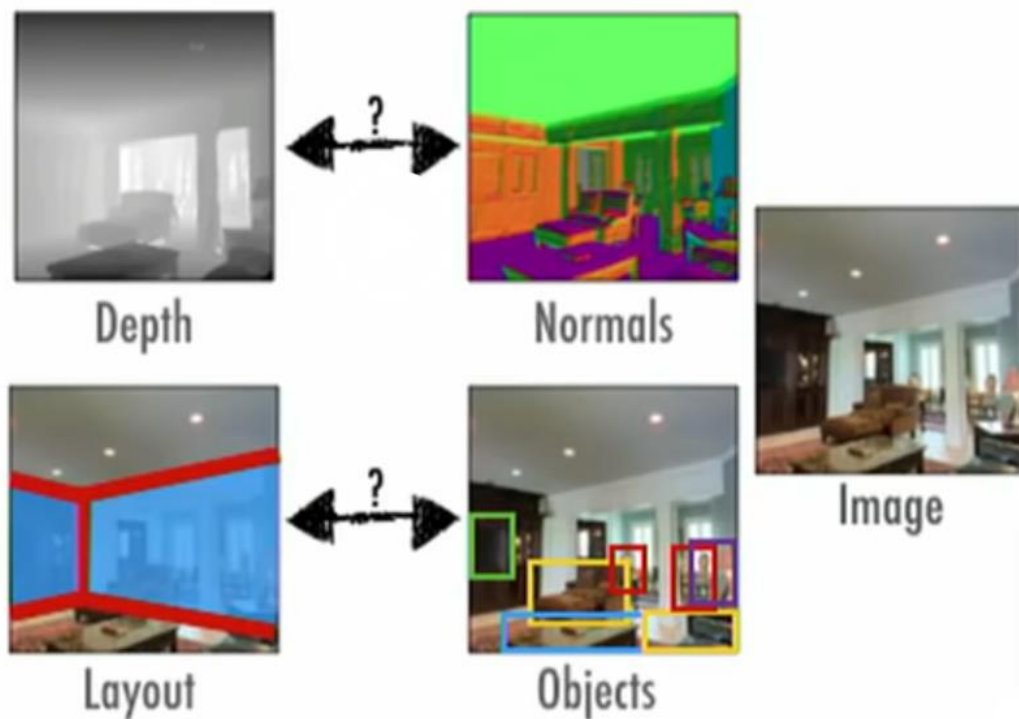


The graph illustrates the dependencies between various components in a 3D reconstruction pipeline. The components are represented as nodes, and the arrows indicate the flow of information or dependencies between them.

- 3D Keypoints** (grey node) is a central hub that depends on **Colorization** (green node) and **Z-Depth** (light blue node). It also provides input to **Distance** (dark blue node), **Reshading** (teal node), **Matching** (yellow node), **Object Class. (1000)** (blue node), **2.5D Segm.** (orange node), and **Curvature** (green node).
- Colorization** (green node) depends on **Z-Depth** (light blue node).
- Z-Depth** (light blue node) depends on **Distance** (dark blue node).
- Distance** (dark blue node) depends on **2D Keypoints** (pink node).
- 2D Keypoints** (pink node) depends on **Layout** (orange node) and **Cam. Pose (fix)** (brown node).
- Layout** (orange node) depends on **Cam. Pose (fix)** (brown node).
- Cam. Pose (fix)** (brown node) depends on **Cam. Pose (nonfix)** (grey node).
- Cam. Pose (nonfix)** (grey node) depends on **Vanishing Pts.** (red node).
- Vanishing Pts.** (red node) depends on **In-painting** (brown node).
- In-painting** (brown node) depends on **Normals** (blue node).
- Normals** (blue node) is a central hub that depends on **Distance** (dark blue node), **Reshading** (teal node), **Layout** (orange node), **Cam. Pose (fix)** (brown node), **Cam. Pose (nonfix)** (grey node), **Vanishing Pts.** (red node), **In-painting** (brown node), **Egomotion** (orange node), **Scene Class.** (purple node), **Occlusion Edges** (purple node), **Matching** (yellow node), **Semantic Segm.** (green node), and **Curvature** (green node).
- Egomotion** (orange node) depends on **Scene Class.** (purple node).
- Scene Class.** (purple node) depends on **Occlusion Edges** (purple node).
- Occlusion Edges** (purple node) depends on **Matching** (yellow node) and **Semantic Segm.** (green node).
- Matching** (yellow node) depends on **Object Class. (1000)** (blue node) and **Random Proj.** (light green node).
- Object Class. (1000)** (blue node) depends on **Random Proj.** (light green node).
- Random Proj.** (light green node) depends on **2.5D Segm.** (orange node) and **Curvature** (green node).
- 2.5D Segm.** (orange node) depends on **Curvature** (green node).
- Curvature** (green node) depends on **Semantic Segm.** (green node).
- Semantic Segm.** (green node) depends on **Object Class. (1000)** (blue node) and **Random Proj.** (light green node).

[illegible]

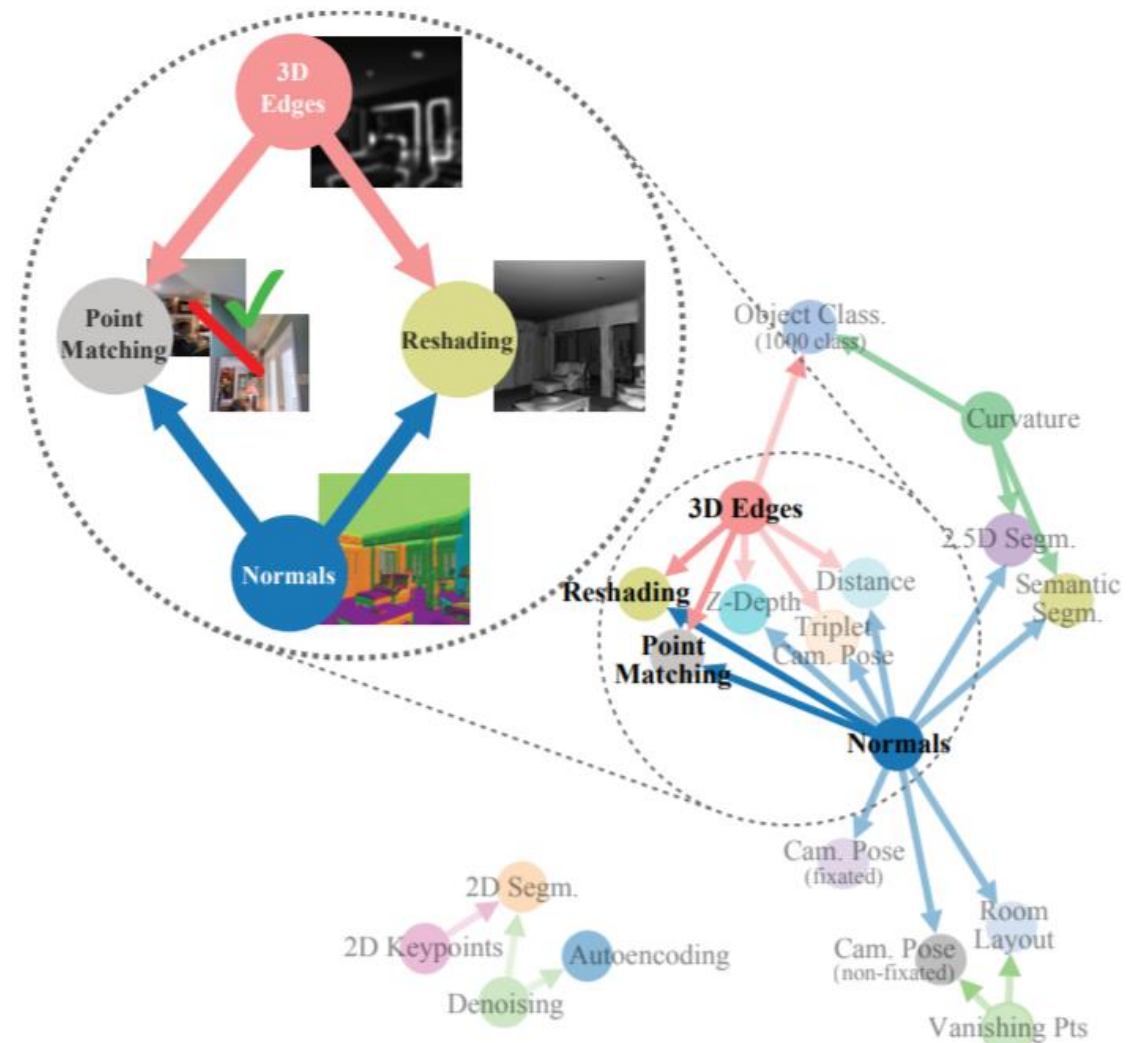
Задачи компьютерного зрения связаны или независимы?



- ✓ Интуиция: среди задач есть некоторая структура
- ✓ Эта структура может быть полезна для решения сразу нескольких задач без повышения общей сложности

Transfer learning

- ✓ Вычислительно найти структуру с помощью нейронных сетей (не учитывая никакие предварительные знания)
- ✓ Использовать скрытое представление данных в одной задаче для решения другой



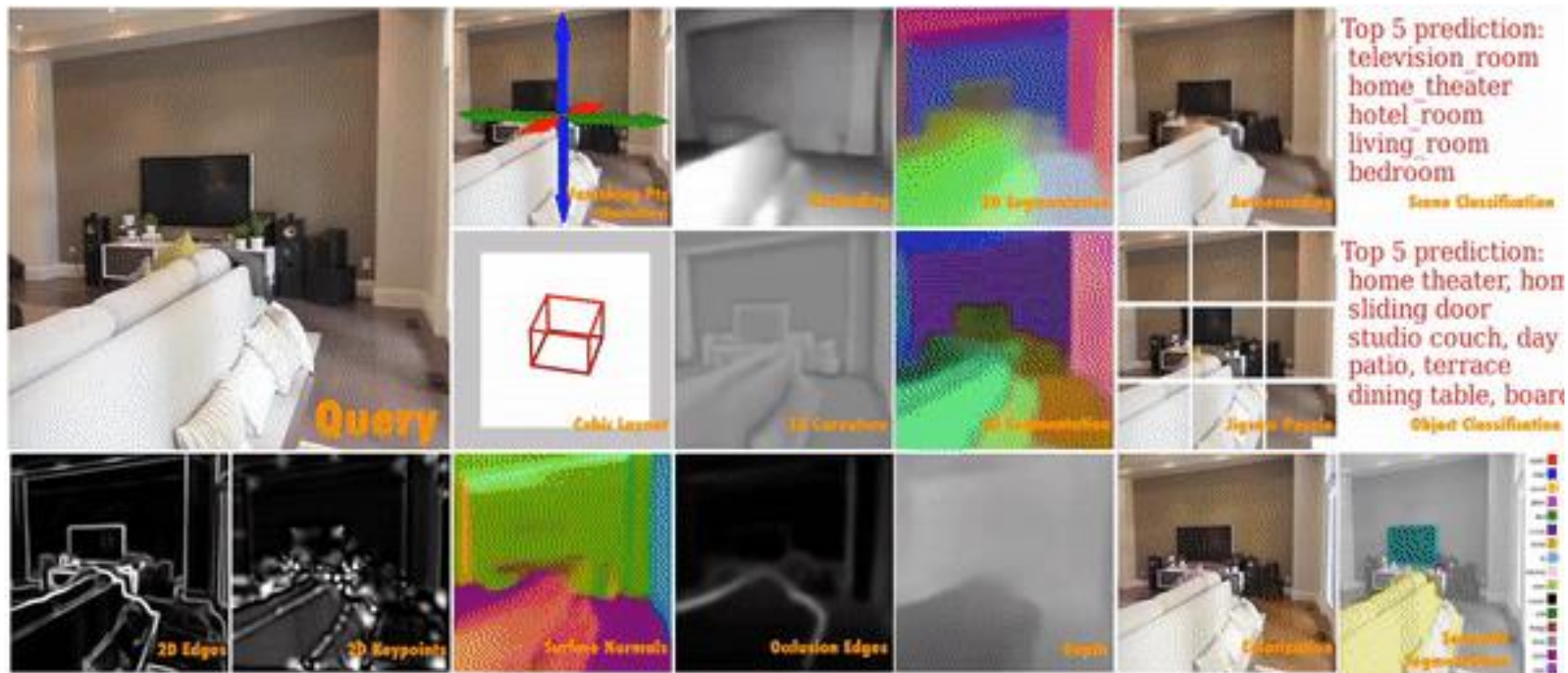
Связанные исследовавшиеся темы

- 1) ***Self-supervised learning*** использует для решения дорогого object detection дешевое colorization
- 2) ***Unsupervised learning*** строит компактные представления, независимые от следующей задачи
- 3) ***Meta-learning*** – обучение происходит на уровень выше того, где происходит обычное обучение. Сходство с нашим подходом: строим мета-структуру пространства задач
- 4) ***Multi-task learning*** – несколько выходных данных за один прогон
- 5) ...

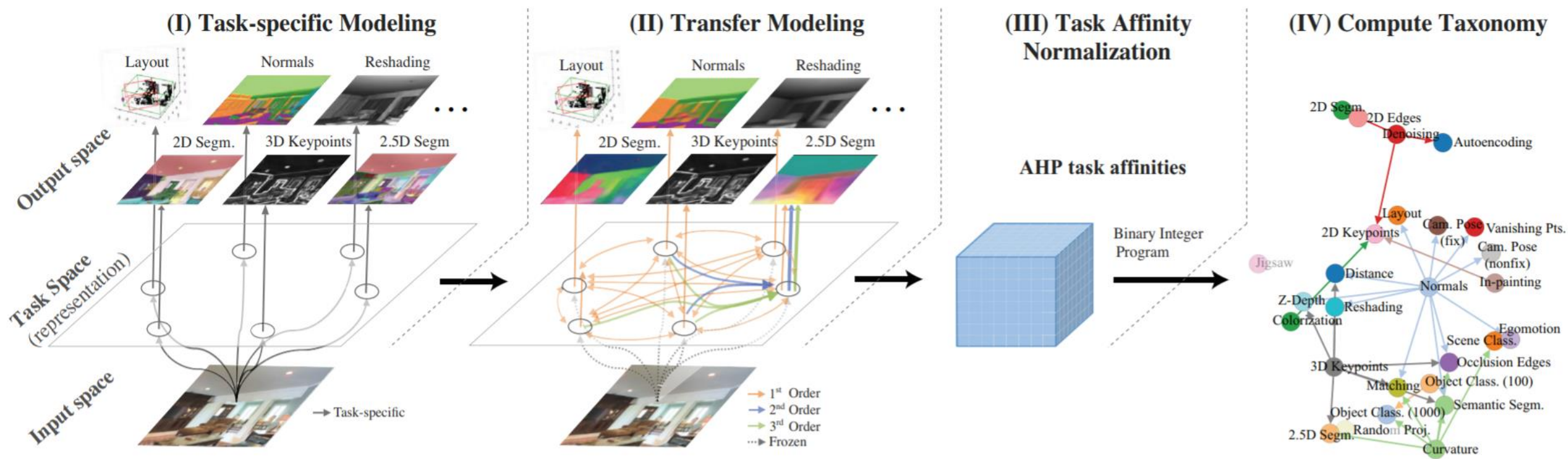
Словарь задач

Множество целевых задач $T = \{t_1, t_2, \dots, t_n\}$; исходных S .

Словарь задач $V = T \cup S$; $T - T \cup S$ – target-only; $S - T \cup S$ – source-only

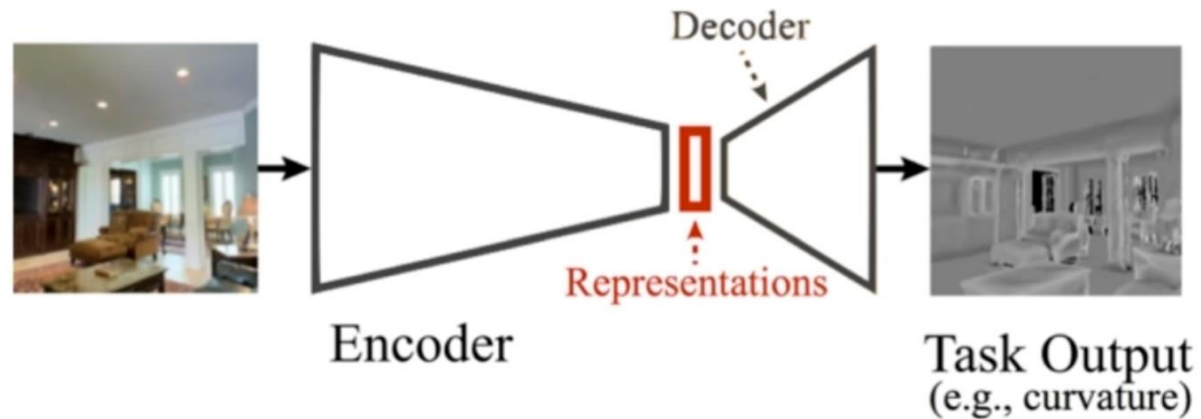


Метод нахождения структуры задач



1 шаг. Task-Specific Modeling

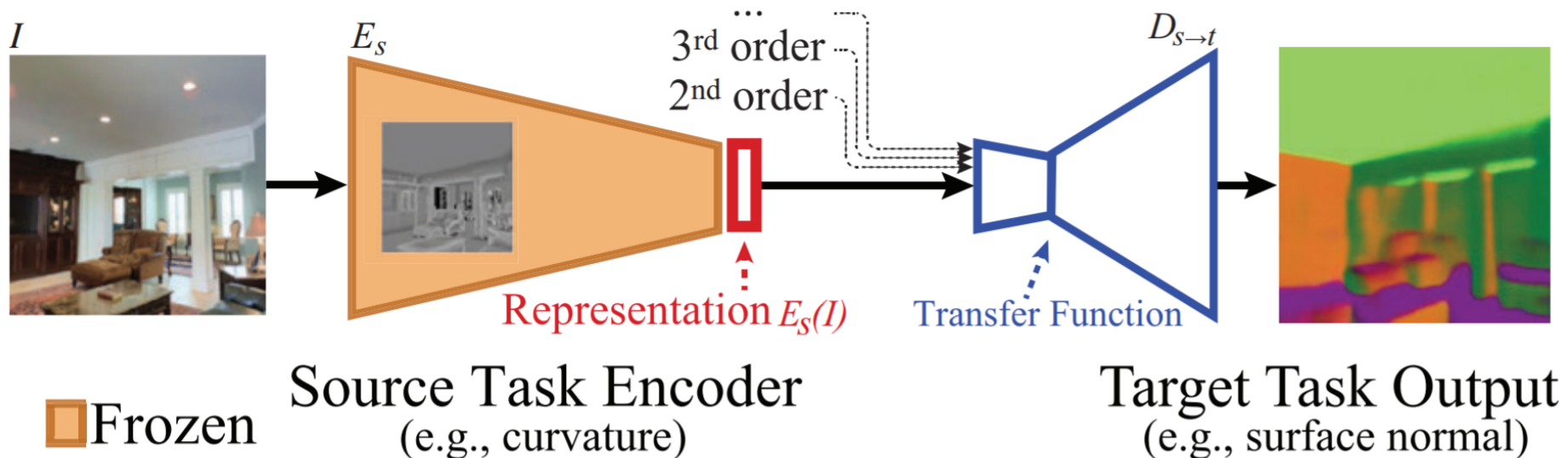
- Обучение сетей для каждой задачи в S
- Сети имеют архитектуру энкодер-декодер



2 шаг. Transfer Modeling

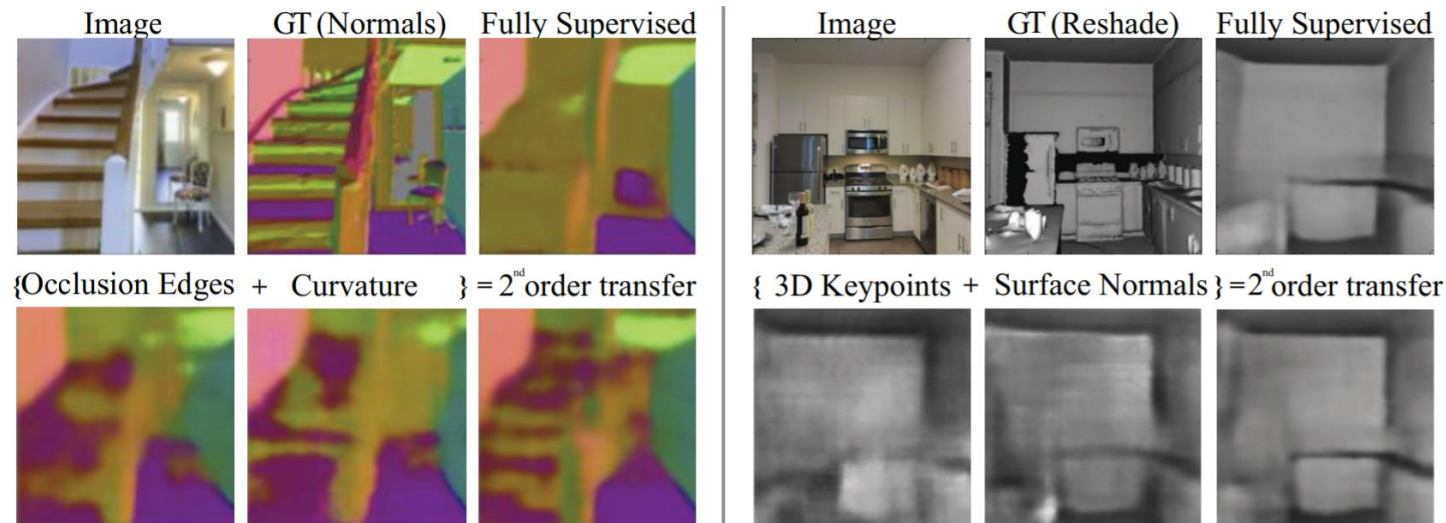
Transfer network по заданной исходной задаче $s \in S$ и целевой задаче $t \in T$, учит функцию $D_{s \rightarrow t}$ по статистике, полученной для s :

$$D_{s \rightarrow t} = \operatorname{argmin}_{\theta} \mathbb{E}_{I \in \mathcal{D}} [L_t(D_{\theta}(E_s(I)), f_t(I))]$$



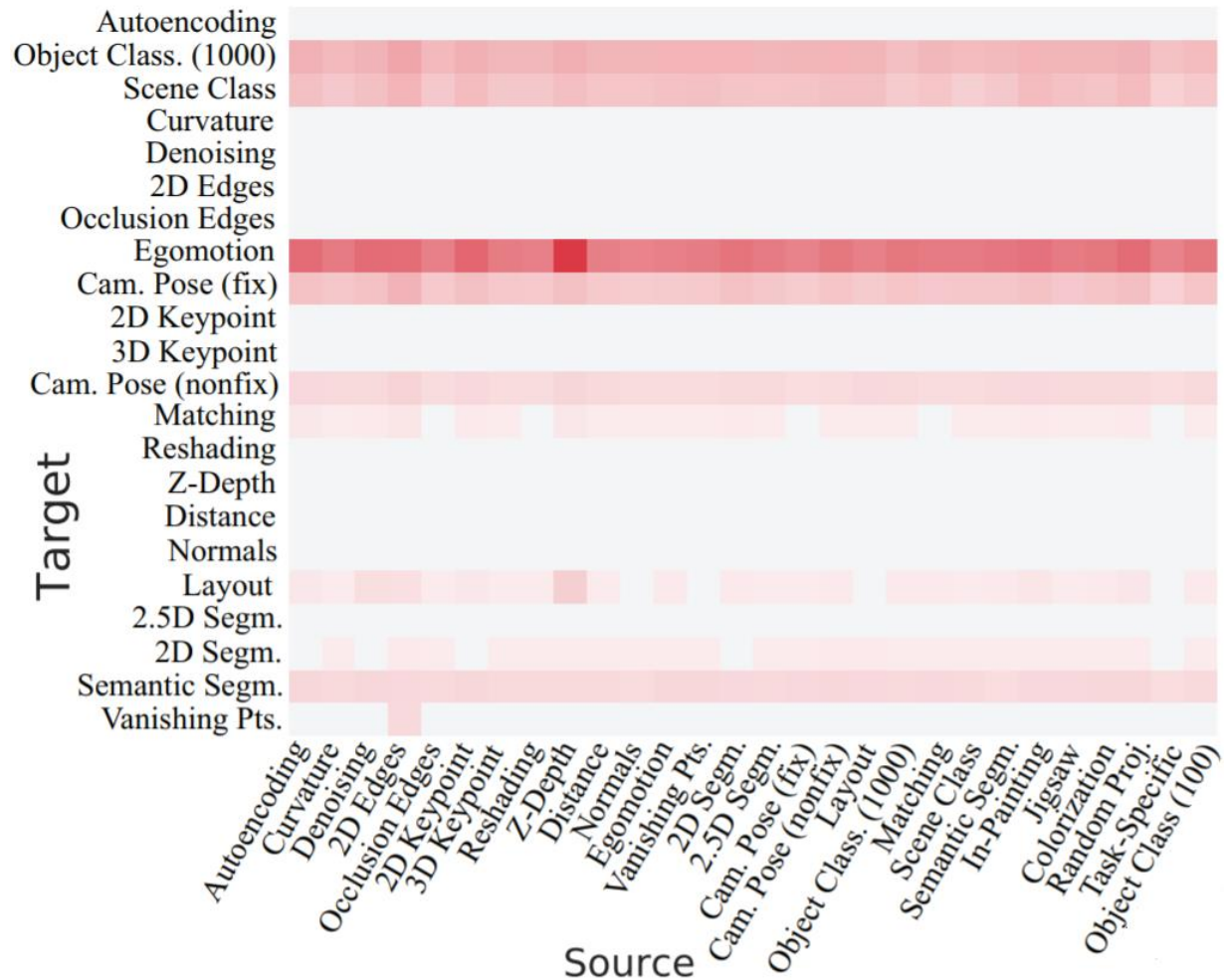
2 шаг. Transfer Modeling

- Accessibility: скрытое представление содержит достаточно ценной информации и к ней легко получить доступ (легко извлекаема)
- Higher-Order Transfers: трансфер-функции $D: \wp(S) \rightarrow T$



- Transitive Transfers: $s \rightarrow t_1 \rightarrow t_2$

3 шаг. Нормализация с использованием Analytic Hierarchy Process (АНП)



- Для каждой задачи t создается матрица W_t

- $W_t(i, j)$ - % изображений, где $D_{s_i \rightarrow t} < D_{s_j \rightarrow t}$

- $W'_t = W_t / W_t^T$

$$w'_{i,j} = \frac{\mathbb{E}_{I \in \mathcal{D}_{test}} [D_{s_i \rightarrow t}(I) < D_{s_j \rightarrow t}(I)]}{\mathbb{E}_{I \in \mathcal{D}_{test}} [D_{s_i \rightarrow t}(I) > D_{s_j \rightarrow t}(I)]}$$

- $T_{s_i \rightarrow t}$ — i -я компонента главного собственного вектора матрицы W'_t



First-order task affinity matrix (P)

4 шаг. Вычисление глобальной таксономии

- Решается при помощи целочисленного (бинарного) программирования

- Индексация: i – ребро (трансфер); j – целевая задача

- Операторы
$$\begin{aligned} (\{s_1^i, \dots, s_{m_i}^i\}, t^i) &\xrightarrow{\text{sources}} \{s_1^i, \dots, s_{m_i}^i\} \\ (\{s_1^i, \dots, s_{m_i}^i\}, t^i) &\xrightarrow{\text{target}} t^i. \end{aligned}$$

- Параметры: бюджет γ ; матрица P . Опционально: r_j – важность целевой задачи; l_i – сложность разметки
- x – бинарный, индикатор source-вершин и взятых трансферов

Boolean Integer Programming (BIP)

$$\begin{aligned} &\text{maximize } c^T x, \\ &\text{subject to } Ax \preceq b \\ &\quad \text{and } x \in \{0, 1\}^{|E|+|\mathcal{V}|} \end{aligned}$$

$$c_i := r_{\text{target}(i)} \cdot p_i$$

2) каждая целевая задача имеет ровно один входящий трансфер:

$$a_{|E|+j,i} := 2 \cdot \mathbb{1}_{\{\text{target}(i)=j\}},$$

$$b_{|E|+j} := -1$$

Ограничения

1) если трансфер включен в подграф, то все его исходные задачи (узлы), также должны быть включены:

$$a_{i,k} = \begin{cases} |\text{sources}(i)| & \text{if } k = i \\ -1 & \text{if } (k - |E|) \in \text{sources}(i) \\ 0 & \text{otherwise} \end{cases}$$

$$b_i = 0.$$

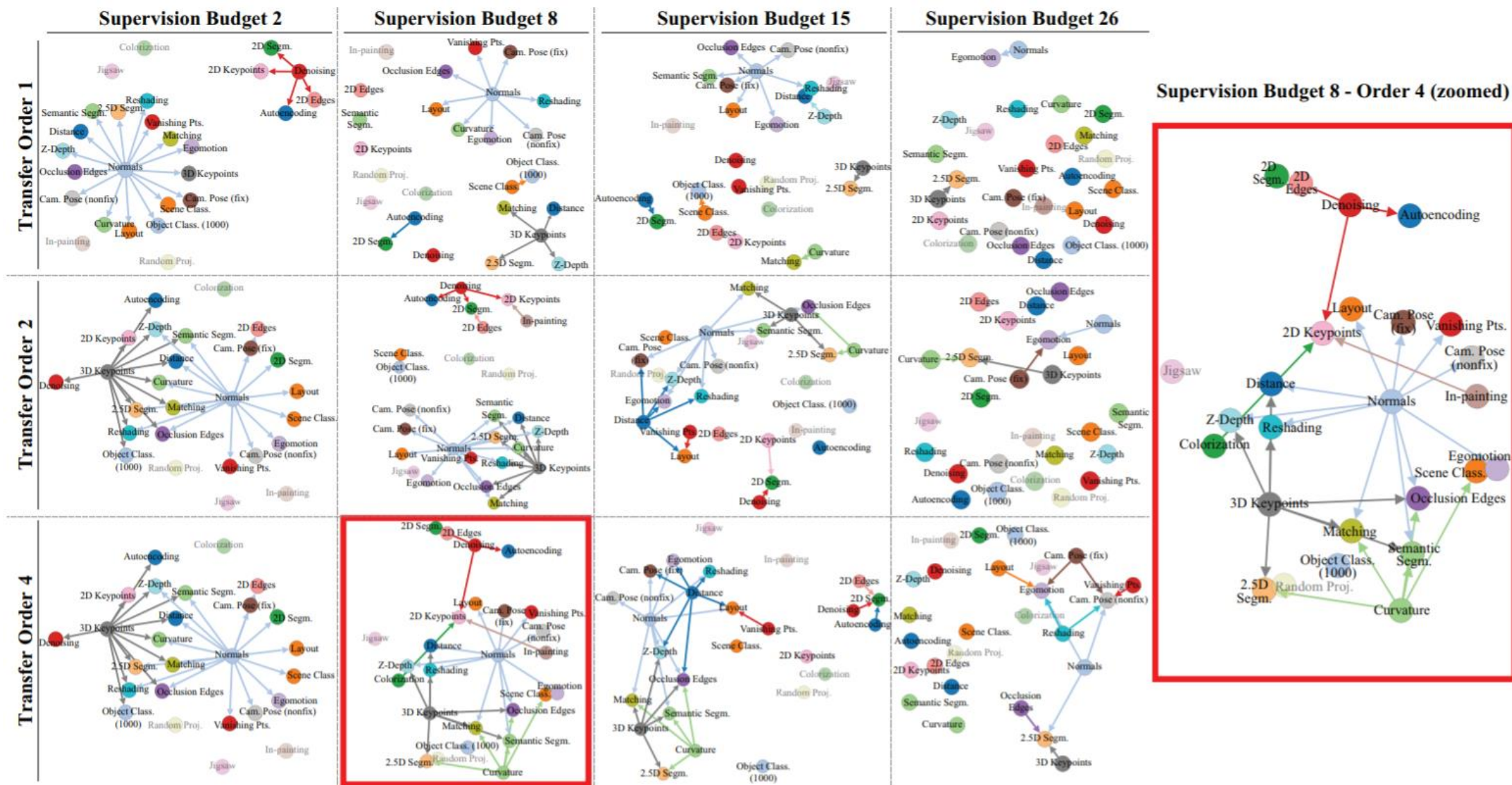
3) не превышен бюджет:

$$a_{|E|+|\mathcal{V}|+1,i} := \ell_i, \quad b_{|E|+|\mathcal{V}|+1} := \gamma$$

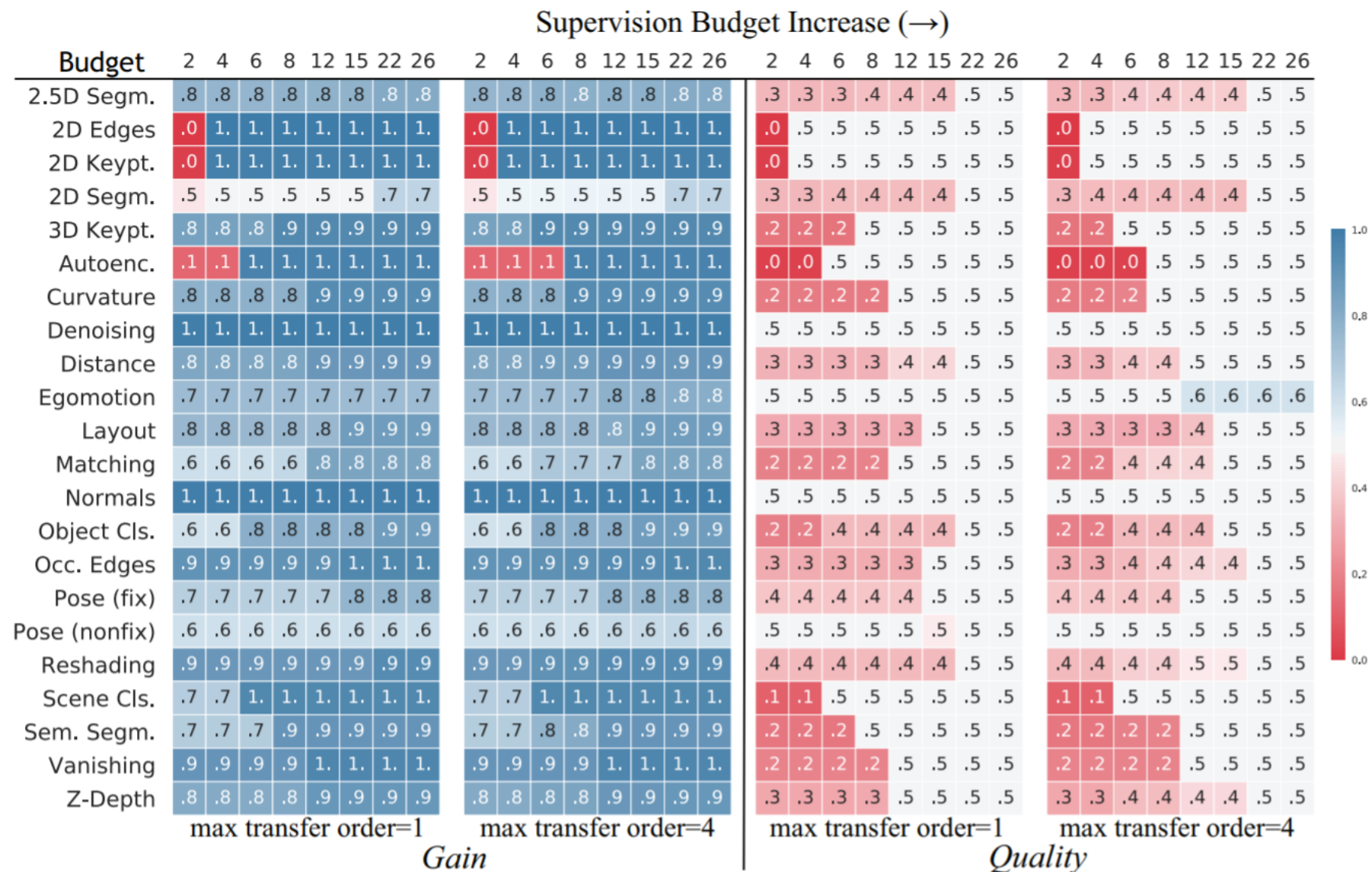
Эксперименты

- 4 source-only задачи (colorization, jigsaw puzzle, in-painting, random projection)
- ~3000 функций передачи
- 47 886 GPU-часов в облаке
- Энкодер – fully-convolutional ResNet50
- Декодер – зависит от задачи

Оценка вычисленных таксономий



Оценка вычисленных таксономий



Обобщение на новые задачи

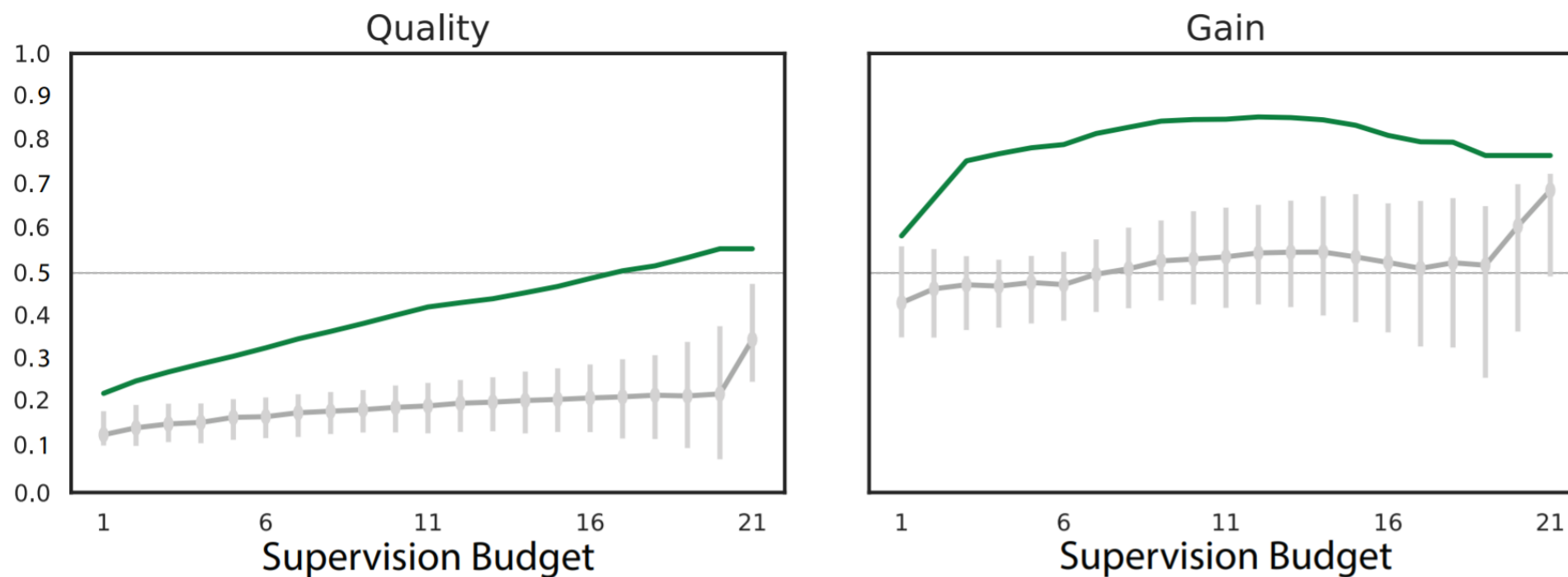
Решение новой задачи по схеме all-for-one

Order Increase (→)

Order	1	2	3	4	1	2	3	4
2.5D Segm.	.1	.1	.1	.9	.0	.0	.0	.0
2D Edges	.8	.8	.8	.8	.4	.4	.4	.4
2D Keypt.	.8	.8	.9	.9	.2	.2	.2	.2
2D Segm.	.8	.9	.9	.9	.2	.3	.3	.3
3D Keypt.	.0	.1	.9	.9	.0	.0	.0	.0
Autoenc.	.9	1.	1.	1.	.0	.0	.0	.0
Curvature	.9	.9	.9	.9	.3	.4	.4	.4
Denoising	.7	.8	.8	.8	.5	.6	.6	.6
Distance	.7	.7	.7	.7	.4	.4	.4	.4
Egomotion	.0	.9	.9	.9	.0	.0	.0	.0
Layout	.9	.9	.9	.9	.1	.2	.2	.2
Matching	.6	.6	.6	.6	.5	.5	.5	.5
Normals	.6	.7	.7	.7	.2	.4	.4	.5
Object Cls.	.9	.9	.9	.9	.4	.5	.5	.5
Occ. Edges	.9	.9	.9	.9	.4	.4	.4	.4
Pose (fix)	.9	.9	.9	.9	.4	.5	.5	.5
Pose (nonfix)	.9	1.	1.	1.	.0	.0	.1	.1
Reshading	.8	.8	.8	.8	.3	.3	.4	.4
Scene Cls.	.8	.8	.8	.8	.4	.4	.4	.4
Sem. Segm.	.5	.5	.5	.5	.4	.4	.4	.4
Vanishing	.7	.8	.8	.8	.2	.2	.2	.2
Z-Depth	.9	.9	.9	.9	.2	.2	.2	.2

Task	scratch	ImageNet[51]	Wang.[96]	Agrawal.[1]	Zamir.[100]	Zhang.[103]	Noroozi.[68]	full sup.	Taxonomy
Depth	88	88	93	89	88	84	86	43	-
	.03	.04	.04	.03	.04	.03	.03	.02	.02
Scene Cls.	80	52	83	74	74	71	75	15	-
	3.30	2.76	3.56	3.15	3.17	3.09	3.19	2.23	2.63
Sem. Segm.	78	79	82	85	76	78	84	21	-
	1.74	1.88	1.92	1.80	1.85	1.74	1.71	1.42	1.53
Object Cls.	79	54	82	76	75	76	76	34	-
	4.08	3.57	4.27	3.99	3.98	4.00	3.97	3.26	3.46
Normals	97	98	98	98	98	97	97	6	-
	.22	.30	.34	.28	.28	.23	.24	.12	.15
2.5D Segm.	80	93	92	89	90	84	87	40	-
	.21	.34	.34	.26	.29	.22	.24	.16	.17
Occ. Edges	93	96	95	93	94	93	94	42	-
	.16	.19	.18	.17	.18	.16	.17	.12	.13
Curvature	88	94	89	85	88	92	88	29	-
	.25	.28	.26	.25	.26	.26	.25	.21	.22
Egomotion	79	78	83	77	76	74	71	59	-
	8.60	8.58	9.26	8.41	8.34	8.15	7.94	7.32	6.85
Layout	80	76	85	79	77	78	70	36	-
	.66	.66	.85	.65	.65	.62	.54	.37	.41

Насколько важно использовать таксономию?



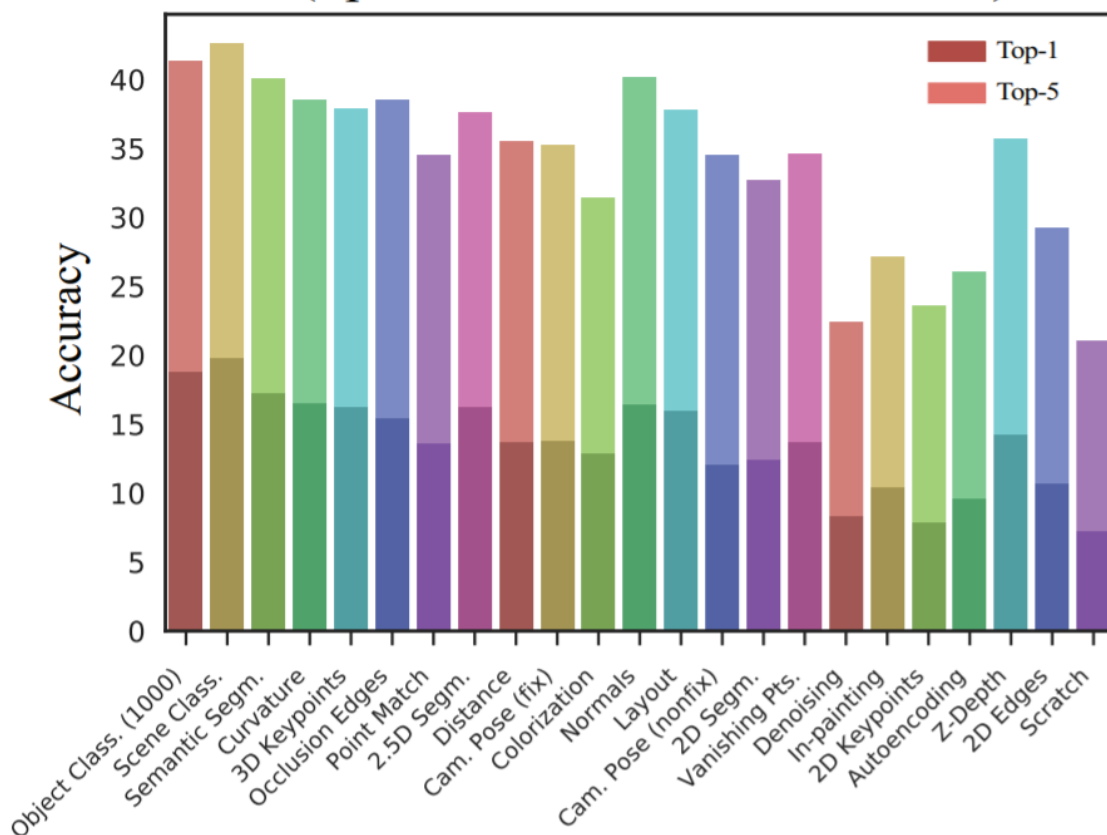
Зеленый – наша таксономия

Серый – случайная с заданным бюджетом

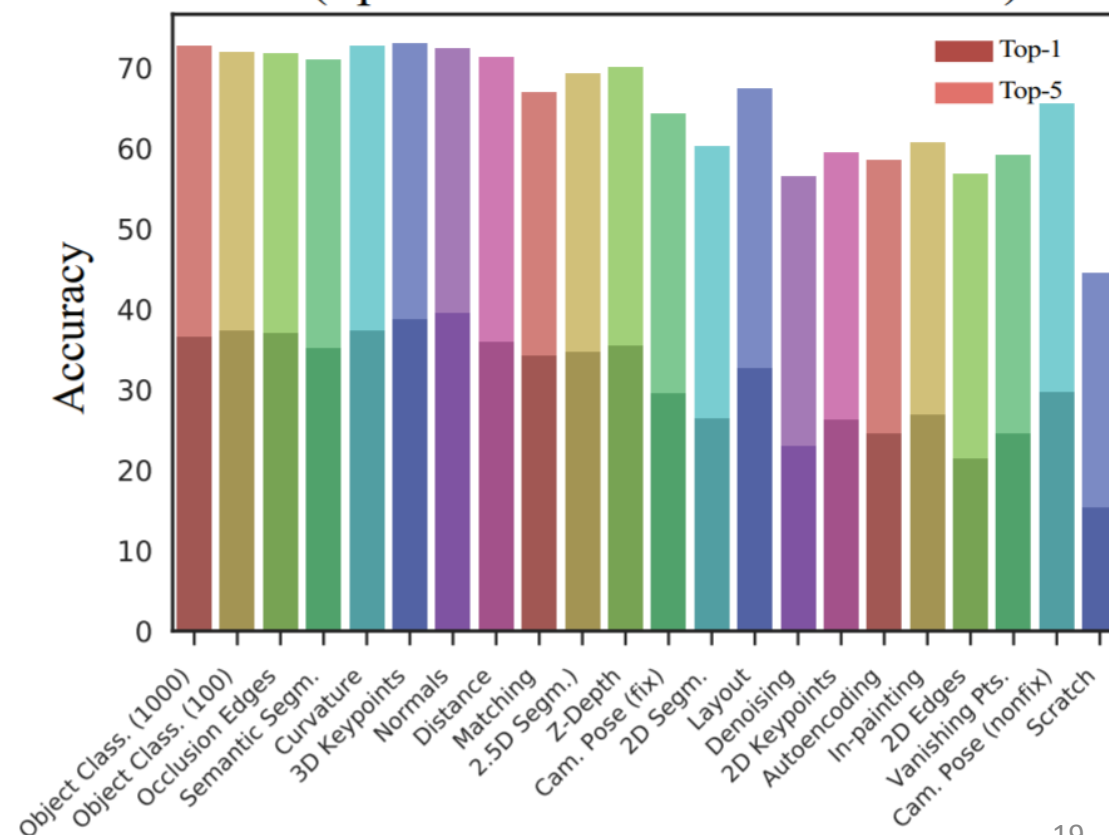
Смена датасета на MIT Places & ImageNet

Способность к трансферу обучения была ранжирована на исходном датасете

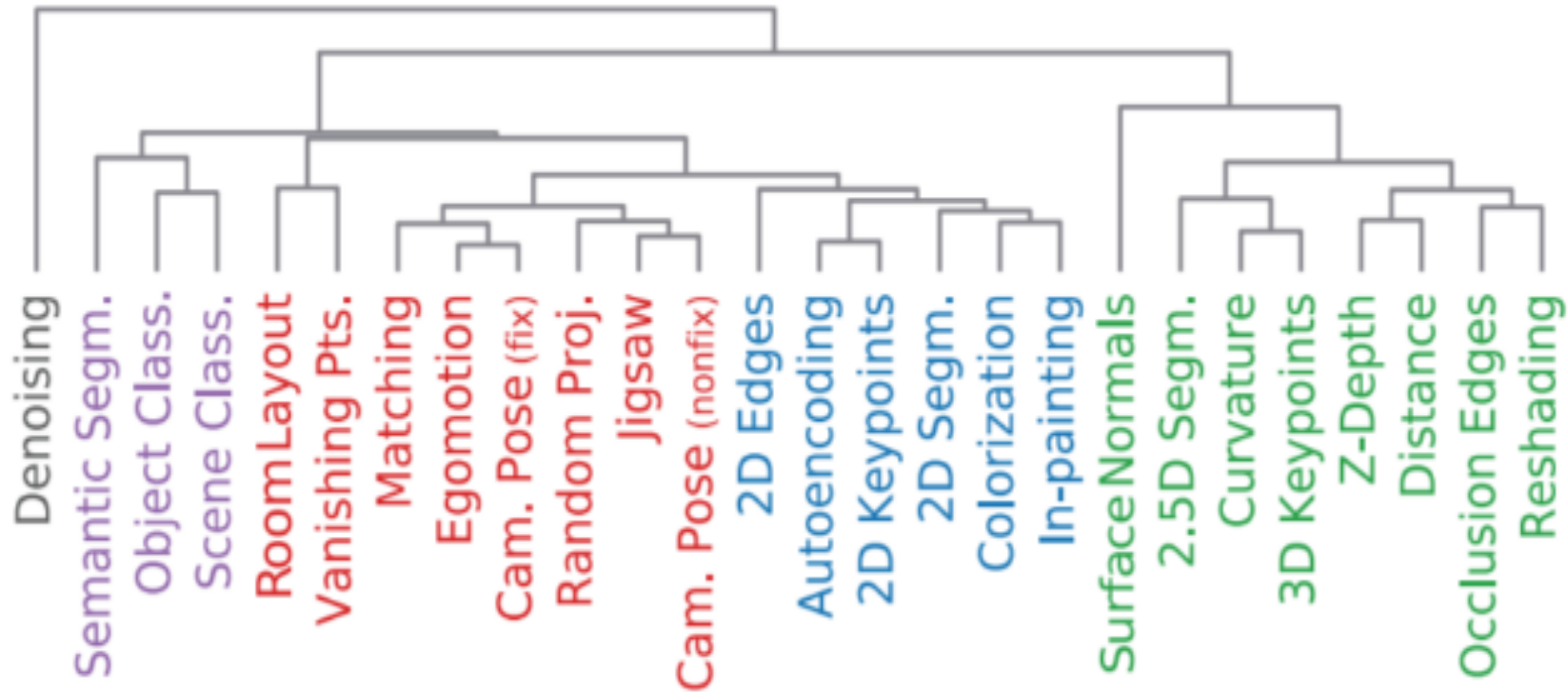
Transferring to ImageNet
(Spearman's correlation = 0.823)



Transferring to MIT Places
(Spearman's correlation = 0.857)



Дерево схожести задач



3D, 2D, low dimensional geometric, semantic

оказались собраны вместе чисто вычислительным подходом

Дальнейшее изучение

- 1) **Model Dependence** – не гарантировано, что результат не зависит от модели и данных
- 2) **Compositionality** – задачи определялись человеком. Можно попробовать их разбить на маленькие подзадачи
- 3) **Space Regularity** – выбор словаря с возможностью переносимости модели на новые задачи
- 4) **Transferring to Non-visual and Robotic Tasks** – решение задач, которые не являются визуальными, но влекут их решение
- 5) **Lifelong Learning** – механизм расширения на основе получаемых знаний

Материалы

1. <https://arxiv.org/pdf/1804.08328.pdf> – Amir Zamir, Alexander Sax, William Shen, Leonidas Guibas, Jitendra Malik, Silvio Savarese: Taskonomy: Disentangling Task Transfer Learning. In 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2018
2. http://taskonomy.stanford.edu/taskonomy_supp_CVPR2018.pdf – Amir Zamir, Alexander Sax, William Shen, Leonidas Guibas, Jitendra Malik, Silvio Savarese: Taskonomy: Disentangling Task Transfer Learning Supplementary Material, 2018
3. <https://youtu.be/9mdCWMVAMLg> – презентация на CVPR2018
4. <http://taskonomy.stanford.edu> – сайт с исследованием и api