Synthesizing Scenes for Instance Detection

Synthesizing Scenes for Instance Detection

Debidatta Dwibedi

Thesis Committee

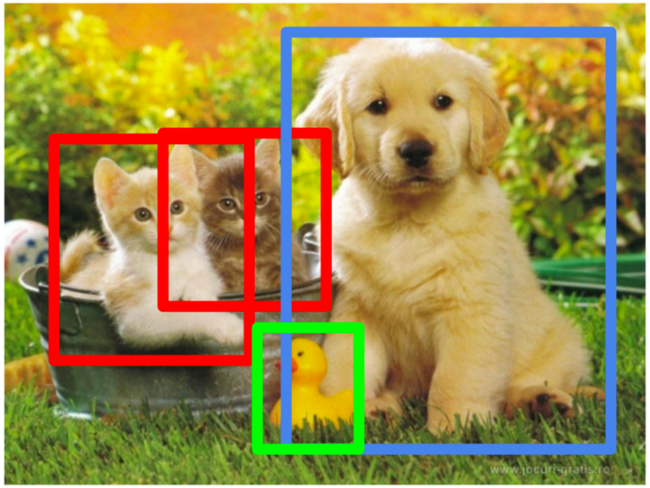
Martial Hebert

Michael Kaess

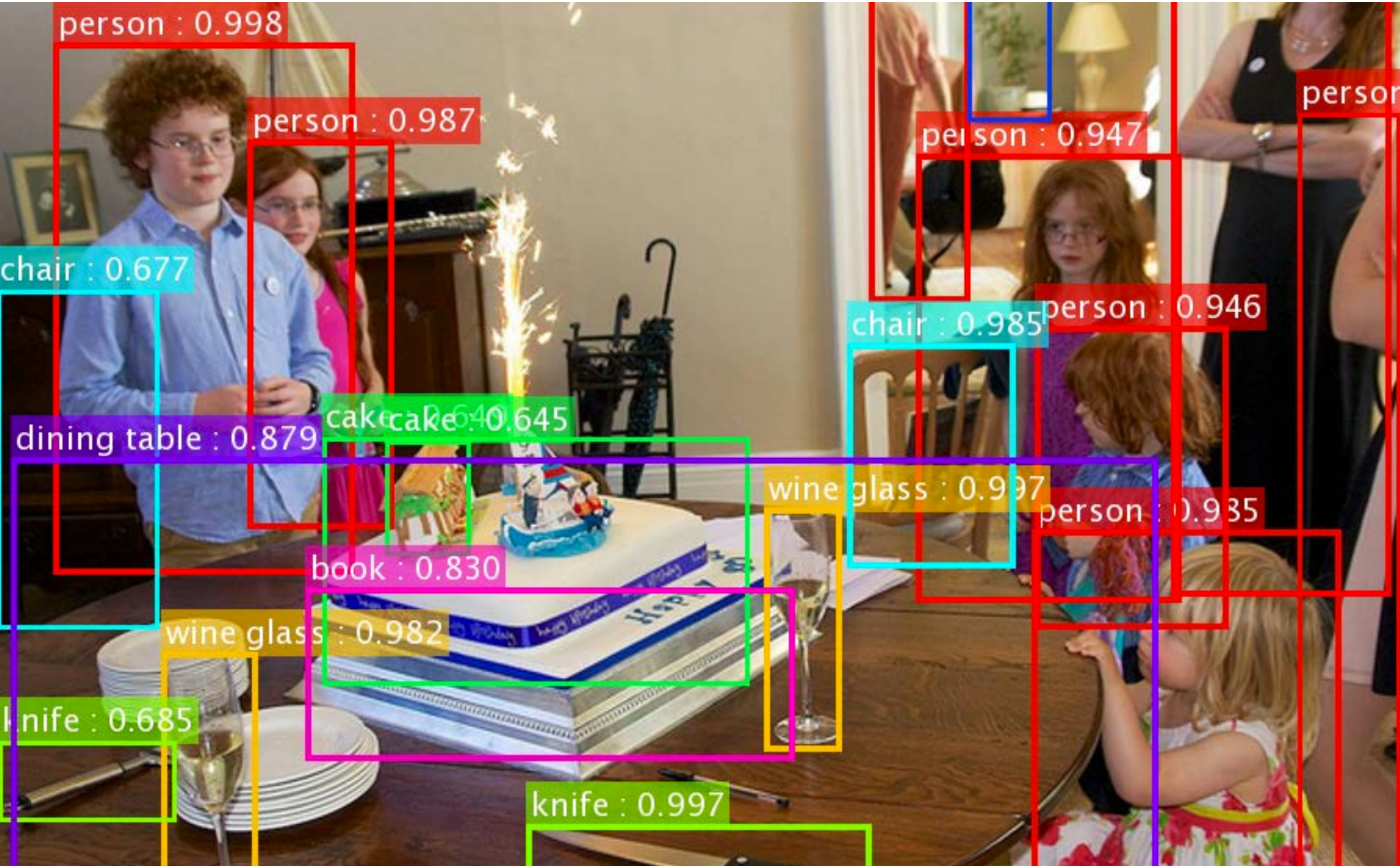
Ishan Misra

INTRODUCTION

Common Vision Tasks

Classification	Classification + Localization	Object Detection
		
CAT	CAT	CAT, DOG, DUCK

Object Detection

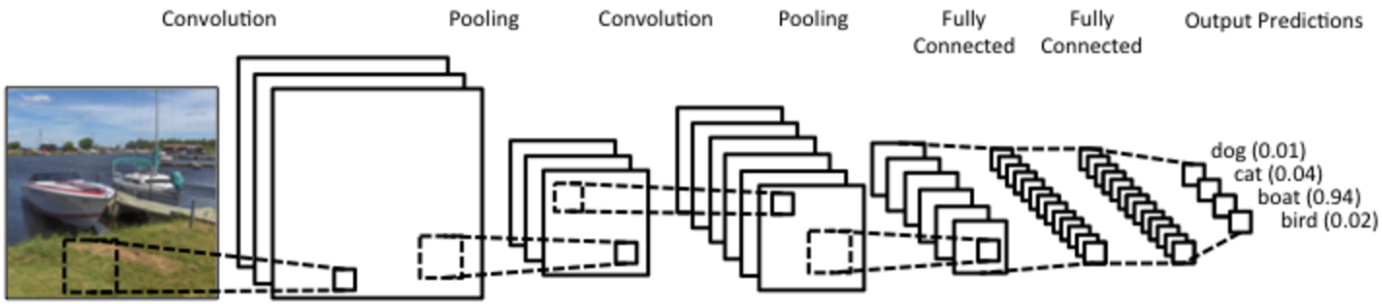


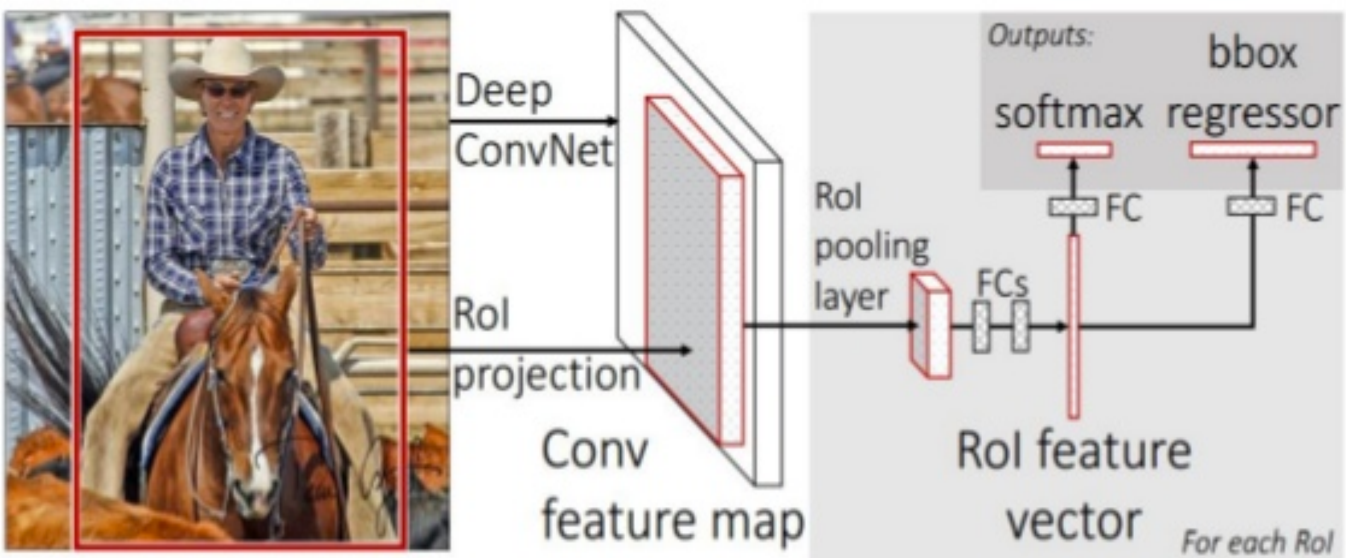
Large Annotated Datasets

14,197,122 labeled images as of April 2017

Humans annotate images on Mechanical Turk

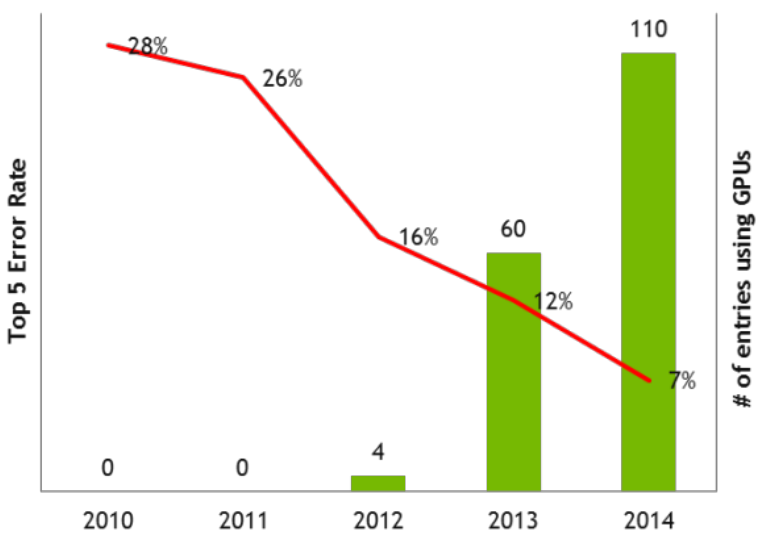
Deep Learning





GPUs

IMAGENET



INSTANCE DETECTION

Comparison with Object Detection

Instance Detection

Instance v/s Object Detection

Object Detection

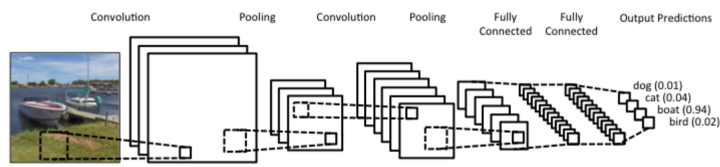
Granola Bars	Cups

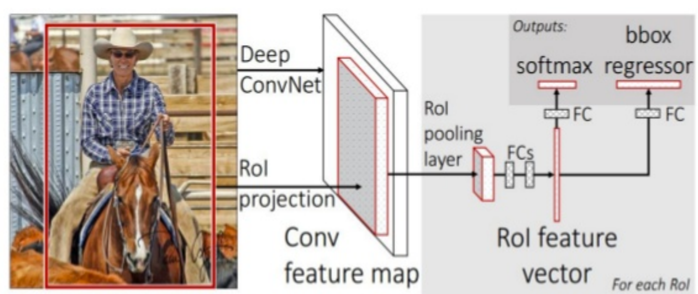
Granola Bar 1	Granola Bar 2	Cup 1	Cup 2	Cup 3	Cup 4

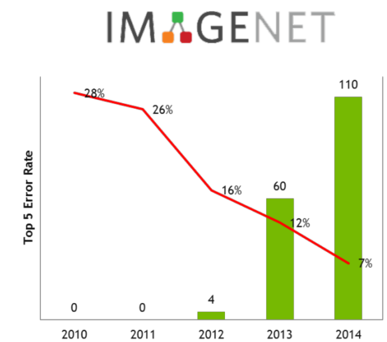
Instance Detection

Object Detection

- Large annotated datasets
- Better machine learning models
- Faster computation





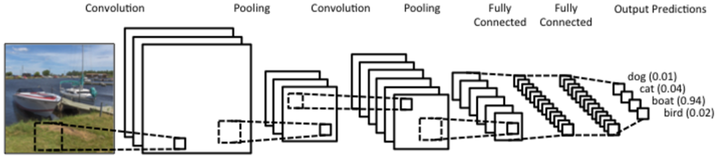


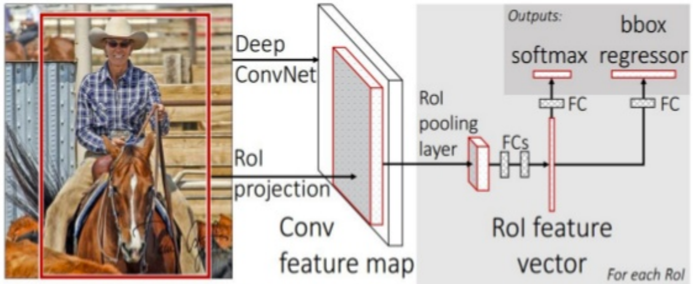
Instance Detection

Useful for instance detection too

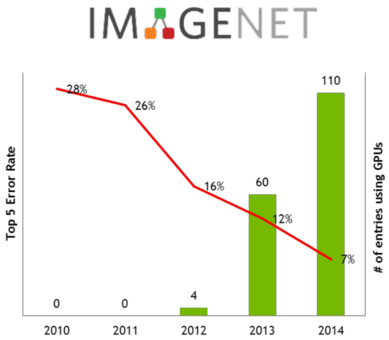
- Large annotated datasets

- Better machine learning models





- Faster computation



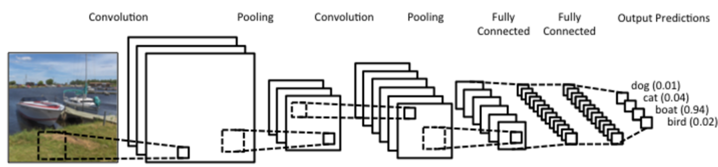
Instance Detection

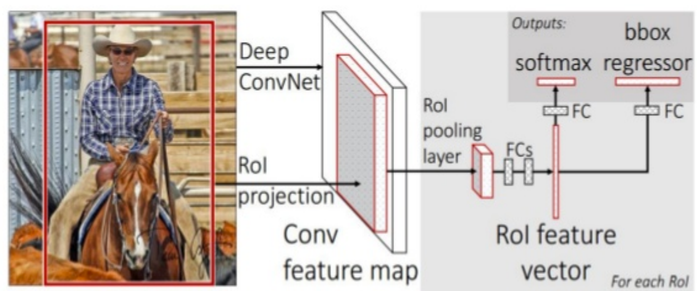
Doesn't exist for
all applications

Useful for instance detection too

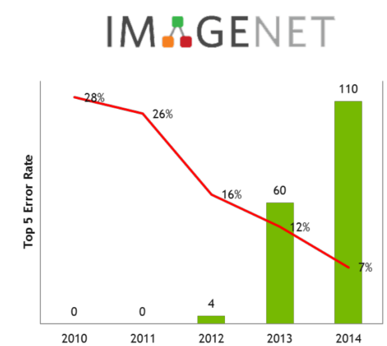
- Large annotated datasets

- Better machine learning models





- Faster computation



CREATING ANNOTATED DATASETS

Methods used to collect annotated data

Data Collection for Object Detection

1. Retrieve image of object from the Internet
2. Label each collected image

(a) Category labeling

(b) Instance spotting

(c) Instance segmentation

Data Collection for Instance Detection

1. Create scenes with relevant instances
2. Capture images
3. Manually label each image

Can we automate the annotated data creation process?

VIDEOS

Leveraging videos to reduce annotation effort

Advantages of using Video

1. Videos are easy to capture
2. Propagate bounding boxes from one frame to the next using object tracking

Manually label one frame

Propagate label to next frames by tracking

Reduction in Effort

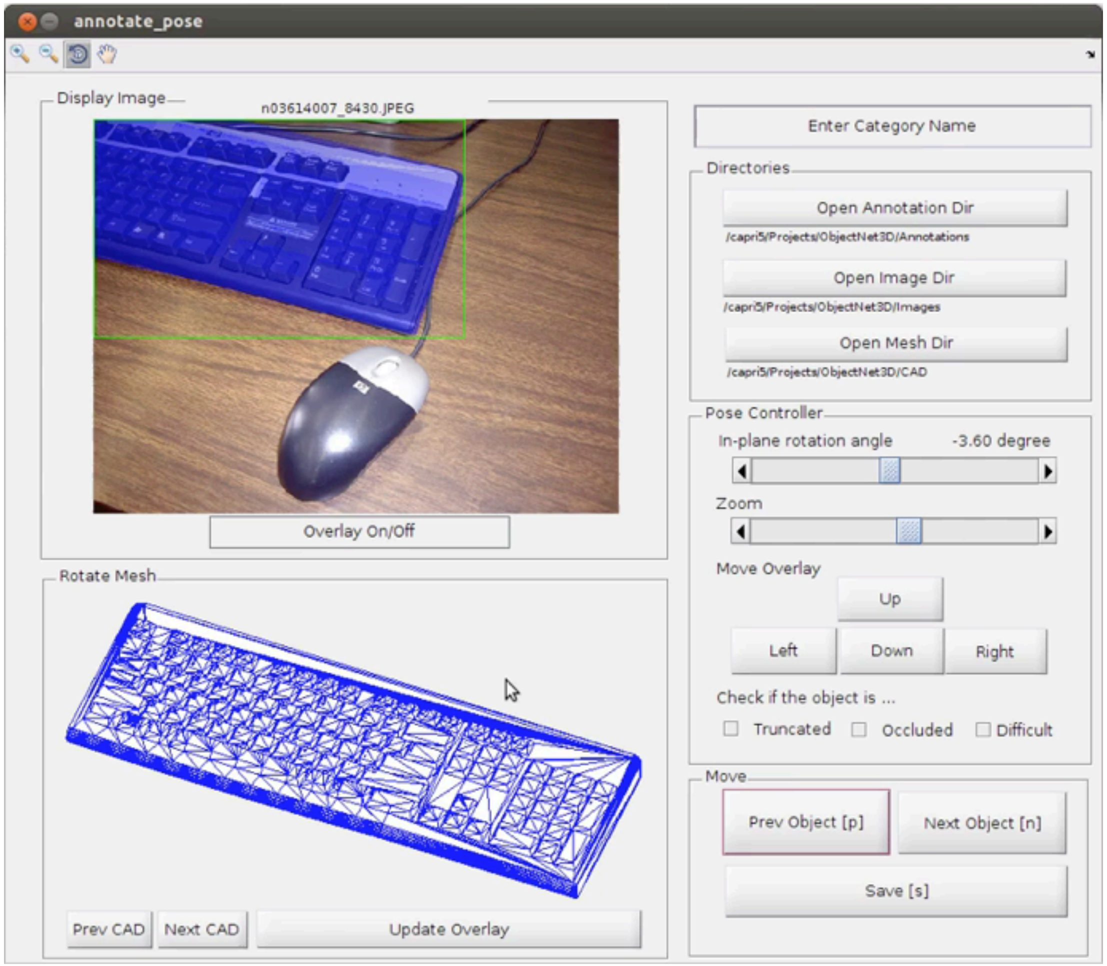
1. Need to manually label 10X fewer frames to get a dataset of equivalent size

2. No reduction in performance of object detector

3D RECONSTRUCTION

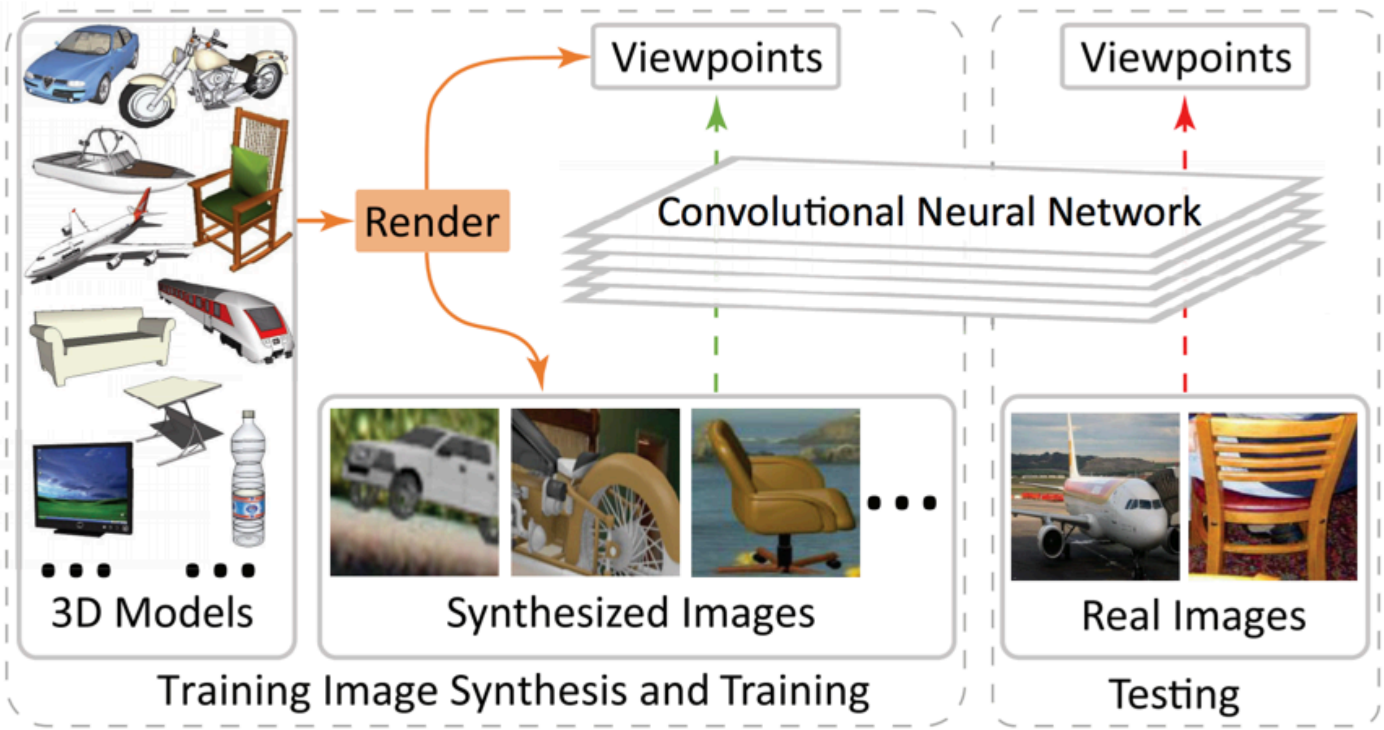
Using SFM to produce pose and bounding box annotations for objects

ObjectNet3D GUI



1. Too much manual effort to annotate pose

Render-For-CNN



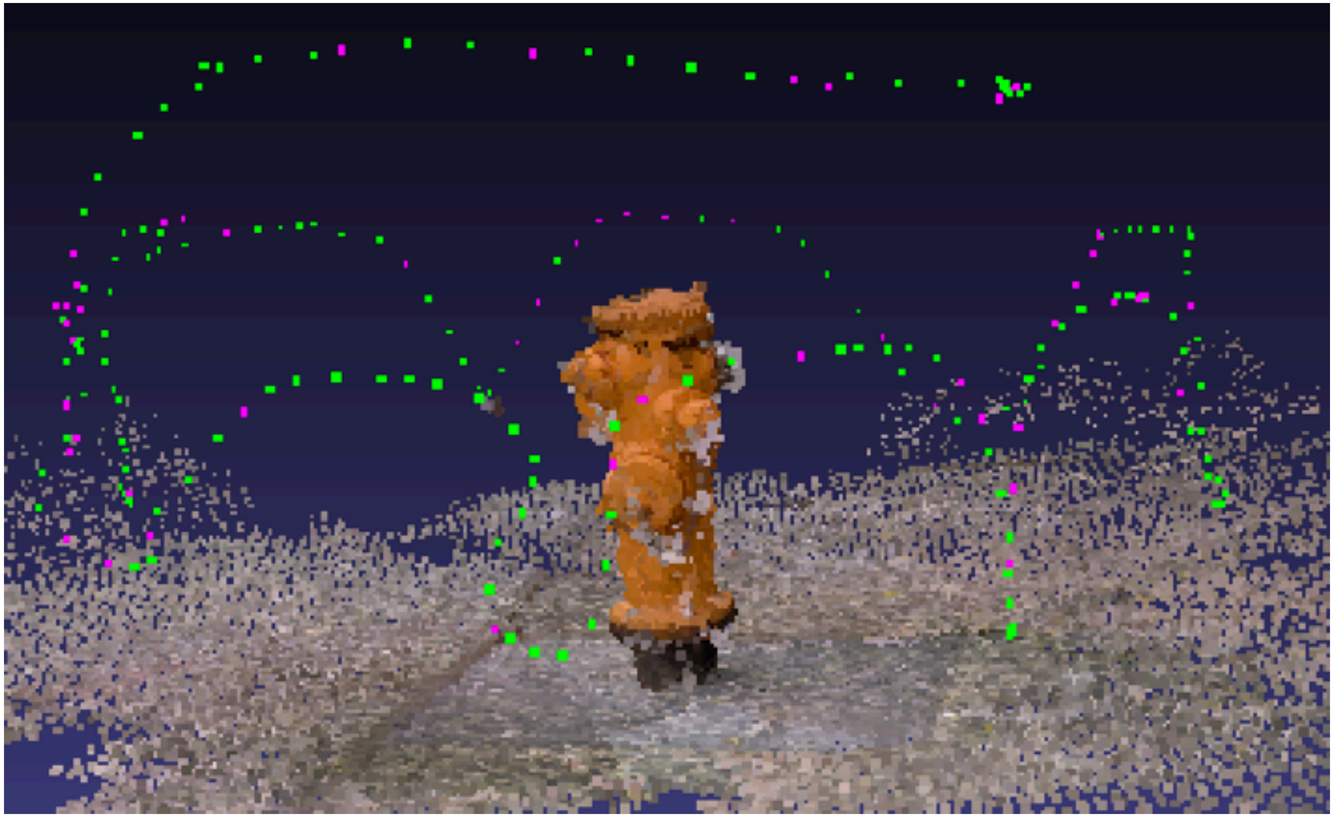
1. No real images of objects used in training
2. Dearth of high-quality models of everyday instances

Can we do better if we have access to the object?

Record Object from Multiple Views

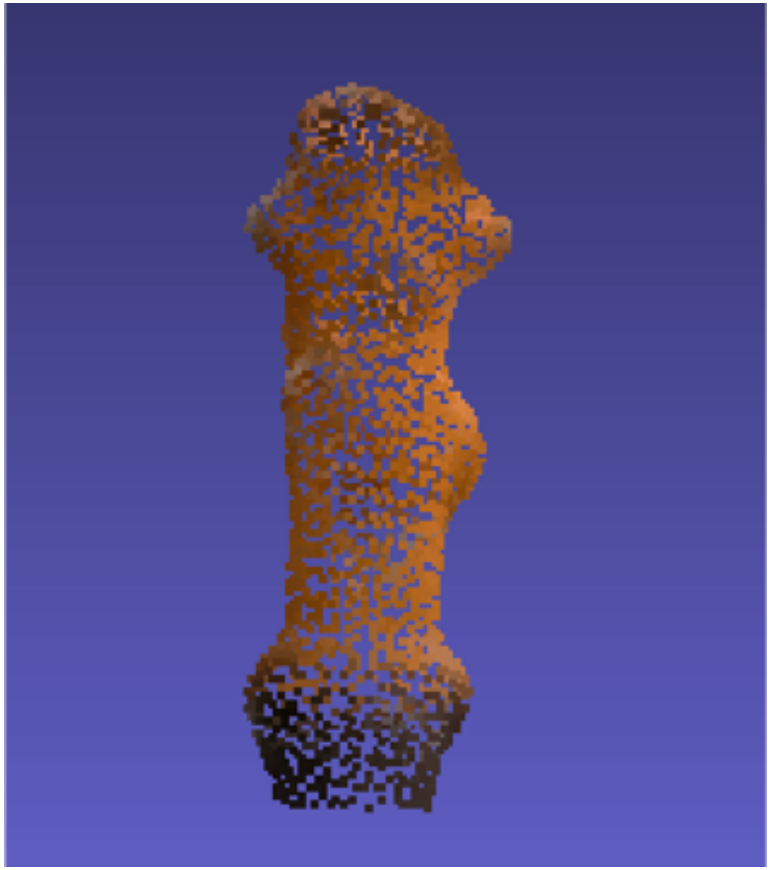



Structure from Motion



Green points represent camera locations in 3D

Structure from Motion



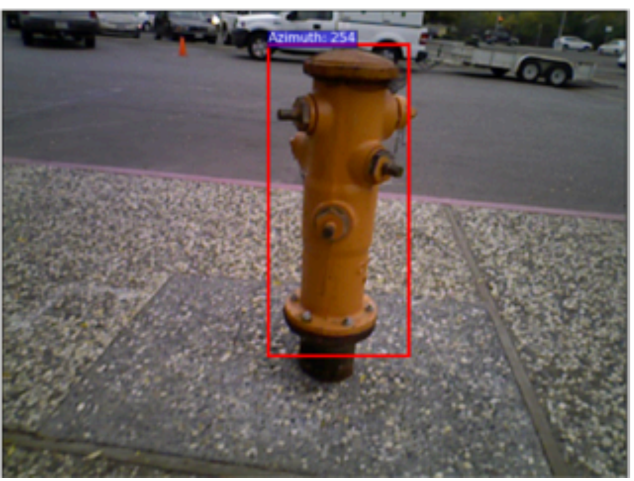
3D points belonging to the object
Project 3D points to 2D to get bounding boxes

Annotation Results

Azimuth = 22

Azimuth = 54

Azimuth = 91

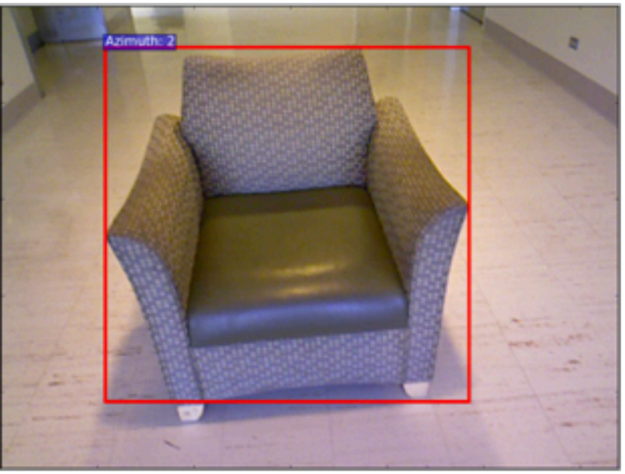


Azimuth = 254

Azimuth = 272

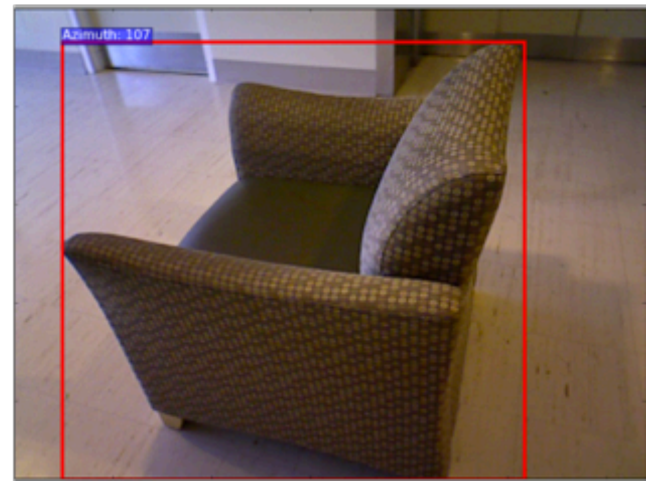
Azimuth = 311

Annotation Results

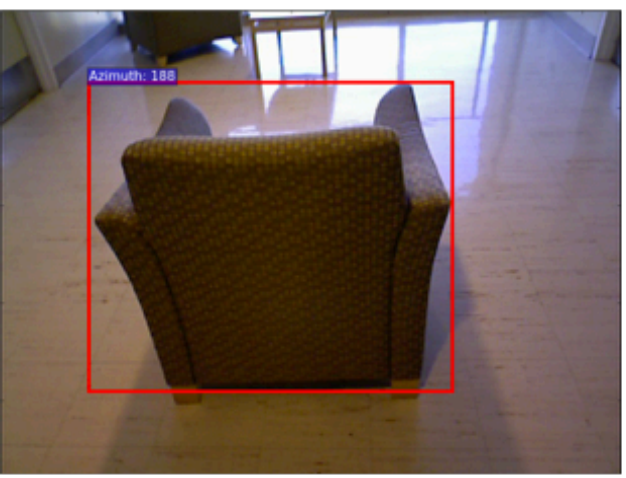


Azimuth = 2

Azimuth = 47



Azimuth = 107

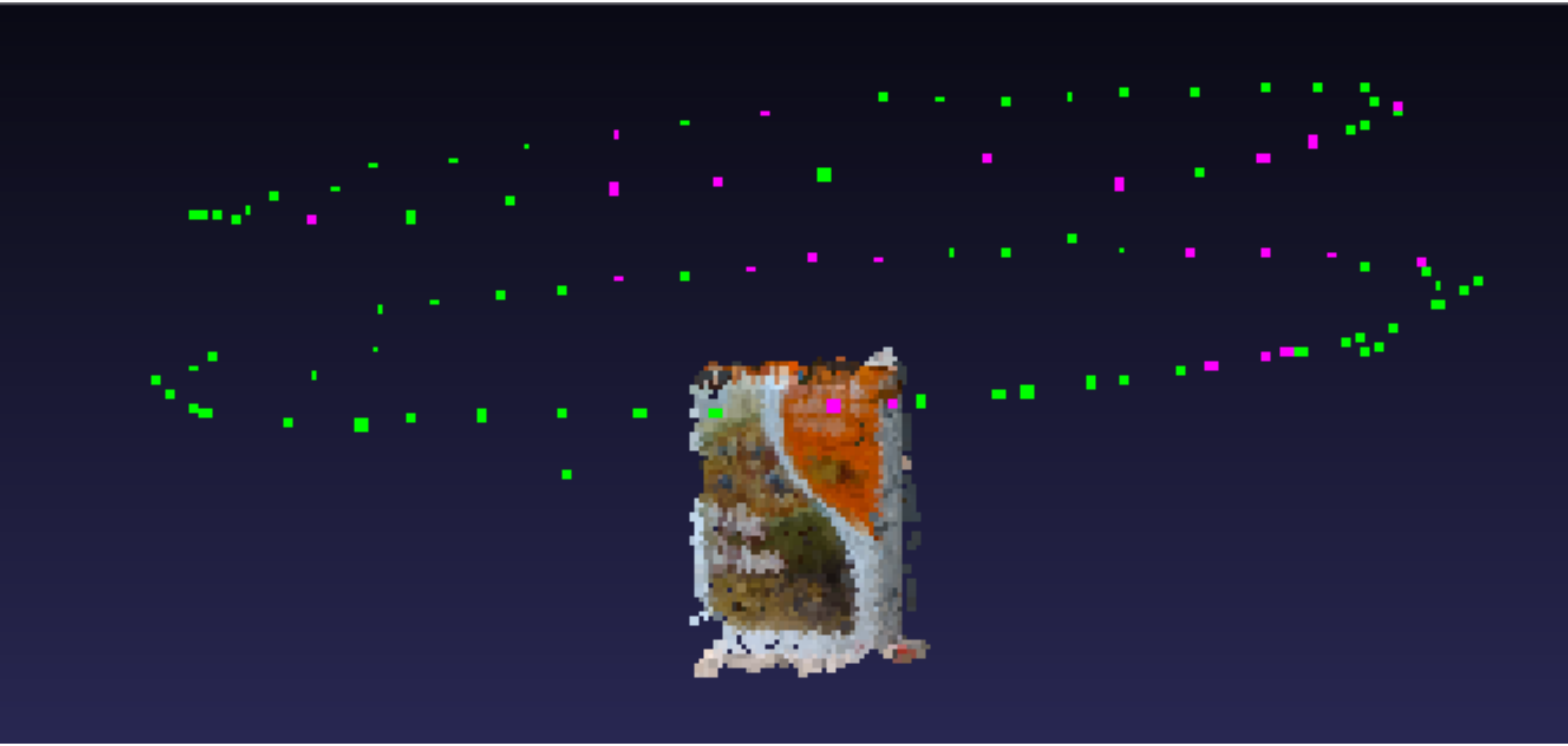


Azimuth = 188

Azimuth = 240

Azimuth = 314

Turntable Results



Can also collect images by using multiple cameras and a turntable to rotate the object

SYNTHESIZING SCENES

Generating synthetic data for the task of instance detection

Proposed Approach

Cut

Object Instances

Background Scenes

Paste

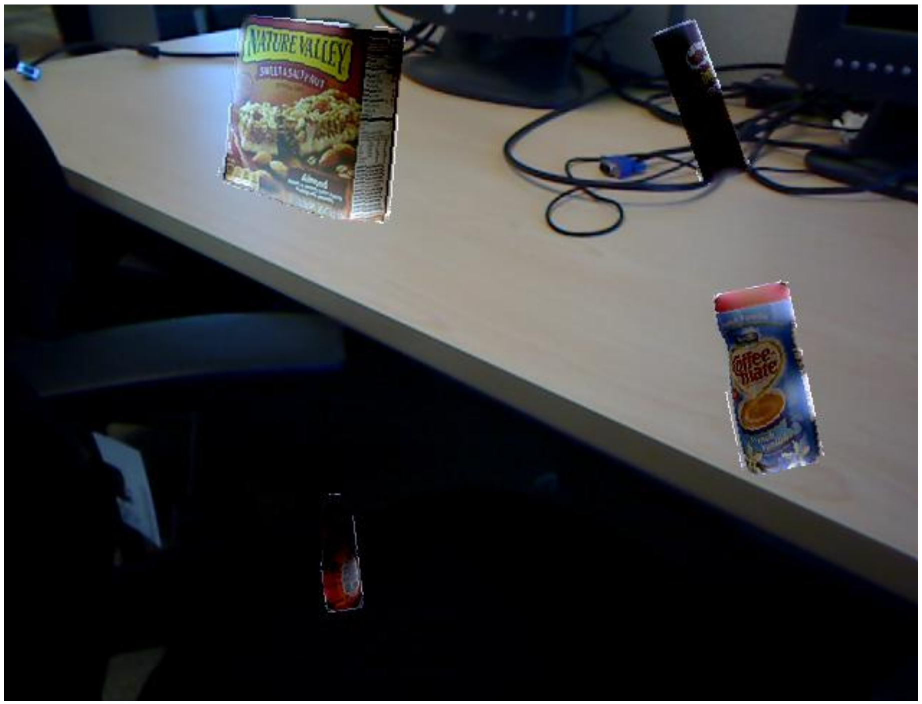
Generated Scenes (Training Data)

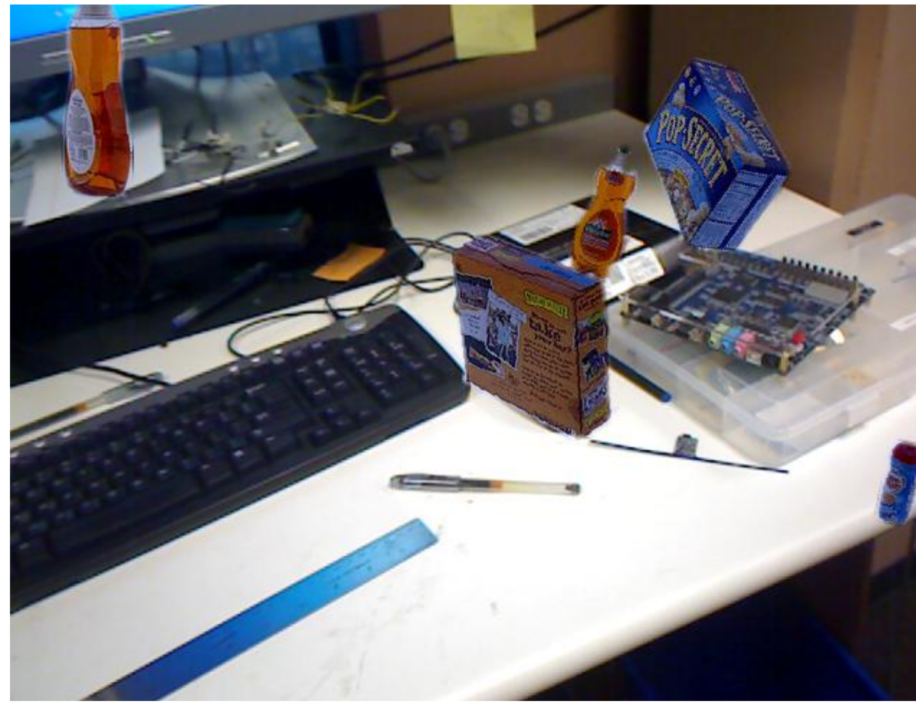
Learn

Detections on Real Images

CHALLENGES

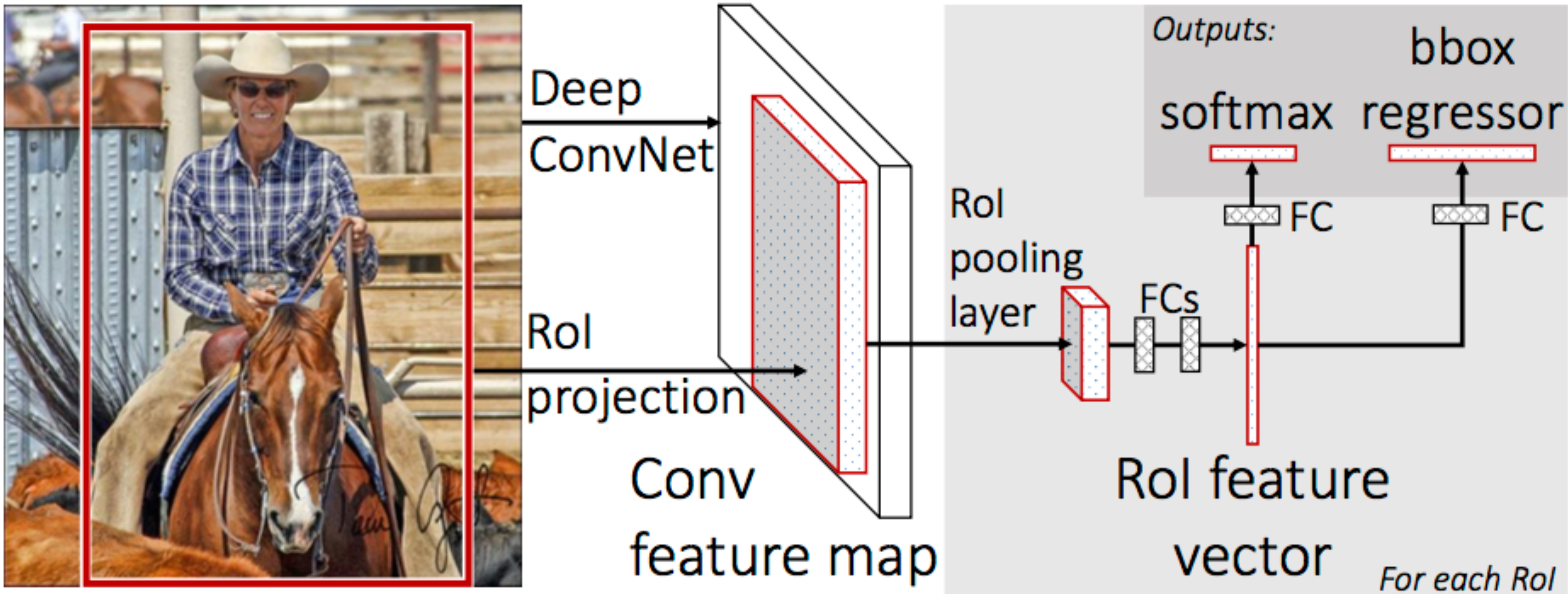
Realism





Don’t training images have to look realistic?

Region based Object Detection Models

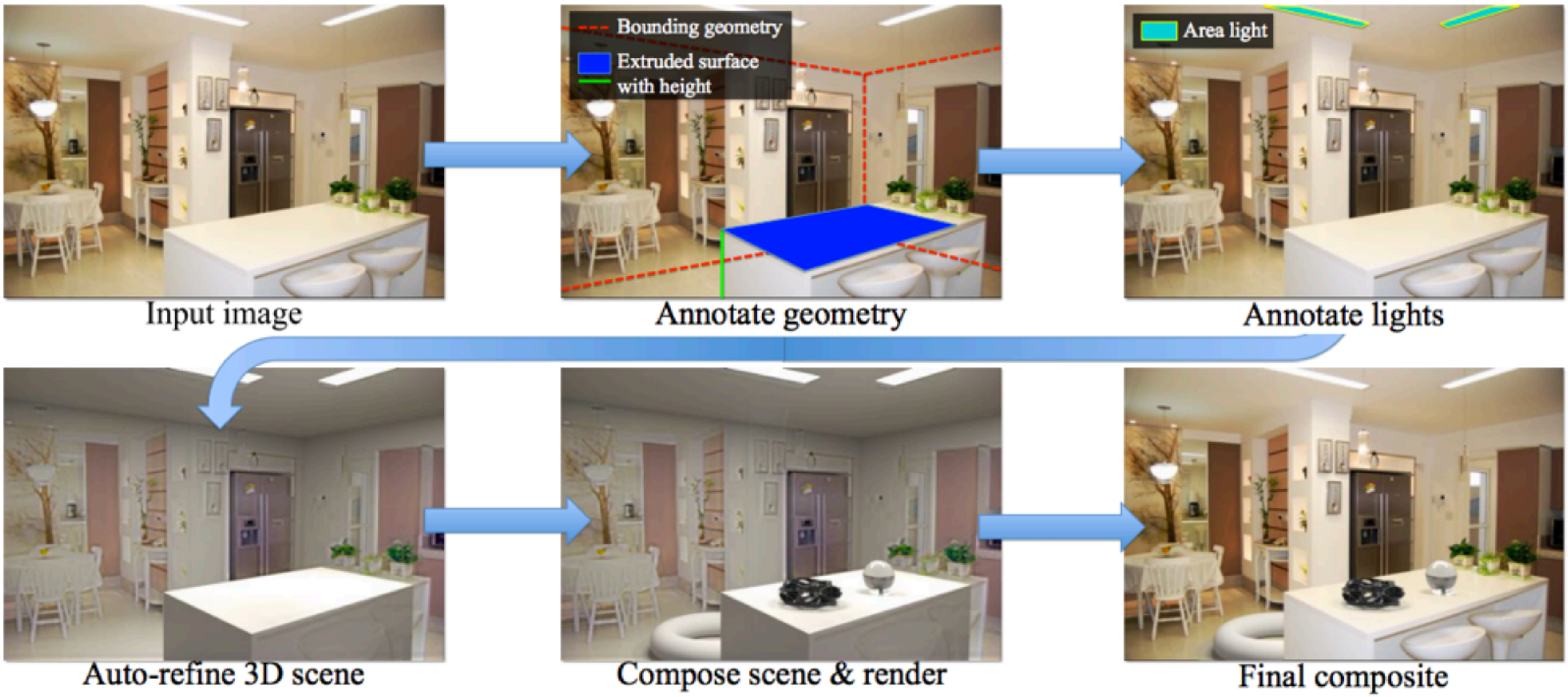


State of the Art Techniques attempt to classify regions
Do we need global realism in training images?

Global Realism

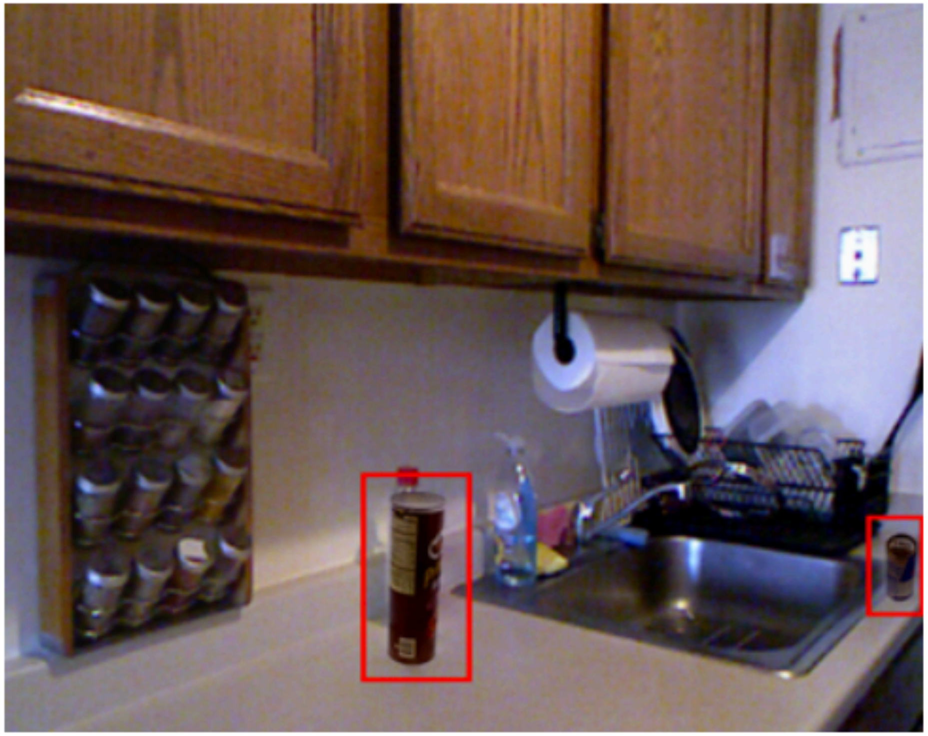
Geometry
Semantics
Scale
Depth
Lighting
Context
Texture
Physics
...

Rendering with Structure Supervision



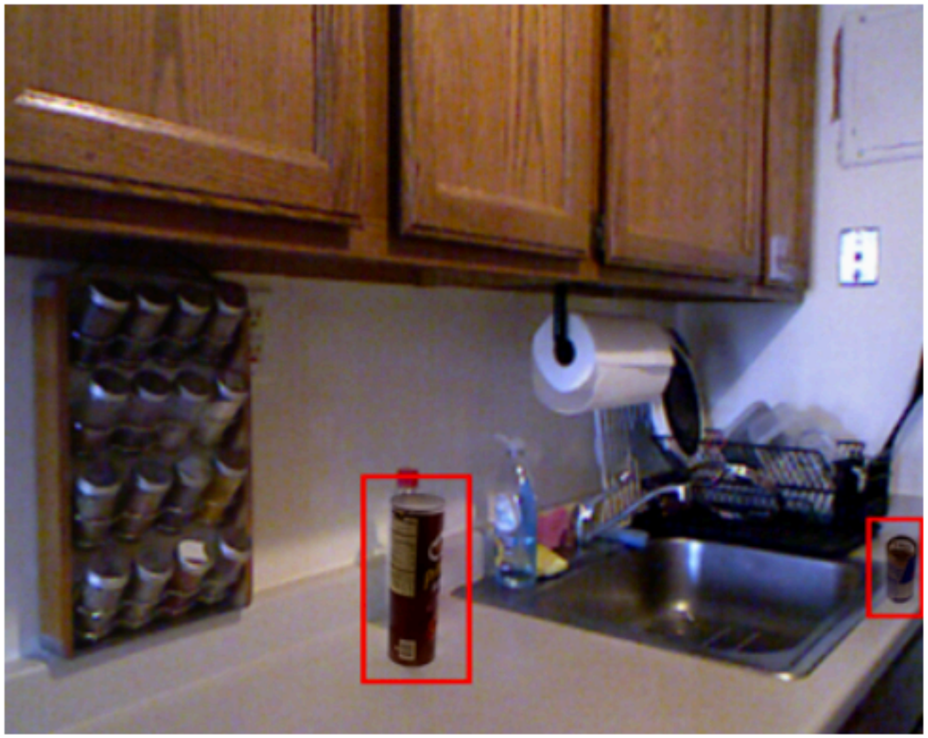
Ensuring global structure is difficult and involves labeling effort

Semantics-and-Geometry Aware Scene Synthesis



Deep learning based approaches can provide decent estimates of semantics and surface normal estimation

Semantics-and-Geometry Aware Scene Synthesis



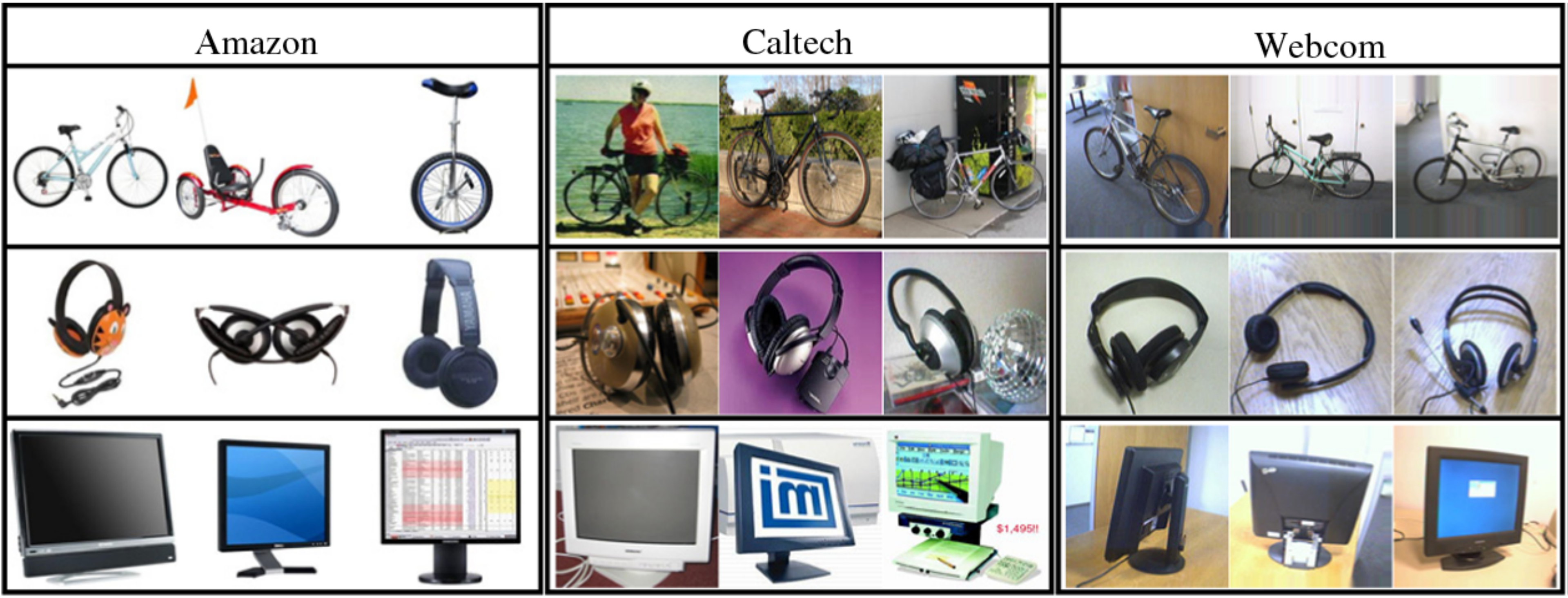
Input to Classification part of Fast R-CNN is only the region
Do we need to render keeping global realism in mind?

Patch Realism

Can we decide from this patch if image is real or synthetic?

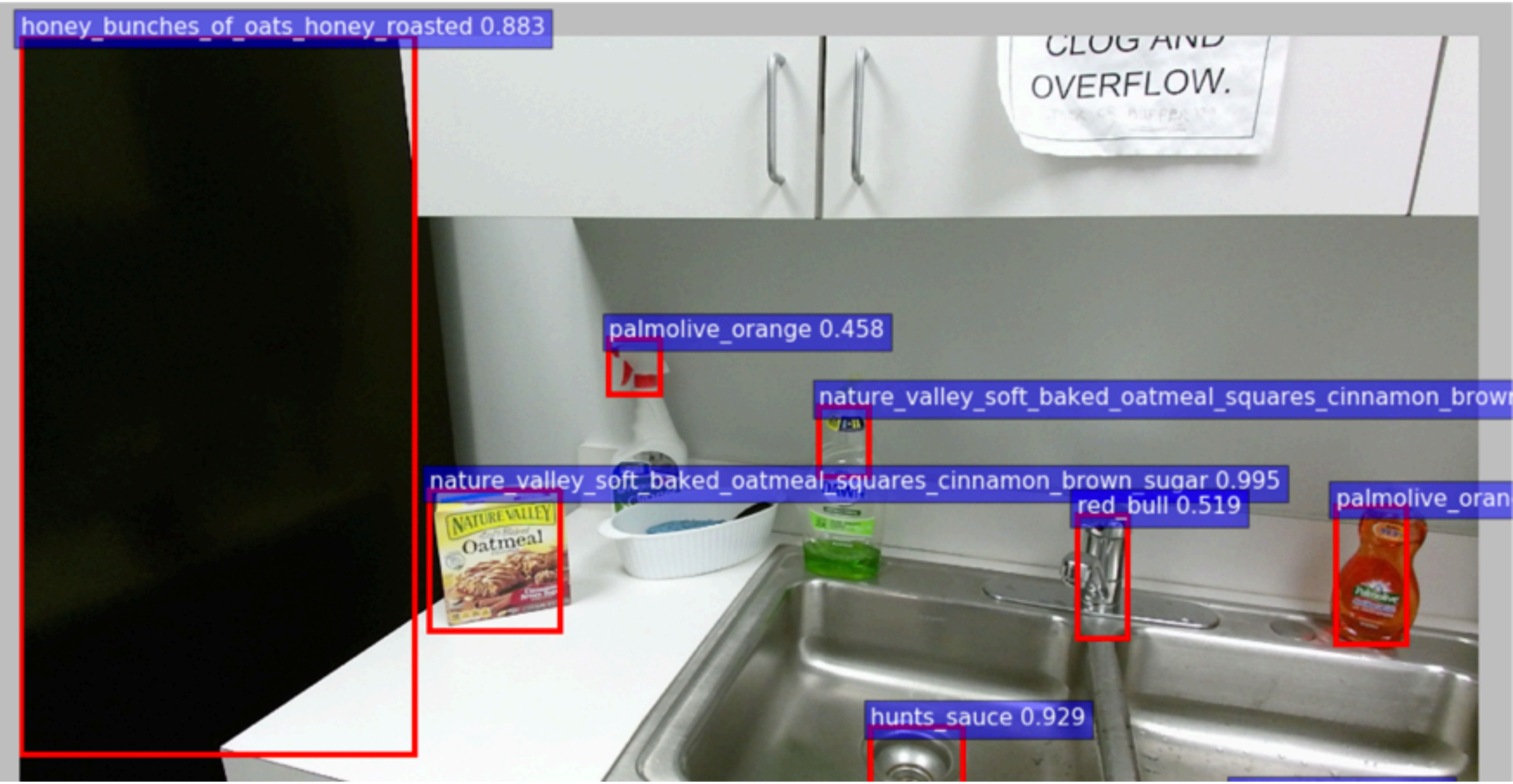
Patch Realism

Domain Adaptation



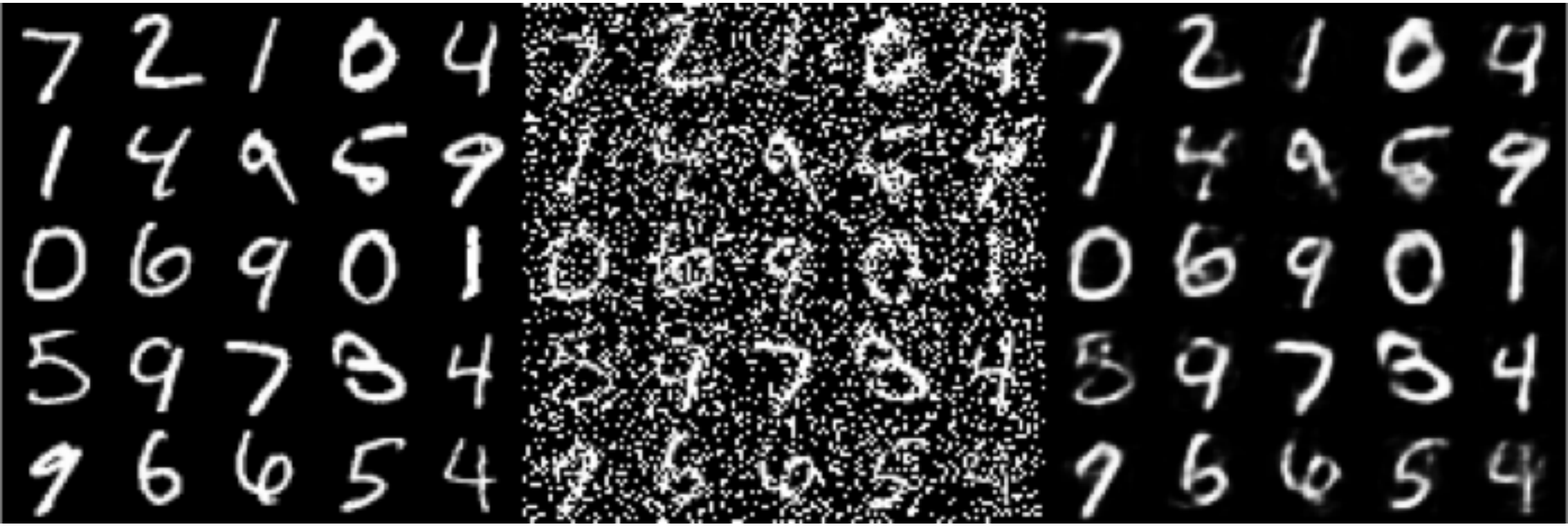
Will the neural network able to detect objects in real images if it trained on synthetic images?

Neural Networks Learn Artifacts Easily



Output of the object detector when trained naively

Noise Can Add Robustness



Raw Input Corrupted Input Reconstructed Input

Adding noise adds robustness to the auto-encoder at test-time

What sort of noise will be useful for our application?

Different Modes of Blending

No Blending

Gaussian Blurring

Poisson Blending

Various blending modes add robustness to the object detector

Dataset Diversity

Misses by a detector trained on hand-annotated scenes
These views were not present/labeled in the training set

Dataset Diversity

Ground Truth Images

Corresponding False Positives

False positives by detector trained on hand-annotated scenes

Proposed Solutions

Realism

- Paste real patches on real images

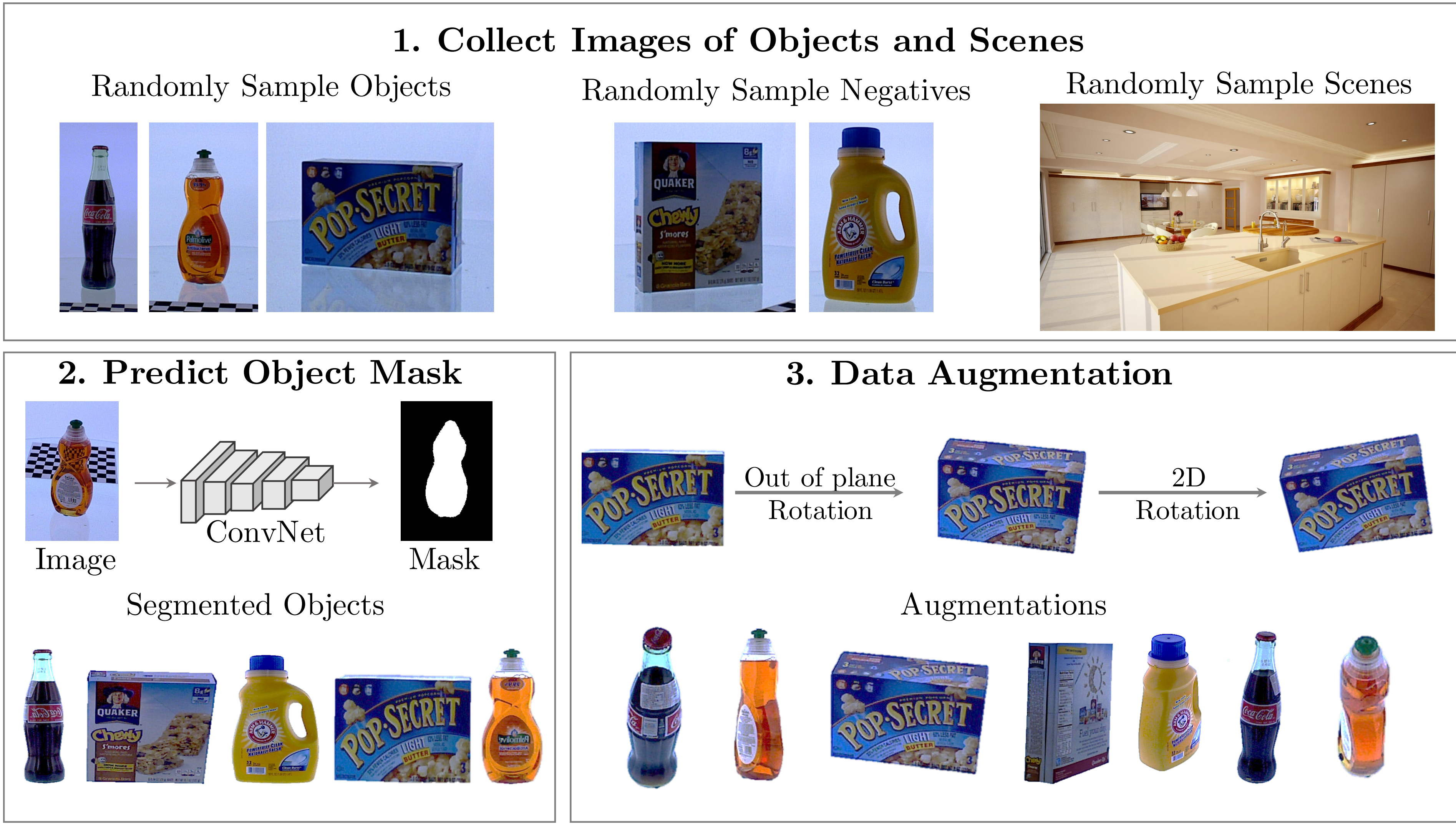
Domain Adaptation

- Add robustness by adding different blending modes for the same scene

Dataset Diversity

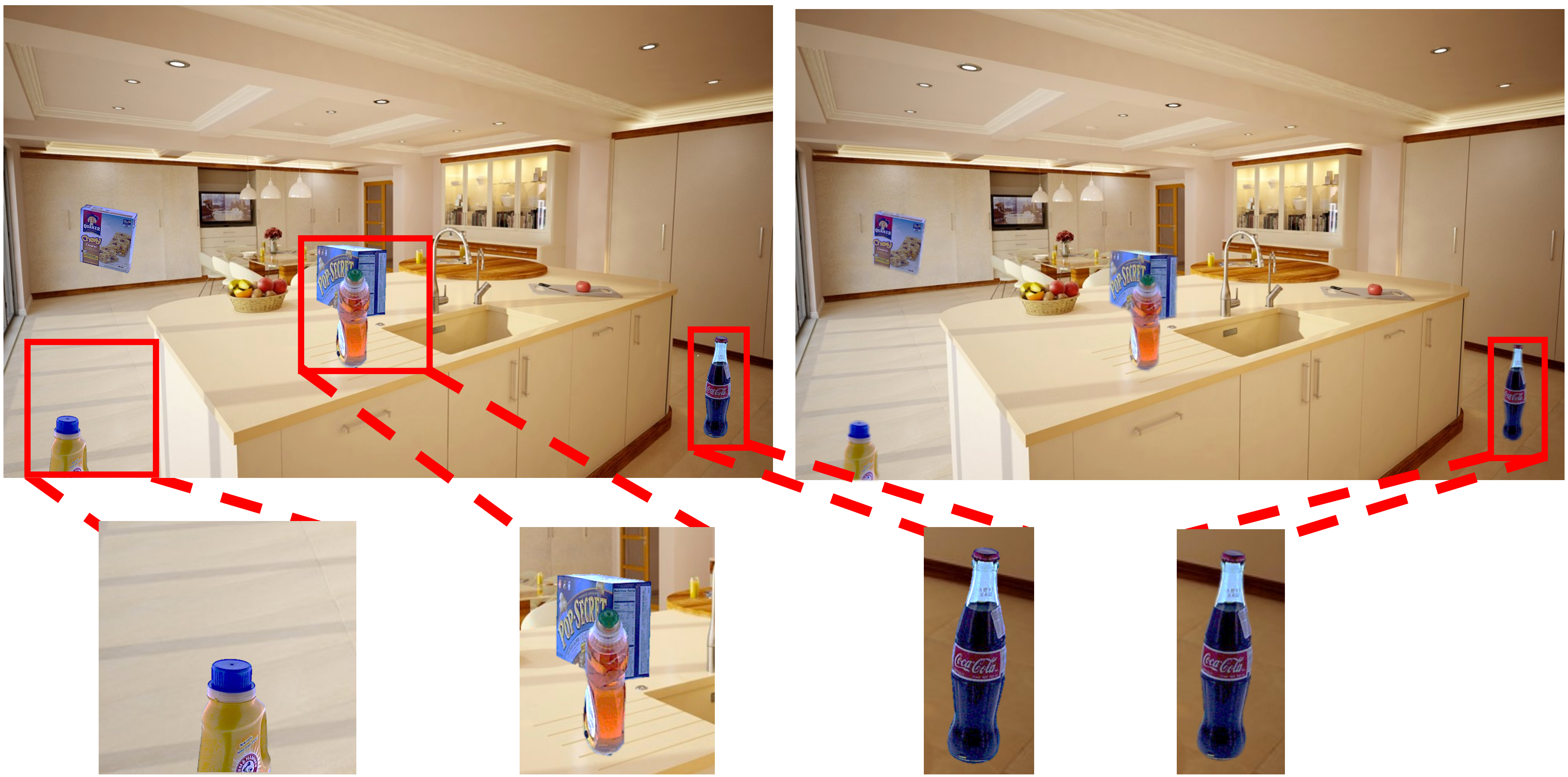
- Capture all views of an object and render adding different modes of data augmentation

Proposed Pipeline



Proposed Pipeline

4. Synthesize <u>Same</u> Scene with <u>Different</u> Blending Modes



Truncations | Occlusions | Different Blending Modes

Model real world scenarios | **Invariant to Local Artifacts**

Examples of Synthesized Images

Which synthesizing factors matter most?

Experimental Setup

Instance Images Dataset: (Big) Berkeley Instance Recognition Dataset

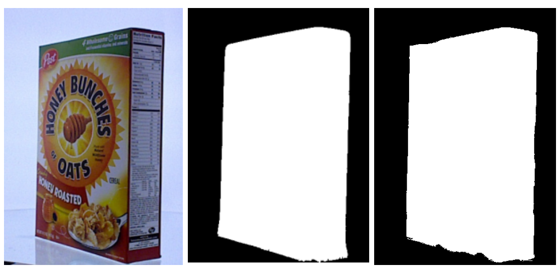
125 Instances, 600 viewpoints of each instances

Mask Generation

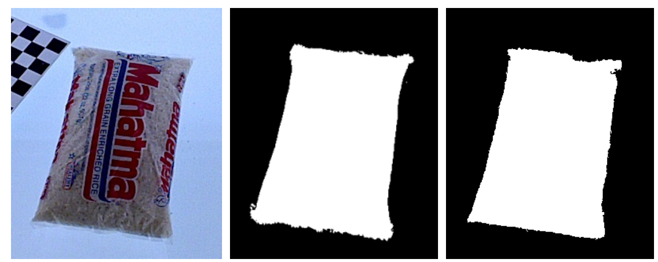
Fully Convolutional Network that predicts background/foreground pixels

Depth map used as proxy for foreground during training

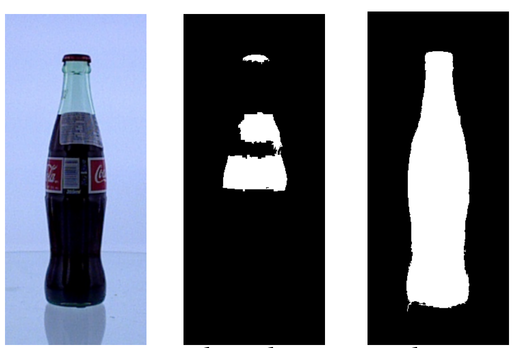
Image **Depth Mask** **Our Mask** **Image** **Depth Mask** **Our Mask**



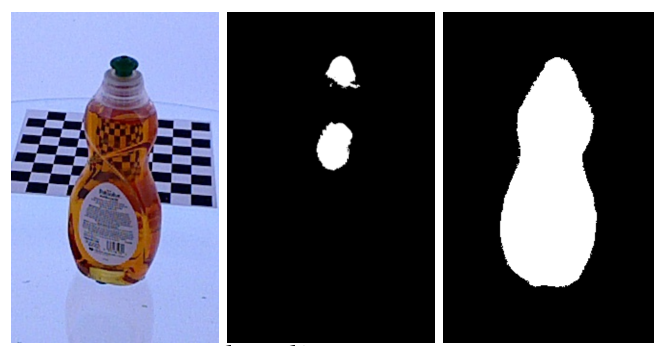
Honey Bunches of Oats



Mahatma Rice



Coca Cola Glass Bottle



Palmolive Orange

GMU Kitchen Scenes

11 Instances from BigBIRD

9 Kitchen Scenes

6,728 Annotated Frames for Evaluation

Effect of Blending

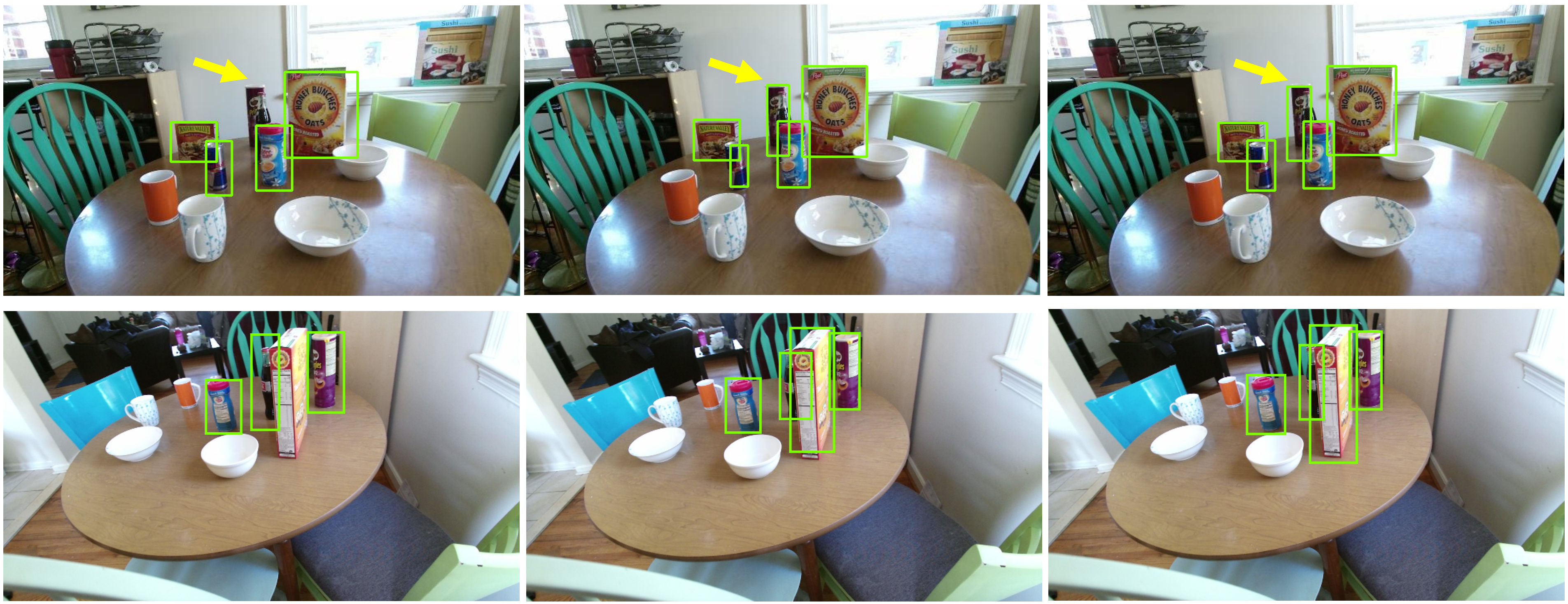
Blending Mode	mAP on GMU Dataset
No Blending	65.9
Gaussian Blending	68.9
Poisson Blending	58.4
All modes of Blending	72.4
All modes + Same Image	**73.7**

Effect of Data Augmentation

Data Augmentation	**mAP on GMU Dataset**
Base Model	73.7
w/o 2D Rotation	69.7
w/o 3D Rotation	68.3
w/o Truncation	71.8
w/o Occlusion	63.1
w Distractor Objects	**76.2**

Results on GMU Kitchen Scenes

Real Data | Synthetic Data | Synthetic + Real Data



1st Row: Synthetic data recognizes occluded instance

2nd Row: Synthetic data detects cereal box in spite of viewpoint change

How do synthetic images compare with real images?

Results on GMU Kitchen Scenes

Dataset	mAP
Real Images from GMU	86.3
Semantic-and-Geometry Aware Synthesis	51.7
Synthetic Images (Ours)	76.2
Semantic-and-Geometry Aware Synthesis + Real	85.0
Synthetic Images (Ours) + Real Images	**88.8**

Active Vision Dataset

6 Instances from GMU Kitchen Scenes

9 Kitchen Scenes, 17,556 Annotated Frames for Evaluation

Instances are usually more difficult to detect as compared to GMU

Can evaluate model trained on real images from GMU Scenes

Results on Active Vision Dataset

Real Data | Synthetic Data | Synthetic + Real Data

1st Row: Synthetic data doesn't throw false positives

2nd Row: Synthetic data detects objects at very small scales also

Results on
Active Vision Dataset

Dataset	mAP
Real Images from GMU	41.9
Synthetic Images	36.5
Synthetic Images + Real Images	**51.1**

Results on Active Vision Dataset

Dataset	mAP
10% Real Images	15.8
10% Real Images + Synthetic Images	43.2
40% Real Images	38.2
40% Real Images + Synthetic Images	50.2
70% Real Images	39.4
70% Real Images + Synthetic Images	**50.6**

Synthetic data captures information complementary to the real images

SUMMARY

Manual effort involved in creating annotated datasets can be reduced significantly

VIDEOS

- Videos to propagate labels from one frame to the next

3D RECONSTRUCTION

- 3D Reconstruction allows us to get pose and bounding box annotations automatically

SYNTHESIZING SCENES

- Instead of chasing global realism, we use noise and data augmentation effectively to build robust detectors

ACKNOWLEDGEMENTS

Ishan Misra

Georgios Georgakis (GMU)

Phil Ammirato (UNC)

Junjue Wang (ELIJAH)

Mahadev Satyanarayanan (ELIJAH)

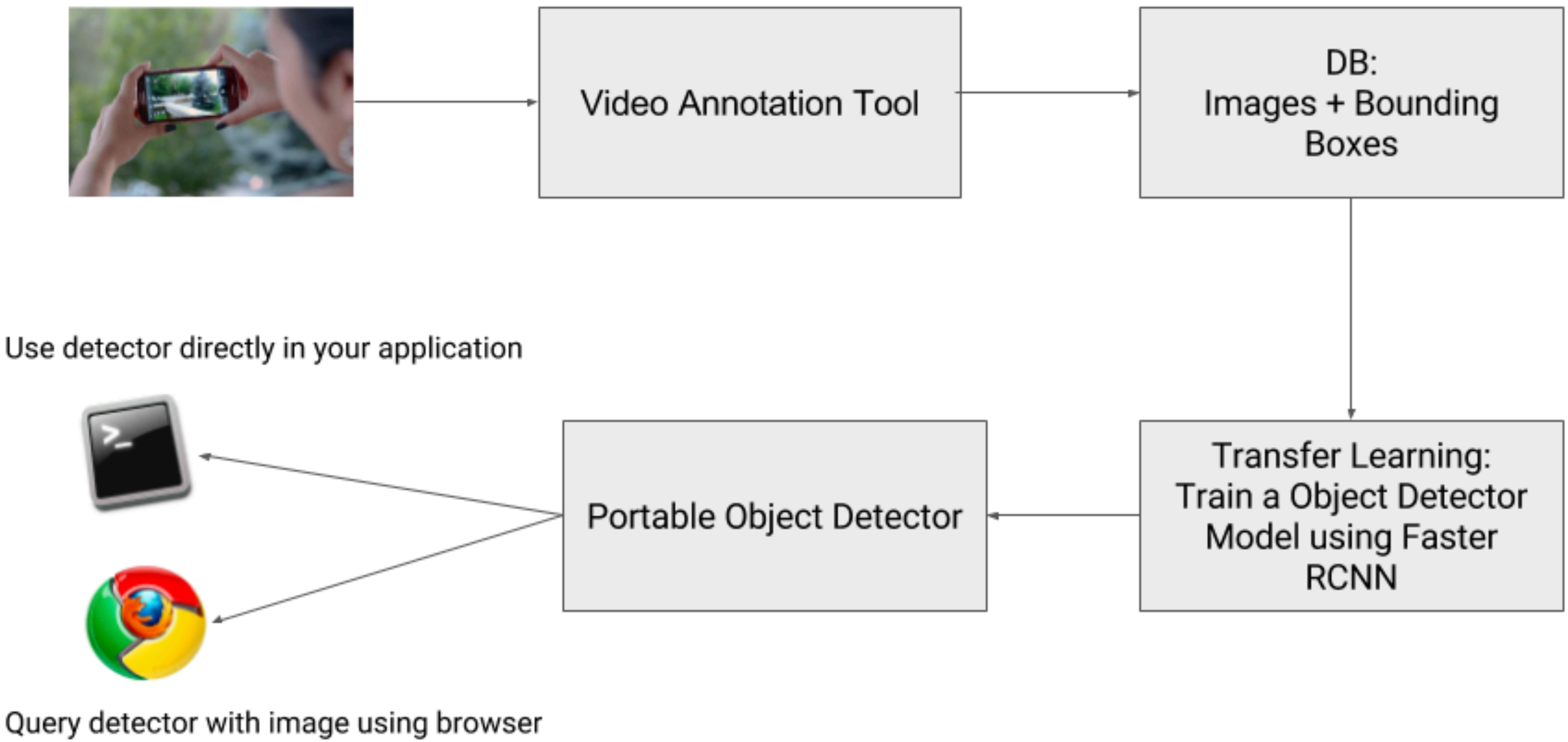
Michael Kaess

Martial Hebert

THANKS!

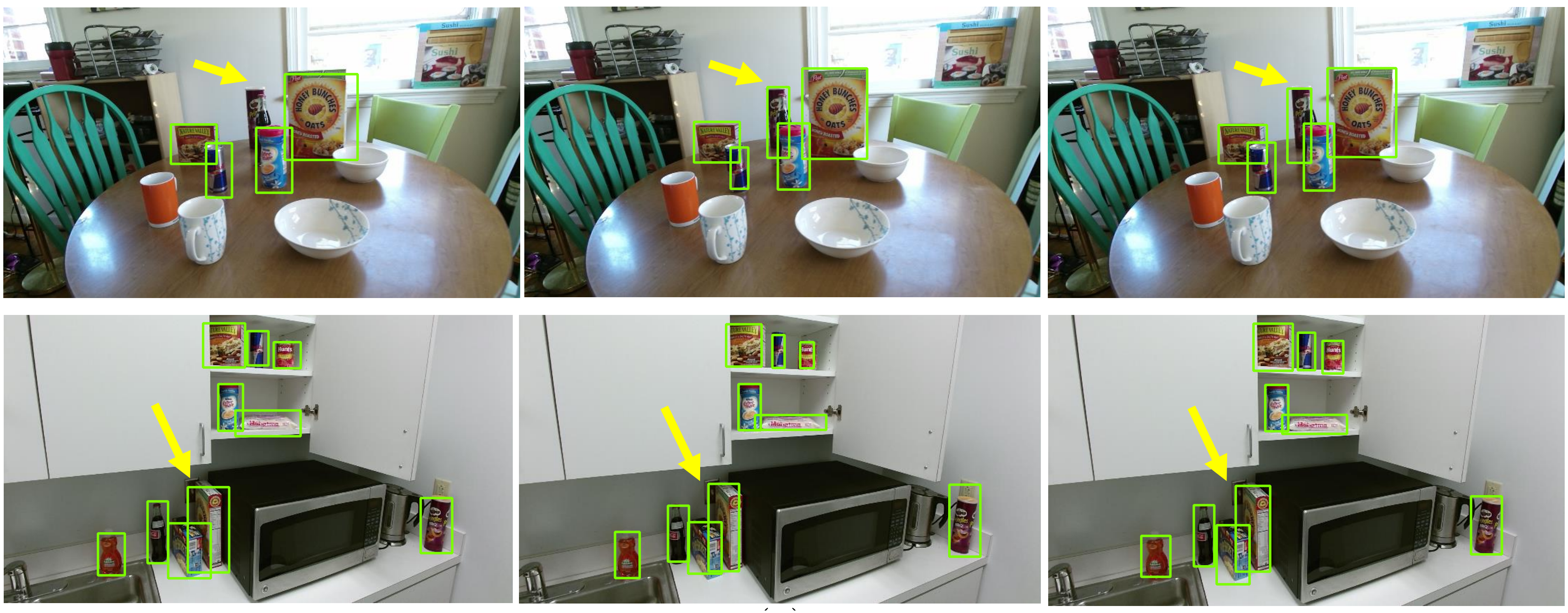
Questions?

Object Detector Pipeline



Results on GMU Kitchen Scenes

Real Data | Synthetic Data | Synthetic + Real Data



1st Row: Synthetic data

Annotation Results

Azimuth = 11

Azimuth = 48

Azimuth = 77

Azimuth = 105

Azimuth = 130

Azimuth = 175

Challenges

Realism

- Don’t training images have to look realistic?

Domain Adaptation

- Models trained on synthetic data don’t work as well on real images

Dataset Bias

- Lack of diversity in training images due to unconscious bias in creating datasets