Check for updates

Article

The International Journal of Robotics Research
2018, Vol. 37(4–5) 405–420
© The Author(s) 2018
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/0278364918770733
journals.sagepub.com/home/ijr

The limits and potentials of deep learning for robotics

Niko Sünderhauf[1], Oliver Brock[2], Walter Scheirer[3], Raia Hadsell[4], Dieter Fox[5], Jürgen Leitner[1], Ben Upcroft[6], Pieter Abbeel[7], Wolfram Burgard[8], Michael Milford[1] and Peter Corke[1]

Abstract

The application of deep learning in robotics leads to very specific problems and research questions that are typically not addressed by the computer vision and machine learning communities. In this paper we discuss a number of robotics-specific learning, reasoning, and embodiment challenges for deep learning. We explain the need for better evaluation metrics, highlight the importance and unique challenges for deep robotic learning in simulation, and explore the spectrum between purely data-driven and model-driven approaches. We hope this paper provides a motivating overview of important research directions to overcome the current limitations, and helps to fulfill the promising potentials of deep learning in robotics.

Keywords

Robotics, deep learning, machine learning, robotic vision

1. Introduction

A robot is an inherently *active* agent that interacts with the real world, and often operates in uncontrolled or detrimental conditions. Robots have to perceive, decide, plan, and execute actions, all based on incomplete and uncertain knowledge. Mistakes can lead to potentially catastrophic results that will not only endanger the success of the robot's mission, but can even put human lives at risk, e.g. if the robot is a driverless car.

The application of deep learning in robotics therefore motivates research questions that differ from those typically addressed in computer vision: How much trust can we put in the predictions of a deep learning system when misclassifications can have catastrophic consequences? How can we estimate the uncertainty in a deep network's predictions and how can we fuse these predictions with prior knowledge and other sensors in a probabilistic framework? How well does deep learning perform in realistic unconstrained open-set scenarios where objects of unknown class and appearance are regularly encountered?

If we want to use data-driven learning approaches to generate motor commands for robots to move and act in the world, we are faced with additional challenging questions: How can we generate enough high-quality training data? Do we rely on data solely collected on robots in real-world scenarios or do we require data augmentation through simulation? How can we ensure the learned policies transfer well to different situations, from simulation to reality, or between different robots?

This leads to further fundamental questions: How can the structure, the constraints, and the physical laws that govern robotic tasks in the real world be leveraged and exploited by a deep learning system? Is there a fundamental difference between model-driven and data-driven problem solving, or are these rather two ends of a spectrum?

This paper explores some of the challenges, limits, and potentials for deep learning in robotics. The invited speakers and organizers of the workshop on *The Limits and*

[1]Australian Centre for Robotic Vision, Queensland University of Technology (QUT), Brisbane, Australia
[2]Robotics and Biology Laboratory, Technische Universität Berlin, Germany
[3]Department of Computer Science and Engineering, University of Notre Dame, IN, USA
[4]DeepMind, London, UK
[5]Paul G. Allen School of Computer Science & Engineering, University of Washington, WA, USA
[6]Oxbotica Ltd., Oxford, UK
[7]UC Berkeley, Department of Electrical Engineering and Computer Sciences, CA, USA
[8]Department of Computer Science, University of Freiburg, Germany

Corresponding author:
Niko Sünderhauf, Queensland University of Technology (QUT), 2 George Street, Brisbane 4000 QLD, Australia.
Email: niko.suenderhauf@roboticvision.org

Potentials of Deep Learning for Robotics at the 2016 edition of the *Robotics: Science and Systems* (RSS) conference (Sünderhauf et al., 2016) provide their thoughts and opinions, and point out open research problems and questions that are yet to be answered. We hope this paper will offer the interested reader with an overview of where we believe important research needs to be done, and where deep learning can have an even bigger impact in robotics over the coming years.

2. Challenges for deep learning in robotic vision

A robot is an inherently *active* agent that acts *in*, and interacts *with* the physical real world. It perceives the world with its different sensors, builds a coherent model of the world, and updates this model over time, but ultimately a robot has to make decisions, plan actions, and execute these actions to fulfill a useful task.

This is where *robotic vision* differs from computer vision. For robotic vision, perception is only one part of a more complex, embodied, active, and goal-driven system. *Robotic* vision therefore has to take into account that its *immediate* outputs (object detection, segmentation, depth estimates, 3D reconstruction, a description of the scene, and so on), will ultimately result in *actions* in the real world. In a simplified view, whereas computer vision takes images and translates them into information, robotic vision translates images into actions.

This fundamental difference between robotic vision and computer vision motivates a number of research challenges along three conceptually orthogonal axes: *learning*, *embodiment*, and *reasoning*. We position individual challenges along these axes according to their increasing complexity, and their dependencies. Figure 1 and Tables 1–3 summarize the challenges.

2.1. Learning challenges

Along this axis we position challenges that are specific for (deep) machine learning in a robotic vision context. These challenges comprise problems arising from deployment in open-set conditions, two flavors of incremental learning, and active learning.

2.1.1. Uncertainty estimation. To fully integrate deep learning into robotics, it is important that deep learning systems can reliably estimate the uncertainty in their predictions. This would allow robots to treat a deep neural network in the same way as any other sensor, and use the established Bayesian techniques (Kaess et al., 2012; Kümmerle et al., 2011; Thrun et al., 2005) to fuse the network's predictions with prior knowledge or other sensor measurements, or to accumulate information over time. Deep learning systems, e.g. for classification or detection, typically return scores from their softmax layers that are proportional to the system's confidence, but are *not* calibrated probabilities, and therefore not useable in a Bayesian sensor fusion framework.

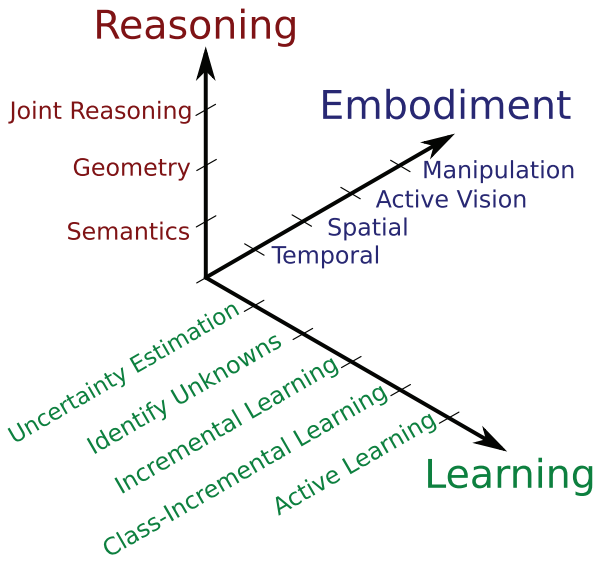

Fig. 1. Current challenges for deep learning in robotic vision. We can categorize these challenges into three conceptually orthogonal axes: learning, embodiment, and reasoning.

Current approaches towards uncertainty estimation for deep learning are calibration techniques (Guo et al., 2017; Hendrycks and Gimpel, 2017), or Bayesian deep learning (MacKay, 1992; Neal, 1995) with approximations such as dropout sampling (Gal and Ghahramani, 2016; Kendall and Gal, 2017) or ensemble methods (Lakshminarayanan et al., 2017).

2.1.2. Identify unknowns. A common assumption in deep learning is that trained models will be deployed under *closed-set* conditions (Bendale and Boult, 2015; Torralba and Efros, 2011), i.e. the classes encountered during deployment are known and exactly the same as during training. However, robots often have to operate in ever-changing, uncontrolled real-world environments, and will inevitably encounter instances of classes, scenarios, textures, or environmental conditions that were not covered by the training data.

In these so-called *open-set* conditions (Bendale and Boult, 2015; Scheirer et al., 2013a), it is crucial to identify the unknowns: the perception system must not assign high-confidence scores to unknown objects or falsely recognize them as one of the known classes. If, for example, an object detection system is fooled by data outside of its training data distribution (Goodfellow et al., 2014; Nguyen et al., 2015a), the consequences for a robot acting on false, but high-confidence detections can be catastrophic. One way to handle the open-set problem and identify unknowns is to utilize the epistemic uncertainty (Gal and Ghahramani, 2016; Kendall and Gal, 2017) of the model predictions to reject predictions with low confidence (Miller et al., 2017).

Table 1. Learning challenges for robotic vision

Level	Name	Description
5	Active learning	The system is able to select the most informative samples for incremental learning on its own in a data-efficient way, e.g. by utilizing its estimated uncertainty in a prediction. It can ask the user to provide labels.
4	Class-incremental learning	The system can learn *new* classes, preferably using low-shot or one-shot learning techniques, without catastrophic forgetting. The system requires the user to provide these new training samples along with correct class labels.
3	Incremental learning	The system can learn off new instances of known classes to address domain adaptation or label shift. It requires the user to select these new training samples.
2	Identify unknowns	In an open-set scenario, the robot can reliably identify instances of unknown classes and is not fooled by out-of-distribution data.
1	Uncertainty estimation	The system can correctly estimate its uncertainty and returns *calibrated* confidence scores that can be used as probabilities in a Bayesian data fusion framework. Current work on Bayesian deep learning falls into this category.
0	Closed-set assumptions	The system can detect and classify objects of classes known during training. It provides uncalibrated confidence scores that are proportional to the system's belief of the label probabilities. State-of-the-art methods, such as YOLO9000, SSD, and Mask R-CNN, are at this level.

Table 2. Embodiment challenges for robotic vision

Level	Name	Description
4	Active manipulation	As an extension of active vision, the system can manipulate the scene to aid perception. For example, it can move an occluding object to gain information about objects hidden underneath.
3	Active vision	The system has learned to actively control the camera movements in the world, for example it can move the camera to a better viewpoint to improve its perception confidence or better deal with occlusions.
2	Spatial embodiment	The system can exploit aspects of spatial coherency and incorporate views of objects taken from different viewpoints to improve its perception, while handling occlusions.
1	Temporal embodiment	The system learned that it is temporally embedded and consecutive images are strongly correlated. The system can accumulate evidence over time to improve its predictions. Appearance changes over time can be coped with.
0	None	The system has no understanding of any form of embodiment and treats every image as an independent from previously seen images.

Table 3. Reasoning challenges for robotic vision

Level	Name	Description
3	Joint reasoning	The system jointly reasons about semantics and geometry in a tightly coupled way, allowing semantics and geometry to co-inform each other.
2	Object and scene geometry	The system learned to reason about the geometry and shape of individual objects, and about the general scene geometry, such as absolute and relative object pose, support surfaces, and object continuity under occlusions and in clutter.
1	Object and scene semantics	The system can exploit prior semantic knowledge to improve its performance. It can utilize priors about which objects are more likely to occur together in a scene, or how objects and overall scene type are correlated.
0	None	The system does not perform any sophisticated reasoning, e.g. it treats every detected object as independent from other objects or the overall scene. Estimates of semantics and geometry are treated as independent.

2.1.3. Incremental learning. For many robotics applications the characteristics and appearance of objects can be quite different in the deployment scenario compared with the training data. To address this domain adaptation problem (Csurka, 2017; Ganin et al., 2015; Patel et al., 2015), a robotic vision system should be able to learn from new training samples of known classes during deployment and adopt its internal representations accordingly.

2.1.4. Class-incremental learning. When operating in open-set conditions, the deployment scenario might contain new classes of interest that were not available during training. A robot therefore needs the capability to extend its knowledge and efficiently learn new classes without forgetting the previously learned representations (Goodfellow et al., 2013). This class-incremental learning would preferably be data-effcient by using one-shot (Bertinetto et al.,

2016; Lake et al., 2015; Rezende et al., 2016; Santoro et al., 2016; Vinyals et al., 2016) or low-shot (Finn et al., 2017; Hariharan and Girshick, 2016; Wang and Hebert, 2016) learning techniques. Semi-supervised approaches (Kingma et al., 2014; Papandreou et al., 2015; Rasmus et al., 2015) that can leverage unlabeled data are of particular interest.

Current techniques for class-incremental learning (Mensink et al., 2012; Rebuffi et al., 2017) still rely on supervision in the sense that the user has to specifically tell the system which samples are new data and therefore should be incorporated. The next challenge in our list, *active learning*, aims to overcome this and automatically selects new training samples from the available data.

2.1.5. Active learning. A robot should be able to select the most informative samples for incremental learning techniques on its own. Since it would have to ask the human user for the true label for these selected samples, data-efficiency is key to minimize this kind of interaction with the user. Active learning (Cohn et al., 1996) can also comprise retrieving annotations from other sources such as the web.

Some current approaches (Dayoub et al., 2017; Gal et al., 2017) leverage the uncertainty estimation techniques based on approximate Bayesian inference (see Section 2.1.1) to choose the most informative samples.

2.2. Embodiment challenges

Embodiment is a cornerstone of what constitutes robotic vision, and what sets it apart from computer vision. Along this axis we describe four embodiment challenges: understanding and utilizing temporal and spatial embodiment helps to improve perception, but also enables robotic vision to perform active vision, and even targeted manipulation of the environment to further improve perception.

2.2.1. Temporal embodiment. In contrast to typical recent computer vision systems that treat every image as independent, a robotic vision system perceives a *stream* of consecutive and therefore strongly correlated images. Whereas current work on action recognition, learning from demonstration, and similar directions in computer vision work on video data (e.g. by using recurrent neural networks or by simply stacking consecutive frames in the input layers), the potential of *temporal* embodiment to improve the quality of the perception process for object detection or semantic segmentation, is currently rarely utilized: a robotic vision system that uses its temporal embodiment can, for example, accumulate evidence over time (preferably using Bayesian techniques, if uncertainty estimates are available as discussed in Section 2.1.1) or exploit small viewpoint variations that occur over time in dynamic scenes.

The new CORe50 dataset (Lomonaco and Maltoni, 2017) is one of the few available datasets that encourages researchers to exploit temporal embodiment for object recognition, but the robotic vision research community should invest more effort to fully exploit the potentials of temporal embodiment.

A challenging aspect of temporal embodiment is that the appearance of scenes changes over time. An environment can comprise dynamic objects such as cars or pedestrians moving through the field of view of a camera. An environment can also change its appearance caused by different lighting conditions (day/night), structural changes in objects (summer/winter), or differences in the presence and pose of objects (e.g. an office during and after work hours). A robotic vision system has to cope with all of those effects.

2.2.2. Spatial embodiment. In robotic vision, the camera that observes the world is part of a larger robotic system that acts and moves in the world: the camera is *spatially* embodied. As the robot moves in its environment, the camera will observe the scene from different viewpoints, which poses both challenges and opportunities to a robotic vision system: observing an object from different viewpoints can help to disambiguate its semantic properties, improve depth perception, or segregate an object from other objects or the background in cluttered scenes. On the other hand, occlusions and the resulting sudden appearance changes complicate visual perception and require capabilities such as object unity and object permanence (Piaget, 2013) that are known to develop in the human visual system (Goldstein and Brockmole, 2016).

2.2.3. Active vision. One of the biggest advantages robotic vision can draw from its embodiment is the potential to *control* the camera, move it, and change its viewpoint to improve its perception or gather additional information about the scene. This is in stark contrast to most computer vision scenarios, where the camera is a passive sensor that observes the environment from where it was placed, without any means of controlling its pose.

Some work is undertaken in the area of next-best viewpoint prediction to improve object detection (Atanasov et al., 2014; Doumanoglou et al., 2016; Malmir et al., 2017; Wu et al., 2015b) or path planning for exploration on a mobile robot (Bircher et al., 2016), but a more holistic approach to active scene understanding is still missing from current research. Such an active robotic vision system could control camera movements through the world to improve the system's perception confidence, resolve ambiguities, mitigate the effect of occlusions, or reflections.

2.2.4. Manipulation for perception. As an extension of active vision, a robotic system could purposefully manipulate the scene to aid its perception. For example, a robot could move occluding objects to gain information about object hidden underneath. Planning such actions will require an understanding of the geometry of the scene, the capability to reason about how certain manipulation actions

will change the scene, and if those changes will positively affect the perception processes.

2.3. Reasoning challenges

In his influential 1867 book on physiological optics, Von Helmholtz (1867) formulated the idea that humans use unconscious *reasoning*, inference or conclusion, when processing visual information. Since then, psychologists have devised various experiments to investigate these unconscious mechanisms (Goldstein and Brockmole, 2016), modernized Helmholtz's original ideas (Rock, 1983), and reformulated them in the framework of Bayesian inference (Kersten et al., 2004).

Inspired by their biological counterparts, we formulate the following three reasoning challenges, addressing separate and joint reasoning about the semantics and geometry of a scene and the objects therein.

2.3.1. Reasoning about object and scene semantics. The world around us contains many semantic regularities that humans use to aid their perception (Goldstein and Brockmole, 2016): objects tend to appear more often in a certain context than in other contexts (e.g. it is more likely to find a fork in a kitchen or on a dining table, but less likely to find it in a bathroom), some objects tend to appear in groups, some objects rarely appear together in a scene, and so on. Semantic regularities also comprise the absolute pose of object in a scene, or the relative pose of an object with respect to other objects.

Although the importance of semantic regularities and contextual information for human perception processes is well known in psychology (Goldstein and Brockmole, 2016; Oliva and Torralba, 2007), current object detection systems (He et al., 2017; Liu et al., 2016b; Redmon and Farhadi, 2016) do not exploit this rich source of information. If the many semantic regularities present in the real world can be learned or otherwise made available to the vision system in the form of prior knowledge, we can expect an improved and more robust perception performance: Context can help to disambiguate or correct predictions and detections.

The work by Lin et al. (2013) is an example of a scene understanding approach that explicitly models and exploits several semantic and geometric relations between objects and the overall scene using conditional random fields. A combination of place categorization and improved object detection utilizing learned scene–object priors has been demonstrated in Sünderhauf et al. (2016). In more recent work by Zhang et al. (2017) a method was devised to perform holistic scene understanding using a deep neural network that learns to utilize context information from training data.

2.3.2. Reasoning about object and scene geometry. Many applications in robotics require knowledge about the *geometry* of individual objects, or the scene as a whole. Estimating the depth of the scene from a single image has become a widely researched topic (Garg et al., 2016; Godard et al., 2017; Liu et al., 2016a). Similarly, there is a lot of ongoing work on estimating the 3D structure of objects from a single or multiple views without having depth information available (Choy et al., 2016; Häne et al., 2017; Yan et al., 2016; Zhu et al., 2017). These methods are typically evaluated on images with only one or a few prominent and clearly separated objects. However for robotic applications, cluttered scenes are very common.

The previously discussed problems of uncertainty estimation and coping with unknown objects apply here as well: a robotic vision system that uses the inferred geometry, for example to grasp objects, needs the ability to express uncertainty in the inferred object shape when planning grasp points. Similarly, it should be able to exploit its embodiment to move the camera to a better viewpoint to efficiently collect new information that enables a more accurate estimate of the object geometry.

As an extension of reasoning over individual objects, inference over the geometry of the whole scene is important for robotic vision, and closely related to the problems of object-based mapping or object-based simultaneous localization and mapping (SLAM) (Cadena et al., 2016; Pillai and Leonard, 2015; Salas-Moreno et al., 2013; Sünderhauf et al., 2017). Exploiting semantic and prior knowledge can help a robotic vision system to better reason about the scene structure, for example the absolute and relative poses of objects, support surfaces, and object continuity despite occlusions.

2.3.3. Joint reasoning about semantics and geometry. The ability to extract information about objects, environmental structures, their various complex relations, and the scene geometry in complex environments under realistic, open-set conditions is increasingly important for robotics. Our final reasoning challenge for a robotic vision system therefore is the ability to reason *jointly* about the semantics and the geometry of a scene and the objects therein. Since semantics and geometry can co-inform each other, a tightly coupled inference approach can be advantageous over loosely coupled approaches where reasoning over semantics and geometry is performed separately.

3. Are we getting evaluation right in deep learning for robotics?

Why does real-world deep learning performance fail to match the published performance on benchmark datasets? This is a vexing question currently facing roboticists, and the answer has to do with the nature of evaluation in computer vision. Robotics is different from much of computer vision in that a robot must interact with a dynamic environment, not just images or videos downloaded from the

Internet. Therefore, a successful algorithm must generalize to numerous novel settings, which shifts the emphasis away from a singular focus on computing the best summary statistic (e.g. average accuracy, area under the curve, precision, recall) over a canned dataset. Recent catastrophic failures of autonomous vehicles relying on convolutional neural networks (Lohr, 2016) highlight this disconnect: when a summary statistic indicates that a dataset has been solved, it does not necessarily mean that the problem itself has been solved. The consequences of this observation are potentially far reaching if algorithms are deployed without a thorough understanding of their strengths and weaknesses (Anthony, 2016).

Whereas there are numerous flaws lurking in the shadows of deep learning benchmarks (Bendale and Boult, 2016; Cox and Dean, 2014; Nguyen et al., 2015b; Szegedy et al., 2013), two key aspects are worth discussing here: (1) the open set nature of decision making in visual recognition problems related to robotics; and (2) the limitations of traditional dataset evaluation in helping us understand the capabilities of an algorithm. *Open set recognition* refers to scenarios where incomplete knowledge of the world is present at training time, and unknown classes can be submitted to an algorithm during its operation (Scheirer et al., 2013b). It is absolutely critical to ask what the dataset is not capturing before setting a trained model loose to perform in the real world. Moreover, if a claim is made about the human-level (or, as we have been hearing lately, superhuman-level) performance of an algorithm, human behavior across varying conditions should be the frame of reference, not just a comparison of summary statistics on a dataset. This leads us to suggest *visual psychophysics* as a sensible alternative for evaluation.

3.1. The importance of open set recognition

In an autonomous vehicle setting, one can envision an object detection model trained to recognize other cars, while rejecting trees, signs, telephone poles, and any other non-car object in the scene. The challenge in obtaining good performance from this model is in the necessary generalization to all non-car objects, both known and unknown. Instead of casting such a detection task as a binary decision problem like most popular classification strategies would do, it is perhaps more useful to think about it within the context of the following taxonomy (Scheirer et al., 2014b), inspired by some memorable words spoken by Rumsfeld (2002).

- *Known classes*: the classes with distinctly labeled positive training examples (also serving as negative examples for other known classes).
- *Known unknown classes*: labeled negative examples, not necessarily grouped into meaningful categories.
- *Unknown unknown classes*: classes unseen in training. These samples are the most problematic for machine learning.

It is not the case that the feature space produced by a deep learning method should help us with the unknown classes? After all, the advantage of deep learning is the ability to learn separable feature representations that are strongly invariant to changing scene conditions. The trouble we find is not necessarily with the features themselves, but in the read-out layer used for decision making. Consider the following problems with three popular classifiers used as read-out layers for convolutional neural networks when applied to recognition tasks where unknown classes are present. A linear support vector machine (SVM) separates the positive and negative classes by a single linear decision boundary, establishing two half-spaces. These half-spaces are infinite in extent, meaning unknown samples far from the support of known training data can receive a positive label (Scheirer et al., 2014b). The Softmax function is a common choice for multi-class classification, but computing it requires calculating a summation over all of the classes. This is not possible when unknown classes are expected at testing time (Bendale and Boult, 2016). Along these same lines, when used to make a decision, cosine similarity requires a threshold, which can only be estimated over known data. The difficulty of establishing decision boundaries that capture a large measure of intraclass variance while rejecting unknown classes underpins several well-known deficiencies in deep learning architectures (Nguyen et al., 2015b; Szegedy et al., 2013).

It is readily apparent that we do not understand decision boundary modeling as well as we should. Accordingly, we suggest that researchers give more attention to decision making at an algorithmic level to address the limitations of existing classification mechanisms. What is needed is a new class of machine learning algorithms that minimize the risk of the unknown. Preliminary work exploring this idea has included slab-based linear classifiers to limit the risk of half-spaces (Scheirer et al., 2013b), nearest non-outlier models (Bendale and Boult, 2015), and extreme value theory-based calibration of decision boundaries (Bendale and Boult, 2016; Scheirer et al., 2014b; Zhang and Patel, 2016). Much more work is needed in this direction, including algorithms that incorporate the risk of the unknown directly into their learning objectives, and evaluation protocols that incorporate data which is both known and unknown to a model.

3.2. The role visual psychophysics should play

One need not resort to tricky manipulations such as noise patterns that are imperceptible to humans (Szegedy et al., 2013) or carefully evolved images (Nguyen et al., 2015b) to fool recognition systems based on deep learning. Simple transformations such as rotation, scale, and occlusion will do the job just fine. Remarkably, a systematic study of a recognition model's performance across an exhaustive range of object appearances is typically not done during the course of machine learning research. This is a major

shortcoming of evaluation within the field. Turning to the study of biological vision systems, psychologists and neuroscientists do perform such tests on humans and animals using a set of concepts and procedures from the discipline of psychophysics. Psychophysics allows scientists to probe the inner mechanisms of visual processing through the controlled manipulation of the characteristics of visual stimuli presented to a subject. The careful management of stimulus construction, ordering, and presentation allows a perceptual threshold, the inflection point at which perception transitions from success to failure, to be determined precisely. As in biological vision, we would like to know under what conditions a machine learning model is able to operate successfully, as well as where it begins to fail. If this is to be done in an exhaustive manner, we need to leverage item response theory (Embretson and Reise, 2000), which will let us map each stimulus condition to a performance point (e.g. model accuracy). When individual item responses are collected to form a curve, an exemplar-by-exemplar summary of the patterns of error for a model becomes available, allowing us to point exactly to the condition(s) that will lead to failure.

Psychophysics is commonplace in the laboratory, but how exactly can it be applied to models? One possibility is through a computational pipeline that is able to perturb 2D natural images or 3D rendered scenes at a massive scale (e.g. millions of images per image transformation being studied) and submit them to a model, generating an item–response curve from the resulting recognition scores (RichardWebster et al., 2016). Key to the interpretability of the results is the ability to identify a model's *preferred view*. Work in vision science has established that humans possess an internalized canonical view (the visual appearance that is easiest to recognize) for individual object classes (Blanz et al., 1999). Similarly, recognition models have one or more preferred views of an object class, each of which leads to a maximum (or minimum) score output. A preferred view, thus, forms a natural starting place for model assessment. Through perturbation, the results will at best stay the same, but more likely will degrade as visual appearance moves outside the variance learned from the training dataset. With respect to the stimuli used when performing psychophysics experiments on models, there is a growing trend in robotics and computer vision to make use of simulations rendered via computer graphics. In line with this, we believe that procedurally rendered graphics hold much promise for psychophysics experiments, where the position of objects can be manipulated in three dimensions, and aspects of the scene, such as lighting and background, changed at will.

Instead of comparing summary statistics related to benchmark dataset performance for different models, relative performance can be assessed by comparing the respective item–response curves. Importantly, not only can any gaps between the behaviors of different models be assessed, but also potential gaps between human and model behavior. Validation by this procedure is necessary if a claim is going to made about a model matching (or exceeding) human performance. Summary statistics only reflect one data point over a mixture of scene conditions, which obscures the patterns of error we are often most interested in. Through experimentation, we have found that human performance vastly exceeds model performance even in cases where a problem has been assumed to be solved (e.g. human face detection (Scheirer et al., 2014a)). Whereas the summary statistics in those cases indicated that both humans and models were at the performance ceiling for the dataset at hand, the item–response curves from psychophysics experiments showed a clear gap between human and model performance. However, psychophysics need not entirely replace datasets. After all, we still need a collection of data from which to train the model, and some indication of performance on a collection of web-scale data is still useful for model screening. Steps should be taken to explore strategies for combining datasets and visual psychophysics to address some of the obvious shortcomings of deep learning.

4. The role of simulation for pixel-to-action robotics

Robotics, still dominated by complex processing stacks, could benefit from a similar revolution as seen in computer vision which would clear a path directly from pixels to torques and enable powerful gradient-driven end-to-end optimization. A critical difference is that robotics constitutes an interactive domain with sequential actions where supervised learning from static datasets is not a solution. *Deep reinforcement learning* is a new learning paradigm that is capable of learning end-to-end robotic control tasks, but the accomplishments have been demonstrated primarily in simulation, rather than on actual robot platforms (Gu et al., 2016; Heess et al., 2015; Levine and Abbeel, 2014; Lillicrap et al., 2015; Mnih et al., 2016; Schulman et al., 2015, 2016). However, demonstrating learning capabilities on real robots remains the bar by which we must measure the practical applicability of these methods. This poses a significant challenge, given the long, data-hungry training paradigm of pixel-based deep robotic learning methods and the relative frailty of research robots and their human handlers.

To make the challenge more concrete, consider a simple pixel-to-action learning task: reaching to a randomly placed target from a random start location, using a three-fingered Jaco robot arm (see Figure 2). Trained in the MuJoCo simulator using Asynchronous Advantage Actor-Critic (A3C) (Mnih et al., 2016), the current state-of-the-art robotic learning algorithm, full performance is only achieved after substantial interaction with the environment, on the order of 50 million steps, a number which is infeasible with a real robot. The simulation training, compared with the real robot, is accelerated because of fast rendering, multi-threaded learning algorithms, and the ability to

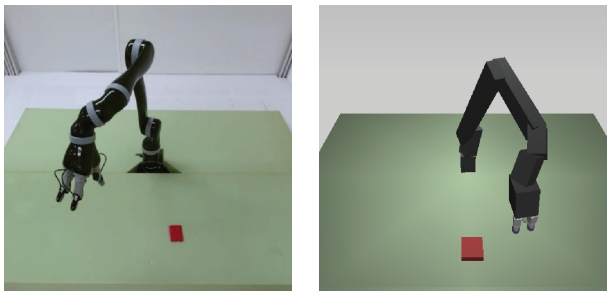

Fig. 2. Sample images from the real camera input image (left) and the MuJoCo-rendered image (right), demonstrating the reality gap between simulation and reality even for a simple reaching task.

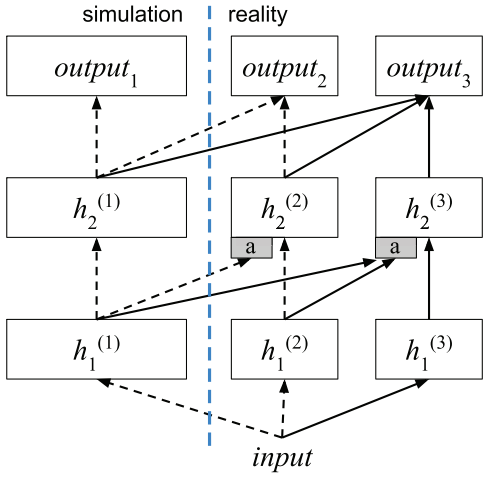

Fig. 3. Detailed schematic of progressive recurrent network architecture, where the left column is trained in simulation, then the weights are frozen while the second column is trained on the real robot. A third column may then be trained on an additional task, taking advantage of the policies and features learnt and frozen in the first two columns.

continuously train without human involvement. We calculate that learning this task, which trains to convergence in 24 hours using a CPU compute cluster, would take 53 days on the real robot even with continuous training for 24 hours a day. Moreover, multiple experiments in parallel were used to explore hyperparameters in simulation; this sort of search would compound further the hypothetical real robot training time.

Taking advantage of the simulation-learnt policies to train real robots is thus critical, but there is a *reality gap* that often separates a simulated task and its real-world analog, especially for raw pixel inputs. One solution is to use transfer learning methods to bridge the reality gap that separates simulation from real-world domains. There exist many different paradigms for domain transfer and many approaches designed specifically for deep neural models, but substantially fewer approaches for transfer from simulation to reality for robot domains. Even rarer are methods that can be used for transfer in interactive, rich sensor domains using end-to-end (pixel-to-action) learning. A growing body of work has been investigating the ability of deep networks to transfer between domains. Some research (Peng et al., 2015; Su et al., 2015) considers simply augmenting the target domain data with data from the source domain where an alignment exists. Building on this work, Long et al. (2015) started from the observation that as one looks at higher layers in the model, the transferability of the features decreases quickly. To correct this effect, a soft constraint is added that enforces the distribution of the features to be more similar. Long et al. (2015) proposed a "confusion" loss that forces the model to ignore variations in the data that separate the two domains (Tzeng et al., 2015b, 2014), and Tzeng et al. (2015a) attempted to address the simulation to reality gap by using aligned data. The work was focused on pose estimation of the robotic arm, where training happens on a triple loss that looks at aligned simulation to real data, including the domain confusion loss. The paper does not show the efficiency of the method on learning novel complex policies. Partial success on transferring from simulation to a real robot has been reported (Barrett et al., 2010; James and Johns, 2016; Zhang et al., 2015; Zhu et al., 2016). They focus primarily on the problem of transfer from a more restricted simpler version of a task to the full, more difficult version. Another promising recent direction is domain randomization (Sadeghi and Levine, 2016; Tobin et al., 2017).

A recent simulation-to-real approach relies on the *progressive nets* architecture (Rusu et al., 2016), which enables transfer learning through lateral connections that connect each layer of previously learnt deep networks to new networks, thus supporting deep compositionality of features (see Figure 3). Progressive networks are well suited for simulation-to-real transfer of policies in robot control domains for multiple reasons. First, features learnt for one task may be transferred to many new tasks without destruction from fine-tuning. Second, the columns may be heterogeneous, which may be important for solving different tasks, including different input modalities, or simply to improve learning speed when transferring to the real robot. Third, progressive nets add new capacity, including new input connections, when transferring to new tasks. This is advantageous for bridging the reality gap, to accommodate dissimilar inputs between simulation and real sensors.

Experiments with the Jaco robot showed that the progressive architecture is valuable for simulation-to-real transfer. The progressive second column gets to 34 points, whereas the experiment with fine-tuning, which starts with the simulation-trained column and continues training on the robot, does not reach the same score as the progressive network.

5. Deep learning and physics-based models

The predominant approach to perception, planning, and control in robotics is to use approximate models of the

physics underlying a robot, its sensors, and its interactions with the environment. These model-based techniques often capture properties such as the mass, momentum, shape, and surface friction of objects, and use these to generate controls that change the environment in a desirable way (Kuindersma et al., 2016; Kunze and Beetz, 2015; Schmidt et al., 2015; Todorov et al., 2012). Whereas physics-based models are well suited for planning and predicting the outcome of actions, to function on a real robot they require that all relevant model parameters are known with sufficient accuracy and can be tracked over time. This requirement poses overly challenging demands on system identification and perception, resulting in systems that are brittle, especially when direct interaction with the environment is required.

Humans, on the other hand, operate under intuitive rather than exact physical models (Baillargeon et al., 20111, 2012; Battaglia et al., 2013; Hespos et al., 2009; McCloskey, 1983; Povinelli, 2000). Whereas these intuitive models have many well-documented deficiencies and inaccuracies, they have the crucial property that they are grounded in real world experience, are well suited for closed-loop control, and can be learned and adapted to new situations. As a result, humans are capable of robustly performing a wide variety of tasks that are well beyond the reach of current robot systems, including dexterous manipulation, handling vastly different kinds of ingredients when cooking a meal, or climbing a tree.

Recent approaches to end-to-end training of deep networks forgo the use of explicit physics models, learning predictive models and controls from raw experiences (Battaglia et al., 2016; Byravan and Fox, 2017; C. Finn and Levine, 2016; Greff et al., 2017; Wu et al., 2015a; Yildirim et al., 2017). Whereas these early applications of large-scale deep learning are just the beginning, they have the potential to provide robots with highly robust perception and control mechanisms, based on an intuitive notion of physics that is fully grounded in a robot's experience.

The properties of model-based and deep-learned approaches can be measured along multiple dimensions, including the kind of representations used for reasoning, how generally applicable their solutions are, how robust they are in real-world settings, how efficiently they make use of data, and how computationally efficient they are during operation. Model-based approaches often rely on explicit models of objects and their shape, surface, and mass properties, and use these to predict and control motion through time. In deep learning, models are typically implicitly encoded via networks and their parameters. As a consequence, model-based approaches have wide applicability, since the physics underlying them are universal. However, at the same time, the parameters of these models are difficult to estimate from perception, resulting in rather brittle performance operating only in local basins of convergence. Deep learning, on the other hand, enables highly robust performance when trained on sufficiently large data sets that are representative of the operating regime of the system. However, the implicit models learned by current deep learning techniques do not have the general applicability of physics-based reasoning. Model-based approaches are significantly more data efficient, related to their smaller number of parameters. The optimizations required for model-based approaches can be performed efficiently, but the basin of convergence can be rather small. In contrast, deep-learned solutions are often very fast and can have very large basins of convergence. However, they do not perform well if applied in a regime outside the training data. Table 4 summarizes the main properties.

Different variants of deep learning have been shown to successfully learn predictive physics models and robot control policies in a purely data-driven way (Agrawal et al., 2016; Byravan et al., 2018; Jonschkowski et al., 2017; Watter et al., 2015). Although such a learning-based paradigm could potentially inherit the robustness of intuitive physics reasoning, current approaches are nowhere near human prediction and control capabilities. Key challenges toward achieving highly robust, physics-based reasoning and control for robots are as follows. (1) Learn general, predictive models for how the environment evolves and how it reacts to a robot's actions. Although the first attempts in this direction show promising results, these only capture very specific scenarios and it is not clear how they can be made to scale to general predictive models. (2) Leverage existing physics-based models to learn intuitive models from less data. Several systems approach this problem in promising ways, such as using physics-based models to generate training data for deep learning or developing deep network structures that incorporate insights from physics-based reasoning. (3) Learn models and controllers at multiple levels of abstractions that can be reused in many contexts. Rather than training new network structures for each task, such an approach would enable robots to fully leverage previously learned knowledge and apply it in new contexts.

6. Towards an automation of informatics

Deep learning will change the foundations of computer science. Already, the successes of deep learning in various domains are calling into question the dominant problem-solving paradigm: algorithm design.[1] This can easily be seen in the area of image classification, where deep learning has outperformed all prior attempts of explicitly programming image-processing algorithms. In contrast to most other applications of machine learning that require the careful design of problem-specific features, deep learning approaches require little to no knowledge of the problem domain. Sure, the search for a suitable network architectures and training procedures remains but the amount of domain-specific knowledge required to apply deep learning methods to novel problem domains is substantially lower than for programming a solution explicitly. As a result,

Table 4. Models versus deep learning

	Model-based	Deep learning
Representation	Explicit: based on or inspired by physics	Implicit: network structure and parameters
Generality	Broadly applicable: physics are universal	Only in trained regime: risk of overfitting
Robustness	Small basin of convergence: requires good models and estimates thereof	Large basin of convergence: highly robust in trained regime
Data efficiency	Very high: only needed for system identification	Training requires significant data collection effort
Computational efficiency	Good in local regime	Highly efficient once trained

the amount of problem-specific expertise required to solve complex problems has reached an all-time low. Whether this is good or bad remains to be seen (it is probably neither and both). However, it might seem that deep learning is currently the winner in the competition between "traditional" *programming* and the clever use of large amounts of *data*.

6.1. Programming versus data

Solutions to computational problems lie on a spectrum along which the relative and complementary contributions of programming and data vary. On one end of the spectrum lies traditional computer science: human experts program problem-specific algorithms that require no additional data to solve a particular problem instance, e.g. quicksort. On the other extreme lies deep learning. A generic approach to learning leverages large amounts of data to find a computational solution automatically. In between these two extremes lie algorithms that are less generic than deep learning and less specific than quicksort, including maybe decision trees for example.

It is helpful to look at the two ends of the spectrum in more detail. The act of *programming* on one end of the spectrum is replaced by *training* on the other end. The concept of *program* is turned into *learning weights* of the network. The programming language, i.e. the language in which a solution is expressed, is replaced by network architecture, loss function, training procedure, and data. Please note that the training procedure itself is again seen as a concrete algorithm, on the opposing end of the spectrum. This already alludes to the fact that solutions to challenging problems probably must combine sub-solutions from the entire spectrum spanned by programming and deep learning.

6.2. Does understand imply one end of the spectrum?

For a programmer to solve a problem through programming, we might say that they have to *understand* the problem. Computer programs therefore reflect human understanding. We might also say that the further a particular solution is positioned towards the deep-learning end of the spectrum, the less understanding about the problem it requires. As science strives for understanding, we should ultimately attempt to articulate the structure of our solutions explicitly, relying on as little data as possible for solving a particular problem instance. There are many reasons for pursuing this goal: robustness, transfer, generality, verifiability, re-use, and ultimately insight, which might lead to further progress.

Consider, for example, the problem of tracking the trajectory of a quad-copter. We can certainly come up with a deep-learning solution to this problem. However, would we not expect the outcome of learning, given an arbitrary amount of data and computational resources, to be some kind of Bayes filter? Either we believe that the Bayes filter captures the computational structure inherent to this problem (recursive state estimation), and then a learned solution eventually has to discover and represent this solution. However, at that point we might simply use the algorithm instead of the deep neural network. If, on the other hand, the deep neural network represents something else than a Bayes filter, something outperforming the Bayes filter, then we discovered that Bayes filters do not adequately capture the structure of the problem at hand. We will naturally be curious as to what the neural network discovered.

From this, we should draw three conclusions. First, our quest for understanding implies that we must try to move towards the programming-end of the spectrum, whenever we can. Second, we need to be able to leverage generic tools, such as deep learning, to discover problem structure; this will help us derive novel knowledge and to devise algorithms based on that knowledge. Third, we should understand how problems can be divided into parts: those parts for which we know the structure (and, therefore, can write algorithms for) and those for which we would like to discover the structure. This will facilitate the component-wise movement towards explicit understanding.

6.3. Generic tools might help us identify new structure

When we do not know how to program a solution for a problem and instead apply a generic learning method, such as deep learning, and this generic method delivers a solution, then we have implicitly learned something about the problem. It might be difficult to extract this knowledge from a deep neural network but that should simply

motivate us to develop methods for extracting this knowledge. Towards this goal, our community should (a) report in detail on the limitations of deep networks and (b) study in similar detail the dependencies of deep-learning solutions on various parameters. This will lead the way to an ability of "reading" networks so as to extract algorithmifiable information.

There have been some recent results about "distilling" knowledge from neural networks, indicating that the extraction of problem structure from neural networks might be possible (Hinton et al., 2015). Such distilled knowledge is still far away from being algorithmifiable, but this line of work seem promising in this regard. The idea of distillation can also be combined with side information (Lopez-Paz et al., 2016), further facilitating the identification of relevant problem structure.

On the other hand, it was shown that our insights about generalization, an important objective for machine learning algorithms, might not transfer easily to neural networks (Zhang et al., 2016). If it turns out that the deep neural networks we learn today simply memorize training data and then interpolate between them (Zhang et al., 2016), then we must develop novel regularization methods to enforce the extraction of problem structure instead of memorization, possibly through the use of side information (Jonschkowski et al., 2015). Otherwise, if neural networks are only good for memorization, they are not as powerful as we thought. There might be evidence, however, that neural networks do indeed find good representations, i.e. problem structure.

6.4. Complex problems should be solved by decomposition and re-composition

In many cases, interesting and complex problems will exhibit complex structure because they are composed of sub-problems. For each of these sub-problems, computational solutions are most appropriate that lie on different points along the programming/data-spectrum; this is because we may have more or less understanding of the sub-problem's inherent structure. It would therefore make sense to compose solutions to the original problem from sub-solutions that lie on different points on the programming/data-spectrum (Jonschkowski and Brock, 2016).

For many sub-problems, we already have excellent algorithmic solutions, e.g. implementations of quicksort. Sorting is a problem on one end of the spectrum: we understand it and have codified that understanding in an algorithm. However, there are many other problems, such as image classification, where human programs are outperformed by deep neural networks. Those problem should be solved by neural networks and then integrated with solutions from other parts of the spectrum.

This re-composition of component solutions from different places on the spectrum can be achieved with differentiable versions of existing algorithms (one end of the spectrum) that are compatible solutions obtained with back-propagation (other end of the spectrum) (Byravan and Fox, 2017; Haarnoja et al., 2016; Jonschkowski and Brock, 2016; Shi and Griffiths, 2009; Tamar et al., 2016; Wilson and Finkel, 2009). For example, Jonschkowski and Brock (2016) solve the aforementioned localization problem for quad-copters by combining a histogram filter with back-propagation-learned motion and sensing models.

6.5. Decomposability of problems

In the previous section, we argued that complex problems often are decomposable into sub-problems that can be solved independently. A problem is called *decomposable* or *near-decomposable* (Simon, 1996) if there is little complexity in the interactions among sub-problems and most of the complexity is handled within those sub-problems. However, are all problems decomposable in this manner? For example, Schierwagen (2012) argued that the brain is not decomposable because the interactions between its components still contain much of the complexity of the original problem. Furthermore, many interpret results on end-to-end learning of deep visuomotor policies to indicate that modular sub-solutions automatically lead to poor solutions (Levine et al., 2015). Of course, a sub-optimal factorization of a problem into sub-problems will lead to sub-optimal solutions. However, the results presented by Levine et al. (2015) do not lend strong support to this statement. The authors showed that end-to-end learning, i.e. giving up strict boundaries between sub-problems improves their solution. However, it is unclear whether this is an artifact of overfitting, an indication of a poor initial factorization, or an indication of the fact that even correct factorizations may exclude parts of the solution space containing the optimal solution.

Irrespective of the degree of decomposability of a problem (and the suitable degree of modularity of the solution), we suspect that there are optimal factorizations of problems for a defined task, agent, and environment. Such a factorization may not always lead to simple interfaces between sup-problems but always facilitates finding an optimal solution.

6.6. Automating programming

Once we are able to (1) decompose problems into sub-problems, (2) solve those sub-problems with solutions from different points along the programming/data-spectrum, *3)* recompose the solutions to sub-problems, and (4) extract algorithmic information from data-driven solutions, we might as well automate programming (computer science) altogether. Programming should be easy to automate, as it takes place entirely within the well-defined world of the computer. If we can successfully apply generic methods to complex problems, extract and algorithmify structural knowledge from the resulting solutions, use the resulting

algorithms to solve sub-problems of the original problem, thereby making that original problem more easily solvable, and so forth, then we can also imagine an automated way of deriving computer algorithms from problem-specific data. A key challenge will be the automatic decomposition or *factorization* of the problem into suitably solvable sub-problems.

This view raises some fundamental questions about the differences between *program* in programming and *weights* in deep learning. Really, this view implies that there is no qualitative difference between them, only a difference of expressiveness and the amount of prior assumptions reflected in them. Programs and weights, in this view, are different instances of the same thing, namely of parameters that specify a solution, given a framework for expressing such solutions. Now it seems plausible that we can incrementally extract structure from learned parameters (weights), leading to a less generic representation with fewer parameters, until the parameters are so specific that we might call them a program.

However, the opposite is also possible. It is possible that problems exists that do not exhibit algorithmifiable structure. It is possible that these problems can (only) be solved in a data-driven manner. To speculate about this, comparisons with biological cognitive capabilities might be helpful. Can these capabilities (in principle) be encoded in a program? Do these capabilities depend on massive amounts of data? These are difficult questions that artificial intelligence researchers have asked themselves for many years.

6.7. Priors to reduce the amount of data

A natural concern for this kind of reasoning is the necessity to acquire large amounts of data. This can be very costly, especially when these data have to be acquired from interaction with the real world, as is the case in robotics. It will then become necessary to reduce the required amount of data by incorporating appropriate priors into learning (Jonschkowski and Brock, 2015). These priors reduce all possible interpretations of data to only those consistent with the prior. If sufficiently strong priors are available, it will become possible to extract (and possibly algorithmify) the problem structure from reasonable amounts of data.

It might also be difficult to separate acquired data into those groups associated with a single task. Recent methods have shown that this separation can be performed automatically (Höfer et al., 2016). Now data can be acquired in less-restrictive settings and the learning agent can differentiate the task associated with a datum by itself.

6.8. Where will this lead?

Maybe in the end, the most lasting impact of deep learning will not be deep learning itself but rather the effect it had. The successes of deep learning, achieved by leveraging data and computation, have made computer scientists realize that there is a spectrum, rather than a dichotomy, between programming and data. This realization may pave the way for a computer science that fully leverages the entire breadth of this spectrum to automatically derive algorithms from reasonable amounts of data and suitable priors.

7. Conclusions

The rather skeptical attitude towards deep learning at the RSS conference in Rome 2015 motivated us to organize a workshop at RSS 2016 with the title *"Are the Skeptics Right? Limits and Potentials of Deep Learning in Robotics"* (Sünderhauf et al., 2016). As it turned out, by then there were hardly any skeptics left. The robotics community had accepted deep learning as a very powerful tool and begun to utilize and advance it. A follow-up workshop on *"New Frontiers for Deep Learning in Robotics"* (Sünderhauf et al., 2017) at RSS 2017 concentrated more on some of the robotics-specific research challenges we discussed in this paper. A surge of deep learning in robotics was seen in 2017: workshops at CVPR (Angelova et al., 2017) and NIPS (Posner et al., 2016) built bridges between the robotics, computer vision, and machine learning communities. Over 10% of the papers submitted to ICRA 2018 used *Deep learning in robotics and automation* as a keyword, making it the most frequent keyword. Furthermore, a whole new Conference on Robot Learning (CoRL)[2] was initiated.

Although much ongoing work in deep learning for robotics concentrates on either perception or acting, we hope to see more integrated approaches in the future: robots that learn to utilize their embodiment to reduce the uncertainty in perception, decision making, and execution. Robots that learn complex multi-stage tasks, while incorporating prior model knowledge or heuristics, and exploiting a semantic understanding of their environment. Robots that learn to discover and exploit the rich semantic regularities and geometric structure of the world, to operate more robustly in realistic environments with open-set characteristics.

Deep learning techniques have revolutionized many aspects of computer vision over the past 5 years and have been rapidly adopted into robotics as well. However, robotic perception, robotic learning, and robotic control are demanding tasks that continue to pose severe challenges on the techniques typically applied. Our paper discussed some of these current research questions and challenges for deep learning in robotics. We pointed the reader into different directions worthwhile for further research and hope our paper contributes to the ongoing advancement of deep learning for robotics.

Funding

This work was supported by the Australian Research Council Centre of Excellence for Robotic Vision (project number CE140100016). Oliver Brock was supported by the DFG (grant number 329426068). Walter Scheirer acknowledges the funding

provided by IARPA (contract number D16PC00002). Michael Milford was partially supported by an Australian Research Council Future Fellowship (FT140101229).

Notes

1. The term *algorithm* refers to the *Oxford Dictionary* definition: "a process or set of rules to be followed in calculations or other problem-solving operations." Here, it includes physics formulas, computational models, probabilistic representations and inference, etc.
2. See http://www.robot-learning.org

References

Agrawal P, Nair A, Abbeel P, Malik J and Levine S (2016) Learning to poke by poking: Experiential learning of intuitive physics. In: *Advances in Neural Information Processing Systems (NIPS)*.

Angelova A, Carneiro G, Murphy K, et al. (2017) Computer Vision and Pattern Recognition (CVPR) Workshop on Deep Learning in Robotic Vision. Available at: http://juxi.net/workshop/deep-learning-robotic-vision-cvpr-2017/ (accessed 31 March 2018).

Anthony SE (2016) The trollable self-driving car. Slate. Available at: http://www.slate.com/articles/technology/future_tense/2016/03/google_self_driving_cars_lack_a_human_s_intuition_ for_what_other_drivers.html (accessed 31 March 2018).

Atanasov N, Sankaran B, Le Ny J, Pappas GJ and Daniilidis K (2014) Nonmyopic view planning for active object classification and pose estimation. *IEEE Transactions on Robotics* 30(5): 1078–1090.

Baillargeon R, Li J, Gertner Y and Wu D (20111) How do infants reason about physical events? In: *The Wiley-Blackwell Handbook of Childhood Cognitive Development* (2nd edn). Oxford: Blackwell.

Baillargeon R, Stavans M, Wu D, et al. (2012) Object individuation and physical reasoning in infancy: An integrative account. *Language Learning and Development* 8(1): 4–46.

Barrett S, Taylor ME and Stone P (2010) Transfer learning for reinforcement learning on a physical robot. In: *Ninth International Conference on Autonomous Agents and Multiagent Systems - Adaptive Learning Agents Workshop (AAMAS - ALA)*.

Battaglia P, Pascanu R, Lai M, Rezende DJ et al. (2016) Interaction networks for learning about objects, relations and physics. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 4502–4510.

Battaglia PW, Hamrick JB and Tenenbaum JB (2013) Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences* 110(45): 18327–18332.

Bendale A and Boult TE (2015) Towards open world recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1893–1902.

Bendale A and Boult TE (2016) Towards open set deep networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1563–1572.

Bertinetto L, Henriques JF, Valmadre J, Torr P and Vedaldi A (2016) Learning feed-forward one-shot learners. In: *Advances in Neural Information Processing Systems*, pp. 523–531.

Bircher A, Kamel M, Alexis K, Oleynikova H and Siegwart R (2016) Receding horizon "next-best-view" planner for 3D exploration. In: *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 1462–1468.

Blanz V, Tarr MJ and Bülthoff HH (1999) What object attributes determine canonical views? *Perception* 28(5): 575–599.

Byravan A and Fox D (2017) SE3-nets: Learning rigid body motion using deep neural networks. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Byravan A, Leeb F, Meier F and Fox D (2018) SE3-Pose-Nets: Structured deep dynamics models for visuomotor planning and control. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

C Finn IG and Levine S (2016) Unsupervised learning for physical interaction through video prediction. In: *Advances in Neural Information Processing Systems (NIPS)*.

Cadena C, Carlone L, Carrillo H, et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics* 32(6): 1309–1332.

Choy CB, Xu D, Gwak J, Chen K and Savarese S (2016) 3D-R2N2: A unified approach for single and multi-view 3D object reconstruction. In: *European Conference on Computer Vision (ECCV)*. New York: Springer, pp. 628–644.

Cohn DA, Ghahramani Z and Jordan MI (1996) Active learning with statistical models. *Journal of Artificial Intelligence Research* 4(1): 129–145.

Cox DD and Dean T (2014) Neural networks and neuroscience-inspired computer vision. *Current Biology* 24(18): R921–R929.

Csurka G (2017) Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*.

Dayoub F, Sünderhauf N and Corke P (2017) Episode-based active learning with Bayesian neural networks. In: *CVPR Workshop on Deep Learning for Robotic Vision. arXiv preprint arXiv:1703.07473*.

Doumanoglou A, Kouskouridas R, Malassiotis S and Kim TK (2016) Recovering 6D object pose and predicting next-best-view in the crowd. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3583–3592.

Embretson SE and Reise SP (2000) *Item Response Theory for Psychologists*. Lawrence Erlbaum Associates, Inc.

Finn C, Abbeel P and Levine S (2017) Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*.

Gal Y and Ghahramani Z (2016) Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In: *International Conference on Machine Learning (ICML)*, pp. 1050–1059.

Gal Y, Islam R and Ghahramani Z (2017) Deep Bayesian active learning with image data. *arXiv preprint arXiv:1703.02910*.

Ganin Y, Ustinova E, Ajakan H, et al. (2015) Domain-adversarial training of neural networks. *CoRR* abs/1505.07818.

Garg R, Carneiro G and Reid I (2016) Unsupervised CNN for single view depth estimation: Geometry to the rescue. In: *European Conference on Computer Vision*. New York: Springer, pp. 740–756.

Godard C, Mac Aodha O and Brostow GJ (2017) Unsupervised monocular depth estimation with left–right consistency. In: *Computer Vision and Pattern Recognition (CVPR)*.

Goldstein EB and Brockmole J (2016) *Sensation and perception*. Cengage Learning.

Goodfellow IJ, Mirza M, Xiao D, Courville A and Bengio Y (2013) An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Goodfellow IJ, Shlens J and Szegedy C (2014) Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Greff K, van Steenkiste S and Schmidhuber J (2017) Neural expectation maximization. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 6694–6704.

Gu S, Lillicrap TP, Sutskever I and Levine S (2016) Continuous deep Q-learning with model-based acceleration. In: *ICML 2016*.

Guo C, Pleiss G, Sun Y and Weinberger KQ (2017) On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*.

Haarnoja T, Ajay A, Levine S and Abbeel P (2016) Backprop KF: learning discriminative deterministic state estimators. *CoRR* abs/1605.07148.

Häne C, Tulsiani S and Malik J (2017) Hierarchical surface prediction for 3D object reconstruction. *arXiv preprint arXiv:1704.00710*.

Hariharan B and Girshick R (2016) Low-shot visual recognition by shrinking and hallucinating features. *arXiv preprint arXiv:1606.02819*.

He K, Gkioxari G, Dollár P and Girshick R (2017) Mask R-CNN. In: *IEEE International Conference on Computer Vision (ICCV)*.

Heess N, Wayne G, Silver D, Lillicrap TP, Erez T and Tassa Y (2015) Learning continuous control policies by stochastic value gradients. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, 7–12 December 2015, Montreal, Quebec, Canada, pp. 2944–2952. Available at: http://papers.nips.cc/paper/5796-learning-continuous-control-policies-by-stochastic-value-gradients (accessed 31 March 2018).

Hendrycks D and Gimpel K (2017) A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: *International Conference on Machine Learning (ICML)*.

Hespos S, Gredeba G, von Hofsten C and Spelke E (2009) Occlusion is hard: Comparing predictive reaching for visible and hidden objects in infants and adults. *Cognitive Science* 33(8): 1483–1502.

Hinton G, Vinyals O and Dean J (2015) Distilling the knowledge in a neural network. *CoRR* abs/1503.02531.

Höfer S, Raffin A, Jonschkowski R, Brock O and Stulp F (2016) Unsupervised learning of state representations for multiple tasks. In: *Deep Learning Workshop at the Conference on Neural Information Processing Systems (NIPS)*.

James S and Johns E (2016) 3D simulation for robot arm control with deep Q-learning. *ArXiv e-prints*.

Jonschkowski R and Brock O (2015) Learning state representations with robotic priors. *Autonomous Robots* 39(3): 407–428.

Jonschkowski R and Brock O (2016) End-to-end learnable histogram filters. In: *Workshop on Deep Learning for Action and Interaction at the Conference on Neural Information Processing Systems (NIPS)*.

Jonschkowski R, Hafner R, Scholz J and Riedmiller M (2017) Pves: Position-velocity encoders for unsupervised learning of structured state representations. *arXiv preprint arXiv:1705.09805*.

Jonschkowski R, Höfer S and Brock O (2015) Contextual learning. *CoRR* abs/1511.06429.

Kaess M, Johannsson H, Roberts R, Ila V, Leonard J and Dellaert F (2012) iSAM2: Incremental smoothing and mapping using the Bayes tree. *The International Journal of Robotics Research* 31(2): 216–235.

Kendall A and Gal Y (2017) What uncertainties do we need in bayesian deep learning for computer vision? *arXiv preprint arXiv:1703.04977*.

Kersten D, Mamassian P and Yuille A (2004) Object perception as Bayesian inference. *Annual Review of Psychology* 55: 271–304.

Kingma DP, Mohamed S, Rezende DJ and Welling M (2014) Semi-supervised learning with deep generative models. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 3581–3589.

Kuindersma S, Deits R, Fallon M, et al. (2016) Optimization-based locomotion planning, estimation, and control design for the Atlas humanoid robot. *Autonomous Robots* 40(3): 429–455.

Kümmerle R, Grisetti G, Strasdat H, Konolige K and Burgard W (2011) g2o: A general framework for graph optimization. In: *Proceedings of International Conference on Robotics and Automation (ICRA)*, pp. 3607–3613.

Kunze L and Beetz M (2015) Envisioning the qualitative effects of robot manipulation actions using simulation-based projections. *Artificial Intelligence* 247: 352–380.

Lake BM, Salakhutdinov R and Tenenbaum JB (2015) Human-level concept learning through probabilistic program induction. *Science* 350(6266): 1332–1338.

Lakshminarayanan B, Pritzel A and Blundell C (2017) Simple and scalable predictive uncertainty estimation using deep ensembles. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 6393–6395.

Levine S and Abbeel P (2014) Learning neural network policies with guided policy search under unknown dynamics. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND and Weinberger KQ (eds.) *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., pp. 1071–1079.

Levine S, Finn C, Darrell T and Abbeel P (2015) End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research* 17(1): 1334–1373.

Lillicrap TP, Hunt JJ, Pritzel A, et al. (2015) Continuous control with deep reinforcement learning. *CoRR* abs/1509.02971.

Lin D, Fidler S and Urtasun R (2013) Holistic scene understanding for 3D object detection with RGBD cameras. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1417–1424.

Liu F, Shen C, Lin G and Reid I (2016a) Learning depth from single monocular images using deep convolutional neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38(10): 2024–2039.

Liu W, Anguelov D, Erhan D, et al. (2016b) SSD: Single shot multibox detector. In: *European Conference on Computer Vision*. New York: Springer, pp. 21–37.

Lohr S (2016) A lesson of Tesla crashes? Computer vision can't do it all yet. *The New York Times*, 19 September. Available

at: https://www.nytimes.com/2016/09/20/science/computer-vision-tesla-driverless-cars.html (accessed 31 March 2018).

Lomonaco V and Maltoni D (2017) Core50: a new dataset and benchmark for continuous object recognition. *arXiv preprint arXiv:1705.03550*.

Long M, Cao Y, Wang J and Jordan MI (2015) Learning transferable features with deep adaptation networks. In: *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, Lille, France, 6–11 July 2015, pp. 97–105.

Lopez-Paz D, Bottou L, Schölkopf B and Vapnik V (2016) Unifying distillation and privileged information. *CoRR* abs/1511.03643.

MacKay DJ (1992) A practical Bayesian framework for backpropagation networks. *Neural Computation* 4(3): 448–472.

Malmir M, Sikka K, Forster D, Fasel I, Movellan JR and Cottrell GW (2017) Deep active object recognition by joint label and action prediction. *Computer Vision and Image Understanding* 156: 128–137.

McCloskey M (1983) Intuitive physics. *Scientific American* 248(4): 122–130.

Mensink T, Verbeek J, Perronnin F and Csurka G (2012) Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. *Computer Vision (ECCV 2012)*, pp. 488–501.

Miller D, Nicholson L, Dayoub F and Sünderhauf N (2017) Dropout sampling for robust object detection in open-set conditions. In: *International Conference on Robotics and Automation (ICRA)*.

Mnih V, Badia AP, Mirza M, et al. (2016) Asynchronous methods for deep reinforcement learning. In: *International Conference on Machine Learning (ICML)*.

Neal RM (1995) *Bayesian learning for neural networks*. PhD Thesis, University of Toronto.

Nguyen A, Yosinski J and Clune J (2015a) Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436.

Nguyen A, Yosinski J and Clune J (2015b) Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436.

Oliva A and Torralba A (2007) The role of context in object recognition. *Trends in Cognitive Sciences* 11(12): 520–527.

Papandreou G, Chen LC, Murphy KP and Yuille AL (2015) Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1742–1750.

Patel VM, Gopalan R, Li R and Chellappa R (2015) Visual domain adaptation: A survey of recent advances. *IEEE Signal Processing Magazine* 32(3): 53–69.

Peng X, Sun B, Ali K and Saenko K (2015) Learning deep object detectors from 3D models. In: *2015 IEEE International Conference on Computer Vision (ICCV 2015)*, Santiago, Chile, 7–13 December 2015, pp. 1278–1286.

Piaget J (2013) *The Construction of Reality in the Child*. London: Routledge.

Pillai S and Leonard J (2015) Monocular slam supported object recognition. In: *Robotics: Science and Systems*.

Posner I, Hadsell R, Riedmiller M, Wulfmeier M and Paul R (2016) *Neural Information Processing Systems (NIPS) Workshop on Acting and Interacting in the Real World: Challenges in Robot Learning*. Available at: http://sites.google.com/view/nips17robotlearning/home (accessed 31 March 2018).

Povinelli DJ (2000) *Folk Physics for Apes: The Chimpanzee's Theory of How the World Works*. Oxford: Oxford University Press.

Rasmus A, Berglund M, Honkala M, Valpola H and Raiko T (2015) Semi-supervised learning with ladder networks. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 3546–3554.

Rebuffi SA, Kolesnikov A, Sperl G and Lampert CH (2017) iCaRL: Incremental classifier and representation learning. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* Available at: http://openaccess.thecvf.com/content_cvpr_2017/papers/Rebuffi_iCaRL_Incremental_Classifier_CVPR_2017_paper.pdf (accessed 31 March 2018).

Redmon J and Farhadi A (2016) Yolo9000: better, faster, stronger. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Rezende D, Danihelka I, Gregor K, et al. (2016) One-shot generalization in deep generative models. In: *International Conference on Machine Learning*, pp. 1521–1529.

RichardWebster B, Anthony SE and Scheirer WJ (2016) Psyphy: A psychophysics driven evaluation framework for visual recognition. *CoRR* abs/1611.06448.

Rock I (1983) *The Logic of Perception*. Cambridge: MIT Press.

Rumsfeld D (2002) DoD News Briefing addressing *unknown unknowns.* Available at: http://archive.defense.gov/Transcripts/Transcript.aspx?TranscriptID=2636 (accessed 31 March 2018).

Rusu A, Rabinowitz N, Desjardins G, et al. (2016) Progressive neural networks. *arXiv preprint arXiv:1606.04671*.

Sadeghi F and Levine S (2016) Cad2rl: Real single-image flight without a single real image. *arXiv preprint arXiv:1611.04201*.

Salas-Moreno RF, Newcombe RA, Strasdat H, Kelly PH and Davison AJ (2013) SLAM++: Simultaneous localisation and mapping at the level of objects. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1352–1359.

Santoro A, Bartunov S, Botvinick M, Wierstra D and Lillicrap T (2016) Meta-learning with memory-augmented neural networks. In: *International Conference on Machine Learning*, pp. 1842–1850.

Scheirer WJ, Anthony SE, Nakayama K and Cox DD (2014a) Perceptual annotation: Measuring human vision to improve computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36(8): 1679–1686.

Scheirer WJ, de Rezende Rocha A, Sapkota A and Boult TE (2013a) Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(7): 1757–1772.

Scheirer WJ, de Rezende Rocha A, Sapkota A and Boult TE (2013b) Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(7): 1757–1772.

Scheirer WJ, Jain LP and Boult TE (2014b) Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36(11): 2317–2324.

Schierwagen A (2012) On reverse engineering in the brain and cognitive sciences. *Natural Computing* 11(1): 141–150.

Schmidt T, Hertkorn K, Newcombe R, Marton Z, Suppa S and Fox D (2015) Robust real-time tracking with visual and physical

constraints for robot manipulation. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

Schulman J, Levine S, Moritz P, Jordan MI and Abbeel P (2015) Trust region policy optimization. In: *Proceedings of the 32nd International Conference on Machine Learning (ICML)*.

Schulman J, Moritz P, Levine S, Jordan M and Abbeel P (2016) High-dimensional continuous control using generalized advantage estimation. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.

Shi L and Griffiths TL (2009) Neural implementation of hierarchical bayesian inference by importance sampling. In: *Proceedings of the Neural Information Processing Systems Conference (NIPS)*.

Simon HA (1996) *The Sciences of the Artificial*. Cambridge, MA: MIT Press.

Su H, Qi CR, Li Y and Guibas LJ (2015) Render for CNN: viewpoint estimation in images using cnns trained with rendered 3d model views. In: *2015 IEEE International Conference on Computer Vision (ICCV 2015)*, Santiago, Chile, 7–13 December 2015, pp. 2686–2694.

Sünderhauf N, Dayoub F, McMahon S, et al. (2016) Place categorization and semantic mapping on a mobile robot. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 5729–5736.

Sünderhauf N, Leitner J, Abbeel P, Milford M and Corke P (2017) Robotics: Science and Systems (RSS) Workshop on New Frontiers for Deep Learning in Robotics. Available at: http://juxi.net/workshop/deep-learning-rss-2017/ (accessed 31 March 2018).

Sünderhauf N, Leitner J, Milford M, et al. (2016) Robotics: Science and Systems (RSS) Workshop Are the Sceptics Right? Limits and Potentials of Deep Learning in Robotics. Available at: http://juxi.net/workshop/deep-learning-rss-2016/ (accessed 31 March 2018).

Sünderhauf N, Pham TT, Latif Y, Milford M and Reid I (2017) Meaningful maps - object-oriented semantic mapping. In: *International Conference on Intelligent Robots and Systems (IROS)*.

Szegedy C, Zaremba W, Sutskever I, et al. *CoRR* abs/1312.6199.

Tamar A, Levine S and Abbeel P (2016) Value iteration networks. *CoRR* abs/1602.02867.

Thrun S, Burgard W and Fox D (2005) *Probabilistic Robotics*. Cambridge, MA: The MIT Press.

Tobin J, Fong R, Ray A, Schneider J, Zaremba W and Abbeel P (2017) Domain randomization for transferring deep neural networks from simulation to the real world. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 23–30.

Todorov E, Erez T and Tassa Y (2012) Mujoco: A physics engine for model-based control. In: *International Conference on Intelligent Robots and Systems IROS*.

Torralba A and Efros AA (2011) Unbiased look at dataset bias. In: *Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp. 1521–1528.

Tzeng E, Devin C, Hoffman J, et al. (2015a) Towards adapting deep visuomotor representations from simulated to real environments. *CoRR* abs/1511.07111.

Tzeng E, Hoffman J, Darrell T and Saenko K (2015b) Simultaneous deep transfer across domains and tasks. In: *2015 IEEE International Conference on Computer Vision (ICCV 2015)*, Santiago, Chile, 7–13 December 2015, pp. 4068–4076.

Tzeng E, Hoffman J, Zhang N, Saenko K and Darrell T (2014) Deep domain confusion: Maximizing for domain invariance. *CoRR* abs/1412.3474.

Vinyals O, Blundell C, Lillicrap T, et al. (2016) Matching networks for one shot learning. In: *Advances in Neural Information Processing Systems*, pp. 3630–3638.

Von Helmholtz H (1867) *Handbuch der physiologischen Optik*, volume 9. Voss.

Wang YX and Hebert M (2016) Learning to learn: Model regression networks for easy small sample learning. In: *European Conference on Computer Vision*. Berlin: Springer, pp. 616–634.

Watter M, Springenberg J, Boedecker J and Riedmiller M (2015) Embed to control: A locally linear latent dynamics model for control from raw images. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 2728–2736.

Wilson RC and Finkel LH (2009) A neural implementation of the Kalman Filter. In: *Proceedings of the Neural Information Processing Systems Conference (NIPS)*.

Wu J, Yildirim I, Lim JJ, Freeman B and Tenenbaum J (2015a) Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 127–135.

Wu Z, Song S, Khosla A, Yu F, Zhang L, Tang X and Xiao J (2015b) 3d shapenets: A deep representation for volumetric shapes. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yan X, Yang J, Yumer E, Guo Y and Lee H (2016) Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision. In: *Advances in Neural Information Processing Systems (NIPS)*, pp. 1696–1704.

Yildirim I, Gerstenberg T, Saeed B, Toussaint M and Tenenbaum J (2017) Physical problem solving: Joint planning with symbolic, geometric, and dynamic constraints. *arXiv preprint arXiv:1707.08212*.

Zhang C, Bengio S, Hardt M, Recht B and Vinyals O (2016) Understanding deep learning requires rethinking generalization. *CoRR* abs/1611.03530.

Zhang F, Leitner J, Milford M, Upcroft B and Corke P (2015) Towards vision-based deep reinforcement learning for robotic motion control. In: *Australasian Conference on Robotics and Automation*.

Zhang H and Patel V (2016) Sparse representation-based open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP(99): 1–1.

Zhang Y, Bai M, Kohli P, Izadi S and Xiao J (2017) Deepcontext: context-encoding neural pathways for 3D holistic scene understanding. In: *IEEE International Conference on Computer Vision (ICCV)*.

Zhu R, Galoogahi HK, Wang C and Lucey S (2017) Rethinking reprojection: Closing the loop for pose-aware shape reconstruction from a single image. In: *IEEE International Conference on Computer Vision (ICCV)*. IEEE, pp. 57–65.

Zhu Y, Mottaghi R, Kolve E, et al. (2016) Target-driven visual navigation in indoor scenes using deep reinforcement learning. *CoRR* abs/1609.05143.