Semi-Supervised Learning for Multi-Task Scene Understanding by Neural Graph Consensus

Marius Leordeanu[1,2], Mihai Cristian Pîrvu[1,2], Dragos Costea[1,2], Alina E Marcu[1,2], Emil Slusanschi[1] and **Rahul Sukthankar[3]**

[1] University Politehnica of Bucharest
[2] Institute of Mathematics of the Romanian Academy
[3] Google Research
{marius.leordeanu,emil.slusanschi}@cs.pub.ro
{mihai_cristian.pirvu,dragos.costea,alina.marcu}@upb.ro
sukthankar@google.com

Abstract

We address the challenging problem of semi-supervised learning in the context of multiple visual interpretations of the world by finding consensus in a graph of neural networks. Each graph node is a scene interpretation layer, while each edge is a deep net that transforms one layer at one node into another from a different node. During the supervised phase edge networks are trained independently. During the next unsupervised stage edge nets are trained on the pseudo-ground truth provided by consensus among multiple paths that reach the nets' start and end nodes. These paths act as ensemble teachers for any given edge and strong consensus is used for high-confidence supervisory signal. The unsupervised learning process is repeated over several generations, in which each edge becomes a "student" and also part of different ensemble "teachers" for training other students. By optimizing such consensus between different paths, the graph reaches consistency and robustness over multiple interpretations and generations, in the face of unknown labels. We give theoretical justifications of the proposed idea and validate it on a large dataset. We show how prediction of different representations such as depth, semantic segmentation, surface normals and pose from RGB input could be effectively learned through self-supervised consensus in our graph. We also compare to state-of-the-art methods for multi-task and semi-supervised learning and show superior performance.

Introduction

We propose the Neural Graph Consensus (NGC) model, a multi-class graph of deep neural networks, which approaches one of the most difficult problems in AI, that of unsupervised learning of multiple scene interpretations. Space-time data is so inexpensive to record, yet so expensive to annotate. While classic deep learning is powerful, it is almost hopeless when no ground truth is available. A popular alternative is reinforcement learning (Sutton, Barto et al. 1998; Silver et al. 2016), which is computationally very demanding when applied to real-world tasks. We show that NGC, based on a set of sound principles, overcomes many of the current limitations in unsupervised learning, by bringing together graphs and neural nets, within a single multi-task structure. We have ample evidence for the power of deep nets (Ciresan et al. 2011; Krizhevsky, Sutskever, and Hinton 2012; LeCun, Bengio, and Hinton 2015); given enough data and sufficient supervision they can master almost any task. On the other hand, graphs are able to transcend to global solutions: starting from local knowledge they reach the global level through iterative message passing (Pearl 2014; Yedidia, Freeman, and Weiss 2005; Boykov, Veksler, and Zabih 2001; Besag 1986). Thus, iterative graph methods tend to converge to stable solutions, where consensus between the local and the global is reached and a higher "truth" emerges gradually. We take advantage of this property of classical graphical models and boost it by replacing the simple functions on nodes and edges, with powerful deep nets that will transform an entire scene representation into another. The idea also differs from current work in graph neural nets, which is mainly supervised and still uses relatively simple functions on edges (e.g., MLPs) and local receptive fields (Nicolicioiu, Duta, and Leordeanu 2019; Duta, Nicolicioiu, and Leordeanu 2020; Battaglia et al. 2016; Xu et al. 2019).

We create a graph (or more generally, a hypergraph) of deep networks, such that each node in the graph represents a different interpretation layer of the world (e.g. semantic segmentation layer, depth layer, motion layer). The edges (or hyperedges) have dedicated deep networks, which predict the layer at one node from the layer, or layers, at one or several other nodes. The deep nets are trained in turn, using as supervision the consensual output of all paths reaching the same output node (when such consensus takes place). Since we do not have strict, well defined "worlds" as in reinforcement learning (RL), we let the natural structure in the data emerge, as in graph clustering. Similar to RL, in which agents, in heavily constrained environments with well defined goals, evolve by playing against each other, we put pressure on the deep nets by training them one versus the others, in complementarity. Thus, each net brings something new, while also learning to agree with the other paths that reach the same output. During such consensus training, the NGC graph will approach a state of equilibrium. This idea is related to early work on grouping from the Gestalt school of psychology (Koffka 2013) and Grossberg's clas-

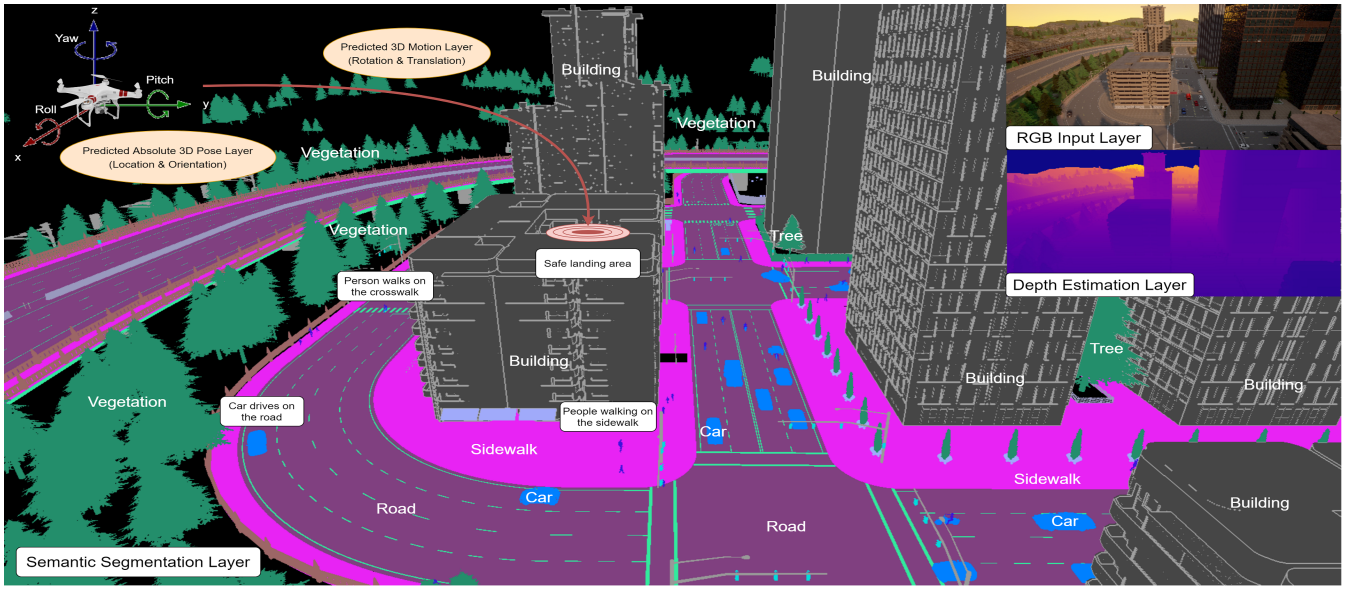


Figure 1: NGC can put together different interpretations of the dynamic scene, such as 3D structure, pose, motion, semantic segmentation of objects and activities in different regions of space and time, into a unified neural graph, in which multiple paths reaching a given node become teacher through consensual agreements to any single edge net reaching the same node. Trained in this self-supervised manner, NGC can reach robust unsupervised learning in the face of unlabeled data. The scene in the figure is taken from a virtual environment used to collect data for our experiments. While in this paper we do not take advantage of the temporal component, we show that NGC can effectively learn semi-supervised, to predict the drone 6D pose, scene depth, 3D structure, and semantic segmentation, from a single RGB image.

sical Adaptive Resonance Theory on how the brain learns and reaches conscious states (Grossberg 1976, 2000, 2015). Interestingly, our idea also relates to the general concept of homeostasis (Betts et al. 2016), which is the innate tendency of all living organisms towards a relatively stable equilibrium between their many interdependent elements, in order to maintain their life.

In Fig. 1 we present the general NGC concept in the context of a drone that learns to understand its environment by finding consensus among the many different interpretations of the space-time world. Within a large but coherent NGC we see how different tasks, such as predicting semantic classes, 3D structure, pose, motion and detection of activities in different space-time regions, are inter-connected.

Scientific Context

There is current work that tackles unsupervised learning by constraining together multiple tasks, such as depth and relative pose (Chen, Schmid, and Sminchisescu 2019; Godard et al. 2019; Zhou et al. 2017; Ranjan et al. 2019; Bian et al. 2019; Gordon et al. 2019; Yang et al. 2018). Other works include semantic segmentation into the equation (Tosi et al. 2020; Guizilini et al. 2020; Chen et al. 2019; Stekovic, Fraundorfer, and Lepetit 2020). There is also work that goes beyond vision, to consider input from other senses, for unsupervised learning by cross-modal prediction (Hu, Nie, and Li 2019; Li et al. 2019; Zhang, Isola, and Efros 2017; Pan et al. 2004; He et al. 2017; Zhao et al. 2018). These works are related to our NGC concept. By constraining different tasks, they are actually learning to satisfy consensus among multiple representation paths, even though they are limited to a few specific tasks with specific domain constraints.

There is also work that shows that clustering together several tasks to learn, through self-supervised agreement more compact features and descriptors could significantly help in learning, but the evaluation is still done in supervised settings. That is mainly because the unsupervised problem is kept at the level of unlabeled, anonymous clusters as it is usually done in the classic unsupervised clustering literature. Such papers include methods based on contrastive learning (Chen et al. 2020; Henaff et al. 2020; Caron et al. 2020). For example, Contrastive Multiview Coding (Tian, Krishnan, and Isola 2020) proposes a compact descriptor from multiple representations. They show that a larger number of representations results in a better descriptor being learned for each scene. Another, related work, aims to maximize information from multiple views of a shared context (Bachman, Hjelm, and Buchwalter 2019). A more general contrastive loss framework is presented in (Patrick et al. 2021). Learning from agreements among collaborative experts also proved useful in video retrieval (Liu et al. 2019).

An important unsupervised source of information is the temporal dimension which provides spatiotemporal consistency. This cue is exploited by an increasing number of recent papers for semantic segmentation propagation or pose propagation (Marcu et al. 2020; Wang, Jabri, and Efros 2019; Jabri, Owens, and Efros 2020) or interpretable keypoints (Jakab et al. 2020). Our NGC model is general and

can easily integrate information from different moments in time and take advantage of temporal consistency.

Multi-task learning research also starts taking advantage of consistency and agreements between multiple tasks: Taskology (Lu et al. 2021) aims to do so by training pairs of tasks, using an unlabeled dataset to add a consistency loss term to the supervised one. However, the method does not have any purely unsupervised learning phase. Probably most related to ours is the interesting work on Cross-Task Consistency (CTC) (Zamir et al. 2020) and Taskonomy (Zamir et al. 2018). In (Zamir et al. 2020) authors learn robust scene representations by using at least two intermediary representation conversions, which immediately relates to our multi-hop pathways in the NGC graph. However, our contributions over (Zamir et al. 2020) are significant and complementary, as could be best understood from the conclusions in (Zamir et al. 2020), which enumerates several topics that fall outside their scope. Unlike (Zamir et al. 2020), our paper uses path ensembles to form strong consensual output for unsupervised learning, created from multiple multi-task pathways in the NGC (hyper)graph. We also tackle the case of categorical data both in the theoretical analysis and in the experiments sections. Our semantic segmentation tasks are treated as categorical data, with a voting-based learning scheme for building pseudo-ground truth. Then we learn from completely unlabeled data over multiple NGC iterations, each learning cycle having a different unlabeled training set. Very different from the learning cost in (Zamir et al. 2020), we show and motivate a way to construct pseudo-ground truth and continually improve on multiple tasks over multiple iterations in an unsupervised manner.

Our NGC model is unique in the way it puts many well-defined tasks together, in a single hyper-graph structure, such that the output of one task can be the input of another. Then, NGC uses the consensual output among multiple pathways that transform one task into another and reach the same node (with a specific interpretation of the scene) as unsupervised, pseudo-ground truth. After each iteration the models on the single task edges improve, overall consensus in the NGC hypergraph also improves, new data is added and the process continues.

Instead of letting the final problem be supervised and confining unsupervised learning at the level of anonymous features (as most prior work does), our approach is in fact the opposite: put the many different final tasks inside the greater unsupervised learning graph system, then train them (in a classical way, using well-known costs published in the literature) on the unsupervised consensual output produced by the many pathways in the graph, which reach the input and output nodes corresponding to each task.

Our work is also related to classic ensemble approaches, which are very effective but limited to a single task. There is also work that shows how to use ensembles of nets for unsupervised learning (Croitoru, Bogolin, and Leordeanu 2019). However, no other work shows how a single structure can put together a combinatorial number of ensembles within itself to learn simultaneously and semi-supervised many different tasks. Our results suggest that by putting all these ensembles and tasks together within a graph, we do not make the tasks harder, but easier, as the graph structure offers them access to a large number of teacher ensembles.

Main Contributions

We propose Neural Graph Consensus (NGC), a novel model for semi-supervised learning of multiple scene interpretations, which connects many deep nets in a single graph structure, with nets on edges and interpretation layers on nodes. Each net transforms one representation from one node into another, from a different node. We show how different tasks can teach themselves through self-supervised consensus. We offer theoretical justification of our approach and also prove its effectiveness in experiments and comparisons to top methods in the field. Moreover, we provide public access to our dataset and code[1].

Neural Graph Consensus Model

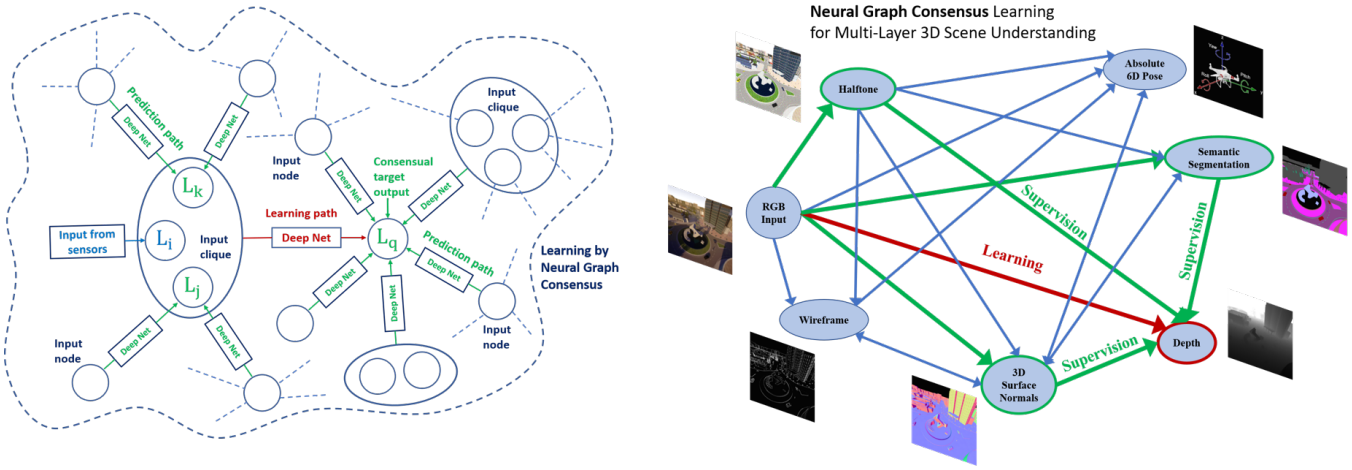
Each node i in the graph (or hypergraph) has an associated layer L_i, which encodes a specific view or interpretation of the space-time world (e.g. depth map, semantic segmentation, regions in space-time for various activities). The layer at one node can be predicted from one or more other layers at other nodes by deep nets. These nets that transform one or multiple layers into another, form edges (or hyperedges) in the graph (hypergraph). In Fig. 2A, we show the general concept: L_i, L_j and L_k form an input clique of nodes and L_q is the layer at the output node. NGC can be arbitrarily connected: many paths starting from sensor inputs can reach a given internal node.

Unsupervised learning with NGC: In general, we expect to have access to limited labeled data to pretrain separately a good part of our nets. Once we connect them into a NGC graph, we can start the unsupervised learning phase. Let us consider unsupervised training of the net associated with the hyperedge having input clique (L_i, L_j, L_k) and output node L_q (in red, Fig. 3A). There could be many prediction paths (in green) going from other layers and sensors reaching L_q. The idea is that we let prediction information flow through the green paths and use their consensual outputs as supervisory signal for the red network. In our experiments we consider the mean output as consensus (in the case of regression) and majority voting (in the case of classification). Note that all layers involved that are not coming from sensors (e.g. L_i, L_j, L_k,) are established by the same consensus strategy as L_q. Unsupervised learning will take turns: each net becomes student of the NGC hypergraph and is trained by the mutual consensus coming from the contextual pathways that reach the same output node. NGC becomes a democratic self-supervised system for which agreement becomes the ultimate teacher in the face of unlabeled data.

Theoretical and Numerical Analysis

Unsupervised learning in the case of regression: Let L_q be a scalar value x_q. The case immediately extends to the

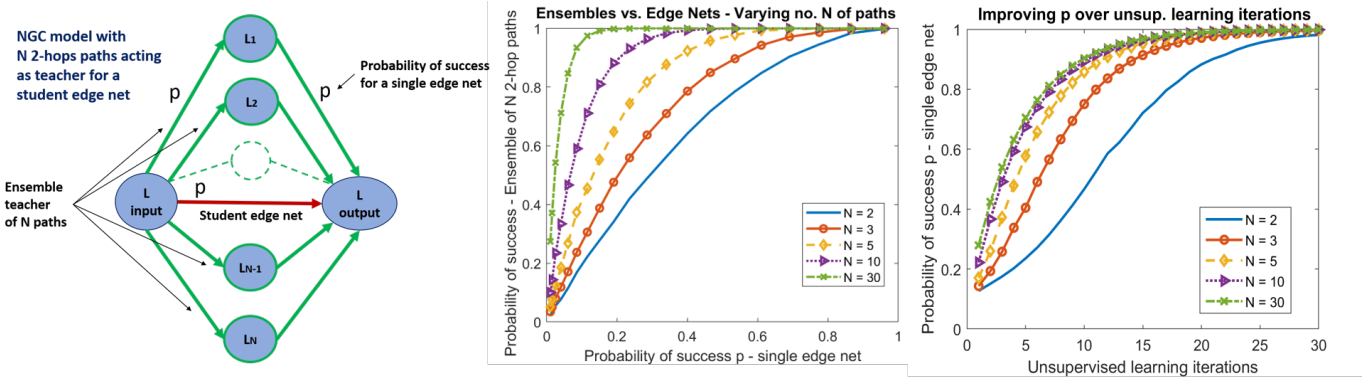
[1]https://sites.google.com/site/aerialimageunderstanding/semi-supervised-learning-of-multiple-scene-interpretations-by-neural-graph



A. General Conceptual NGC Model

B. Actual NGC model implemented

Figure 2: A: – formal representation of the NGC model as graph (or hypergraph) of deep nets. Each node has an interpretation layer of the "world". Each edge learns to predict the layer at one node from one or (several) input layers from other nodes. Each layer has an associated space-time region. NGC can operate in both space and time, with access to past input layers. The deep net shown in red, with input from a clique of layers learns to predict layer L_q, using as pseudo-ground truth, the state of L_q reached by consensus by the green nets reaching node q. Layers at all nodes in NGC are either input from sensors or decided through consensus. The deep nets on edges take turns in being "trained students" or in "collaborating through consensus" as part of teacher ensembles, to train other nets. B: The actual NGC structure and representations used in the paper: learn semi-supervised to predict depth, semantic segmentation, absolute 6D pose (position and orientation) and 3D world structure, in the case of a drone flying in a virtual environment.



A B C

Figure 3: A. The simple NGC model with 2-hop pathways used in simulations. B. Performance of the teacher ensemble vs. p, for different N of ensemble paths and 100 classes per node. Plots are averages over 10k tests. C. Improving single edge nets over iterations, for different number of paths, when the next iteration edge nets recover 20% of the performance gap to their teacher.

multivariate case. Let $x_q^{(c)}$ be the value at node q predicted by the net coming from input clique c to node q. Let t_q be the ideal true value for node q. For ease of presentation, we consider L2 loss for training. In the supervised case, for the net going from input clique c to output node q, the objective loss over all training cases is: $J_{\text{sup}} \sum (x_q^c - t_q)^2$. In the unsupervised case, with many different paths reaching node q, we assume that the errors made by the predictive nets from all cliques a to node q are independent and unbiased. Then, we have $E_a(X_q^{(a)}) = t_q$. Also, based on the law of large numbers we expect that the ground truth can be approximated by $t_q = E_a(X_q^{(a)}) \approx \frac{1}{N_q} \sum_{q=1}^{N_q} x_q^{(a)}$, in the small variance sense, as the number of paths going into node q is large. Thus, in the unsupervised case we could use instead of t_q its empirical approximation. Note that in practice we could do

Algorithm 1 Learning with Neural Graph Consensus

Step 1: Pre-train a set of deep neural networks that transform different input to output representations, using the labeled data available.
Step 2: Form the NGC graph by linking the nets such that the output of one (or several) becomes input to another.
Step 3: On a completely new unlabeled set, re-train the nets using as pseudo-ground truth for a specific node (representation) the consensual output of all paths that reach that node. Repeat Step 3, by choosing a new unlabeled set and newly trained nets, until convergence.

even better and use as pseudo-ground truth, the median or a smart voting scheme. If we use the above approximation of t_q in the L2 loss, the unsupervised loss J_{unsup} becomes:

$$J_{\text{unsup}} = \sum (x_q^c - \frac{1}{N_q} \sum_{q=1}^{N_q} x_q^{(a)})^2 \quad (1)$$

The unsupervised loss above is also an approximation of variance. Thus, by minimizing it we expect to minimize variance in outputs along different paths that reach the same node. This leads to the following conclusion:

Proposition 1: In a densely connected NGC graph, we expect the variance over the outputs reaching a given node to decrease during unsupervised learning.

Observation: In our experiments over seven different tasks, in a single iteration of unsupervised learning the standard deviation over ensemble outputs reaching a specific node reduced nearly half ($56\% \pm 13\%$). This is empirical evidence that our proposed approach (Algorithm 1) increases not just performance but also the level of overall agreement in the NGC graph.

Unsupervised learning in the case of classification: Classification offers a more complex case: the agreements among paths are found by voting, harder to analyze than simple averages. We consider the case with 2-hop paths that start from input and reach the same output node, after passing through some intermediate layer, to become teacher ensemble for the direct input-output link (Fig. 2A). For clarity of presentation we consider a single class per node, without loss of generality. At each node, a class is chosen by the incoming edge net. We assume that all edge nets function independently and the probability of success (correct class chosen given a specific input) is p. Then, what is the probability of success of the 2-hop teacher ensemble using majority voting? How does the number of classes C, the number N of 2-hop paths and probability p influence the performance of the ensemble vs. the single direct edge. This is important since the teacher provides the pseudo-ground truth for the next generation single edge nets (Alg. 1).

We make the following assumptions: for a given edge net, if the input class is correct then the probability of success is p, for a total number of C classes. For a correct input, the chance of a wrong class at output is uniformly distributed among the remaining classes. For the first, input node, coming from sensors, we always consider its input as being correct. According to this model, we can show that the probability of success along any 2-hop pathway is $p_e^{(+)} = p^2 + (1-p)\frac{1}{C}$ and the probability of a wrong output over a 2-hop path is $p_e^{(-)}(1-p)(p + \frac{C-1}{/}C)$. We can show that $p_e^{(+)}$ for a single 2-hop path is still better than random chance $p_e > \frac{1}{c}$, if we assume $p > \frac{1}{c}$.

Several key observations are relatively easy to prove: the percentage of correct votes v_c over N different 2-hop paths in an ensemble is expected to be $p_e^{(+)}$, while the expected percentage of wrong votes for any given wrong class v_w is $p_e^{(-)}/(C-1)$. While the expected values do not depend on N, their variance decreases towards zero as $N \to \infty$, since: $Var(v_c) = \frac{p_e^{(+)}(1-p_e^{(+)})}{N}$ and $Var(v_w)\frac{p_e^{(-)}(1-p_e^{(-)})}{N}$. Then we reach the following result:

Proposition 2: If the success probability p for an edge net is better than random $p > \frac{1}{C}$, then the probability of success p_{eN} of the teacher ensemble (2-hop paths) using majority voting goes towards 1 as the number of paths $N \to \infty$.

Proof: Let $\mu = p_e^{(+)} - p_e^{(-)}$ and $\sigma = \sqrt{Var(v_c) + Var(v_w)}$ Then, the probability of error of an ensemble of N 2-hop paths is (by Chebyshev's inequality): $p_{eN}^{(-)} = Pr(v_c - v_w < 0) \leq Pr(\|v_c - v_w - \mu\| < \mu) \leq \frac{\sigma^2}{\mu^2}$. If we plug in the formulas for σ, μ, $Var(v_c)$ and $Var(v_w)$, we obtain:

$$p_{eN}^{(-)} \leq \frac{1}{N} \frac{p_e^{(+)}(1-p_e^{(+)}) + p_e^{(-)}(1-p_e^{(-)})}{(p_e^{(+)} - p_e^{(-)})^2} \quad (2)$$

The result leads to the conclusion that the accuracy of the ensemble improves towards 1, as $N \to \infty$. We performed simulation experiments to visualize the accuracy of the teacher ensemble of 2-hop paths vs. the single edge nets, based on the model presented here (Fig. 3A and 3B). The improvement of the ensemble over the single net depends mostly on the number of paths. As predicted by the model, our simulations also verified that the number of classes C does not influence significantly these curves. We also simulated the effect of learning over multiple iterations. We consider the case when the single edge net of next iteration recovers only 20% from the gap to the teacher, after being trained on the teacher output at the previous iteration (Fig. 3D). While our simulations are based on simplified mathematical models, they suggest an interesting intuitive observation: **if** 1) paths are many and independent (by having a large diverse pool of representations at nodes in NGC), and 2) single nets are able to keep up with their ensemble pathways over iterations, and 3) we keep bringing new unlabeled data from one iteration to the next (to maintain independence over time), then we could expect continuous improvement over many iterations.

Complexity considerations: The complexity cost of training one NGC iteration is O(№ of edges). If new unlabeled data is added, which could potentially bring novel information, NGC can learn continuously. Otherwise iterations should stop when differences between student and ensemble teacher stops decreasing.

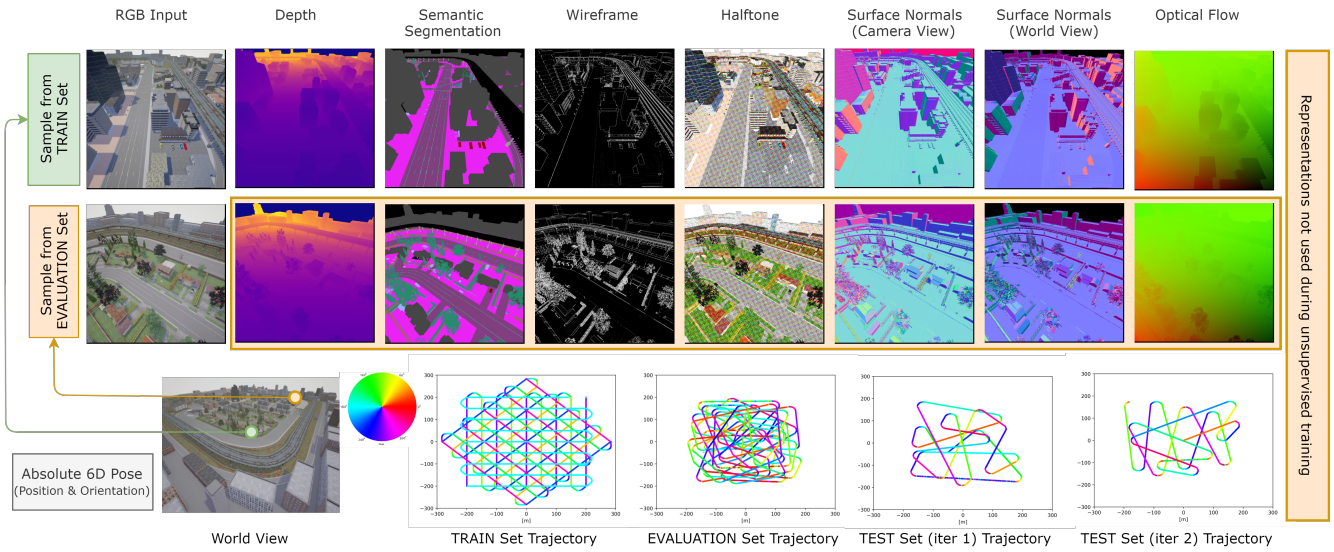


Figure 4: Samples from the synthetic dataset (train, first row, evaluation, second row). The hue for the trajectories (third row) encodes the yaw angle (see the color circle above). The paths simulate a drone trajectory, with small random variations in all angles. While the training path is a grid-based, traditional surveying flight, the test path aims to capture as much viewpoints as possible, for a robust evaluation. The distance between two adjacent training trajectories is 50m, chosen such that two parallel images have a small overlap.

Representation	Evaluation Metric	Iteration 0	Iteration 1		Iteration 2	
		EdgeNet	NGC	Distil. EdgeNet	NGC	Distil. EdgeNet
Depth	L1 (meters)	4.9844	3.4867	4.2802	**3.2994**	**3.9508**
	Pixels ↑ (%)	-	79.30	60.66	**79.69**	**61.90**
Surface Normals (C)	L1 (degrees)	8.4862	7.7914	8.2891	**7.4503**	**7.6773**
	Pixels ↑ (%)	-	74.18	53.59	**74.61**	**53.94**
Surface Normals (W)	L1 (degrees)	11.8859	8.8248	10.7500	**8.5282**	**8.6714**
	Pixels ↑ (%)	-	79.95	57.88	**81.12**	**61.14**
Semantic Segmentation	Accuracy	0.9001	0.9181	0.9019	**0.9245**	**0.9283**
	mIOU	0.4840	0.4978	0.4980	**0.5258**	**0.5159**
	Pixels ↑ (%)	-	79.46	69.62	**81.49**	**71.95**
Wireframe	Accuracy	0.9617	0.9655	0.9654	**0.9661**	**0.9655**
	Pixels ↑ (%)	-	77.71	72.57	**78.02**	**73.46**
Position	L2 (meters)	25.7597	15.5383	20.0204	**12.0764**	**15.5599**
Orientation	L1 (degrees)	3.8439	2.5001	3.3961	**2.2088**	**3.0005**

Table 1: Results for our proposed ensemble NGC and distilled EdgeNets on 6 representations, over 2 iterations of unsupervised learning. We show best results over ensembles in NGC (bolded) and single EdgeNets (bolded). Note the consistent improvements from one iteration to the next for both ensembles teachers as well as single student nets.

Experimental Analysis

Dataset details and description: To test the NGC approach in the case of many scene representations we capture a large dataset using a customized virtual environment based on the CARLA simulator (Dosovitskiy et al. 2017), in which a drone flies above a city and learns to predict from a single image the scene depth, the 3D surface normals (both from the world and camera system of reference), its absolute location and orientation (6D pose), the scene wireframe (object boundaries) as well as the semantic segmentation of the scene in 12 classes: building, fence, pedestrian, pole, road line, road, sidewalk, vegetation, vehicle, wall, traffic sign, other (Fig. 4). The dataset is divided into four subsets: supervised training set (subdivided in 8k images for training and 2k for validation), 2 test sets (10 k images each, for unsupervised learning iterations 1 and 2) and a separate evaluation set (10 k images, never seen during learning).

Implementation details: We developed a general NGC framework on top of the existing deep learning framework PyTorch (Paszke et al. 2019), which can model arbitrary complex graphs and which we make publicly available. For

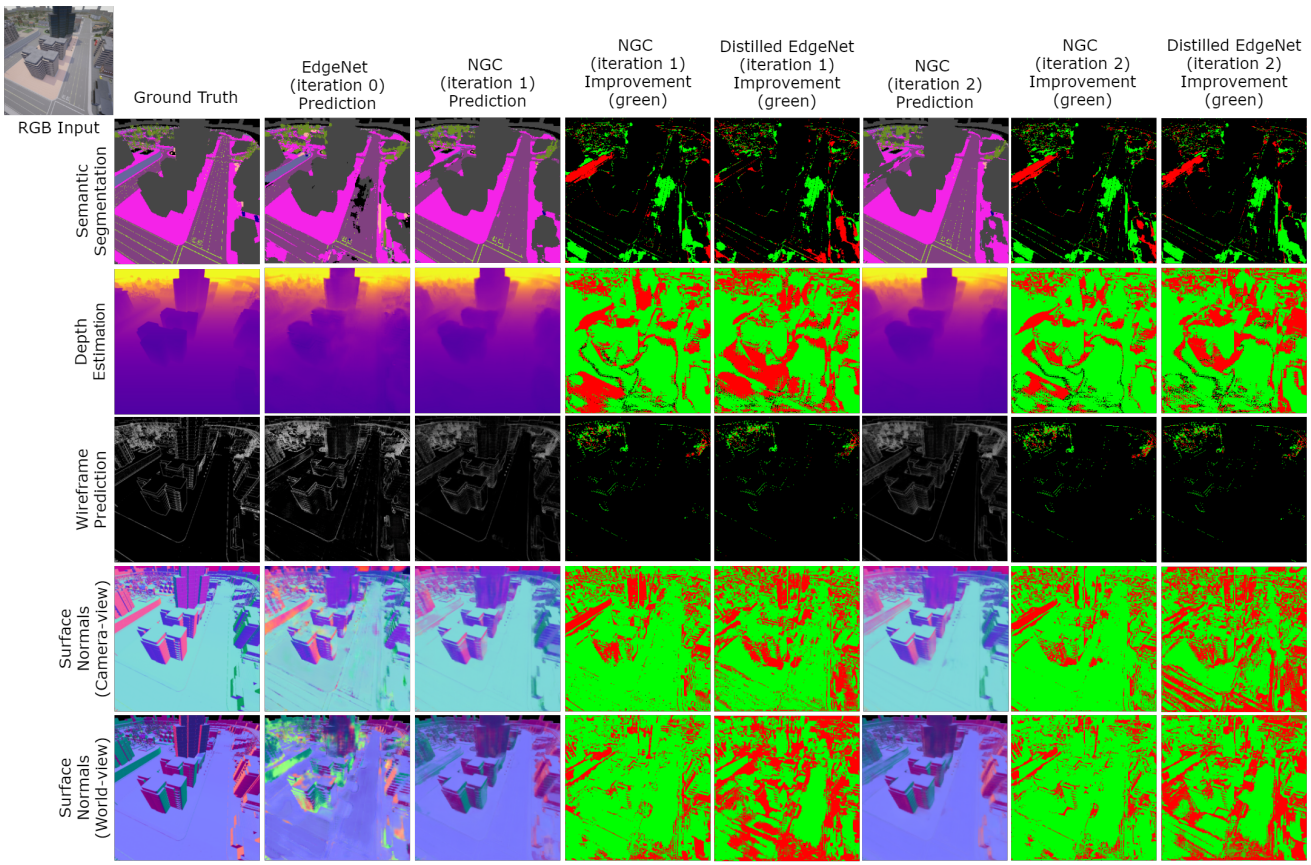


Figure 5: Qualitative results for NGC. The plots show performance improvement (green) vs performance degradation (red) on five tasks. We compare the original predictions with two graph iterations and their distilled EdgeNets.

	RGB	Depth	SSeg	Halftone	Surf. Norm. (C)	Surf. Norm. (W)	Wireframe	Pose
Depth	✓		✓	✓	✓	✓		
SSeg	✓			✓		✓		
Halftone	✓							
Surf. Norm. (C)	✓					✓	✓	
Surf. Norm. (W)	✓			✓	✓		✓	
Wireframe	✓			✓				
Pose	✓		✓	✓	✓			

Table 2: Dependency table showing the best graph configuration found automatically, used for both iterations 1 and 2. The element in the dependency table at position (i, j) = ✓ if and only if node at row i uses intermediate node from column j, during NGC learning (i.e. there is an edge $j \rightarrow i$, between nodes j and i in NGC). We use the direct edge RGB $\rightarrow$ any of the output nodes, in the first column, for all of the learned representations. All the other dependencies, marked in the dependency table above, were automatically determined by optimizing the accuracy over the training set.

Task	Metric	EdgeNet(iter 0)	NGC	NDDR*(no pretrain)	NDDR*(pretrain)	NDDR	MTL-NAS
Semantic Segm.	mIOU	0.484	**0.498**	0.141	0.343	0.315	0.368
	Acc.	90.017	**91.816**	48.7	84.2	86.9	87.8
Normals (C)	Err (deg.)	8.4862	7.7914	9.820	7.727	6.801	**6.533**

Table 3: Multi-task learning results. All methods were trained on the same supervised data (our train set) and tested on the evaluation set. NGC uses the ensemble outputs of EdgeNet (iter 0). Single task networks (marked NDDR* below) were trained with and without pretraining (on other datasets than ours). Best results are shown in bold.

Metric	EdgeNet (iter 0)	NGC (iter 2)	EdgeNet (iter 2)	CCT (supervised)	CCT (semi-supervised)
mIOU	0.484	**0.526**	0.516	0.353	0.353
Accuracy	0.9001	0.9245	**0.9283**	0.8463	0.8503

Table 4: Semantic segmentation comparisons on our evaluation set with the semi-supervised CCT(Ouali, Hudelot, and Tami 2020). We outperform CCT both in absolute measures as well as relative improvement (semi-supervised vs. supervised). Best results are shown in bold.

EdgeNets we used two types of networks: Map2Map and Map2Vector. The Map2Map architecture, used for most edge nets (since most output a image-size map) is based on a small UNet-style architecture. The Map2Vector has the same encoder as Map2Map, but the decoder outputs an output vector instead of a map (with a fully connected layer at the end) and is used only for predicting the 3D position vector and the 3D orientation vector. All architectures have about 1.1M trainable parameters, making them very light compared to most state-of-the-art nets for similar tasks. They are trained for 100 epochs, with AdamW optimizer, using our novel Pytorch-based NGC graph framework. **Note:** We have a total of 27 edge nets in NGC, each with a unique transformation from input to output representations, totaling about 30M parameters in the entire NGC model.

Building the graph from independent single edges: The NGC graph system is composed of multiple EdgeNets, which are edges of the graph trained independently and then connected to create NGC. In our setup, the maximum depth of the graph is 2, so for each $Task$, we either have the direct edge $RGB \rightarrow Task$ or $RGB \rightarrow Intermediate \rightarrow Task$. Thus, we have two types of edges, the ones for the first hop $RGB \rightarrow Intermediate$ and the ones for the second hop $Intermediate \rightarrow Task$. Now the question is how to build the graph from these edges in order to create the full NGC. In order to keep the graph structure discovery computationally feasible, we take a greedy approach. First, we sort the 2-hop pathways $RGB \rightarrow Intermediate \rightarrow Task$ in descending order of their accuracy on the labeled training set. Note that the direct edge $RGB \rightarrow Task$, as expected, always performs better than any 2-hop pathway, in our tests:

- $RGB \rightarrow Task$
- $RGB \rightarrow I_1 \rightarrow Task$
- ...
- $RGB \rightarrow I_n \rightarrow Task$

For a given $Task$, we create N possible graph ensembles, by adding a 2-hop pathway at a time, in descending order of its training accuracy. Then we measure the accuracy of the graph ensemble. Finally, for each such $Task$ we chose the graph ensemble that has best performance on the labeled training set. These selected ensembles of 2-hop pathways form the final NGC graph system. In Tab. 2 we present the graph structure of our NGC, discovered in the greedy manner presented above.

Unsupervised learning over multiple iterations: The unsupervised learning experiments follow the steps of Algorithm 1: during the supervised stage (Step 1), we use the 8k labeled images (train set) to train single EdgeNets to predict one representation from another. Next (Step 2), we create 2-hop paths, from RGB inputs to all representations, by using all the remaining representations as intermediate nodes. Then, we evaluate individually each 2-hop path on the 2k images validation set (part of the labeled train set) and add them, in greedy fashion to the NGC graph, based on their individual performance, to form ensembles, as long as the ensemble performance on the validation set keeps improving. This graph construction procedure is very fast, since the validation set is small (2k images). The actual graph structure discovered in this manner (simpler version in Fig. 2B) is (list of input nodes with edge nets towards the same output node): 1) (rgb, halftone, semantic segmentation, surface normals (camera-view), surface normals (world view)) $\rightarrow$ depth, 2) (rgb, wireframe, surface normals (world-view)) $\rightarrow$ surface normals (camera-view), 3) (rgb, wireframe, surface normals (camera-view), halftone) $\rightarrow$ surface normals (world-view), 4) (rgb, halftone, surface normals (world-view)) $\rightarrow$ semantic segmentation, 5) (rgb, halftone) $\rightarrow$ wireframe 6) (rgb, surface normals (camera-view), surface normals (world-view), semantic segmentation, halftone) $\rightarrow$ absolute 6D pose (3D position + 3D orientation). This gives a total of 27 EdgeNets in our NGC model, each having a unique pair of (input, output) representations. Then we proceed with the unsupervised learning phase (Step 3), for two iterations of learning on test set 1 (iteration 1), then test set 2 (iteration 2). The evaluation is performed on the unseen evaluation set and results reported in Table 1. Note that NGC learning generally improves, for all tasks and representations, from one iteration to the next, significantly outperforming the supervised version (trained only on the labeled train set).

Comparisons on multi-task learning: NGC shares common goals with multi-task and semi-supervised learning. Most multi-task methods focus on how to combine the weights of several neural networks (Misra et al. 2016; Rosenbaum, Klinger, and Riemer 2018; Ruder 2017). We selected NDDR (Gao et al. 2019) and MTL-NAS (Gao et al. 2020), state-of-the-art in 2019 and 2020 on NYU-Depthv2 (Silberman et al. 2012) on two-task learning (semantic segmentation and camera normals) and tested them on our dataset (Tab. 3).

Comparisons on semi-supervised learning: Despite a large body of research on semi-supervised learning, most solutions are tailor-made for specific tasks. For comparison,

Representation	Ensembles	Metric
Semantic Segmentation		mIOU
	Mixt	0.4605
	EdgeNets	0.5248
	NGC (iteration 2)	**0.5258**
Depth		L1 (meters)
	Mixt	3.6608
	EdgeNets	3.3509
	NGC (iteration 2)	**3.2994**

Table 5: Comparison with baseline ensembles. Results are reported on our evaluation set for two learned representations: semantic segmentation and depth estimation. Results show that our NGC is superior to both types of ensembles on both learned tasks. Best results are shown in bold.

we select CCT (Ouali, Hudelot, and Tami 2020), one of the more general approaches with state-of-the-art results on PascalVOC (Everingham et al. 2010). The method is weakly related to ours, as it also exploits consensus by perturbing the output of multiple decoders. Again we test CCT on our dataset, using the same labeled and unlabeled data during semi-supervised learning and report results in Tab. 4. Our NGC (with 30M parameters total and no pretraining on other datasets) is smaller and more effective than the CCT model (46M parameters, with pretrained ResNet50 (He et al. 2016) backbone). We outperform CCT also in relative improvements over the supervised baselines.

Comparisons to different ensembles of networks: Another relevant comparative experiment is against multiple types of vanilla ensembles. We present results on the evaluation set in Table 5 on semantic segmentation and depth prediction. The number of nets forming the ensembles is the same as the ones in the NGC corresponding to each specific tasks. All nets forming the ensembles were trained using the same setup and training dataset as all our EdgeNets. We consider two types of ensembles: **1) Mixt ensemble:** we combine different semantic segmentation models, PSPNet (Zhao et al. 2017), DeepLabV3+ (Chen et al. 2018) together with our EdgeNet and took the average over the three outputs. We lower the number of parameters of PSPNet and DeepLabV3+ to match the number of EdgeNet (1.1M). We adopt the same procedure for depth prediction as well, for which we also add Unet (Ronneberger, Fischer, and Brox 2015) and a larger variant of EdgeNet. **2) EdgeNets ensemble:** We combined different variants of EdgeNet, by varying the number of parameters (1M, 2M or 10M parameters). We also modified the architecture such that instead of concatenating the features in a Unet-like style, we added them.

Conclusions

We present the neural graph consensus (NGC) model, a novel approach to multi-task semi-supervised learning, which brings together the power of neural networks and that of discrete graphs, by combining many different deep networks into a large neural graph that learns, semi-supervised from mutual consensus among multiple pathways. Our extensive experiments and comparisons to top methods, backed by sound theoretical analysis, clearly prove the effectiveness of the model. Our actual implemented NGC, totaling 30M parameters but incorporating no less than 27 nets, learns to predict semi-supervised, from single images and with top performance, seven different scene interpretations. Our results show that learning from consensual outputs in such large, collaborative multi-task neural graphs, is a powerful direction in unsupervised learning.

Acknowledgements

This work was funded by UEFISCDI, under Projects EEA-RO-2018-0496 and PN-III-P4-ID-PCE-2020-2819. We want to express our sincere gratitude towards Aurelian Marcu and The Center for Advanced Laser Technologies (CETAL) for their generosity and providing access to GPU computational resources.

References

Bachman, P.; Hjelm, R. D.; and Buchwalter, W. 2019. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, 15535–15545.

Battaglia, P.; Pascanu, R.; Lai, M.; Rezende, D. J.; et al. 2016. Interaction networks for learning about objects, relations and physics. In *Advances in neural information processing systems*, 4502–4510.

Besag, J. 1986. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society: Series B (Methodological)* 48(3): 259–279.

Betts, J.; Desaix, P.; Johnson, E.; Johnson, J.; Korol, O.; Kruse, D.; and Young, K. 2016. Anatomy and physiology. OpenStax.

Bian, J.; Li, Z.; Wang, N.; Zhan, H.; Shen, C.; Cheng, M.-M.; and Reid, I. 2019. Unsupervised scale-consistent depth and ego-motion learning from monocular video. In *Advances in Neural Information Processing Systems*, 35–45.

Boykov, Y.; Veksler, O.; and Zabih, R. 2001. Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence* 23(11): 1222–1239.

Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; and Joulin, A. 2020. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M. F.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 9912–9924. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf.

Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; and Adam, H. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, 801–818.

Chen, P.-Y.; Liu, A. H.; Liu, Y.-C.; and Wang, Y.-C. F. 2019. Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2624–2632.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, Y.; Schmid, C.; and Sminchisescu, C. 2019. Self-supervised Learning with Geometric Constraints in Monocular Video: Connecting Flow, Depth, and Camera. In *Proceedings of the IEEE International Conference on Computer Vision*, 7063–7072.

Ciresan, D. C.; Meier, U.; Masci, J.; Gambardella, L. M.; and Schmidhuber, J. 2011. Flexible, high performance convolutional neural networks for image classification. In *Twenty-second international joint conference on artificial intelligence*.

Croitoru, I.; Bogolin, S.-V.; and Leordeanu, M. 2019. Unsupervised learning of foreground object segmentation. *International Journal of Computer Vision* 127(9): 1279–1302.

Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; and Koltun, V. 2017. CARLA: An open urban driving simulator. In *Conference on robot learning*, 1–16. PMLR.

Duta, I.; Nicolicioiu, A.; and Leordeanu, M. 2020. Dynamic Regions Graph Neural Networks for Spatio-Temporal Reasoning. In *ORLR Workshop, Neural Information Processing Systems (NeurIPS)*.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The Pascal visual object classes (VOC) challenge. *International journal of computer vision* 88(2): 303–338.

Gao, Y.; Bai, H.; Jie, Z.; Ma, J.; Jia, K.; and Liu, W. 2020. Mtl-nas: Task-agnostic neural architecture search towards general-purpose multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11543–11552.

Gao, Y.; Ma, J.; Zhao, M.; Liu, W.; and Yuille, A. L. 2019. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3205–3214.

Godard, C.; Mac Aodha, O.; Firman, M.; and Brostow, G. J. 2019. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, 3828–3838.

Gordon, A.; Li, H.; Jonschkowski, R.; and Angelova, A. 2019. Depth From Videos in the Wild: Unsupervised Monocular Depth Learning From Unknown Cameras. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 8976–8985. IEEE Computer Society.

Grossberg, S. 1976. Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological cybernetics* 23(3): 121–134.

Grossberg, S. 2000. The complementary brain: Unifying brain dynamics and modularity. *Trends in cognitive sciences* 4(6): 233–246.

Grossberg, S. 2015. From brain synapses to systems for learning and memory: Object recognition, spatial navigation, timed conditioning, and movement control. *Brain research* 1621: 270–293.

Guizilini, V.; Hou, R.; Li, J.; Ambrus, R.; and Gaidon, A. 2020. Semantically-Guided Representation Learning for Self-Supervised Monocular Depth. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=ByxT7TNFvH.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

He, L.; Xu, X.; Lu, H.; Yang, Y.; Shen, F.; and Shen, H. T. 2017. Unsupervised cross-modal retrieval through adversarial learning. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 1153–1158. IEEE.

Henaff, O. J.; Srinivas, A.; Fauw, J. D.; Razavi, A.; Doersch, C.; Eslami, S. M. A.; and van den Oord, A. 2020. Data-Efficient Image Recognition with Contrastive Predictive Coding. URL https://openreview.net/forum?id=rJerHlrYwH.

Hu, D.; Nie, F.; and Li, X. 2019. Deep Multimodal Clustering for Unsupervised Audiovisual Learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jabri, A.; Owens, A.; and Efros, A. A. 2020. Space-Time Correspondence as a Contrastive Random Walk. *Advances in Neural Information Processing Systems* .

Jakab, T.; Gupta, A.; Bilen, H.; and Vedaldi, A. 2020. Self-supervised Learning of Interpretable Keypoints from Unlabelled Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8787–8797.

Koffka, K. 2013. *Principles of Gestalt psychology*, volume 44. Routledge.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature* 521(7553): 436–444.

Li, Y.; Zhu, J.-Y.; Tedrake, R.; and Torralba, A. 2019. Connecting Touch and Vision via Cross-Modal Prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liu, Y.; Albanie, S.; Nagrani, A.; and Zisserman, A. 2019. Use What You Have: Video retrieval using representations from collaborative experts. In *BMVC*.

Lu, Y.; Pirk, S.; Dlabal, J.; Brohan, A.; Pasad, A.; Chen, Z.; Casser, V.; Angelova, A.; and Gordon, A. 2021. Taskology: Utilizing Task Relations at Scale. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Marcu, A.; Licaret, V.; Costea, D.; and Leordeanu, M. 2020. Semantics through Time: Semi-supervised Segmentation of Aerial Videos with Iterative Label Propagation. In *Asian Conference on Computer Vision (ACCV), 2020*, 2881–2890.

Misra, I.; Shrivastava, A.; Gupta, A.; and Hebert, M. 2016. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3994–4003.

Nicolicioiu, A.; Duta, I.; and Leordeanu, M. 2019. Recurrent Space-time Graph Neural Networks. In *Advances in Neural Information Processing Systems*, 12838–12850.

Ouali, Y.; Hudelot, C.; and Tami, M. 2020. Semi-Supervised Semantic Segmentation with Cross-Consistency Training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12674–12684.

Pan, J.-Y.; Yang, H.-J.; Faloutsos, C.; and Duygulu, P. 2004. Automatic multimedia cross-modal correlation discovery. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 653–658. ACM.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32*, 8024–8035. Curran Associates, Inc.

Patrick, M.; Asano, Y.; Kuznetsova, P.; Fong, R.; Henriques, J. F.; Zweig, G.; and Vedaldi, A. 2021. Multi-modal Self-Supervision from Generalized Data Transformations. URL https://openreview.net/forum?id=mgVbI13p96.

Pearl, J. 2014. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier.

Ranjan, A.; Jampani, V.; Balles, L.; Kim, K.; Sun, D.; Wulff, J.; and Black, M. J. 2019. Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 12240–12249.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.

Rosenbaum, C.; Klinger, T.; and Riemer, M. 2018. Routing Networks: Adaptive Selection of Non-Linear Functions for Multi-Task Learning. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=ry8dvM-R-.

Ruder, S. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* .

Silberman, N.; Hoiem, D.; Kohli, P.; and Fergus, R. 2012. Indoor Segmentation and Support Inference from RGBD Images. In *ECCV*.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of Go with deep neural networks and tree search. *nature* 529(7587): 484–489.

Stekovic, S.; Fraundorfer, F.; and Lepetit, V. 2020. Casting Geometric Constraints in Semantic Segmentation as Semi-Supervised Learning. In *The IEEE Winter Conference on Applications of Computer Vision*, 1854–1863.

Sutton, R. S.; Barto, A. G.; et al. 1998. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge.

Tian, Y.; Krishnan, D.; and Isola, P. 2020. Contrastive Multiview Coding. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 776–794. Cham: Springer International Publishing. ISBN 978-3-030-58621-8.

Tosi, F.; Aleotti, F.; Ramirez, P. Z.; Poggi, M.; Salti, S.; Stefano, L. D.; and Mattoccia, S. 2020. Distilled semantics for comprehensive scene understanding from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4654–4665.

Wang, X.; Jabri, A.; and Efros, A. A. 2019. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2566–2576.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=ryGs6iA5Km.

Yang, Z.; Wang, P.; Xu, W.; Zhao, L.; and Nevatia, R. 2018. Unsupervised learning of geometry from videos with edge-aware depth-normal consistency. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Yedidia, J. S.; Freeman, W. T.; and Weiss, Y. 2005. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on information theory* 51(7): 2282–2312.

Zamir, A. R.; Sax, A.; Cheerla, N.; Suri, R.; Cao, Z.; Malik, J.; and Guibas, L. J. 2020. Robust Learning Through Cross-Task Consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11197–11206.

Zamir, A. R.; Sax, A.; Shen, W.; Guibas, L. J.; Malik, J.; and Savarese, S. 2018. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3712–3722.

Zhang, R.; Isola, P.; and Efros, A. A. 2017. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1058–1067.

Zhao, H.; Gan, C.; Rouditchenko, A.; Vondrick, C.; McDermott, J.; and Torralba, A. 2018. The sound of pixels. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 570–586.

Zhao, H.; Shi, J.; Qi, X.; Wang, X.; and Jia, J. 2017. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2881–2890.

Zhou, T.; Brown, M.; Snavely, N.; and Lowe, D. G. 2017. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1851–1858.