Colorization as a Proxy Task for Visual Understanding
(2017)

Gustav Larsson Michael Maire Gregory Shakhnarovich

Notes

Contributions

In the paper, the authors conduct an in depth analysis of colorization as a pretext task, by studying the impact of loss, network architecture and training details and how they influence the results. And they also present various formulations of ImageNet pre-training and how they compare to self supervision.

Method

Pretext task training

- **Loss** Regression loss in the `Lab` color space and KL divergence between hue/chroma histograms, the histograms are computed from 7x7 window around each target pixel, and placed into 32 bins for hue and 32 bins for chroma.
- **Hypercolumns** The network use hyper-columns, with sparse training (at each iteration, only a limited number of hyper-columns are computed), and a ResNet network is not used given hyper-columns might disrupt residual training.
- **Dataset** ImageNet & Places205: total of 3.7M images.
- **Training** SGD with 0.9 momentum, 352x352 patches are used as inputs after desaturation, Training for 3 / 10 / 35 epochs (resnet, smaller inputs).

Downstream task training

- **Early stopping** For cross validations, the training set is divided into 90% / 10%, the 10% is used to monitor over-fitting, each time the score stops improving, the learning rate is dropped, this is done twice, and then the training is stopped. The same learning rate is then used to train of 100% of data.
- **Receptive filed** Have a large receptive field is important for some downstream tasks, such as segmentation, but instead of changing the dilation rates of pretrained convolutions, which require re-training, they add two blocks at the top of the network to expand the receptive field and use larger input image to take advantage of this.
- **Batch normalization** BN is either frozen, or fine tuned for ResNet. For pretrained models without BN such as VGG16, the network is rebalanced such that each layer's activation has unit variance.
- **Colors** Given that the pretrained was done using gray scale image, the fine tuning is also done using gray scale image, if RGB image were to be used, the first convolution is duplicated three times and divided by three.

Results

VOC segmentation and classification

Initialization		Architecture	Class.	Seg.
			%mAP	%mIU
ImageNet	(+FoV)	VGG-16	86.9	69.5
Random (ours)		AlexNet	46.2	23.5
Random [32]		AlexNet	53.3	19.8
k-means [20, 5]		AlexNet	56.6	32.6
k-means [20]		VGG-16	56.5	-
k-means [20]		GoogLeNet	55.0	-
Pathak *et al.* [32]		AlexNet	56.5	29.7
Wang & Gupta [39]		AlexNet	58.7	-
Donahue *et al.* [5]		AlexNet	60.1	35.2
Doersch *et al.* [4, 5]		AlexNet	65.3	-
Zhang *et al.* (col) [43]		AlexNet	65.6	35.6
Zhang *et al.* (s-b) [44]		AlexNet	67.1	36.0
Noroozi & Favaro [29]		Mod. AlexNet	68.6	-
Larsson *et al.* [21]		VGG-16	-	50.2
Our method		AlexNet	65.9	38.4
	(+FoV)	VGG-16	**77.2**	56.0
	(+FoV)	ResNet-152	**77.3**	**60.0**

Network architecture & Loss

ImNt-100k/ImNt-10k. Similar to ImageNet classification with 1000 classes, except we have limited the training data to exactly 100 and 10 samples/class, respectively. And test on ImageNet.

Pretraining Loss	Seg. (%mIU)
Regression	48.0
Histograms (no hypercolumn)	52.7
Histograms	52.9

Table 2. **Self-supervision loss.** (VGG-16) The choice of loss has a significant impact on downstream performance. However, pretraining with a hypercolumn does not seem to benefit learning. We evaluate this on VOC 2012 Segmentation (val) with a model that uses hypercolumns, regardless of whether or not it was used during pretraining.

Architecture	Init.	Seg.	+FoV	ImNt-100k	10k
		%mIU		%top-5	
AlexNet	Rnd	23.5	24.6	39.1	6.7
AlexNet	Col	36.2	40.8	48.2	17.4
VGG-16	Rnd	32.8	35.1	43.2	8.6
VGG-16	Col	50.7	52.9	59.0	23.3
ResNet-152	Rnd	*9.9	*10.5	42.5	8.1
ResNet-152	Col	52.3	53.9	63.1	29.6

Table 3. **Architectures.** We compare how various networks perform on downstream tasks on random initialization (Rnd) and colorization pretrained (Col). For our segmentation results, we also consider the effects of increasing the receptive field size (+FoV). Training residuals from scratch (marked with a *) is possibly compromised by the hypercolumn, causing the low values.

ImageNet pretraining

C1000 = All 1000 classes, H16 = 16 buckets (H2 = 2 buckets) (found using WordNet), R16 and R50 = take the 1000 ImageNet classes and randomly place them in 50 (R50) or 16 (R16) buckets

Pretraining	Samples	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

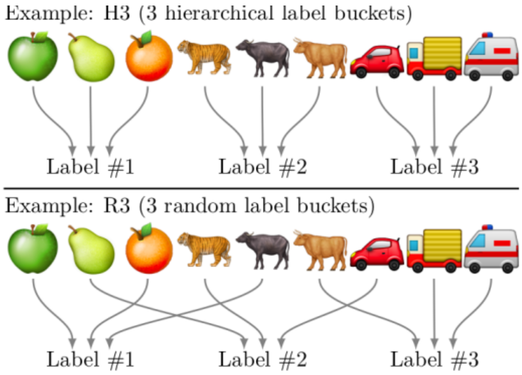
Example: H3 (3 hierarchical label buckets)
Label #1
Label #2
Label #3
Example: R3 (3 random label buckets)
Label #1
Label #2
Label #3

Table 4. **ImageNet pretraining.** We evaluate how useful various modifications of ImageNet are for VOC 2012 Segmentation (val-gray). We create new datasets either by reducing sample size or by reducing the label space. The former is done simply by reducing sample size or by introducing 10% (E10) or 50% (E50) label noise. The latter is done using hierarchical label buckets (H16 and H2) or random label buckets (R50 and R16). The model trained for 80 epochs is the publicly available VGG-16 (trained for 76 epochs) that we fine-tuned for grayscale for 4 epochs. The rest of the models were trained from scratch on grayscale images.

Color and finetuning

Fine-tuned layers (VGG-16)		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

Table 6. **VOC 2012 Segmentation.** (%mIU) Classification-based pretraining (Cls) needs less fine-tuning than our colorization-based method (Col). This is consistent with our findings that our network experiences a higher level of feature shift (Fig. 3). We also include results for a randomly initialized network (Rnd), which does not work at all without fine-tuning (3.6%). This is to show that it is not simply by virtue of the hypercolumn that we are able to do reasonably well (36.5%) without any fine-tuning of the base network.

Initialization	Grayscale input	Color input
Classification	66.5	69.5
Colorization	56.0	55.9

Table 5. **Color vs. Grayscale input.** (VOC 2012 Segmentation, %mIU) Even though our classification-based model does 3 points better using color, re-introducing color yields no benefit.