

000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044

Colorful Image Colorizations

Supplementary Material

000
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044

Anonymous ECCV submission

Paper ID 1031

1 Overview

This document is divided into three sections. Section 2 adds some clarifications regarding filtering grayscale images from the dataset, along with additional details about the network architecture. Section 3 contains a discussion of our algorithm in comparison to Deep Colorization [1]. Section 4 contains a more detailed explanation of the VGG category analysis presented in Section 4.1 and Figure 5 of the paper, along with an additional analysis on common category confusions after recolorization.

2 Clarifications

2.1 Network architecture

Figure 2 in the paper showed a diagram of our network architecture. Table 1 in this document thoroughly lists the layers used in our architecture during training time. During testing, the temperature adjustment, softmax, and bilinear upsampling are all implemented as subsequent layers in a feed-forward network. Note the column showing the effective dilation. The effective dilation is the spacing which consecutive elements of the convolutional kernel are evaluated, relative to the input pixels, and is computed by the product of the accumulated stride and the layer dilation. Through each convolutional block from `conv1` to `conv5`, the effective dilation of the convolutional kernel is increased. From `conv6` to `conv8`, the effective dilation is decreased.

2.2 Filtering Grayscale Images

Some of the images in the Imagenet [2] dataset are in grayscale and were filtered out of training, validation, and testing sets. An image was considered to be grayscale if no pixel had a value of ab above 5 and was withheld from training and testing. The threshold was set to be conservative. A more aggressive threshold would remove more grayscale images at the expense of color images which happened to contain a very desaturated palette.

045 2.3 Training, Validation, and Testing Splits

046
047
048 The full 1.3M images, minus the grayscale images were used for training. The
049 first 2000 images in the Imagenet [2] validation set were used for validation. For
050 the aggregated *quantitative* testing results shown in Table 1 of the paper, we
051 used the last 10,000 images in the validation set, 9803 of which were color. For
052 the *qualitative* VGG classification results shown in Figure 5, we used the last
053 48,000 images in the validation set were used, 47,023 of which contained color.
054 Since we sorted on classification performance across 1000 categories for the VGG
055 analysis, we used the full validation set for testing to maximize the number of
056 samples per category.

	X	C	S	D	Sa	De	BN	L
data	224	3	-	-	-	-	-	-
conv1_1	224	64	1	1	1	1	-	-
conv1_2	112	64	2	1	1	1	✓	-
conv2_1	112	128	1	1	2	2	-	-
conv2_1	56	128	2	1	2	2	✓	-
conv3_1	56	256	1	1	4	4	-	-
conv3_2	56	256	1	1	4	4	-	-
conv3_3	28	256	2	1	4	4	✓	-
conv4_1	28	512	1	1	8	8	-	-
conv4_2	28	512	1	1	8	8	-	-
conv4_3	28	512	1	1	8	8	✓	-
conv5_1	28	512	1	2	8	16	-	-
conv5_2	28	512	1	2	8	16	-	-
conv5_3	28	512	1	2	8	16	✓	✓
conv6_1	28	512	1	2	8	16	-	-
conv6_2	28	512	1	2	8	16	-	-
conv6_3	28	512	1	2	8	16	✓	✓
conv7_1	28	256	1	1	8	8	-	-
conv7_2	28	256	1	1	8	8	-	-
conv7_3	28	256	1	1	8	8	✓	✓
conv8_1	56	128	.5	1	4	4	-	-
conv8_2	56	128	1	1	4	4	-	-
conv8_3	56	128	1	1	4	4	-	✓

081 **Table 1.** Our network architecture. **X** spatial resolution of output, **C** number of chan-
082 nels of output; **S** computation stride, values greater than 1 indicate downsampling
083 following convolution, values less than 1 indicate upsampling preceding convolution;
084 **D** kernel dilation; **Sa** accumulated stride across all preceding layers (product over all
085 strides in previous layers); **De** effective dilation of the layer with respect to the input
086 (layer dilation times accumulated stride); **BN** whether BatchNorm layer was used after
087 layer; **L** whether a 1x1 conv and cross-entropy loss layer was imposed

3 Comparison to Deep Colorization [1]

Quantitative comparisons to Deep Colorization [1] are not possible, as the authors have not released their code or test set results. We provide qualitative comparisons to the 23 test images in [1] on the attached website, which we obtained by manually cropping from the paper. Our results are about the same qualitative level as [1]. Note that Deep Colorization [1] has several advantages in this setting: (1) the test images are from the SUN dataset [3], which we did not train on and (2) the 23 images were hand-selected from 1344 by the authors, and is not necessarily representative of algorithm performance. We were unable to obtain the 1344 test set results through correspondence with the authors.

Additionally, we compare the methods on several important dimensions in Table 2: algorithm pipeline, learning, dataset, and run-time. Our method is faster, straightforward to train and understand, has fewer hand-tuned parameters and components, and has been demonstrated on a broader and more diverse set of test images than Deep Colorization [1].

	Deep Colorization [1]	Ours
Algorithm	(1) Extract feature sets (a) 7x7 patch (b) DAISY (c) FCN on 47 categories (2) 3-layer NN regressor (3) Joint-bilateral filter	Feed-forward CNN
Learning	Extract features. Train FCN [4] on pre-defined categories. Train 3-layer NN regressor.	Train CNN from pixels to color distribution. Tune single parameter on validation.
Dataset	2688/1344 images from SUN [3] for train/test. Limited variety with only scenes.	1.3M/10k images from ImageNet [2] for train/test. Broad and diverse set of objects and scenes.
Run-time	4.9s/image on Matlab implementation	100ms/image in <i>Caffe</i> on K40 GPU

Table 2. Comparison to Deep Colorization [1]

4 VGG Evaluation

4.1 Classification Performance

In Section 4.1, we investigated the grayscale and re-colored images using the VGG classifier [5] for the last 48,000 images in the Imagenet validation set. For each category, we computed the top-5 classification performance on grayscale and recolorized images, $\mathbf{a}_{gray}, \mathbf{a}_{recolor} \in [0, 1]^C$, where $C = 1000$ categories. We sorted the categories by $\mathbf{a}_{recolor} - \mathbf{a}_{gray}$ and plotted examples of some selected

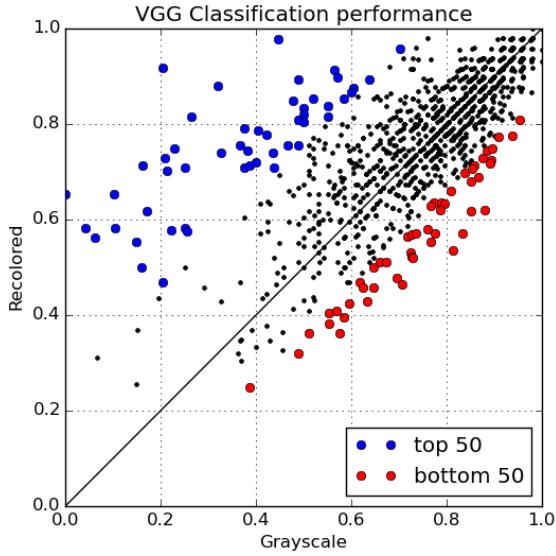


Fig. 1. Performance of VGG top-5 classification on recolorized images vs grayscale images per category. Test was done on last 48,000 images in Imagenet validation set.

top and bottom classes in Figure 5 of the paper. The re-colored vs grayscale performance per category is shown in Figure 1, with top and bottom 50 categories highlighted. For the top example categories, the individual images are sorted by ascending rank of the correct classification of the recolored image, with tiebreakers on descending rank of the correct classification of the grayscale image. For the bottom example categories, the images are sorted in reverse, in order to highlight the instances when recolorization results in an errant classification relative to the grayscale image.

4.2 Common Confusions

To further investigate the biases in our system, we look at the common class confusions which often occur after image recolorization but not with the original ground truth image. We compute the rate of top-5 confusion $\mathbf{C}_{orig}, \mathbf{C}_{recolor} \in [0, 1]^{C \times C}$, with ground truth colors and after recolorization. A value of $\mathbf{C}_{c,d} = 1$ means that every image in category c was classified as category d in the top-5. We find the class-confusion added after recolorization by computing $\mathbf{A} = \mathbf{C}_{recolor} - \mathbf{C}_{orig}$, and sort the off-diagonal entries. Figure 2 shows all $C \times (C-1)$ off-diagonal entries of $\mathbf{C}_{recolor}$ vs \mathbf{C}_{orig} , with the top 100 entries from \mathbf{A} highlighted.

For each category pair (c, d) , we extract the images which contained the confusion after recolorization but not with the original colorization. We then sort the images in descending order of the classification score of the confused category. Examples for some top categories are shown in Figure 3. An image of a “minibus” is often colored yellow, leading to a misclassification as “school bus”. Animal classes are sometimes colored differently than ground truth, leading to

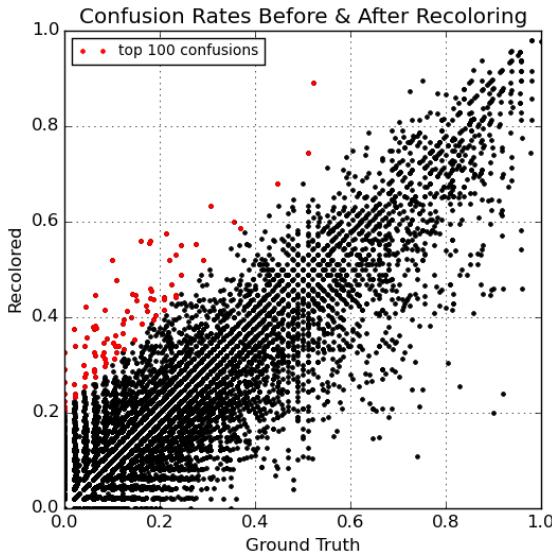


Fig. 2. Top-5 confusion rates with recolorizations and original colors. Test was done on last 48,000 images in Imagenet validation set [2].

misclassification to related species. Note that the colorizations are often visually realistic, even though they lead to a misclassification.

(2) jacamar
bulby



(3) standard schnauzer



(5) minibus



(7) weinramer
redphone



Fig. 3. Examples of some most-confused categories. Top rows show ground truth image. Bottom rows show recolorized images. Rank of common confusion in parentheses. **Ground truth** and **confused** categories after recolorization are labeled.

270 References

- 271
- 272 1. Cheng, Z., Yang, Q., Sheng, B.: Deep colorization. In: Proceedings of the IEEE
273 International Conference on Computer Vision. (2015) 415–423
- 274 2. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z.,
275 Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recog-
276 nition challenge. International Journal of Computer Vision **115**(3) (2015) 211–252
- 277 3. Patterson, G., Hays, J.: Sun attribute database: Discovering, annotating, and rec-
278 ognizing scene attributes. In: Computer Vision and Pattern Recognition (CVPR),
279 2012 IEEE Conference on, IEEE (2012) 2751–2758
- 280 4. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic
281 segmentation. In: Proceedings of the IEEE Conference on Computer Vision and
282 Pattern Recognition. (2015) 3431–3440
- 283 5. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image
recognition. arXiv preprint arXiv:1409.1556 (2014)
- 284
- 285
- 286
- 287
- 288
- 289
- 290
- 291
- 292
- 293
- 294
- 295
- 296
- 297
- 298
- 299
- 300
- 301
- 302
- 303
- 304
- 305
- 306
- 307
- 308
- 309
- 310
- 311
- 312
- 313
- 314