

# End-to-End Learning of Semantic Grasping

**Eric Jang**  
Google Brain  
ejang@google.com

**Sudheendra Vijayanarasimhan**  
Google  
svnaras@google.com

**Peter Pastor**  
X  
peterpastor@x.team

**Julian Ibarz**  
Google Brain  
julianibarz@google.com

**Sergey Levine**  
Google Brain, UC Berkeley  
slevine@google.com

**Abstract:** We consider the task of semantic robotic grasping, in which a robot picks up an object of a user-specified class using only monocular images. Inspired by the two-stream hypothesis of visual reasoning, we present a semantic grasping framework that learns object detection, classification, and grasp planning in an end-to-end fashion. A “ventral stream” recognizes object class while a “dorsal stream” simultaneously interprets the geometric relationships necessary to execute successful grasps. We leverage the autonomous data collection capabilities of robots to obtain a large self-supervised dataset for training the dorsal stream, and use semi-supervised label propagation to train the ventral stream with only a modest amount of human supervision. We experimentally show that our approach improves upon grasping systems whose components are not learned end-to-end, including a baseline method that uses bounding box detection. Furthermore, we show that jointly training our model with auxiliary data consisting of non-semantic grasping data, as well as semantically labeled images without grasp actions, has the potential to substantially improve semantic grasping performance<sup>1</sup>.

**Keywords:** semantic grasping, deep learning

## 1 Introduction

Robotic manipulation presents a number of challenges as well as exciting opportunities for computer vision. Robots that interact with the world must recognize and localize objects with a very high degree of precision, which places a high demand on the performance of computer vision systems. The substantial domain shift between objects and scenes seen by a robot and those in labeled object recognition datasets makes it nontrivial to generalize conventional computer vision systems to robotic tasks. On the other hand, robots have the ability to autonomously collect their own data as they perform manipulation tasks in the real world. By developing methods for self-supervision, we can utilize robots to autonomously collect large self-labeled datasets that can help us exceed the performance of standard vision systems that are trained on fixed-size, human-labeled datasets.

In this paper, we explore this in the context of robotic grasping with semantic labels. The purpose of our method is to allow a robot to successfully grasp objects in its environment based on high-level user commands, which specify the type of object to pick up. This task aims to combine both spatial and semantic reasoning: the robot must determine which of the objects in the scene belong to the requested class, and how those objects might be grasped successfully. We aim to answer two questions: first, how can we design a model that effectively combines both spatial and semantic reasoning, and second, how can we exploit the capacity of the robots to collect self-supervised data while also making use of modest numbers of human-provided labels to learn about semantics. Answering these questions successfully could lead not only to more effective and useful robot systems, but to advances in spatial and semantic visual reasoning and automated data gathering.

---

<sup>1</sup>A video overview of this paper can be found at <https://youtu.be/WR5WUKXUQ8U>.

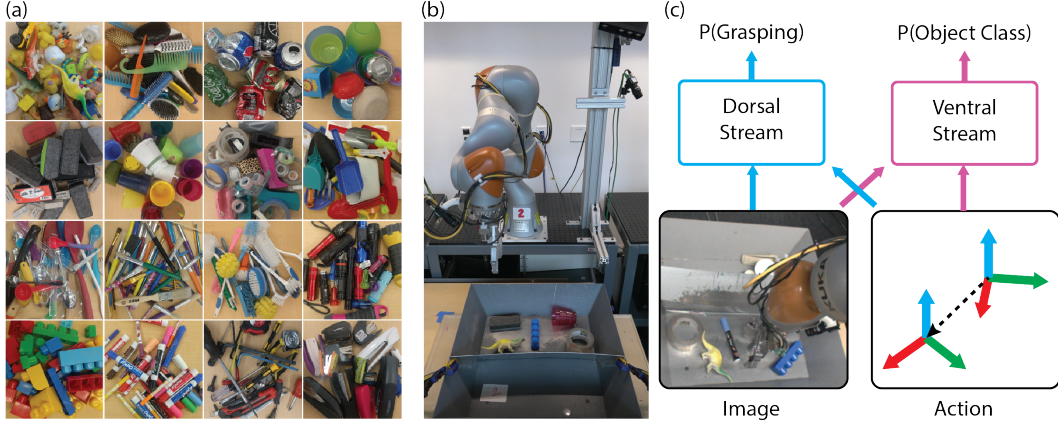


Figure 1: (a) We consider the task of learning to pick up objects from 16 object classes, conditioned on a user-commanded class. (b) Our robotic setup is an arm with a 2-finger gripper that serves using monocular camera images. (c) Inspired by the two-stream hypothesis of vision [3], we propose a model that is divided into a dorsal and ventral stream. The dorsal stream selects actions that result in good grasps, and the ventral stream selects the class of the grasped object. Best viewed in color.

Previous work [1, 2] has demonstrated that a non-semantic grasp outcome predictor can be trained in an entirely self-supervised manner via autonomous data collection. However, those works did not consider object class in grasp planning. In order to solve the semantic grasping task, we require semantically labeled data and a model that is capable of reasoning about object identity.

In designing our method, we take inspiration from the “two-stream hypothesis” of human vision, where visual reasoning that is both spatial and semantic can be divided into two streams: a ventral stream that reasons about object identity, and a dorsal stream that reasons about spatial relationships without regard for semantics [3]. We train a deep neural network for performing the semantic grasping task that is decomposed into these two streams, as illustrated in Figure 1. The dorsal stream predicts whether a particular grasping action will result in a successful grasp. The ventral stream is trained to predict what type of object will be picked up. We tackle the problem of assigning semantic labels to large amounts of robot-collected data via a label propagation scheme: if the robots successfully grasp an object, they can then bring it up to their cameras for closer inspection. A modest number of such human-labeled images can be used to train a classifier for these images, and then propagate labels to the images used for learning grasping in a semi-supervised fashion.

The main contribution of our work is a two-stream neural network model for learning semantic grasping from off-policy data. We use a combination of self-supervised robotic data collection and label propagation to construct large object classification dataset for training our models. Finally, we show that both dorsal and ventral streams can benefit independently from domain transfer, by integrating auxiliary spatial information from non-semantic grasping datasets and auxiliary semantic information from image classification datasets.

Our experimental results demonstrate that: (1) Our proposed two-stream model substantially outperforms a more standard detection pipeline for grasping, as well as single-stream approaches. (2) Integrating auxiliary non-semantic data substantially improves grasping performance. (3) Auxiliary semantically labeled data without grasping actions has the potential to improve generalization, but this improvement is highly dependent on the degree of domain shift between the auxiliary data and the grasping images.

## 2 Related Work

We consider semantic grasping as our model task, where the robot must pick up objects of a specific, user-selected semantic category. A complete discussion of prior grasping work is outside of the scope of this paper, and a more complete overview can be found in recent survey work [4]. In contrast to most grasping methods in robotics [5, 6, 7], we use only monocular RGB images, and do not make use of any hand-designed geometric cues, camera calibration, or even feedback from

the gripper to execute a grasp sequence. Prior data-driven methods in grasping have used human supervision [8, 9] and automated geometric criteria [10] for supervision. Both types of data-driven grasp selection have recently incorporated deep learning [11, 9, 12, 13].

While most grasping methods focus on the spatial aspects of the problem, a number of works have sought to incorporate semantic information into grasping, including affordances and task-specific knowledge [14, 15, 16], instance-based object models [17], known shape information about objects [10], and prior knowledge that objects of the same type should be grasped in similar ways [18]. However, while these prior methods incorporate semantic knowledge to aid spatial reasoning [19], our approach instead aims to combine both spatial and semantic reasoning to pick up objects of a particular category in cluttered scenes. Incorporating additional feedback from semantic knowledge into the spatial problem of grasping, as in prior work, is also a very valuable research direction, but largely orthogonal to the main focus of this work. In this sense, our method is complementary.

Self-supervision and large-scale data collection have recently been proposed for image-based robotic grasping [2, 1]. Our work also follows this paradigm for training the dorsal stream of our semantic grasping network, with large numbers of grasp attempts collected with real robotic manipulators and labeled as successful or failed grasps automatically. However, we also incorporate a ventral stream into our system that is trained on semantic categories, with knowledge transfer from object classification datasets and automatic label propagation from a small number of human-provided labels. In this sense, our aims are orthogonal to this prior work; our goal is not to simply build a better grasping system, but to explore how semantics and label propagation interact with real-world robotic manipulation tasks.

### 3 Deep Semantic Grasping

In our setup, a robotic arm with a two-finger gripper learns to grasp particular objects from a bin based on a user-provided object class. We train our grasping system end-to-end, using off-policy data collected by a non-semantic grasping policy and a modest number of semantic labels provided by humans. We propose several deep neural net architectures for learning semantic grasping, discuss our methodology for collecting large datasets for training our models, and describe how information from external datasets can be leveraged to improve grasping performance.

#### 3.1 Two-Stream Value Function

Our policy critic, visualized in Figure 1c, takes as input images  $I_t, I_0$ , which correspond to the current image seen by the robot’s camera, and the initial image seen during the episode. The initial image is recorded before the grasping episode begins and is included to allow the system to handle self-occlusions caused by the robot arm. The network also receives as input a candidate task-space motion command  $a_t$ , which is parameterized by a Cartesian displacement of the gripper along with a vertical rotation of the gripper. The network outputs  $p(g, c|I_t, a_t)$ , the joint probability that executing the command  $a_t$  will result in  $g$ , the event that an object is grasped and  $c$ , the event that the grasped object is the correct semantic label. Our servoing function samples grasps randomly and uses the neural critic to choose the grasp with the highest probability of picking up the desired object. See Appendix C for additional details on the model architecture and the servoing policy.

A straightforward way to implement a semantic neural critic is to extend the final layer of the model proposed in prior work on self-supervised grasping [1] to output a softmax distribution over object classes, plus an additional “failure” class for when no objects are grasped (Figure 2b). This model, which we call “Single Stream” in our experiments, lacks a dorsal stream that explicitly learns what it means to “pick up any object”. Instead, it merges spatial and semantic inference into a single output, predicting grasp success for each object class independently.

Additional supervision of spatial reasoning capabilities can be obtained by distinguishing between failures of classification and failures of detection. We decompose the prediction problem into a dorsal stream  $p(g|I_t, a_t)$  that is trained on non-semantic grasping data (whose labels are success or failure), while the ventral stream  $p(c|I_t, a_t, g)$  is trained to classify successful grasps whose labels are one of 16 object classes. The model learns to infer where actions will send the gripper, and implicitly attends to that location in order to extract its spatial relevance to grasping (Figure 2a, left) and semantic identity (Figure 2a, right).

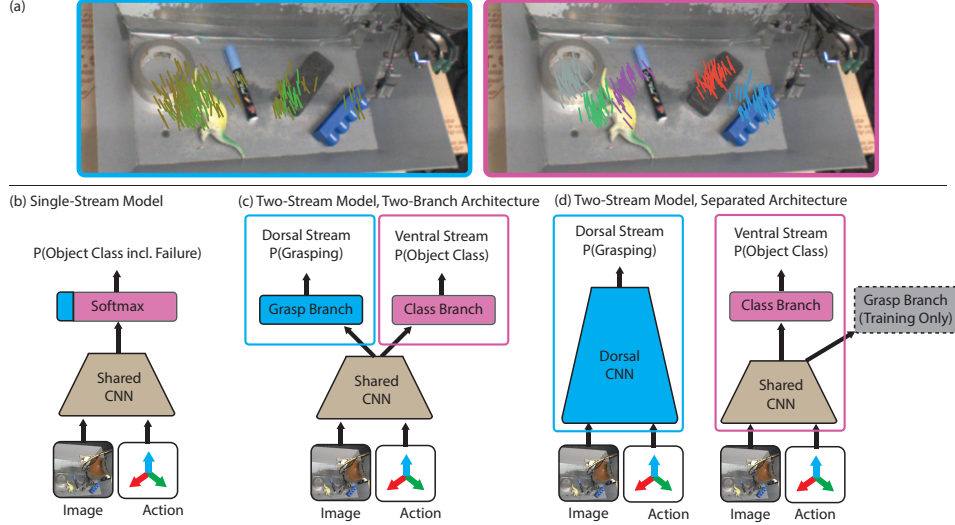


Figure 2: (a) Left: visualization of grasp actions projected into screen space, with brighter green actions corresponding to actions with higher probability of grasp success. Right: projected grasp actions, with each color corresponding to the most probable actions for picking up an object class. Hand-eye coordination binds spatial awareness with visual semantics, enabling the policy to learn object classification and detection end-to-end. (b) Single Stream model that predicts grasp success for each class independently, without explicitly modeling object semantics. (c) Two-Stream model that shares model parameters between a grasp branch and class branch, which comprise the dorsal (cyan box) and ventral streams (magenta box). (d) Our best two-stream model uses the class branch from (c) for the ventral stream, but uses a separate dorsal stream trained only on grasp prediction. Best viewed in color.

We consider Two-Stream models as not a single neural net architecture, but rather a family of architectures that functionally decouple the prediction of grasp success (dorsal stream) and object class (ventral stream). An example of a Two-Stream model is a “Two-Branch” (2B) architecture that shares inputs and action processing before branching out to grasp-specific and class-specific parameters (Figure 2c). Another example is a “Separated” architecture which takes advantage of the fact that Two-Stream models can mix-and-match dorsal and ventral streams at inference time. We train both branches of the 2B architecture as before, but only use the class branch as the ventral stream. The dorsal stream of the Separated architecture is an entirely separate convolutional network trained only to predict grasp success (Figure 8b). Because this separate dorsal stream outperforms non-semantic grasping capabilities of our 2B models, we use this separated architecture for most of our Two-Stream models.<sup>2</sup> Sharing the dorsal stream between Two-Stream architectures also allows for a controlled comparison between different ventral stream architectures.

Instead of using a deep CNN that implicitly attends to an object using the action vector, we wondered whether an explicit attention mechanism would improve the model’s ability to extract a representation of visual features that was invariant to the object’s location in the bin. In addition to our 16-layer CNN, we also try a shallower 9-layer network that bottlenecks predictions through soft keypoint locations [20]. Both architectures perform similarly, though the soft keypoint architecture has substantially fewer parameters and therefore trains more efficiently. Further details on both architectures are provided in Appendix B.1.

### 3.2 Parallel Data Collection & Model Training

We chose 16 object categories found in typical kitchen and office settings (Table 1). We obtained 30-40 object instances per category and divided these 500 objects into a set of training and test objects

<sup>2</sup>On terminology: we use “Two-Stream model” to refer to the functional decomposition of spatial and semantic reasoning into the multiplication of  $p(g|I_t, a_t) \cdot p(c|I_t, a_t, g)$ , while we use “Two-Branch” and “Separated” to reflect architectural choices on how to implement a Two-Stream model.





Figure 3: Images captured during the execution of the “present” pose of the robot, which facilitate easier classification and label propagation to cluttered grasping images.

Object Category	Eraser	Scotch tape	Comb	Shovel	Legos	Torch	Pen	Plastic cup	Toy	Tools	Stapler	Toilet brush	Paintbrush	Spoon	Bowl	Soda can
Human-rated training examples	395	464	587	649	591	288	437	517	1191	189	860	128	613	331	201	91
Human-rated test examples	14	110	320	4	123	25	60	286	533	62	28	79	114	-	312	-

Table 1: The object categories in our dataset along with the number of training and testing instances.

(which are used to evaluate generalization to unseen class instances). To collect the data used to train our semantic grasping models, we first learn a non-semantic grasping policy that is capable of picking up objects in a self-supervised manner, similar to the approach from [1]. We utilized 8 KUKA iiwa arms to collect data in parallel and introduce diverse lighting and variations in camera calibration into the dataset. Labels for grasp success/failure can be generated autonomously via a simple threshold on an image delta, as described in [1].

Next, we use two approaches to annotate the collected dataset with object labels, in order to supervise learning of the ventral branch. In the first approach, each robot’s bin is filled with objects sharing the same semantic label. This makes it trivial to assign labels to large amounts of data, for which we gather 10,000 such examples. Object collections are frequently shuffled between the different robots so that our models do not overfit to a specific robot’s idiosyncratic visual cues. In our second approach, we program the robot to lift grasped objects close to the camera at the end of a grasping sequence for further inspection, at which point a close-up image of the object being held in the gripper is recorded (e.g. Figure 3). Presenting objects in this manner isolates the grasped objects for the purpose of easier classification, somewhat analogously to how children pick up and examine objects in order to isolate it from surrounding visual context. We label 20,000 present images using Amazon Mechanical Turk and use this information to train a CNN classifier. By classifying the remaining 300,000 present images and propagating those labels to their corresponding grasping images, we can generate a large autonomously collected dataset for semantic grasping.

### 3.3 Transfer from External Datasets

Although the robots can gather large amounts of data autonomously and then label them via automatic label propagation, the diversity of object instances in such datasets is limited. Fortunately, our two-stream decomposition of spatial and semantic perception affords us a straightforward mechanism for incorporating images from either (1) auxiliary grasping data with no semantic labels, including objects that are difficult to classify or do not belong to the semantic classes, or (2) auxiliary semantic data of different object instances without grasping actions.

For auxiliary grasping data, we jointly train our two-stream models on all the grasping data we collected, including objects that do not belong to our semantic categories, supervising only the grasp branch of the model. In all, we had 11 million grasping images that we could use to train the dorsal stream (corresponding to about 2 million grasps), of which 330,000 had semantic labels and could be included for training the ventral stream. This auxiliary data serves to not only improve our model’s ability to generalize to the test objects, but also benefits classification by improving the early visual representations learned by the shared CNN. To incorporate auxiliary semantic data, we include images from object classification datasets directly into the grasping model with a random action.

In our experiments, we present results with two auxiliary semantic datasets, which we call “S1” and “S2.” S1 contains images from ImageNet and JFT [21, 22] and with present images of training objects. S2 contains present images of objects in the test set (but are not paired with grasping actions). The rationale behind including S2 in our evaluations is to determine how much improvement can be had from a classification dataset that perfectly represents the distribution of objects seen at test time, from a different viewpoint and with a different camera. This represents a “best case” scenario for



Figure 4: In our evaluation, the robots are tasked to pick up specific categories from bins containing 5 randomly selected objects from different classes. Top row: Training objects. Bottom row: Test objects. Evaluation objects were selected at random to cover all object categories uniformly.

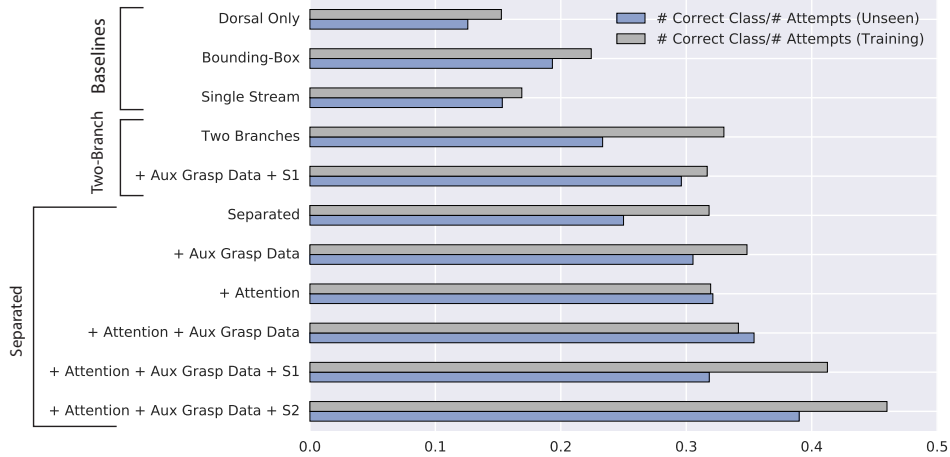


Figure 5: Real-world robotic evaluation of semantic grasping models on training and unseen test objects, showing the percentage of grasp attempts that result in the correct class being grasped. See Appendices D, E for a complete table of evaluation metrics and class confusion matrices per model.<sup>2</sup>

semantic but non-spatial transfer. The performance of the method with both of these two datasets is presented in Section 4.3.

## 4 Experimental Results

We evaluate semantic grasping performance on real-world robots by assigning 5 randomly selected objects from different categories to each robot (Figure 4) and commanding the robot to pick up the objects in a random order. For a sequence of 5 grasp attempts, the robot removes grasped objects one at a time from the bin and drops them outside the workspace. This is repeated 10 times per robot with 6 robots in total, for a set of 30 training objects and 30 unseen testing objects. As shown in Appendix D, we evaluate policies based on the fraction of grasp attempts that were successful at picking up some object (Objects/Attempts) and the fraction of all grasp attempts that picked up the correct object (Class/Attempts). We demonstrate through baseline comparisons and ablation studies the effect of various architectural decisions in our semantic grasping models.

**Summarily:** (1) End-to-end learning of semantic grasping outperforms a traditional detection-classification-grasping method. (2) Two-stream decompositions of grasp and class prediction outperform a single-stream model. (3) A separated architecture is better at grasping than two-branch architectures. (4) A 9-layer CNN with attention performs comparably to a 16-layer CNN without attention. (5) Auxiliary grasping data improves classification of our two-stream architectures. (6) Auxiliary semantic data that reflects the distribution of objects we are evaluating on improves class grasping accuracy.

#### 4.1 Baseline Comparisons

The primary challenge in moving from non-semantic grasping to semantic grasping is that the policy must now extract a specific object from clutter, instead of just reaching for the object that is easiest to grasp. Furthermore, the actions that result in good grasps on a target object may not be the best actions that suggest the object’s semantic identity (e.g., a handle on a shovel is easy to grasp but could also be the handle for a toilet brush). Our “Dorsal Only” baseline lacks semantic reasoning capabilities; while it can pick up an object 73.1% and 73.5% of the time on training and test objects, the fraction of successful grasps being of the right class is random (20.8% on training objects and 17.2% on test objects, Table 2, “Dorsal Only”) <sup>3</sup>.

We compare our end-to-end semantic grasping policies to a semantic grasping approach in which object detection, classification, and grasp planning are separated into separate pipelines, which we refer to as the “Bounding Box Model”. Since our grasping dataset lacks annotated segmentation masks for training region proposal models, we cannot train an object detector or classifier on grasping images; instead, we train an Inception-Resnet-V2 classifier [23] on the S1 dataset (Section 3.3) to obtain a classifier with 95% accuracy on held-out “present images” of training objects and 35% accuracy on “present images” test objects collected during the data-collection phase. For object detection, we adopt a simplified version of RCNN [24, 25] that performs a grid search within the initial image  $I_0$  to find the fixed-size crop which most likely corresponds to the target class, as predicted by the trained classifier. The crop bounds in screen space are then projected to the workspace plane using precise camera extrinsic measurements, forming a quad that bounds the detected object. The non-semantic dorsal stream described in Section 3.1 is then instructed to grasp an object constrained to these quad bounds (Figure 6).

The “Bounding Box” model has 41.6% and 32.4% class accuracy among successfully grasped objects, which is a significant improvement over the random behavior of a non-semantic dorsal grasping model. However, this baseline still had lower classification performance than all of our end-to-end learned models ( $> 51.7\%$ ). Failure modes of this model include (1) failing to center the object in the detected crop, causing a poor grasp location on the object and (2) including more than one object in the crop, which may result in the arm picking up another object close to the target. These errors are primarily caused by a large domain shift from canonical image classification datasets to grasping images, despite the classifier being trained on present images of the same objects. Although learning a region-proposal model would result in better canonical crops for grasping, acquiring these labels is a labor-intensive procedure. In contrast, our models exploit closed loop robotic interaction with visual feedback to circumvent the need to acquire manually-annotated region-proposal annotations, and uses learned hand-eye coordination rather than camera calibration.

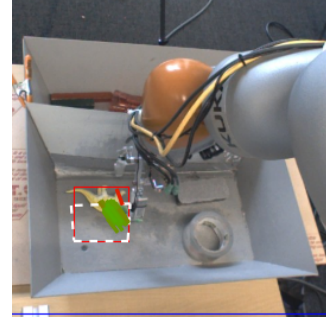


Figure 6: Bounding box baseline.

#### 4.2 Two-Stream Models

We trained and evaluated 8 two-stream models, with 2 using a Two-Branch architecture and 6 using Separated architecture. Separated architectures performed better than Two-Branch models that jointly predicted class probability and grasp success (Appendix D, rows 4,5 vs. rows 6,10).

Figure 1a visualizes the value function learned by our two stream models. In the left image, the image is superimposed with the actions that correspond to the highest probabilities of picking up any object. On the right, the image is superimposed with actions which are color-coded by the most likely object class at that location. The colors roughly correspond to the locations of objects, showing our model’s ability to jointly learn visual grounding of semantic concepts.

A shallower model with soft keypoint attention (Section B.1) yields about the same semantic grasping performance as a deeper convolutional model while using half as many parameters (Appendix D, rows 6,7 vs. rows 8,9). The effect of auxiliary grasping data has similar effects on both attention and non-attention models, suggesting that attention allows us to obtain comparable accuracy with fewer

<sup>3</sup>We compute this via  $.208 = .153/.735$  and  $.172 = .126/.731$ , see Appendix D

parameters. We use “Separated + Attention” as the base model for further experiments with transfer learning from auxiliary semantic datasets.

### 4.3 Effect of External Dataset Transfer

We first examine the effect of auxiliary grasping data that is used to provide additional supervision to the grasp branch of our Two Branch architecture. In our “Separated” architectures, we train both grasp and class branches from the Two Branch model but only use the latter as the ventral stream of the two-stream model. Even when the grasp branch is replaced by a separately-trained model and not used for grasp prediction at evaluation, the auxiliary data still helps to learn low-level sensorimotor representations that are useful for classifying semantic objects. The Two-Stream Separated + Aux Grasp data yields a 3.0% improvement on Class/Attempts over the Two-Stream Separated model, and the corresponding attention-based model yields a 2.2% improvement on training objects and 3.3% improvement on unseen objects.

Next, we consider the effect of transfer from the S1 and S2 datasets described in Section 3.3. While our models do not seem to benefit from external domain transfer of ImageNet/JFT, our experiments suggest that training on the “optimal” auxiliary dataset does improve classification success: S1 improves classification performance on training objects only, while training jointly on present images of test objects (S2) improves grasping accuracy on both training/test objects. These mixed results on domain transfer suggest that either JFT/ImageNet do not provide much information about our random selection of evaluated training and test objects, or that a more powerful domain adaptation architecture is required to generalize recognition of our object categories from these classification datasets. From a practical standpoint, this transfer is still useful as it enables robots to learn about object semantics without having to grasp them. However, it remains an open question as to how we can perform more effective knowledge transfer from datasets like ImageNet.

Overall, the results in Table 2 point to two-stream architectures being better suited to semantic grasping than single-stream models or a pipelined approach using an object detector. The overall success rate of the methods still leaves room for improvement, which indicates the difficulty of semantic grasping in clutter. This task combines the physical challenge of picking up diverse and varied objects with the finegrained perception challenge of predicting which object will be picked up. We hope that future work will continue to improve on these results, and that end-to-end semantic grasping will emerge as a useful benchmark task for both robotic control and situated object detection.

## 5 Discussion

In this work we propose a learning-based system for visual robotic grasping that combines spatial and semantic reasoning into a single neural network policy. Our network is decomposed into two streams, with the “ventral” stream being selective for the commanded object class while the “dorsal” stream selects for good grasps, independent of object identity.

Semantic grasping is not only a compelling research area for studying the intersection of spatial and semantic reasoning, but is also an as-yet unsolved task of considerable practical value. Semantic grasping capabilities could enable a household robot to clean a room, perform industrial sorting tasks on food products, or pack an ever-changing inventory of products into boxes. In each of these situations, the robot is likely to encounter objects that it has never seen or grasped before, and must appropriately generalize both spatial and semantic understanding to handle object diversity.

More broadly, robotics present a unique opportunity for cheaply computing labels for large amounts of complex data via active perception. We address the difficult problem of assigning semantic labels to large amounts of cluttered data via label propagation: during data collection, the robotic arm isolates the grasped object so that an uncluttered view of the object may be captured. This can be done by either lifting up the object to the camera for closer inspection, or using an auxiliary wrist-mounted camera to view the contents of the gripper directly. Such “present images” are centered, uncluttered, and easy to classify, and the temporal coherence of a grasping sequence allows us to retroactively assign the resulting semantic label to an earlier cluttered grasping scene. Our setup enables us to collect additional present poses per grasped object (e.g. multiple orientations), though we leave this for future work.



## Acknowledgments

We would like to thank Vincent Vanhoucke for organization and discussion, John-Michael Burke for assistance with data collection and robot hardware, Zbigniew Wojna for helpful advice on attention mechanisms, Mrinal Kalakrishnan, Matthew Kelcey, and Ian Wilkes for code reviews, and Colin Raffel for helpful comments and feedback on the paper.

## References

- [1] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen. Learning hand-eye coordination for robotic grasping with large-scale data collection. In *International Symposium on Experimental Robotics (ISER 2016)*, 2016.
- [2] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *IEEE International Conference on Robotics and Automation*, 2016.
- [3] L. Ungerleider and J. Haxby. What and where in the human brain. *Current Opinion in Neurobiology*, 4, 1994.
- [4] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-Driven Grasp Synthesis A Survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.
- [5] C. Goldfeder, M. Ciocarlie, J. Peretzman, H. Dang, and P. K. Allen. Data-Driven Grasping with Partial Sensor Data. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1278–1283, 2009.
- [6] J. Weisz and P. K. Allen. Pose Error Robust Grasping from Contact Wrench Space Metrics. In *IEEE International Conference on Robotics and Automation*, 2012.
- [7] A. Rodriguez, M. T. Mason, and S. Ferry. From Caging to Grasping. *The International Journal of Robotics Research*, 31(7):886–900, June 2012.
- [8] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, J. Bohg, T. Asfour, and S. Schaal. Learning of Grasp Selection based on Shape-Templates. *Autonomous Robots*, 36(1-2):51–65, 2014.
- [9] I. Lenz, H. Lee, and A. Saxena. Deep Learning for Detecting Robotic Grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.
- [10] C. Goldfeder, M. Ciocarlie, H. Dang, and P. Allen. The Columbia Grasp Database. In *IEEE International Conference on Robotics and Automation*, 2009.
- [11] D. Kappler, B. Bohg, and S. Schaal. Leveraging Big Data for Grasp Planning. In *IEEE International Conference on Robotics and Automation*, 2015.
- [12] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *IEEE International Conference on Robotics and Automation*, 2015.
- [13] M. Gualtieri, A. ten Pas, K. Saenko, and R. Platt. High precision grasp pose detection in dense clutter. *arXiv preprint arXiv:1603.01564*, 2016.
- [14] H. Dang and P. K. Allen. Semantic grasping: Planning task-specific stable robotic grasps. *Autonomous Robots*, 37(3):301–316, 2014.
- [15] D. Katz, A. Venkatraman, M. Kazemi, J. A. Bagnell, and A. Stentz. Perceiving, learning, and exploiting object affordances for autonomous pile manipulation. *Autonomous Robots*, 37(4): 369–382, 2014.
- [16] M. Gupta, J. Müller, and G. S. Sukhatme. Using Manipulation Primitives for Object Sorting in Cluttered Environments. *IEEE Transactions on Automation Science and Engineering*, 12(2): 608–614, 2015.
- [17] J. Oberlin and S. Tellex. Autonomously Acquiring Instance-based Object Models from Experience. *Int. S. Robotics Research (ISRR)*, 2015.

- [18] E. Nikandrova and V. Kyrki. Category-based task specific grasping. *Robotics and Autonomous Systems*, 70:25 – 35, 2015.
- [19] B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, and K. Goldberg. Cloud-based robot grasping with the google object recognition engine. In *IEEE International Conference on Robotics and Automation*, pages 4263–4270, May 2013.
- [20] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel. Learning visual feature spaces for robotic manipulation with deep spatial autoencoders. *CoRR*, abs/1509.06113, 2015. URL <http://arxiv.org/abs/1509.06113>.
- [21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- [22] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- [23] C. Szegedy, S. Ioffe, and V. Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016. URL <http://arxiv.org/abs/1602.07261>.
- [24] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.
- [25] A. t. Pas, M. Gualtieri, K. Saenko, and R. Platt. Grasp pose detection in point clouds. *arXiv preprint arXiv:1706.09911*, 2017.
- [26] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning*, 2015.
- [27] R. Rubinstein and D. Kroese. *The Cross-Entropy Method*. Springer-Verlag, 2004.

## A Data Collection Details



Figure 7: A snapshot of all the objects used for data collection. The set contains 20-30 instances over 15-20 small objects that we might encounter in typical kitchen and office settings.

The Amazon Mechanical Turk service was used to label 20,000 present images for label propagation. The raters’ choices included the 16 categories in Table 1 as well as *No Objects*, *Multiple*, and *None of these*. We used a simple majority consensus to obtain the final label of each grasp sequence. Each task took under 20 seconds and 2 cents to complete on average.

## B Network Architectures

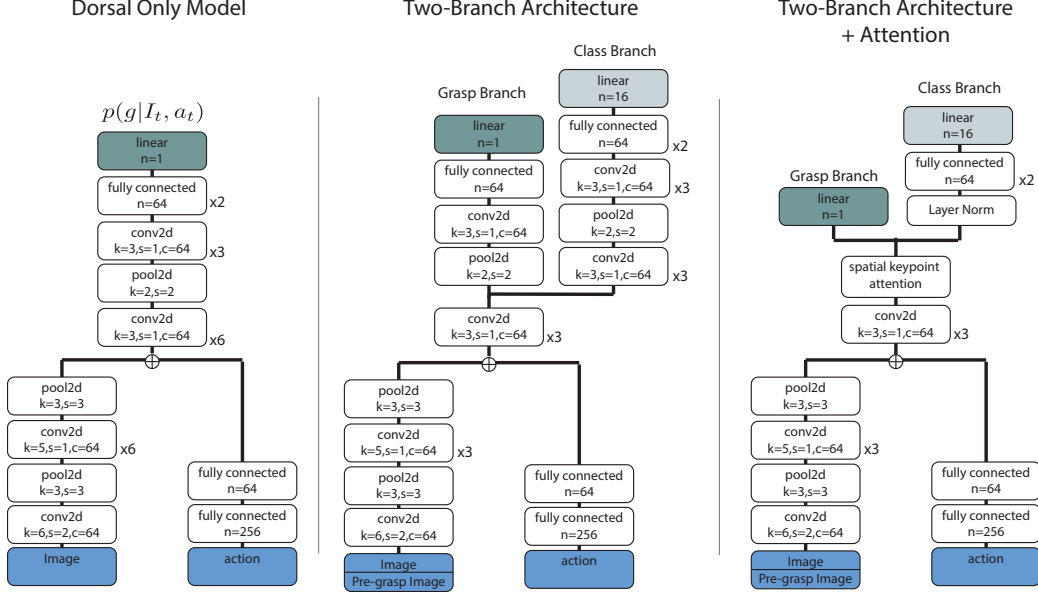


Figure 8: Two-stream model for semantic grasping. ReLU activations and batch normalization [26] are used at all but the final layer of each model.

### B.1 Soft Keypoint Attention

In soft keypoint attention [20], the output  $a_{cij}$  of a convolutional layer is normalized across the spatial extent via a softmax  $s_{cij} = e^{a_{cij}/\tau} / \sum_{i'j'} e^{a_{ci'j'}/\tau}$ . The spatial softmax for each channel is then used as weights for spatially averaging a mesh grid  $f_c = (\sum_i i \cdot s_{cij}, \sum_j j \cdot s_{cij})$ , which yields

expected screen coordinates for salient visual features in that channel. These features are normalized to mean 0 and standard deviation 1 prior to classification, since class identity should be invariant to where the spatial features are located and their scale.

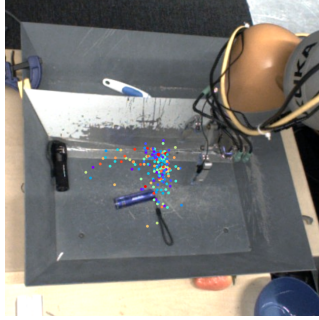


Figure 9: Visualization of soft keypoint attention. Each point corresponds to the mean screen coordinates of a spatial softmax over channels in a convolution layer.

Since soft keypoint attention returns a low-dimensional representation of attended features in the form of coordinates, we can directly model the dorsal and ventral branches of our model using a few fully-connected layers 8. Although our models experimentally suggest that soft keypoint attention does not contribute to substantially better performance than simply using a deeper CNN, it is a convenient way to reduce the number of parameters in the model and accelerate training.

## C Evaluation Details

Each grasp attempt consists of up to 10 time steps and at each time step, the robot records the current image  $I_t$  and samples an action  $a_t$  along which to move the gripper. At each step, action is optimized for several iterations via CEM [27], which optimizes the neural critic described in Section 3.1 with respect to the proposed action. At the final grasp step  $T$ , the robot closes the gripper and evaluates the success of the grasp by recording feedback from the gripper. Data collection initially is bootstrapped on top of a policy that issues random motor commands at each time step. Off-policy data was collected  $T = 10$ , with 2 iterations of CEM per step. At inference time, we used  $T = 4$  with 3 iterations of CEM per step to speed up the grasping system. The total numbers of grasps vary due to hardware failures that required restarting the evaluation protocol, resulting in some robots/objects having more attempts than others. However, grasps for a model were only scored after all evaluations were completed.

Occasionally, the robot picks up more than one object. If one of the objects grasped was the correct commanded one, we still count the grasp as a success (though the removal of two objects may result in another grasp failing).

Our semantic grasping models are trained using off-policy data from a non-semantic grasping policy, which tends to servo to the nearest object during data collection. Consequently, our dorsal and ventral streams are not trained on actions that are distal from the gripper, and thus cannot accurately reason about semantics and spatial reasoning for distant actions. Instead of the random approach move in [1], we modify the approach trajectory to hover the end-effector over 5 equally-spaced regions of the bin, so that the action search is able to consider moves from a variety of starting points (one of which will be close to the best candidate object). For each “hover pose”, we optimize for the action that maximizes the semantic probability  $p(c|I_t, a_t, g)$ , and select the maximum-likelihood action among all such poses as the actual approach pose, in order to bring the gripper closer to the object.

Small probabilities from the ventral stream can destabilize predictions from the dorsal stream, even if the value function correctly identifies the right object to grasp. At inference time, we smooth the ventral probability via  $\min(p(c|I_t, a_t, g) + 0.5, 1)$ , prior to multiplying with the dorsal probability, which allows the robot to pick up an object even if it is unsure about the semantic identity of an object.



## D Table of Results

	Model Name	Training Objects		Unseen Objects	
		Objects/Attempts	Class/Attempts	Objects/Attempts	Class/Attempts
Baselines	Dorsal Only	73.5	15.3	73.1	12.6
	Bounding Box	53.9	22.4	59.7	19.3
	Single Stream	26.7	16.9	24.3	15.3
Two-Stream, Two-Branch	Two Branches	50.3	33.0	43.0	23.3
	Two Branches + aux grasp data + S1	52.0	31.7	49.7	29.6
Two-Stream, Separated	Separated	51.7	31.8	61.0	25.0
	Separated + Aux Grasp Data	62.5	34.8	56.4	30.5
	Separated + At- tention	48.6	31.9	62.1	32.1
	Separated + At- tention + Aux Grasp Data	51.1	34.2	63.0	35.4
	Separated + At- tention + Aux Grasp Data + S1	62.0	41.2	66.9	31.8
	Separated + At- tention + Aux Grasp Data + S2	64.0	<b>46.0</b>	61.7	<b>39.0</b>

Table 2: Objects/Attempts = number of objects grasped divided by number of grasp attempts made (%). Class/Attempts = number of *correct* objects grasped divided by number of grasp attempts (%).

# E Confusion Matrices

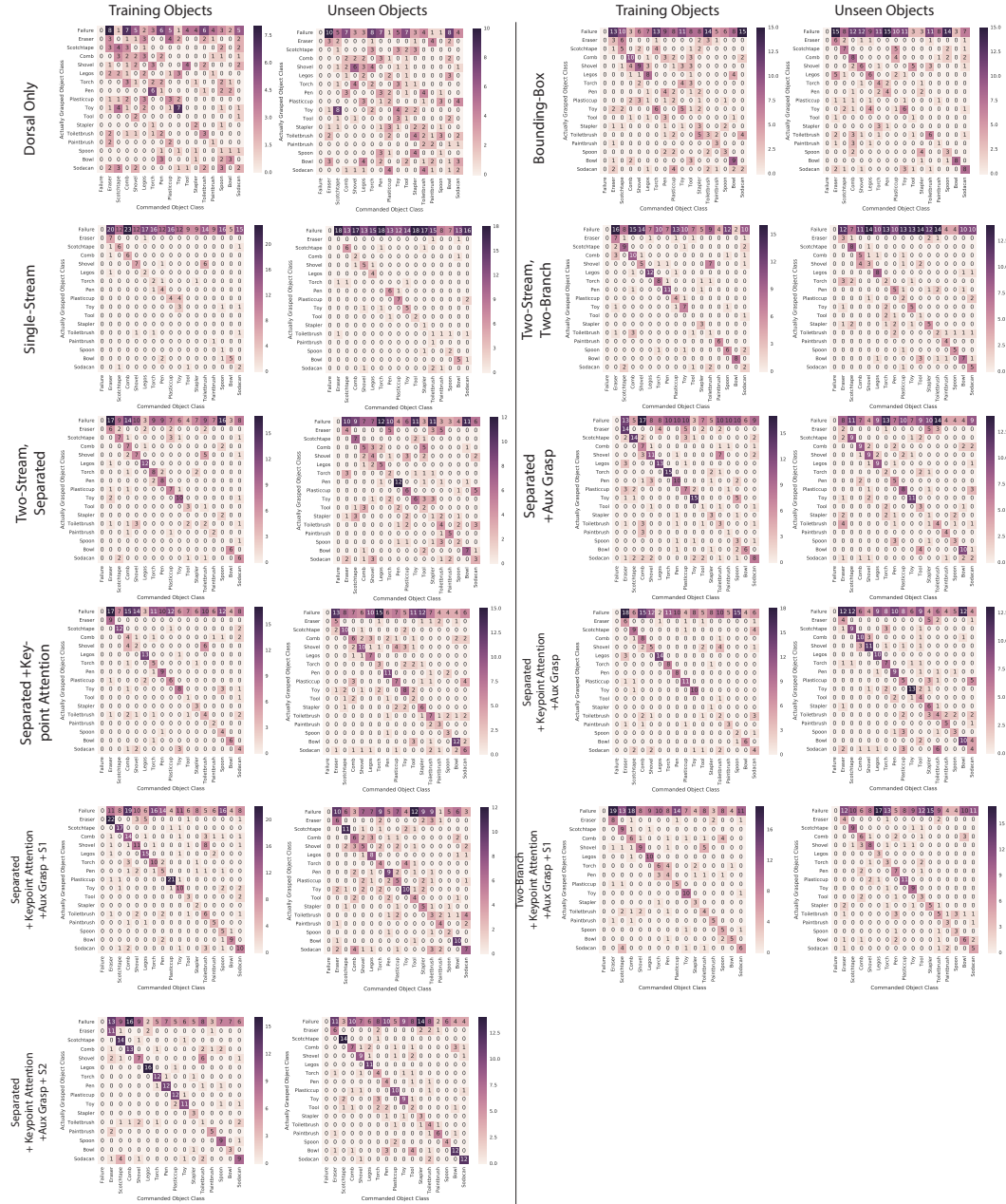


Figure 10: Confusion matrices on grasping training and unseen objects for all evaluated models. Row  $i$ , column  $j$  in each matrix corresponds to the number of times the robot picked up class  $i$  when it was told to pick up class  $j$ .