Disentangling Task Transfer Learning

Hongzhi Liu

Jun 22, 2018

Abstract

As we all know, object recognition, depth estimation, edge detection and pose estimation are examples of common vision tasks deemed useful and tackled by the research community. People understand that surface normals and depth are related or vanishing points in a room are useful for orientation but other relationships are less clear. Today, I read a thesis of CVPR 2018 written by Amir R. Zamir, who is from Stanford University. His team introduce a fully computational approach for modeling the structure of the space of visual tasks. This is done via finding transfer learning dependencies across a dictionary of twenty-six 2D, 2.5D, 3D and semantic tasks in a latent space. The team study the consequences of this structure, e.g. nontrivial emerged relationships and exploit them to reduce the demand for labeled data.

1. Overview of Computational Approach

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 [5] rather than the final outcome or Jean Piaget's works on developmental stages using previously learned stages as sources [2], and have extended to recent works [3, 1, 6]. Here the team make an attempt to actually find this structure.

The field of computer vision has indeed gone far without explicitly using these relationships. They have made remarkable progress by developing advanced learning machinery (*e.g.* ConvNets) capable of finding complex mappings from X to Y when many pairs of $(x, y) s.t. \quad x \in X, y \in Y$ are given as training data. This is usually referred to as fully supervised learning and often leads to problems being solved in isolation. Doing so ignores their quantifiably useful relationships leading to a massive labeled data requirement.

In this paper, Zamir and his team attempt to shed light on the underlying structure and present a framework for mapping the space of visual tasks [7]. The basis of their approach is that the team 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 exhaustively sampled, and a Binary Integer Programming formulation extracts a globally efficient transfer policy from them. They show this model leads to solving tasks with far less data than learning them independently and the resulting structure holds on common datasets.

2. Structure For Mapping the Space of Visual Tasks

The task taxonomy (taskonomy) is a computationally found directed hypergraph that captures the notion of task transferability over any given task dictionary. Zamir uses these edges to estimate the globally optimal transfer policy to solve problem.

Taxonomy is built using a four step process depicted in Fig. 1. In stage I, a task-specific network for each task in $\mathcal{S}$ is trained. In stage II, all feasible transfers between sources and targets are trained. The team include higher-order transfers which use multiple inputs task to transfer to one target. In stage III, the task affinities acquired from transfer function performances are normalized, and in stage IV, they synthesize a hypergraph which can predict the performance of any transfer policy and optimize for the optimal one.

2.1. TaskSpecific Modeling

Zamir and his team train a fully supervised task-specific network for each task in $\mathcal{S}$. Task-specific networks have an encoderdecoder architecture homogeneous across all tasks, where the encoder is large enough to extract powerful representations, and the decoder is large enough to achieve a good performance but is much smaller than the encoder.

2.2. Transfer Modeling

Given a source task s and a target task t, where $s \in \mathcal{S}$ and $s \in \mathcal{T}$, a transfer network learns a small readout function for t given a statistic computed for s as shown in Fig. 2. The statistic is the representation for image I from the encoder of $s : E_s(I)$. The readout function $(D_{s \to t})$ is parameterized

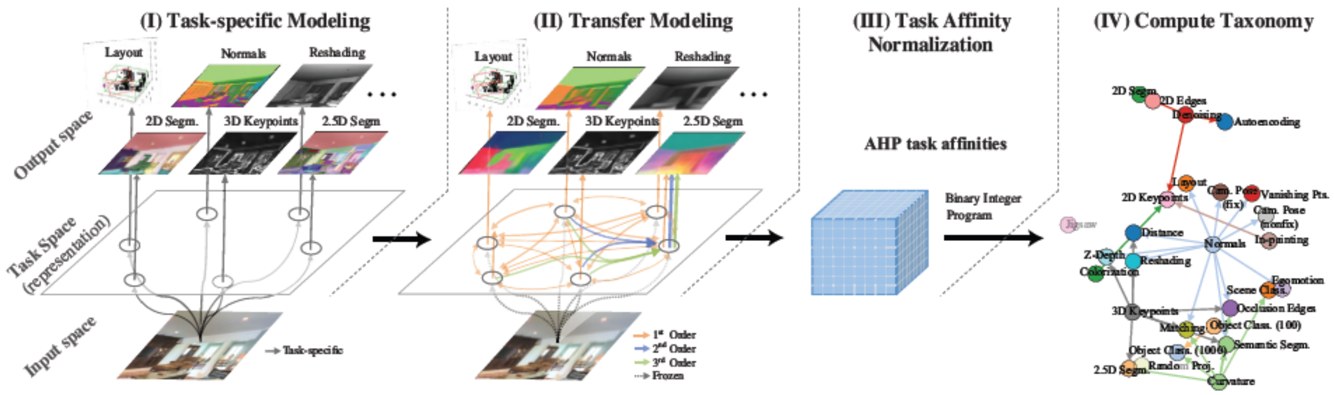

Figure 1. **Computational modeling of task relations and creating the taxonomy.** From left to right: I. Train task-specific networks. II. Train transfer functions among tasks in a latent space. III. Get normalized transfer affinities using Analytic Hierarchy Process. IV. Find global transfer taxonomy using Binary Integer Program.

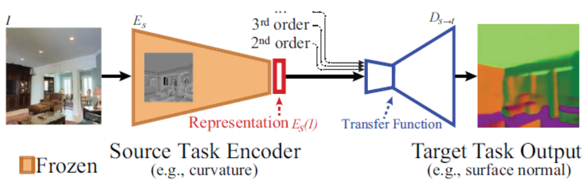

Figure 2. Transfer Function. A small readout function is trained to map representations of source task's frozen encoder to target task's labels. If order > 1, transfer function receives representations from multiple sources.

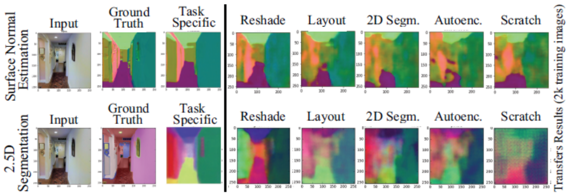

Figure 3. Transfer results to normals and 2.5D Segmentation from 5 different source tasks. The spread in transferability among different sources is apparent with reshading among top-performing ones in this case. Task-specific networks were trained on 60x more data. "Scratch" was trained from scratch without transfer learning.

by $\theta_{s\to t}$ minimizing the loss L_t as Eq. 1:

$$D_{s\to t} := arg\min_\theta \mathbb{E}_{I\in D}[L_t(D_\theta(E_s(I)), f_t(I))]. \qquad (1)$$

where $f_t(I)$ is ground truth of t for image I. $E_s(I)$ may or may not be sufficient for solving t depending on the relation between t and s as shown in Fig. 3. Thus, the performance of $(D_{s\to t})$ is a useful metric as task affinity. The team train transfer functions for all feasible source-target combinations.

2.3. Ordinal Normalization using AnalyticHierarchy Process

The team use an ordinal approach in which the output quality and loss are only assumed to change monotonically. For each t, they construct W_t a pairwise tournament matrix between all feasible sources for transferring to t. The element at (i, j) is the percentage of images in a held-out test set, D_{test}, on which s_i transfered to t better than s_j did.

This approach is derived from Analytic Hierarchy Process [4], a method widely used in operations research to create a total order based on multiple pairwise comparisons.

2.4. Computing the Global Taxonomy

Given the normalized task affinity matrix, Zamir needs to devise a global transfer policy which maximizes collective performance across all tasks while minimizing the used supervision. This problem can be formulated as subgraph selection where tasks are nodes and transfers are edges. The optimal subgraph picks the ideal source nodes and the best edges from these sources to targets while satisfying that the number of source nodes does not exceed the supervision budget.

The team solve this subgraph selection problem using Boolean Integer Programming (BIP), described below which can be solved optimally and efficiently. The canonical form for a BIP is: maximize $c^T x$, subject to $Ax \preceq b$ and $x \in 0, 1$. The elements of A not defined above are set to 0. The problem is now a valid BIP and can be optimally solved in a fraction of a second. The BIP solution $\hat{x}$ corresponds to the optimal subgraph which is their taxonomy.

Table 1. Task-Specific Networks'Sanity: Win rates vs. random (Gaussian) network representation readout and statistically informed guess avg.

Task	avg	rand	Task	avg	rand	Task	avg	rand
Denoising	100	99.9	Layout	99.6	89.1	Scene Class.	97.0	93.4
Autoenc.	100	99.8	2D Edges	100	99.9	Occ. Edges	100	95.4
Reshading	94.9	95.2	Pose (fix)	76.3	79.5	Pose (nonfix)	60.2	61.9
Inpainting	99.9	-	2D Segm.	97.7	95.7	2.5D Segm.	94.2	89.4
Curvature	78.7	93.4	Matching	86.8	84.6	Egomotion	67.5	72.3
Normals	99.4	99.5	Vanishing	99.5	96.4	2D Keypnt.	99.8	99.4
Z-Depth	92.3	91.1	Distance	92.4	92.1	3D Keypnt.	96.0	96.9
Mean	92.4	90.9						

3. Experiments of the Approach

Zamir and his team have evaluated the results of the trained task-specific networks. Tab. 1 provides win rates of the taskspecifc networks vs. two baselines. Visual outputs for a random test sample. Win rate (%) is the proportion of test set images for which a baseline is beaten. The high win rates in Tab. 1 and qualitative results show the networks are well trained and stable and can be relied upon for modeling the task space. See results of applying the networks on a YouTube video frame-by-frame here. A live demo for user uploaded queries is available here.

References

[1] C. Doersch and A. Zisserman. Multi-task self-supervised visual learning. In *ICCV*, 2017. 1

[2] A. Gopnik, C. Glymour, D. M. Sobel, L. E. Schulz, T. Kushnir, and D. Danks. A theory of causal learning in children: Causal maps and bayes nets. *Psychological Review*, 2004. 1

[3] S. R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In *ICCV*, 2017. 1

[4] R. W. Saaty. The analytic hierarchy process—what it is and how it is used. *Mathematical Modelling*, 1987. 2

[5] A. M. Turing. Computing machinery and intelligence. *Mind*, 1950. 1

[6] X. Wang, K. He, and A. Gupta. Transitive invariance for self-supervised visual representation learning. *arXiv preprint arXiv:1708.02901*, 2017. 1

[7] A. Zamir, A. Sax, W. Shen, and L. Guibas. Taskonomy: Disentangling task transfer learning. In *CVPR*, 2018. 1