Supplementary Materials
Inpaint2Learn: A Self-Supervised Framework for Affordance Learning

1. Inpaint2Learn Data Generation

Given an RGB image in the wild, we first use an instance segmentation network Mask-RCNN[2] to segment the object/person of interest. Then we cut the instance out and fill the hole with an image inpainting network ProFill[5]. If multiple objects/humans exist in an image, we cut out one instance at a time and inpaint the hole, yielding multiple inpainted images from the original image. In the case of human affordance prediction, we run a pose estimator Alphapose[1] on the original image to obtain pseudo ground truth pose for the person. Please see Figure 1 for an illustration of our Inpaint2Learn data generation pipeline.

2. Data Pre-screening for Human Pose Labels

We pre-screen the raw data based on Mask-RCNN detection results and Alphapose predictions. Specifically, data points satisfying the following four criteria are selected:

1) Mask-RCNN detects no more than 10 people in the image. We find empirically that crowded scenes yield small, highly occluded human bodies, which make inpainting hard.

2) The detected bounding box area is no less than 1% and no more than 65% of the area of the original image. This screens out zoomed-in images that usually focus on a small part of the human body and images where the human is too small for Alphapose to generate accurate results.

3) At least 60% of the 25 joints are detected, and at least 3 out of the 5 facial joints must be present. Alphapose predictions yield confidence score for each of the 25 joints. We use a confidence score threshold of 0.05 to determine if a joint is present, following Alphapose's convention. This removes images with highly occluded bodies and ensures the head is always present.

4) If Alphapose detects multiple humans, we keep the one with the largest bounding box area. In the preprocessing stage, we mask out the regions outside the predicted bounding boxes on the origianl images and feed the resulting images to Alphapose, which not only predicts joint locations and confidence scores, but also bounding boxes enclosing each detected human. In the event that the network detects multiple humans even on the masked images, we apply this rule to select the largest, most dominant human body.

3. Network Implementations and Training Details of Affordance Models

In this section, we discuss the training details of each affordance model presented in the paper.

3.1. Human Affordance Prediction

The training pipeline of our human affordance prediction model is demonstrated in Figure 3 in the main paper. The inference pipeline, demonstrated in Figure 2, is almost the same as in training, except the encoder is discarded. During inference, a random variable is sampled from the Gaussian distribution $\mathcal{N}(0, 1)$ and passed to a decoder, which takes a conditional input and decodes to the output space (either affine matrices or pose joints).

Next, we discuss the details of each network component in the pipeline. In Figure 3, we show an illustration of an encoder that first encodes a ground truth theta (an affine matrix) or a pose into a 4 dimensional latent code using three fully connected (FC) layers, then reparametrize the code and replicate it both spatially and channel-wise so that we have a noise of dimension $H \times W \times 4$. In the bounding box generator shown in Figure 4, the RGB image, segmentation, depth, and the tiled noise are concatenated channel-wise and fed into several convolutional layers to predict the scale (c_x, c_y) and the translation (t_x, t_y) in an affine matrix. Then, we use the predicted affine matrix to warp a canonical white mask to a mask representing a bounding box, where we have all ones within the box region and all zeros outside. For the pose generation pipeline, we first pretrain a VAE[4] model that embeds pose joints to a latent space of 16 dimensions and a pose heatmap renderer that maps pose joints to a heatmap the same dimension as the RGB image. As shown in Figure 5, the pose generator takes in the predicted bounding box mask, the RGB image, segmentation, depth and the tiled noise as input and predicts a 16 dim latent code for the pretrained pose VAE. The predicted code is first passed to the decoder in the pretrained pose VAE, then to the heatmap renderer to generate a heatmap of the predicted pose.

Finally, in Figure 6, we show the bounding box discriminator on the left and the pose discriminator on the right. The bounding box discriminator takes the concatenated bounding box, the RGB image, segmentation and depth as input, and learns to discriminate the predicted bounding boxes from the real bounding box distribution present in natural scenes. Similarly, the pose discriminator takes the pose heatmap, the bounding box mask, the RGB image, segmentation and depth as input, and learns to discriminate the predicted pose from the real pose distribution present in natural images.

For training details, the human bounding box module and the human pose module are first trained separately for 20 epochs with a learning rate of 2e-4, and then jointly trained for 5 epochs with a learning rate of 2e-5. We use Adam optimizer [3] for both the generator and the discriminator. We also use a batch size of 32.

3.2. Location2Object

In the Location2Object model, we use a ResNet-18 network to extract features and use two three-layered fully connected networks with ReLU activation and dropouts to predict the classification label and the bounding box. We use an Adam optimizer with a learning rate of 0.001 to train the network. We set the maximum number of epochs to 20 and the batch size to 32.

3.3. 6D Object Pose Hallucination

Similar to the Location2Object model, in the task of 6D object pose hallucination, we adopt a ResNet-18 network to extract features and use two three-layered fully connected networks with ReLU activation and dropouts to predict the translation and the rotation sequentially. We use an Adam optimizer with a learning rate of 0.001 to train the network. We set the maximum number of epochs to 100 and the batch size to 32.

4. Generated Pseudo Labeled Affordance Data

Please see Figure 7 for visualizations of the generated data for human pose affordance prediction and Figure 8 for visualizations of the generated data for the Location2Object task.

5. More Qualitative Results

More qualitative results for human affordance prediction, Location2Object and 6D object pose hallucination tasks can be found in Figure 9, 10, 11 and 12.

6. User Study

We demonstrate our user study interface for all of our affordance tasks in Figure 13, 14, and 15. In each figure, we show the survey instruction on the left and the survey question on the right. Please refer to these figures for more details.

References

[1] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[2] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[4] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[5] Yu Zeng, Zhe Lin, Jimei Yang, Jianming Zhang, Eli Shechtman, and Huchuan Lu. High-resolution image inpainting with iterative confidence feedback and guided upsampling. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020.

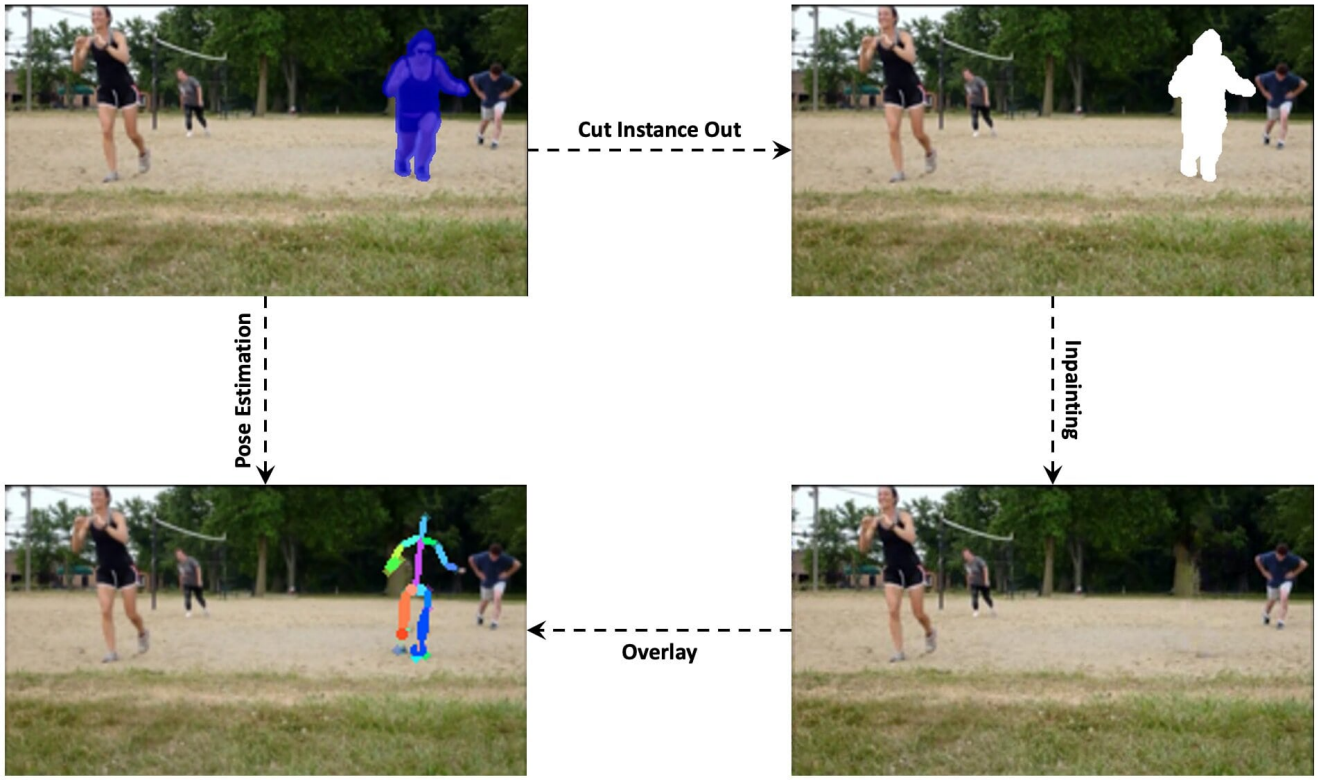


Figure 1. An illustration for our Inpaint2Learn data generation pipeline. First, we use an instance segmentation algorithm Mask R-CNN[2] to segment the object of interest, shown in the top left image. Then, we cut the instance out using the predicted mask, as shown in the top right. We use an image inpainting algorithm ProFill[5] to fill the hole (bottom right). We also run a pose estimator AlphaPose[1] to generate the pseudo ground truth pose for the person of interest. The bottom left shows our pose label visualized on the inpainted background.

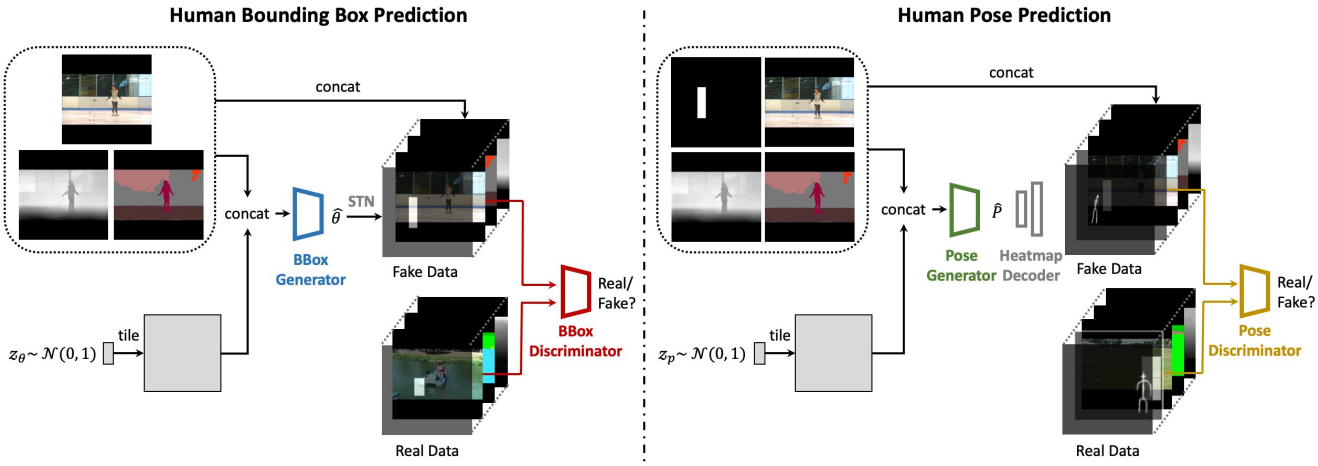


Figure 2. An illustration for the inference pipeline for the human affordance prediction model.

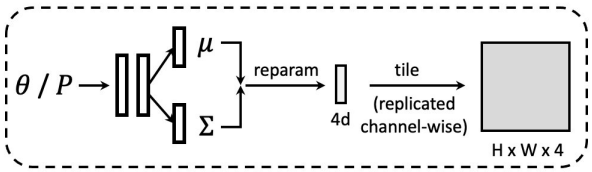


Figure 3. An illustration for the theta(affine matrix) / pose encoder.

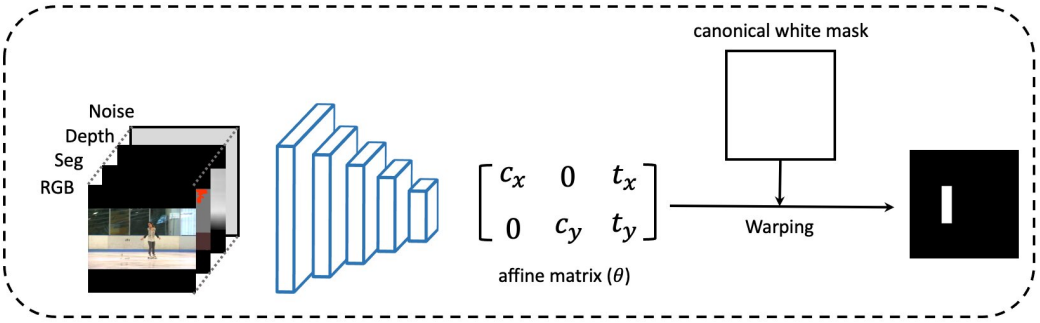


Figure 4. An illustration for the bounding box generator.

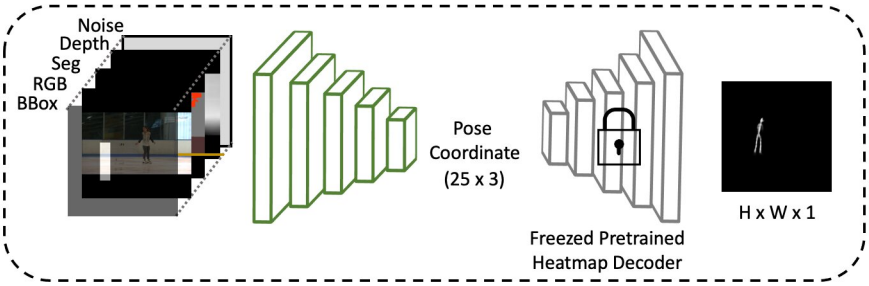


Figure 5. An illustration for the human pose generator.

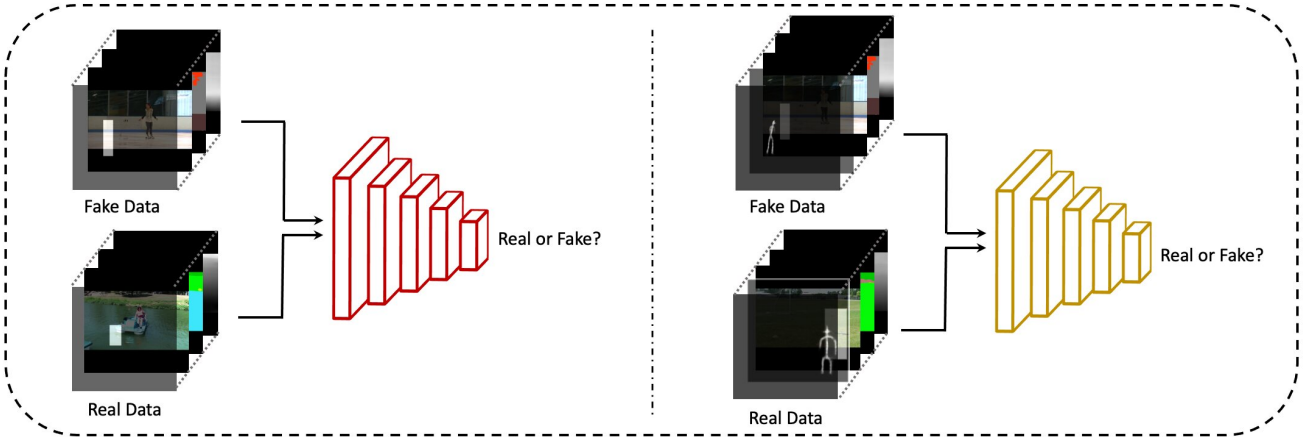


Figure 6. An illustration for the discriminators. **Left**: BBox Discriminator. **Right**: Pose Discriminator.

Figure 7. Generated training data visualization for human pose affordance learning. The first and the third columns are the original images, and the second and the fourth columns are the generated pseudo pose labels overlayed on the inpainted images.

Figure 8. Generated training data visualization for Location2Object learning. The first and the third columns are the original images, and the second and the fourth columns are the generated class and bounding box labels overlayed on the inpainted images.

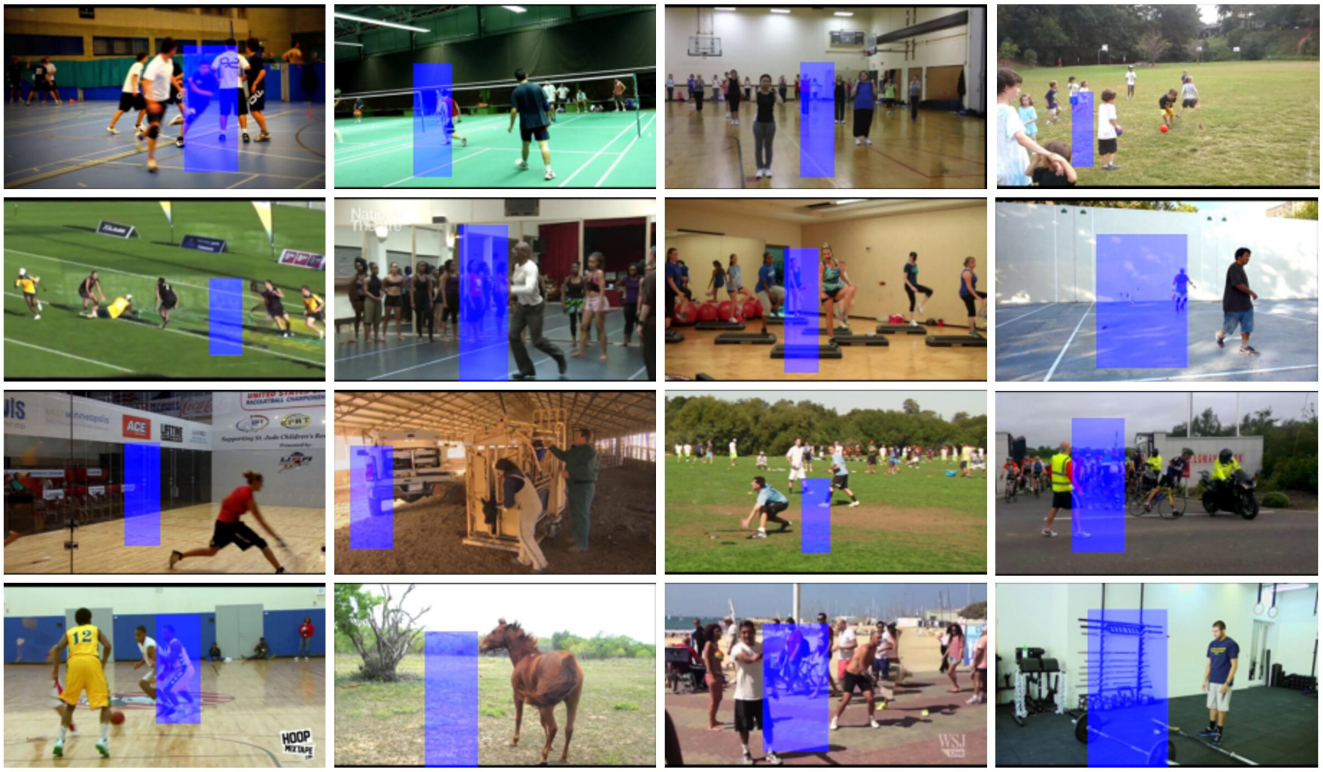


Figure 9. More qualitative results for predicted bounding boxes in human affordance prediction.

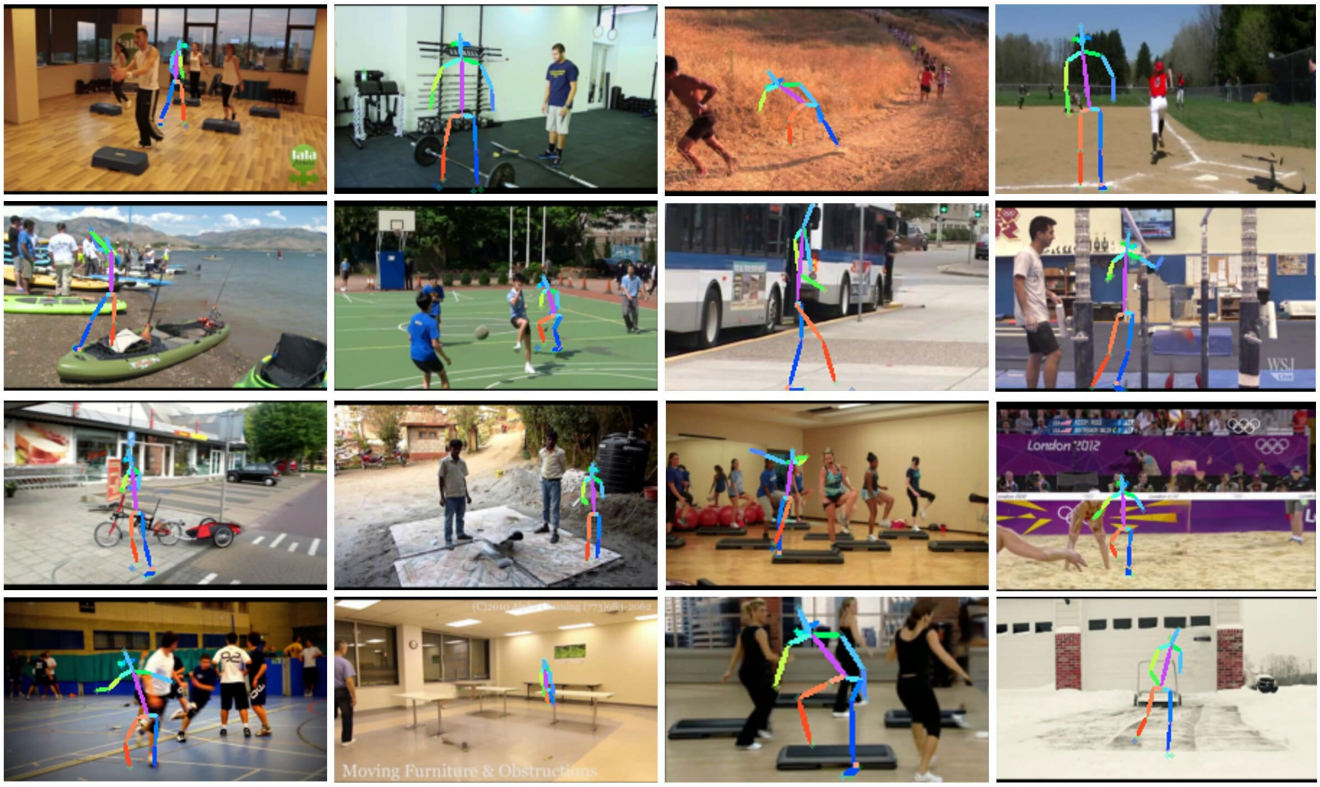


Figure 10. More qualitative results for synthesized human poses in human affordance prediction.

Figure 11. More qualitative results for Location2Object.

Figure 12. More qualitative results for 6D object pose hallucination.

We designed a machine that can predict what location and shape a human can be inserted and how his/her pose look like in an image. In the left image, the red circle indicates the specified location. In the left image, the blue bounding box shows the predicted location and shape. In the right image, the joints show the predicted human pose.

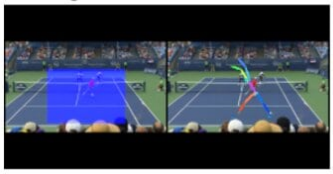
Box -> Good: The predicted location and shape is reasonable to insert a human in the image.
Box -> Bad: The predicted location and shape is not reasonable to insert a human in the image.
Pose -> Good: The predicted human pose is reasonable in the image.
Pose -> Bad: The predicted human pose is not reasonable in the image.

Below are few examples.

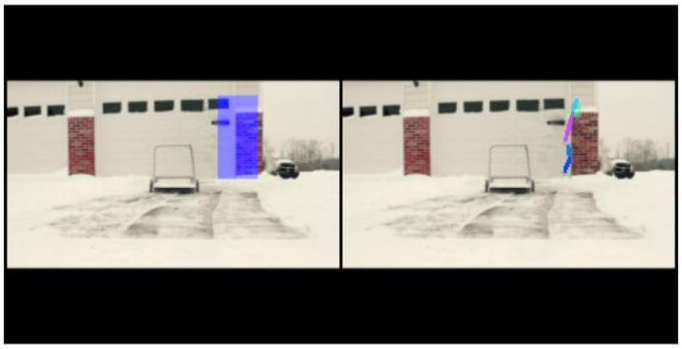
[**Box -> Good**] The predicted bounding box has both reasonable location and shape to insert a human
[**Pose -> Good**] The predicted human pose is reasonable in the image.

[**Box -> Bad**] The predicted bounding box has wrong location in the image.
[**Pose -> Bad**] The predicted human pose is not reasonable in the image, since the location is wrong.

[**Box -> Bad**] The predicted bounding box is too big in this image.
[**Pose -> Bad**] The predicted human pose is not reasonable in the image.



0% 100%



Please assess the prediction made by the model:

	Yes - it's a good prediction	No - it's a bad prediction
Is the predicted bounding box (location and shape) reasonable?	○	○
Is the predicted human pose reasonable?	○	○

Figure 13. User study interface for human affordance prediction. **Left**: Survey instruction. **Right**: Survey question example.

We designed a machine that can predict what object can be placed at a specific location in an image and the object's corresponding shape.

There are six objects that can be placed in the image:

1. a person
2. a car
3. a bicycle
4. a motorcycle
5. a traffic light
6. a stop sign

We gave the machine an image with a orange dot on it, and then told the machine to predict which of the six objects can be placed at the red dot, and the approximate size (length and width) of the object given by a rectangular box.

Here is an example. The image on the left is the original image we gave the machine. The orange dot in the image is where we tell the machine to make predictions about what object to place in the place of the dot. On the right side is the image with the machines prediction. The machine predicts that we can place a bicycle at the orange dot (object type is written in yellow on top of the yellow box). And the machine predicts that the shape of the bicycle should be approximately the size of the yellow box. It seems like the machine's predictions for both object type and shape size are reasonable.

This example is another example, where the machine has made reasonable predictions about the object type (the machine think we can place a car at the orange dot) but it did not make a good prediction about the shape and size of the car (the shape looks too narrow and too small for a car).

We want to know whether the machine's prediction is reasonable. So for each of the following examples, please look at the machine's prediction given an original image, and indicate whether the machine has 1) predicted a good/bad label and 2) predicted a good/bad shape for the object.

When you are ready, please start assessing the images on the next page!

0% 100%

Please assess the prediction made by the model:

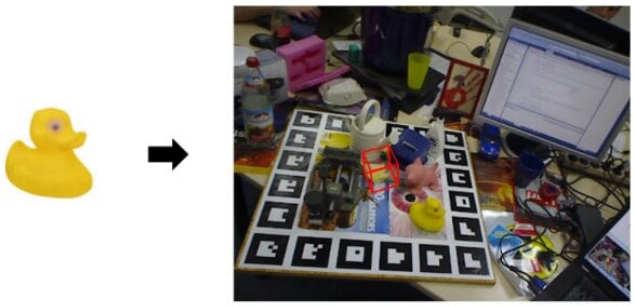
	Yes - it's a good prediction	No - it's a bad prediction
Is the predicted object type reasonable?	○	○
Is the predicted object size (length and width) reasonable?	○	○

Figure 14. User study interface for Location2Object. **Left**: Survey instruction. **Right**: Survey question example.

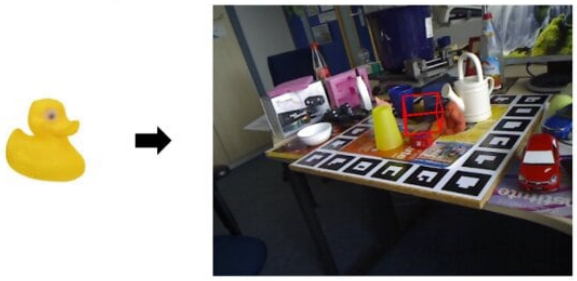
We designed a machine that can predict an given object can be inserted into an image. On the left is the given object, and on the right is the predicted 3D bounding box to place this object into the image. Please help us to assess whether the predicted bounding box has the right location (translation) and rotation.

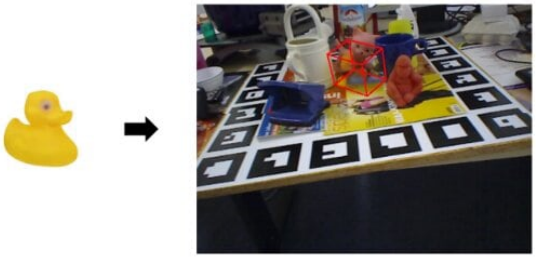
Location -> Good: The predicted location and shape is reasonable to insert a human in the image.
Location -> Bad: The predicted location and shape is not reasonable to insert a human in the image.
Rotation -> Good: The predicted human pose is reasonable in the image.
Rotation -> Bad: The predicted human pose is not reasonable in the image.

Below are few examples.
[**Location -> Good**] The predicted location to place this rubber duck is reasonable.
[**Rotation -> Good**] The predicted rotation of this rubber duck on the desk is reasonable.



[**Location -> Good**] The predicted location to place this rubber duck is reasonable.
[**Rotation -> Bad**] The predicted rotation to place this rubber duck is reasonable.





Please assess the prediction made by the model:

	Yes - it's a good prediction	No - it's a bad prediction
Is the predicted location reasonable?	○	○
Is the predicted rotation reasonable?	○	○

Figure 15. User study interface for 6D object pose hallucination. **Left**: Survey instruction. **Right**: Survey question example.