Dense object Nets: Learning dense visual object descriptor by and for robotic manipulation

Gabriel Iturralde Duenas
Contributions

- Introduce dense descriptors as a representation useful for robotic manipulation.
- Self-supervised dense visual descriptor learning can be applied to a wide variety of non-rigid object and classes.
- It can be learned quickly (20 min).
- Enables new manipulation tasks.
- Provided general training techniques for dense descriptors with good performance in practice.
Combination of global descriptors framework. This is described with ResNet-50 backbone. Each of the $n$ global descriptor branch outputs a $k$-dimensional vector that is concatenated into the combined descriptor loss.

Robosuite: simulation framework for robot learning powered by MuJoCo physics engine. It provides a modular design for creating robotic tasks.

Methods: Self-supervised pixelwise contrastive loss

• Training is performed in a Siamese fashion. A pixel that is the best match from image $I_a$ (that is sampled from an RGBD video) is a true match of a pixel in image $I_b$ if they correspond to the same vertex. $f()$ is the dense descriptor mapping and $D()$ is the L2 distance between a pair of pixel descriptors. $D()$ is defined as:

$$D(I_a, u_a, I_b, u_b) = \| f(I_a)(u_a) - f(I_b)(u_b) \|_2.$$  

• The loss function is intended to reduce the sum of matches and non-matches descriptors. It tries to minimize the distance between descriptors corresponding to a match, while non-matching descriptors should be at least a $M$ distance apart.

$$\mathcal{L}_{\text{matches}}(I_a, I_b) = \frac{1}{N_{\text{matches}}} \sum_{N_{\text{matches}}} D(I_a, u_a, I_b, u_b)^2$$

$$\mathcal{L}_{\text{non-matches}}(I_a, I_b) = \frac{1}{N_{\text{non-matches}}} \sum_{N_{\text{non-matches}}} \max(0, M - D(I_a, u_a, I_b, u_b))^2$$

$$\mathcal{L}(I_a, I_b) = \mathcal{L}_{\text{matches}}(I_a, I_b) + \mathcal{L}_{\text{non-matches}}(I_a, I_b)$$
Overview of the data collection and training: a) automated collection with an arm robot. b) change detection using the dense 3D reconstruction. c) – f) matches depicted in green, non-matches depicted in red.
Training for multi object descriptors

- **Object masking via 3D change detection:** training test showed that models focused on the objects rather than the background were more efficient.

- **Background domain randomization:** learned descriptors were enforced to not be reliant on the background for cross-scene consistency.

- **Hard negative scaling:**

\[
N_{\text{hard-negatives}} = \sum_{N_{\text{non-matches}}} \mathbb{1}(M - D(I_a, u_a, I_b, u_b) > 0)
\]

\[
L_{\text{non-matches}}(I_a, I_b) = \frac{1}{N_{\text{hard-negatives}}} \sum_{N_{\text{non-matches}}} \max(0, M - D(I_a, u_a, I_b, u_b))^2
\]

- **Data diversification and augmentation:** diversity was strongly considered, and data augmentation was achieved by using random end-effector rotations and varying light conditions.
Multi object dense descriptors

- **Cross-object loss**: it was implemented to ensure that different objects occupy different subsets of a descriptor space.
- **Direct training on multi-object scenes**: pixelwise contrastive loss provide the ability to directly train on multi-object cluttered scenes without any individual object masks.
- **Synthetic multi-object scenes**: this can be created by layering masks
Experiment setup

- Raw data was collected with an RGDB video of an object.
- 7 DoF robot arm Kuka IIWA LBR.
- TDSF was used for dense reconstruction and SLAM method was used to collect data that not require a calibrated robot.
- Training dense descriptors followed the single object within scene, different object across scene, multi object within scene and synthetic multi object.
Results: single-object dense descriptors

- Learned object descriptors can be consistent across deformation, b)-d) and across object classes.
- For each a) and b)-d) RGB images are in the top and the descriptor images at the bottom.
- e)-f) shows that we can learn descriptors from for low texture objects.
Results: multi-object dense descriptors

a) Description of the different types of networks.
b) Plots the class descriptor of the L2 pixel distance between the best match and the true match. In 93% of image pairs the normalized pixel distance is less than 13%.
c) Plots the class descriptor of the fraction of the best match pixels that are closer in descriptor space than the true match.
Results: selective class generalization

- Comparison of training without any distinct object loss (a) vs. using cross-object loss (b).
- In a) 100% of training iterations applied cross-object loss and single-object scene loss. For b) 50% of the training iterations applied object loss.
- c) shows the L2 pixel distance between a best match and a true match for different number of descriptors.

\[
\hat{u}_b = \arg\min_{u_b \in I_b} D(I_a, u^*_a, I_b, u_b)
\]
Results: selective class generalization

- Depiction of “grasp specific point” demonstrations. For each the user specifies a pixel, and the robot automatically grasps the best match in test configuration. “Right ear” is an example of the ability to break symmetry on symmetrical objects.
Limitations

• The performance of dense objects nets were not compared with any other learned dense visual object descriptors algorithm, either for learned descriptors, self-supervised visual learning robots and robot learning for specific tasks.
• The result are biased.
• The M distance parameter for a non match can be misinterpreted and lead to an error amplification, considering that you are squaring both terms of the cost function.
• Grasping tasks should be also evaluated with target achieving performance and precision and point targeting.
Related work

Related work

Collaboration between Maestro team and CNBI Lab for EEG Control of Exoskeleton