Trends in
Cognitive Sciences

CellPress

Review

Next-generation deep learning based on simulators and synthetic data

Celso M. de Melo [1,*] Antonio Torralba,[2] Leonidas Guibas,[3] James DiCarlo,[4] Rama Chellappa,[5] and Jessica Hodgins[6]

Deep learning (DL) is being successfully applied across multiple domains, yet these models learn in a most artificial way: they require large quantities of labeled data to grasp even simple concepts. Thus, the main bottleneck is often access to supervised data. Here, we highlight a trend in a potential solution to this challenge: synthetic data. Synthetic data are becoming accessible due to progress in rendering pipelines, generative adversarial models, and fusion models. Moreover, advancements in domain adaptation techniques help close the statistical gap between synthetic and real data. Paradoxically, this artificial solution is also likely to enable more natural learning, as seen in biological systems, including continual, multimodal, and embodied learning. Complementary to this, simulators and deep neural networks (DNNs) will also have a critical role in providing insight into the cognitive and neural functioning of biological systems. We also review the strengths of, and opportunities and novel challenges associated with, synthetic data.

Highlights

Despite their initial successes, it is becoming apparent that modern deep learning (DL) models are hindered by an important bottleneck: the need for large quantities of annotated data to train the models.

Synthetic data provide a solution to this challenge. They are easy to generate, error-free, inexhaustible, pre-annotated, and avoid many ethical and practical concerns.

The past decade has experienced unprecedented progress in data synthesis and domain adaptation techniques that close the (statistical) gap between synthetic and real data.

Beyond sustaining the DL revolution, synthetic data will enable a next generation of DL models that understand the physical composition of the world and learn continually, multimodally, and interactively.

Integrated synthesis and learning pipelines can support life-long structured learning that is more similar to biological learning systems.

The bottleneck is labeled data

The past decade has experienced a revolution in interest and investment in DL that has enabled successful applications in visual perception, natural language processing, and robotic control, among others [1]. The success of DL benefited from converging trends in the development of algorithms to train these models (e.g., backpropagation), the availability of 'big data' (e.g., social media), and advances in computational power [e.g., powerful graphical processing units (GPUs)]. However, despite these initial successes, it is becoming apparent that the current generation of DNNs has important practical and theoretical limitations. DNNs are sample inefficient in that they require large amounts of annotated data (e.g., images of vehicles with bounding boxes) to optimize all their parameters (typically in the order of millions). Therefore, rather than algorithm or computational capability, the availability of annotated data is often the main bottleneck in the development of DL models. **Synthetic data** (see Glossary) and **simulators** have emerged as a promising solution to this challenge [2]. Synthetic data are comparatively easier to generate, inexhaustible, pre-annotated, and less expensive. Synthetic data also have the potential to avoid ethical (e.g., privacy concerns) and practical issues (e.g., security concerns). They further introduce unique opportunities in that they enable training data that may be impractical or impossible to collect in the real world.

More fundamentally, DNNs still lack important capabilities seen in biological systems. Humans can learn rich representations of the world, including its (hierarchical) compositional and physical nature [3,4]. Humans are more efficient learners, often being able to grasp novel concepts from a small sample of examples [5] and in a mostly unsupervised fashion [6]. Moreover, human learning is sophisticated, often relying on rich interactive experiences, in contrast to static data sets, which capture 'moments in time' (e.g., ImageNet). Synthetic data and simulators are a new catalyst for

[1]Computational and Information Sciences Directorate, DEVCOM US Army Research Laboratory, Playa Vista, CA, USA
[2]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, USA
[3]Computer Science Department, Stanford University, Stanford, CA, USA
[4]Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA, USA
[5]Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD, USA

these richer representations of the world and more sophisticated forms of learning, including **multimodal learning** [7] (e.g., fusing visual and audio information), **continual learning** [8] (e.g., understanding gradually more complex tasks in sequence), and **embodied learning** [9] (e.g., interactive exploratory play to understand object affordances). Complementary to this, simulators can be used to gather unique insight into biological systems [10,11]. By comparing how well different artificial neural models simulate cognitive functionality and predict brain activity, it becomes possible to test, validate, and extend existent theory [12–14]. Insofar as simulated data enable the training and testing DNNs, they therefore have an instrumental role in the study of biological systems. Simulated data further present novel opportunities for scientific exploration. By analyzing the properties of DNNs, it is possible to synthesize optimized stimuli to activate specific neural populations with relevant application to the study of brain function [13]. Simulated environments, perhaps even fully immersive (e.g., virtual reality), can further provide a unique opportunity for direct comparisons of behavior and neural activation in DNNs versus humans versus nonhuman primates in embodied interactive tasks. Therefore, synthetic data can further the development of artificial neural networks (ANNs) that model critical functions seen in biological systems, simultaneously contributing to our understanding of these systems and offering solutions with broad practical relevance.

[6]Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

*Correspondence:
celso.miguel.de.melo@gmail.com
(C.M. de Melo).

Here, we provide an overview of successful methodologies used to synthesize data for DL models, emphasizing the integration of the synthesis and machine learning pipelines. Next, we focus on a central challenge to using simulated data: aligning synthetic data to real data at both the pixel and feature level. We then articulate how synthetic data and simulators can enable DL solutions that can learn richer representations of the world and in more sophisticated ways, while simultaneously providing insight into the biological systems they draw inspiration from.

Synthesizing data and integrating with the deep learning pipeline

Progress in computer graphics tools, such as game engines (e.g., Unity and Unreal), and the increasing availability of 3D assets, are making it easier to develop simulators for custom domains (Figure 1A). This approach has been used to synthesize training data for a variety of tasks, including object detection [15–17], object tracking [18,19], viewpoint estimation [20], semantic segmentation [21–23], robot manipulation and control [24–28], pose estimation [29–31], gaze estimation [32], and activity recognition [33,34] (for a detailed review of simulators and synthetic data sets, see [35]). Across these diverse domains, synthetic data have often improved DNN performance when tested in real domains, especially when combined with real data. In this approach, synthesis relies on a **computer graphics**-**rendering pipeline**, which takes as input 3D information about the scene (e.g., points in 3D space specifying a vehicle), information about the materials and lighting properties (e.g., vehicle color and light sources), and rendering parameters (e.g., rasterization or raytracing algorithm), and produces a 3D visualization of the scene (Figure 1D). Since the pipeline has information about the scene details, it can automatically generate error-free ground-truth (e.g., bounding boxes for objects of interest, depth information, and scene segmentation masks). By increasing the amount of 3D information (e.g., the number of 3D vertices specifying the objects of interest) and the sophistication of the algorithms used to render the scene, it is possible to increase the visual realism of the output (i.e., the visual fidelity of the scene compared with the real world). Similarly, it is possible to increase the motion realism of the output by using 3D motion capture techniques (e.g., for human activity recognition) and sophisticated physics engines (e.g., for robot manipulation). In general, increasing the realism of the synthesized output tends to improve DL performance [20,36–39], although, in some cases, it is less important [17,40–43]. However, achieving high levels of realism (e.g., as seen in movies) can be costly. One alternative approach is to generate synthetic data and then improve realism by using domain adaptation techniques, as discussed in the next section. Another alternative is to use generative adversarial models.

Generative adversarial networks (GANs) are a promising technique to synthesize novel images that match the statistical properties of the training data (Figure 1B) [44] (for a recent survey see [45]); for instance, GANs can generate faces of people that do not exist from a training set of existing human faces [46]. GANs comprise two models trained to optimize opposite objectives (i.e., adversarial): a generator and a discriminator (Figure 1E). The generator learns a lower-dimension latent representation of the training data domain and can generate new samples by receiving as input a random vector in the latent space. The discriminator, in turn, learns to distinguish original images from synthesized images. By training the generator and discriminator simultaneously, the generator learns to synthesize better samples to fool the discriminator. GANs are becoming increasingly popular due to the high visual quality of synthesized imagery [46–49], particularly when compared with other generative approaches, such as variational autoencoders [45]. However, in its original formulation, it is hard to control the output produced by GANs, although this remains an area of active research. A promising trend involves conditioning GANs on additional input that characterizes the samples being fed in training (e.g., labels specifying the gender of human faces) [50]. This idea has been extended to allow sophisticated control in the generation of images [46,51] (e.g., pose and hairstyle of human faces). One challenge with using GANs is that the synthesized imagery is not produced with the associated ground truth data, as is done for graphics pipelines. However, good progress is being made extending GANs to produce imagery that already comes with detailed annotation, such as images of scenes with automatically generated scene segmentation ground truths [52]. Another recent trend has been to train big generative models (e.g., with billions of parameters and terabytes of data) [53], including language [54] and multimodal models [55], that can subsequently be reused to synthesize novel content and be integrated with other pipelines to solve domain-specific tasks.

A third approach for synthesizing data comprises creating imagery by fusing from multiple data sources (Figure 1C). This is often accomplished by superimposing virtual objects [15,56,57] or people [58,59] on real backgrounds while ensuring that the virtual entities fit consistently with the background (e.g., by aligning surfaces and lighting). Extending this approach to fusing real entities on real backgrounds brings the extra challenge of cropping the real entities from the original backgrounds. Although this could be done manually, GAN-based methods have shown promise in automatically finding the cropping region (i.e., the semantic mask) with minimal annotation (e.g., bounding boxes) [60,61]. By combining segmentation with domain adaptation techniques, it is further possible to replace in place one type of entity for another (e.g., a bicycle for a motorcycle) while preserving the rest of the image [62,63].

Finally, we see much promise in integrating the synthesis and learning pipelines. There is a long history of integrating simulators with the learning process in deep reinforcement learning, where it is often impractical or impossible to train in the real world [64,65]. Reinforcement learning agents learn an action policy (e.g., grasping objects or playing a game) by practicing (millions of times) in simulators [66,67]. The key distinction is the integration of the learning process with the simulator, rather than relying on a static data set of simulated data for training. This powerful idea can be extended to support more sophisticated forms of learning, such as continuous (lifelong) learning and embodied (interactive) learning, which we discuss further below. The concept can be applied to supervised learning by using error signals, such as a task classification loss, to optimize data synthesis generation [68] (Figure 1F). When using graphics-rendering pipelines, one challenge is to propagate the error signal through the nondifferentiable functions implemented in traditional pipelines. An emerging field, called **neural rendering**, aims to build differentiable rendering

Glossary

Computer graphics-rendering pipeline: sequence of algorithms that produces a 3D visualization in screen space from parameters that describe the scene, such as object 3D and material information, lighting and camera properties, and rendering parameters.

Continual learning: process of learning a sequence of tasks without forgetting about how to perform earlier tasks in the sequence.

Domain adaptation: techniques that seek to align the statistical properties across domains (e.g., synthetic and real), so that DL models training in one domain can be deployed in another.

Domain randomization: techniques to create a data set that is diverse and broadly representative of a target domain, for the purpose of increasing the robustness and generalizability of a DL model.

Domain shift: change in the domain distribution that occurs when a DL model is trained with data from one domain (e.g., synthetic) and tested on another (e.g., real).

Embodied learning: process of learning from multimodal information obtained through interactive exploration of the environment.

Hybrid models: DL models trained with a mix of real and synthetic data.

Inverse rendering: process of automatically retrieving, from 2D imagery, scene attributes, such as 3D object information, lighting properties, and camera parameters.

Latent space disentanglement: technique that seeks to learn a lower dimension representation (e.g., letter category, rotation, and color) of a high-dimension space (e.g., images of letters) to support classification and generation in the lower dimension space.

Multimodal learning: process of learning knowledge from time-locked synchronized information from multiple sensors, such as audio, visual, and haptic input.

Neural rendering: controlled rendering of 3D realistic imagery using DL models. In contrast to traditional rendering pipelines, neural rendering pipelines are differentiable and can acquire 3D and physics knowledge from 2D training data.

Simulators: software that is able to generate, often in real time, data and ground-truth annotation from metadata for DL models, such as 3D imagery of a scene and segmentation masks.

pipelines and is showing fast progress in generating controllable visually realistic rendering [69,70] (reviewed in [71]). Thus, the integration of DL and differentiable rendering pipelines has the potential to support the generation of customized curricula for more sample-efficient learning.

Synthetic data: data used to train and test DL models that are created by artificial means, such as by rendering pipelines, GANs, and fusion models.

Closing the gap between synthetic and real data

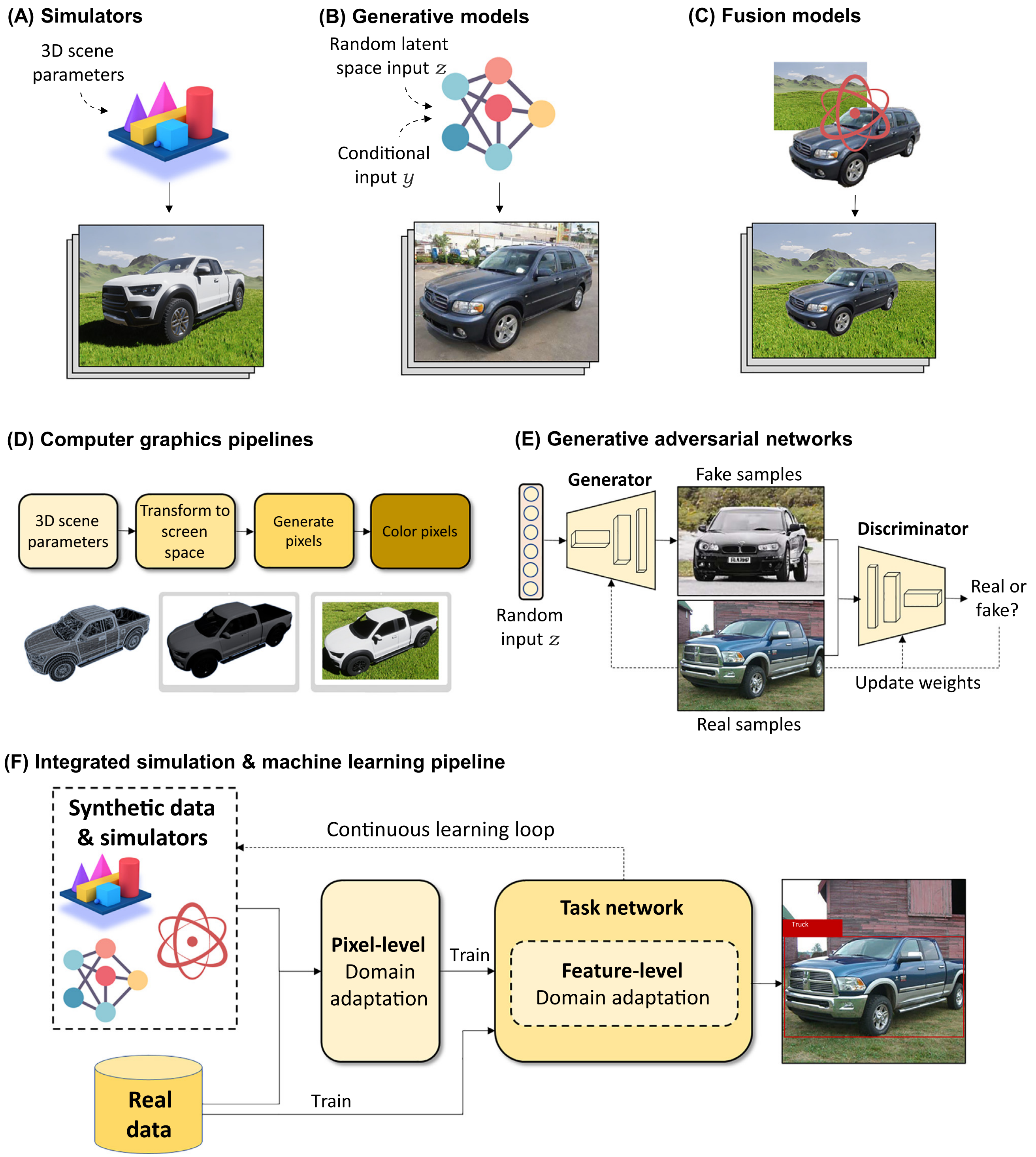
Despite their success in achieving state-of-the-art performance in several visual recognition tasks, neural networks suffer from **domain shift** (i.e., the performance of neural networks drops significantly when the test distribution is different from the training distribution, such as when training on synthetic data and testing on real data). To close this gap, several techniques have been developed to enhance the value of synthetic data. **Domain randomization** comprises varying the parameters used to generate the synthetic data so that the data set broadly captures the distribution in the target domain [16,28]. By training on such a diverse data set, the hope is that the model will be more robust to variation in the target domain and generalize better to novel samples. In some cases, this idea was even pushed to create nonphotorealistic versions of the data (e.g., vehicles with random textures) to encourage the model to learn better representations of the target concepts (e.g., features that capture the shape, rather than texture, of vehicles) [17]. Mixing real and synthetic data (**hybrid models**) has also often led, in practice, to a boost in performance compared with training only on one type [34]. The idea is that mixing data allows different data types to strengthen training, where others may have weaknesses (e.g., synthetic data tend to be more diverse, but real data may capture low-level details better).

An increasingly prominent technique is **domain adaptation**, which involves aligning the synthetic data pixel and feature distribution to the real data (Figure 2). Pixel-level adaptation comprises transferring the style, or visual appearance properties, of the target to the source domain. Approaches based on adversarial generative models are showing increasing success in creating realistic versions of the synthetic data, even without the need for any supervision (i.e., no labels are necessary) [32,72–75]. Recent promising techniques preserve semantic consistency when translating from source to target through cycle consistency (i.e., the translation needs to learn to go from source to target and back) [73], patch consistency (i.e., image patches in the source and target domain should reflect the same content) [74], and leveraging intermediate representations from an integrated computer graphics pipeline (e.g., depth and color masks) [75].

Whereas the goal in pixel alignment is to adjust the visual style of the source domain, in feature-level adaptation the distributional distance between source and target feature spaces is minimized, while simultaneously training a task network (e.g., segmentation model). Visual realism, in this case, is not the main concern, because the focus is on optimization for task performance. This problem is often presented as unsupervised domain adaptation, with labeled synthetic source data being available, but without labels for the target real data. Several feature alignment approaches have been explored, including through minimization of some distance between source and target distributions [76,77], weight sharing, and discriminators to encourage the network to learn domain invariant representations [78,79], projecting the distance minimization problem to pixel space to increase the network capacity and preserve semantic content [80], adapting while accounting for cross-domain label imbalances [81], and learning disentangled internal representations that abstract away irrelevant transformations in the target domain [82]. Often, the best results have been achieved by combining pixel and feature alignment approaches [83].

Enabling the next generation of deep learning

Drawing from cognitive psychology and neuroscience [14], there are several desirable functional and architectural requirements for DNNs. Approaching human-level intelligence likely requires grasping key concepts related to the physical world and its composition [3,14,84], as well as



Trends in Cognitive Sciences

(See figure legend at the bottom of the next page.)

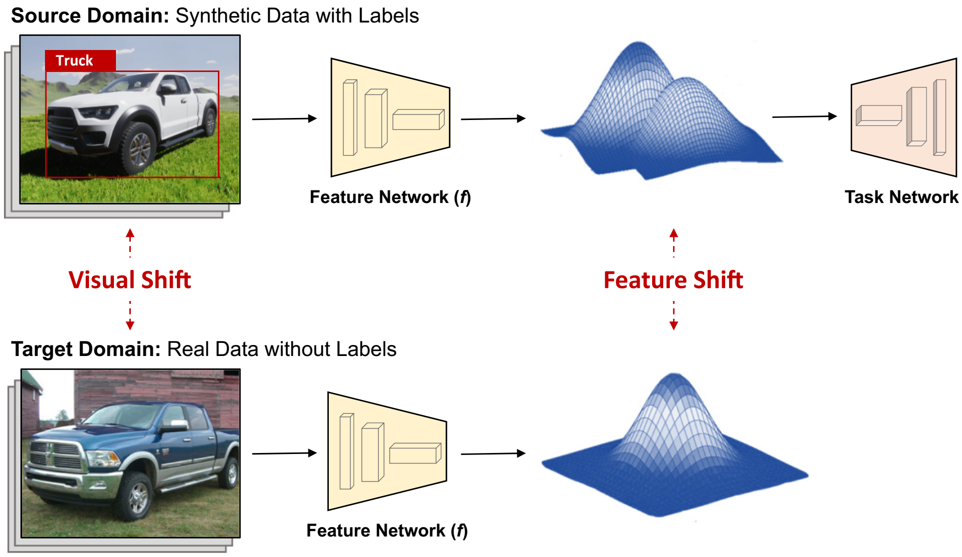


Figure 2. Domain adaptation at the pixel- and feature-level seeks to close the gap due to visual style and feature distribution shift when moving from synthetic to real data. In unsupervised domain adaptation, the source domain (synthetic data) is labeled (top row), whereas the target domain (real data) is unlabeled (bottom row). The goal is to close the domain gap by aligning the pixel style of the source to the target domain (i.e., close the visual shift) and learn an embedded representation that is invariant to the domains, while optimizing for a certain downstream task (i.e., close the feature shift).

the ability to learn continually, interactively, and multimodally [9,85]. Here, we emphasize the central role of synthetic data and simulators in enabling this next generation of DL and, complementarily, in providing insight into biological systems (Figure 3).

Deep learning for scientific exploration

There is a long history of drawing from artificial intelligence to further theory in cognitive psychology and neuroscience [10,14]. However, DNNs are gaining increasing attention as models of cognitive and neural function due to their ability to learn complex behavior from low-level sensory input, such as image pixels [11]. Some recent successes include predicting behavior and neural activity in perception [86,87] and memory [88] systems. The benchmarking of different DNNs with respect to how well they predict brain activity allows scientists to test current theory and formulate novel hypotheses about cognitive and neural functioning in biological systems [12]. Given that simulators can systematically recreate environmental conditions to test different learning processes and dynamics in DNNs, they are a key enabling technology to the study of biological systems (Figure 3A). Progress in techniques to ‘open up’ DNNs and gather insight into the representations embedded in the hidden layers [89] also introduces the opportunity to retrieve novel post hoc explanations for the functions modeled in DNNs. Here too, synthetic data can be useful to generate stimuli that target portions of the neural networks to study its function

Figure 1. Data synthesis approaches and integration with the machine learning pipeline. Data can be synthesized using (A) computer graphics rendering pipelines, (B) generative adversarial models, and (C) fusion models. (D) The traditional computer graphics pipeline receives as input 3D information about the scene and renders, through stages, a visualization of the scene in screen space. (E) Generative adversarial networks rely on a generator and discriminator network learning simultaneously on competing objectives, which leads the former to improve the quality of the synthesized imagery. (F) Integrating the synthesis and deep learning pipelines enables more sophisticated learning, such as embodied continuous learning.

CellPress

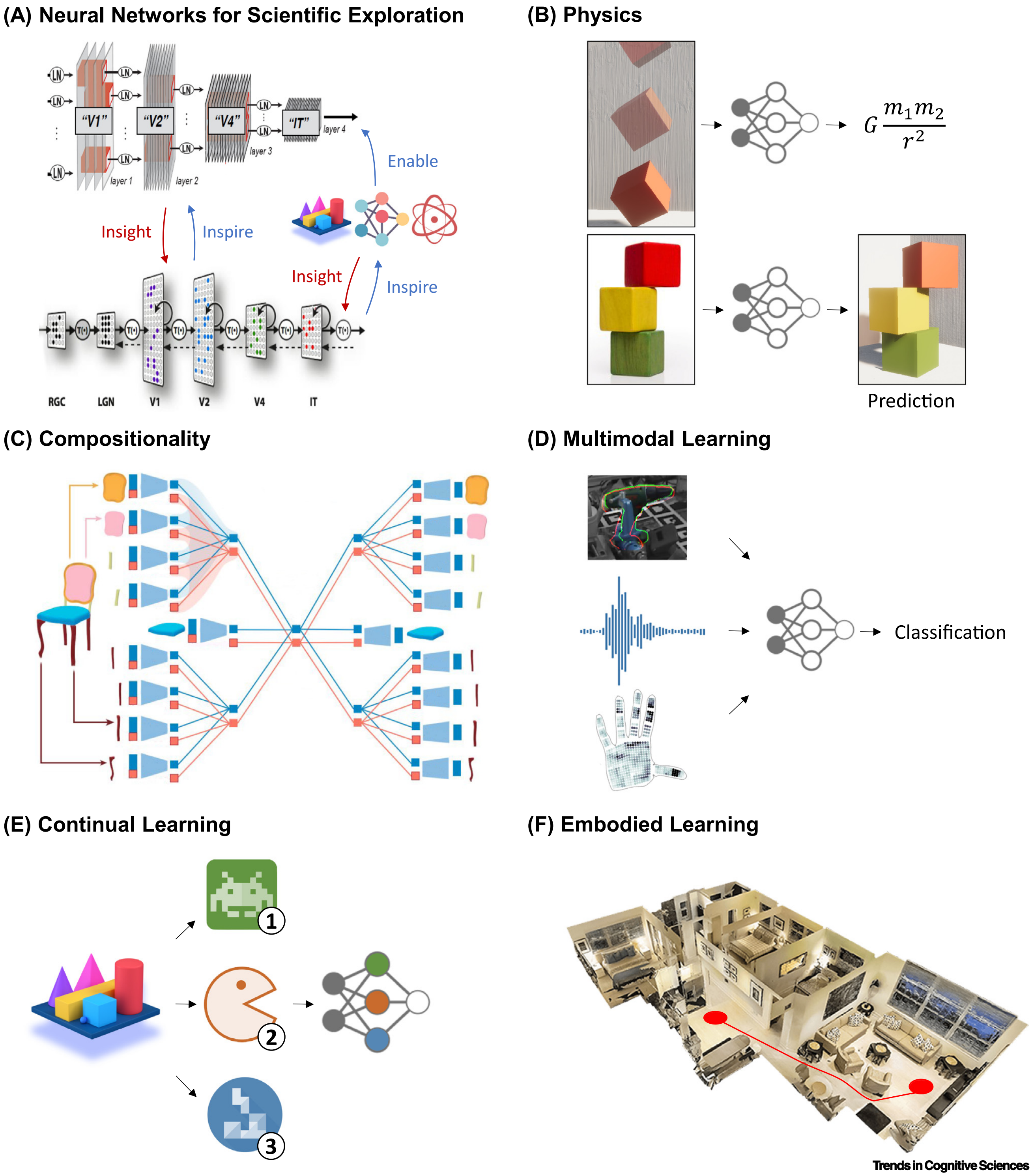


Figure 3. Simulators and synthetic data enable key capabilities for the next generation of machine learning models. (A) Simulated data and deep neural networks are important tools in the study of cognitive and neural function in biological systems, not only enabling exploration of novel (artificial) models for biological function, but also supporting the creation of optimized stimuli to test neural systems. (B) Simulated data can be used to learn (known and unknown) physics and make

(Figure legend continued at the bottom of the next page.)

and formulate explanations. For instance, synthetic data have been used to generate optimal stimuli to activate neural subpopulations in primate cortical regions [13]. Synthetic data and simulators also support sophisticated comparisons of artificial and biological systems. The use of synthetic materials to study biological systems has a long tradition in cognitive psychology, neuroscience, and artificial intelligence [3,90]. However, the increasing sophistication and realism of current approaches, as reviewed above, afford novel possibilities. For instance, fully immersive environments (e.g., virtual reality) could support direct comparisons of behavior and neural activation in DNNs versus humans versus nonhuman primates in embodied interactive tasks. Here, we review how synthetic data and simulators can help gain insight and model key cognitive functions.

Physics

At a very young age, humans have a basic understanding of the physical world, such as notions of what constitutes an object and expectations about how they interact with the environment [3,84]. This knowledge enables mental models about the composition of the world and predictions about what may happen next [14]. Thus, endowing DL models with this type of knowledge would support more sophisticated learning and inference. Recent work attempts to teach neural networks about physics on the fly (Figure 3B, top). One approach is to train the network with representative examples of the physical domain (e.g., collapsing block towers) and to rely on standard learning algorithms to implicitly learn physical knowledge [91]. Another involves developing specialized architectures that learn physical laws governing the domain (e.g., ball trajectories) [92]. In either case, simulators with an appropriate physics engine are often used to generate the training data [91,92]. It has been argued that human intuitive physics relies on a mental simulator to make predictions about the world [4]. In this paradigm, a physics simulator can be explicitly used to make relevant predictions and the challenge is then transformed into perceiving the environment (e.g., constructing a scene graph representation through **inverse rendering** [93]) and feeding that information to the simulator (Figure 3A, bottom). Finally, researchers have also begun embedding physical priors into the learning process (e.g., loss objectives that reflect pertinent physical constraints) to improve transfer from synthetic to real domains [94] and to synthesize more realistic 3D models from 2D imagery [71].

Compositionality

One way to address the complexity of modeling complex synthetic environments is through compositionality. Scenes in the world are decomposable into stable entities, animate (e.g., people or animals) or inanimate (e.g., vehicles or furniture), that we can generically call objects. Such decomposition is useful because objects recur in scenes in many different arrangements, but maintain their appearance, properties, and functionality. Similarly, objects themselves comprise parts that have a simpler structure and are often shared across related semantic categories (e.g., chairs, tables, and beds can all have legs). There is a long history in computer vision [95] and computer graphics (e.g., scene graphs) demonstrating the utility of exploiting compositionality in both analysis and synthesis tasks, both static and dynamic. Compositional representations are typically hierarchical groupings, based on many possible criteria, including spatial proximity, symmetry, causality, functionality, and others [96,97], related to principles studied in Gestalt psychology. In synthesis settings, such hierarchies provide natural scaffolds for editing operations, allowing convenient manipulation respecting the semantics of the object or scene, and facilitating the generation of multiple variations.

physically realistic predictions. (C) They can also be used to teach latent disentangled representations that capture the structure and shape of objects. (D) Simulated audio, visual, and haptic data provide redundancy and complementary that lead to more generalizable internal representations. (E) Simulators are ideal for generating a sequence of tasks, which are neither too narrow nor too disjoint to support continual learning without forgetting prior tasks. (F) Open-ended interactive exploration in simulators can enable the kind of embodied learning seen in biological systems.

The machine learning era has created the need for annotated compositional data. In the 3D object domain, data sets that provide fine-grained part decompositions have begun to emerge, in which objects are mapped into manually curated hierarchies [98]. Hierarchical neural nets, as well as hierarchical convolutional graph networks, have been used in the synthetic generation of objects and scenes, incorporating joint structure and geometry synthesis [99–101]. Scenes naturally exhibit more compositional variability than do objects, because many of their constituent entities are mobile or movable. In that setting, probabilistic formulations make sense, supporting rule statistics for a generative probabilistic scene grammar to be learned from data [69] and grammar productions themselves to be inferred [102]. Many variations are possible and a recent survey of generative 3D models for objects and scenes is available [103]. As with all generative approaches, defining appropriate losses for assessing the quality of the generated compositions remains a challenge. Generative compositional models can be conditioned on partial scans, images, or even language. Ideally, one looks for low-dimensional parametrizations of compositional variability with disentangled parameters. Composition often reflects function, and structured models can be useful in simulators, with either real or qualitative physics. It has also been suggested that compositionality will be a key attribute in building machines that think more like humans [3].

Multimodal learning

Humans experience and learn about the world through multiple senses, including vision, hearing, and touch [9]. The ability of multiple sensory neural structures to participate in the same function [104] enables redundancy and self-supervision in learning. Redundancy pertains to the ability to learn to perform a task using different modalities (e.g., vision or touch to grasp an object). Self-supervision pertains to the ability of different sensory systems to educate each other about performing a task (e.g., visual-haptic feedback to reach for an object inside a transparent container). In DL systems, it is also possible to use redundant and complementary information from multiple data sources to learn more robust and generalizable concept representations [105]. This is perhaps best exemplified by models that integrate audio and visual information, which often co-occur in nature, and learn correspondences that enable predictions on visual tasks from audio information [7] and vice versa [106]. Recently, haptic information was further shown to be useful for learning features that are pertinent to visual recognition tasks [107]. Given its relevance to building robust robotics and autonomous systems, there has also been considerable interest in merging RGB camera information with complementary sensors, such as depth, light detection and ranging (LiDAR), and infrared [108]. However, multimodal sensor data often require alignment or registration, both nontrivial tasks. Synthetic data generation can mostly alleviate the need for data alignment because data generation is under our control. This motivated the simulation of various sensor modalities, often by enhancing rendering pipelines with specialized physics engines [109,110]. A recent promising trend is to develop open-ended simulators that support multimodal training (e.g., physically realistic audiovisual data), as well as explorative incremental learning [111] (more on this in the 'Embodied learning' subsection).

Continual learning

Humans and animals are remarkably apt at adapting to a changing environment and learning continuously [8,112]. Replicating this capability in DL models would support learning of a potentially infinite series of tasks (e.g., detection of a growing number of categories). Therefore, researchers have explored several mechanisms to support this type of learning, often taking inspiration from biological systems. One approach prevents older tasks from being forgotten by protecting the weights relevant to those tasks [113], similarly to synaptic plasticity mechanisms in biological brains. Another approach integrates memory systems to support replay and episodic memory of relevant prior information [114]. Yet another approach replicates modularity in the brain, often achieved through interactive expansion of the network parameters while simultaneously trying to meet sparsity constraints [115]. A common challenge to these methods is the need to have a representative set of tasks, which is neither too narrow nor disjoint, to train continual learning algorithms. Due to the difficulty of collecting these training sets from the real world, researchers have often resorted to simulated tasks, such as the Atari game suite [66] and robot manipulation tasks [116], to train these algorithms. A natural extension is the development of simulated open-ended environments [85,111] that would not only enable lifelong, but also structured [117], continual learning. A practical consideration in this setting is designing computationally efficient data generation given the extended training timelines [118].

Embodied learning

Exploration is essential for human learning. Babies acquire foundational knowledge about the compositionality and the affordances of the physical world through free play with objects in their environments [3,84]. This interactive engagement leads to rich time-locked correlated visual, haptic, and auditory feedback that contributes to the formation of general internal representations of concepts. The idea that aspects of human intelligence are grounded and emerge from embodied interaction with the world has been associated with not only learning basic concepts (e.g., intuitive physics [91]) but also sophisticated symbolic systems (e.g., language [119]). Consequently, researchers noted that, in contrast to training from static data sets that capture moments in time, interactive explorative learning could lead DL systems to acquire more robust and generalizable representations of objects, actions, and functions [120,121]. This paradigm shift calls for data sets that, rather than capturing the world from a third-person perspective, represent first-person experiences. Whereas data sets have started emerging to support embodied learning [122,123], collecting this type of data is particularly labor intensive [124]. Accordingly, researchers have started developing open-ended physically realistic simulated environments that provide multimodal feedback [85,111].

The notion of embodied learning implies, at a fundamental level, knowledge about the 3D properties of the world. For instance, to understand how to interact with a novel object, it is necessary to understand its 3D affordances [120]. Whereas this information is readily available in simulators, there is also research in inverse rendering that tries to retrieve this information directly from 2D imagery [93,125,126]. However, reconstructing 3D shapes requires training data with multiple views of the target object or scene, which are seldom available in practice. Nevertheless, in promising recent work, researchers attempted to automatically retrieve, or disentangle, implicit 3D information from the latent space in GANs [127]. Therefore, simulators, inverse rendering, and **latent space disentanglement** techniques establish a comprehensive foundation to enabling embodied learning in DL models.

Concluding remarks

The next generation of DNNs will be able to learn rich models of the world in a continual, multimodal, and embodied fashion, matching cognitive capabilities seen only in biological systems. Simulators and synthetic data will have a central role in this transformation. The current generation of DL models is limited by access to high-quality training data. This challenge will only be exacerbated due to increased scrutiny of data privacy and security practices. Current research shows that synthetic data can be successfully leveraged to train DL models, especially when used in conjunction with domain adaptation techniques that align, statistically at the pixel and feature levels, synthetic and real data. This trend is bound to become more pervasive because it is becoming easier to synthesize realistic data due to impressive advancements in computer graphics rendering pipelines, generative adversarial models, and fusion models. However, beyond meeting current demands for data, synthetic data will meet novel demands. Open-ended interactive multimodal simulation will shift the training paradigm from static data sets usually from a third-

Outstanding questions

How do we generate realistic complex scenes using generative adversarial models? How do we control and condition generation? How do we automatically generate labeled data?

How do we build fully differentiable graphics rendering and animation pipelines, and through these enable structured learning? How can synthetic data generation be integrated into machine learning training pipelines to provide content finely tailored to the learner's needs?

How can synthetic data and simulators enable continual, multimodal, interactive, embodied learning as seen in human and nonhuman primate systems? How can simulators teach DNNs about the physical world and its composition?

What are the relative strengths and weaknesses of graphics-rendering pipelines, generative adversarial models, and fusion models? Whereas much progress has been made in each of these synthesis methods, systematic comparisons of the methods in terms of machine learning performance are still mostly missing.

How do we close the domain gap, at the pixel and feature levels, between synthetic and real data? How can we learn disentangled domain-independent internal representations? What is the relative importance of image-space alignment versus feature-space alignment between synthetic and real data? Are there fundamental limitations with synthetic data?

How do we know that a model is learning naturally (i.e., like a biological system)? How do we optimize data synthesis to match natural distributions, both in terms of content and order? How can simulators be used to benchmark the performance of DNNs compared with humans and nonhuman primates?

Will people trust DNNs that were trained with synthetic data? How can we build trust in them? Are there any fundamental differences between synthetic and real data? How do we interpret and explain what a synthetic DNN learnt and how it makes decisions?

person perspective to first-person embodied experience data sets, which are difficult to collect in the real world. Integration of the synthesis and learning pipelines will support continuous life-long structured learning more similarly to how humans learn and, thus, are likely to produce richer, robust, and generalizable knowledge about the world. Therefore, the paradox of using synthetic data to model natural forms of learning may, in practice, be no paradox at all.

Nevertheless, several open issues remain with respect to synthesizing data that are optimal for DL models (see Outstanding questions). From a modeling perspective, it is essential to assess how similar the learning and decision process in DNNs are compared with biological systems. Performance on existent data sets may provide insight into the predictive ability of the model, but the explanation for the prediction can be obscure. Progress in techniques to dissect and visualize the internal representations of DNNs [89] will likely have an essential role in retrieving these explanations. Furthermore, synthetic data can be systematically created (e.g., with increasing levels of complexity) precisely to study how internal representations are built. Synthetic data are also ideal for exploration [10], not only allowing the creation of stimuli to study brain and behavior [86,87], but also to create stimuli (e.g., virtual environments) for sophisticated interactive comparison of behavioral outcome and neural activation of artificial versus biological systems. From a practical perspective, before deploying DNNs in the real world, one needs to provide assurance that systems built using synthetic data will perform close to systems that were built using data collected by real sensors. Such assurances will require theoretically sound metrics for synthetic data quality that go beyond subjective impressions (e.g., 'looks good') and performance on benchmark data sets. Considerable investment has been made developing simulators for mainstream domains (e.g., driving), yet another practical difficulty is that there are still no sophisticated simulators for other, perhaps more complex, domains (e.g., social interaction). Nevertheless, good progress is being made in furthering these types of simulation (e.g., cognitive models of emotion and social expression [128]), as well as using simulated environments to facilitate the collection of data for these domains (e.g., virtual environments to study social interaction [90]). However, a more fundamental challenge may be whether people will trust and adopt systems trained exclusively, or mostly, with synthetic data (e.g., would people trust a self-driving car that was trained on simulators?). Therefore, it is important to understand the differences, in terms of not only performance, but also representation in feature space between models trained with synthetic versus real data. Here too, visualization and dissection techniques [89] will likely have a crucial role in explaining how synthetic networks work and help build trust. Nevertheless, despite these challenges, synthetic data introduce a unique opportunity that is worth exploring to enable a new generation of DL models that are not limited by available data, but by our imagination alone.

Acknowledgments

This research was supported by the US Army. The content does not necessarily reflect the position or the policy of any government, and no official endorsement should be inferred. We would like to thank Raghuveer Rao, Achuta Kadambi, and Judy Hoffman for their insightful comments during the preparation of this manuscript.

Declaration of interests

None declared by authors.

References

1. LeCun, Y. *et al.* (2015) Deep learning. *Nature* 521, 436–444
2. Rao, R. *et al.* (2021) *Synthetic environments for artificial intelligence (AI) and machine learning (ML) in multi-domain operations*. DEVCOM Army Research Laboratory May, ARL-TR-9198
3. Lake, B. *et al.* (2016) Building machines that learn and think like people. *Behav. Brain Sci.* 40, E253
4. Battaglia, P. *et al.* (2013) Simulation as an engine of physical scene understanding. *Proc. Natl. Acad. Sci. U.S.A.* 5, 18327–18332
5. Xu, F. and Tenenbaum, J. (2007) Word learning as Bayesian inference. *Psychol. Rev.* 114, 245–272
6. Barlow, H. (1989) Unsupervised learning. *Neural Comput.* 1, 295–311

7. Owens, A. *et al.* (2016) Ambient sound provides supervision for visual learning. *Lect. Notes Comput. Sci* 9905, 801–816
8. Hadsell, R. *et al.* (2020) Embracing change: continual learning in deep neural networks. *Trends Cogn. Sci.* 24, 1028–1040
9. Smith, L. and Gasser, M. (2005) The development of embodied cognitions: six lessons from babies. *Artif. Life* 11, 13–29
10. Cichy, M. and Kaiser, D. (2019) Deep neural networks as scientific models. *Trends Cogn. Sci.* 25, 305–317
11. Saxe, A. *et al.* (2021) If deep learning is the answer, what is the question? *Nat. Rev. Neurosci.* 22, 55–67
12. Schrimpf, M. *et al.* (2020) Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron* 108, 413–423
13. Bashivan, P. *et al.* (2019) Neural population control via deep image synthesis. *Science* 364, eaav9436
14. Hassabis, D. *et al.* (2017) Neuroscience-inspired artificial intelligence. *Neuron* 95, 245–258
15. Alhaija, H. *et al.* (2018) Augmented reality meets computer vision: efficient data generation for urban driving scenes. *Int. J. Comput. Vis.* 126, 961–972
16. Prakash, A. *et al.* (2019) Structured domain randomization: bridging the reality gap by context-aware synthetic data. In *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*, pp. 7249–7255, IEEE
17. Tremblay, J. *et al.* (2018) Training deep networks with synthetic data: bridging the reality gap by domain randomization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1082–10828, IEEE
18. Gaidon, A. *et al.* (2016) Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4340–4349, IEEE
19. Muller, M. *et al.* (2018) Sim4cv: a photo-realistic simulator for computer vision applications? *Int. J. Comput. Vis.* 126, 902–919
20. Movshovitz-Attias, Y. *et al.* (2016) How useful is photorealistic rendering for visual learning? *arXiv* 2016. arXiv:1603.08152v2
21. Ros, G. *et al.* (2016) The SYNTHIA dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–3243, IEEE
22. Richter, S. *et al.* (2016) Playing for data: ground truth from computer games. *Lect. Notes Comput. Sci* 9905, 102–118
23. Shafaei, A. *et al.* (2016) Play and Learn: Using video games to train computer vision models. In *Proceedings of the British Machine Vision Conference (BMVC)* (Wilson, R. *et al*., eds), pp. 26.1–26.13, BMVA Press
24. Sizikoval, E. *et al.* (2016) Enhancing place recognition using joint intensity - depth analysis and synthetic data. *Lect. Notes Comput. Sci* 9905, 901–908
25. Wijmans, E. *et al.* (2019) Embodied question answering in photorealistic environments with point cloud perception. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6652–6661, IEEE
26. Bousmalis, K. *et al.* (2018) Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *Proceedings of 2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4243–4250, IEEE
27. Saxena, A. *et al.* (2008) Robotic grasping of novel objects using vision. *Int. J. Robot. Res.* 27, 157–173
28. Tobin, R. *et al.* (2017) Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, IEEE
29. Hattori, H. *et al.* (2018) Augmented reality meets computer vision: efficient data generation for urban driving scenes. *Int. J. Comput. Vis.* 126, 1027–1044
30. Ionescu, C. *et al.* (2014) Human3.6m: large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.* 36, 1325–1339
31. Shotton, J. *et al.* (2013) Efficient human pose estimation from single depth images. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 2821–2840
32. Shrivastava, A. *et al.* (2017) Learning from simulated and unsupervised images through adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2242–2251, IEEE
33. Dosovitskiy, A. *et al.* (2015) Flownet: Learning optical flow with convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2758–2766, IEEE
34. de Melo, C. *et al.* (2020) Vision-based gesture recognition in human-robot teams using synthetic data. In *Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10278–10284, IEEE
35. Nikolenko, S. (2019) Synthetic data for deep learning. *arXiv* 2019. arXiv:1909.11512
36. Planche, B. *et al.* (2017) Depthsynth: Real-time realistic synthetic data generation from CAD models for 2.5d recognition. In *Proceedings of 2017 International Conference on 3D Vision (3DV)*, pp. 1–10, IEEE
37. Tsirikoglou, A. *et al.* (2017) Procedural modeling and physically based rendering for synthetic data generation in automotive applications. *CoRR* 2017, abs/1710.06270
38. Wrenninge, M. and Unger, J. (2018) Synscapes: a photorealistic synthetic dataset for street scene parsing. *arXiv* 2018. arXiv:1810.08705
39. Zhang, Y. *et al.* (2016) Physically-based rendering for indoor scene understanding using convolutional neural networks. *arXiv* 2016. arXiv:1612.07429
40. Howard, A. *et al.* (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv* 2017. arXiv:1704.04861
41. Hu, G. *et al.* (2018) Frankenstein: learning deep face representations using small data. *IEEE Trans. Image Process.* 27, 293–303
42. Lopez, A. *et al.* (2017) From virtual to real world visual perception using domain adaptation-the DPM as example. In *Domain Adaptation in Computer Vision Applications* (Csurka, G., ed.), pp. 243–258, Springer International Publishing
43. Mayer, N. *et al.* (2018) What makes good synthetic training data for learning disparity and optical flow estimation? *Int. J. Comput. Vis.* 126, 942–960
44. Goodfellow, I. *et al.* (2014) Generative adversarial networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS)*, pp. 2672–2680, MIT Press
45. Liu, M.Y. *et al.* (2020) Generative adversarial networks for image and video synthesis: algorithms and applications. *arXiv* 2020. arXiv:2008.02793
46. Karras, T. *et al.* (2019) A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4396–4405, IEEE
47. Miyato, T. *et al.* (2018) Spectral normalization for generative adversarial networks. *arXiv* 2018. arXiv:1802.05957
48. Brock, A. *et al.* (2019) Large scale GAN training for high fidelity natural image synthesis. *arXiv* 2019. arXiv:1809.11096
49. Azadi, S. *et al.* (2019) Semantic bottleneck scene generation. *arXiv* 2019. arXiv:1911.11357
50. Mirza, M. and Osindero, S. (2014) Conditional generative adversarial nets. *arXiv* 2014. arXiv:1411.1784
51. Niemeyer, M. and Geiger, A. (2021) GIRAFFE: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11448–11459, IEEE
52. Zhang, Y. *et al.* (2021) DatasetGAN: efficient labeled data factory with minimal human effort. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11448–11459, IEEE
53. Bommasani, R. *et al.* (2021) On the opportunities and risks of foundational models. *arXiv* 2021. arXiv:2108.07258v2
54. Brown, T. *et al.* (2020) Language models are few-shot learners. *arXiv* 2020. arXiv:2005.14165v4
55. Radford, A. *et al.* (2021) Learning transferable visual models from natural language supervision. *arXiv* 2021. arXiv:2103.00020v1
56. Dwibedi, D. *et al.* (2017) Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1310–1319, IEEE

57. Georgakis, G. *et al.* (2017) Synthesizing training data for object detection in indoor scenes. In *Proceedings of the Robotics: Science and Systems (RSS)* (Amato, N. *et al.*, eds), pp. 043, RSS
58. Hattori, H. *et al.* (2015) Learning scene-specific pedestrian detectors without real data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3819–3827, IEEE
59. Fernández, C. *et al.* (2011) Augmenting video surveillance footage with virtual agents for incremental event evaluation. *Pattern Recogn. Lett.* 32, 878–889
60. Remez, T. *et al.* (2018) Learning to segment via cut-and-paste. *Lect. Notes Comput. Sci* 11219, 39–54
61. Ostyakov, P. *et al.* (2018) SEIGAN: towards compositional image generation by simultaneously learning to segment, enhance, and inpaint. *arXiv* 2018. arXiv:1811.07630
62. Liang, X. *et al.* (2018) Generative semantic manipulation with mask-contrasting GAN. *Lect. Notes Comput. Sci* 11217, 574–590
63. Chen, X. *et al.* (2018) Attention-GAN for object transfiguration in wild images. *Lect. Notes Comput. Sci* 11206, 167–184
64. Mnih, V. *et al.* (2015) Human-level control through deep reinforcement learning. *Nature* 518, 529–533
65. Botvinick, M. *et al.* (2019) Reinforcement learning, fast and slow. *Trends Cogn. Sci.* 23, 408–422
66. Mnih, V. *et al.* (2013) Playing Atari with deep reinforcement learning. *arXiv* 2013. arXiv:1312.5602
67. Akkaya, I. *et al.* (2019) Solving Rubik's cube with a robot hand. *arXiv* 2019. arXiv:1910.07113
68. Kar, A. *et al.* (2019) Meta-sim: Learning to generate synthetic datasets. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 4550–4559, IEEE
69. Eslami, S. *et al.* (2018) Neural scene representation and rendering. *Science* 360, 1204–1210
70. Kato, H. *et al.* (2018) Neural 3d mesh renderer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3907–3916, IEEE
71. Tewari, A. *et al.* (2020) State of the art on neural rendering. *Comput. Graph. Forum* 39, 701–727
72. Bousmalis, K. *et al.* (2017) Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 95–104, IEEE
73. Zhu, J.-Y. *et al.* (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 2242–2251, IEEE
74. Park, T. *et al.* (2020) Contrastive learning for unpaired image-to-image translation. *Lect. Notes Comput. Sci* 12354, 319–345
75. Richter, S. *et al.* (2021) Enhancing photorealism enhancement. *arXiv* 2021. arXiv:2105.04619
76. Long, M. and Wang, J. (2015) Learning transferable features with deep adaptation networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 97–105, JMLR
77. Sun, B. and Saenko, K. (2016) Deep CORAL: correlation alignment for deep domain adaptation. In *Computer Vision – ECCV 2016 Workshops* (Hua, G. and Jégou, H., eds), pp. 443–450, Springer
78. Liu, M.Y. and Tuzel, O. (2016) Coupled generative adversarial networks. In *Proceedings of Advances in Neural Processing Systems (NeurIPS)*, pp. 469–477, Curran Associates Inc.
79. Tzeng, E. *et al.* (2017) Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2962–2971, IEEE
80. Sankaranarayanan, S. *et al.* (2018) Generate to adapt: aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8503–8512, IEEE
81. Prabhu, V. *et al*. SENTRY: Selective entropy optimization via committee consistency for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, IEEE
82. Litany, O. *et al*. Representation learning through latent canonicalizations. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE
83. Hoffman, J. *et al.* (2017) CyCADA: Cycle-consistent adversarial domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1989–1998, PMLR
84. Spelke, E. and Kinzler, E. (2005) Core knowledge. *Dev. Sci.* 10, 89–96
85. Savva, M. *et al.* (2019) Habitat: a platform for embodied AI research. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 9338–9346, IEEE
86. Khaligh-Razavi, S.-M. and Kriegeskorte, N. (2014) Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Comput. Biol.* 10, e1003915
87. Yamins, D.L.K. *et al.* (2014) Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proc. Natl. Acad. Sci. U.S.A.* 111, 8619–8624
88. Lloyd, K. *et al.* (2012) Learning to use working memory: a reinforcement learning gating model of rule acquisition in rats. *Front. Comput. Neurosci.* 6, 87
89. Bau, D. *et al.* (2020) Understanding the role of individual units in a deep network. *Proc. Natl. Acad. Sci. U.S.A.* 117, 30071–30078
90. Blascovich, J. *et al.* (2002) Immersive virtual environment technology as a methodological tool for social psychology. *Psychol. Inq.* 13, 103–124
91. Lerner *et al.* (2016) Learning physical intuition of block towers by example. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 430–438, JMLR
92. Chari, P. *et al.* (2019) Visual physics: Discovering physical laws from videos. *arXiv* 2019. arXiv:1911.11893
93. Ost, J. *et al.* (2021) Neural scene graphs for dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2855–2864, IEEE
94. Chen, Z. *et al.* (2021) PSD: Principled synthetic-to-real dehazing guided by physical priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7176–7185, IEEE
95. Felzenszwalb, P. and Huttenlocher, D. (2005) Pictorial structures for object recognition. *Int. J. Comput. Vis.* 61, 55–79
96. Lun, Z. *et al.* (2017) Learning to group discrete graphical patterns. *ACM Trans. Graph.* 36, 1–11
97. Wang, Y. *et al.* (2011) Symmetry hierarchy of man-made objects. *Comput. Graph. Forum* 30, 287–296
98. Mo, K. *et al.* (2019) PartNet: a large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 909–918, IEEE
99. Li, J. *et al.* (2017) GRASS: generative recursive autoencoders for shape structures. *ACM Trans. Graph.* 36, 1–14
100. Mo, K. *et al.* (2020) StructureNet: hierarchical graph networks for 3D shape generation. *ACM Trans. Graph.* 38, 1–19
101. Li, M. *et al.* (2019) *GRAINS:* Generative recursive autoencoders for INdoor scenes. *ACM Trans. Graph.* 38, 1–16
102. Devaranjan, J. *et al.* (2020) Meta-Sim2: unsupervised learning of scene structure for synthetic data generation. *Lect. Notes Comput. Sci* 12362, 715–733
103. Chaudhuri, S. *et al.* (2020) Learning generative models of 3D structures. *Comput. Graphics Forum* 39, 643–666
104. Edelman, G. and Gally, J. (2001) Degeneracy and complexity in biological systems. *Proc. Natl. Acad. Sci. U.S.A* 98, 13763–13768
105. Baltrušaitis, T. *et al.* (2017) Multimodal machine learning: a survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 423–443
106. Arandjelović, R. and Zisserman, A. (2017) Look, listen and learn. *arXiv* 2017. arXiv:1705.08168v2
107. Sundaram, S. *et al.* (2019) Learning the signatures of the human grasp using a scalable tactile glove. *Nature* 569, 698–702
108. Grigorescu, S. *et al.* (2020) A survey of deep learning techniques for autonomous driving. *J. Field Robot.* 37, 362–386
109. Wu, B. *et al.* (2019) SqueezeSegV2: improved model structure and unsupervised domain adaptation for road-object segmentation from a LiDAR point cloud. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pp. 4376–4382, IEEE
110. Ahmadibeni, A. *et al.* (2020) Automatic target recognition of aerial vehicles based on synthetic SAR imagery using hybrid stacked denoising auto-encoders. *Proc. SPIE* 11393, 113930J

111. Gan *et al.* (2020) ThreeDWorld: a platform for interactive multi-modal physical simulation. *arXiv* 2020. arXiv:2007.04954
112. Cichon, J. and Gan, W.-B. (2015) Branch-specific dendritic Ca^{2+} spikes cause persistent synaptic plasticity. *Nature* 520, 180–185
113. Kirkpatrick, J. *et al.* (2017) Overcoming catastrophic forgetting in neural networks. *Proc. Natl. Acad. Sci. U.S.A.* 114, 3521–3526
114. Aljundi, R. *et al.* (2019) Online continual learning with maximally interfered retrieval. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 11872–11883, ACM
115. Bellec, G. *et al.* (2017) Deep rewiring: training very sparse deep networks. *arXiv* 2017. arXiv:1711.05136v5
116. Yu, T. *et al.* (2021) Meta-World: a benchmark and evaluation for multi-task and meta learning. *arXiv* 2021. arXiv:1910.10897
117. Bengio, Y. *et al.* (2009) Curriculum learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 41–48, ACM
118. Mason, K. *et al.* (2019) An 'on the fly' framework for efficiently generating synthetic big data sets. In *Proceedings of the IEEE International Conference on Big Data*, pp. 3379–3387, IEEE
119. McNeill, D. (1992) *Hand and Mind: What Gestures Reveal about Thought*, The University of Chicago Press
120. Mandikal, P. and Grauman, K. (2021) Learning dexterous grasping with object-centric visual affordances. *arXiv* 2012. arXiv:2009.01439v2
121. Nagarajan, T. *et al.* (2020) Ego-Topo: environment affordances from egocentric video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 160–169, IEEE
122. Sigurdsson, G. *et al.* (2018) Charades-Ego: a large-scale dataset of paired third and first person videos. *arXiv* 2018. arXiv:1804.09626
123. Damen, D. *et al.* (2018) Scaling egocentric vision: the EPIC-KITCHENS dataset. *arXiv* 2018. arXiv:1804.02748v2
124. Grauman, K. *et al.* (2021) Ego4D: around the world in 3,000 hours of egocentric video. *arXiv* 2021. arXiv:2110.07058v1
125. Wang, N. *et al.* (2018) Pixel2mesh: generating 3D mesh models from single RGB images. *Lect. Notes Comput. Sci* 11215, 55–71
126. Mescheder, L. *et al.* (2019) Occupancy networks: learning 3D reconstruction in function space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4455–4465, IEEE
127. Zhang, Y. *et al.* (2021) Image GANs meet differentiable rendering for inverse graphics and interpretable neural rendering. *arXiv* 2021. arXiv:2010.09125v2
128. Marsella, S. *et al.* (2010) Computational models of emotion. In *A Blueprint for an Affectively Competent Agent: Cross-Fertilization between Emotion Psychology, Affective Neuroscience, and Affective Computing* (Scherer, K. *et al.*, eds), pp. 21–41, Oxford University Press