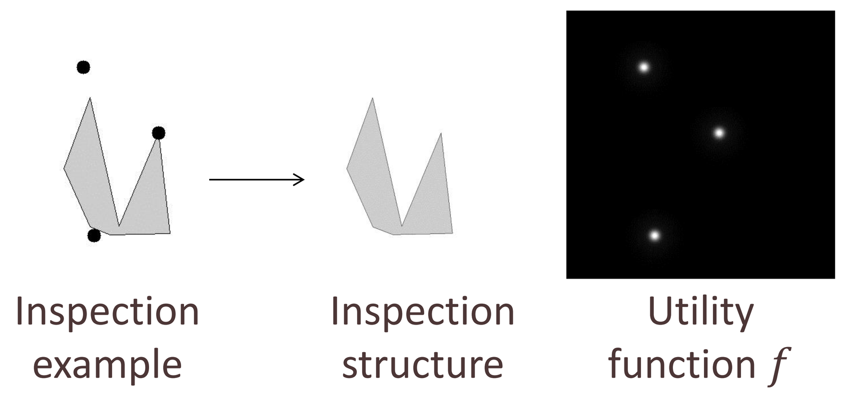
Policy learning from demonstration for autonomous inspection

Assia Benbihi, Matthieu Geist, Cédric Pradalier
abenbihi@georgiatech-metz.fr

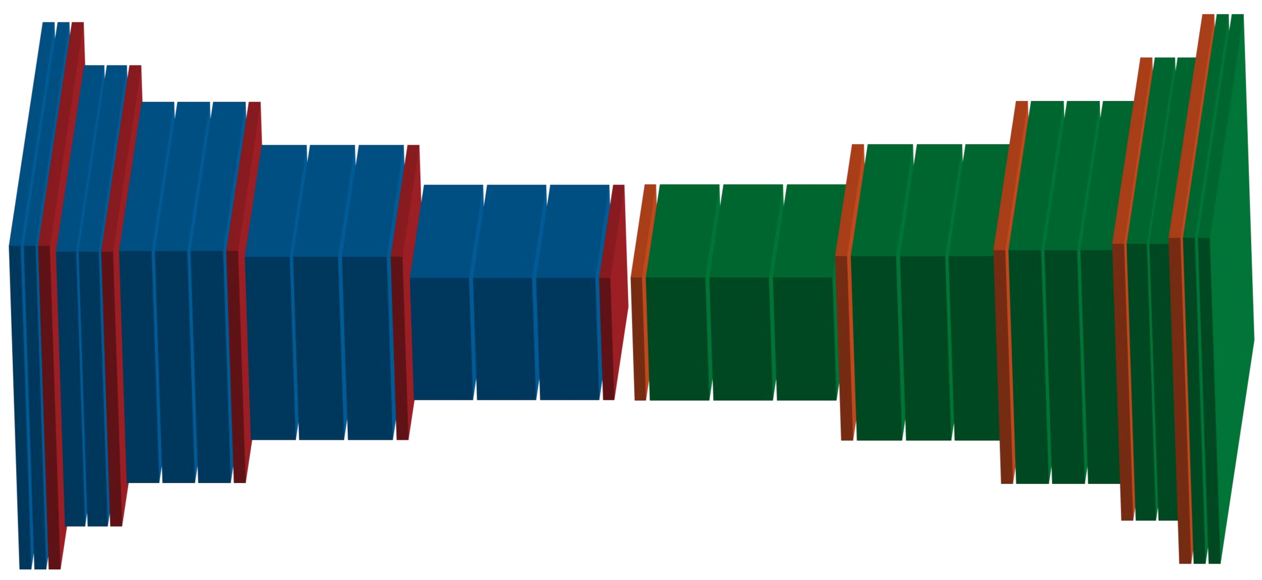
Use Case: Learn inspection utility function policy from expert inspection examples

1. Learning from demonstration



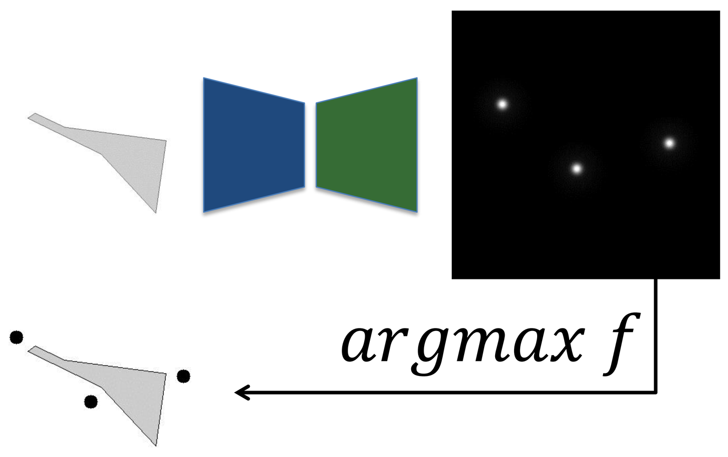
Gather utility function examples following an inspection pattern (e.g. Exterior Art Gallery Problem)

2. Train utility function policy Π



The policy is modeled by a convolutional neural network.

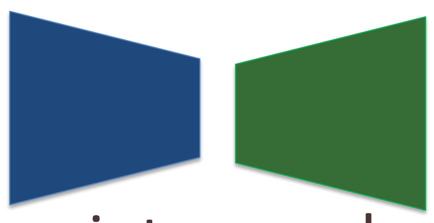
3. Inference



The network produces a utility function with the same maxima patterns

Autonomous feature extraction and utility function parametric learning

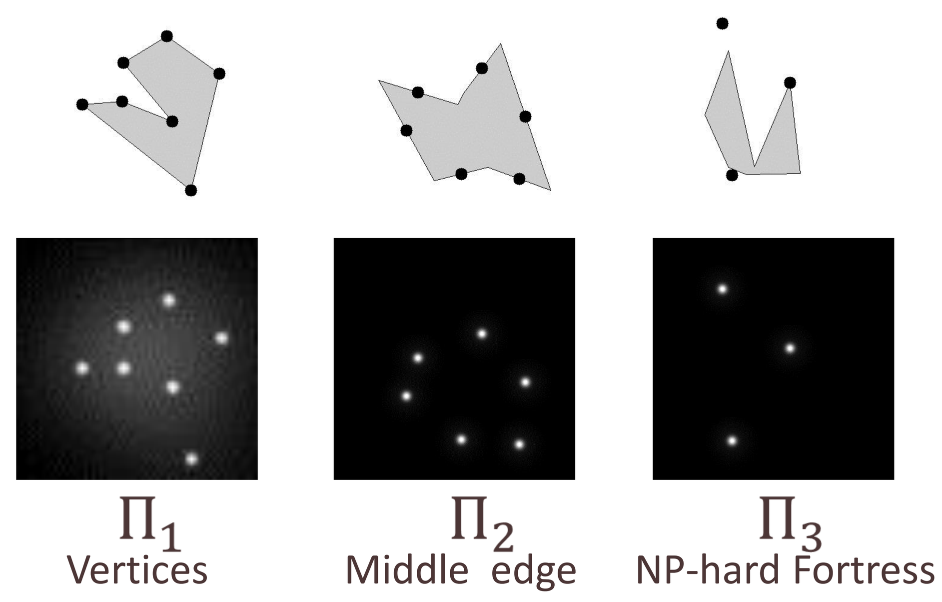
Encoder-Decoder architecture



The **encoder** autonomously extracts relevant features from the inspection examples into a reduced representation space. The **decoder** part generates an image from this reduced feature vector.

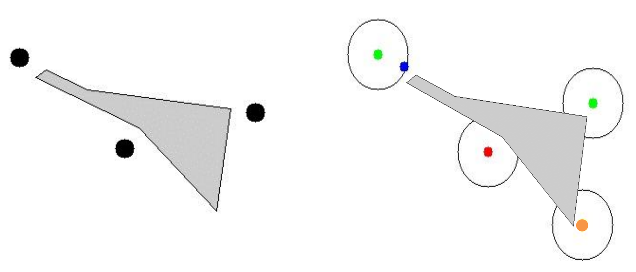
Data representation

Instead of hand-crafting a utility function, we learn it from demonstration. This method is evaluated on three inspection patterns. Utility function are represented with saliency maps.



Geometric similarity metrics

Equality between ground-truth and inferred inspection viewpoints.



$$recall = \frac{TP}{P}$$

$$precision = \frac{TP}{TP + FP}$$

TP True Positive
TP2 2° True Positive
FP False Positive
TN True Negative

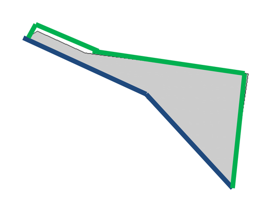
Semantic similarity metrics

Equality between inspection results when utility functions are different. Specific to the inspection pattern.

Example: Pattern Π_3 aims at covering the exterior perimeter.

The relevant metrics are cov_{model}, $cov_{regular}$.

p_X perimeter covered by X with $X \in \{model, expert, regular\}$.



$$cov_{model} = \frac{p_{model}}{p_{expert}}$$

$$cov_{regular} = \frac{p_{random}}{p_{expert}}$$

Experiments and results

Simulation results

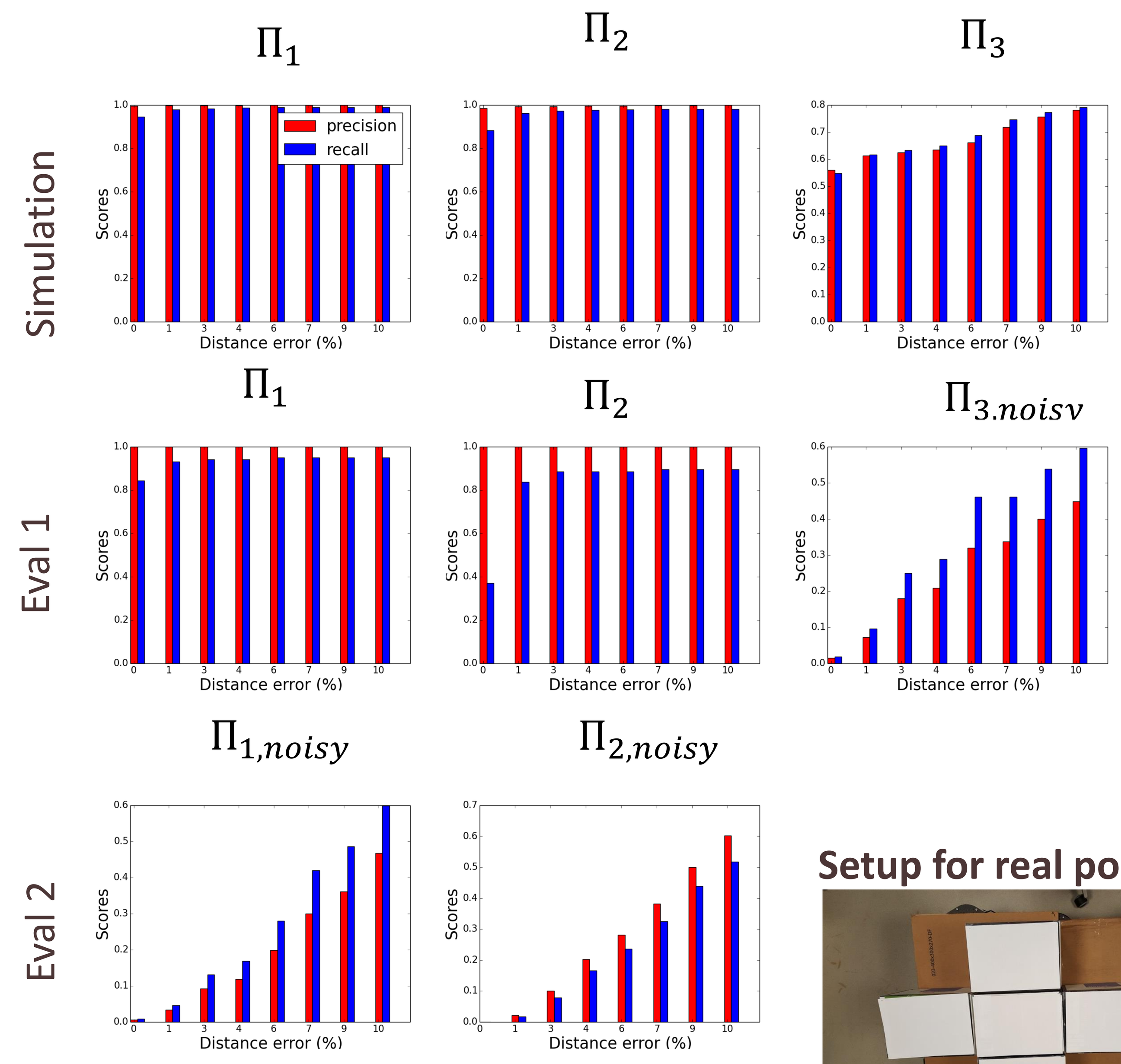
Geometric similarity increases as the inspection pattern becomes simpler. For Π_3, semantic similarity achieves 85% even when geometric similarity is low.

Experimental results

Policies trained on noisy simulation data maintain at least half of the performance on real data.

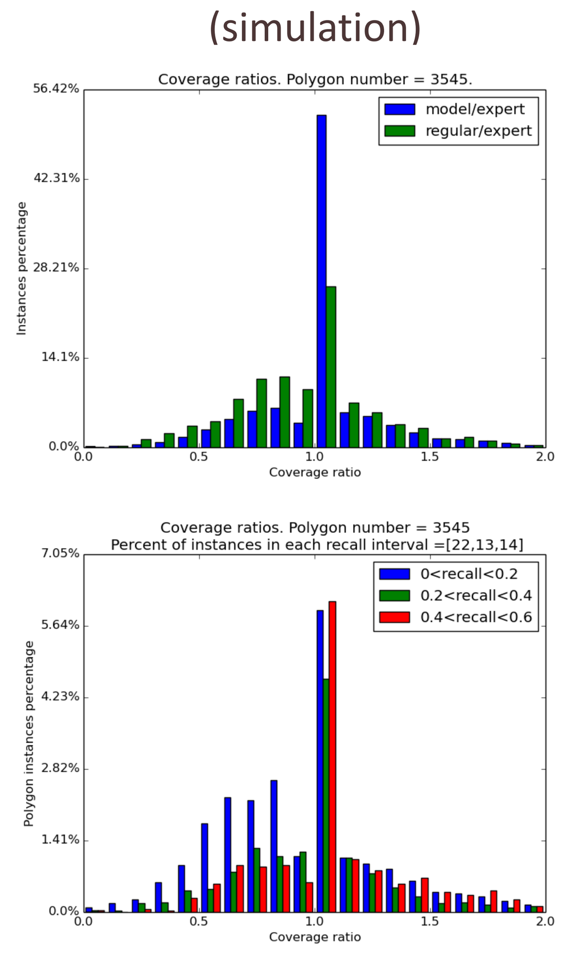
Simulation: Infer on simulated polygons
Eval 1: Inference on polygon approximation of real objects
Eval 2: Inference on real object contours

Geometric similarity



Semantic similarity

(simulation)



Setup for real polygonal structures