Reproducing *Colorful Image Colourization*

Timo Nicolai
tnicolai@kth.se

Álvaro Orgaz Expósito
alvarooe@kth.se

Carolina Bianchi
cbianchi@kth.se

Abstract

In this project we reproduce part of the work presented by Zhang et al. in *Colorful Image Colorization* [1]. We implement their network architecture from scratch in *PyTorch* [2] and train it on a subset of ImageNet [3]. We find that our network is able to produce realistic colourizations on par with the reference implementation provided by Zhang et al. in many cases. We further describe several experiments that support this conclusion, providing quantitative metrics for the closeness between colourizations produced by our network and corresponding ground truth images. We also show how, in some cases, our network is able to fool human testers in a perceptual realism study. Finally we show that our network can improve greyscale image classification accuracy when employed as a preprocessing step.

1 Introduction

Greyscale image colourization has been a longstanding topic of interest in the computer vision and machine learning communities. The principle problem is to *plausibly* colourize greyscale input images, implying that the results do not necessarily have to match the "correct" colours for any given image. Instead they only have to be believable, i.e. able to fool humans observers into mistaking them for real colour images.

Besides its appeal in areas such as the colourization of historical images, colourization networks can also serve as a preprocessing step for image classification and segmentation networks or be used in transfer learnings tasks. A possible commercial application is the improvement of surveillance systems without hardware upgrades.

The problem of colourization is clearly under-constrained and ill-posed: many objects could be plausibly coloured in multiple different ways. This inherent multimodality suggests the use of a network that predicts a distribution of possible colours for each pixel which can then be converted into a suitable point estimate (e.g. the distributions mean or mode).

The multimodality of the problem also makes assessing a method's performance difficult: metrics that simply measure some form of distance between ground truth and prediction provide limited insight into the plausibility of the latter. Under such metrics, methods producing conservative (i.e. desaturated) results often outperform methods producing more vibrant images whose colour scheme differ from the ground truth but remain similarly plausible. The plausibility of colourized images can thus be most reliably assessed by human testers.

In section 2 we first provide a short overview of related previous and current approaches to the colourization problem. Section 3 describes the datasets which we utilized to train our network. Section 4 provides a summary of the method developed by Zhang et al. [1] on which our work is based. In section 5 we describe how we trained our network and present some exemplary results as well as the outcome of a perceptual realism study with human testers. Here we also evaluate how preprocessing greyscale images using our network can lead to improved image classification results.

2 Related work

Image colourization has been well studied in the literature. Here we provide an overview of major previous and current approaches with a focus on (deep) learning based methods.

Very early work in this field was performed by Welsh et al. [4] whose approach transferred colours from an existing reference image through matching of luminance information and texture.

Shortly thereafter Levin et al. [5] formulated colourization as an optimization problem and designed a loss function which penalizes colour differences between neighboring pixels of similar intensity. However, their approach required user annotation to produce acceptable results.

Early learning based approaches include work by Deshpande et al. [6] who implemented colourization by training a linear system and work by Cheng et al. [7] who utilized a multi-layer fully connected neural network.

Zhang et al. [1] leveraged large-scale data and a deep (almost) end-to-end trainable convolutional neural network. Similar methods were developed concurrently by Larsson et al. [8] and Iizuka et al. [9]. These mainly differ in the utilized network architectures and loss functions, i.e. L_2 reconstruction loss (Iizuka et al.) and classification loss (Zhang et al, Larsson et al.).

More recent approaches include work by Nazeri et al. [10] which leverages Deep Convolutional Generative Adversarial Networks in order to learn both a mapping from lightness input to colour output and a loss function via a discriminator network. This is in turn based on prior work by Isola et al. [11] and mainly improves on the proposed training approach and generalizes it to higher resolution images.

An interesting variant of the colourization problem is exemplar-based colourization in which the colourization is aided by a user provided reference image. A deep learning approach was first applied to this problem by He et al. [12]. The authors also describe an image retrieval algorithm that can automatically determine suitable reference images. The key advantage of this approach is that by selecting different references, more than one plausible colourization can be produced per input image.

3 Data

We work with images in the Lab colour space, in which each pixel is encoded by three values: its lightness (black to white), its a* value (green to red) and its b* value (blue to yellow). This colour space was originally designed such that the distances within it roughly correspond to perceptual "distances" [13], making it especially well suited for tasks such as this where the ultimate aim is to fool the human eye.

The image colourization task has the appealing property that any colour image is a potential self labeled training sample. After converting it into the Lab colour space, its lightness channel can serve as a network input and its a and b channels as supervisory signal.

To train the our network we decided to use a subset of the ImageNet [3] training set. We created this subset by first choosing 33 semantically closely related[1] synsets from the 1000 synsets making up the complete training set. These include $42,556$ images of mostly fruits and vegetables.

Our motivation for this was twofold: Firstly, a training set made up of few, visually related objects is likely to speed up generalization to test images depicting similar objects. Secondly, most images in the synsets we chose depict objects with distinct and vibrant colours and easily identifiable textures. These make for excellent examples when demonstrating our networks performance in section 5.

We do not employ a validation set. Even though our training set is relatively small, overfitting is unlikely because of the short training time and appropriate regularization.

In accordance with Zhang et al. we resize all images in the training set to 256×256. During training, these are randomly cropped to 176×176 and mirrored horizontally with probability 0.5.

[1] We defined closeness via WordNet [14] path similarity, see http://www.nltk.org/howto/wordnet.html.

4 Methods

4.1 Loss function

Starting from an image of size $H \times W$ in the Lab colour space, we use its luminance channel $X_L \in \mathbb{R}^{H \times W \times 1}$ as input to our network, which we train to estimate the remaining channels (X_a, X_b) in order to generate a fully coloured version of the image $X = (X_L, X_a, X_b)$.

Given the input lightness channel the objective is to learn a mapping $\hat{X}_{ab} = F(X_L)$ to the two associated colour channels $X_{ab} \in \mathbb{R}^{H \times W \times 2}$. A natural objective function would be the Euclidean distance L_2 between the predicted and ground truth colours of each pixel. However, due to the multimodal nature of the colourization problem, we can achieve more vibrant results using a classification loss. The classes in question result from the quantization of the original *ab* output space into bins of size 10×10. $Q = 313$ of these bins are *in-gamut*, i.e. representable in the sRGB colour space.

The network predicts the per pixel colour probability distribution $\hat{Z} \in [0, 1]^{H \times W \times Q}$. More specifically, for every pixel of the input image (after several downsampling steps) the network outputs Q values which roughly correspond to the log probabilities of the *ab* value associated with that pixel falling into the respective bins. This distribution can be converted back to the original *ab* space by taking its *annealed-mean* as described below.

In order to train the networks parameters we additionally need to convert the ground truth images *ab* channels into a ground truth distribution Z. It would be reasonable to encode the ground truth value of each pixel as a 1-hot vector $Z_{h,w}$ of length $Q = 313$ by searching for the nearest quantized *ab* bin. Instead we employ the soft-encoding scheme proposed by Zhang et al. which encodes every pixel by its five nearest neighbours in the binned *ab* space, weighted proportionally to their distance from the ground truth values using a Gaussian kernel with $\sigma = 5$. This allows the network to learn relationships between elements in the output space more easily.

Finally, the model parameters are optimized by minimizing the multinomial cross entropy loss defined in equation 1.

$$L(Z, \hat{Z}) = -\sum_{w,h} v(Z_{w,h}) \sum_{q} Z_{w,h,q} \cdot log(\hat{Z}_{w,h,q}) \tag{1}$$

Here, $v(Z_{h,w})$ is a weighting term that rebalances each pixel's contribution based on the rarity of its closest *ab* bin. More specifically, Zhang et al. provide a *ab* bin prior distribution $\tilde{\boldsymbol{p}}$ calculated from all images in ImageNet. The weights are then chosen as a mixture of this prior and a uniform distribution over all *ab* bins, i.e.:

$$v(Z_{w,h}) \propto \left((1 - \lambda)\tilde{\boldsymbol{p}} + \frac{\lambda}{Q} \right)^{-1} \tag{2}$$

In practice this reweighting is implemented by a custom PyTorch layer which, on every backward pass, multiplies the per pixel gradients by the weighting terms which are recalculated on the GPU for each input image.

This class rebalancing is crucial in order to achieve plausible colourizations since the distribution of colours in natural images is strongly biased towards low *ab* values (mostly due to background objects such as clouds, pavement, dirt, walls etc.). Without rebalancing the loss function is dominated by desaturated *ab* values and network outputs are significantly less vibrant.

4.2 From colour probabilities to point estimates

In order to convert the distribution $\hat{Z}$ into point estimates in the original *ab* space $\hat{X}_{ab}$ we implement the annealed-mean operation in equation 3.

$$H(Z_{h,w}) = E[f(Z_{h,w})] \qquad f(z) = \frac{exp(log(z)/0.38)}{\sum_q exp(log(z)/0.38)} \tag{3}$$

Figure 1: Effect of temperature parameter on network output. Note how higher values of T results in a less vibrant but more spatially consistent image.

Table 1: Network architecture

Layer	C	S	D	BN	K
conv1_1	64	1	1	-	3
conv1_2	64	2	1	✓	3
conv2_1	128	1	1	-	3
conv2_2	128	2	1	✓	3
conv3_1	256	1	1	-	3
conv3_2	256	1	1	-	3
conv3_3	256	2	1	✓	3
conv4_1	512	1	1	-	3
conv4_2	512	1	1	-	3
conv4_3	512	1	1	✓	3
⋮	⋮	⋮	⋮	⋮	⋮
conv5_1	512	1	2	-	3
conv5_2	512	1	2	-	3
conv5_3	512	1	2	✓	3
conv6_1	512	1	2	-	3
conv6_2	512	1	2	-	3
conv6_3	512	1	2	✓	3
conv7_1	512	1	1	-	3
conv7_2	512	1	1	-	3
conv7_3	512	1	1	✓	3
conv8_1	256	.5	1	-	4
conv8_2	256	1	1	-	3
conv8_3	256	1	1	-	3
conv9_1	313	1	1	-	1

Through this operation it possible to make a trade off between the mean and mode of the predicted output distribution. The former often leads to spatially consistent but desaturated images and the latter to vibrant images containing unwanted image artefacts (patches of high saturation). Setting $T = 0.38$ as suggested by Zhang et al. offers a nice compromise between the two.

Figure 1 demonstrates how adjusting the temperature parameter results in a trade-off between mean and mode of the colour bin distribution predicted by our network.

4.3 Network architecture

Table 1 lists our network architecture.[2] It is adapted from Figure 2 in [1], activation layers are not depicted. Downsampling is achieved with strided convolutions (i.e. without pooling layers). Also note the dilated (otherwise referred to as atrous) convolution layers which increase the receptive field of the trained convolution kernels without an explosion in the number of parameters.

5 Experiments

5.1 Training

Because extensive search for good optimization hyperparameters would have been prohibitively expensive given the available GPU resources, we simply applied the same optimization regimen used by Zhang et al. to train their publicly available models.[3] We also utilized class rebalancing with $\lambda = 0.5$.

[2] C = number of filters, S = stride, D = dilation, BN = batch normalization, K = filter diameter

[3] ADAM [15] optimizer with $\beta_1 = .9$, $\beta_2 = .99$, weight decay = 10^{-3}, $\eta = 3.16e \times 10^{-5}$, batch size = 40.

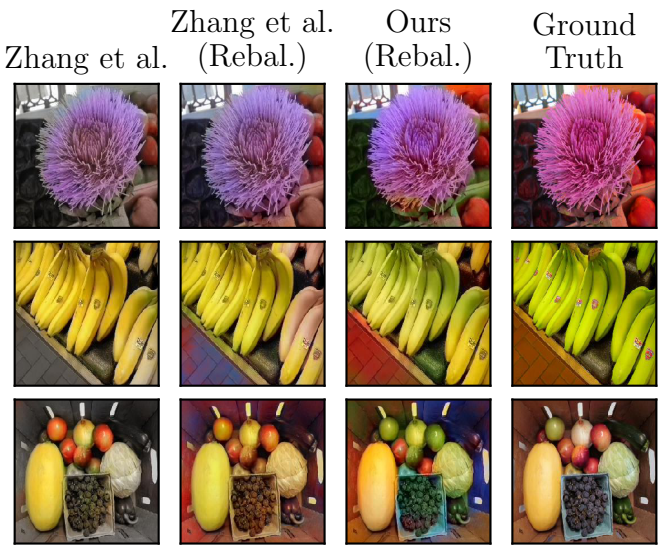

Figure 2: Particularly good examples where our network produces more vibrant and realistic colours than the pretrained models provided by Zhang et al. trained on the full ImageNet training set.

We trained our network for $150,000$ iterations which took approximately 20 hours on an *NVIDIA Tesla V100*.

5.2 Examples

The trained network produces results which are in many cases obviously fake. Nevertheless, the network seems to consistently apply realistic and vibrant colours to certain prominent objects that are abundant in the training set such as bananas or blooming artichokes. Figure 2 shows some test set images for which our network performs especially well.

We believe that the achieved vibrancy is a result of class rebalancing and that the overall lackluster network performance is caused by the challenging nature of the chosen dataset which contains many colourful and visually very different objects (unlike images of natural scenes on which the network trained by Zhang et al. performs especially well and which are mostly composed of large areas homogeneous in texture and colour, such as forest, sea and air).

5.3 Perceptual realism study

To test whether the colour images produced by our network are realistic enough to fool the human eye, we colourized fifty images from the ImageNet validation set (from the same synsets making up the training set) and presented them to ten volunteers alongside the corresponding ground truth images, displaying each of these in turn for exactly one second. The volunteers were then given unlimited time to identify the real ground truth image.

Figure 3 displays the images which fooled the greatest number of participants and a selection of those which were never mistaken for the ground truth image. On average, participants were fooled 18.78% of the time indicating that while our network is able to produce realistic colourization in some "easy" cases, results are mostly obviously artificial.

5.4 Colourization accuracy and semantic interpretability

We made use of the *area under curve* (AuC) metric introduced in [6] to also obtain a quantitative measure of how close the results produced by our network are to the ground truth. Zhang et al. improve on this metric by reweighting pixels by the inverse prior likelihood of the colour bin into which they fall. This alleviates the comparatively better results achieved when "conservatively"

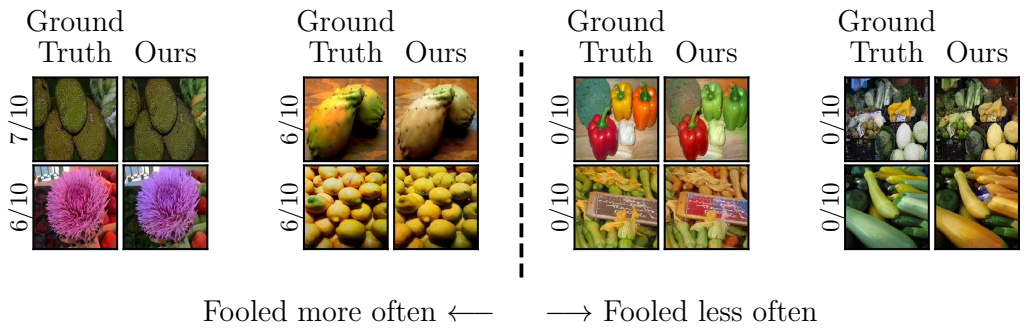

Figure 3: User study results, labels indicate how many volunteers were fooled by the corresponding image.

Table 2: Quantitative coluorization results

Method	Auc	AuC (rebal.)	VGG-16 Top-5 Acc.
Ground truth	1	1	92.1
Grayscale	79.9	68.3	46.4
Random colour	79.0	68.2	17.4
Zhang et al.	83.8	78.6	67.7
Ours	85.5	81.0	81.0

predicting images with low saturation due to the prevalence of pixels with low saturation in natural images.

Additionally, in order to show that our results are not only visually appealing but also suitable as inputs to classification networks trained on colour images, we tested whether using our network as a preprocessing step results in improved top-5 classification accuracy for a VGG-16 [16] network trained on the complete ImageNet training set.

We calculated the average AuC and top-5 classification accuracy for several image set variants. To this end, we first randomly sampled a thousand images belonging to twenty synsets from the ImageNet validation set. The twenty synsets are a subset of the sysets from which we composed the training set. We then also converted these images to greyscale and reconstructed their colour channels using both the trained network provided by Zhang et al. as well as our trained network. To show that the semantic colourization learned by both networks is superior to arbitrary (spatially consistent) colourization, we also colourized the greyscale images by treating them as the L channel of an image in the Lab colour space and appending the a and b channels from randomly chosen ImageNet training set images.

Table 2 lists the average AuC obtained on the different dataset variants. We find that, as expected, relative improvement in AuC for colourized images compared to greyscale images is greater when applying rebalancing.

Also listed are the top-5 classification accuracies we obtained. We find that colourization significantly improves classification performance. This effect is more pronounced for our network. We believe this is due to the smaller number of distinct objects depicted in the reduced training set which allows our network to mostly assigns very plausible colours to these primary objects despite reduced training time. This is likely to have a larger positive influence on classification than the plausible colourization of background objects.

Figure 4 displays classification accuracy achieved for images colourized using our network, relative to greyscale images, averaged over each of the twenty synsets present in the test set. Colourization improves average accuracy in all twenty cases, especially when accuracy on greyscale images is low to begin with.

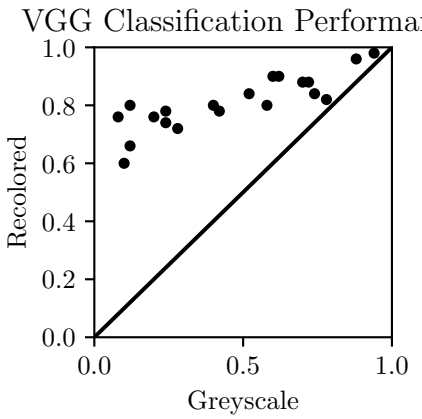

Figure 4: Classification performance comparison for greyscale images before and after colourization by our network, averaged for twenty different synsets.

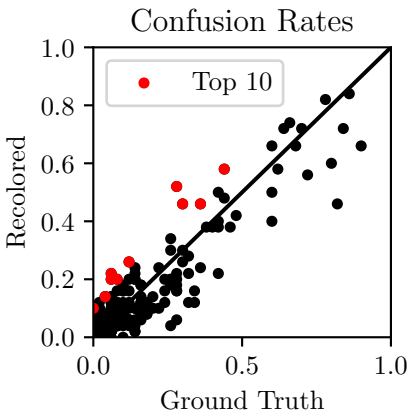

Figure 5: Distribution of off-diagonal confusion matrix entries.

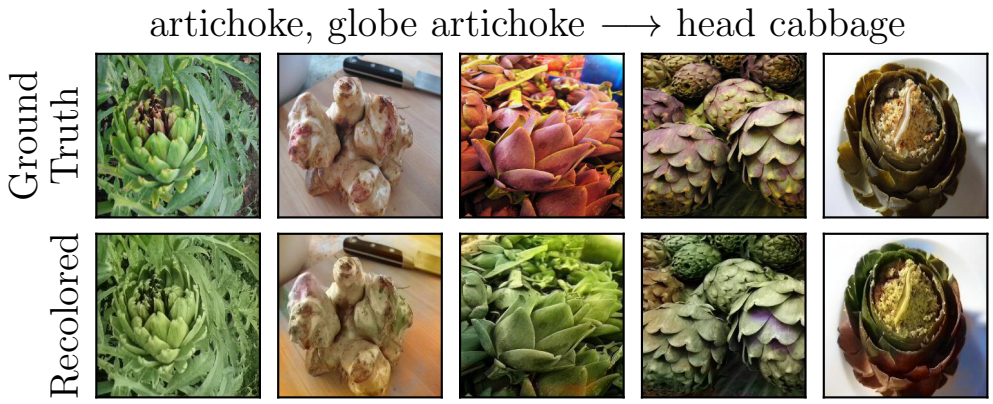

Figure 6: Confusion case most strongly amplified by colourization, note that both classes of objects are visually similar to begin with and a slight shift towards a "greener" palette is enough to cause the confusion to occur.

For some synsets colourization can amplify the occurence of certain confusions cases during classification. To quantify this effect, we calculated the top-5 confusion matrices $C_{groundtruth}$ and $C_{colourized}$ with entries $C_{i,j}$ equal to the proportion of images with ground truth label i for which j is one of the five labels predicted as most likely. Figure 5 displays the off-diagonal elements of $C_{colourized}$ plotted over the corresponding elements of $C_{groundtruth}$ with those for which the difference $C_{colourized} - C_{groundtruth}$ is largest highlighted.

Clearly, colourization makes certain confusion cases more and others less likely. Figure 6 visualizes the confusion case for which colourization results in the largest increase of the corresponding confusion matrix entry. The images shown are those for which the confusion occurs for those colourized from greyscale images but not for the corresponding ground truth images.

6 Conclusions

Our work validates the image colourization approach employed by Zhang et al. The network that we trained from scratch is able to determine suitable colours for the most common fruits and vegetables, as proven by our user-study. This suggests that our network has implicitly learnt to classify/segment

input images. It is thus feasible that the representations learnt by our network could be useful in transfer learning tasks. Zhang et al. explore this possibility in the original paper by successfully training linear classifiers and segmentation networks on top of their trained colourization network.

However the network often fails in colourizing backgrounds objects: it tends to produce coloured spots which are an obvious giveaway of artificial colouring. To mitigate this effect, it would be interesting to explore post-processing approaches based on enforcing spacial consistency or to pre-process images via explicit segmentation into fore- and background and use different networks to colourize both.

The measured improvement in classification accuracy for images which have undergone the colourization process suggests that our network could be useful as a preprocessing step in greyscale image classification tasks.

References

[1] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*.

[2] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

[4] Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller. Transferring color to greyscale images. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '02, pages 277–280. ACM.

[5] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 689–694. ACM.

[6] A. Deshpande, J. Rock, and D. Forsyth. Learning large-scale automatic image colorization. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 567–575.

[7] Z. Cheng, Q. Yang, and B. Sheng. Deep colorization. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 415–423.

[8] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *European Conference on Computer Vision (ECCV)*.

[9] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. 35(4):110:1–110:11.

[10] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. Image colorization using generative adversarial networks. In Francisco José Perales and Josef Kittler, editors, *Articulated Motion and Deformable Objects*, pages 85–94. Springer International Publishing.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks.

[12] Mingming He, Dongdong Chen, Jing Liao, Pedro V. Sander, and Lu Yuan. Deep exemplar-based colorization. 37(4):47:1–47:16.

[13] K McLAREN. The development of the cie 1976 (l*a*b*) uniform colour-space and colour-difference formula. 92:338–341.

[14] George A. Miller. Wordnet: A lexical database for english. 38(11):39–41.

[15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*.

[16] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.