URAI 2017
14th International Conference on Ubiquitous Robots and Ambient Intelligence
June 28 - July 1, 2017 / Maison Glad Jeju, Jeju, Korea

Synthetic Learning Set for Object Pose Estimation:
Initial Experiments

JOO-HAENG LEE, WOO-HAN YUN, JAEYEON LEE, JAEHONG KIM
HUMAN-MACHINE INTERACTION LAB
ETRI

Outline

GIVEN:

- **Target objects** for **pose estimation** in robotic manipulation task
- A small set of real images for training

PROBLEM:

- To obtain a **data set** with **quatatative diversity** in *condition* and **qualitative precision** in *annotation*.
- While considering **constraints** in human *labor* and *error*, production *cost*, physical limits in *environments*, and deploy *time*.

PROPOSED SOLUTION:

- To synthesize a learning set using **computer graphics**: SLS (Synthetic Learning Set)
- **Diversity**: pose (translation+rotation), illumination, …
- **Precise** and **rich** annotation: 3D coord, normal, depth, camera extrinsic, …

RESULTS

- Synthesis pipeline based on open source tools: ***Blender***
- **Synthetic Learning Set** (SLS): 516,672 images = 23 objects * 5,616 viewpoints * 4 types (RGB, xyz, normal, depth)
- **Initial experiments** of applying SLS on pose estimation
- Soon open at **github**: *https://github.com/joohaeng/SLS4ML*

joohaeng@etri.re.kr

Background

- Object **pose estimation** for robotic manipulation.
  - Diverse **objects**: shape, physical property,
  - Complex and dynamic **environments**
  - Machine learning requires a nice **data set** with **quality annotation** and/or **applicable in simulation**

Amazon Robotics Challenge 2016— Winner Team & Contest Objects

joohaeng@etri.re.kr

Examples of Acquiring a **Real Learning Set**

A subset of
APC 2015 objects

Manual acquisition

Labor
Cost
Error
Precision
Diversity
…

joohaeng@etri.re.kr

Proposed Solution: Synthetic Learning Set

Target Objects

Data Set:
Images + Annotations

Object Identification

Pose Estimation

Other Tasks

joohaeng@etri.re.kr

Proposed Solution: Synthetic Learning Set

Target Objects

Physical imaging

Control Param.
Pose, light, configuration, ..

Real Acquisition

(Ex) Physcal HW
Scanner

Annotation/Refinement

RGB + various tags

Real Learning Set

Data Set:
Images + Annotations

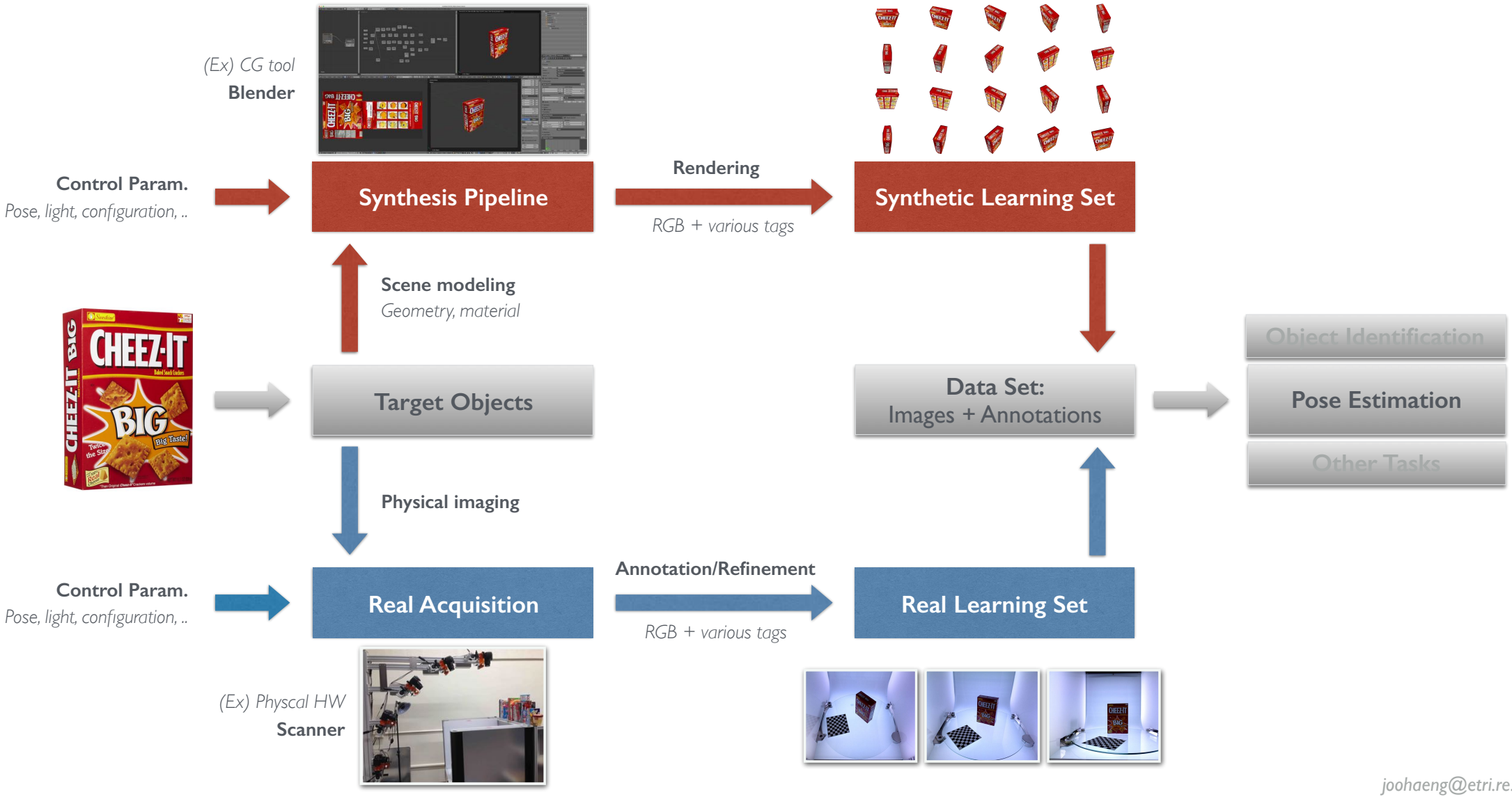
Object Identification

Pose Estimation

Other Tasks

joohaeng@etri.re.kr

Proposed Solution: Synthetic Learning Set



joohaeng@etri.re.kr

Related Work

- **A list of synthetic dataset and tools for computer vision**
  - *https://github.com/unrealcv/synthetic-computer-vision*

joohaeng@etri.re.kr

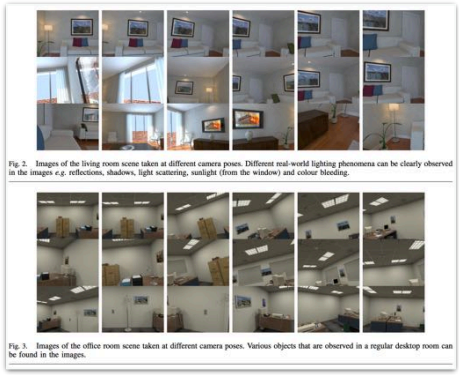
Related Work

- **A list of synthetic dataset and tools for computer vision**
  - https://github.com/unrealcv/synthetic-computer-vision

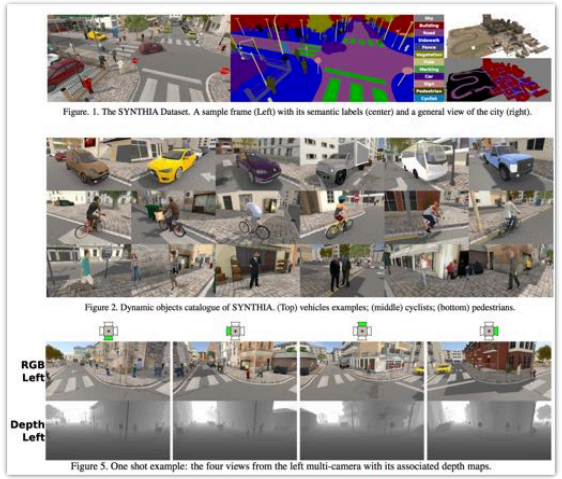
Synthetic image dataset
- SURREAL
- Virtual KITTI
- Synthia
- Sintel, A synthetic dataset for optical flow
- SceneFlow
- 4D Light Fields
- ICL-NUIM dataset
- Driving in the Matrix

SURREAL
(CVPR 2017)
Human motion

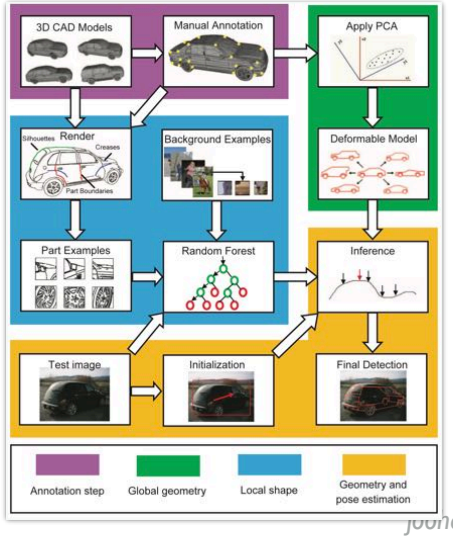
ICL-NUIM
(ICRA 2014)
SLAM



Synthia
(CVPR 2016)
Urban scene



Sintel
(PAMI 2013)
3D Pose



joonaeng@etri.re.kr

Related Work

- **A list of synthetic dataset and tools for computer vision**
  - *https://github.com/unrealcv/synthetic-computer-vision*

Synthetic image dataset
- SURREAL
- Virtual KITTI
- Synthia
- Sintel, A synthetic dataset for optical flow
- SceneFlow
- 4D Light Fields
- ICL-NUIM dataset
- Driving in the Matrix

Tools
- Render SMPL human bodies on Blender, see CVPR2017
- Render for CNN, based on Blender, see ICCV2015
- UETorch, based on UE4, see ICML2016
- UnrealCV, based on UE4, see ArXiv
- VizDoom, based on Doom, see ArXiv
- OpenAI Universe, see project page
- Blender addon for 4D light field rendering, see project page

Resources

Virtual/Augmented Reality for Visual Artificial Intelligence (ECCV 2016 Workshop)
Role of Simulation in Computer Vision (ICCV 2017 Workshop)
Virtual Reality Meets Physical Reality (Siggraph Asia 2016 workshop)
CVPR 2017 Workshop THOR Challenge

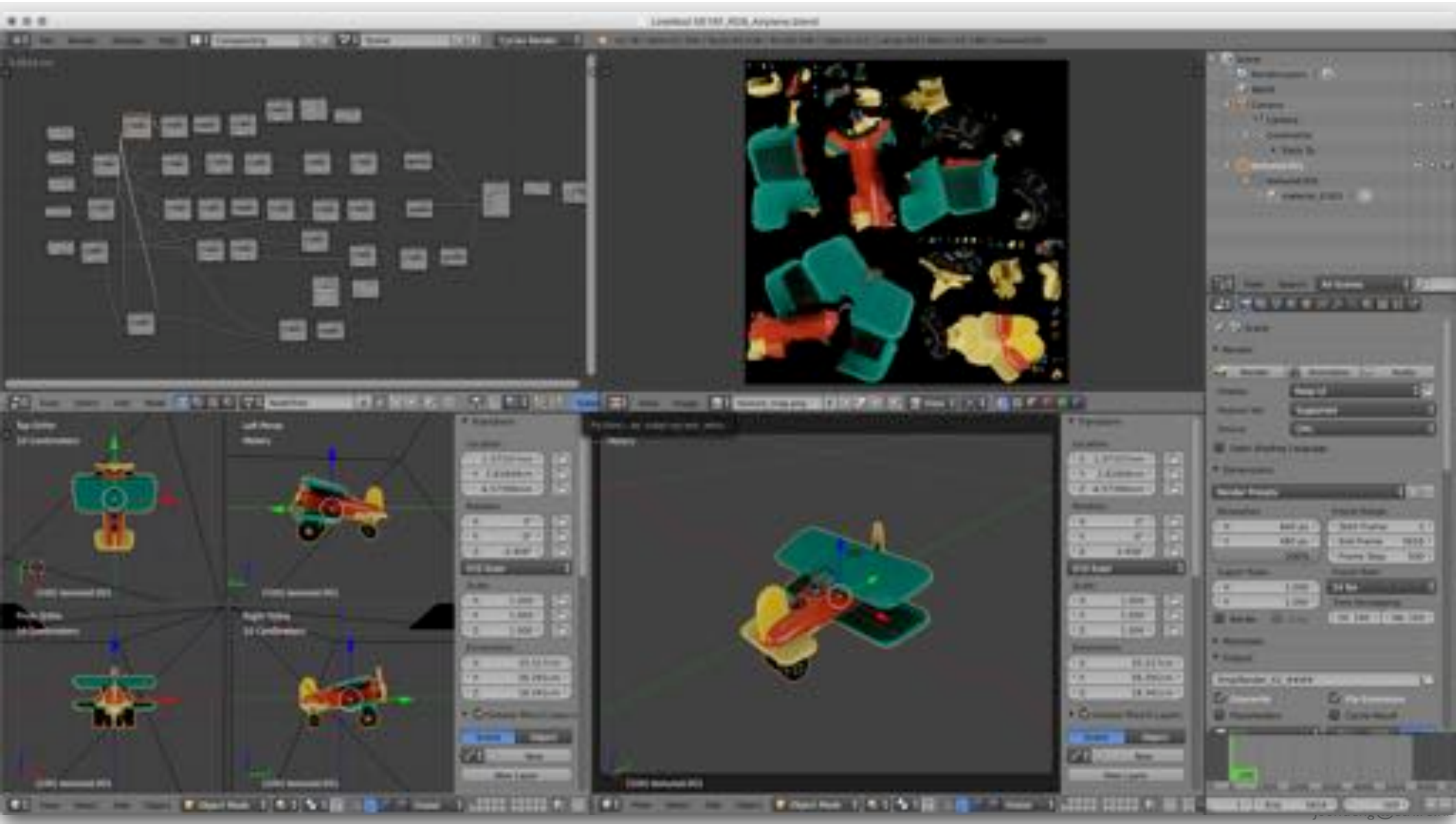
3D Model Repository
- ShapeNet
- 3dscan
- seeing3Dchairs

joohaeng@etri.re.kr

Synthesis Pipeline

- **Blender**
  - … **free** and **open** source 3D creation suite.
  - It supports the **entirety of the 3D pipeline**— modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation.

joohaeng@etri.re.kr



Scene Modeling for SLS

- **Objects**
  - Geometry, material (RGB texture, BRDF), deformation, …
- **Camera**
  - Pose, intrinsics, …
- **Light**
  - Env map lighting

joohaeng@etri.re.kr

Object Modeling

- Manual modeling
  - **Cuboid**
  - Free-form
- Automatic
  - 3D scanner
- Open data
  - **YCB data set**

joohaeng@etri.re.kr

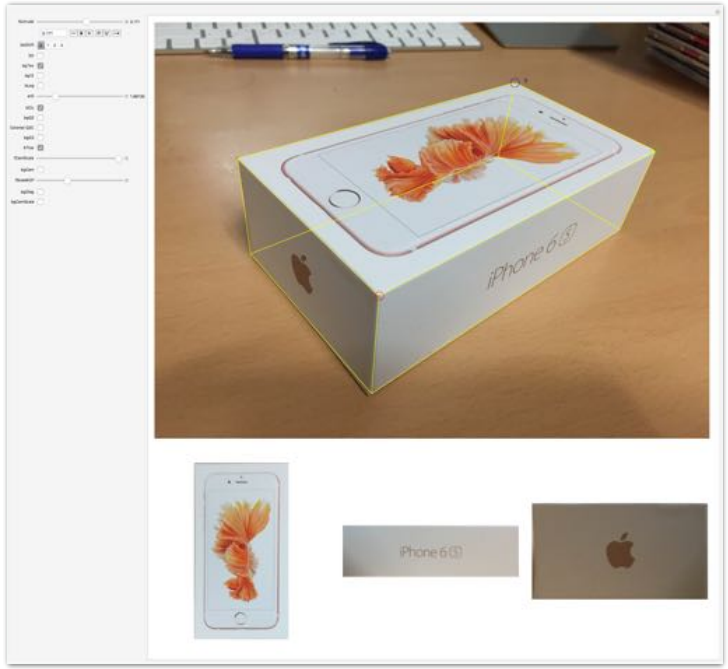
Cuboid-like Objects

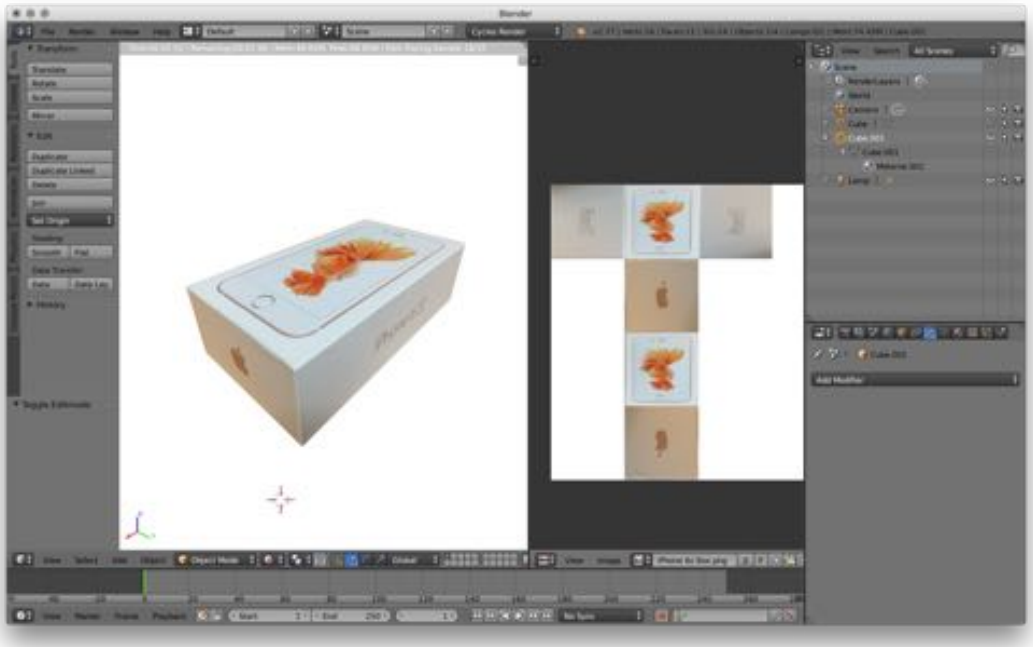
9 objects from
APC 2015

5 objects from
Korean grocery

joohaeng@etri.re.kr

Modeling — *Cuboids*

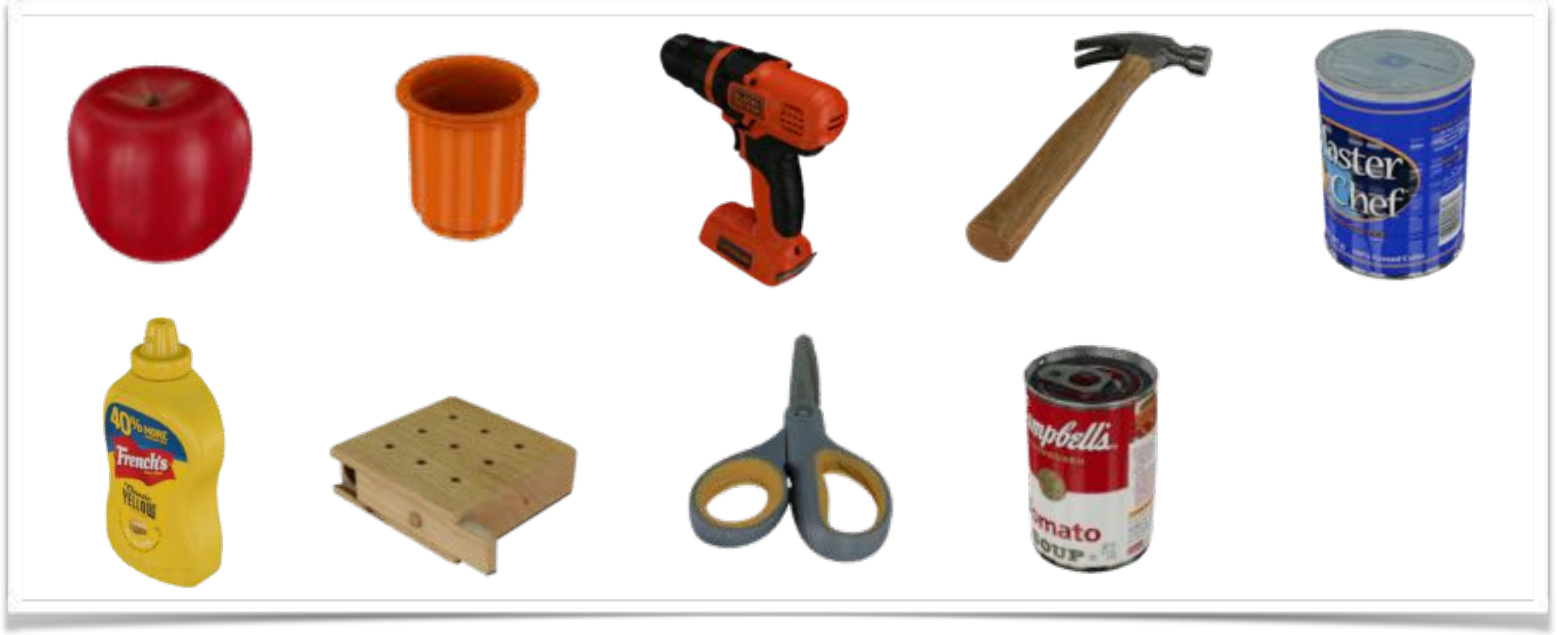




joohaeng@etri.re.kr

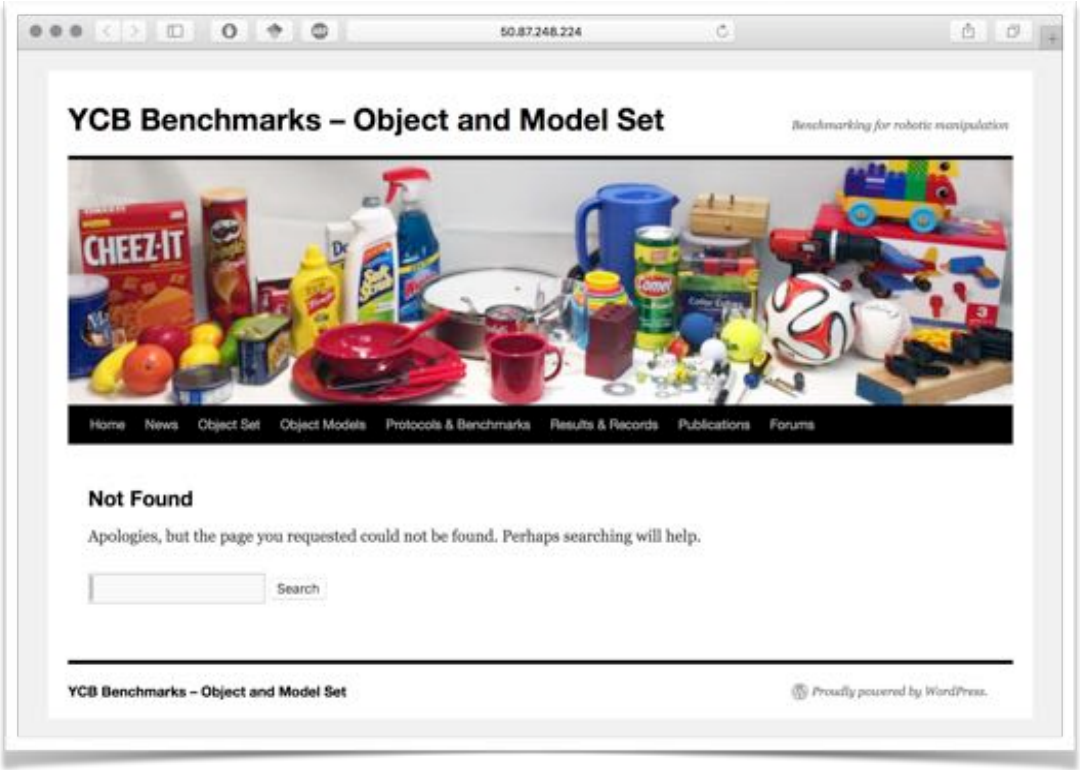
Free-form Objects

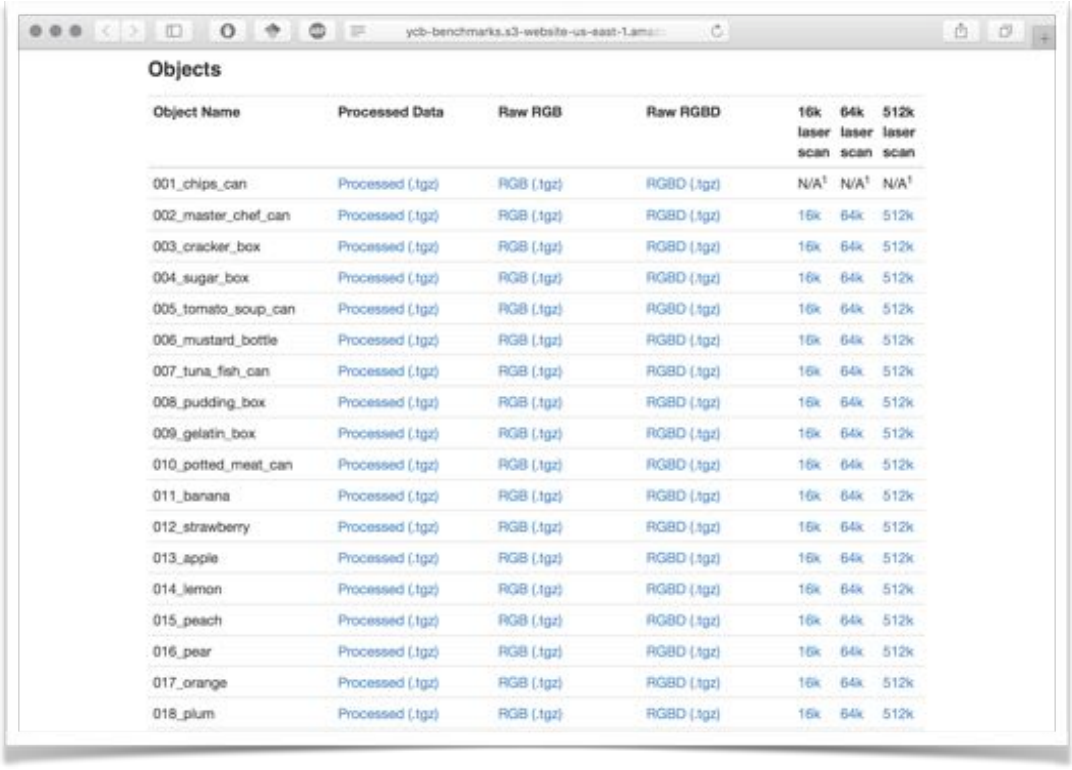
9 objects from
YCB Benchmark



joohaeng@etri.re.kr

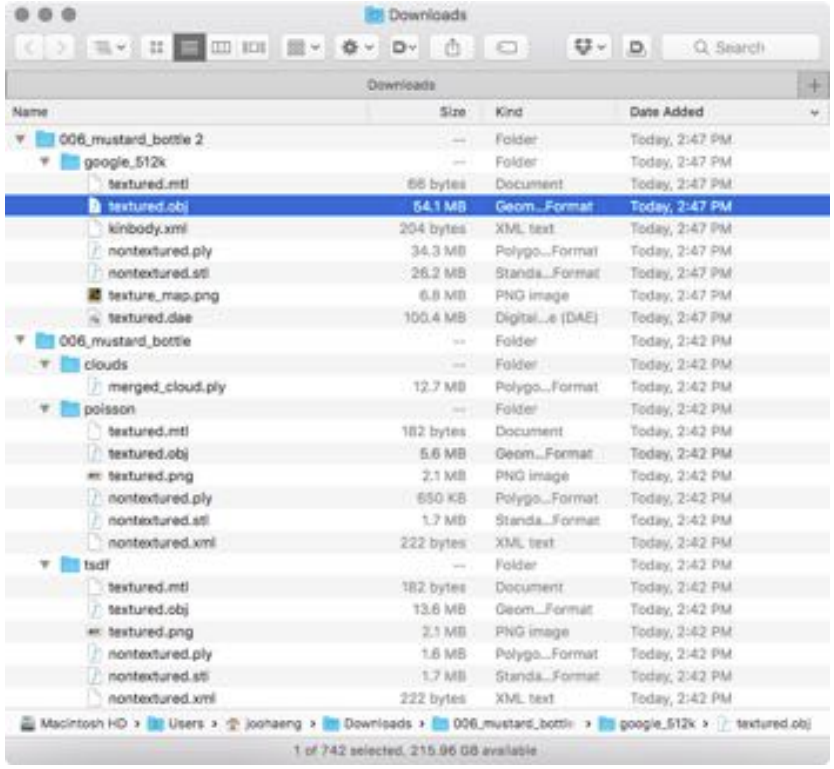
YCB Benchmark Dataset

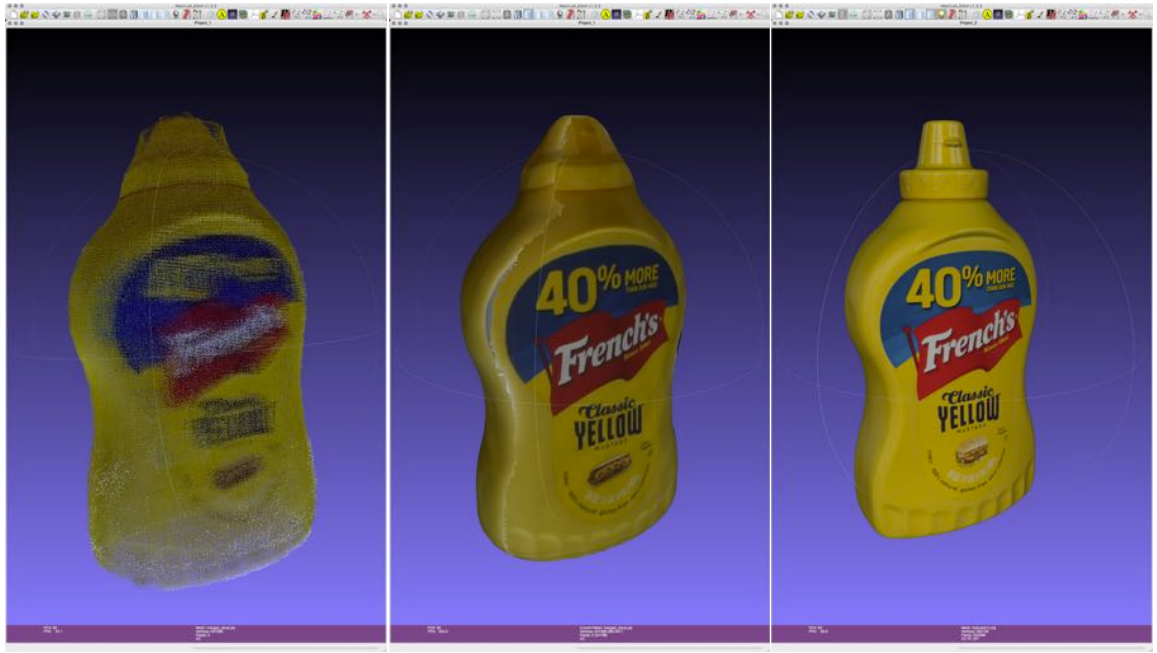




joohaeng@etri.re.kr

YCB Benchmark Dataset

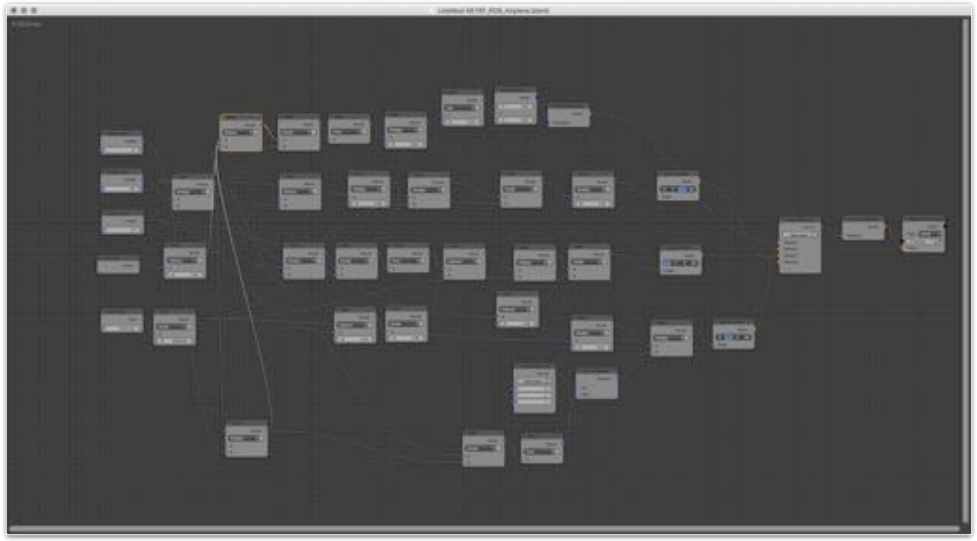


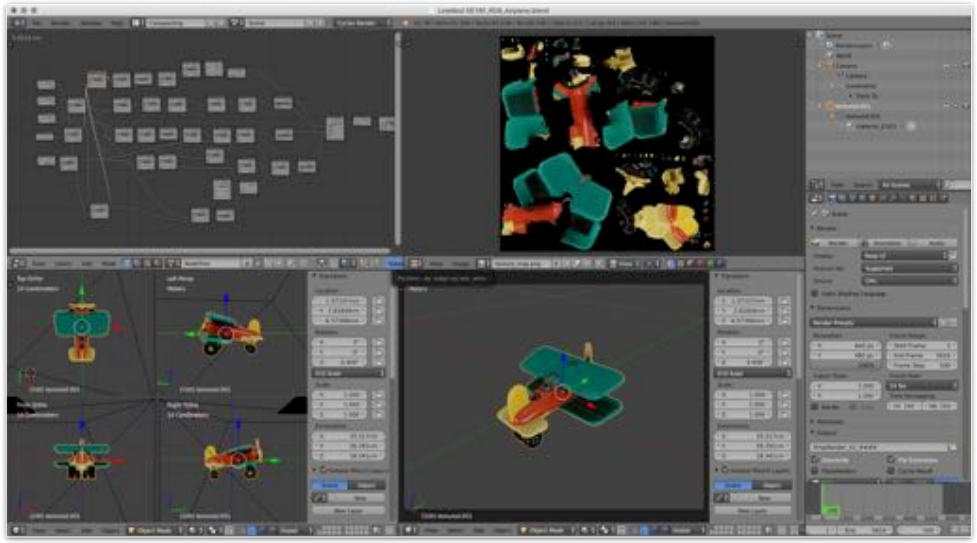


joohaeng@etri.re.kr

Camera

- **5,616 poses**
  - **Pan**: 24 samples in [0,2Pi]
  - **Tilt**: 13 samples in [-Pi, Pi]
  - **Yaw**: 3 samples in [-Pi/4,Pi/4]
  - **Zoom**: 6 samples in [0.5, 2] m

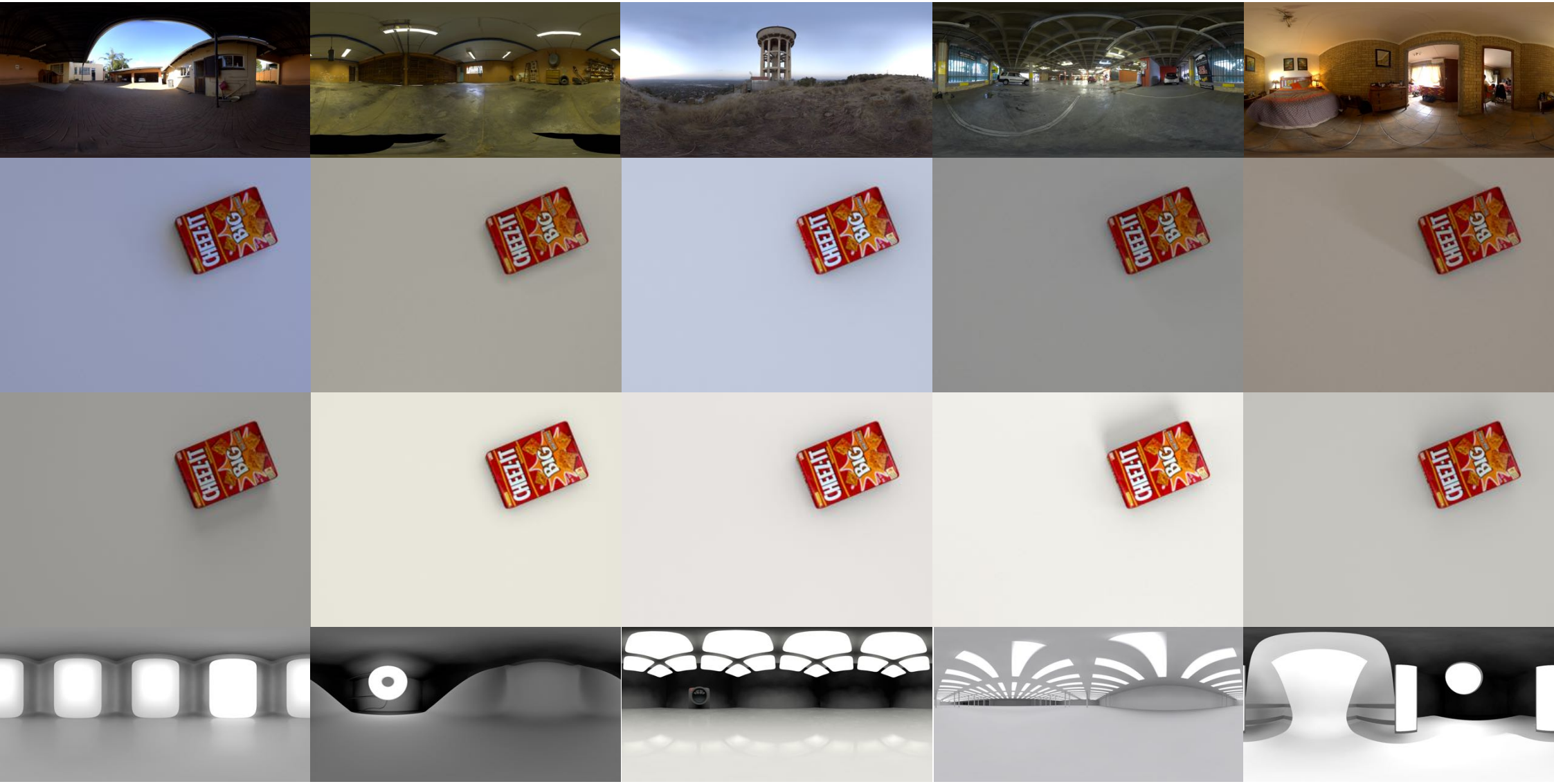




joohaeng@etri.re.kr

Lighting

joohaeng@etri.re.kr



이주행 **Joo-Haeng Lee** joohaeng at etri dot re dot kr

Rendering

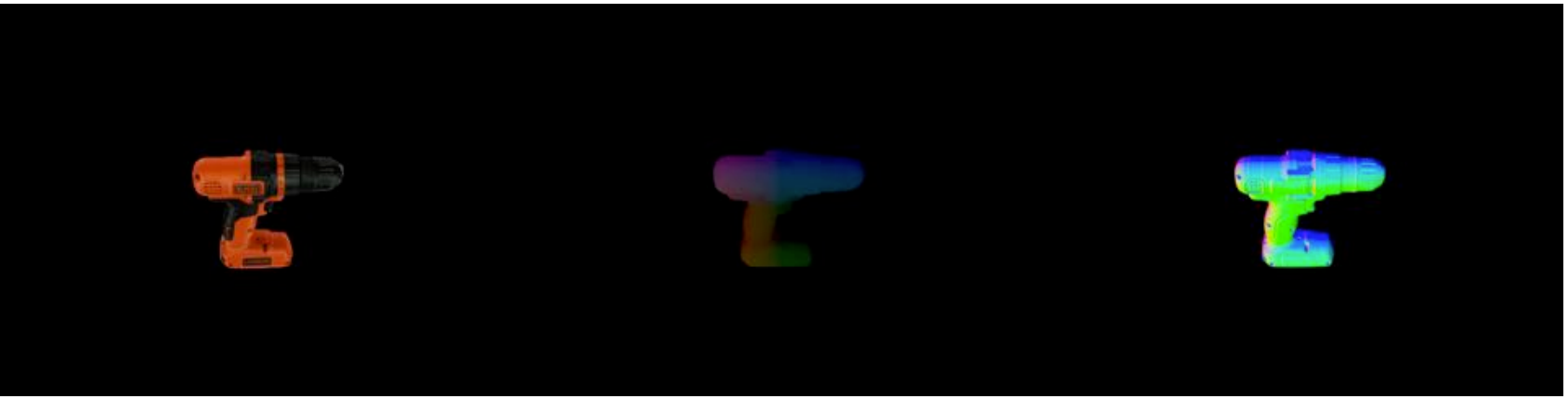
- **RGB** + **XYZ** + **Depth** + **Normal** for each camera pose
  - Mask is encoded in alpha channel (A).
  - Other types are rendered using **special shaders**
  - Pixel-wise correspondence between RGB and annotations

RGB (.PNG)

XYZ (.EXR)

Normal (.EXR)

Depth (.EXR)

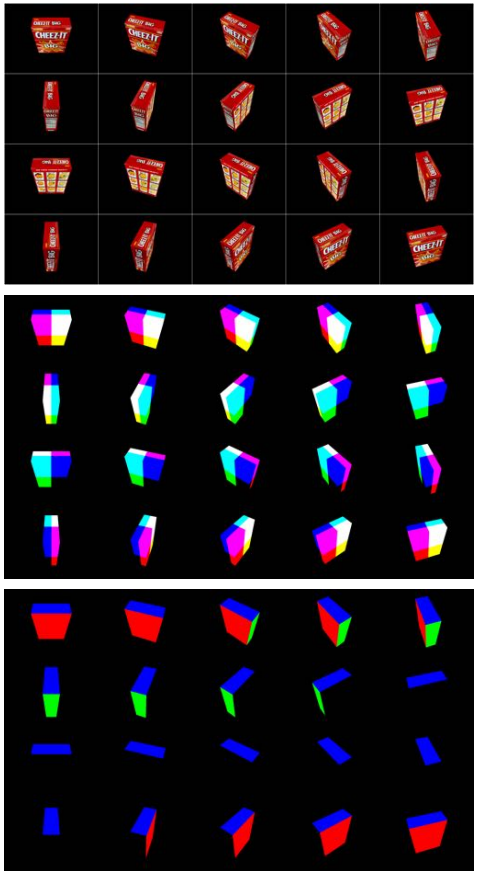


Application in Pose Estimation

joohaeng@etri.re.kr

Application in Pose Estimation

Synthetic Learning Set



Feature matching
Specialized for P*n*P

Feature DB with object ID and annotations

SURF	Obj ID	XYZ	Normal
#1..#1	1	(10,12,13)	(0,0,1)
#1..#2	1	(-1,12,13)	(1,0,1)
#2..#1	2	(10,-8,0)	(0,1,1)
#2..#1	2	(3,12,13)	(0,0,-1)

joohaeng@etri.re.kr


AGAST
200 pt 0.007759 sec
AKAZE
200 pt 0.051365 sec
BRISK
200 pt 0.340431 sec
FAST
200 pt 0.006808 sec
KAZE
200 pt 0.146259 sec
ORB
200 pt 0.027921 sec
SURF
200 pt 0.041335 sec
AGAST
200 pt 0.004181 sec
AKAZE
79 pt 0.011322 sec
BRISK
200 pt 0.219661 sec
FAST
200 pt 0.00355 sec
KAZE
200 pt 0.04895 sec
ORB
200 pt 0.018334 sec
SURF
57 pt 0.014781 sec
AGAST
200 pt 0.004728 sec
AKAZE
133 pt 0.018193 sec
BRISK
200 pt 0.220954 sec
FAST
200 pt 0.003711 sec
KAZE
200 pt 0.069875 sec
ORB
200 pt 0.021221 sec
SURF
76 pt 0.015323 sec
AGAST
200 pt 0.005781 sec
AKAZE
47 pt 0.030463 sec
BRISK
200 pt 0.290269 sec
FAST
200 pt 0.005234 sec
KAZE
124 pt 0.081064 sec
ORB
200 pt 0.020441 sec
SURF
29 pt 0.016421 sec

Application in Pose Estimation

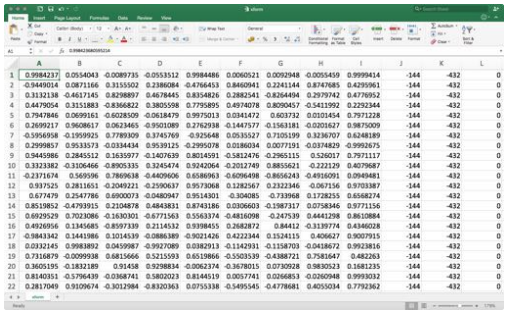
Synthetic *Verification* Set

Feature matching
Specialized for P*n*P

Feature DB with object ID and annotations

SURF	Obj ID	XYZ	Normal
#1..#1	1	(10,12,13)	(0,0,1)
#1..#2	1	(-1,12,13)	(1,0,1)
#2..#1	2	(10,-8,0)	(0,1,1)
#2..#1	2	(3,12,13)	(0,0,-1)

Pose estimation
Solve P*n*P



Verification with ref.
$X = [R|t]$

joohaeng@etri.re.kr

Application in Pose Estimation

Online Test

Feature matching
Specialized for P*n*P

Feature DB with object ID and annotations

SURF	Obj ID	XYZ	Normal
#1..#1	1	(10,12,13)	(0,0,1)
#1..#2	1	(-1,12,13)	(1,0,1)
#2..#1	2	(10,-8,0)	(0,1,1)
#2..#1	2	(3,12,13)	(0,0,-1)

Pose estimation
Solve P*n*P

joohaeng@etri.re.kr


720
Kinect v2:
프로그램 화면
Sony Cyber-shot:
외부 카메라

Application in Pose Estimation

ETRI Electronics and Telecommunications Research Institute

Object recognition and pose estimation for modular manipulation system: overview and initial results

Woo-han Yun, Jaeyeon Lee, Joo-Haeng Lee, and Jaehong Kim
HMI Research Group, ETRI

Introduction

- Task adaptaion of robots becomes an important factor. Modular manipulation system could be a solution.
- In modular manipulation system, recognition engine gives object instance ID, object location and pose information.
- Local feature based approach is selected because of its simplicity and small training dataset requirement.
- We sketch the overview of the process and show initial results using 10 cuboid-type object instances.

(Top) 10 cuboid-type objects
(Bottom) Depth-based background subtraction

Object recognition engine

- Data generation
  - Select 10 cuboid-type objects.
  - Scanning six surfaces and registering them.
  - 2D synthetic images are generated at intervals of 18 degree in yaw, roll, and pitch.
  - On synthetic images, extract local feature, SURF, and store two positional information (2D position in image – 3D position in object-centric coordinate)
- Depth-based background subtraction.
  - To reduce local features in noisy background and extract feature points on the foreground.
  - Depth data in each pixel is modeled by unimodal Gaussian distribution.
  - After subtraction, apply post-processing such as removing small blobs.

Object recognition engine

- Local feature matching
  - Extract SURF feature in query image.
  - Find the correspondence between features in query image and stored database.
    - Caluclate Euclidean distance for each pair.
    - Select distinctive correspondence (D1/D2 < 0.65 or D1 < 0.25).
  - Features having same 3D position are considered as one feature having minimum distance.
- PnP problem solving
  - After matching, distinctive correspondeces (2D position in image coor. - 3D position in object-centric coor.) are obtained.
  - With these correspondences, find the final objet instance and pose estimation by solving PnP problem with RANSAC.

Input query image (left) and its results (right)

Experimental result

- For train, 10 objects x 231 images were used.
- Object recognition test
  - 400 real images (1 or 2 objects in the shelf) were tested.
- Pose estimation test
  - 1,000 synthetic images are tested (for exact groundtruth).
  - Repeated texture pattern degraded the accuracy.

Object detection/recognition rate and rotation/translation error

Object	Detection rate(%)	Recognition rate(%)	Rotation error(°)	Translation error(mm)
cheezit	98	100	16.07	2.63
genuine	82	96	0.89	1.20
crayola	98	100	9.15	8.46
mark	82	94	1.76	0.82
expo	92	100	11.55	5.55
ace	96	100	3.07	1.94
chiffon	78	94	5.07	4.81
chococo	96	98	5.17	3.95
moncher	98	100	2.15	3.85
waffle	84	98	4.03	3.96
Average	90.60	98	5.89	3.72

URAI 2017, Jeju, Korea

joohaeng@etri.re.kr

Application in Object Detection

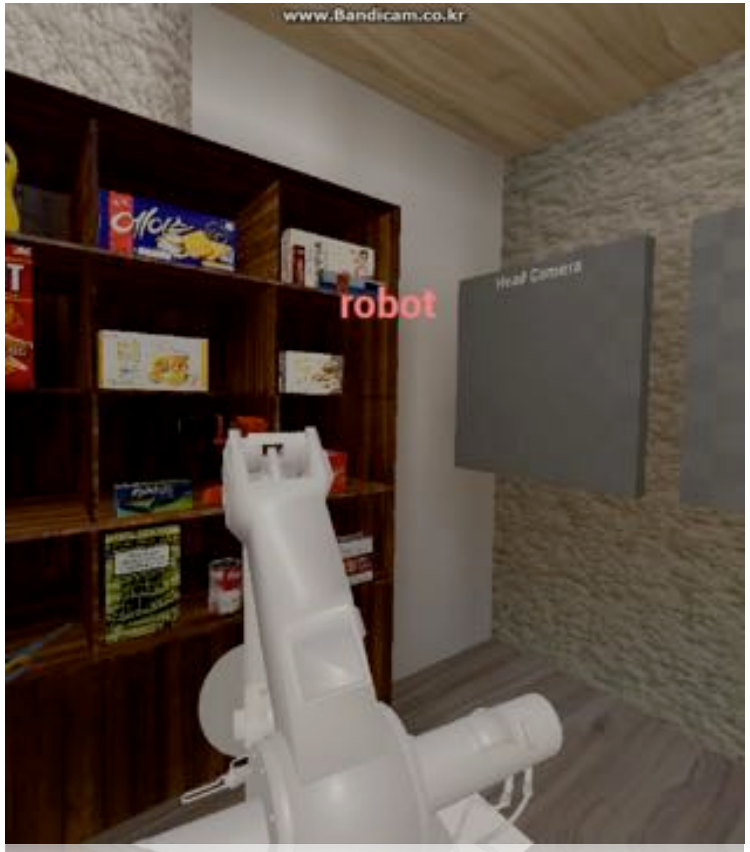
- Toward **end-to-end network** for pose estimation **with SLS**
  - No explicit feature extraction, at least.

joohaeng@etri.re.kr

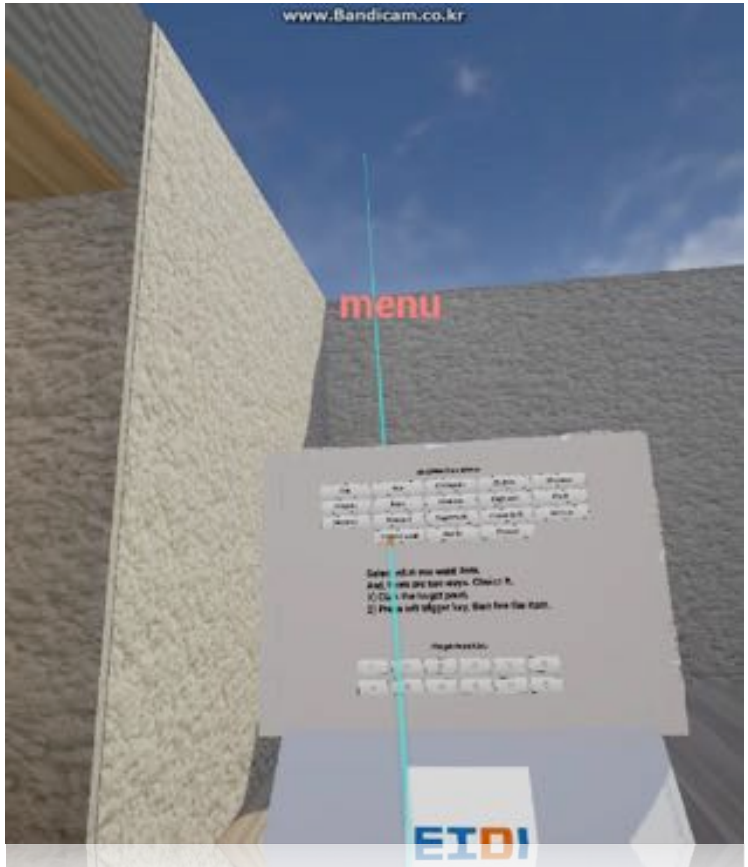
Ongoing Work — **Complex Environments**

Ongoing Work — VR Simulation (UE4 + HTC Vive)

"I'm your arm."



Interactive placement of objects



Camera-Projector simulation

joohaeng@etri.re.kr

Summary

GIVEN:

- **Target objects** for **pose estimation** in robotic manipulation task
- A small set of real images for training

PROBLEM:

- To obtain a **data set** with **quatatative diversity** in *condition* and **qualitative precision** in *annotation*.
- While considering **constraints** in human *labor* and *error*, production *cost*, physical limits in *environments*, and deploy *time*.

PROPOSED SOLUTION:

- To synthesize a learning set using **computer graphics**: SLS (Synthetic Learning Set)
- **Diversity**: pose (translation+rotation), illumination, …
- **Precise** and **rich** annotation: 3D coord, normal, depth, camera extrinsic, …

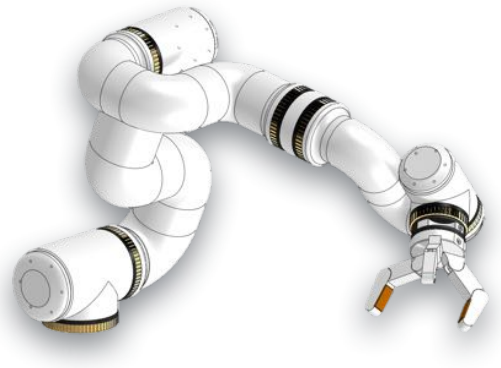
RESULTS

- Synthesis pipeline based on open source tools: ***Blender***
- **Synthetic Learning Set** (SLS): 516,672 images = 23 objects * 5,616 viewpoints * 4 types (RGB, xyz, normal, depth)
- **Initial experiments** of applying SLS on pose estimation
- Soon open at **github**: *https://github.com/joohaeng/SLS4ML*

joohaeng@etri.re.kr

Acknowledgement

This research was supported by the MOTIE under the Robot industry fusion core technology development project (10048920) *supervised by the KEIT.*



Development of **Mod**ular **Man**ipulation System Capable of Self-Reconfiguration of Control and Recognition System for Expansion of Robot Applicability

Q&A