# Sim2Real Object-Centric Keypoint Detection and Description


Chengliang Zhong[12], Chao Yang[2], Jinshan Qi[3], Fuchun Sun[2], Huaping Liu[2], Xiaodong Mu[1], Wenbing Huang[2]

[1] Xi'an Research Institute of High-Tech [2] Tsinghua University [3] Shandong University of Science and Technology

Homepage: https://zhongcl-thu.github.io/rock/

## Summary of our work

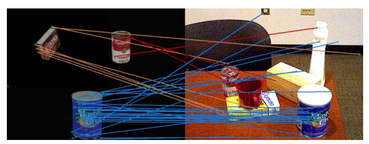
In this paper, we propose a novel conception: *object-centric* keypoint detection and description, in contrast to the conventional scene-centric setting. To be specific, it contains three main contributions:

- We are the first to raise the notion of object-centric keypoint detection and description, which better suits the object-level tasks;
- We develop a novel sim2real training method, which enforces uncertainty, intra-object salience/inter-object distinctness, and semantic consistency;
- Experiments on image matching and 6D pose estimation verify the encouraging generalization ability of our method from simulation to reality.

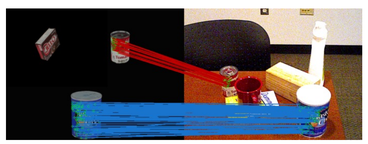
## Background

Keypoint detection and description play a central role in computer vision. Most existing methods are initially targeted on image-level/scene-centric tasks, making them less adaptive for other more fine-grained problems, e.g., object-level matching or pose estimation.

If we apply the previous methods (such as R2D2) straightly on synthetic (CAD model) and real image matching, the detected keypoints from the scene image usually contain not only the desired points on the target object, but also those unwanted points located in the background that share a similar local texture with the object CAD model.



**Scene-centric**



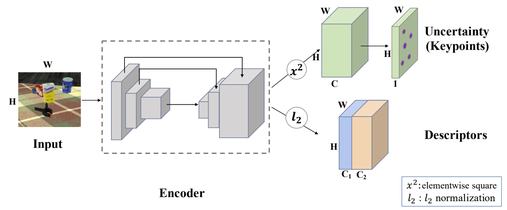
**Object-centric**

Beyond keypoint detection and description, the proposed object-centric formulation further teaches the algorithm to identify which object each keypoint belongs to. Figure. 1 depicts that the object-centric method accurately predicts the object correspondence (different colors) and matches the keypoints on different objects between the scene image and the CAD model.

## Our method

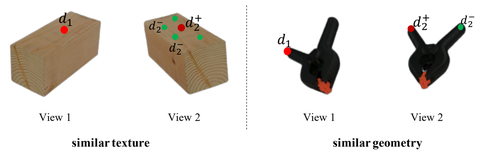
### ➢ Network Architecture

Given an input image, the detector outputs a **non-negative confidence map**, denoted as sigma, called a repeatability map in R2D2. If a pixel's confidence is above a certain threshold, it will be considered a keypoint.




In contrast to R2D2, there are two subparts of each description vector, one for intra-object salience, and the other one for inter-object distinctness.

### ➢ Contrastive learning with uncertainty



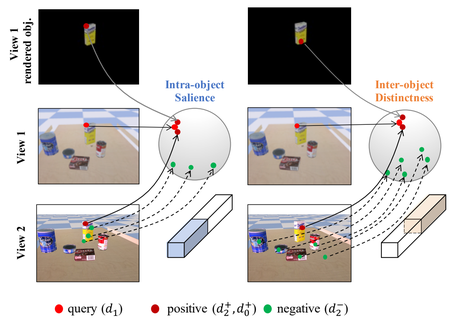

**InfoNCE Loss**

$$\mathcal{L}_c(d_1, d_2^+, \mathbb{D}_2^-) = -\log \frac{\exp(d_1 \cdot d_2^+/\tau)}{\exp(d_1 \cdot d_2^+/\tau) + \sum_{d_2^- \in \mathbb{D}_2^-} \exp(d_1 \cdot d_2^-/\tau)}$$

**InfoNCE Loss + uncertainty**

$$\mathcal{L}_d(I_1, I_2) = \frac{1}{M} \sum_{i=1}^{M} \frac{\mathcal{L}_c(d_1^i, d_2^{i+}, \mathbb{D}_2^{i-})}{(\sigma_1^i)^{-1}} + \log(\sigma_1^i)^{-1}$$

### ➢ Disentangled descriptor learning




The middle and bottom rows denote the synthetic scenes from two different viewpoints. The top line renders the object with a clean background at view 1. We decouple the descriptor into two parts for learning the intra-object salience (first column) and the inter-object distinctness (second column).

**Intra-object salience**

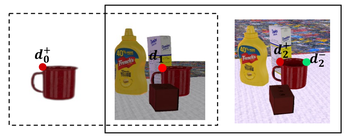
$$\mathcal{L}_{d_s}(I_1, I_2) = \frac{1}{M} \sum_{i=1}^{M} \frac{\mathcal{L}_c(d_1^i, d_{2,s}^{i+}, \mathbb{D}_{2,s}^{i-})}{(\sigma_1^i)^{-1}} + \log(\sigma_1^i)^{-1}$$

**Inter-object distinctness**

$$\mathcal{L}_{d_c}(I_1, I_2) = \frac{1}{M} \sum_{i=1}^{M} \mathcal{L}_c(d_1^i, d_{2,c}^{i+}, \mathbb{D}_{2,c}^{i-})$$

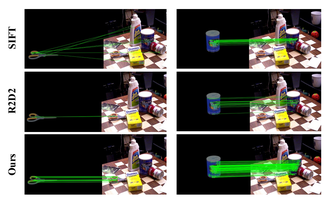
### ➢ Semantic consistency

The local region of the yellow box behind the red dot could be used as a shot-cut reference for the distinctness between the red and green points, which is NOT what we desire. So we render the cup according to its pose in I1 and remove all other things, leading to image I0. We perform contrastive learning by further taking the positive from I0 into account.

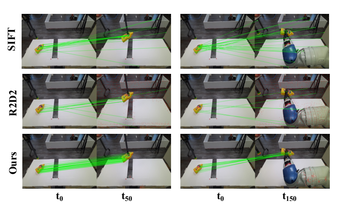



## Experiments

Comprehensive experiments on image matching and 6D pose estimation verify the encouraging generalization ability of our method from simulation to reality. Particularly for 6D pose estimation, our method significantly outperforms typical unsupervised/sim2real methods.




Visualization of synthetic-real image matching




Visualization of real-real image matching

Synthetic-real image matching results

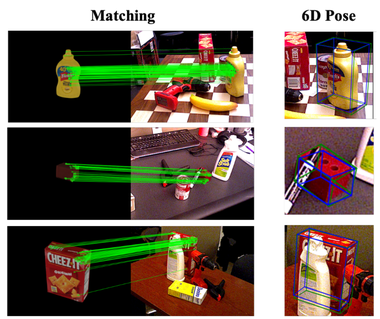
| Method | Dim | Kpts | MMA5 | MMA7 |
|---|---|---|---|---|
| SIFT | 128 | 16.9 | 24.2% | 30.1% |
| Superpoint | 256 | 15.7 | 33.6% | 43.8% |
| R2D2 | 128 | 20.0 | 34.8% | 44.6% |
| DISK | 128 | 15.8 | 28.2% | 35.1% |
| Ours | 96 | **92.1** | **50.0%** | **57.2%** |

Weak/unsupervised 6D pose estimation

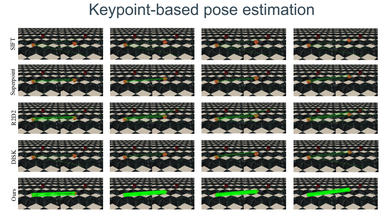
| | W-Supervised | Sim2Real/Unsupervised | | |
|---|---|---|---|---|
| Objects | Self6D(R) | Self6D | PoseCNN | Ours |
| mustard_bottle | **88.2** | 73.7 | 3.7 | 72.8 |
| tuna_fish_can | 69.7 | 26.6 | 3.1 | **73.5** |
| banana | 10.3 | 4.0 | 0.0 | **28.5** |
| mug | 43.4 | 23.9 | 0.0 | **45.7** |
| power_drill | 31.4 | 21.4 | 0.0 | **76.5** |
| ALL | 48.6 | 29.9 | 1.4 | **59.4** |

Matching evaluation on unseen objects

| Method | Seen class | | Unseen class | |
|---|---|---|---|---|
| | Kpts | MMA5 | Kpts | MMA5 |
| SIFT | 23.3 | 32.5% | 16.8 | 29.6% |
| R2D2 | 21.2 | 61.3% | 16.1 | 49.3% |
| Ours | **90.5** | **65.4%** | **75.9** | **61.3%** |




Keypoint-based pose estimation




Keypoint matching of unseen object