# Dataset specifications



---

# Objects

Object parts (object that belongs to a parent object)
YCB objects + new non rigid, cloth, transparent, deformable, articulated...

---

# Attributes

(alphabetically ordered)

## Articulation (Kinematic structure) (ICL)

Estimation method: Clustering + Label Spreading
Representation: Input: RGB-D data sequences, Output: Kinematic structure
Link to code: https://github.com/ImperialCollegeLondon/3DKSL (both off-line and online methods)
Link to paper:
https://openaccess.thecvf.com/content_ICCV_2019/papers/Nunes_Online_Unsupervised_Learning_of_the_3D_Kinematic_Structure_of_Arbitrary_ICCV_2019_paper.pdf

## Category (ICL)

Estimation method: trained CNN

Representation 1: labels 1x20D
[the name of the object, e.g. bowl, glass, kettle]
Link to code 1 : NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB

Representation 2: labels 1x37D

- 'bottle', 'cylinder', 'food box', 'toaster', 'dice' [can be changed to ‘die’], 'can', 'fork', 'lego', 'pen', 'hammer', 'fruit', 'soda can ', 'clamp', 'blender', 'pill', 'spoon ', 'box', 'thermos', 'mug', 'drill', 'ball', 'pan', 'spoon', 'baking tray', 'knife', 'kettle', 'bowl', 'scissors', 'wineglass', 'wrench', 'spatula', 'coffee maker', 'pot', 'screwdriver', 'chopping board', 'plate', 'glass'
    - The objects “cylinder”, “dice”, and “pill” were added to take into account some of the soft objects. Note that there is a mix of actual object names and shapes. In the future, one may consider having e.g. a category “sponge” rather than the shape of it. For future use, it may be useful to rename “dice” to “die” and “pill” to “capsule”.
- Superset of YCB, SHOP-VRB, and soft objects in desktop environment (adding “die”)

Link to code 2 :
https://gitlab.fel.cvut.cz/body-schema/ipalm/ipalm-vir2020-object-category-from-image
Link to dataset 2: LINK

## Colour (ICL)

Estimation method: trained CNN

Representation 1: labels 1x12D
[gray, red, blue, green, brown, purple, cyan, yellow, white, metallic, transparent, black]
Link to code 1 : NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB
Representation 2: RGB 8x8x8D histogram

## Density (CTU)

Can be calculated from volume/size and weight.

## Elasticity / Elastic modulus (CTU)

Estimation method: Squeezing / pressing the object.
Action parameters: Gripper type, Finger configuration (if available), Squeezing speed
Input: Stress/strain relationship. Depending on gripper, can be current/force vs. gripper position/aperture.
Representation 1: Elastic modulus / Young's modulus.

- Or, Elastic moduli at different points on the stress/strain response curve. E.g. elastic modulus at 40% of object compression - for soft objects.
- Units: Pascals.
- Requirements: Calibrated gripper to get absolute numbers. If not present, relative comparison of objects possible.

Representation 1.5: something in between 1 and 2 ;)
Representation 2: predefined categories - e.g. hard, slightly deformable, soft (these terms are not in line with the treatment of elasticity in physics)
Link to code: Gripper dependent. https://gitlab.fel.cvut.cz/body-schema/ipalm/ipalm-grasping
Link to other attributes: Stiffness.
Definition: Elasticity is a property of a material - stress (pressure - restorative internal force per unit area) / strain (relative deformation) characteristic. Contact surface area when pressing is thus taken into account. Young's modulus describes tensile elasticity.

## Friction

See Surface roughness.

## **Grasping techniques (?)** labels **(IRI, CTU, AAL)**

Estimation method: trained CNN
Representation: tuple: gripper type (4-5 ? ), grasp taxonomy (<15 discrete numbers), object type (?)
Link to code: Ganhand code (for human hand) and the subsequent work done by AAL-IRI-Naver (for Barrett gripper), PointNet GDP for YCB (Shubhan)

## Hardness

Similar but not identical to stiffness / elasticity.

## Material (IRI, ENPC, CTU)

1. From images
    - Examples of classification: smooth plastic, shiny plastic…
    - ICL for SHOP-VRB
        i. Estimation method: trained CNN
        ii. Representation: labels 1x6D [rubber,metal,plastic,wood,ceramic,glass]
        iii. Link to code: NS-VQA
        iv. Link to dataset: https://michaal94.github.io/SHOP-VRB
    - YCB, SHOP-VRB, soft objects (CTU)
        i. Estimation method: mask-RCNN (detectron2)
        ii. Representation: labels 1x9D: 'paper', 'wood', 'hard plastic', 'soft plastic', 'metal', 'ceramic', 'glass', 'foam', 'other'
        iii. Link to code 2 : https://gitlab.fel.cvut.cz/body-schema/ipalm/ipalm-vir2020-object-category-from-image (Note: currently, the network outputs only tuples: category - material)
        iv. Link to dataset 2: LINK
    - Other datasets for training:
        i. OpenSurfaces - http://opensurfaces.cs.cornell.edu/
        ii. MinC - http://opensurfaces.cs.cornell.edu/publications/minc/
            - 23: brick, carpet, ceramic, fabric, foliage, food, glass, hair, leather, metal, mirror, other, painted, paper, plastic, polishedstone, skin, sky, stone, tile, wallpaper, water, wood
2. From “semantics” / object category. (see Category)
    - The object will be recognized as “chair”.
    - Chairs are typically made of wood.
    - Physical properties (material, density, elasticity, surface roughness) of “typical wood” can be looked up in some online database.

## **natural language description (free style text) (ICL)**- category level description, instance level

Estimation method: trained LSTM
Representation: 1x36D
[different attributes, e.g. heat resistant, with removable parts]
Link to code 1: NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB

# Shape (ENPC)

Estimation method1: SfM, RGBD
Representation: 3D mesh, texture
Representation 1 : labels 1x6D
[cuboid,irregularly shaped,hemisphere,cylindrical,long and thin shaped,flat]
Link to code 1 : NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB

Estimation method: deep NN
Input2: partial view of the object
Representation 2 : shape as voxel grid
Representation of uncertainty2: shape samples (Monte Carlo sampling)
Link to code 2 :

- https://github.com/aalto-intelligent-robotics/shape_completion_network
- [https://github.com/aalto-intelligent-robotics/robust_grasp_planning_over_uncertain_shape_completions]

Link to dataset 2:
Paper: Lundell, J., Verdoja, F., & Kyrki, V. (2019, November). Robust Grasp Planning Over Uncertain Shape Completions. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 1526-1532). IEEE.

Several representations are possible:

- Volumetric shape, possibly a sampling-based approximation (Lundell, Kyrki)
- P(occupied) per voxel
- Gaussian processes
- Level-set methods

We will need something with a “local description” of the object, not just a global shape, so it can be updated from actions….

# Public code for 3D shape prediction from single images

a) AtlasNet: http://imagine.enpc.fr/~groueixt/atlasnet/ (from ENPC, but pre-IPALM): predicts a point cloud.
b) Hasson CVPR 2019 paper: https://hassony2.github.io/obman.html (predicts a mesh, tends to smooth the objects)
c) Occupancy networks: https://github.com/autonomousvision/occupancy_networks trains a classifier that predicts if a 3D point is inside or outside an object. Then, the 3D space is sampled using 3D points and the object can be reconstructed using Marching Cubes.
d) DeepSDF: https://github.com/facebookresearch/DeepSDF similar to occupancy networks but predicts a Signed Distance Function rather than a classifier (the difference is the same as between classification and regression).

c) and d) have been demonstrated mostly on synthetic images with uniform background, and the object is centered in the image. However, we can now use Mask-RCNN / detectron2 to segment the objects → we can replace the background by a uniform one, and center the image on the object. This should help the methods above.

## **Size (vector)** - meters of what? Convex hull?

Estimation method:
Representation 1: labels 1x3D
[small,medium,large]
Link to code 1: NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB
Representation 2: BBox 1x3D

## **Sound (action-reaction labels, audio samples) - (IRI)**

Estimation method: trained NN
Representation: labels 1x11D
[rubber,metal,plastic,wood,ceramic,(paper, fiber, foam,leather,felt),glass] (these are all the materials in the YCB dataset)
Link to code: (code to predict material is not available, we will do that). These are some related papers. http://gamma.cs.unc.edu/ISNN/
http://andrewowens.com/vis/

## Stiffness (N/m) (CTU)

Estimation method: Squeezing / pressing the object.
Action parameters: Gripper type, Finger configuration (if available), Squeezing speed
Input: Stress/strain relationship. Depending on gripper, can be current/force vs. gripper position/aperture.
Representation 1: k = F/d. (in one axis)

- Units: N/m.
- Requirements: Calibrated gripper to get absolute numbers. If not present, relative comparison of objects possible.

Representation 1.5: something in between 1 and 2 ;)
Representation 2: predefined categories - e.g. hard, slightly deformable, soft (these terms are not in line with the treatment of stiffness in physics; there: the opposite of stiffness is flexibility or compliance)
Link to code: Gripper dependent. https://gitlab.fel.cvut.cz/body-schema/ipalm/ipalm-grasping
Link to other attributes: Elasticity.
Definition: Measure of the resistance offered by an elastic body to deformation. A property of an object, i.e. material + shape. (“That is, the [[elastic] modulus is an intensive property of the material;

stiffness, on the other hand, is an extensive property of the solid body that is dependent on the material and its shape and boundary conditions." https://en.wikipedia.org/wiki/Stiffness)

## **Structure (?) -** parts, labels in 3D **(ICL)**

Estimation method:
Representation:
Link to code:

## Texture

See Fishel & Loeb 2012

## Visible Surface (ENPC)

Estimation method: RANSAC-based Plane Removal from RGB-D input(s)
Representation: 3D mesh
Link to code: https://github.com/dulucas/plane_detection_RGBD

## Surface roughness / Friction (AAL)

Estimation method: probabilistic estimation based on sliding action + vision
Representation: point on the object
Link to code:
Link to publication: https://arxiv.org/abs/2010.08277

## **use cases (?)** labels

Estimation method:
Representation:
Link to code:

## volume (m3)

Estimation method:
Representation:
Link to code:

## Weight, mass (kg)  (CTU, AAL)

Estimation method: robot interaction - lift object and move into specific position, exploiting joint torque sensors in robot manipulator - gravity acting on specific robot axes….
Representation 1: labels 1x3D
[light,medium,heavy]
Link to code 1: NS-VQA
Link to dataset 1: https://michaal94.github.io/SHOP-VRB
Representation 2: Mass (kg)
Link to code: TODO

# Object instance properties

(alphabetically ordered)

## Objectness (ENPC):

Estimation method: class-agnostic MASK-RCNN
Input: single RGB image
Representation: segmented masks
Link to code:  git@github.com:dulucas/ObjSeg_MaskRCNN.git (please contact yuming.du@enpc.fr if you want to use it)

## Object/Camera pose: (ENPC)

Estimation method1: EPOS
Input: single RGB image, tested on T-LESS, **YCB**-V and LM-O datasets
Representation: 3D rotation+3D translation of the object in the camera coordinate system.
Link to code: https://github.com/thodan/epos_internal
Paper: Hodan, T., Barath, D., & Matas, J. (2020). EPOS: Estimating 6D Pose of Objects with Symmetries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 11703-11712). Link
Notes: Running on CTU setup

## Object/scene reasoning: (ICL)

Estimation method:
Representation:
Link to code: