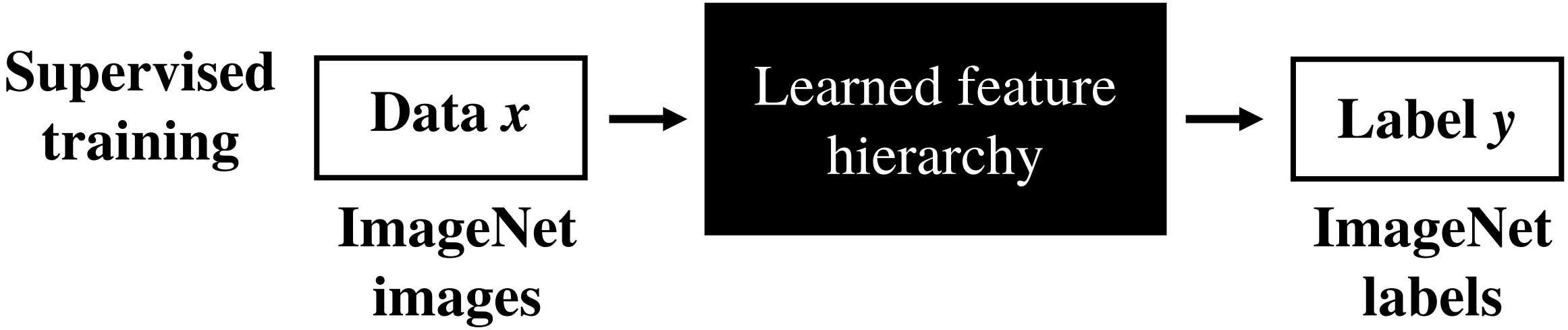
Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery.

To Appear at CVPR 2018

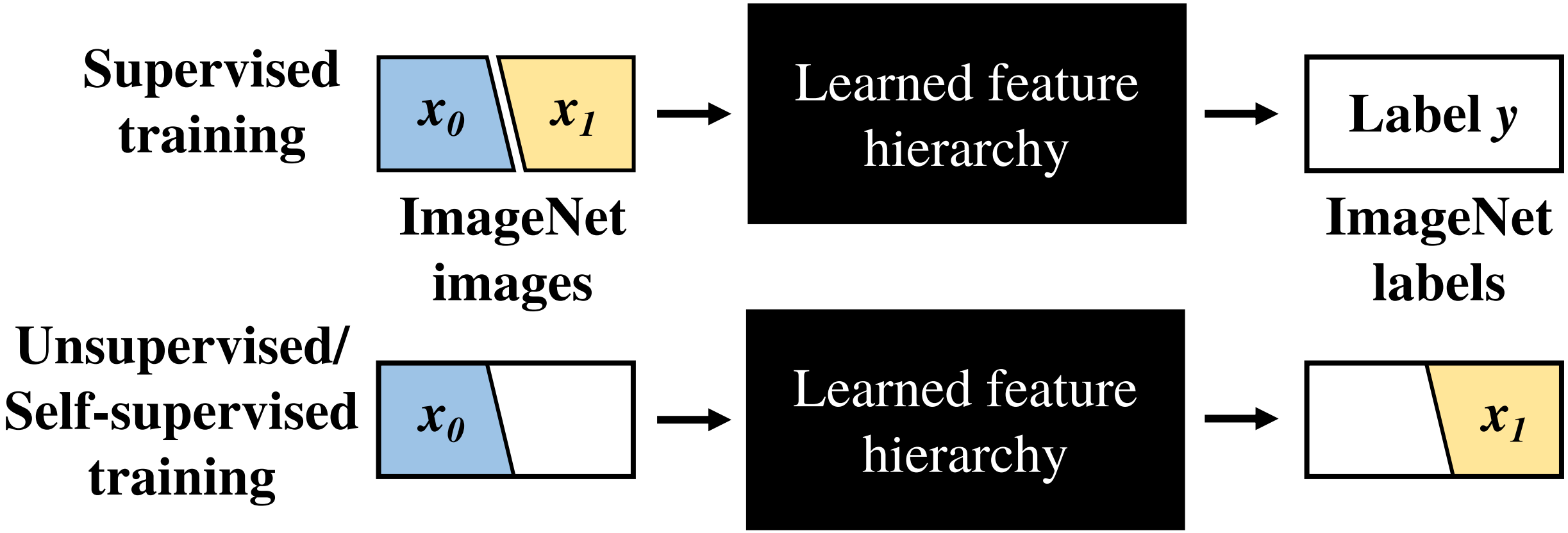
Zhongzheng (Jason) Ren, Yong Jae Lee
UC Davis

Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery.

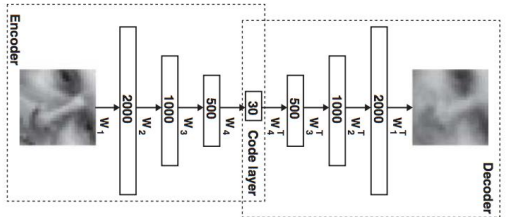
Predicting Labels from Data



Predicting Data from Data

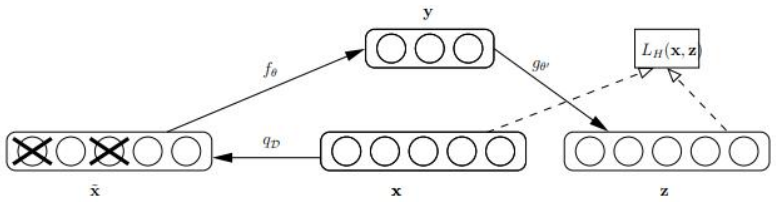


Autoencoders



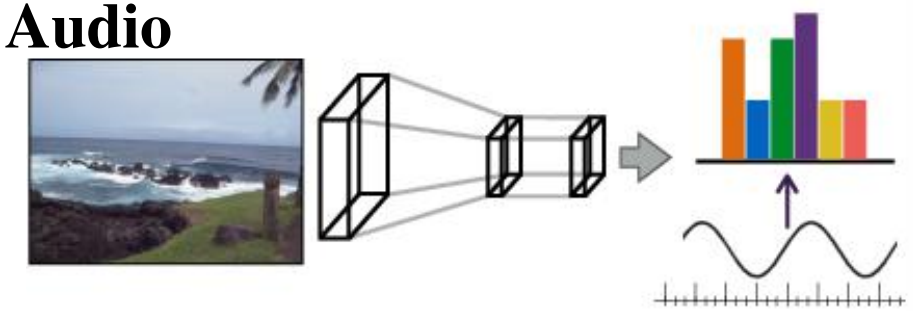
Hinton & Salakhutdinov. Science 2006.

Denoising Autoencoders



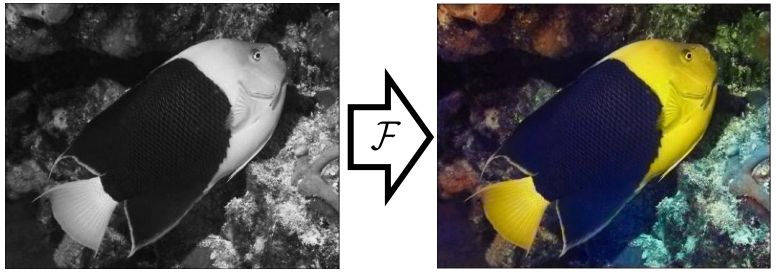
Vincent *et al.* ICML 2008.

Audio



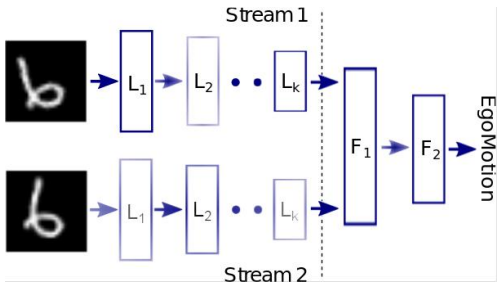
Owens *et al.* CVPR 2016, ECCV 2016

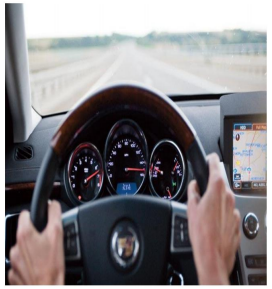
Colorization

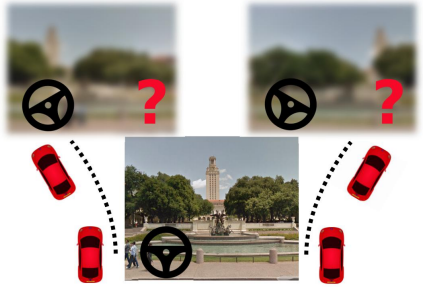


Zhang *et al.* ECCV 2016.

Egomotion

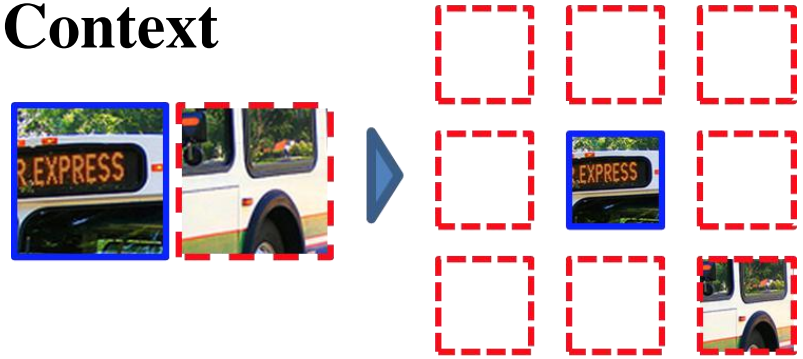






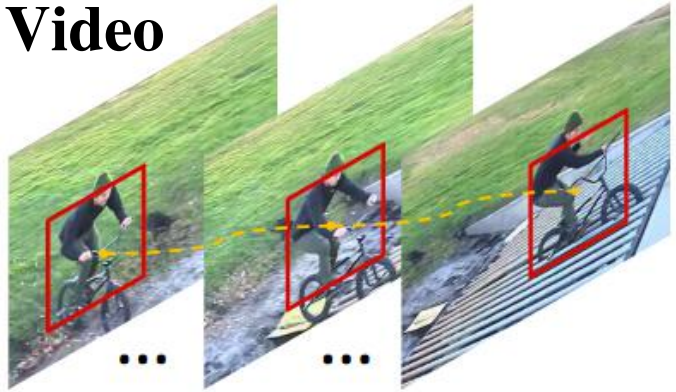
Agrawal *et al.* ICCV 2015 Jayaraman *et al.* ICCV 2015

Context



Doersch *et al.* ICCV 2015 Pathak *et al.* CVPR 2016

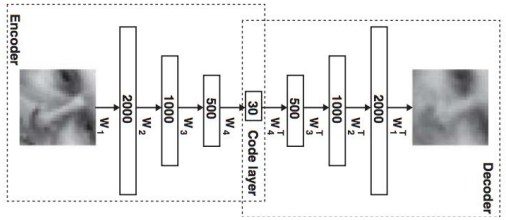
Video



Wang *et al.* ICCV 2015

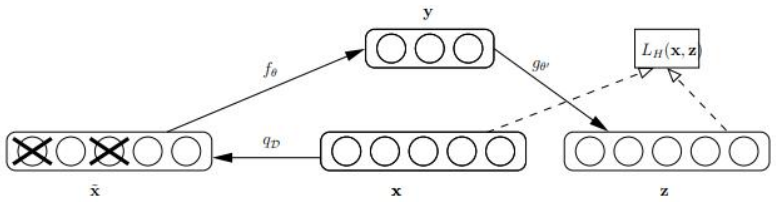
Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery.

Autoencoders



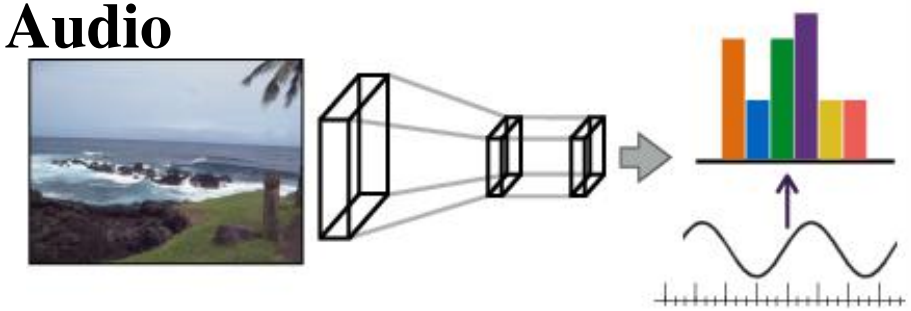
Hinton & Salakhutdinov. Science 2006.

Denoising Autoencoders



Vincent *et al.* ICML 2008.

Audio



Owens *et al.* CVPR 2016, ECCV 2016

Colorization

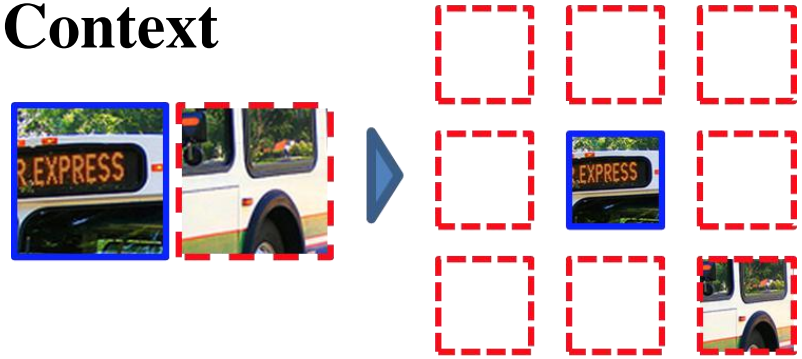
Zhang *et al.* ECCV 2016.

Egomotion

Agrawal *et al.* ICCV 2015 Jayaraman *et al.* ICCV 2015

These are single task

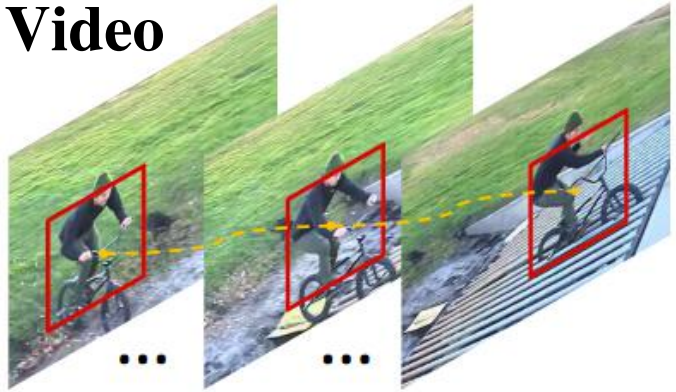
Context



Doersch *et al.* ICCV 2015

Pathak *et al.* CVPR 2016

Video

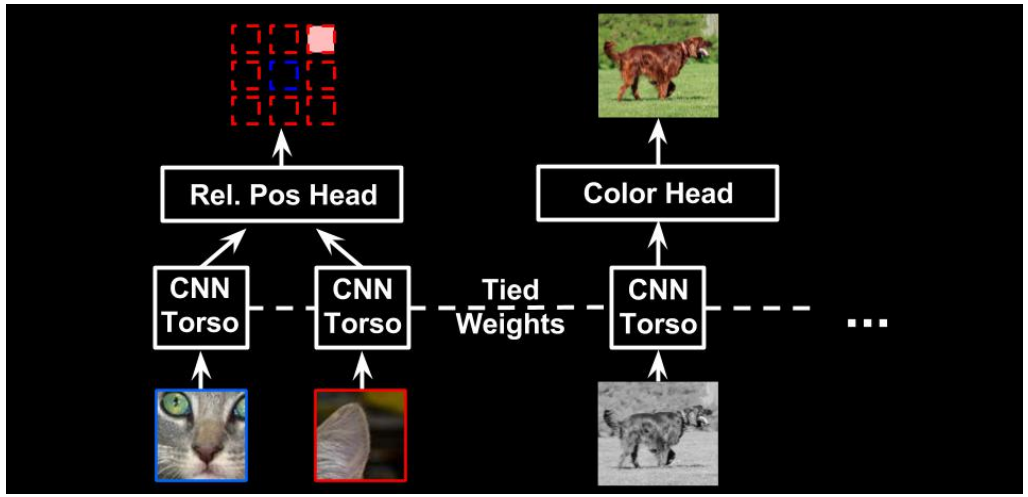


Wang *et al.* ICCV 2015

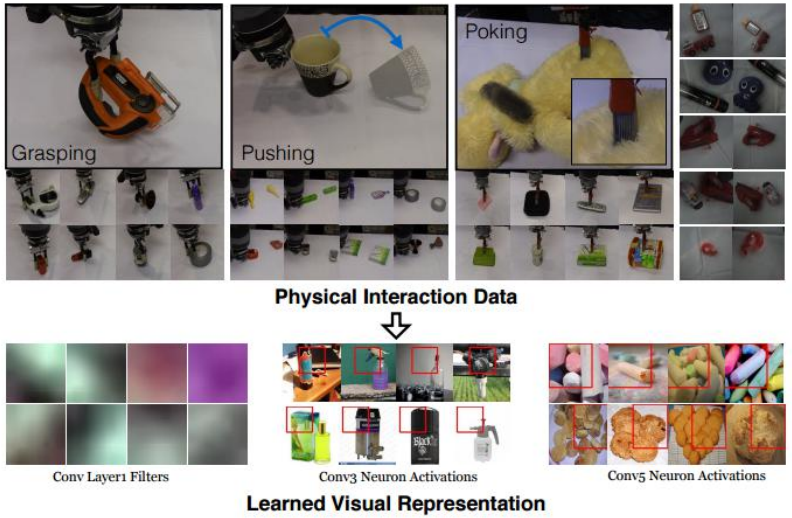
Multi-task could help

- **Lessos learnt from other tasks:**
  - RCNN Family: bbox regression + Classification + (Inst. Segmentation)
  - 3D geometry: Depth + Surface Normal
  - ...

- **For Feature learning:**



Doersch *et al.* ICCV 2017

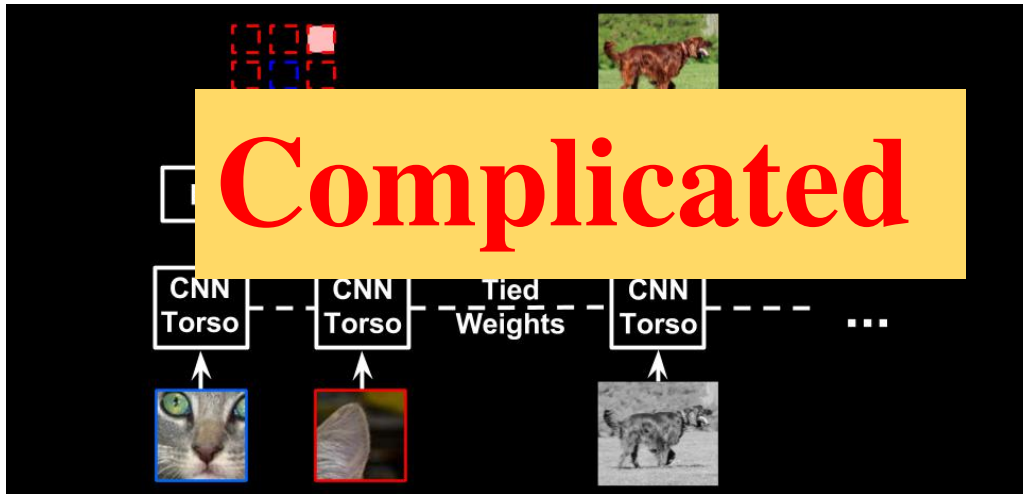


Pinto *et al.* ECCV 2016

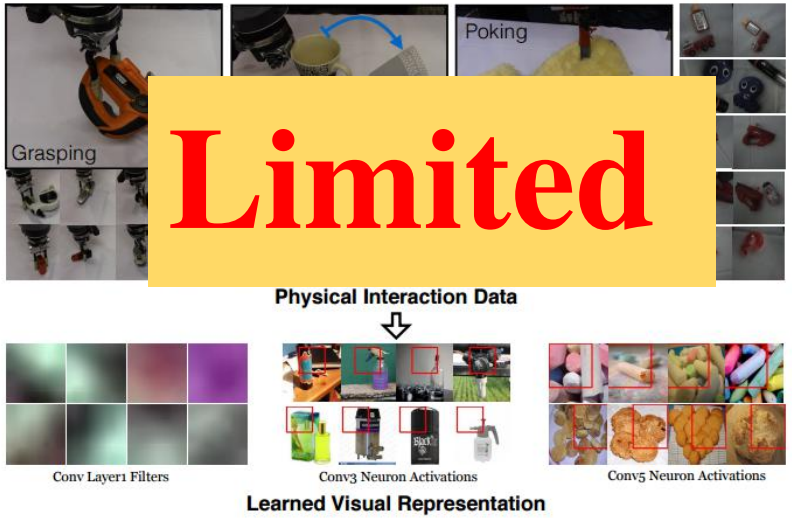
Multi-task could help

- **Lessos learnt from other tasks:**
  - RCNN Family: bbox regression + Classification + (Inst. Segmentation)
  - 3D geometry: Depth + Surface Normal

- **For Feature learning:**



Doersch *et al.* ICCV 2017



Pinto *et al.* ECCV 2016

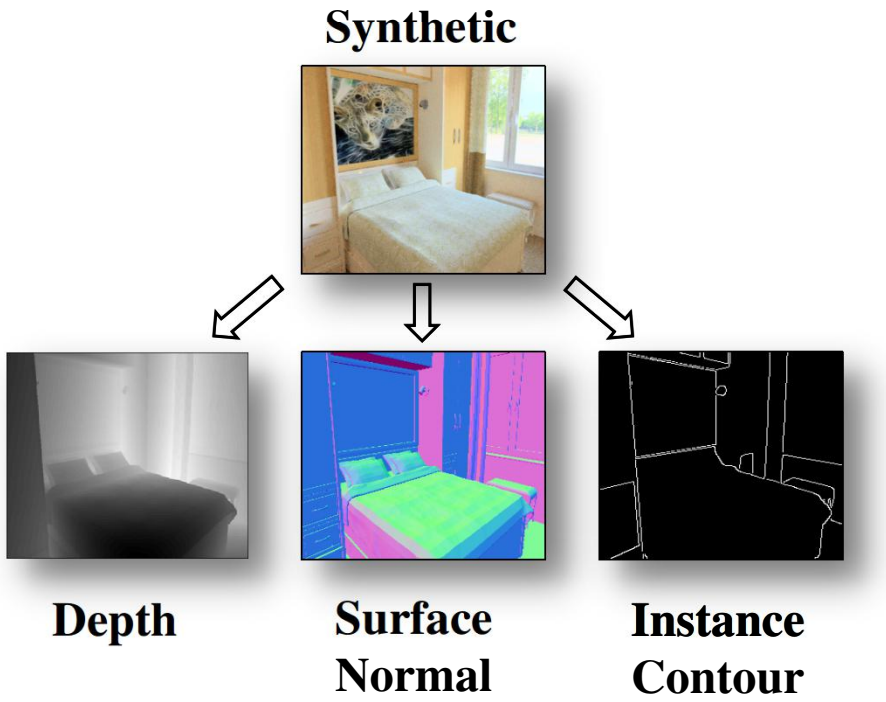
Where to get free and multiple Annotations?

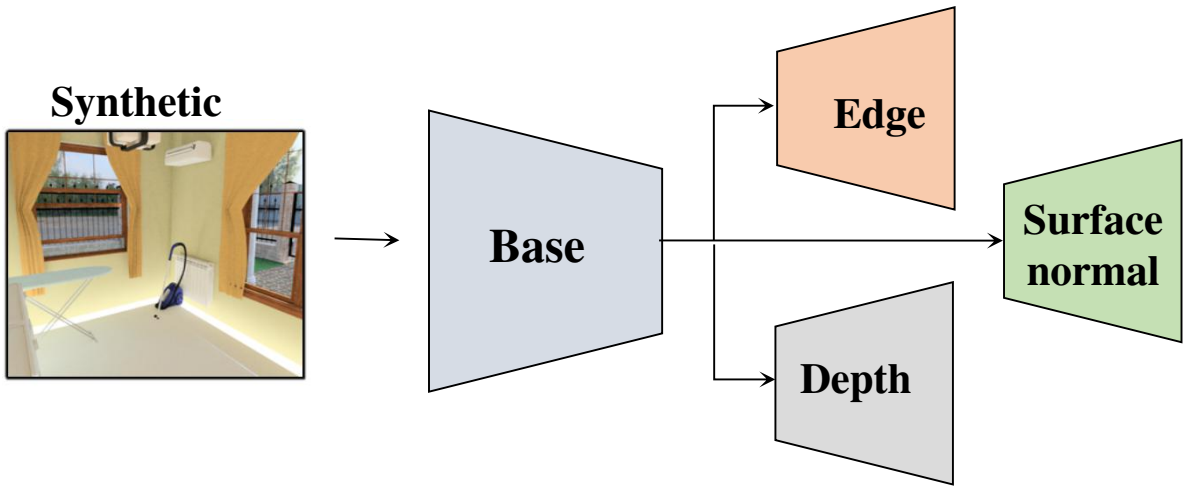
Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery.

The Advantage of Game Engine

- It's become more and more realistic and will only become better
- Tremendous resources that already on the market
- Full freedom to change anything
- Renderring with GT is easy
- Easy to scale

The Architecture

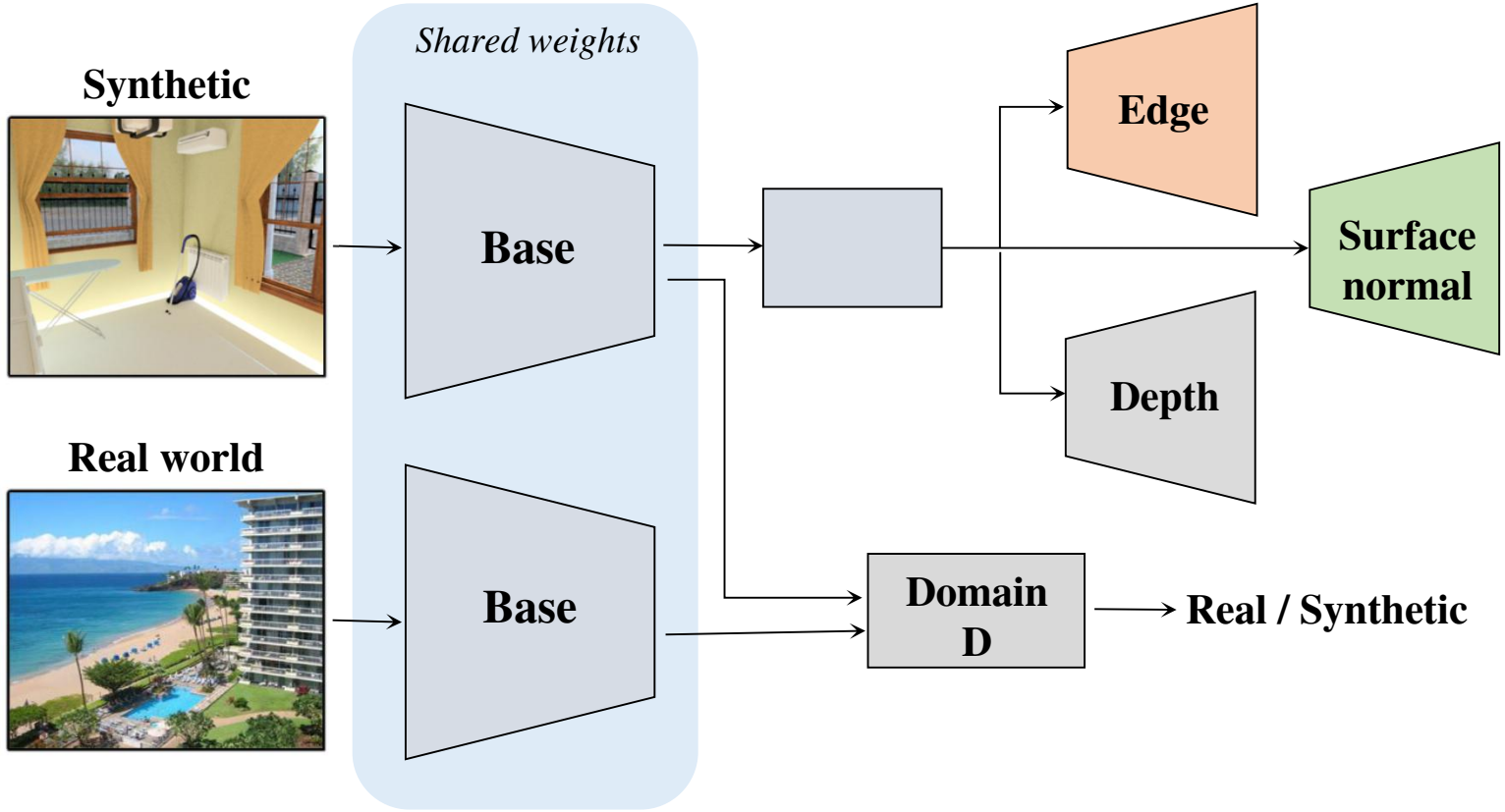




Three pixel-level tasks using Encoder-Decoder arch.

Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery.

To minimize the domain gap

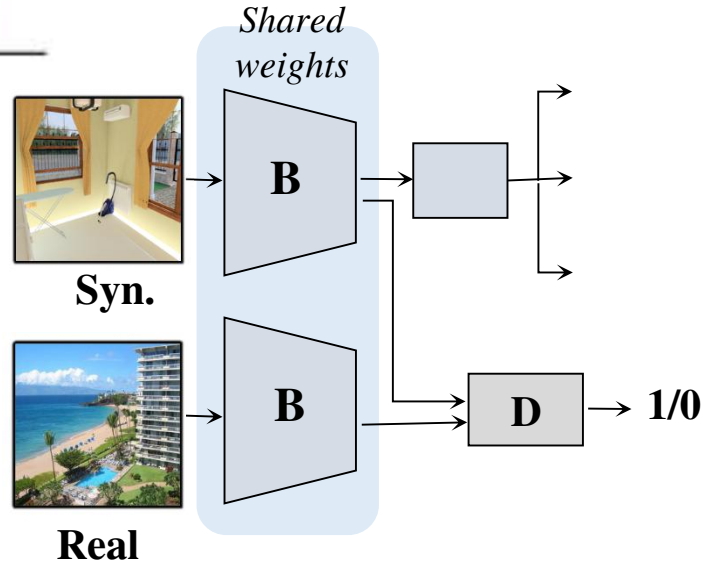


How does Domain Adaptation work?

Algorithm 1 Multi-task Adversarial Domain Adaptation

Input: Synthetic images X, real images Y, max iteration T
Output: Domain adapted base network B

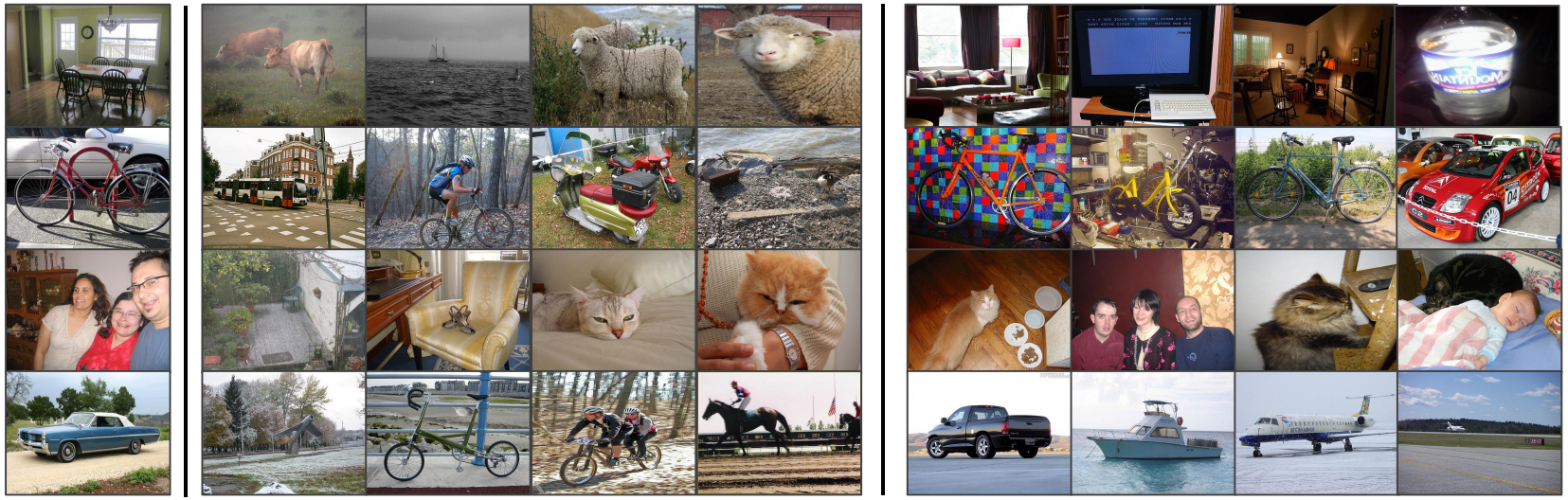
```
for t = 1 to T do
    Sample a batch of synthetic images x = {x_i}
    Sample a batch of real images y = {y_j}
    Extract feature for each image: z_{x_i} = B(x_i), z_{y_j} = B(y_j)
    Keep D frozen, update B, H through L_{BH}(φ_B, φ_H | z_x)
    Keep B frozen, update D through L_D(φ_D | z_x, z_y)
```



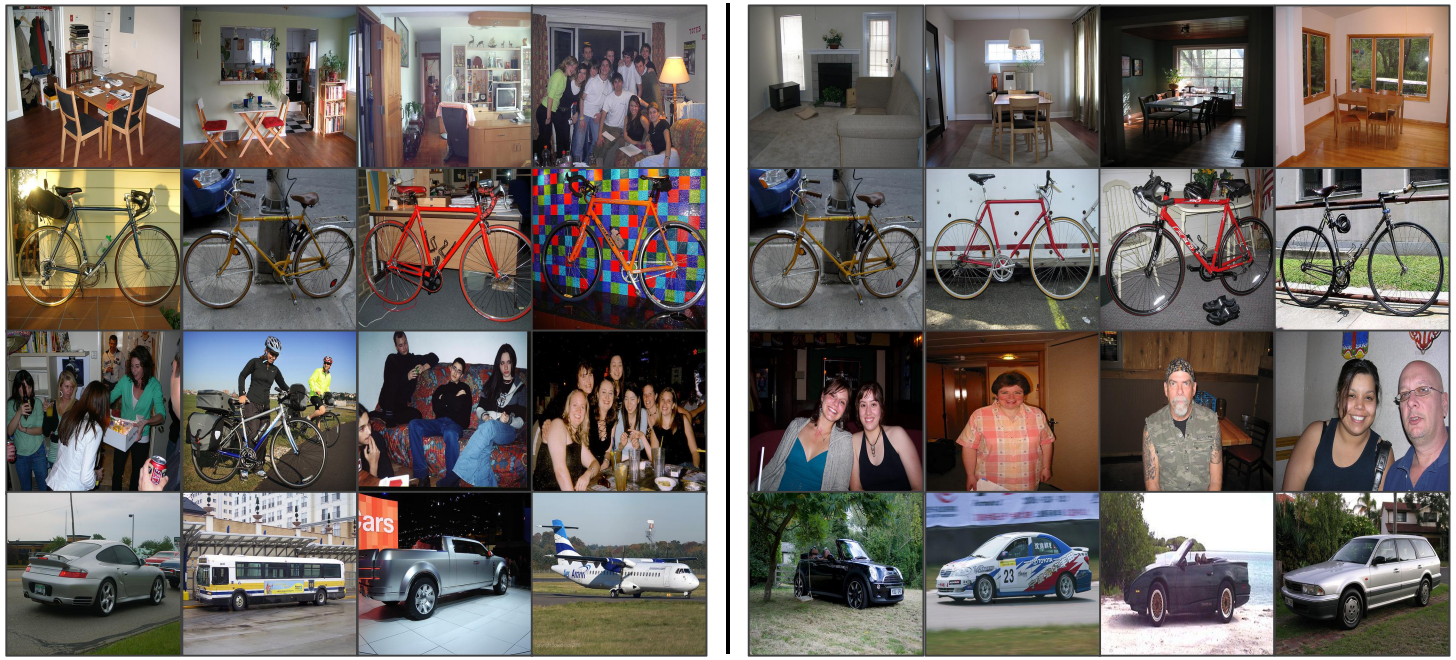
- Inspired on Generative Advasarial Network (GAN)
- B tries to generate features which can fool D (close to natural) which D also learns to discrminate

Experiments
Qualitative

Nearest Neighbor



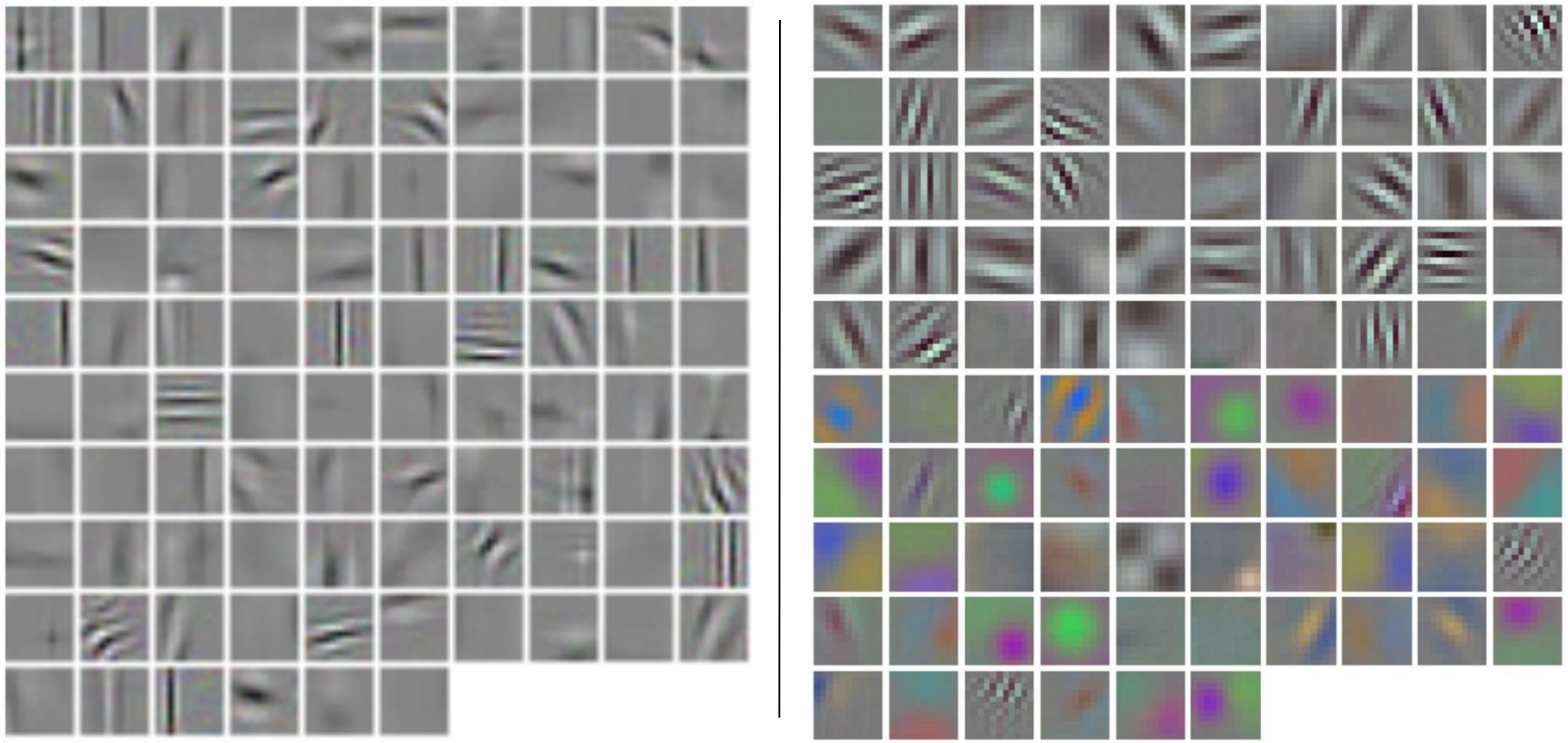
Query **Random weights** **Ours w/o Domain Adaptation**



Ours **ImageNet**

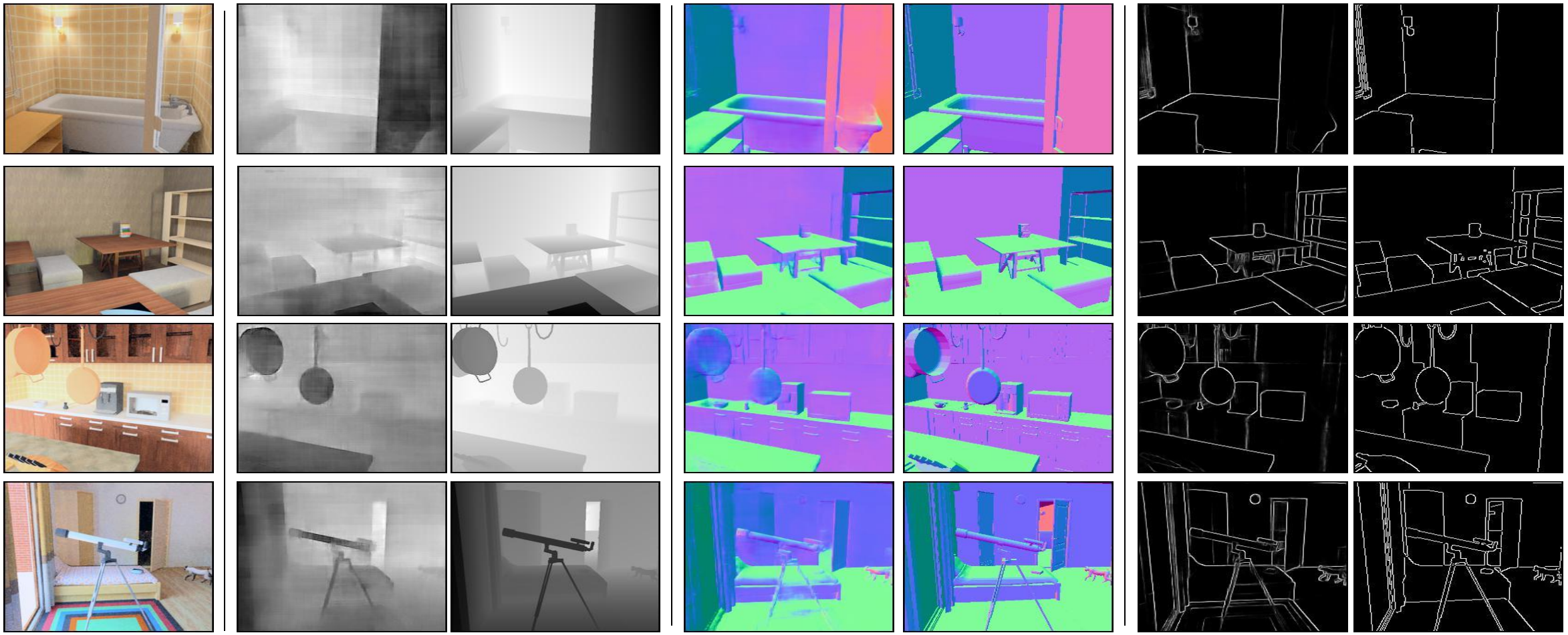
Experiments
Qualitative

conv1 filters



Ours **ImageNet**

Experiments Qualitative Prediction



Synthetic RGB | **Depth Pred.** | **Depth GT** | **Surface normal Pred.** | **Surface normal GT** | **Instance contour Pred.** | **Instance contour GT**

Experiments
Quantative

w/ fine-tuning

Dataset Tasks	07 CLS.	07 DET.	12 DET.
ImageNet [36]	79.9	56.8	56.5
Gaussian	53.4	41.3	-
Autoencoder [35]	53.8	41.9	-
Krahenbuel *et al.* [35]	56.6	45.6	42.8
Ego-equivariance [30]	-	41.7	-
Egomotion [3]	54.2	43.9	-
context-encoder [55]	56.5	44.5	
BiGAN [16]	58.6	46.2	44.9
sound [53]	61.3	-	42.9
flow [54]	61	52.2	48.6
motion [70]	63.1	47.2	43.5
clustering [9]	65.3	49.4	-
context [35]	65.3	51.1	49.9
colorization [75]	65.9	46.9	44.5
jigsaw [51]	67.6	**53.2**	-
splitbrain [76]	67.1	46.7	43.8
counting [52]	67.7	51.4	-
Ours	**68.0**	52.6	**50.0**

Table 1. Transfer learning results on PASCAL VOC 2007 classification and VOC 2007 and 2012 detection. We report the best numbers for each method reported in [35, 76, 52].

Experiments
Quantative
(Transfer learning)

w/o fine-tuning

method	conv1	conv2	conv3	conv4	conv5
ImageNet [36]	19.3	36.3	44.2	48.3	50.5
Gaussian	11.6	17.1	16.9	16.3	14.1
Krähenbühl *et al.* [35]	17.5	23.0	24.5	23.2	20.6
context [14]	16.2	23.3	30.2	31.7	29.6
BiGAN [16]	17.7	24.5	31.0	29.9	28.0
context-encoder [55]	14.1	20.7	21.0	19.8	15.5
colorization [75]	12.5	24.5	30.4	31.5	30.3
jigsaw [51]	**18.2**	28.8	34.0	33.9	27.1
splitbrain [76]	17.7	29.3	**35.4**	**35.2**	**32.8**
counting [52]	18.0	**30.6**	34.3	32.5	25.7
Ours	16.5	27.0	30.5	30.1	26.5

Table 2. Transfer learning results on ImageNet [13]. We freeze the weights of our model and train a linear classifier for ImageNet classification [13]. Our model is trained purely on synthetic data while all other methods are trained on ImageNet [13] (without labels). Despite the domain gap, our model still learns useful features for image classification.

Experiments
Quantative
(Transfer learning)

ablation study

Task	Adaptation	#data	07-C	07-D	12-D
Edge	-	0.5M	63.9	46.9	44.8
Depth	-	0.5M	61.9	48.9	45.8
Surf.	-	0.5M	65.3	48.2	45.4
3 tasks	-	0.5M	**65.6**	**51.3**	**47.2**
3 tasks	conv1	0.5M	61.9	48.7	46
3 tasks	conv4	0.5M	63.4	49.5	46.3
3 tasks	conv5	0.5M	**67.4**	**52.0**	**49.2**
3 tasks	conv6	0.5M	66.9	51.5	48.2
3 tasks	conv5 Bi-fool	0.5M	66.2	51.3	48.5
3 tasks	conv5	1.5M	**68.0**	**52.6**	**50.0**

Table 3. Ablation study results. We evaluate the impact of multi-task learning, feature space domain adaptation, and amount of data on transfer learning. All of these factors contribute together to make our model learn transferable visual features from large-scale synthetic data.

Experiments
Quantative
Cross-Domain

NYUD

		Lower the better		Higher the better		
GT	Methods	Mean	Median	11.25°	22.5°	30°
[17]	Zhang *et al*. [78]	22.1	14.8	**39.6**	65.6	75.3
[17]	Ours	**21.9**	**14.6**	39.5	**66.7**	**76.5**
[38]	Wang *et al*. [71]	26.0	18.0	33.9	57.6	67.5
[38]	Ours	**23.8**	**16.2**	**36.6**	**62.0**	**72.9**

Table 4. Surface normal estimation on the NYUD [50] test set.

Thanks!

* Paper: https://arxiv.org/pdf/1711.09082v1.pdf

* Code coming soon at jason718.github.io

* Self-Supervised Learning Resources:
https://github.com/jason718/awesome-self-supervised-learning