AUTOMATIC COLORIZATION

A SEARCH SPACE ODYSSEY

Kartik Sawhney & Mahesh Agrawal

BACKGROUND

Given a grayscale image as input, produce a plausible/realistic colorization of the image

No ‘correct answer’ since car can be red or green -> Multimodal Problem

This makes it interesting and challenging at the same time

Diverse applications from aesthetic image/video restoration to image enhancement

Contributes to the field of self-supervised learning

PROBLEM STATEMENT

Compare different architectures, models and approaches to obtain the most appealing colored image for the input grayscale image

Use a mix of Quantitative Measures (L2 Loss) and Qualitative Measures (Colorization Turing Test)

DATASET

Trained the baseline model on CIFAR-10

Moved to Imagenet 64*64 and 32*32 for main models

~10k images for training and ~1k test images

Imagenet > CIFAR-10 because more varied images so model can learn better colorization schemes

APPROACH

Implement different architectures, models and loss functions, and train on different datasets to compare the different approaches being used for auto colorization

METHODS & ALGORITHMS

Implemented a similar CNN architecture as in "colorful image colorization" by Zhang et al.
Experimented with different loss functions: MSE/L2 Loss, Smooth L1 Loss, Cross Entropy Loss
Implementing the conditional GAN as in "Image-to-image Translation with Conditional Adversarial Networks", experimenting with architectures and loss functions
As a stretch goal, we hope to implement the CycleGAN as used in Zhou et al.

THE WORKINGS

CIE Lab color space: L channel gives lightness (0-100) and a/b channel color (-127 to 127 in value)

The model from L channel (N * C * H *1) to ab channels (N * H *W *2)

We provide the ground truth ab channels to compute pixel wise loss on the predicted ab channel

CNN

CNN learns what features in L channel map to what ab channel colorization

GAN

Generator is an encoder/decoder consisting of a CNN that includes convolution followed by deconvolution layers. We feed the Generator the L channel and some noise. Discriminator is similar architecture to our CNN architecture

MOTIVATING MODEL | OUR MODEL

The Motivating Model (Colorful Image Colorization by Zhang et al 2016)

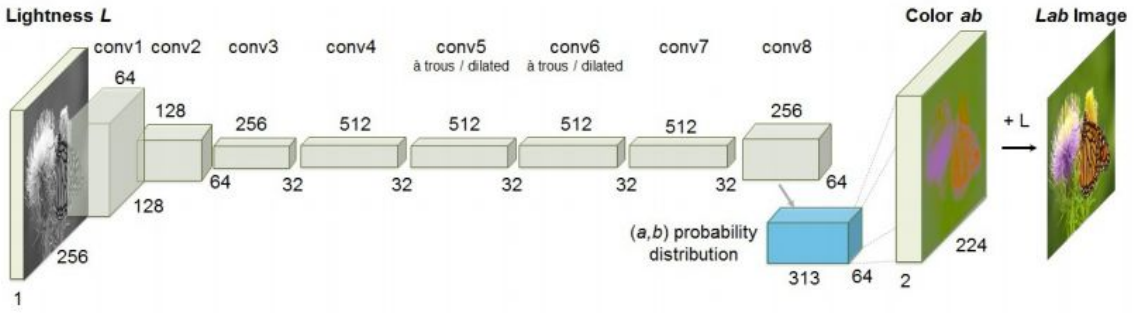


Fig. 2. Our network architecture. Each `conv` layer refers to a block of 2 or 3 repeated `conv` and `ReLU` layers, followed by a `BatchNorm` [30] layer. The net has no `pool` layers. All changes in resolution are achieved through spatial downsampling or upsampling between `conv` blocks.

Our model follows a similar architecture with fewer parameters due to memory/compute constraints

OUR MODEL ->

General Architecture is {Conv2D -> ReLU -> BatchNorm2D} * 6 -> Flatten -> FC Layer

Deep Conv Layers to learn distinct coloring patterns

ReLU gave best Non-Linearity & BatchNorm2D helped train/converge faster

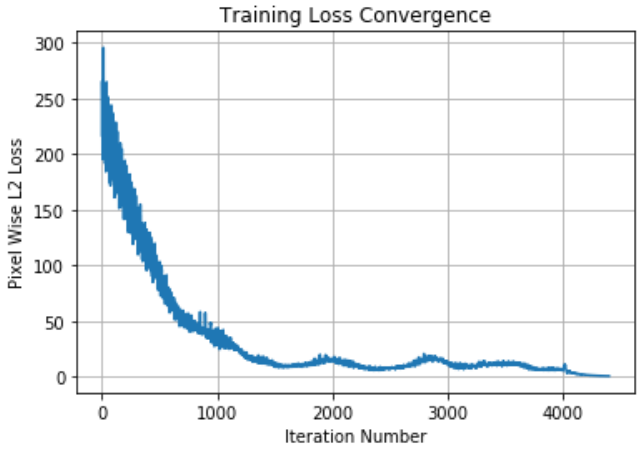
```
Sequential (
  (0): Conv2d(1, 8, kernel_size=(4, 4), stride=(1, 1))
  (1): ReLU (inplace)
  (2): BatchNorm2d(8, eps=1e-05, momentum=0.1, affine=True)
  (3): Conv2d(8, 16, kernel_size=(4, 4), stride=(1, 1))
  (4): ReLU (inplace)
  (5): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True)
  (6): Conv2d(16, 32, kernel_size=(4, 4), stride=(1, 1))
  (7): ReLU (inplace)
  (8): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True)
  (9): Conv2d(32, 64, kernel_size=(4, 4), stride=(1, 1))
  (10): ReLU (inplace)
  (11): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True)
  (12): Conv2d(64, 128, kernel_size=(4, 4), stride=(1, 1))
  (13): ReLU (inplace)
  (14): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True)
  (15): Conv2d(128, 16, kernel_size=(4, 4), stride=(1, 1))
  (16): ReLU (inplace)
  (17): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True)
  (18): Flatten (
  )
  (19): Linear (33856 -> 8192)
)
```

TRAINING LOSS CONVERGENCE

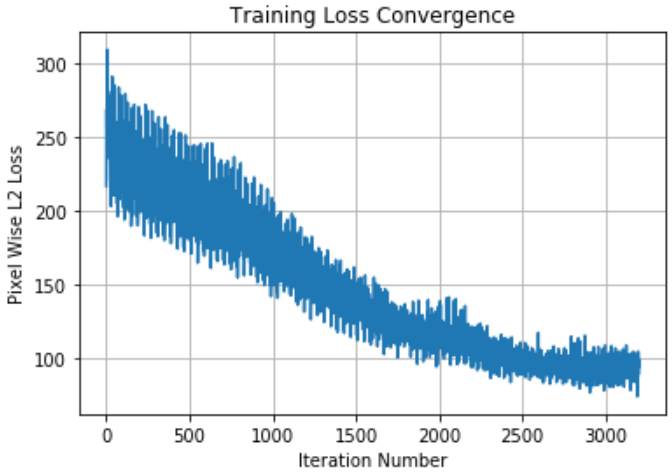
Adam Optimizer: 1e-3 lr with decay to 1e-4 after plateau

Trained for ~100 epochs. Batchsize: ~200

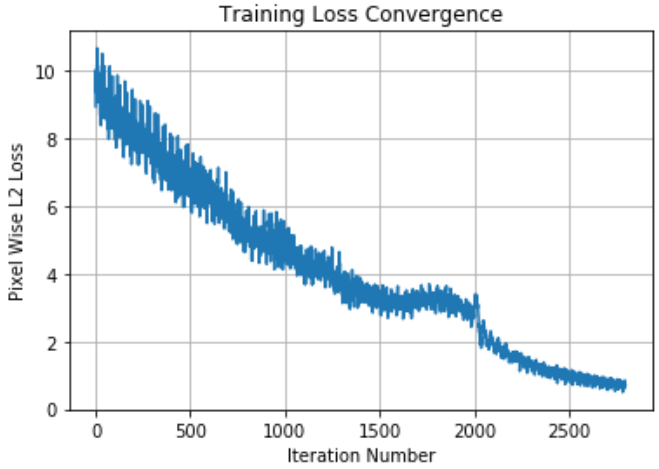
CNN: L2 Loss



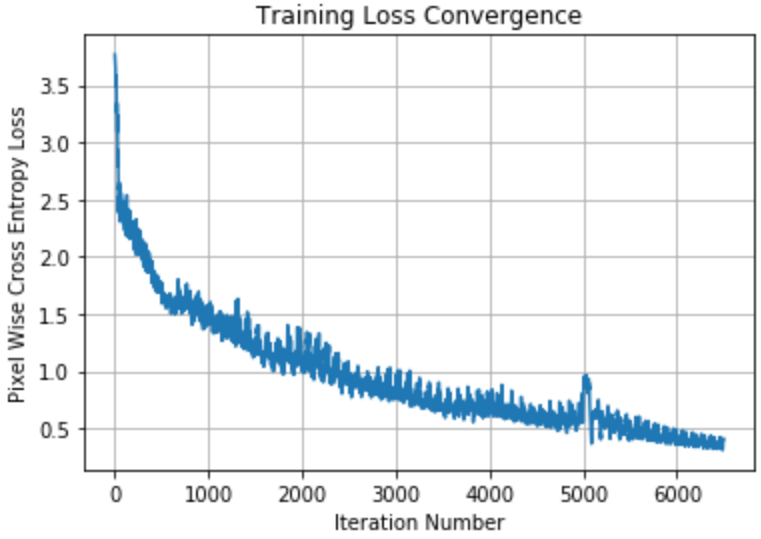
CNN: L2 Loss with Dropout after BatchNorm2D



CNN: Smooth L1 Loss



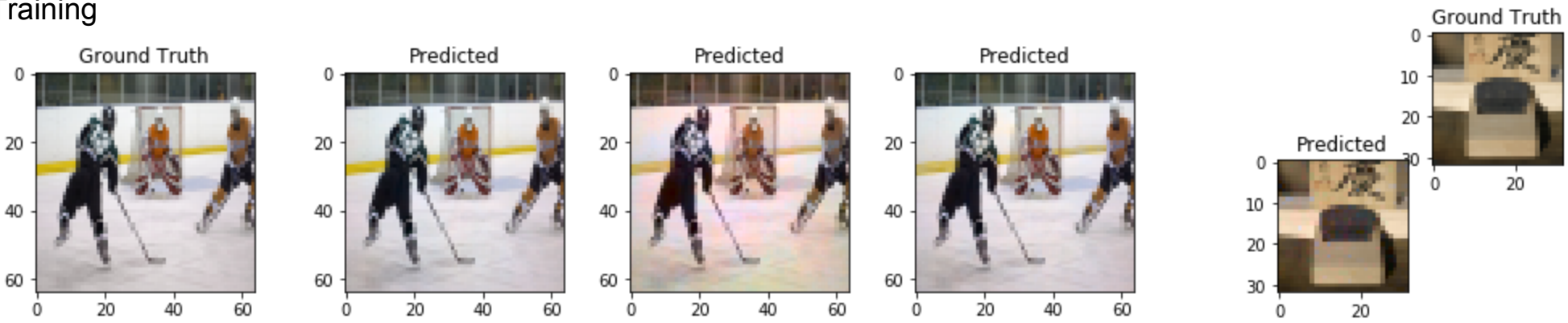
CNN: Cross Entropy Loss



RESULTS

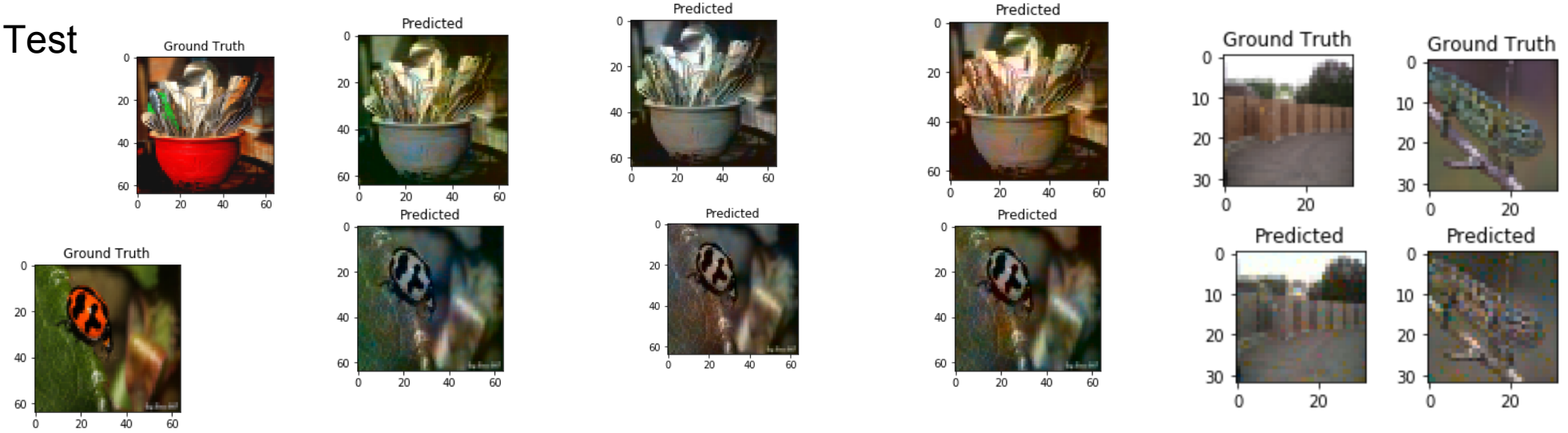
Ground Truth	L2 Loss	L2 w Dropout	Smooth L1 Loss	Ground Truth + Cross Entropy Loss Output

Training



Pixel Wise L2 Loss	0.96	5.46	2.41	1.21

Test



Pixel Wise L2 Loss	20.28 | 13.1	19.4 | 12.04	16.7 | 11.6	2.53 | 3.56

CONCLUSIONS

Deep CNN's seem well suited to the task of automatic colorization

Dropout and Smooth L1 Loss do not perform well in this setting

L2 Loss seems to perform reasonably well but learns the mean coloring leading to distorted outcomes

Cross Entropy and GAN rectify this issue

EXTENSIONS

Models are better with more and varied data for training. State of the art model trained ~1M images

Are certain domains more well suited for this task than others? Nature images generally easier to colorize

Are there other Loss Functions that we were not able to explore that capture the multi-modal property better?

RELATED WORK

Zhang et al., Larsson et al., and Iizuka et al. use deep CNNs, large datasets and novel loss functions to account for the multimodal nature of the problem, giving interesting and rich images

Zhang et al. also approach this problem using conditional generative adversarial networks (CGANs)

Zhou et al. use cycle GANs, thus eliminating the need for input and output image pairs at training time