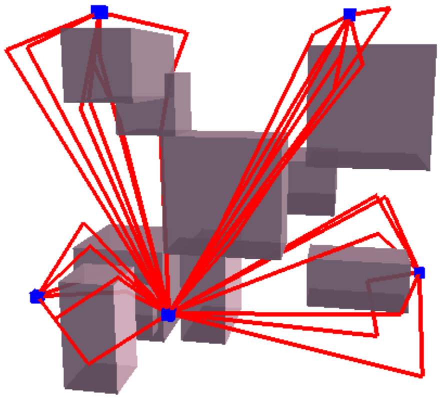
UCSD advanced robotics and controls lab



Active Continual Learning for Planning and Navigation

Ahmed Qureshi, Yinglong Miao, Michael C. Yip

Correspondence: a1quresh@ucsd.edu

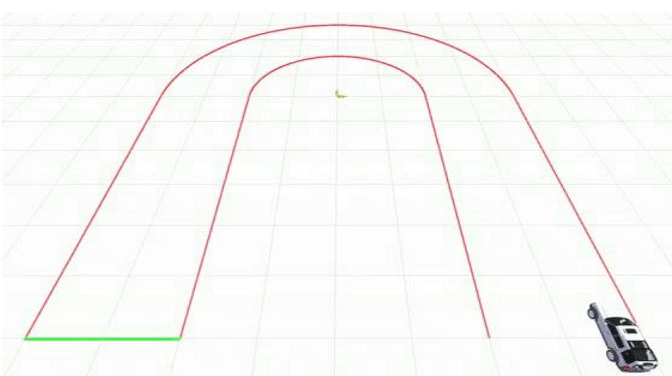
UC San Diego

Orignal paper: A.H.Qureshi, Y.Miao, A.Simeonov, and M.C.Yip. “Motion Planning Networks: Bridging the Gap Between Learning-based and Classical Motion Planners”, IEEE Transactions on Robotics (TRO), accepted.

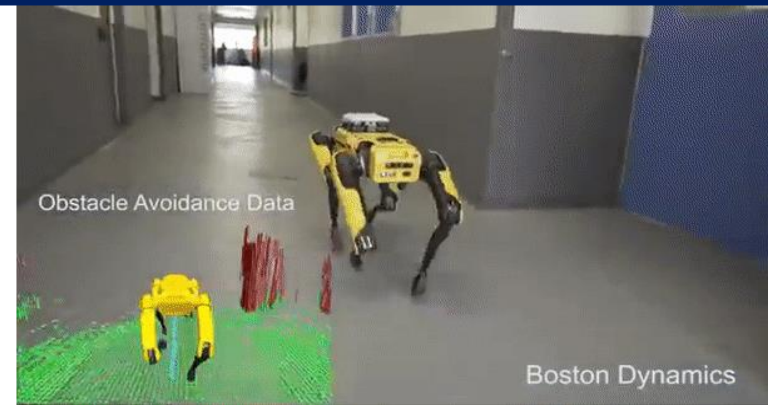
Motion Planning

Find a path that satisfies all constraints between the given start and goal configurations.

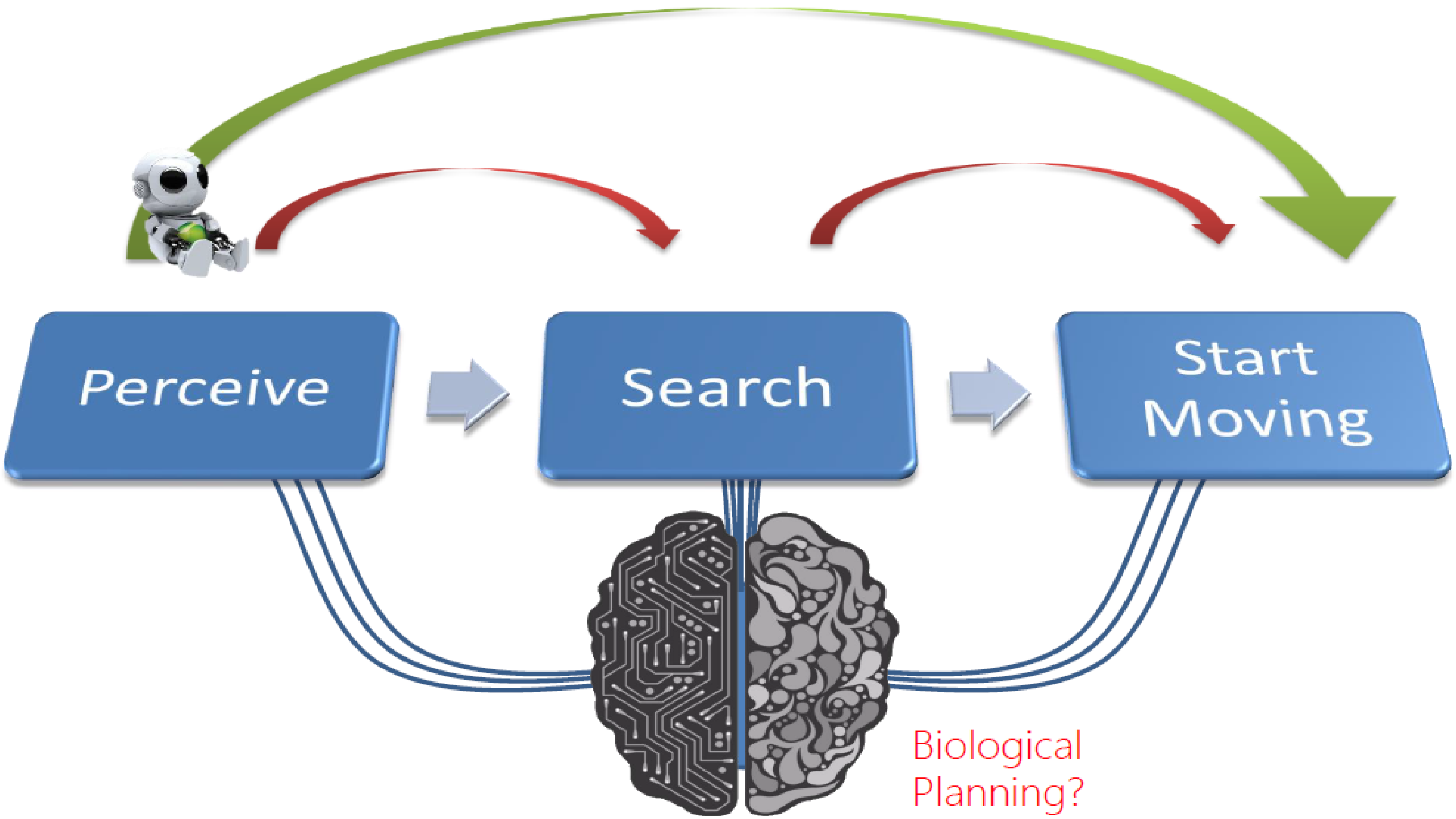
- Collision Avoidance
- Dynamics
- Kinematics (e.g., end-effector)



Karaman et. el, 2011

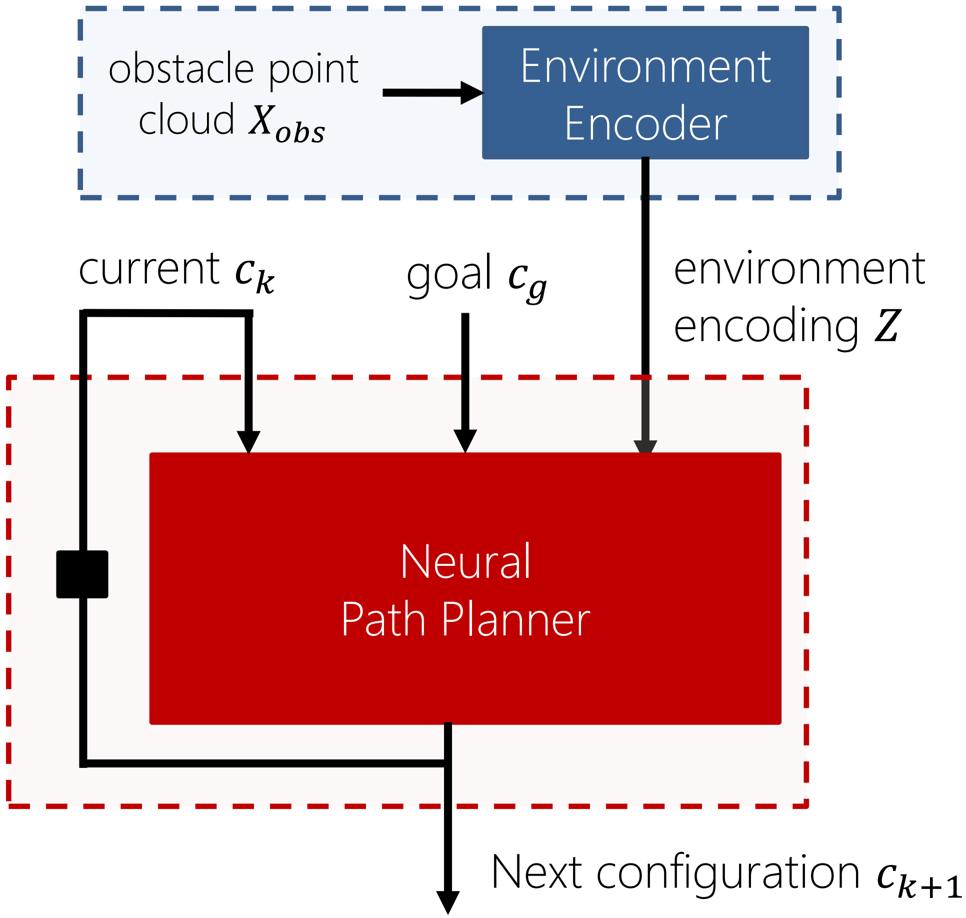


Sequence in Robot Thinking



MPNet: Motion Planning Networks

Encoder Network (Enet):
- Input: obstacles point cloud $x_{obs} \in \mathbb{R}^d$
- Output: Embedding $Z \in \mathbb{R}^m$
- 3D CNN (Preprocess point-cloud to voxel)
- Feed forward neural network



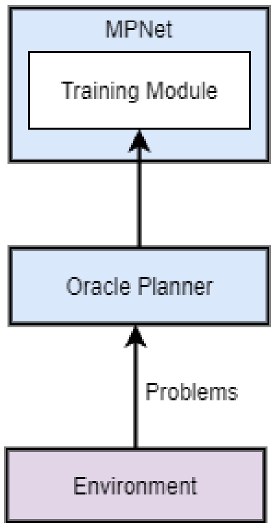
Planning Network (Pnet):
- Input: $\mathbf{Z}, c_t, c_T$
- Output: $\hat{c}_{t+1} \leftarrow \mathbf{PNet}(c_t, c_T, Z)$
- Stochastic feed-forward neural network

Training Methods:
- Batch offline learning
- Active continual learning

Planning Algorithm:
- Bidirectional iterative planning method.
- Informed sampler integrated with SMPs

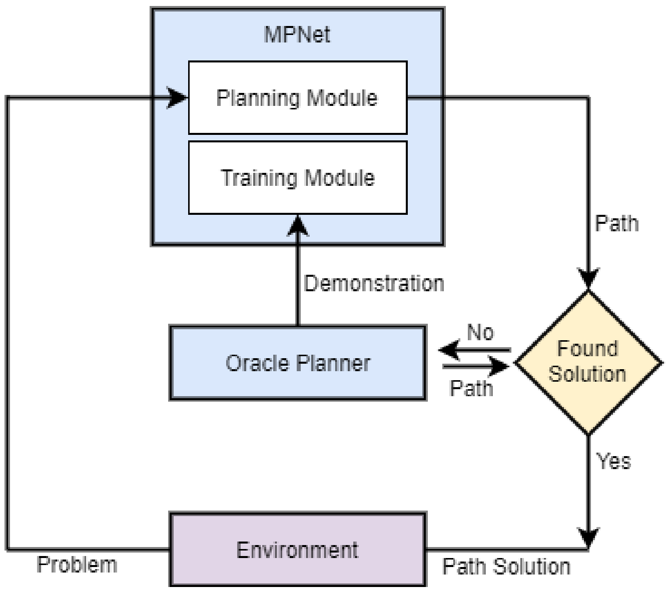
MPNet: Training Methods

Batch Learning



$$\frac{1}{N_p}\sum_{j,t} || \hat{c}_{j,t+1} - c_{j,t+1}||^2$$

Active Continual Learning

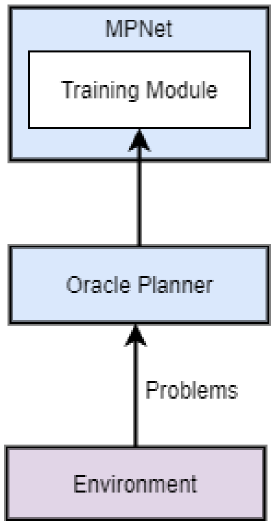


- $\min_{\theta} l\left(f_\theta^t(s), y\right) \; s.t.$
  $$\mathbb{E}_{(s,y)\sim M}\left[l\left(f_\theta^t(s), y\right)\right] \leq \mathbb{E}_{(s,y)\sim M}\left[l\left(f_\theta^{t-1}(s), y\right)\right]$$
- $\langle g, g_M \rangle = \left\langle \nabla_\theta l(f_\theta(s), y), \mathbb{E}_{(s,y)\sim M} \nabla_\theta l(f_\theta(s), y) \right\rangle$
- if $\langle g, g_M \rangle < 0$:
  $$\min_{g'} \|g - g'\| \; s.t. \; \langle g', g_M \rangle \geq 0$$

Lopez-Paz et al., 2017

MPNet: Training Methods

Batch Learning



$$\frac{1}{N_p}\sum_{j,t} || \hat{c}_{j,t+1} - c_{j,t+1}||^2$$

Active Continual Learning

$$\langle g, g_M \rangle = \left\langle \nabla_\theta l(f_\theta(s), y), \mathbb{E}_{(s,y)\sim M} \nabla_\theta l(f_\theta(s), y) \right\rangle$$
$$\boldsymbol{if}\ \langle \boldsymbol{g}, \boldsymbol{g}_{\boldsymbol{M}} \rangle < \mathbf{0}:$$
$$\min_{g'} ||g - g'||\ s.t.\ \langle g', g_M \rangle \geq 0$$

Learning from streaming data:

1- Given a feasible trajectory σ

2- M ←Update episodic memory (σ)

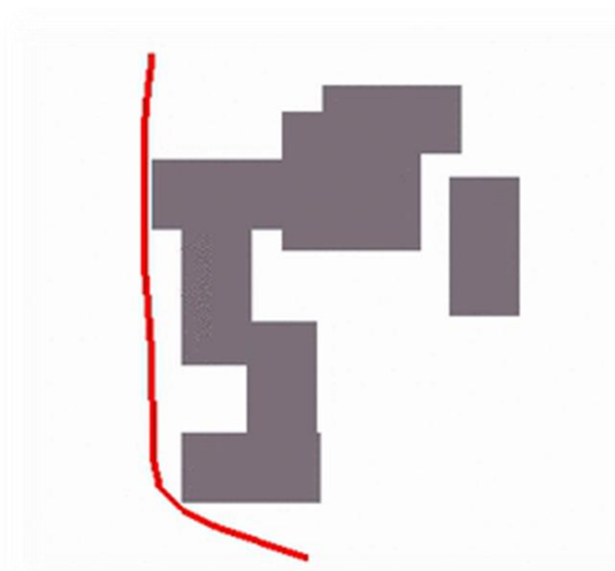
3- $g_M \leftarrow \mathbb{E}_{(s,y)\sim M} \nabla_\theta l(f_\theta(s), y)$

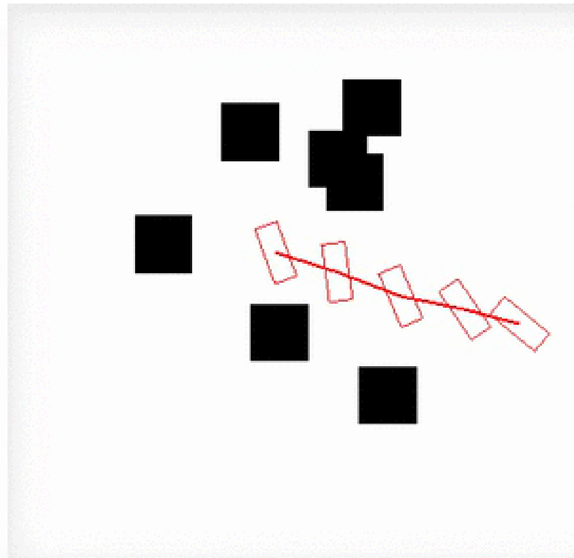
4- $g \leftarrow \mathbb{E}_{(s,y)\sim \sigma} \nabla_\theta l(f_\theta(s), y)$

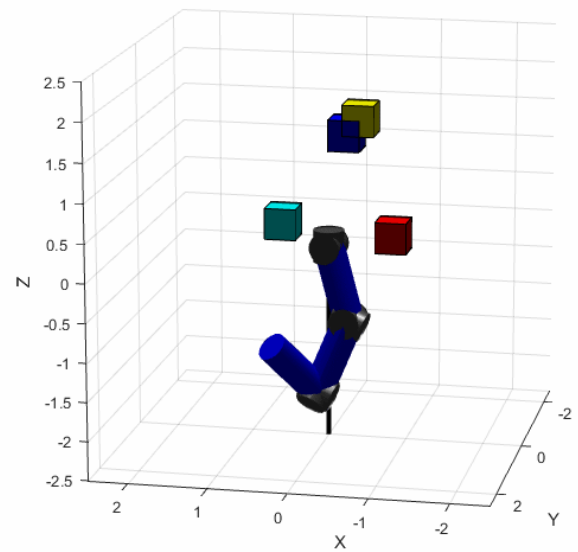
5- project g to g' using above eq

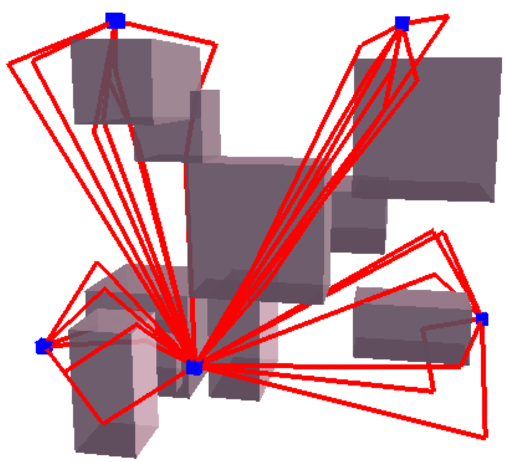
6- Update MPNet parameters θ with g'

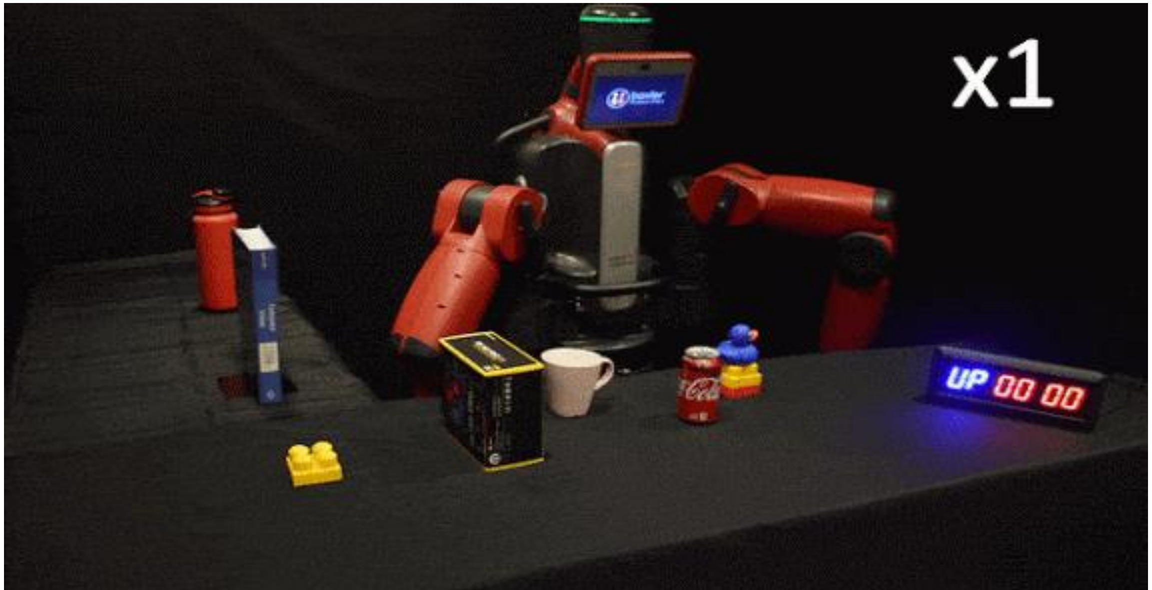
MPNet: Online Path Generation



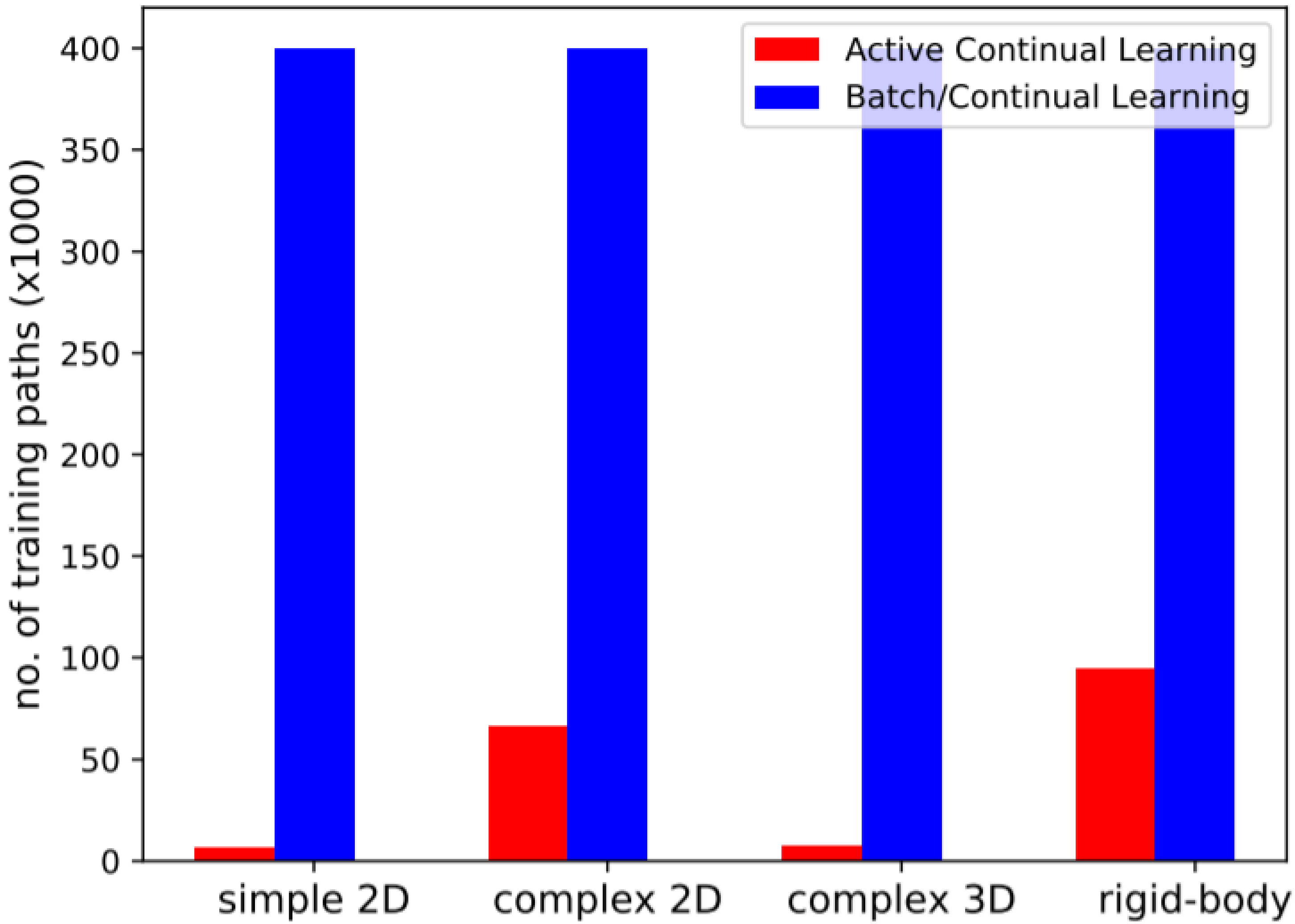








MPNet: Data Efficient Active Continual Learning



Thank you