Taskonomy: Disentangling Task Transfer Learning

Amir Zamir[1,2] **Alexander Sax**[1*] **William Shen**[1*] **Leonidas Guibas**[1] **Jitendra Malik**[2] **Silvio Savarese**[1]

[1] **Stanford University** [2] **University of California, Berkeley**

http://taskonomy.vision/

Abstract

Do visual tasks have relationships, or are they unrelated? For instance, could having surface normals simplify estimating the depth of an image? Intuition answers these questions positively, implying existence of a certain structure among visual tasks. Knowing this structure has notable values; it provides a principled way for identifying relationships across tasks, for instance, in order to reuse supervision among tasks with redundancies or solve many tasks in one system without piling up the complexity.

We propose a fully computational approach for modeling the transfer learning structure of the space of visual tasks. This is done via finding transfer learning dependencies across tasks in a dictionary of twenty-six 2D, 2.5D, 3D, and semantic tasks. The product is a computational taxonomic map among tasks for transfer learning, and we exploit it to reduce the demand for labeled data. For example, we show that the total number of labeled datapoints needed for solving a set of 10 tasks can be reduced by roughly $\frac{2}{3}$ (compared to training independently) while keeping the performance nearly the same. We provide a set of tools for computing and visualizing this taxonomical structure at http://taskonomy.vision.

1 Introduction

Object recognition, depth estimation, edge detection, pose estimation, etc are examples of common vision tasks deemed useful and tackled by the research community. Some of them have rather clear relationships: we understand that surface normals and depth are related (one is a derivate of the other), or vanishing points in a room are useful for layout estimation and orientation. Other relationships are less clear: how edge detection and the shading in a room can, together, assist with pose estimation.

The field of computer vision has indeed gone far without explicitly using these relationships. We have made remarkable progress by developing advanced learning machinery

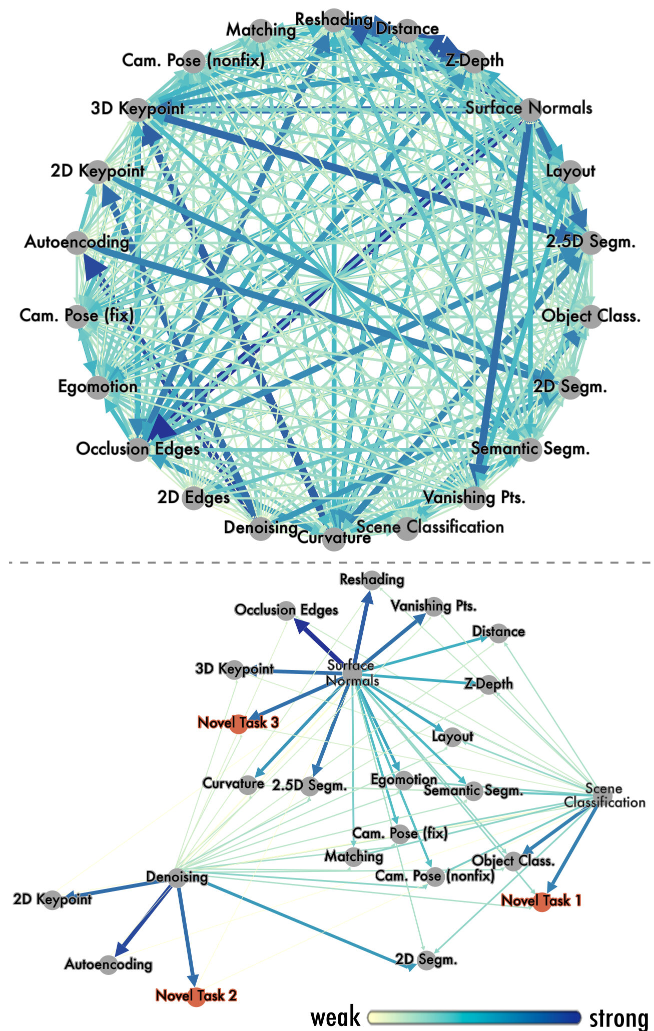

Figure 1: **A transfer learning Task Taxonomy (*Taskonomy*). Upper**: Computationally measured transfer learning relationships across visual tasks. The color-thickness of edges denote the strength of the relationship. **Lower**: A taxonomy extracted from the relationships to maximize the overall performance of solving many tasks while using minimum supervision by accordingly transferring information.

*Equal.

(e.g. ConvNets) capable of finding complex mappings from X to Y when many pairs of (x, y) s.t. $x \in X, y \in Y$ are given as training data. This is usually referred to as fully supervised learning and often leads to problems being solved in isolation. Siloing tasks makes training a new task or a comprehensive perception system a Sisyphean challenge, whereby each task needs to be learned individually from scratch. Doing so ignores their quantifiably useful relationships leading to a massive labeled data requirement.

Alternatively, a model aware of the relationships among tasks demands less supervision, uses less computation [Standley *et al.*, 2019], and behaves in more predictable ways. Incorporating such a structure is the first stepping stone towards developing provably efficient comprehensive perception models [Ge, 2013], i.e. ones that can solve a large set of tasks before becoming intractable in supervision or computation demands. However, this task space structure and its effects are still largely unknown. The relationships are non-trivial, and finding them is complicated by the fact that we have imperfect learning models and optimizers. In this paper, we attempt to shed light on this underlying structure and present a framework for mapping the space of visual tasks by way of transfer learning. Here what we mean by "structure" is a collection of computationally found relations specifying which tasks supply useful information to another, and by how much. This is depicted as graphs in Fig. 1.

We employ a fully computational approach for this purpose, with neural networks as the adopted computational function class. In a feedforward network, each layer successively forms more abstract representations of the input containing the information needed for mapping the input to the output. These representations, however, can transmit statistics useful for solving other outputs (tasks), presumably if the tasks are related in some way [Sharif Razavian *et al.*, 2014]. This is the basis of our approach: we compute an affinity matrix among tasks based on whether the solution for one task can be sufficiently easily read out of the representation trained for another task. Such transfers are sampled and evaluated, then a Binary Integer Program extracts a globally efficient transfer policy, represented as a subgraph, from them (Fig. 1). This model leads to solving tasks with far less data than learning them independently and the structure holds on other datasets (ImageNet and MIT Places [Russakovsky *et al.*, 2015; Zhou *et al.*, 2014]).

Being fully computational and representation-based, the proposed approach avoids imposing prior (possibly incorrect) assumptions on the task space. This is crucial because the priors about task relations are often derived from either human intuition or analytical knowledge, while neural networks need not operate on the same principles [Mccloskey and Cohen, 1989; Hoshen and Peleg, 2015]. For instance, although we might expect depth to transfer to surface normals better (derivatives are easy), the opposite is found to be the computationally better direction (i.e. suited neural networks better).

An interactive taxonomy solver, visualization of all transfer functions, a live demo, dataset, and code are available at http://taskonomy.vision/. For the full details of the methodology and experimental results overviewed in the rest of this paper, please refer to the full paper [Zamir *et al.*, 2018].

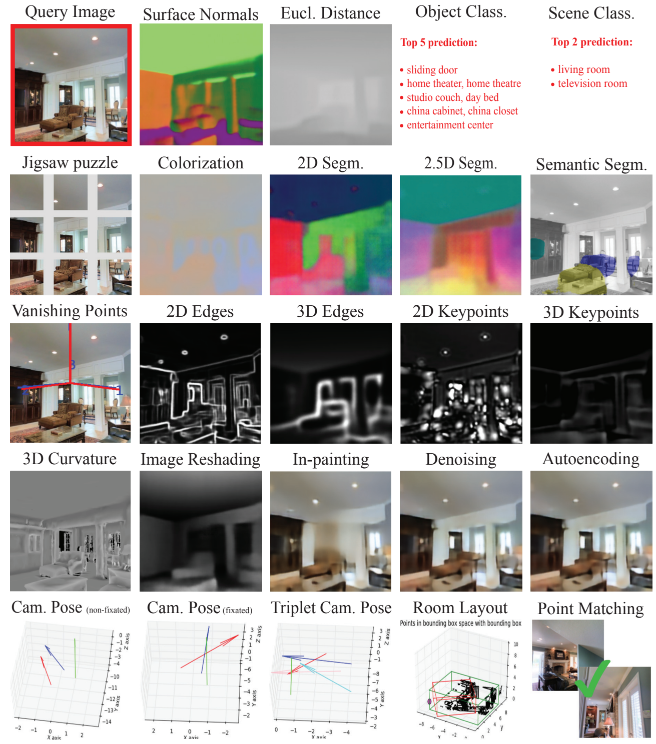

Figure 2: **Task Dictionary.** Outputs of 24 (of 26) task-specific networks for a query (top left). See results of applying frame-by-frame on a YouTube video here.

2 Method

A vision task is usually an abstraction read from raw images. We denote a task t with a function f_t which maps image I to $f_t(I)$, for instance $image \rightarrow depth$.

We define the problem as follows: we want to maximize the collective performance when solving a set of tasks $\mathcal{T} = \{t_1, ..., t_n\}$, subject to the constraint that we have a limited supervision budget γ (due to financial, computational, or time constraints). We define our supervision budget γ to be the maximum allowable number of tasks that we are willing to train from scratch (i.e. *source* tasks). The task dictionary is defined as $\mathcal{V}$=$\mathcal{T} \cup \mathcal{S}$ where $\mathcal{T}$ is the set of tasks that we want solved (*target* tasks), and $\mathcal{S}$ is the set of tasks that can be trained (*source* tasks). Therefore, $\mathcal{T} - \mathcal{T} \cap \mathcal{S}$ are the tasks that we want solved but cannot train ("target-only"), $\mathcal{T} \cap \mathcal{S}$ are the tasks that we want solved but could play as source too, and $\mathcal{S} - \mathcal{T} \cap \mathcal{S}$ are the "source-only" tasks which we are not directly interested in (e.g. jigsaw puzzle) but can be optionally used if they increase the performance on $\mathcal{T}$.

The **task** tax**onomy** (***taskonomy***) is a computationally found directed hypergraph that captures the notion of task transferability over a given dictionary. An edge between a set of source tasks and a target task represents a feasible transfer case and its weight represents its performance. We use these edges to estimate the globally optimal transfer policy to solve $\mathcal{T}$. Taxonomy produces a family of such graphs, parameterized by the available supervision budget, chosen tasks, transfer orders, and transfer functions' expressiveness.

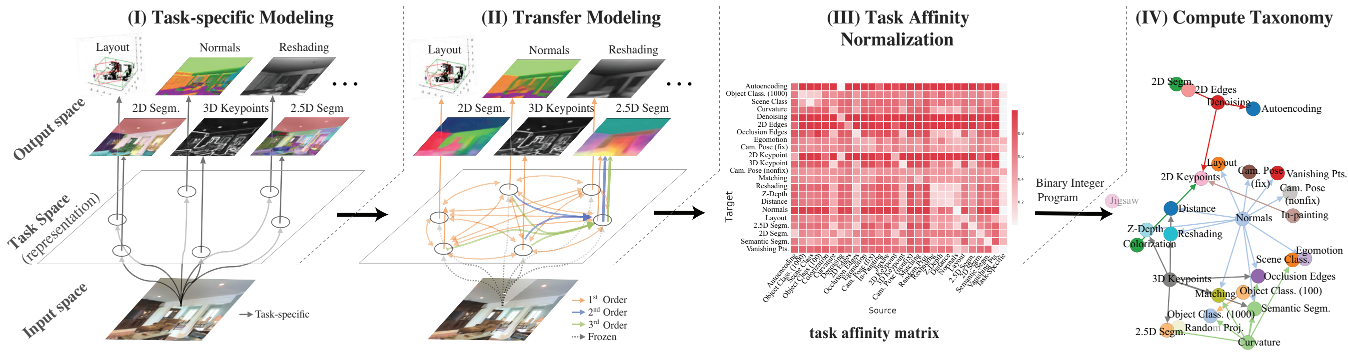

Figure 3: **Computational modeling of task relations and creating the taxonomy.** From left to right: **I.** Train task-specific networks. **II.**Train (first order and higher) transfer functions among tasks in a latent space. **III.** Normalize transfer affinities using AHP (Analytic Hierarchy Process). Here the first-order (directed) task affinity matrix after normalization is show (the lighter the color, the stronger the transfer). Fig. 1-upper shows the same values in a graph. **IV.** Find global transfer taxonomy using BIP (Binary Integer Program).

Taxonomy is built using a 4 step process depicted in Fig. 3.

Step I: Task-Specific Modeling: A task-specific network for each task in $\mathcal{S}$ is trained. The networks have an architecture homogeneous across all tasks

Step II: Transfer Modeling: Given a source task s and a target task t, where $s \in \mathcal{S}$ and $t \in \mathcal{T}$, a transfer network learns a small readout function for t given a statistic computed for s. The statistic is the image representation computed using the encoder of the task-specific network of s. Thus, the performance of this transfer network at predicting t is a useful metric for quantifying the (directed) task affinity $s \to t$. We train and evaluate all feasible transfers between sources and targets yielding a directed affinity matrix across tasks.

Note that for a transfer to be successful, the latent representation of the source should both be *inclusive* of sufficient information for solving the target but also have the information *accessible*, i.e. easily extractable (otherwise, the raw image would be the optimal representation itself). Thus, we adopt a low-capacity architecture as transfer function trained with a small amount of data, in order to measure transferability conditioned on being highly accessible.

Step III: Ordinal Normalization using Analytic Hierarchy Process (AHP): Different tasks are represented by different output spaces with vastly different units and numerical properties. Thus, the task affinities acquired from transfer function performances need to be normalized. We use an *ordinal* scheme for this purpose, derived from the Analytic Hierarchy Process [Saaty, 1987]. The motivation behind this choice and details are provided in Sec. "Ordinal Normalization using AHP" of the full paper. The post-normalization affinity matrix is shown in Fig. 3-III & Fig. 1-up graphically.

Step IV: Computing the Global Taxonomy: Given the normalized task affinity matrix, we need to devise a global transfer policy which maximizes collective performance across all tasks, while minimizing the used supervision. This problem can be formulated as a constraint satisfaction subgraph selection where tasks are nodes and transfers are edges. The optimal subgraph picks the best source nodes and the edges from these sources to targets that maximize the total performance across all targets tasks while ensuring that the number of source nodes does not exceed the allocated supervision budget. We solve this subgraph selection problem using Boolean Integer Programming (BIP), which can be solved optimally and efficiently [Gurobi Optimization, 2016]. The detailed formulation is available in Sec. "Computing the Global Taxonomy" of the full paper

Task Dictionary: Our mapping of task space is done via 26 sample tasks included in the dictionary, so we ensure they cover common themes in computer vision (2D, 3D, semantics, etc) with various levels of perceptual abstraction to elucidate fine-grained structures of task space. See Fig. 2 for some of the tasks with detailed definitions provided in the full paper. It is critical to note the task dictionary is meant to be a *sampled set, not an exhaustive list*, from a denser space of all conceivable visual tasks/abstractions. Sampling gives us a tractable way to sparsely model a dense space, and the hypothesis is that (subject to a proper sampling) the derived model should generalize to out-of-dictionary tasks. This is evaluated in Sec. "Generalization to Novel Tasks" of the full paper with supportive results.

3 Experimental Results

With 26 tasks in the dictionary (4 "source-only" tasks), our approach leads to training 26 fully supervised task-specific networks, 22×25 transfer networks in 1^{st} order, and $22 \times \binom{25}{k}$ for k^{th} order. The total number of transfer functions trained for the taxonomy after sampling was ~3,000 which took 47,886 GPU hours on the cloud. We preserved the architectural and training details across tasks as homogeneously as possible to avoid injecting any architectural bias. A live demo for user uploaded queries is available here.

Dataset: We created a dataset of 4 million images of indoor scenes from about 600 buildings; every image has an annotation for every task. Training all of our tasks on exactly the same pixels eliminates the possibility that the observed transferabilities are affected by different input data peculiarities rather than only task intrinsics. The images are registered on and aligned with building-wide meshes similar to [Armeni *et al.*, 2017] enabling us to programmatically compute the ground truth for many tasks without human labeling.

Evaluation of Computed Taxonomies: Fig. 4-left shows the computed taxonomies optimized to solve the full dictionary, i.e. all tasks are placed in $\mathcal{T}$ and $\mathcal{S}$ (except for 4 source-only tasks that are in $\mathcal{S}$ only). This was done for various

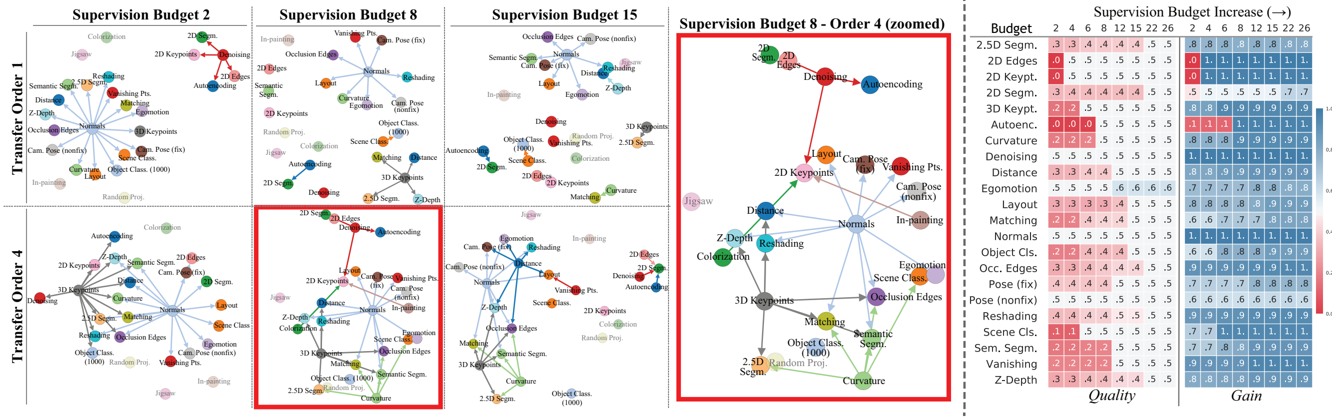

Figure 4: **Left: Sample computed taxonomies** given various supervision budgets (columns), and maximum allowed transfer orders (rows). One is magnified for better visibility. Nodes with incoming edges are target tasks, and the number of their incoming edges is the order of their chosen transfer function. See the interactive solver website for color coding of the nodes based on quantitative evaluations. **Right: Quantitative evaluation of the taxonomy**. *Gain* and *Quality* values for each task using the policy suggested by the taxonomy, as the supervision budget increases(→).

supervision budgets (columns) and maximum allowed order (rows) constraints. However, the method is applicable to any partitioning of the dictionary into $\mathcal{T}$ and $\mathcal{S}$ and arbitrary budget arguments. The interactive solver website allows the user to specify any partition and arguments and see the results.

While Fig. 4-left qualitatively shows the structure and connectivity, Fig. 4-right quantifies the results of taxonomy recommended transfer policies by two metrics of *Gain* (win rate against a network trained without leveraging transfer learning) and *Quality* (win rate against a gold-standard fully supervised network). For detailed discussions and complete definitions, see Sec. "Experiments" of the full paper.

4 From Visual Tasks to Visuomotor Tasks

Taskonomy devises a transfer learning structure among **visual** tasks and enables transferring the knowledge to novel ones. It is worthwhile to consider if and how **visual tasks** can assist with (i.e. "transfer to") learning **downstream robotic tasks**, e.g. navigation in an unseen building. This is of particular importance as one of the primary applications of computer vision is enabling autonomous agents to perceive the world toward their downstream goal, which often entail solving a set of (a priori unknown) visual tasks.

We systematically study this question in [Sax *et al.*, 2018], by integrating a generic perceptual skill set based on Taskonomy's dictionary within a reinforcement learning framework (see Fig. 5). This skill set (**mid-level vision**) provides the policy with a more processed state of the world compared to raw images. We find that using a mid-level vision confers significant advantages over training end-to-end from scratch (i.e. not leveraging visual priors about the world) in navigation-oriented tasks. Agents are able to generalize to situations where the from-scratch approach fails and training becomes significantly more sample efficient. However, we show that realizing these gains requires careful selection of the mid-level vision skills. Therefore, we use the structure among visual tasks found by Taskonomy to devise an efficient *max-coverage task set* that can be adopted in lieu of raw images. Please see [Sax *et al.*, 2018] for full details of this study.

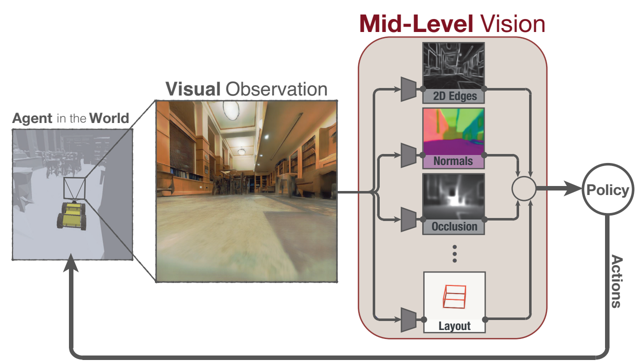

Figure 5: **A mid-level vision module in an end-to-end framework for learning active robotic tasks.** We systematically study if/how a set of generic mid-level **visual tasks** (based on Taskonomy's dictionary) can help with learning downstream **robotic tasks**.

5 Related Literature

Assertions of existence of a structure among tasks date back to the early years of modern computer science, e.g. with Turing arguing for using learning elements [Turing, 1950; Winograd, 1991] rather than the final outcome or Jean Piaget's works on developmental stages using previously learned stages as sources [Piaget and Cook, 1952; Gopnik *et al.*, 1999], and have extended to recent works [Pentina and Lampert, 2017; Kokkinos, 2016]. Here we make an attempt to actually find this structure. We acknowledge that this is related to a breadth of topics, e.g. compositional modeling [Geman *et al.*, 2002; Boiman and Irani, 2007; Lake *et al.*, 2016], few-shot learning [Salakhutdinov *et al.*, 2012; Fe-Fei and others, 2003; Socher *et al.*, 2013], transfer learning [Pratt, 1993], un/semi/self-supervised learning [Erhan *et al.*, 2010; Bengio *et al.*, 2013; Doersch *et al.*, 2015; Donahue *et al.*, 2014; Wang *et al.*, 2017; Thrun and Pratt, 2012; Bingel and Søgaard, 2017], homomorphic cryptography [Henry, 2008], lifelong learning [Chen and Liu, 2016; Silver *et al.*, 2013], just to name a few. For a discussion on how our study relates to self-supervised learning, unsupervised learning, meta-learning, domain adaptation, and multi-task learning, please see "Related Work" in the full paper.

References

[Armeni *et al.*, 2017] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 3

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013. 4

[Bingel and Søgaard, 2017] Joachim Bingel and Anders Søgaard. Identifying beneficial task relations for multi-task learning in deep neural networks. *arXiv preprint*, 2017. 4

[Boiman and Irani, 2007] Oren Boiman and Michal Irani. Similarity by composition. In *Advances in neural information processing systems*, pages 177–184, 2007. 4

[Chen and Liu, 2016] Zhiyuan Chen and Bing Liu. *Lifelong Machine Learning*. Morgan & Claypool Publishers, 2016. 4

[Doersch *et al.*, 2015] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015. 4

[Donahue *et al.*, 2014] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014. 4

[Erhan *et al.*, 2010] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 2010. 4

[Fe-Fei and others, 2003] Li Fe-Fei et al. A bayesian approach to unsupervised one-shot learning of object categories. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1134–1141. IEEE, 2003. 4

[Ge, 2013] Rong Ge. *Provable algorithms for machine learning problems*. PhD thesis, Princeton University, 2013. 2

[Geman *et al.*, 2002] Stuart Geman, Daniel F Potter, and Zhiyi Chi. Composition systems. *Quarterly of Applied Mathematics*, 60(4):707–736, 2002. 4

[Gopnik *et al.*, 1999] Alison Gopnik, Andrew N Meltzoff, and Patricia K Kuhl. *The scientist in the crib: Minds, brains, and how children learn.* William Morrow & Co, 1999. 4

[Gurobi Optimization, 2016] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2016. 3

[Henry, 2008] Kevin Henry. The theory and applications of homomorphic cryptography. 2008. 4

[Hoshen and Peleg, 2015] Yedid Hoshen and Shmuel Peleg. Visual learning of arithmetic operations. *CoRR*, 2015. 2

[Kokkinos, 2016] Iasonas Kokkinos. Ubernet: Training auniversal'convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. *arXiv preprint arXiv:1609.02132*, 2016. 4

[Lake *et al.*, 2016] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, pages 1–101, 2016. 4

[Mccloskey and Cohen, 1989] Michael Mccloskey and Neil J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 24, 1989. 2

[Pentina and Lampert, 2017] Anastasia Pentina and Christoph H Lampert. Multi-task learning with labeled and unlabeled tasks. *stat*, 1050:1, 2017. 4

[Piaget and Cook, 1952] Jean Piaget and Margaret Cook. *The origins of intelligence in children*, volume 8. International Universities Press New York, 1952. 4

[Pratt, 1993] Lorien Y Pratt. Discriminability-based transfer between neural networks. In *Advances in neural information processing systems*, pages 204–211, 1993. 4

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 2

[Saaty, 1987] R. W. Saaty. The analytic hierarchy process – what it is and how it is used. *Mathematical Modeling*, 1987. 3

[Salakhutdinov *et al.*, 2012] Ruslan Salakhutdinov, Joshua Tenenbaum, and Antonio Torralba. One-shot learning with a hierarchical nonparametric bayesian model. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 2012. 4

[Sax *et al.*, 2018] Alexander Sax, Bradley Emi, Amir R Zamir, Leonidas Guibas, Silvio Savarese, and Jitendra Malik. Mid-level visual representations improve generalization and sample efficiency for learning visuomotor policies. *arXiv preprint*, 2018. 4

[Sharif Razavian *et al.*, 2014] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014. 2

[Silver *et al.*, 2013] Daniel L. Silver, Qiang Yang, and Lianghao Li. Lifelong machine learning systems: Beyond learning algorithms. In *in AAAI Spring Symposium Series*, 2013. 4

[Socher *et al.*, 2013] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, pages 935–943, 2013. 4

[Standley *et al.*, 2019] Trevor Standley, Amir R Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? *arXiv preprint*, 2019. 2

[Thrun and Pratt, 2012] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012. 4

[Turing, 1950] Alan M Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950. 4

[Wang *et al.*, 2017] Xiaolong Wang, Kaiming He, and Abhinav Gupta. Transitive invariance for self-supervised visual representation learning. *arXiv preprint arXiv:1708.02901*, 2017. 4

[Winograd, 1991] Terry Winograd. *Thinking machines: Can there be? Are we*, volume 200. University of California Press, Berkeley, 1991. 4

[Zamir *et al.*, 2018] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3712–3722, 2018. 2

[Zhou *et al.*, 2014] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 2