인공지능 기반의 로봇 파지 계획 기술

2021. 06. 23(Wed.)

Sungkyunkwan University
Robotics and Intelligent System Engineering Lab.
Minseok Kang

SUNG KYUN KWAN UNIVERSITY

RISE LAB
Robotics and Intelligent System Engineering

연구의 최종목표

• Problem Statement •

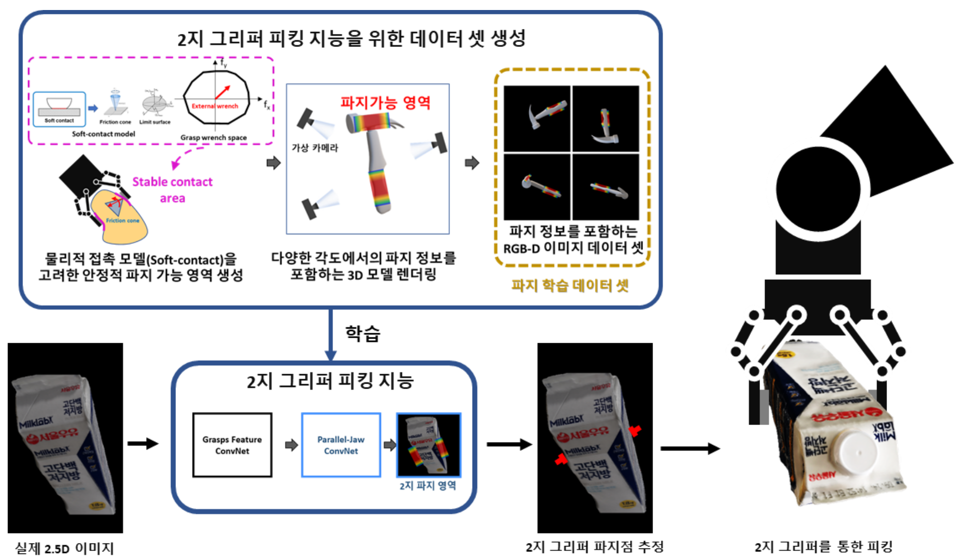
- 산업에서 성공적으로 사용되는 공압 및 2지 그리퍼로 **다양한 물체를 빠르고 안정적으로 피킹 할 수 있는 지능**은 무엇인가
- **시간과 노력을 최소화**하며 **양질의 학습 데이터**를 얻기 위한 방법은 무엇인가?
- 센싱, 제어에서 발생하는 **오차에 강인하도록 피킹 지능을 학습하기 위한 방법**은 무엇인가?

• Proposed Solution •

- 2지 및 공압 그리퍼 피킹 **지능의 융합**을 통해 **2지와 공압이 융합된 그리퍼로 물체를 피킹**할 수 있도록 지능 증강
- 실제 RGB-D 데이터와 파지 정보를 포함한 가상 RGB-D 데이터를 **Canonical-form 형태로 변환** 후 학습하여 **예측 성공률을 높임**
- 피킹 지능 학습 데이터의 안정적인 파지위치 라벨을 **파지점으로부터 파지영역으로 확장**

❑ 그리퍼 피킹 지능의 학습을 위한 물체 파지 데이터 생성 기술 개발

❑ 비전 센서, 3D 센서 등의 센서 데이터를 기반으로 물체 파지 가능 위치를 추정하는 그리퍼 피킹 지능 기술 개발



<2지 그리퍼 피킹을 위한 물체 파지 학습 데이터 생성 및 지능 개발>

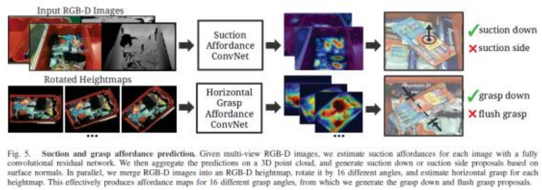
What is good grasp?

- Find a gripper configuration that maximizes a success(or quality) metric
  - ***Empirical methods***

1) Human Label [1-2]

Cornell Grasp Dataset

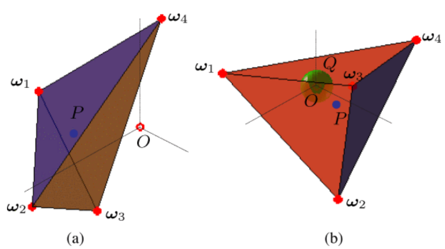
Amazon Picking Challenging (MIT-Princeton)



2) Physical error [3]

  - ***Analytic method***: Consider performance according to physical models

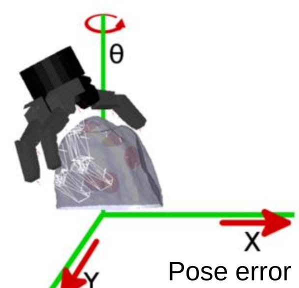
1) Grasp Wrench Space(GWS) [4]



Non-FC grasp

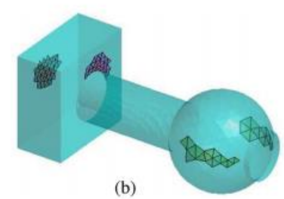
FC grasp

2) Robust GWS [5]



Pose error perturbation

3) Independent Contact Region [4]

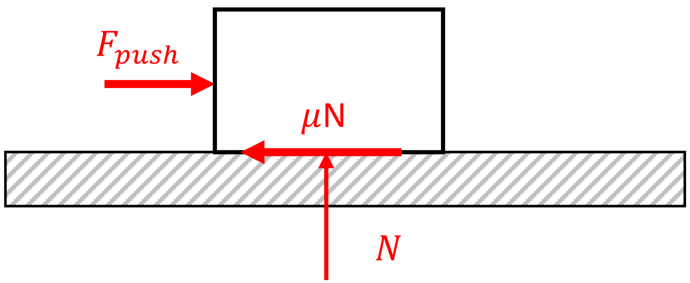


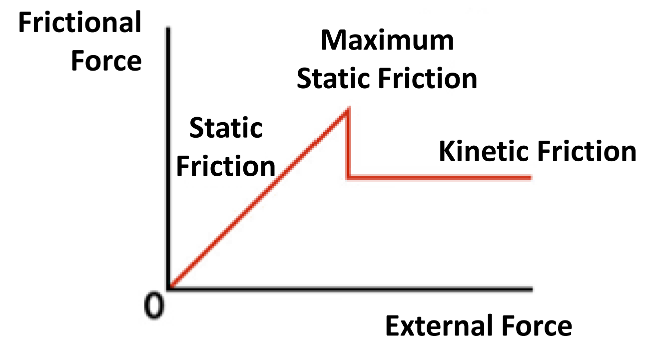
Graspable region

[1] http://pr.cs.cornell.edu/grasping/rect_data/data.php
[2] Zeng, Andy, et al. "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching." *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018.
[3] Pinto, Lerrel, and Abhinav Gupta. "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours." *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016.
[4] Roa, Máximo A., and Raúl Suárez. "Computation of independent contact regions for grasping 3-d objects." IEEE Transactions on Robotics 25.4 (2009): 839-850.
[5] Weisz, Jonathan, and Peter K. Allen. "Pose error robust grasping from contact wrench space metrics." *2012 IEEE international conference on robotics and automation*. IEEE, 2012.

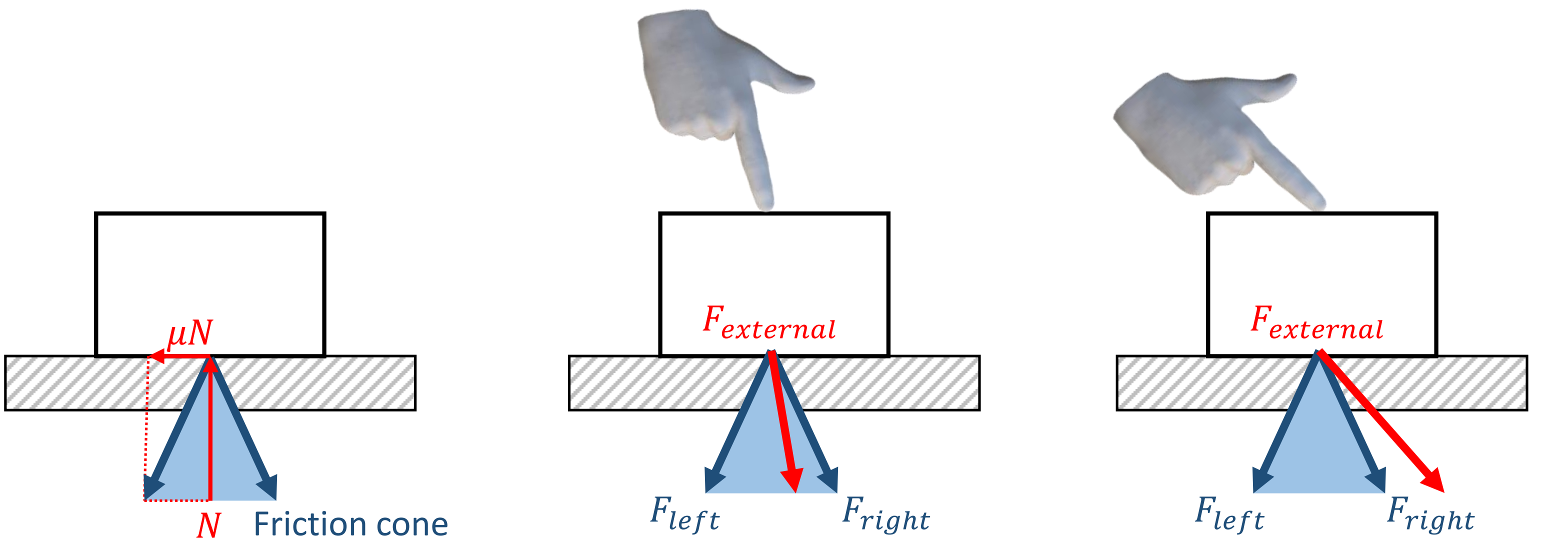
Grasp Quality

- Coulomb friction





- Friction cone

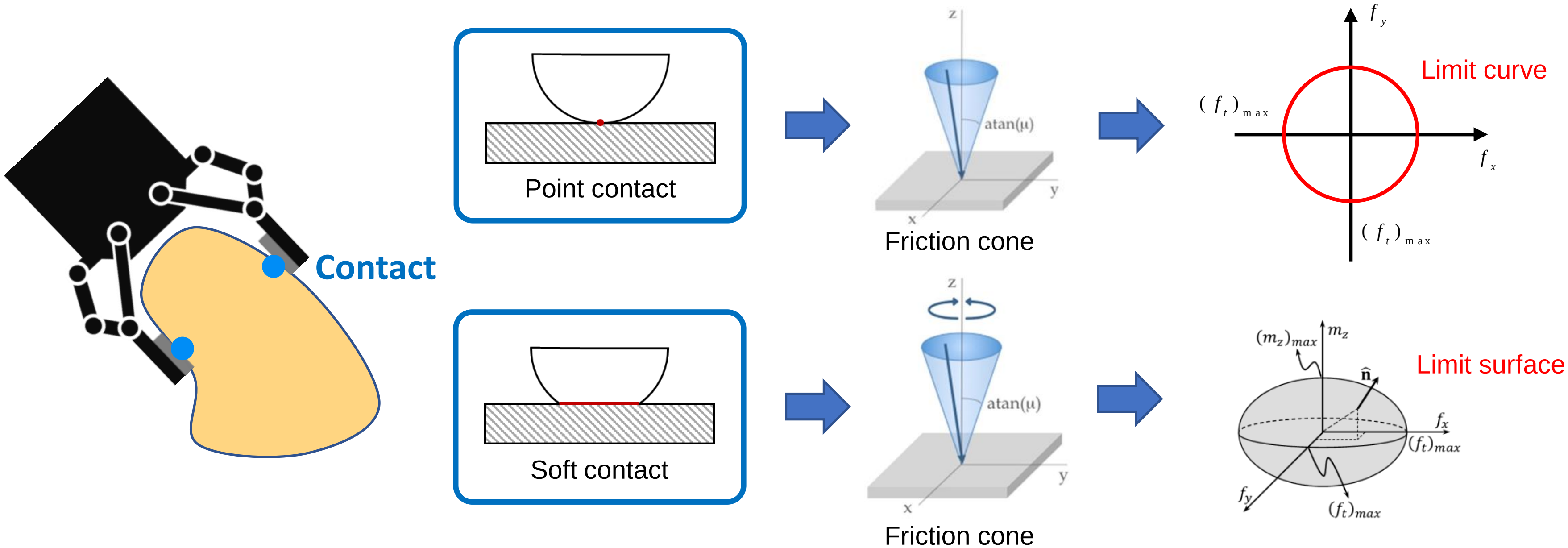


It can resist external force

It can not resist external force

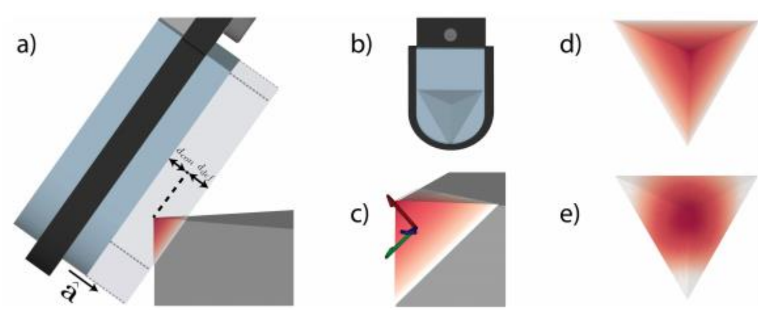
Grasp Quality

❑ Contact model

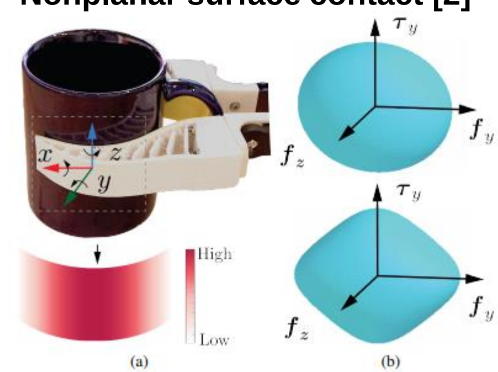


❑ Complicate contact models

Assume gripper pad is deformable [1]



Nonplanar surface contact [2]



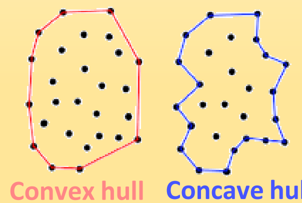
[1] Danielczuk, Michael, et al. "Reach: Reducing false negatives in robot grasp planning with a robust efficient area contact hypothesis model." International Symposium on Robotics Research (ISRR). 2019.
[2] Xu, Jingyi, et al. "6DLS: Modeling Nonplanar Frictional Surface Contacts for Grasping Using 6-D Limit Surfaces." IEEE Transactions on Robotics (2021).

Grasp Quality

Wrench?
Force and torque vectors

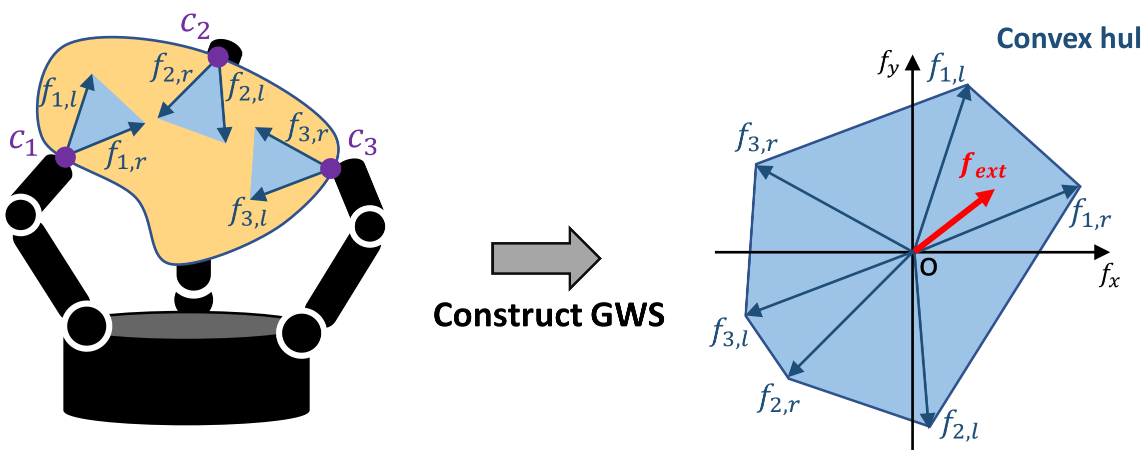
$$[f_x, f_y, f_z, \tau_x, \tau_y, \tau_z] \in \mathbb{R}^6$$

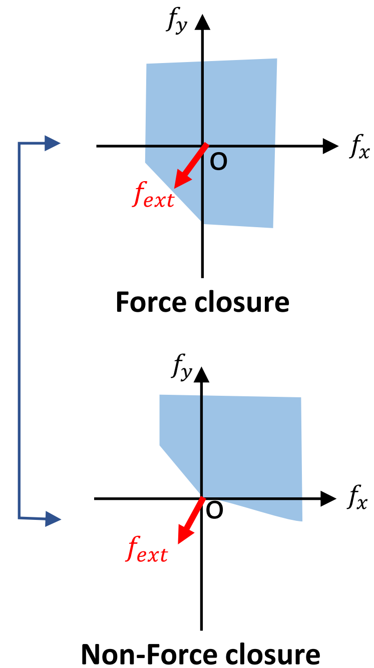
Convex Hull?



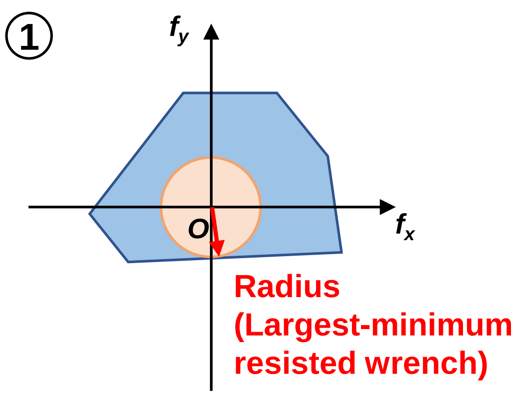
- Grasp Wrench Space(GWS)

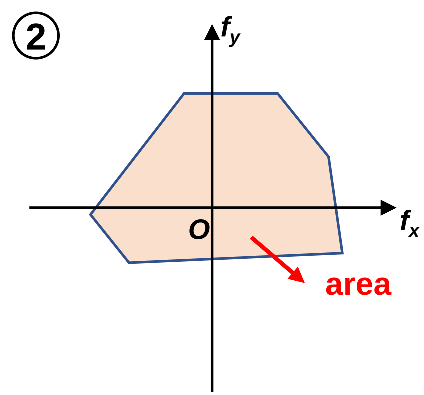
  Assumption :
  1) 2D plane object
  2) No rotational motion. Thus, we don't consider torque





- Grasp Quality



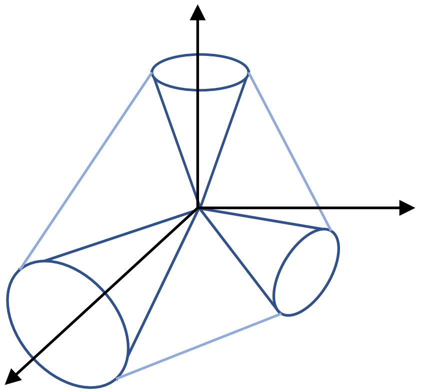


…

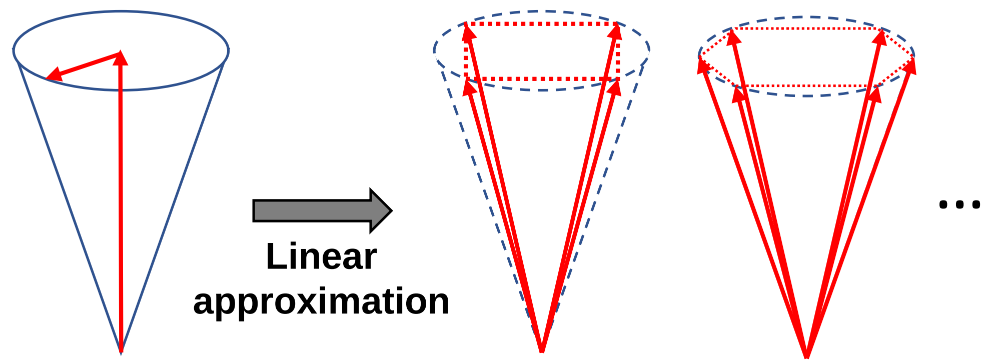
There are more grasp quality metrics.

Grasp Quality

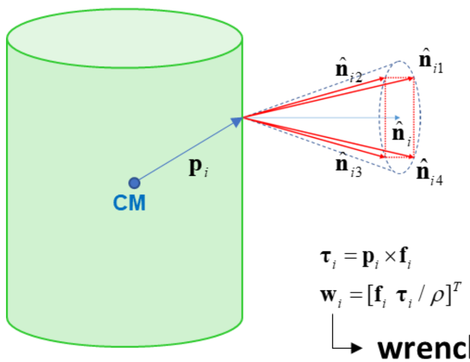
- GWS on 3D object
  - ✓ **Approximation of friction cone**



GWS on *n*-dimensional space
$(\boldsymbol{n} \geq 3)$



  - ✓ **GWS in 3D object**



$\boldsymbol{\tau}_i = \mathbf{p}_i \times \mathbf{f}_i$

$\mathbf{w}_i = [\mathbf{f}_i \;\; \boldsymbol{\tau}_i / \rho]^T$

↳ **wrench**

Wrenches applied at $\mathbf{p}_i$

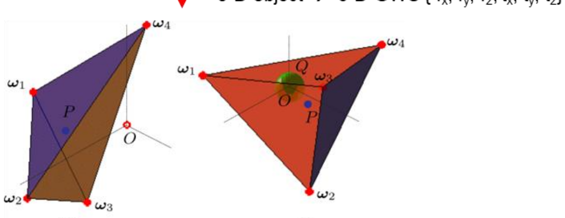
1) $\mathbf{w}_i$: wrench generated by a unitary force $\mathbf{f}_i$ orthogonal to the object surface
2) $\mathbf{w}_{ij}$: wrench generated by a unitary force $\mathbf{f}_i$ along an edge of the linearized friction cone → **Primitive wrench**

$$C = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$$

$$G = \{\mathbf{w}_1, \ldots, \mathbf{w}_n\}$$

$$W = \{\mathbf{w}_{11}, \ldots, \mathbf{w}_{1m}, \ldots, \mathbf{w}_{n1}, \ldots, \mathbf{w}_{nm}\}$$

2-D object → 3-D GWS {f_x, f_y, t_z}
3-D object → 6-D GWS { f_x, f_y, f_z, t_x, t_y, t_z}



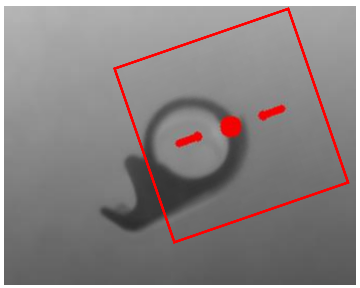
(a) **Non-FC grasp** (b) **FC grasp**

Dex-Net 2.0

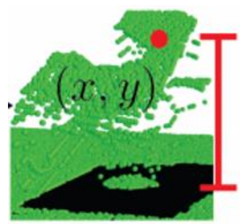
Dex-Net 2.0

- Grasp Quality Convolutional Neural Network(GQ-CNN)

Aligned image

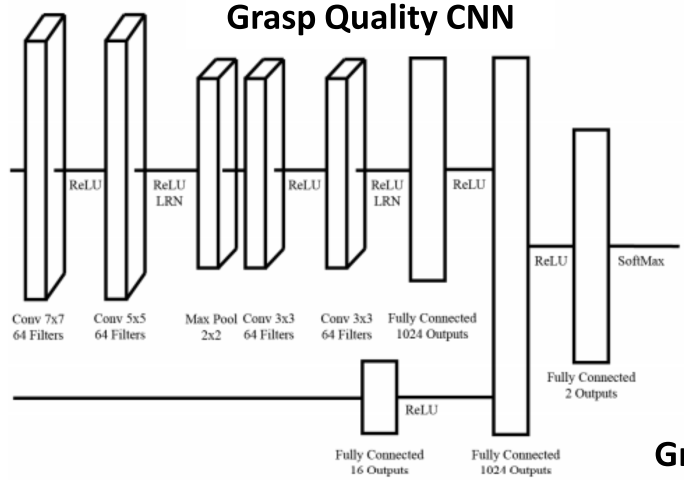


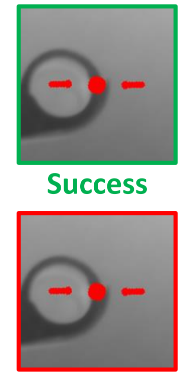
Grasp candidate



z

Grasp Quality CNN





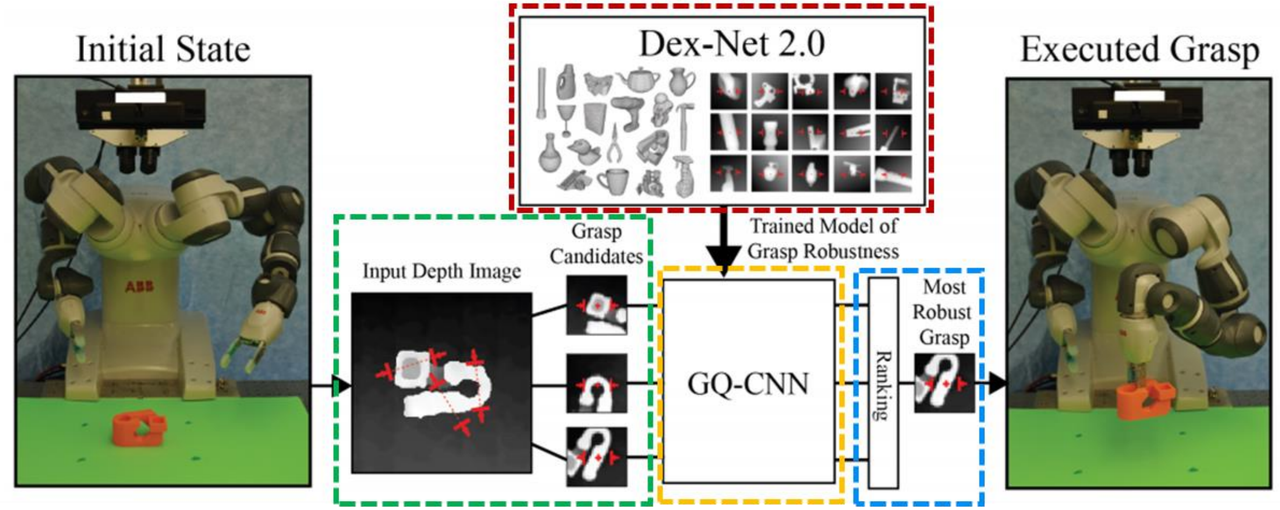
Success

Fail

Grasp Success Metric

- Architecture

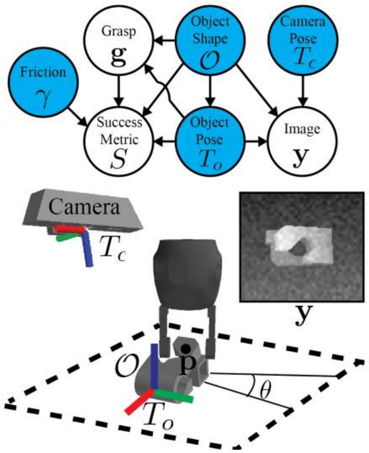
Dataset Generation



Grasp candidate sampling Grasp Quality CNN Rank grasp candidates

Dex-Net 2.0

❑ Problem Statements



Graphical model for robust parallel-jaw grasping of objects

States $\mathbf{x} = (O, T_o, T_c, \gamma)$

obj info, obj pose, cam pose, friction coef

Point cloud $\mathbf{y} = \mathbb{R}_+^{H\times W}$

Grasps $\mathbf{u} = (\mathbf{p}, \varphi) \in \mathbb{R}^3 \times S^1$

Grasp metric $S(\mathbf{u}, \mathbf{x}) \in \{0,1\}$

(robust analytic)

$p(S, \mathbf{u}, \mathbf{x}, \mathbf{y}) = p(\mathbf{x})\, p(\mathbf{y} \mid \mathbf{x})\, p(\mathbf{u} \mid \mathbf{x})\, p(S \mid \mathbf{u}, \mathbf{x})$

state distribution
observation model
grasp candidate model
analytic model of grasp success

Let $p(S,\mathbf{u},\mathbf{x},\mathbf{y})$ be a joint distribution on … imprecision in sensing and control.

Let the *robustness* of a grasp given an observation: $Q(\mathbf{u},\mathbf{y})=\mathrm{E}(S|\mathbf{u},\mathbf{y})$

1) Goal is to find(learn) a *robustness function* : $Q_{\theta^*}(\mathbf{u}, \mathbf{y}) \in \{0,1\}$

$$\theta^* = \underset{\theta\in\Theta}{\operatorname{argmin}} E_{p(S,\mathbf{u},\mathbf{x},\mathbf{y})}[\mathcal{L}(S, Q_\theta(\mathbf{u},\mathbf{y}))]$$

NN parameters

Cross entropy loss

Cross Entropy

$$H(p,q) = -\sum_{x\in X} p(x) \log q(x)$$

$p(x)$: true distribution

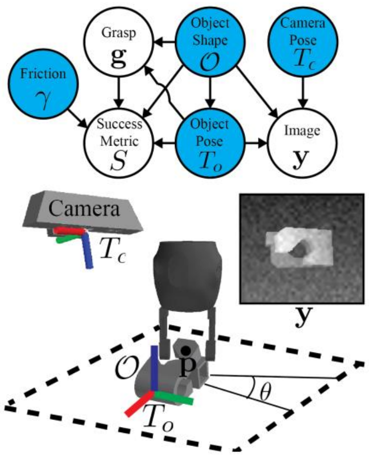
$q(x)$: estimate distribution

2) Using in grasp planning

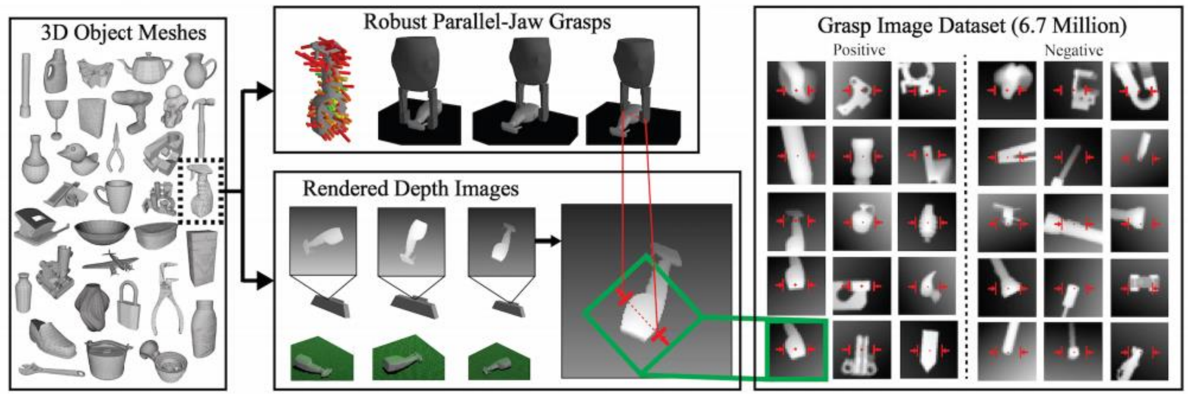
$$\pi_\theta(\mathbf{y}) = \underset{\mathbf{u}\in C}{\operatorname{argmax}}\, Q_\theta(\mathbf{u}, \mathbf{y})$$

Grasp candidates

Dex-Net 2.0 – Dataset Generation



Graphical model for robust parallel-jaw grasping of objects



Dex-Net 2.0 pipeline for training dataset generation

state distribution, observation model

Model $(S_1, \mathbf{u}_1, \mathbf{x}_1, \mathbf{y}_1), \ldots, (S_N, \mathbf{u}_N, \mathbf{x}_N, \mathbf{y}_N) \sim p(S, \mathbf{u}, \mathbf{x}, \mathbf{y}) = p(\mathbf{x})\, p(\mathbf{y} \mid \mathbf{x})\, p(\mathbf{u} \mid \mathbf{x})\, p(S \mid \mathbf{u}, \mathbf{x})$

i.i.d samples

grasp candidate model, analytic model of grasp success

1) state distribution p(x)

$$p(\mathbf{x}) = p(\gamma)\, p(O)\, p(T_O \mid O)\, p(T_c)$$

Distribution	Description
$p(\gamma)$	truncated Gaussian distribution over friction coefficients
$p(\mathcal{O})$	discrete uniform distribution over 3D object models
$p(T_o \mid \mathcal{O})$	continuous uniform distribution over the discrete set of object stable poses and planar poses on the table surface
$p(T_c)$	continuous uniform distribution over spherical coordinates for radial bounds $[r_\ell, r_u]$ and polar angle in $[0, \delta]$

$p(\gamma) \sim \mathrm{N}(0.5, 0.1)$ truncated to $[0,1]$

$p(O) \sim \mathrm{U}$(given 3D obj dataset)

$p(T_O \mid O) = p(T_O \mid T_s)\, p(T_s \mid O)$

$p(T_s \mid O) \sim \mathrm{U}$(stable poses)

$p(T_O \mid T_s) \sim \mathrm{U}([-0.1, 0.1] \times [-0.1, 0.1] \times [0, 2\pi])$

$p(T_c) \sim \mathrm{U}([0.65, 0.75] \times [0, 2\pi) \times [005\pi, 0.1\pi])\ (\ r, \theta, \varphi)$

2) grasp candidate model p(u|x)

Uniform distribution over 1) & 2) & 3)

1) Pairs of antipodal contact points on obj surface
2) Grasp axis ⊥ table plane
3) Reject no FC nor no parallel (μ=0.6)

3) observation model p(y|x)

$$\mathbf{y} = \alpha \hat{\mathbf{y}} + \varepsilon$$

$\hat{\mathbf{y}}$: depth image created using OSMesa offscreen rendering

α : Gamma random variable with shape=1000.0 and scale=0.001

ε : Gaussian Process noise over pixel coordinates

with measurement noise $\sigma = 0.005$ and kernel bandwidth $\sqrt{2}\,px$

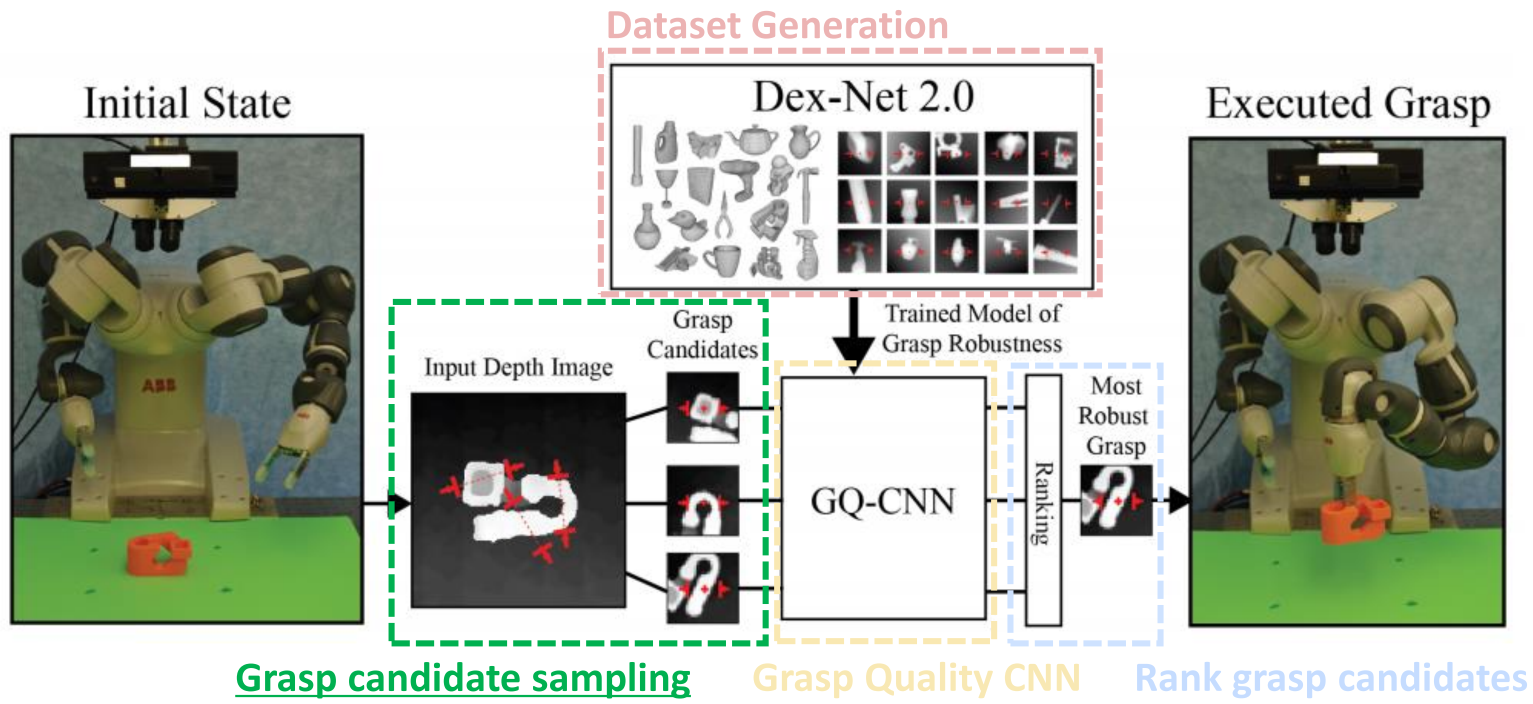
4) Analytic model of grasp success p(S|u,x)

Robust epsilon quality

$$S(\mathbf{u}, \mathbf{x}) = \begin{cases} 1 & E_Q > \delta \text{ and } collfree(\mathbf{u}, \mathbf{x}) \\ 0 & otherwise \end{cases}$$

E_Q : robust epsilon quality

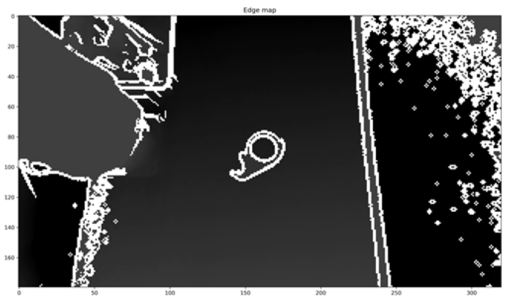
Dex-Net 2.0 – Grasp Candidate Sampling



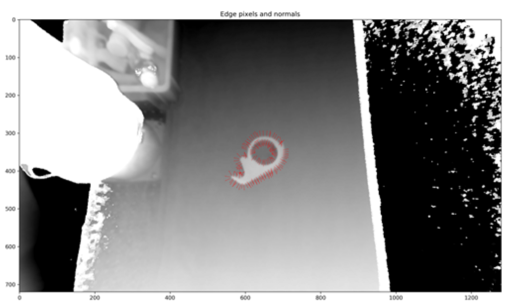
Dex-Net 2.0 – Grasp Candidate Sampling

❑ **Image based parallel-jaw grasp candidate sampling**

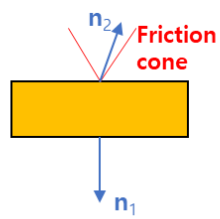
1. Get edge(gradient threshold)

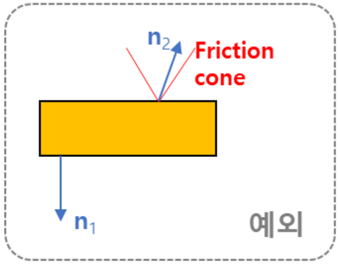


2. Get normal vectors at edge pixel(gradient)



3-1. Get antipodal points(F.C)



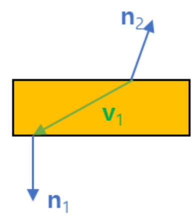


<conditions>

contact points dist(px) < grasp width(px)

$\mathbf{n}_1 \cdot \mathbf{n}_2 < -\cos(\operatorname{atan}(\mu))$

3-2. Get antipodal points(F.C)



<conditions>

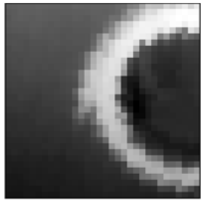
$\angle(\mathbf{n}_1, \mathbf{v}) \le \operatorname{atan}(\mu)$

$\angle(\mathbf{n}_2, -\mathbf{v}) \le \operatorname{atan}(\mu)$

4. contact1, contact2 pixel → center pixel, theta → crop image

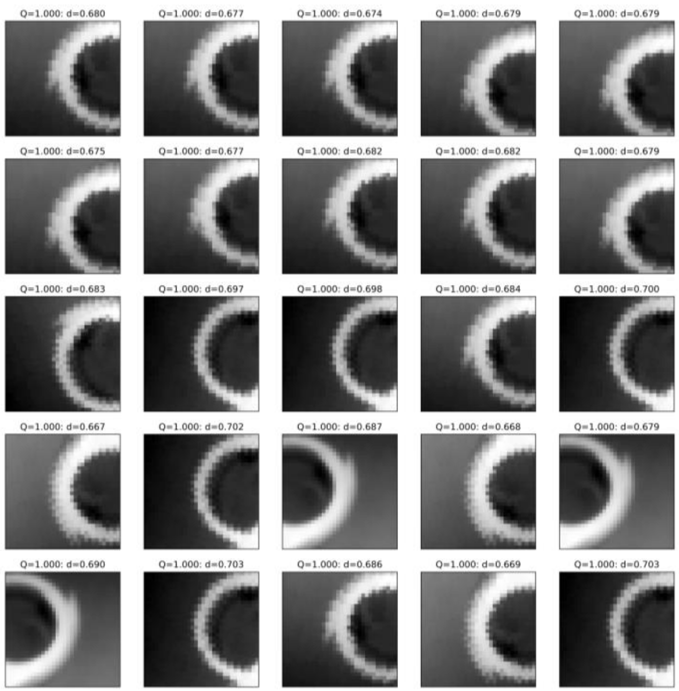
5. depth: depth at center pixel + offset(0.015~0.05)

d = 0.677

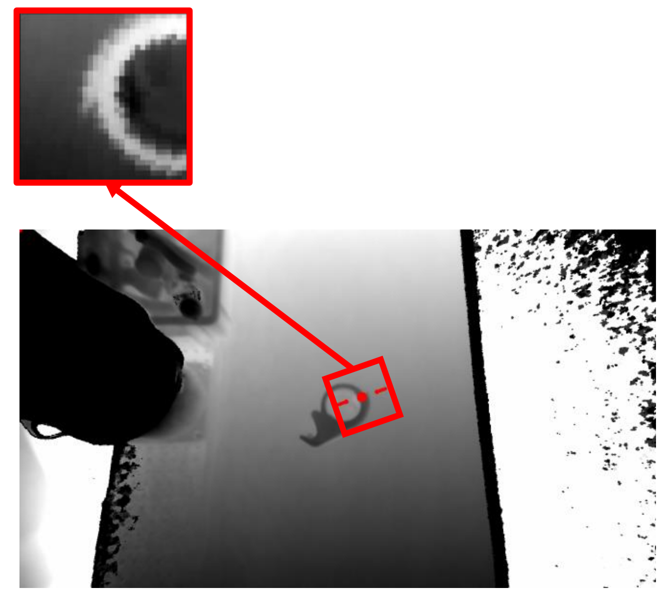


Crop image

Dex-Net 2.0 – Rank grasp candidates

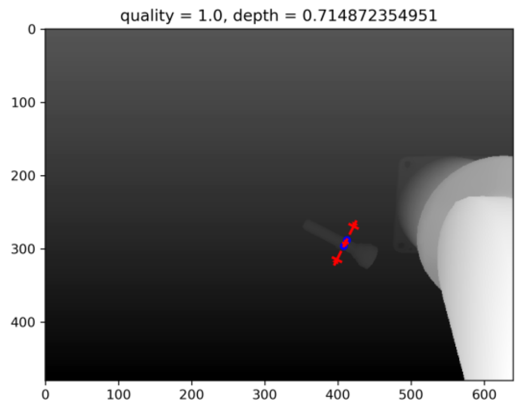


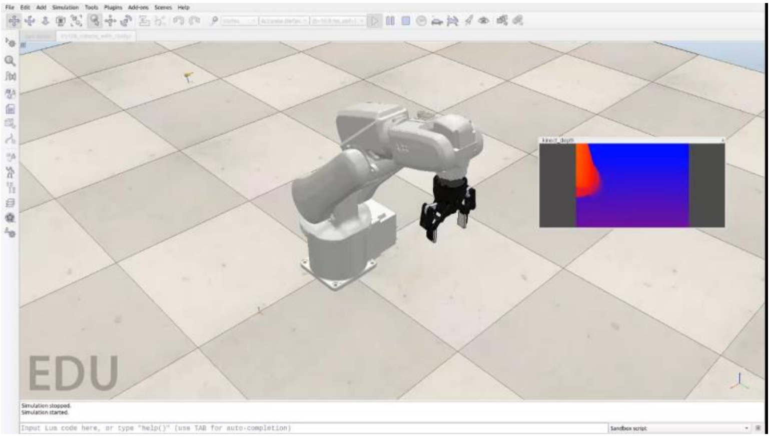
Grasp candidates

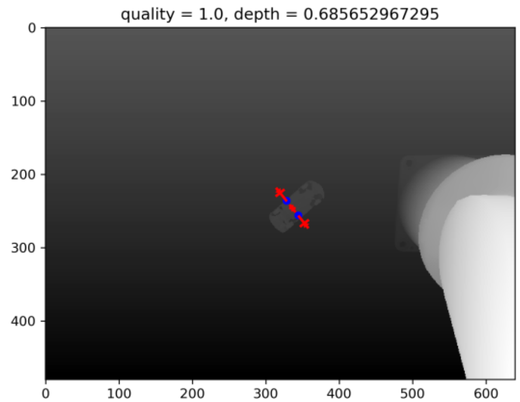


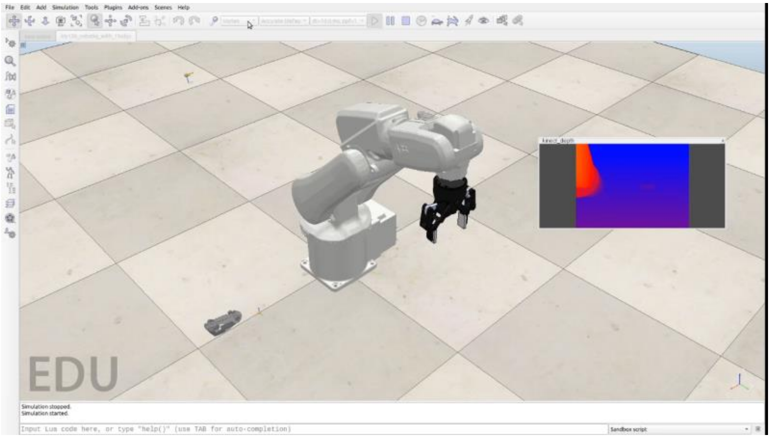
Top rank grasp

Dex-Net 2.0 – Simulation







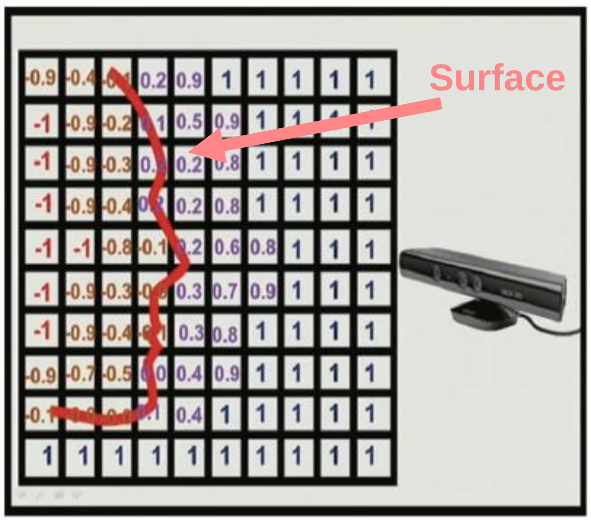


Simulation results

VGN

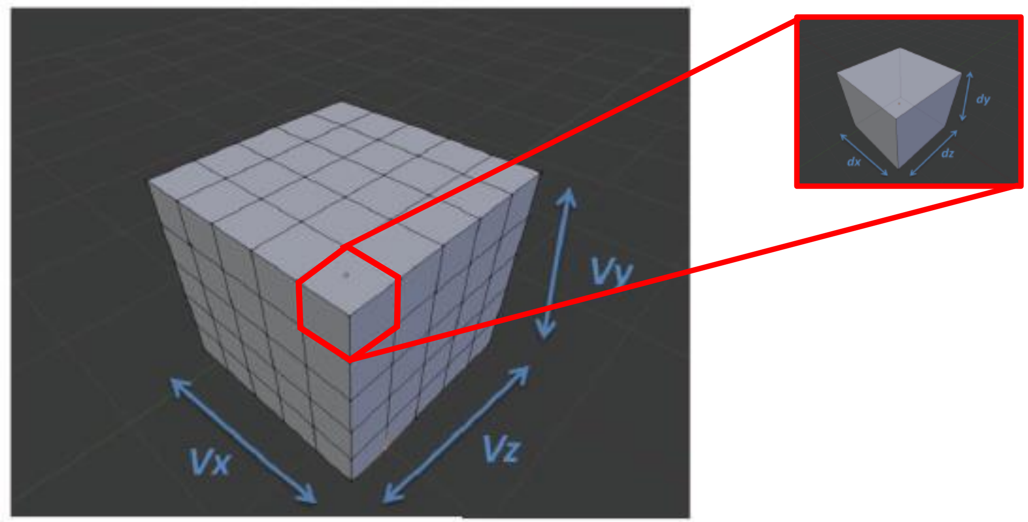
VGN – What is TSDF?

- **Truncation Signed Distance Function (TSDF)**
  - The truncated distance value formed when the camera observes the surface of object



TSDF on 2D pixel grid

Voxel

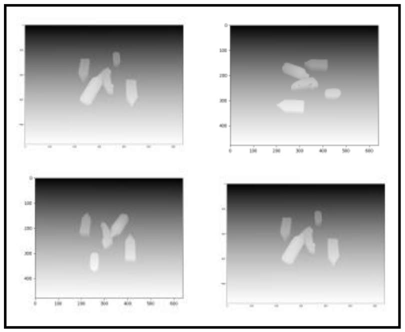


Voxel grid

$$\boldsymbol{TSDF} = \begin{cases} (\mathbf{0}, \mathbf{1}] & \textit{(Outside of the surface)} \\ \mathbf{0} & \textit{(On the surface)} \\ [-\mathbf{1}, \mathbf{0}) & \textit{(Inside the surface)} \end{cases}$$

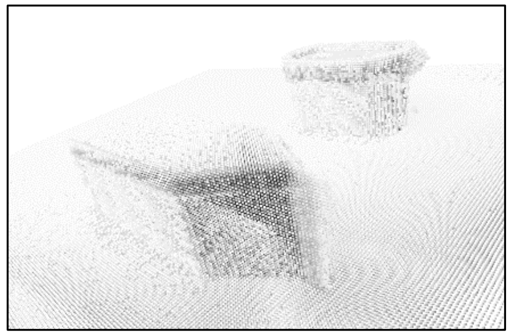
VGN – What is TSDF?

❑ 3D reconstruction with TSDF volume [1]



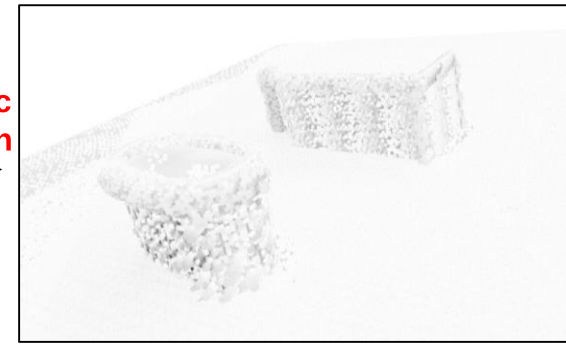
Depth image

ICP

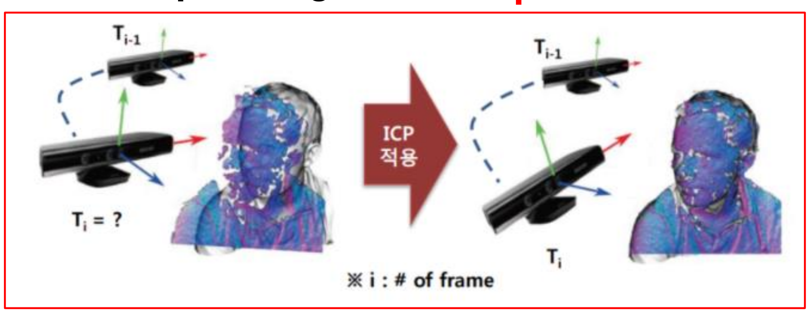


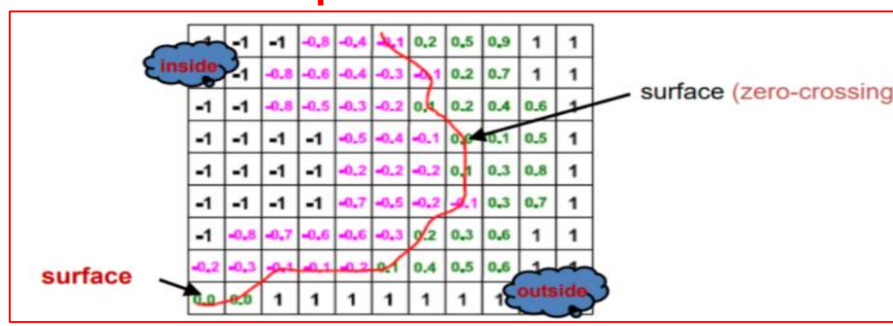
point cloud

Volumetric integration



TSDF



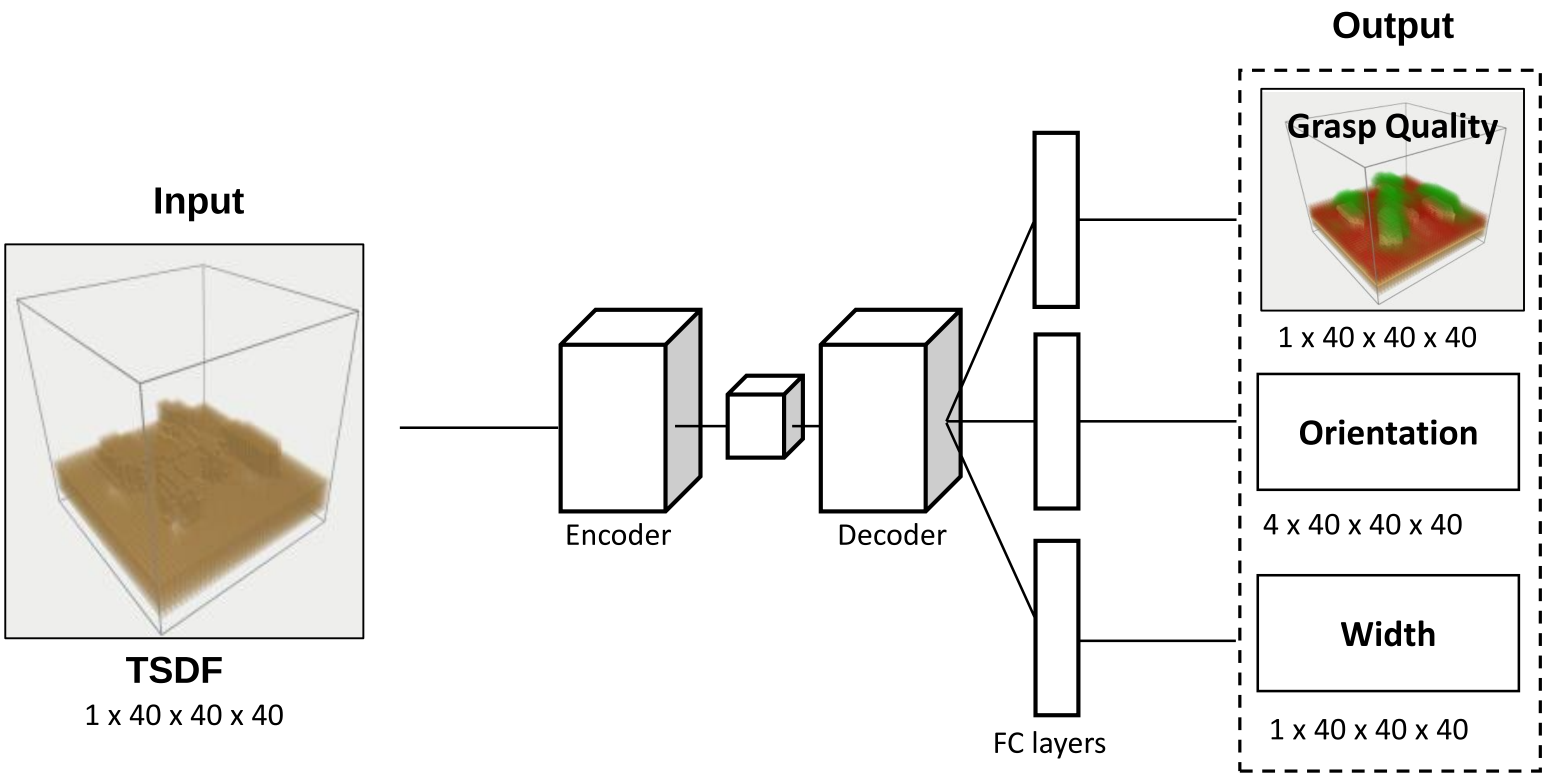


- Each Voxel of TSDF include distance function
- Reduces the noise of the sensor

[1] Izadi, Shahram, et al. "KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera." *Proceedings of the 24th annual ACM symposium on User interface software and technology*. 2011.

VGN

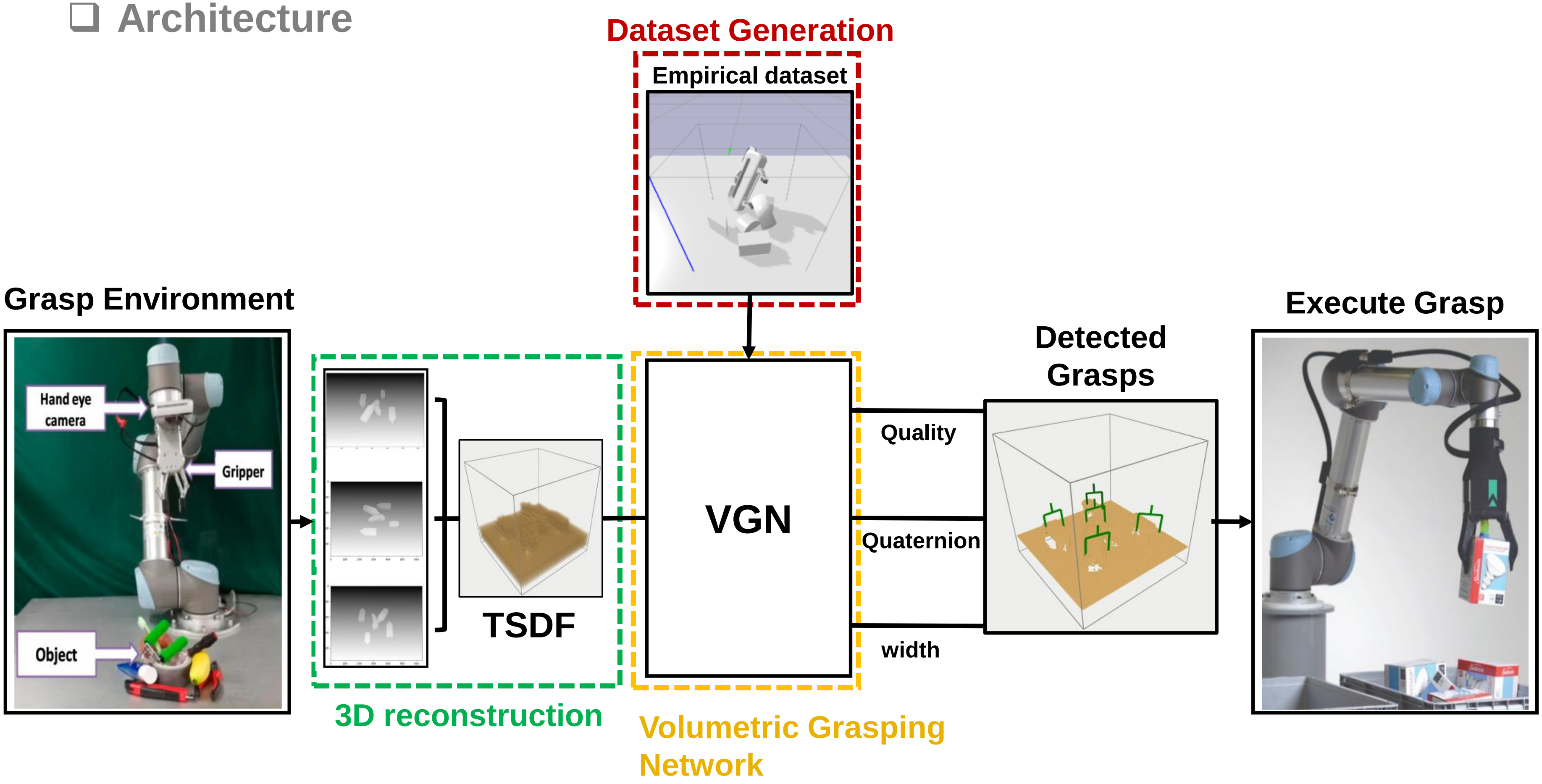
❑ Volumetric Grasping Network (VGN)



Volumetric Grasping Network

VGN

❑ Architecture

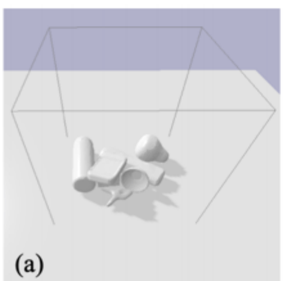


VGN

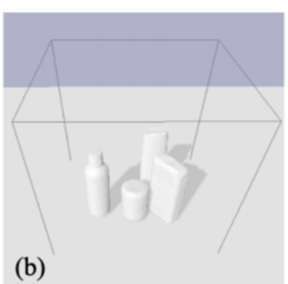
❑ Dataset Generation

- ❑ Empirical methods : Physical Trials using Pybullet simulator
- ❑ Active Search : Spherical coordinate system (r, θ, φ)
  : N ~ $\mathcal{U}(1,6)$, $r \sim \mathcal{U}(0.48, 0.72)$, $\theta \sim \mathcal{U}\left(0, \frac{\pi}{4}\right)$, $\varphi \sim \mathcal{U}(0, 2\pi)$
- ❑ Random point Grasp

1. Select scenes

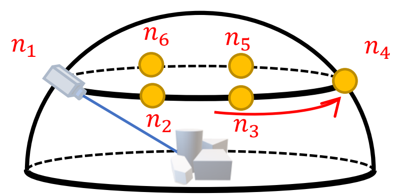


- Pile scenes : 4DOF

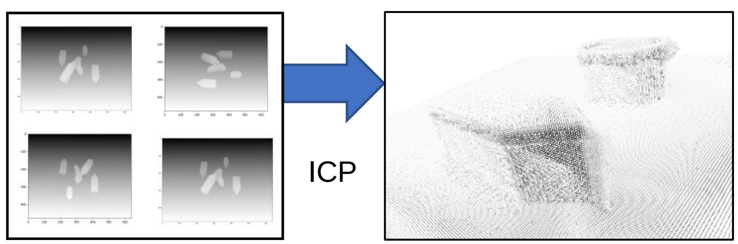


- Packed scenes : 6DOF

2. Object recognition



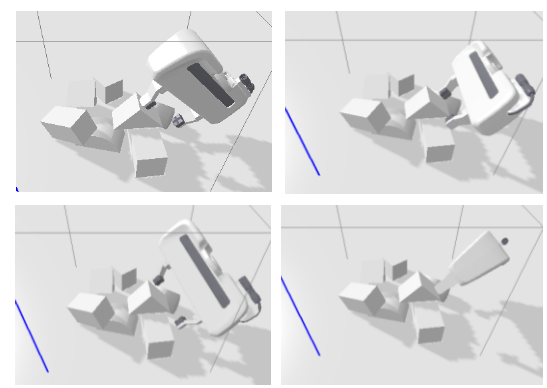
Active search



Depth image **Point cloud**

3. Random point grasp

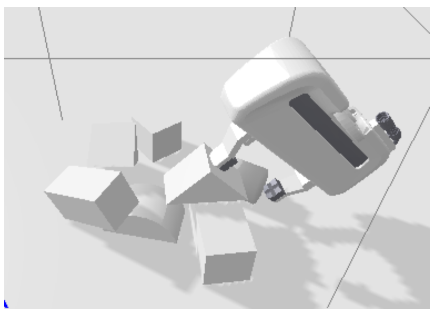
- Z axis rotation grasp (Yaw ~ $\mathcal{U}(0, \pi)$)

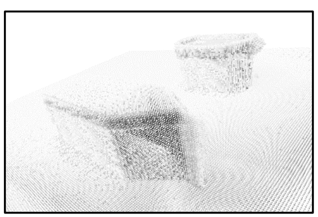


VGN

- ❑ **Dataset Generation**
  - ❑ Transform from point cloud to TSDF
  - ❑ Data labelling

4. Transform from point cloud to TSDF

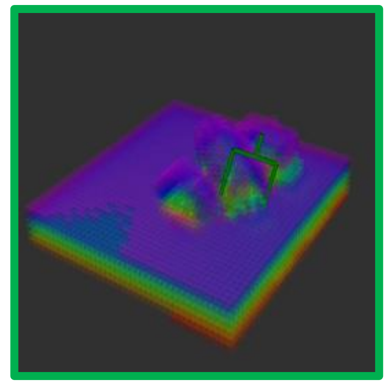




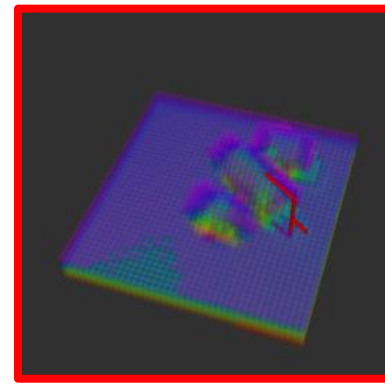
Point cloud

TSDF transform & Result Labelling

Success



Fail



$1\times40\times40\times40$

Orientation

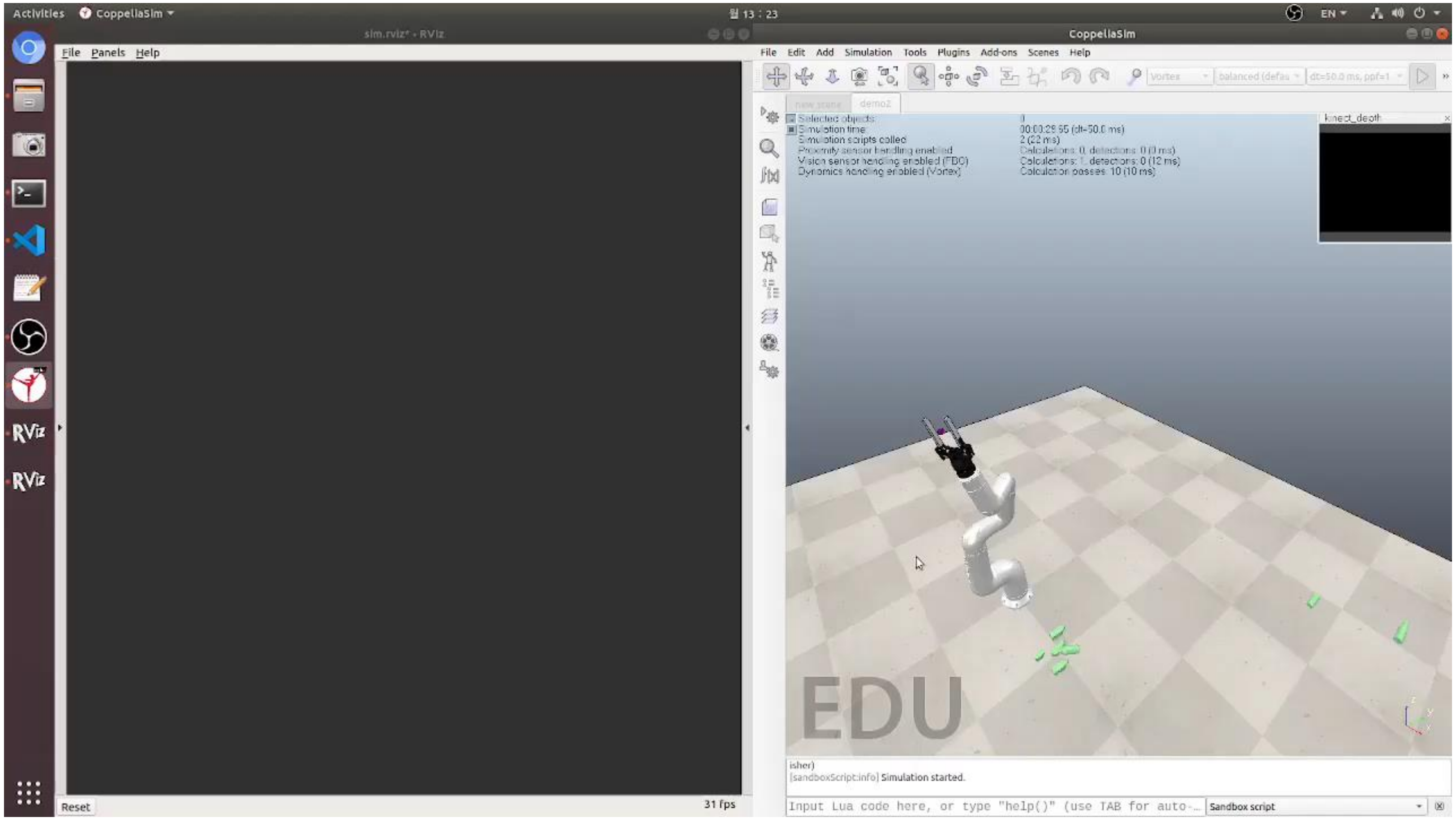
$4\times40\times40\times40$

Width

$1\times40\times40\times40$

VGN

❑ Simulation



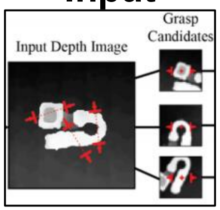
Simulation results

Conclusion

Dex-net 2.0 vs VGN Compare

Dex-Net 2.0

Input

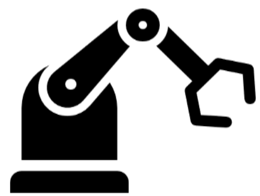


Depth image
Grasp candidates

Output

Grasp quality

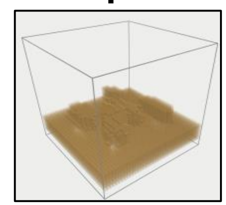
Analytic method
4DOF Grasp
Discriminative



Grasp

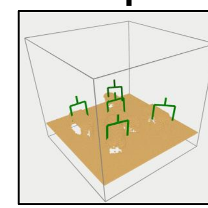
VGN

Input



TSDF

Output



Grasp candidates

Empirical method
6DOF Grasp
Generative

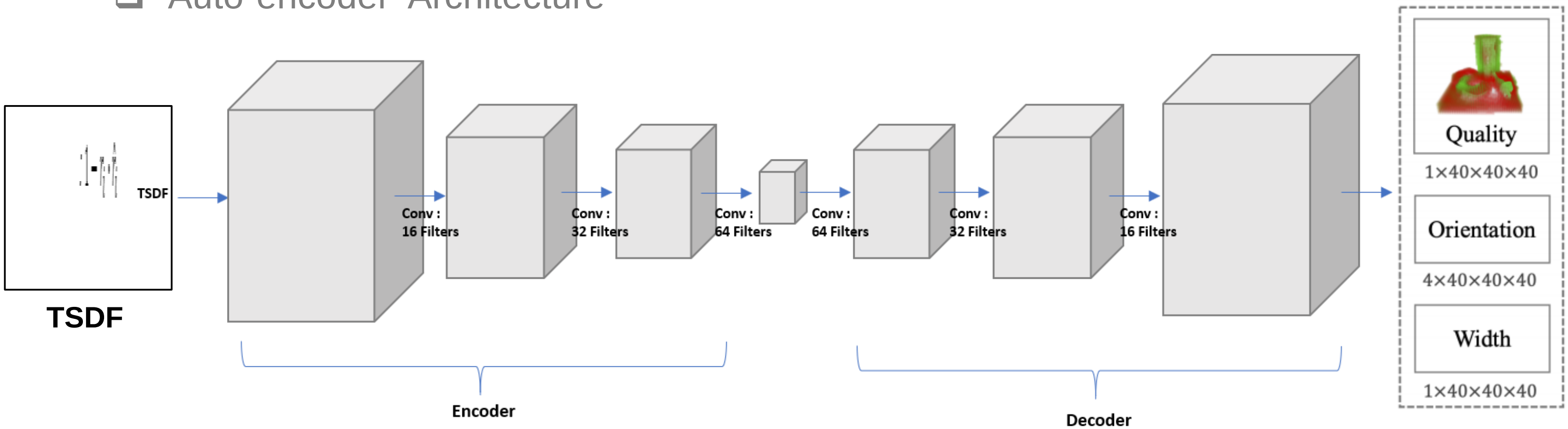
Thank you for your attention!

Appendix

VGN – 2) Volumetric Grasping Network

❑ VGN- NET

❑ Auto-encoder Architecture



❑ Loss Function

$$loss\ function:\ \mathcal{L}(g_i, \hat{g}_i) = \mathcal{L}(q_i,\ \hat{q}_i\ ) + q_i(\mathcal{L}(r_i, \hat{r}_i) + \mathcal{L}(w_i, \hat{w}_i\ ))$$

voxel | Grasp quality | Grasp quaternions | Grasp width

$$\begin{cases} \mathcal{L}(q_i, \hat{q}_i) & : \textit{cross entropy loss function} \\ \mathcal{L}(r_i, \hat{r}_i) & : 1 - r_i \cdot \hat{r}_i \\ \mathcal{L}(w_i, \hat{w}_i\ ) & : \textit{MSE} \end{cases}$$

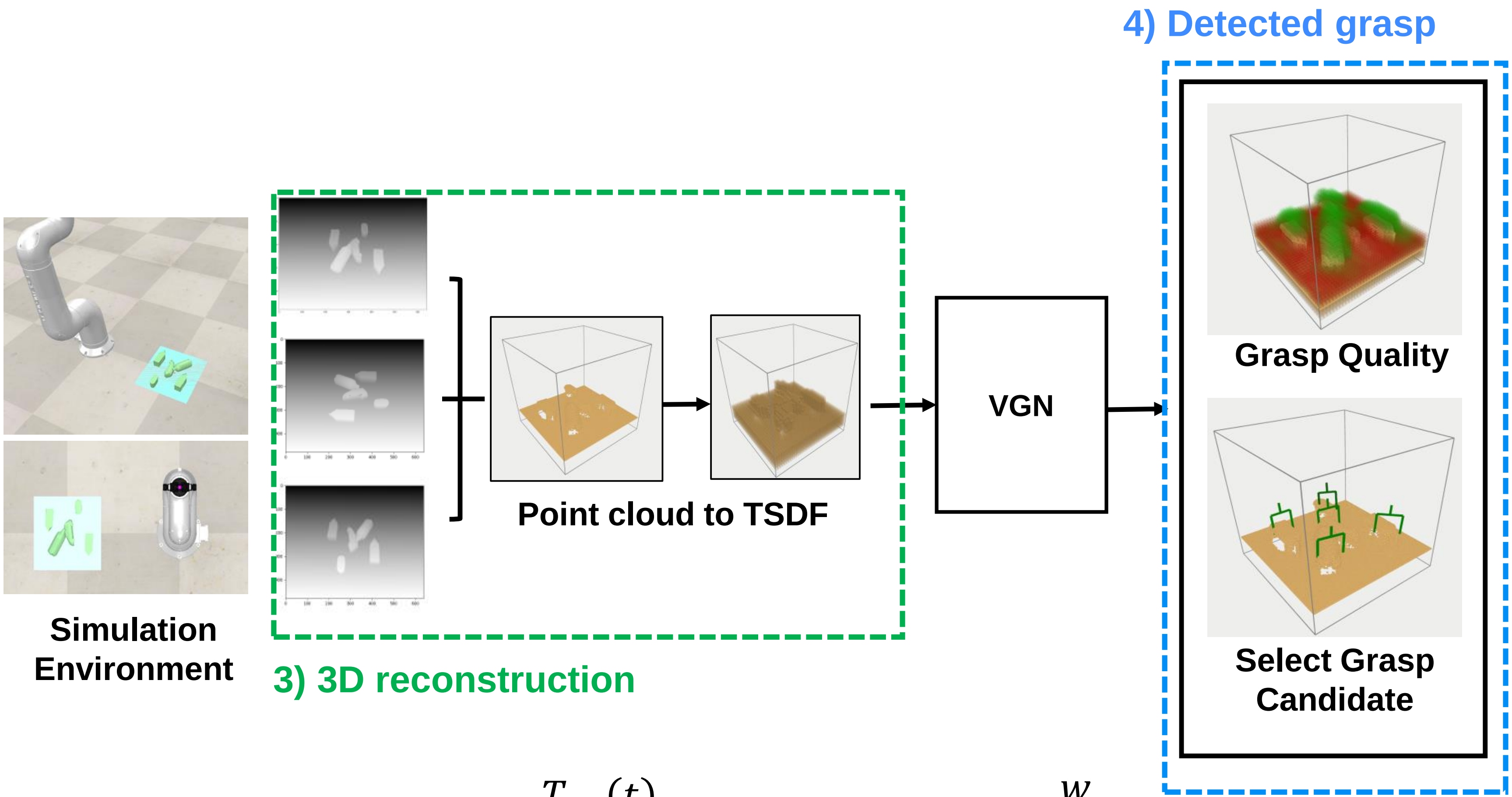
g_i : voxel

q_i : grasp label $\in \{0,1\}$ | $\hat{q}_i$: Ground truth grasp label

w_i: grasp quaternions | $\hat{w}_i$: Ground truth grasp quaternions

r_i : grasp width | $\hat{r}_i$: Ground truth grasp width

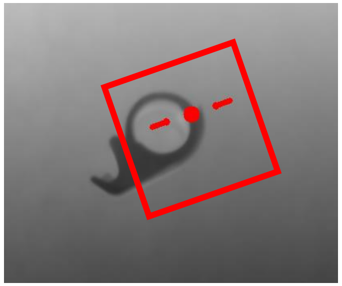
VGN – 3) Grasp Planning

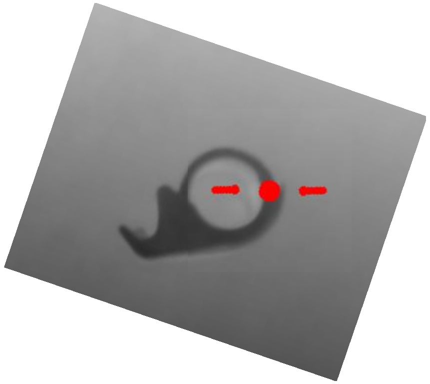


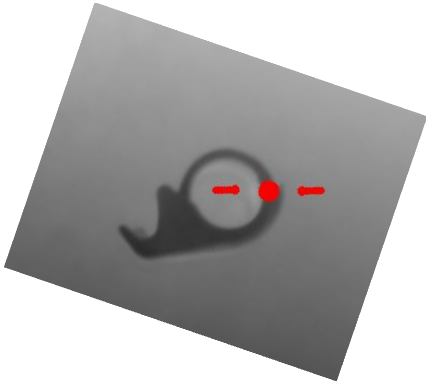
$$f : V \rightarrow Q, R, W \quad \longrightarrow \quad t' = \frac{T_{RV}(t)}{v} \quad r' = T_{RV}(r) \quad w' = \frac{w}{v}$$

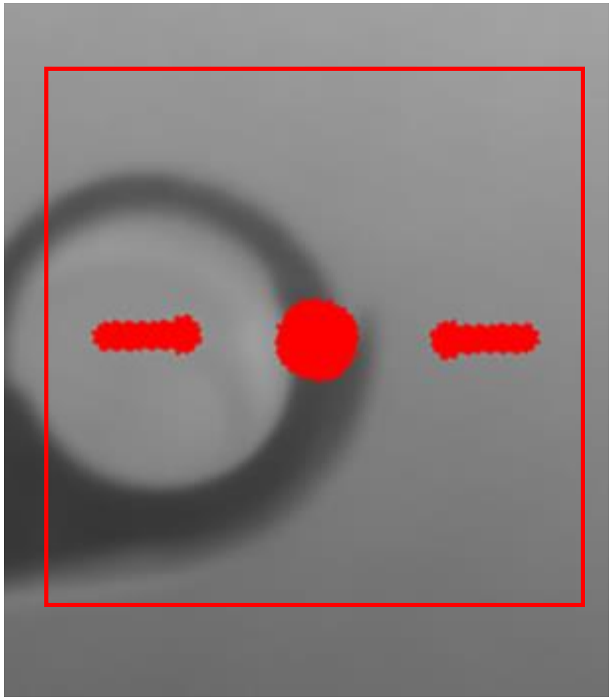
v : *voxel size*

T_{RV}: *Transfromation matrix between base , TSDF frame*









Dex-net 2.0

❑ Architecture