Demo2Vec: Reasoning Object Affordances from Online Videos

Kuan Fang*
Stanford University
kuanfang@stanford.edu

Te-Lin Wu*
University of Southern California
telinwu@usc.edu

Daniel Yang
University of Southern California
danielxy@usc.edu

Silvio Savarese
Stanford University
ssilvio@stanford.edu

Joseph J. Lim
University of Southern California
limjj@usc.edu

Abstract

Watching expert demonstrations is an important way for humans and robots to reason about affordances of unseen objects. In this paper, we consider the problem of reasoning object affordances through the feature embedding of demonstration videos. We design the Demo2Vec model which learns to extract embedded vectors of demonstration videos and predicts the interaction region and the action label on a target image of the same object. We introduce the Online Product Review dataset for Affordance (OPRA) by collecting and labeling diverse YouTube product review videos. Our Demo2Vec model outperforms various recurrent neural network baselines on the collected dataset.

1. Introduction

Humans often appeal to expert demonstrations when learning to interact with unseen objects. Through watching the demonstration by another person, one can understand the object affordances, *i.e.* functionalities of different parts and possible actions that can be taken. Upon seeing the same object in a different environment, humans can map the learned affordances onto the object and imitate the actions they observed from the previous demonstration. To teach a robot about how humans manipulate and interact with objects, previous methods learn this knowledge from simulated agent-object interactions [33, 23, 29], demonstrations observed by the robot camera in the robot workspace [22, 17] or demonstrations observed from a third-person viewpoint [13, 14]. Different forms of object affordances are learned from these demonstrations and used for tasks such as imitation learning and action prediction.

However, there exists a much richer data resource of human demonstrations on object affordances which can be utilized from the Internet. Specifically, there is a great number of product review videos uploaded onto YouTube and other video-sharing websites by product manufacturers and users. These videos cover diverse sets of object categories which people interact with in everyday life, including kitchenware, garage tools, consumer electronics, appliances, toys and so on. In each video, there is usually a human demonstrator (*e.g.* user or salesperson) showing the functionality in details through a sequence of actions on the product object. These videos provide large-scale, high-quality data for teaching robots about the functionality of the products and how people interact with them.

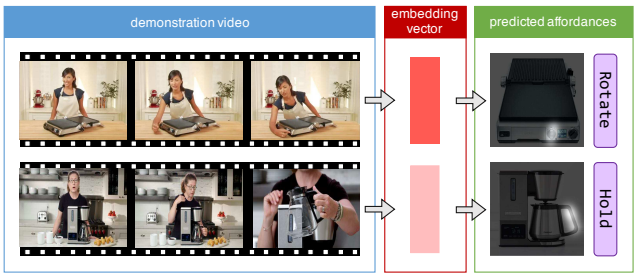


Figure 1. Our Demo2Vec model summarizes the demonstration video via the embedded vector. The embedded vector is used to predict object affordances (*i.e.* the interaction region and the action label) for the target image. The demonstration videos are from online product reviews on YouTube. More examples can be found at: https://sites.google.com/view/demo2vec/

Given human demonstrations from these product review videos, our goal is to learn to summarize the feature embedding of demonstration videos, and thus to predict the interaction region and the corresponding action label on a target image of the same object. Consider the griddle in Fig. 1. By watching the demonstrator turning on the griddle in the product review demonstration video, we aim to predict the action label as *rotate* and the heatmap centered around the knob in the middle on the target image. This problem is challenging for mainly two reasons: First, the

*indicates equal contribution, sorted in alphabetical order.

appearance of the object can have large variations between the demonstration video and the target image, which makes it difficult to transfer the learned knowledge between the two. Second, the interaction between the human and the object is usually very scarce across time and most video frames do not provide useful information for understanding the affordances. To tackle these challenges, we design the Demo2Vec model. The model is composed of a demonstration encoder and an affordance predictor. The demonstration encoder takes the demonstration video as input and encode it into a low-dimensional embedded vector. And the affordance predictor utilizes the embedded vector to predict the interaction region and the action label for the target image. The embedded vector summarizes the information of the human action and the object appearance from the observed demonstration video.

For training Demo2Vec, we introduce the Online Product Review dataset for Affordance (OPRA). Our dataset consists of 20,612 sets of video clips, corresponding product images, and annotations of both the interaction heatmaps and the action labels labeled by human annotators. These video clips are sourced from full-length videos of YouTube product review channels and encompass human-demonstrator interactions with a variety of common everyday objects like kitchenware, household appliances, and consumer electronics. Each target image can correspond to different interaction heatmaps when paired up with different demonstration videos, which covers most of the available functionalities. Action labels of possible actions are grouped into seven classes.

Our main contributions are:

- We propose the Demo2Vec model which extracts feature embeddings from demonstration videos, and predicts affordances of the same object by transferring the extracted knowledge onto a target image.
- We introduce the Online Product Review dataset for Affordance (OPRA). This is one of the first datasets providing a testbed for affordance reasoning based on demonstration videos in the wild.
- We evaluate the Demo2Vec on the newly introduced OPRA dataset. Our model outperforms a list of recurrent neural network baselines as shown in Sec. 5.1.

2. Related Work

Learning Affordances Previous works rely on RGB images and videos augmented with additional information, such as depth or estimated human poses, to learn affordances. Koppula et al. propose an algorithm to learn semantic labels, spatial regions, and temporal trajectories of interactions from labeled RGB-D videos [12, 13] using a skeleton tracker to extract estimated human poses. Zhu et al. perform 3D scene reconstruction from RGB-D videos which requires explicitly tracking the tool in use, the object, and the hand movements [32, 33].

Many RGB-D image-based approaches perform pixel-wise classification of a scene segmenting it into regions with different affordance classes. Roy et al. predict affordance maps with human-scale labels like *walkable* and *sittable* [21]. Srikantha et al. perform fully-supervised pixel-wise classification along with weaker forms of supervision such as key points and image-level annotations [25]. Nguyen et al. also predict object affordances as heatmaps and apply their method to a real humanoid robot for simple grasping tasks [18]. Other RGB image-based approaches obtain additional 3D information from estimated human poses. Yao et al. measure the relative poses of musical instruments and human players to cluster different types of interactions applied to the instruments [30]. Similarly, Kjellstrom et al. track hand poses and reconstruct them onto the object to determine object-action pairs [11].

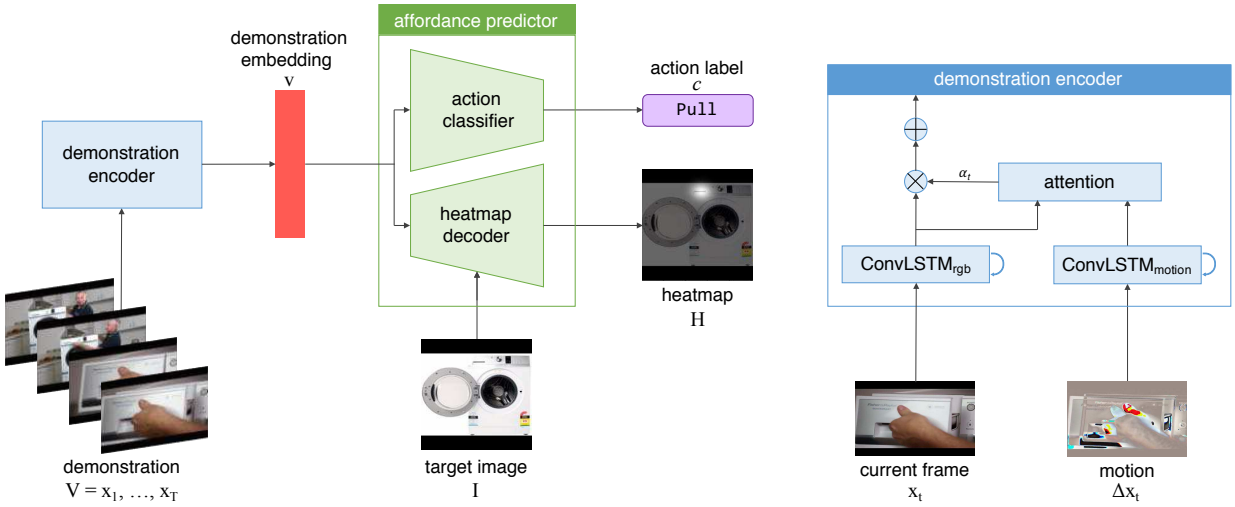
In contrast to these methods, our approach learns the affordances purely from RGB-only video demonstrations and does not require any additional information that the aforementioned methods rely on. Furthermore, our videos are more diverse in many aspects such as viewpoints, interactions being taken, and potentially occlusion of the demonstrator (or some parts of the target object), as these videos are directly scraped from the Internet.

Learning from Demonstrations (LfD) Imitation learning is a method to teach a learning agent to mimic policies from presumed expert demonstrations. Ross et al. propose DAGGER [20], an iterative algorithm which learns stationary deterministic policies from the expert policy. Duan et al. devise a one-shot imitation learning framework [3] to teach a robot to stack blocks using novel demonstrations during testing time. Stadie et al. design a neural network that learns from third-person demonstrations in simulations [26]. Ho et al. propose an algorithm based off generative adversarial networks [5] to learn the reward functions and devise new policy optimization update rules from expert trajectories [6].

In these scenarios, the demonstrations and the predictions are from the same domain. However, in our work, we aim to learn from on-line product review videos, and transfer the learned knowledge onto a target image.

3. Method

Our goal is to predict affordances (*i.e.* action labels and interaction regions) of an unseen object through an embedded vector of the demonstration video. The embedded vector summarizes the object appearance and the human-object interaction in the demonstration video. Specifically, we define the input and output of the model as follows:



(a) Model overview

(b) Demonstration encoder

Figure 2. (a) **Model overview.** Our Demo2Vec model is composed of a demonstration encoder and an affordance predictor. (b) **Demonstration encoder.** The demonstration encoder summarizes the input demonstration video into a demonstration embedding. The affordance predictor then uses the demonstration embedding to predict the interaction heatmap and the action label.

- **Demonstration**: The demonstration is a video $\mathbf{V} = \mathbf{x}_1, ..., \mathbf{x}_T$ which has T video frames. Each demonstration contains a single human-object interaction, while the same interaction can be applied on multiple regions (*e.g.* two handles of a pressure cooker). The camera viewpoint can change in each demonstration video.
- **Target Image**: The target image $\mathbf{I}$ contains the same object demonstrated in the video. The object appearances shown in the image and the video can be very different due to changes of the object status (*e.g.* open *vs.* close), camera viewpoints, backgrounds and other factors.
- **Interaction Heatmap**: Given a demonstration video $\mathbf{V}$, we use a heatmap $\mathbf{H}$ to represent the interaction region on $\mathbf{I}$. $\mathbf{H}$ is defined as a probability distribution over $\mathbf{I}$ and has the same size with $\mathbf{I}$.
- **Action Label**: Given a demonstration video, we predict the action label c. We group the action labels into 7 classes as described in Sec. 4.

The Demo2Vec model is shown in Fig. 2. The model is composed of a demonstration encoder and an affordance predictor. The demonstration encoder extracts the demonstration embedding $\mathbf{v}$ as a low-dimensional feature vector from the video $\mathbf{V}$. Given $\mathbf{v}$, the affordance predictor predicts the action label c and projects the interaction region onto the target image $\mathbf{I}$ to generate the heatmap $\mathbf{H}$.

3.1. Demonstration Encoder

The major challenge of learning the demonstration embedding is to extract useful visual cues about the human-object interaction. Usually the human-object interaction only happens in an instant, while in most video frames the demonstrator stays still and explain the functionality in words. In addition, there could be many distractions in the scene such as other objects on the table and the cluttered background.

To tackle these challenges, we propose a demonstration encoder model using convolutional LSTM networks (ConvLSTM) [7, 4, 27] and soft attention model [28]. For each video frame, two input modalities are used: the RGB image modality $\mathbf{x}_t$ at each time step t, and the motion modality $\Delta\mathbf{x}_t = \mathbf{x}_t - \mathbf{x}_{t-1}$ between t and $t-1$. We refer $\Delta\mathbf{x}_t$ as the motion modality since it captures the foreground motion while ignoring the static background. Both modalities are fed into ConvLSTM to extract the spatial and temporal information across time. On top of the ConvLSTM outputs, we utilize the temporal soft attention mechanism to aggregate the outputs from the ConvLSTM. The attention scores α_t is computed from the concatenation of image features and motion features. Finally, we obtain the demonstration embedding $\mathbf{v}$ by applying α_t onto the image features through element-wise multiplication, ie. $\mathbf{v} = \sum_{t=1}^{T} \alpha_t \odot x_t$, where T denotes the total time steps of the video, and $\odot$ indicates element-wise product, and also note that the summation here implies element-wise summation of vectors.

3.2. Affordance Predictor

The affordance predictor is composed of the action classifier and the heatmap decoder. The action predictor uses an LSTM to predict the action label. The heatmap decoder is implemented as a modified version of fully convolutional neural network [15]. It first encodes the target image $\mathbf{I}$ us-

ing fully convolutional layers. The computed convolutional features are then concatenated with the tiled demonstration embedding $\mathbf{v}$. Finally, the heatmap is computed by feeding the concatenated features into transpose convolutional layers [31]. A softmax layer is applied to normalize the sum of the heatmap to one.

The demonstration embedding $\mathbf{v}$ is learned and evaluated through the affordance predictor. For action classification, we apply the cross entropy loss on the predicted action label $\mathbf{c}$. For the heatmap prediction, we use the KL-divergence between the predicted heatmap and the ground truth heatmap as the loss, where the ground truth heatmap is rendered by applying a Gaussian blur to annotated points.

3.3. Implementation Details

Network Architecture: All images and all video frames are resized to 256×256 as inputs. The video is subsampled to 5 FPS. We use a VGG16 [24] as the feature extractor where the pretrained weights are restored from Faster R-CNN trained on MS-COCO dataset [19]. The pool5 layer is used as the extracted visual representation and fed into the ConvLSTMs and the heat map decoder. Each ConvLSTM uses a kernel size of 3 and stride of 1, producing a recurrent feature of 512 channels. For the heatmap decoder, we apply two consecutive convolutional layers, both with a kernel size of 1 and stride of 1, to the concatenated image and video feature. For the transposed convolution layers in the fully convolutional neural network, we use a kernel size of 64 and a stride of 32.

Training: Our model trains on a single Nvidia Titan X GPU for approximately 48 hours using an Adam optimizer [10]. The learning rate is initially set to 2×10^{-5}, with a decay ratio of 0.1 every $100,000$ iterations. We train our model on 16,976 examples and test it on 3,798 examples with the test-train split described in Sec. 4 where we ensure the products in both sets are distinct.

4. Dataset

The main goal of our paper is to develop a model that can learn affordance reasoning using human demonstrations from videos in the wild. In order to train our model and provide a testbed for other approaches, we need a dataset containing a large number of demonstrations of multiple human interactions with various objects.

For this purpose, we propose the Online Product Review dataset for Affordance (OPRA) collected for learning affordance reasoning. The dataset contains 11,505 demonstration clips and 2,512 object images scraped from 6 popular YouTube product review channels as well as corresponding affordance information. The products demonstrated in these videos include kitchenware objects, household appliances, consumer electronics, tools, and other objects. To generate these clips, 1,091 full-length videos were each split into

Figure 3. Example demonstration videos from our dataset. Each data point consists of a video, an image, 10 annotated points (shown as red dots) representing the interaction region, and an action label (shown in purple boxes). Here we show three representative frames for each video.

2 to 15 clips. Each segmented clip contains only a single interaction between the demonstrator and the object. For each product review video, 1 to 5 product images are collected from the Internet based on the product information on the YouTube video description written by the uploader. This produces totally 20,774 pairs of demonstration video clips and associated target images. We split the dataset into 16,976 pairs for training and 3,798 pairs for testing. This is done manually to avoid identical objects from different view points or too similar objects, such as different branded coffee machines, from appearing in both the training and testing sets. Samples are shown in Fig. 3.

The affordance information contains the interaction region and the action label, which are annotated through Amazon Mechanical Turk. Given each video clip containing a single human-object interaction, the annotator is asked to first watch the demonstration video clip and then annotate the action label along with the corresponding location of the interaction on the target image. Here, we follow the annotation routine from previous works on visual saliency [1, 2, 9]. In order to specify the interaction region, we ask the annotator to mark ten pixels on the target image to indicate the corresponding location where the interaction happened in the video. Then, the heat map is computed as a mixture of Gaussian centered at these chosen points. Notably, the heat map might cover more than one part of the object, such as the two handles of a pot.

The action classes consists of 7 different labels, and their types and associated distribution among the entire dataset (including both training and testing) is as follows: *hold:* 3992 (19.22%), *touch:* 9373 (45.12%), *rotate:* 1435 (6.91%), *push:* 2645 (12.73%), *pull:* 1138 (5.48%), *pick up:* 1342 (6.46%), *put down:* 849 (4.09%). In average each video comes with 2.55 target images of the same object from a different viewpoints.

An example demonstration video in our dataset is shown in Fig. 4. The video is segmented into several clips. Each box is a different video clip containing an action being applied to an object by the demonstrator. Note that not all video clips are of the same length. Each video clip is associated with a ground truth interaction heat map, generated from the ten annotated points, as well as a demonstrator-object interaction at that region, referred to as the action type. Notably, as our video clips consists of segments of a continuous video, extracting the interaction region and action can be thought of as extracting a high-level action manual for repeating the demonstration.

Compared to existing datasets for affordance reasoning [12, 16], our dataset is substantially different in several aspects. First, instead of having a single consistent viewpoint, each on-line product review video can be captured from multiple different camera angles, even within the same video. Second, the diversity of objects, environments and styles of videos recorded by different uploaders is quite large. Third, the numbers of videos and images collected in this dataset is significantly larger than all datasets from previous object affordance works. These characteristics distinguish our dataset from others and provide large-scale data for solving object affordance reasoning in the real world.

5. Experiments

In this section, we examine the performance of our model on learning an effective demonstration embedding. The demonstration encoder should encode the raw video to a latent representation termed as the demonstration embedding. The affordance predictor then take this demonstration embedding as inputs to accurately render the interaction heatmap on the target image as well as predict the action label being applied to that region. We first show qualitative results including success and failure cases in Sec. 5.2, and then report quantitative results in Sec. 5.1 where we compare our proposed model and the baselines. In Sec. 5.3 we analyze our models and conduct studies to demonstrate the capabilities of our model and interpret what it learns.

We compare several variants of our model that mainly differ in the architecture of the demonstration encoder as follows:

CNN+LSTM+Deconv: A standard linear LSTM network used as the demonstration encoder, the image feature (pool5 of the ConvNet) and the $1 \times 1 \times 4096$ demonstration embedding are concatenated by tiling the embedding to "fill" all the spatial locations of the image feature.

CNN+ConvLSTM+Deconv: A ConvLSTM [27] is used as the demonstration encoder to better capture temporally correlated spatial information, the demonstration embedding and the image feature are concatenated directly in their respective spatial dimensions.

CNN+ConvLSTM+STN+Deconv: As described in [8], the spatial transformation of a latent feature can be viewed as a spatial attention to that feature. Our ConvLSTM demonstration encoder is then taking as input the extracted frame feature after spatial transformation.

CNN+ConvLSTM+TSA+Deconv: In Sec. 3.1, we try to apply temporal soft attention on top of the demonstration encoder to aggregate the output features. The rest of the model is identical to CNN+ConvLSTM+STN+Deconvl.

CNN+ConvLSTM+Motion+TSA+Deconv: The temporal soft attention scores are computed using both the RGB image modality and motion modality. The rest of the model is identical to CNN+ConvLSTM+STN+Deconv. We refer this model as our core model.

For models without the temporal attention mechanism, we simply apply an average pooling to aggregate the output features.

5.1. Quantitative Results

Interaction Heatmap Prediction Table 1 summarizes the quantitative results. We evaluate performances of our core model and the baselines with two different metrics on our test set: KL-Divergence(KLD) and Negative Log Likelihood (NLL). KL-Divergence evaluates the mismatch of the two spatial distributions of the heatmap layouts. As shown in Table 1, our core model outperforms all the baselines significantly in both evaluation metrics.

Action Classification Table 2 shows the Top-1 prediction accuracies of our core model and several baselines. The action classification task here is fine-grained and the motion trajectory is one of the most important factors to

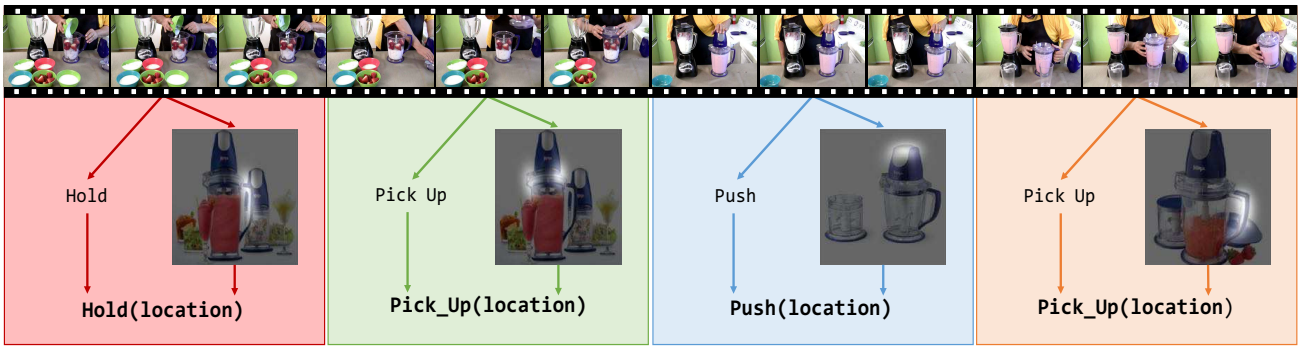


Figure 4. Example of a segmented demonstration video. The full video is about the task of making a smoothie, which is composed of sequential primitive actions: *hold*, *pick up*, *push*, *pick up*. The demonstration is segmented into 4 short clips, each shown in a different colored box. The sequential nature of some demonstration videos allows for learning sequential action planning.

successfully infer the action in the video. Our core model utilizes the motion modality and it outperforms every other baselines. Many failure cases are caused by similar action classes such as rotating and holding from the video.

Model	KLD	NLL
LSTM	3.45	113.65
ConvLSTM	3.31	112.17
STN+ConvLSTM	3.26	116.52
TSA+ConvLSTM	3.34	117.22
Motion+TSA+ConvLSTM	**2.34**	**102.50**

Table 1. Performances of the interaction heatmap prediction.

Demonstration Encoder	Top-1 (%)
LSTM	20.41
ConvLSTM	30.20
TSA+ConvLSTM	38.47
Motion+TSA+ConvLSTM	**40.79**

Table 2. Performances of the action label prediction.

5.2. Qualitative Results

Example qualitative results are shown in Fig. 7, including seven rows of successful cases and three rows of failure cases. The figure shows three sample frames of a decomposed video clip, the ground truth interaction heatmap rendered from the annotated points (by applying a Gaussian blur on a binary image), the predicted interaction heatmap from our model, and the ground truth and the predicted action label. The heatmaps are overlaid on the target image for visualization.

Our model is able to estimate the interaction region of interest and map this region to the target image for a variety of commonly seen interactable parts such as handles, buttons, knobs, and lids. Notably, these examples include videos with diverse l viewpoints, scales, and levels of occlusions.

One common failure case is caused by similar action classes. For example, our affordance predictor often confuses holding for rotating, since the motion dynamics in rotation are hardly captured and the overall hand gestures are similar to holding or grasping. In addition, it would be hard to predict correct heatmaps when the target image contains too many similar objects. In the first failure example, the model predicts an interaction heatmap that attends to a confounding disc-like object in the target image instead of the desired one. In the second example, the model properly predicts a lid being the interaction region, based off the video, but does not attend the proper lid in the image which includes multiple visually similar containers. For the last example, the model correctly predicts that the demonstrator interacts with a knob, but incorrectly projects the heatmap onto the wrong knob.

5.3. Analysis

Multi-Interaction Regions: An effective model for inferring the interaction region and associated action label should be responsive to different video demonstration inputs. The model should not only fixate on specific object parts as it may seem to be the saliency regardless of the demonstration video. In other words, given the same image but a set of different videos containing different interactions (potentially in both interaction regions and action labels), the predictions of the model should change accordingly. In Fig. 5, we show such an example that our model is trained to output different interaction heatmaps based on the interaction taken place within each demonstration video. This suggest that our demonstration encoder correctly encode different interactions between the hand motion and the proper interaction region of the object.

Viewpoint Robustness: In our dataset, the same video clip may be associated with multiple images of the same product. These target images are normally taken from different viewpoints or may contain other differences including extra objects or color changes. Even with this varying target image input, our model still must learn to infer the proper, corresponding interaction regions based off the demonstration videos. In Fig. 6, we show that our model is indeed robust to such variance in the target image. The

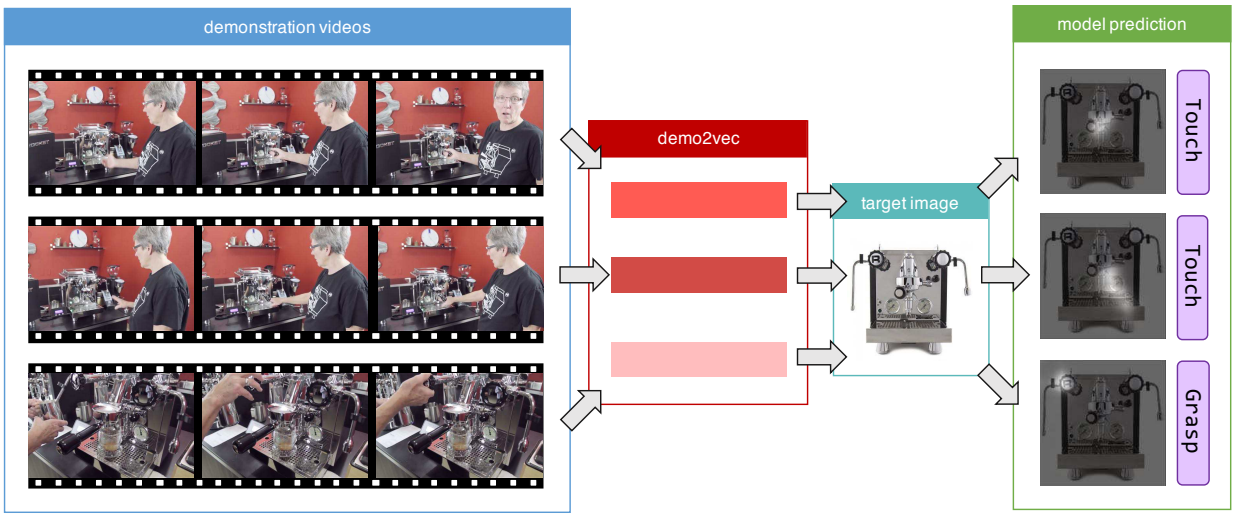


Figure 5. We show examples of predicting different affordances on the same target image given different video demonstrations. The three demonstrations of using a coffee machine show a person touches the handle, touches the thermometer, and grasps the knob respectively. For a target image of the same coffee machine, our model predicts distinct interaction heatmaps and corresponding action labels.

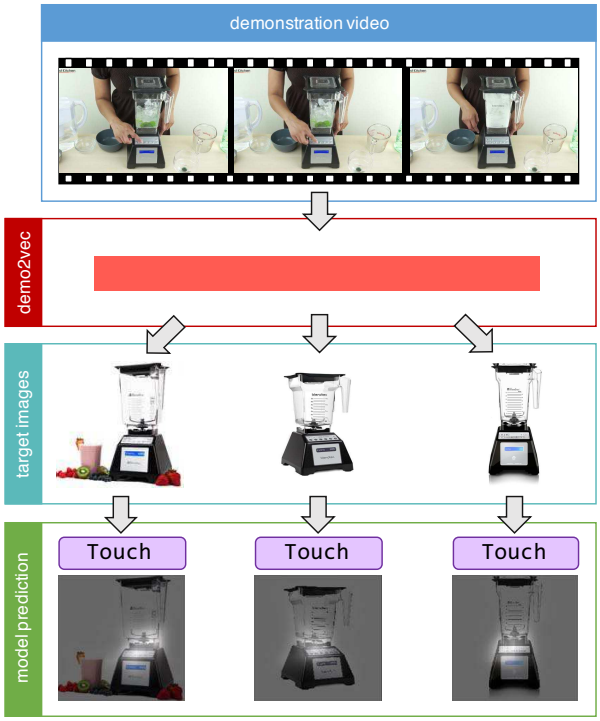


Figure 6. We show that our model is robust to variations in the target image. Given the same demonstration video of touching a button on a blender, the correct interaction region on three different images of the same blender highlight the same part of the blender.

affordance predictor takes the same demonstration embedding and learn to fixate on the same semantic interaction region for different viewpoints of the target object.

Visualization of the learned temporal soft attention: We further inspect what our model is fixating on throughout the entire video clip. We visualize the learned temporal soft attention scores as shown in Fig. 8. It is noticeable our model starts to attend more when demonstrator is interacting with the object at its proper interaction regions, which implies the motion dynamics is successfully captured and interpreted correctly by the affordance predictor.

6. Conclusion

In this paper, we tackle the problem of reasoning affordances based on demonstration videos in the wild. We introduce the Demo2Vec model which extracts feature embeddings from demonstration videos, and predicts affordances of the same object by transferring the extracted knowledge onto a target image. The Demo2Vec model is composed of a demonstration encode and a afforance predictor. To train and evaluate the model, we collect YouTube product review videos and introduce the Online Product Review dataset for Affordance (OPRA). The OPRA dataset is one of the first datasets providing a testbed for affordance reasoning based on demonstrations from YouTube product review videos. Our model achieves better performances on the OPRA dataset comparing with various neural network baselines, in terms of interaction heatmap prediction and action label prediction.

Acknowledgement

We acknowledge the support of the SKT gift funding, the startup funding of University of Southern California, and Toyota (1191689-1-UDAWF). We also thank Yuke Zhu and Fei-Fei Li for constructive discussions.

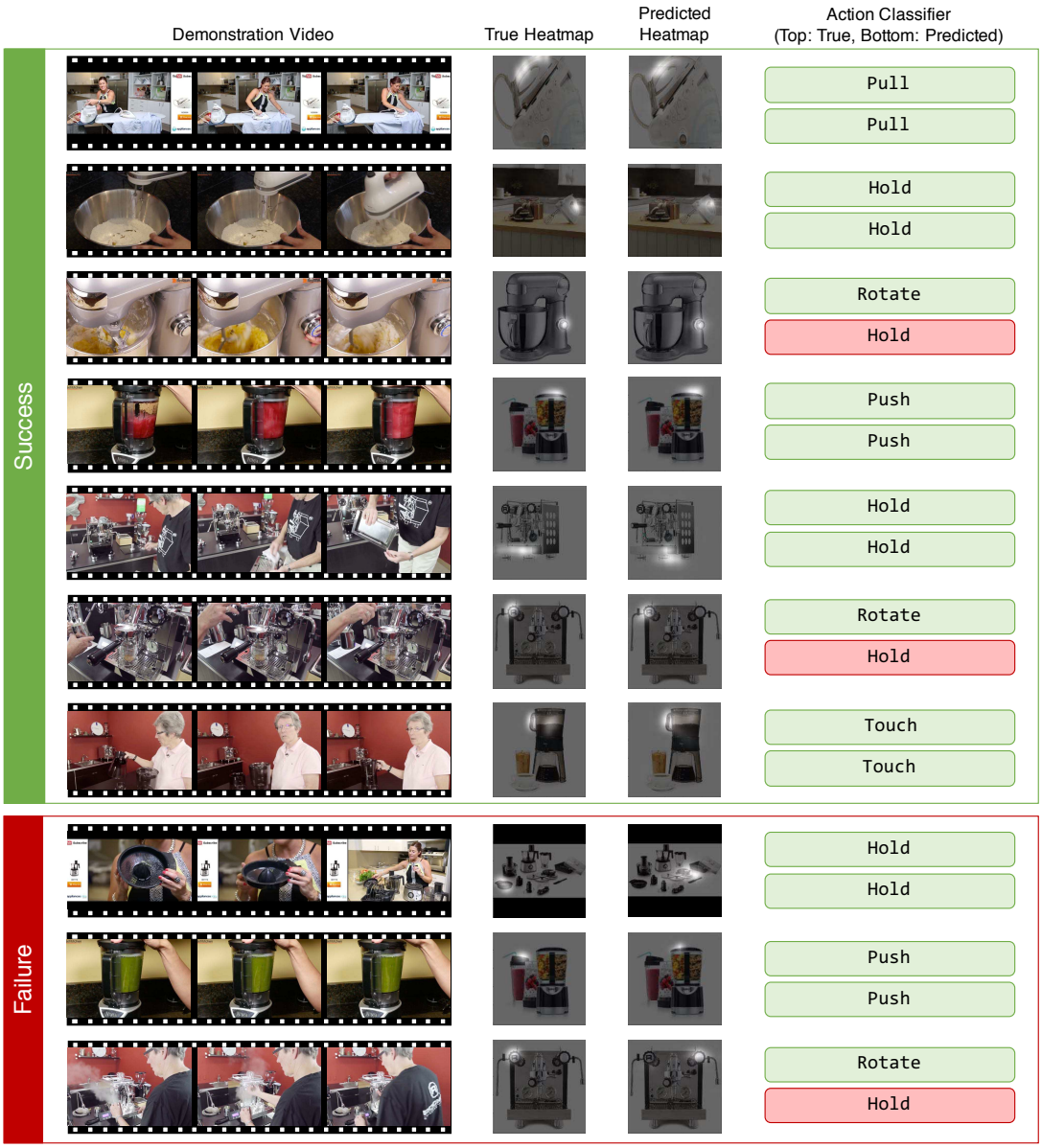


Figure 7. **Qualitative results on the OPRA dataset.** Our model is able to predict the interaction heatmap and action label for a variety of video and object scenarios. Here we show common failure cases which are caused by self-occlusions in the target image (first row), confounding parts on the object (last two rows).

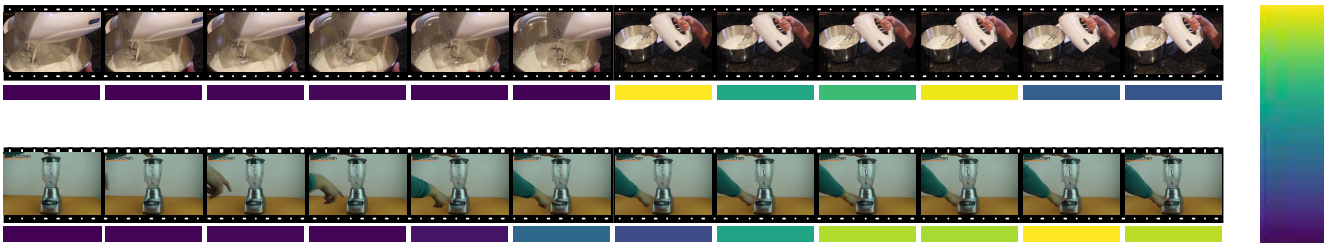


Figure 8. **Visualization of the learned temporal soft attention.** The predicted temporal soft attention coefficients are shown below each video frame, plotted according to the color bar on the right. Yellow and blue colors indicate high and low attention coefficients respectively.

References

[1] Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba. Mit saliency benchmark. 5

[2] Z. Bylinskii, T. Judd, A. Oliva, A. Torralba, and F. Durand. What do different evaluation metrics tell us about saliency models? *arXiv preprint arXiv:1604.03605*, 2016. 5

[3] Y. Duan, M. Andrychowicz, B. Stadie, J. Ho, J. Schneider, I. Sutskever, P. Abbeel, and W. Zaremba. One-shot imitation learning. *arXiv preprint arXiv:1703.07326*, 2017. 2

[4] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with lstm. 1999. 3

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2

[6] J. Ho and S. Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, 2016. 2

[7] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3

[8] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 5

[9] T. Judd, F. Durand, and A. Torralba. A benchmark of computational models of saliency to predict human fixations. In *MIT Technical Report*, 2012. 5

[10] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[11] H. Kjellström, J. Romero, and D. Kragić. Visual object-action recognition: Inferring object affordances from human demonstration. *Computer Vision and Image Understanding*, 115(1):81–90, 2011. 2

[12] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. *The International Journal of Robotics Research*, 32(8):951–970, 2013. 2, 5

[13] H. S. Koppula and A. Saxena. Physically grounded spatio-temporal object affordances. In *European Conference on Computer Vision*, pages 831–847. Springer, 2014. 1, 2

[14] H. S. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):14–29, 2016. 1

[15] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 3

[16] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos. Affordance detection of tool parts from geometric features. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1374–1381. IEEE, 2015. 5

[17] A. Nair, D. Chen, P. Agrawal, P. Isola, P. Abbeel, J. Malik, and S. Levine. Combining self-supervised learning and imitation for vision-based rope manipulation. *arXiv preprint arXiv:1703.02018*, 2017. 1

[18] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis. Detecting object affordances with convolutional neural networks. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 2765–2770. IEEE, 2016. 2

[19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 4

[20] S. Ross, G. J. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*, pages 627–635, 2011. 2

[21] A. Roy and S. Todorovic. A multi-scale cnn for affordance segmentation in rgb images. In *European Conference on Computer Vision*, pages 186–201. Springer, 2016. 2

[22] J. Schulman, J. Ho, C. Lee, and P. Abbeel. Learning from demonstrations through the use of non-rigid registration. In *Robotics Research*, pages 339–354. Springer, 2016. 1

[23] T. Shu, M. S. Ryoo, and S.-C. Zhu. Learning social affordance for human-robot interaction. *arXiv preprint arXiv:1604.03692*, 2016. 1

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 4

[25] A. Srikantha and J. Gall. Weakly supervised learning of affordances. *arXiv preprint arXiv:1605.02964*, 2016. 2

[26] B. C. Stadie, P. Abbeel, and I. Sutskever. Third-person imitation learning. *arXiv preprint arXiv:1703.01703*, 2017. 2

[27] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015. 3, 5

[28] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 3

[29] X. Yan, M. Khansari, Y. Bai, J. Hsu, A. Pathak, A. Gupta, J. Davidson, and H. Lee. Learning grasping interaction with geometry-aware 3d representations. *arXiv preprint arXiv:1708.07303*, 2017. 1

[30] B. Yao, J. Ma, and L. Fei-Fei. Discovering object functionality. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2512–2519, 2013. 2

[31] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2528–2535, 2010. 4

[32] Y. Zhu, C. Jiang, Y. Zhao, D. Terzopoulos, and S.-C. Zhu. Inferring forces and learning human utilities from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3823–3833, 2016. 2

[33] Y. Zhu, Y. Zhao, and S. Chun Zhu. Understanding tools: Task-oriented object modeling, learning and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2855–2864, 2015. 1, 2