Multi-Task Image Colorization and Classification

Filip Anjou Stanislaw Kolodziejczyk

***Abstract*—We employed multi-task learning using generative adversarial networks to evaluate how the task of image colorization is affected when also performing classification. We train three models: two single-task models performing each of the two tasks, as well as a multi-task model that performs both tasks. The multi-task model produces slightly more convincing colorized images than the single-task counterpart, and is a bit more robust against new input on the colorization task compared to the single-task colorization model. However, it overfits on the classification task, thus achieving a slightly lower validation accuracy than the single-task classifier.**

I. Introduction

This work aims to examine the effects of applying multi-task learning to the task of image colorization and classification. Specifically, we train and compare the performance of a model trained to perform both tasks with models trained to perform each task individually. We then investigate whether the multi-task model becomes better at any of the two tasks than the corresponding single-task model.

All models are trained on the same subset of the ImageNet (ILSVRC) dataset [1] providing both color images for the colorization task and labels for the classification task. To reduce the scope of the project, only five classes are used, namely `lionfish`, `pufferfish`, `sea-anemone`, `sea-cucumber`, and `sea-snake`. The training set contains 1 300 images per class and thus 6 500 samples in total. The validation set consists of the same five classes taken from the ImageNet validation set, and contains 50 samples per class for a total of 250 samples. To accommodate the smaller training set, and to reduce training time, all three models utilize transfer learning with a ResNet-18 [2] feature extractor network pre-trained on the whole ImageNet dataset.

This work is based on a previous work on image colorization with generative adversarial networks [3].

II. Background

The task of applying deep learning to image colorization has seen various approaches in previous work, from using plain feed forward neural networks [4], to different flavors of generative adversarial networks (GANs), to vision transformers [5], and diffusion models [6].

GANs consist of two models—a *generator* and a *discriminator*—each with its own sub-task. The generator is tasked to produce synthetic samples that mimic the distribution of the training data, while the discriminator is tasked with determining the probability that a given sample is authentic, that is whether it is sampled from the training data or is synthesized by the generator [7]. The models are trained in tandem, with the performance of the discriminator directly affecting the loss and subsequent gradients used to update the generator, and vice versa. This makes the GAN approach appealing for image colorization since the loss feedback adapts to the performance of the model producing the images. By comparison, using L1 loss alone to compare the generated output directly to the ground-truth tends to result in desaturated images [5]. Another helpful metric is Fréchet Inception Distance which aims to quantify the visual similarity between two sets of images [8].

The generator is typically conditioned on a latent variable sampled from a distribution of noise during training, which is a critical component enabling the generator to produce different outputs given the same set of parameters. It serves as a way to sample the distribution modeled by the generator. A common variation of this scheme is the *conditional GAN* (cGAN) [9], which involves conditioning both the generator and the discriminator on some additional piece of information, such as an image. In the field of image colorization, this would typically be the grayscale image that is to be colorized.

Finally, previous work on multi-task learning (MTL) in the context of facial landmark detection [10] has shown that the addition of related sub-tasks during training results in a model that is more robust against variations in the input.

III. Method

A. Data pre-processing

All images are scaled to fit a target resolution of 256×256 pixels, and are converted to the CIELAB color space, allowing for easy separation between luminance information ($L*$ channel) and color information ($a*$ and $b*$ channels). The images are then normalized such that pixel values for each channel are within the interval $[-1, 1]$. To address the small training set, data augmentation is used in the form of random horizontal flipping with a probability of 50%.

B. Model architectures

The single-task classification model consists of a pre-trained ResNet-18 feature extractor without its final two layers, with a new head consisting of two fully connected layers. The former has 1 000 neurons and is followed by a ReLU activation, while the latter has five neurons and uses softmax activation.

The single-task colorization model consists of a convolutional autoencoder with a pre-trained ResNet-18 as the encoder. The decoder is composed of blocks of transposed convolutions, ReLU activations, and batch normalization, such that its output resolution matches the input resolution of the encoder, but with two channels instead of one. Its final layer uses Tanh activation to restrict the outputs between -1 and 1. The autoencoder takes a single grayscale image as input and predicts color information for each input pixel in the form of the two color channels in the CIELAB color space. The model is trained in a cGAN scheme, with the colorization autoencoder as the generator, and a CNN patch discriminator which subdivides its input image into a grid of 30×30 patches, and predicts the authenticity of each patch [11]. Instead of conditioning the generator on a latent variable, dropout is used in the decoder (the colorization head).

The multi-task model is nearly identical to the single-task colorization model, except with a separate classification head identical to that of the single-task classifier forking out of the generator's bottleneck. It, too, is trained using a cGAN scheme as depicted in `Fig.1`.

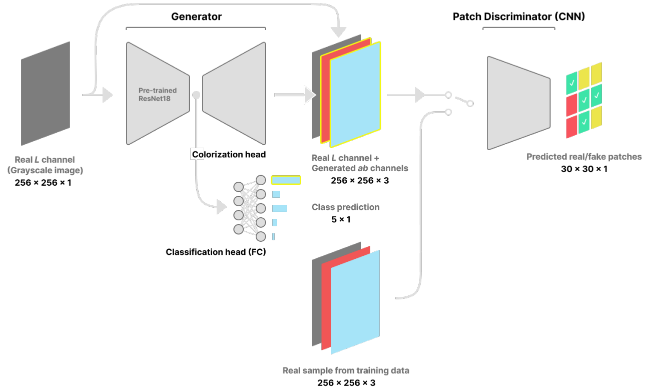
Generator
Patch Discriminator (CNN)
Pre-trained ResNet18
Real L channel (Grayscale image) 256 × 256 × 1
Colorization head
Real L channel + Generated ab channels 256 × 256 × 3
Predicted real/fake patches 30 × 30 × 1
Class prediction 5 × 1
Classification head (FC)
Real sample from training data 256 × 256 × 3

Fig. 1. Multi-task model training setup

C. Loss functions

The classification task uses regular categorical cross-entropy. For the colorization task, a combined loss is used in the form of a weighted sum of L1 loss and adversarial (GAN) loss. To give both terms about equal influence over the combined loss, the L1 loss is scaled by a factor of 100. The classification loss in the multi-task model is scaled by a factor of 10 to bring it within the range of the combined colorization loss. This aims to balance the importance of the two tasks evenly. The discriminator used L1 loss, comparing each patch's prediction with the true label.

D. Training

All models are first trained for 30 epochs with the weights of their respective ResNet-18 base frozen. These weights are then unfrozen and the models are trained for 15 more epochs. The Adam [12] optimizer is used for all models, with a learning rate of $2 \cdot 10^{-4}$ in both phases. All hyper-parameters are the same as in [3].

Each iteration of cGAN training consists of two steps. First, the discriminator is updated by predicting the authenticity of a batch of samples from the training data, all labeled as *real*, and the corresponding colorized images from the generator, labeled as *fake*. The losses on the two sets of images are averaged. Second, the generator is updated by feeding those same colorized images into the updated discriminator to produce an adversarial loss, as well as comparing them directly to the ground-truth with L1 loss. The two losses are weighted and summed. In the multi-task case, the generator also performs class predictions, with the corresponding loss being weighted and added to the total loss.

Validation accuracy and losses are computed and logged twice per epoch.

IV. Results

Subset	Model	Accuracy (%)	L1 loss
Train	Single-task	96.4	10.9
Train	Multi-task	98.0	11.1
Val	Single-task	84.8	13.3
Val	Multi-task	81.6	11.6

TABLE I
QUANTITATIVE COMPARISON OF MULTI-TASK AND SINGLE-TASK MODELS

A. Colorized images

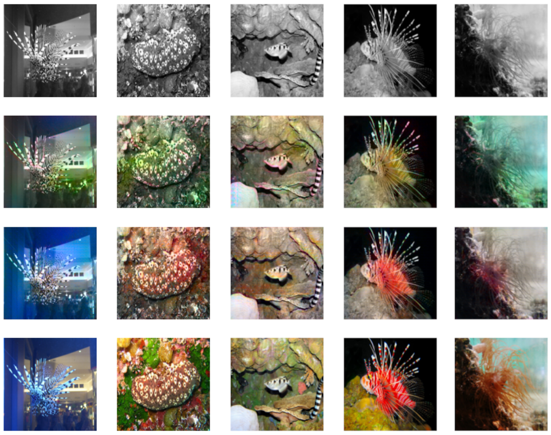

Fig. 2. Colorized batch of images from the validation set. From top to bottom, the rows contain the grayscale input, the single-task colorization, the multi-task colorization, and the ground-truth.

B. Training and validation losses

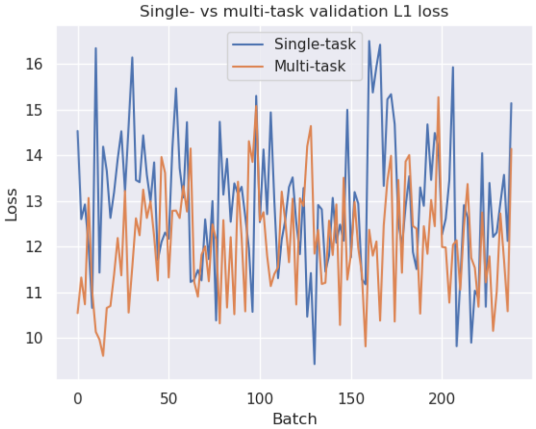

Fig. 3. Comparison of L1 losses between multi-task and colorization models

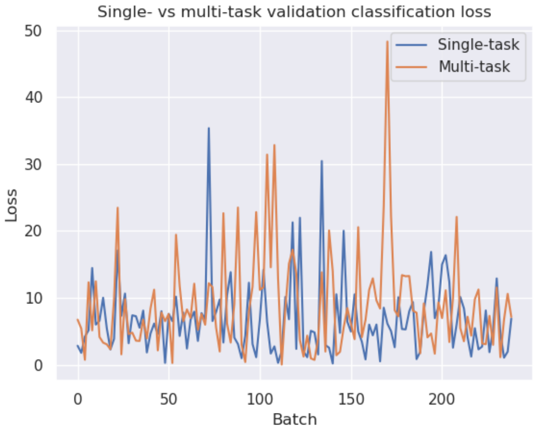

Fig. 4. Comparison of classification losses between multi-task and single-task models

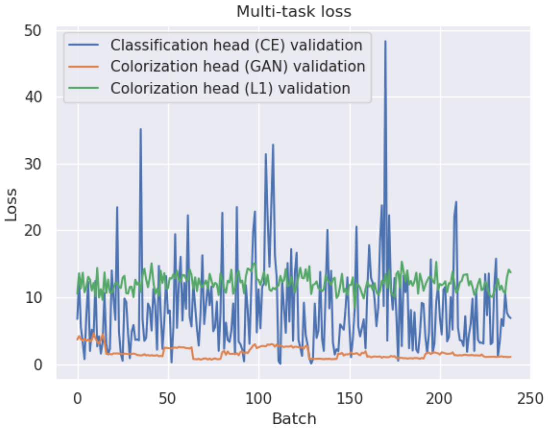

Fig. 5. Losses of multi-task model on validation set

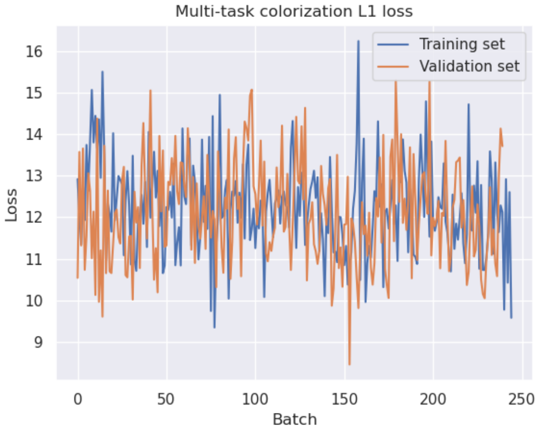

Fig. 6. Comparison of multi-task model L1 loss on training and validation set

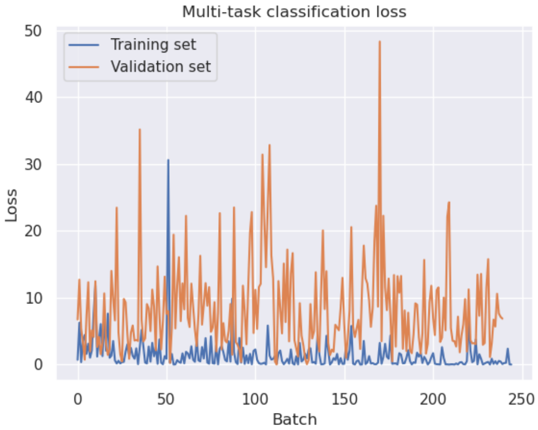

Fig. 7. Comparison of multi-task model classification loss on training and validation set

V. Discussion

A. Discussion of results

Visual comparison of colorization results. When assessing performance on the colorization task, a crucial aspect is visual perception by humans. While L1 loss provides valuable insight into a model's ability to reproduce ground-truth color, this might not coincide with how human perception determines the similarity of images. The use of a GAN architecture plays a vital role in this task leading to more realistic and colorful generated images. Although `Fig.3` does not indicate much lower L1 for the multi-task model than the single-task one, `Tab.I` shows a slightly lower L1 on the validation set, and its outputs are indeed a bit more appealing to a human eye and look more realistic, as shown in `Fig.2`.

Dataset complexity and sealife diversity. The choice of using underwater animals may have inadvertently resulted in the colorization task becoming more difficult. Many images are captured with the camera close to the subject, thus providing very little context about the surroundings. This context seems important since the colors of the subject depend on information such as the depth at which the photo is taken. Photos taken in shallower waters tend to have a light-blue tint which those taken in deeper waters do not.

`Fig.6` shows that the model has significant bias and negligible variance, judging by the lack of a difference in L1 loss between the training and validation set. This indicates that a black-and-white image often is an insufficient indicator of the exact colors of sea animals which have many variations of complex patterns and colors. This makes colorization remarkably challenging and proves that the sea animals dataset is complex and poses challenges for the model.

Classification task overfitting. `Fig.7` clearly indicates large overfitting by the multi-task model on the classification task. The model exhibits large variance, with the loss on the validation set is considerably larger than the loss on the training set. This may be caused by the difference in complexity between the two tasks and the lack of task-specific early stopping. Assuming classification is easier, the classification head is expected to converge and eventually overfit before the colorization head has converged.

B. Future improvements

A few shortcuts were taken due to technical challenges and time constraints, opening the door for further improvements. The most important of which are discussed below.

Larger, more diverse, dataset. While using a pre-trained base model allows the task-specific training set to be small, using a larger and more diverse set of classes from the ImageNet dataset would likely prove beneficial for the colorization task. The training set is currently restricted in size by the fact that all images require labels to facilitate simultaneous classification. However, since colorization can be performed in a self-supervised manner, it would be possible to train on a much larger corpus of images.

Using a larger training set with more samples would also allow for a more rigorous filtering of the training images. This could involve excluding images that differ far from the target dimensions, or that are already black-and-white (of which at least one was found in the current training set).

U-Net architecture. The accuracy on the colorization task may be improved with the use of an established state-of-the-art autoencoder architecture instead of a custom one. The U-net architecture, which was designed to be trained on small datasets with liberal use of data augmentation [13], is a promising candidate. Like ResNets, U-nets have skip connections between early and late layers which could result in faster convergence.

Improved loss weighting. The loss functions for the single-task colorization model and the multi-task model both consist of weighted sums of terms whose weights are hand-picked to reach a uniform scale. However, de-emphasizing the L1 loss by reducing its weight may result in more plausible-looking colorized images, at the expense of them more frequently having incorrect colors. The performance of multi-task models has been shown to depend heavily on these weights [14], suggesting that learning them instead during training may result in even better performance.

VI. Conclusion

While the results of our experiments are somewhat inconclusive, qualitative examination suggests that a multi-task model tends to produce slightly more natural-looking images. This difference is not immediately apparent when comparing L1 scores, emphasizing the importance of using more appropriate evaluation metrics that quantify the *plausibility* of a colorization, such as Fréchet Inception Distance. The multi-task model showed some robustness in the colorization task, with the L1 loss differing slightly less between the two subsets than for the corresponding single-task model.

References

[1] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[2] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[3] Moein Shariatnia. *Image Colorization with U-Net and GAN Tutorial*. 2022. URL: https://colab.research.google.com/github/moein-shariatnia/Deep-Learning/blob/main/Image%20Colorization%20Tutorial/Image%20Colorization%20with%20U-Net%20and%20GAN%20Tutorial.ipynb.

[4] Zezhou Cheng, Qingxiong Yang, and Bin Sheng. *Deep Colorization*. 2016. arXiv: 1605.00075 [cs.CV].

[5] Shanshan Huang et al. "Deep learning for image colorization: Current and future prospects". In: *Engineering Applications of Artificial Intelligence* 114 (2022), p. 105006. ISSN: 0952-1976. DOI: https://doi.org/10.1016/j.engappai.2022.105006. URL: https://www.sciencedirect.com/science/article/pii/S0952197622001920.

[6] Chitwan Saharia et al. *Palette: Image-to-Image Diffusion Models*. 2022. arXiv: 2111.05826 [cs.CV].

[7] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[8] Martin Heusel et al. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium*. 2018. arXiv: 1706.08500 [cs.LG].

[9] Mehdi Mirza and Simon Osindero. *Conditional Generative Adversarial Nets*. 2014. arXiv: 1411.1784 [cs.LG].

[10] Zhanpeng Zhang et al. "Facial Landmark Detection by Deep Multi-task Learning". In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Cham: Springer International Publishing, 2014, pp. 94–108. ISBN: 978-3-319-10599-4.

[11] Phillip Isola et al. "Image-to-Image Translation with Conditional Adversarial Networks". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5967–5976. DOI: 10.1109/CVPR.2017.632.

[12] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by Nassir Navab et al. Cham: Springer International Publishing, 2015, pp. 234–241. ISBN: 978-3-319-24574-4.

[14] Alex Kendall, Yarin Gal, and Roberto Cipolla. *Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics*. 2018. arXiv: 1705.07115 [cs.CV].