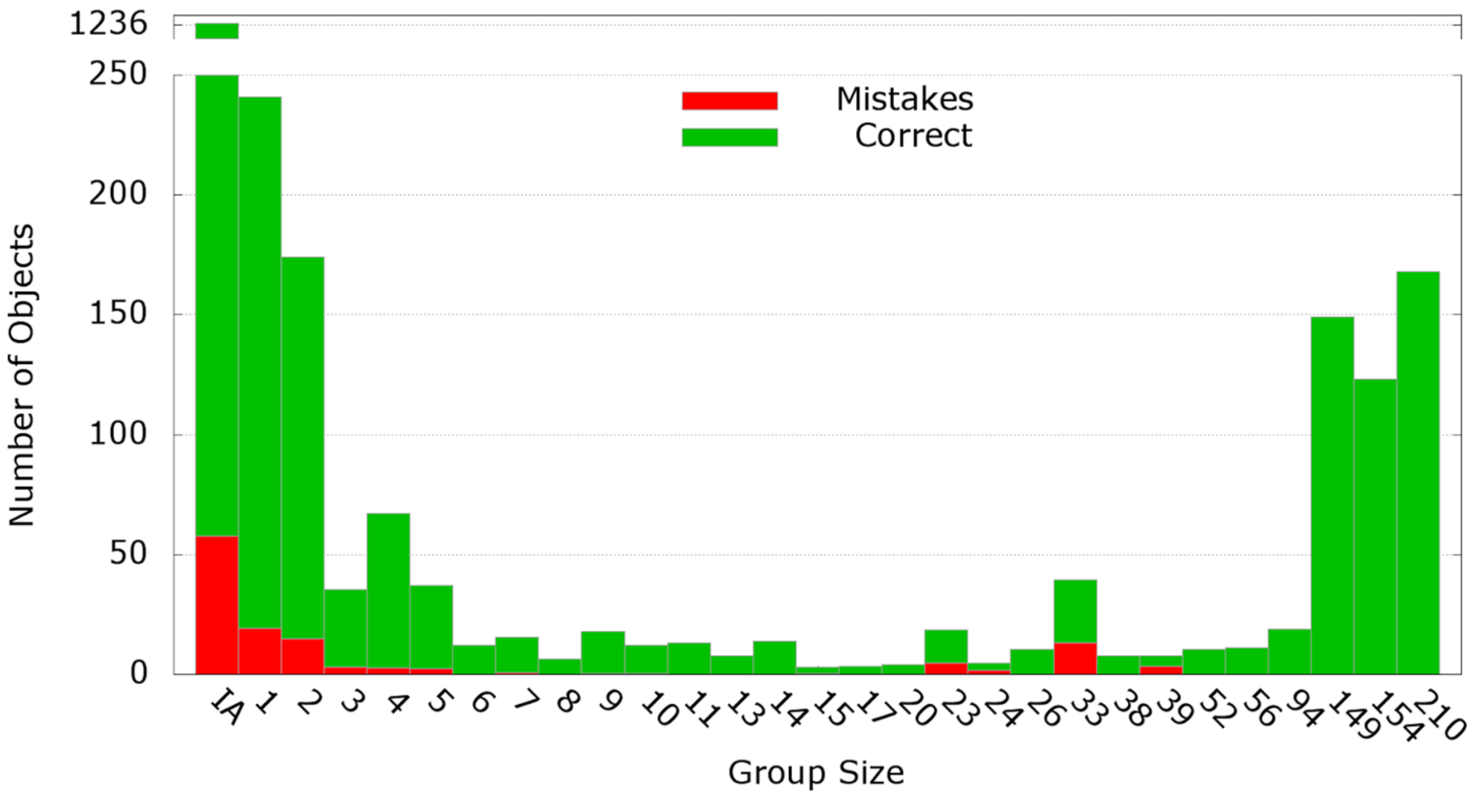
# Group Active Learning Results

Subjects label large groups created by our algorithm ☺



# Group Active Learning Results

Labeling by group does not increase mistakes ☺



# Comparison of Results

| | Completion (seconds) | Final F-measure (%) |
| --- | --- | --- |
| Interactive Learning | 4401 +/- 787 | 94 +/- 1 |
| Active Learning | 3855 +/- 837 | 96 +/- 1 |
| Group Active Learning | 2281 +/- 561 | 95 +/- 3 |

# Summary

Motivation:

- Almost every real application of semantic labeling requires manual annotation (to achieve production quality)

Research question?

- How to design labeling interfaces that help users achieve 100% accuracy in the least amount of time?

Some ideas from this work:

- Use omain-specific interfaces to accelerate labeling
- Interleave training and prediction during interactive process
- Utilize predicted object classes to filter interactive selections
- Automate camera control and search for objects
- Leverage Gestault principles to label groups of objects

# Future Work

Joint localization/segmentation/labeling:

- What is the best interactive interface for simulatenous localization, segmentation, and labeling of objects?

Computational steering:

- Can interactive techniques guide training of deep networks (user-in-the-loop training)?

Other media:

- Can group active learning accelerate labeling of images, sounds, or other media not natively embedded in 3D?

# Acknowledgments


Princeton students:

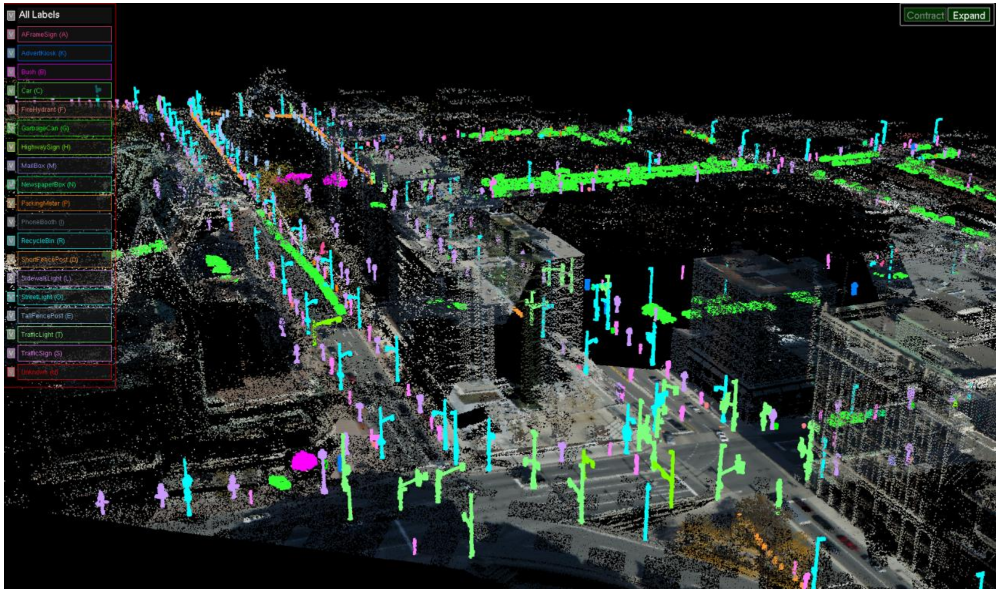
- Aleksey Boyko, Aleksey Golovinskiy

Data:

- Neptec, Google

Funding:

- Intel, NSF, Google


Thank You!