U-Net, Deep CNN, and Conditional GAN for Auto-Colorization

Wanjia Fu Zixuan Guo Linghai Liu Matianyu Zang

BROWN

CSCI 1430

Motivation

Colorizing black-and-white images has a wide range of applications in various fields. It helps to revive historical images, enhance medical imagery, and can even be used for artistic purposes. Colorizing grayscale images helps to discover more information. Finding an effective and efficient technique of auto-colorizing grayscale images is still a challenging task in the field of computer vision today.

Fig.1 Examples of results from auto-colorization models

Define the problem

Various models have been proposed for the task of auto-colorization. Their relative strengths and weaknesses, however, still remain to be investigated. Some popular approaches include using Convolutional Neural Networks (Deep CNN) and Generative Adversarial Networks (GANs), which have been widely tested and adopted. Among them, the U-net model, the Deep CNN model and the Conditional GAN model are three examples that caught our attention for their highly-rated performance on the auto-colorization task.

Goal

Our aim for this project is to:

1. Propose, build and compare these three models and their performance on the same dataset.
2. By comparing the training process and the results of each model, we hope to gain insights into their strengths and weaknesses and provide guidance on the best approach to use for this task.
3. In addition, we hope to gain a better understanding of the various factors that affect the performance of a model on auto-colorization tasks.

Notes

Dataset

We use two different datasets, the first has 5000 grayscale and RGB images on a wide range of subjects, including animals, scenery, daily objects etc, and about 700 grayscale and RGB image pairs for testing.
The second dataset is a 723-image dataset with grayscale and RGB images of fruits or vegetables from 20 different categories. Each image has a centered object on a white background.

Observations

On the first dataset: U-net performed the best. Our adaptation of Deep Koalarization [1] results in some colorization in addition to blurring the image. CGAN does not perform quite well on the testing dataset, but reached fair performance after replacing its generator with our trained U-net.
On the second dataset: deep koalarization produces completely white images, which we believe might be attributed to the large patches of white in the original gray-scale images and ground-truth color images. Some green vegetables are colorized as red by U-net and CGAN, which might result from the uneven distribution of the colors of vegetables in the training dataset.

Methodology

Deep-Koalarization [1]

Encoder The encoder consists of a series of convolution layers. ReLU is used as activation.
Fusion The fusion layer combines image features extracted by a pre-trained VGG16 model with the encoded input.
Decoder The decoder first uses a convolution layer with kernels of size 3 to map the fused input to 128 channels. The number of channels goes down from 128 to 64, 32, and 3. ReLU is used as activation, except for tanh used before the last upsample layer.
Loss Mean Squared Error between predicted RGB and ground truth.

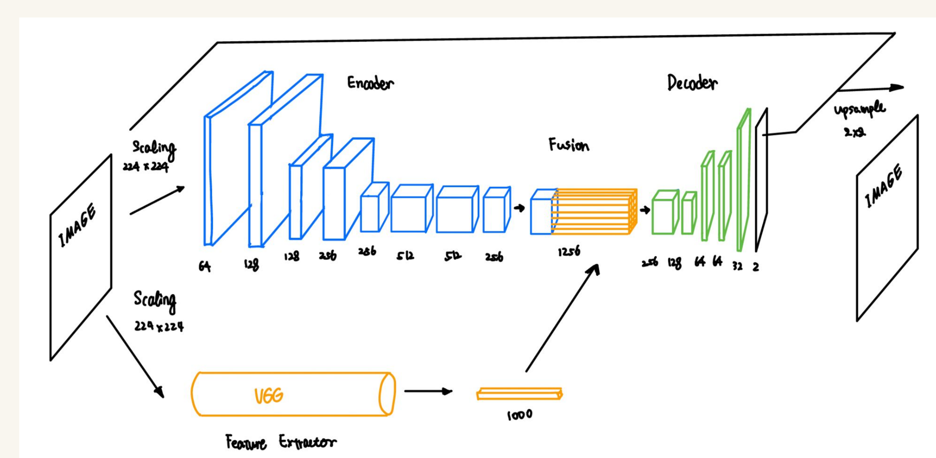


Fig.2 Model Architecture Diagram for Deep-Koalarization

U-net

We use the almost the same U-net model as proposed by O. Ronneberger et al. [2], except that we made the modification that the output has three channels (RGB) rather than the 'a' & 'b' channels in 'LAB' Color Space. Our implemented U-net uses Mean Squared Error as its loss function.

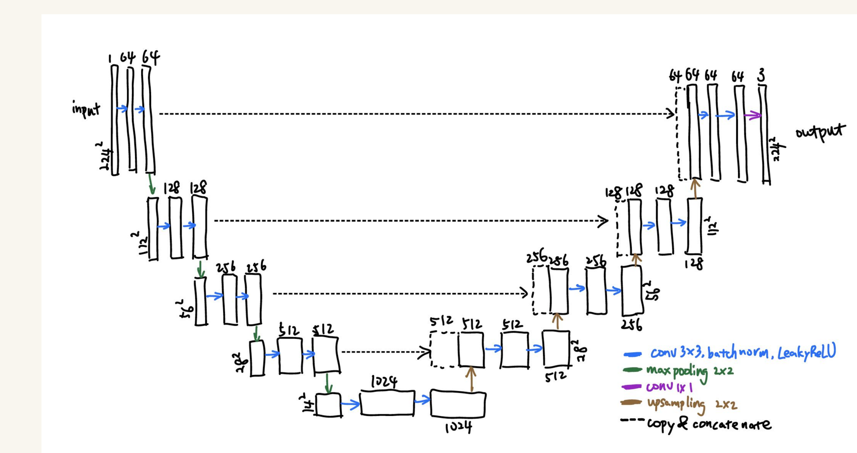


Fig.3 Model Architecture Diagram for U-net

Conditional GAN

Generator We use our implemented U-net.
Discriminator The discriminator takes in both a grayscale image and its corresponding colored image, be it the ground truth or generated by the generator, and predicts a probability that the colored image does correspond to the grayscale image.
Loss We use GAN loss and also incorporated L1 loss from y to y_pred.
Numerical Stability During training, we encounter severe numerical instability when using the original GAN loss. Hence, we introduced a buffer term ε.

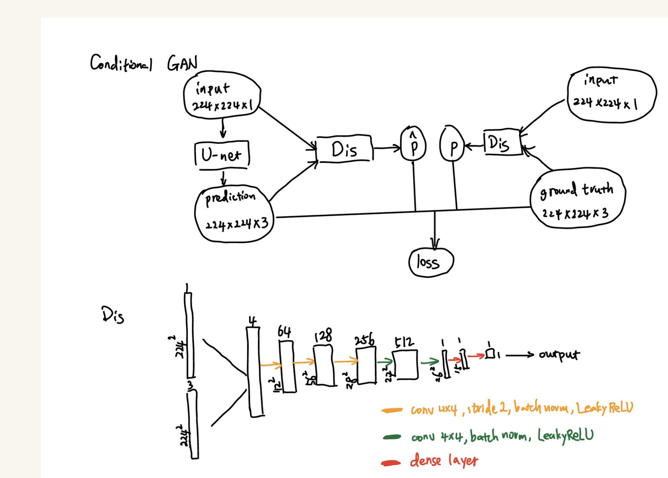
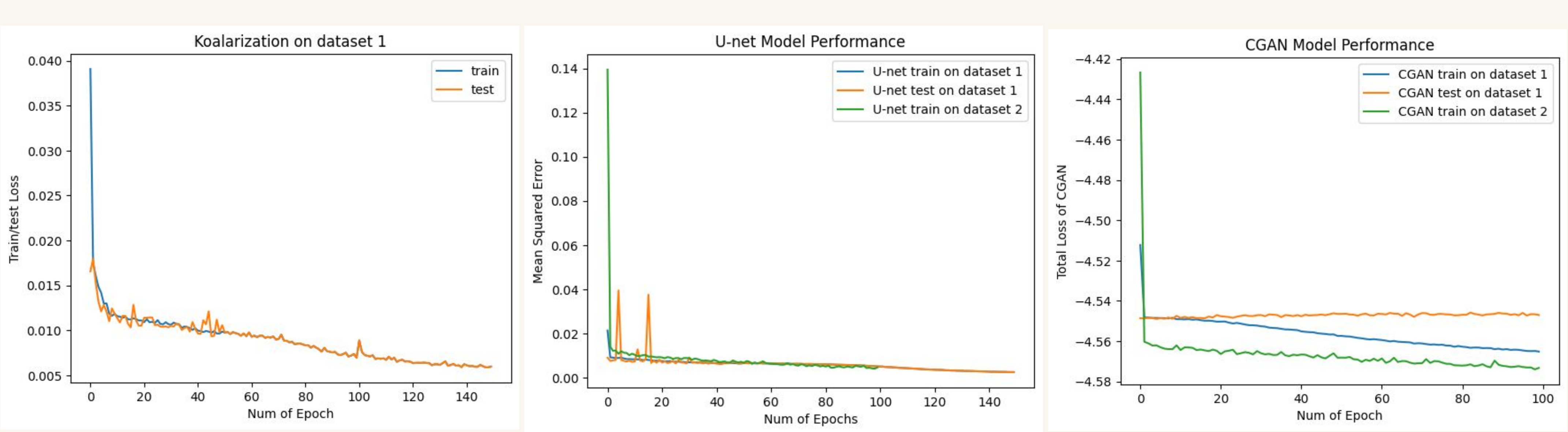


Fig.4 Model Architecture Diagram for Conditional GAN

Results (images/figures)

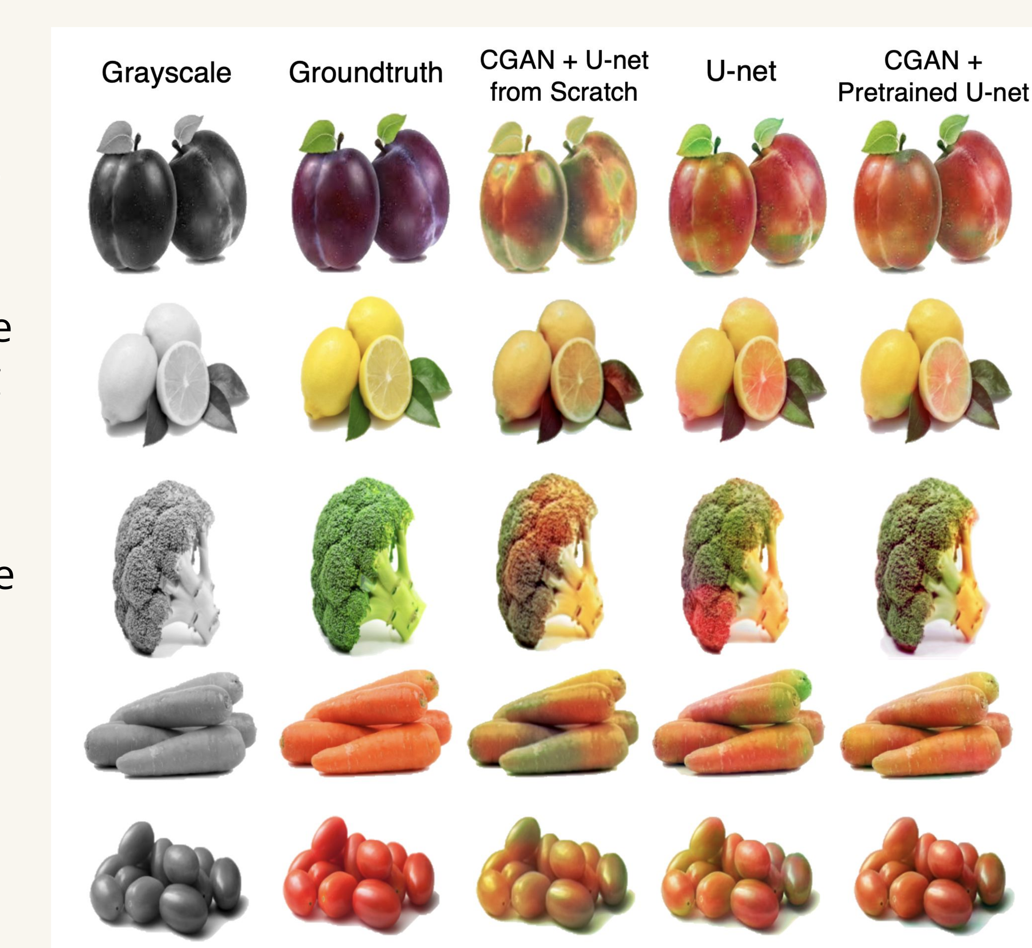


Loss Plots for three models. Our adaptation of Deep Koalarization does not converge, so the loss is omitted. For conditional GAN in the plots, the generator parts are U-net shown in the middle panel. If the CGAN's are trained from scratch, the loss would be much lower, but the contributions are not mainly by minimizing the L1 loss.
We trained Koalarization and U-net on dataset 1 for 150 in total, but we first trained for 50 epochs and continued training for the rest 100 epochs, so there may be some spikes around epoch #50. For all models (Deep Koalarization, U-net, and CGAN) on dataset 2 and all CGAN models, we trained for 100 epochs and record training losses only.

More results

Left: Our results on dataset 1. The performance of all three models are not satisfactory, the models learnt limited colors such as green, brown, and blue, and are not able to color complex objects. We can conclude that the models gain basic understanding of the underlying patterns.

Right: Our results on dataset 2. We observe that the performance is much better. They successfully categorize the objects and give out vibrant colors that almost correspond to the ground truth. Among them, CGAN with our pre-trained U-net performs the best overall.



References

[1] Federico Baldassarre, Diego Gonzalez Morin, and Lucas Rodes-Guirao. Deep koalarization: Image colorization using cnns and inception-resnet-v2, 2017
[2] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention (MICCAI), volume 9351 of LNCS, pages 234–241. Springer, 2015.

Acknowledgements

Parts of the experiments are conducted using the computational resources and services provided by the Center for Computation and Visualization, Brown University.