Colorization as a Proxy Task for Visual Understanding

Gustav Larsson *University of Chicago* larsson@cs.uchicago.edu
Michael Maire *TTI-Chicago* mmaire@ttic.edu
Gregory Shakhnarovich *TTI-Chicago* greg@ttic.edu

Overview

Problem statement

- Learning a general-purpose visual representation from **unlabeled data**

Motivation

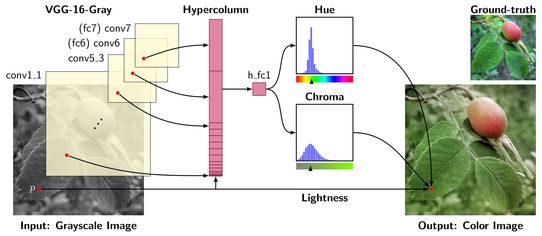
- Reducing reliance on costly data annotation, especially for new problem domains

Solution

- Training a network for automatic image colorization **from scratch**
- Use the trained network as a starting point for other visual tasks

Colorization as a Target Task

- Work in automatic colorization uses feed-forward networks for per-pixel color predictions Larsson et al. (2016); Zhang et al. (2016); Iizuka et al. (2016)
- We use the colorization model with hypercolumns from Larsson et al. (2016):



Source: Larsson et al. (2016)

Computer vision models are typically trained in a two-step process: **pretraining** and **fine-tuning**. Recent methods of colorization also follow this paradigm:

Step 1: Pretrain Classification ⟶ **Step 2: Fine-tune** Colorization

Colorization as a Proxy Task

The steps can be reversed in order for colorization to benefit classification (or other visual tasks):

Step 2: Fine-tune Classification ⟵ **Step 1: Pretrain** Colorization

- Training from scratch, colorization results suffer only slightly although converges slower
- This way of priming a network can be compared to unsupervised pretraining (*e.g.*, autoencoders)
- However, colorization is **self-supervised** and learns using a supervised loss on labeled pairs:

$$(x, y) = \left(\begin{array}{c} \\ \end{array}, \begin{array}{c} \\ \end{array} \right), \left(\begin{array}{c} \\ \end{array}, \begin{array}{c} \\ \end{array} \right), \ldots$$

- Colorization is a great proxy task since this task requires **high-level visual understanding**
- Idea introduced by Larsson et al. (2016); Zhang et al. (2016)
- We extend this work with analysis and best practices, significantly raising state-of-the-art

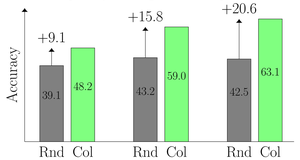
Empirical Study

We summarize some of our findings and describe best practices:

1. **Model complexity**
   → Colorization facilitates scaling up model complexity
2. **Loss**
   → Histogram predictions are significantly better than regression
3. **Training time**
   → Longer is better (does not plateau quickly, best model trained for 4 months)
4. **Learning rate**
   → Important to drop during pretraining, even though downstream fine-tuning awaits
5. **End-to-end fine-tuning on downstream task**
   → Much more important for colorization pretraining than supervised pretraining

Model Complexity 1 & Loss 2

Model complexity has significant impact



Random (Rnd) versus colorization (Col) initialization evaluated on small-sample ImageNet (100 per class), evaluated on regular val (top-5, %)

We consider two different pretraining losses and evaluate their representation learning by using the pretrained models for VOC 2012 Segmentation (val) fine-tuning

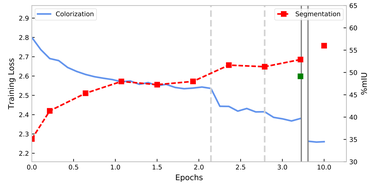
Pretraining Loss	Seg. (%mIU)
Regression	48.0
Histograms	52.9

The technique of predicting histograms turns out to drive better representation learning

Training Time 3 & Learning Rate 4

Relationship between proxy loss (colorization) and downstream score (%mIU for semantic segmentation)

- Long training helps (see tables →)
- Dropping learning rate (dashed lines) improves results



Pretraining times for VGG-16:

Epochs (∼4M)	Seg. (%mIU)
0	35.1
3	52.9
10	56.0

Pretraining times for ResNet-152:

Epochs (∼4M)	Seg. (%mIU)
0	*10.5
3	53.9
10	59.1
35	60.0

*Issues training from scratch

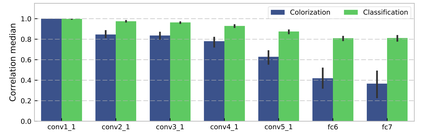
End-to-end Fine-tuning 5

When using colorization pretraining, end-to-end fine-tuning is important

Fine-tuned layers (VGG-16)	Seg. (%mIU) with initialization: Rnd	Col	Cls
∅	3.6	36.5	60.8
fc6, fc7	-	42.6	63.1
conv4_1..fc7	-	53.6	64.2
conv1_1..fc7	35.1	56.0	66.5

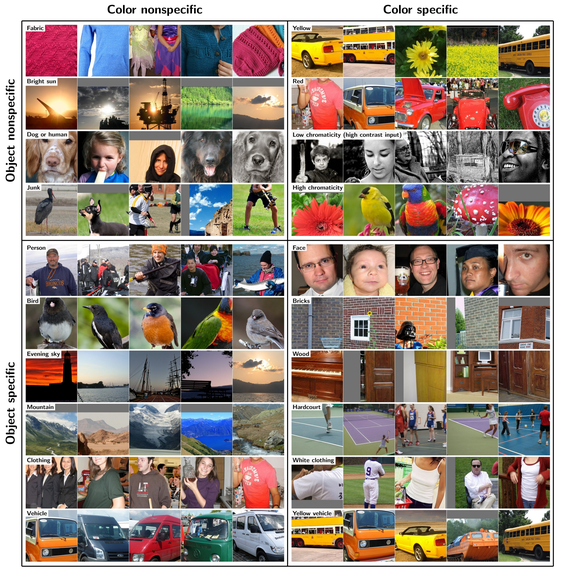
VOC 2012 semantic segmentation results with various configurations of fine-tuning

- ImageNet pretraining (Cls) does well without fine-tuning
- Colorization (Col) offers large improvement over random initialization (Rnd)
- However, colorization is most effective when fine-tuning end-to-end
- The correlation between activations before and after downstream fine-tuning is shown below:



- Features change signficantly more for colorization pretraining than classification pretraining

Feature Visualization



Examples of top activations (`fc7`) of a colorization network

Results

Task: Downstream training without supervised pretraining

Initialization	Architecture		Classification %mAP	Segmentation %mIU
ImageNet pretrained	VGG-16	(+FoV)	86.9	69.5
Random (ours)	AlexNet		46.2	23.5
Autoencoder (ours)	AlexNet		53.3	28.7
Random Pathak et al. (2016)	AlexNet		53.3	19.8
k-means Donahue et al. (2017))	AlexNet		56.6	32.6
k-means Krähenbühl et al. (2016))	VGG-16		56.5	-
k-means Krähenbühl et al. (2016))	GoogLeNet		55.0	-
Inpainting Pathak et al. (2016))	AlexNet		56.5	29.7
Frame Order Wang and Gupta (2015)	AlexNet		58.7	-
BiGAN Donahue et al. (2017)	AlexNet		60.1	35.2
Context Prediction Doersch et al. (2015)	AlexNet		65.3	-
Colorization Zhang et al. (2016)	AlexNet		65.6	35.6
Colorization Larsson et al. (2016)	VGG-16		-	50.2
Split-brain Zhang et al. (2017)	AlexNet		67.1	36.0
Jigsaw Noroozi and Favaro (2016)	Modified AlexNet		68.6	-
Our method	AlexNet		65.9	38.4
	VGG-16	(+FoV)	**77.2**	56.0
	ResNet-152	(+FoV)	**77.3**	**60.0**
Our ensemble	3×ResNet-152	(+FoV)	79.8	61.6

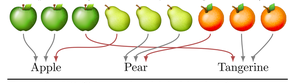
VOC Comparison. Comparison with other self-supervised pretraining methods on VOC 2007 Classification (test) and VOC 2012 Segmentation (val).

Green: Current **state-of-the-art** that uses no additional labeled training data

Bonus: Re-visting supervised pretraining

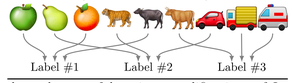
Pretraining	*N*	Epochs	Seg. %mIU
None	-	-	35.1
C1000	1.3M	80	66.5
C1000	1.3M	20	62.0
C1000	100k	250	57.1
C1000	10k	250	44.4
E10 (1.17M)	1.3M	20	61.8
E50 (0.65M)	1.3M	20	59.4
H16	1.3M	20	60.0
H2	1.3M	20	46.1
R50	1.3M	20	57.3
		40	59.4
R16	1.3M	20	42.6
		40	53.5

Example: E30 (30% randomly re-assigned labels)



Example: H3 (3 hierarchical label buckets)

Example: R3 (3 random label buckets)



ImageNet pretraining variations. We evaluate how useful various modifications of ImageNet are for VOC 2012 Segmentation. We create new datasets either by reducing sample size or by reducing the label space as described by the figure.

Read More

Full paper, source code, models and more at:
http://people.cs.uchicago.edu/~larsson/color-proxy/