Automatic Image Colorization

Srinivas Tunuguntla, Adithya Bhat

Problem Definition

Previous Work

- User assisted
- Fully automated

User assisted Scribble based

- User provides scribbles that give reference colors.
- The system extends the colors to the rest of the image.
- However, the colors have to be given as input, they are not learnt.

Photo: Levin et. al (2004)

User assisted
Reference based

- User provides reference image with similar features.
- The system transfers colors to the given image.
- However, providing/selecting suitable reference images is vital to this method.

Photos: Welsh et. al (2002)

Fully Automated

Feature Engineering

- Image features created via pre-processing.
- Feature engineering plays a major role.

Neural Networks

- Convolutional Neural Networks are the current state of the art.
- Features are automatically learned.

Setup

Infrastructure

- Google Cloud
- Nvidia Tesla K80 GPU

Framework

- Keras

Training Data

- Imagenet
    - Geological formation synset

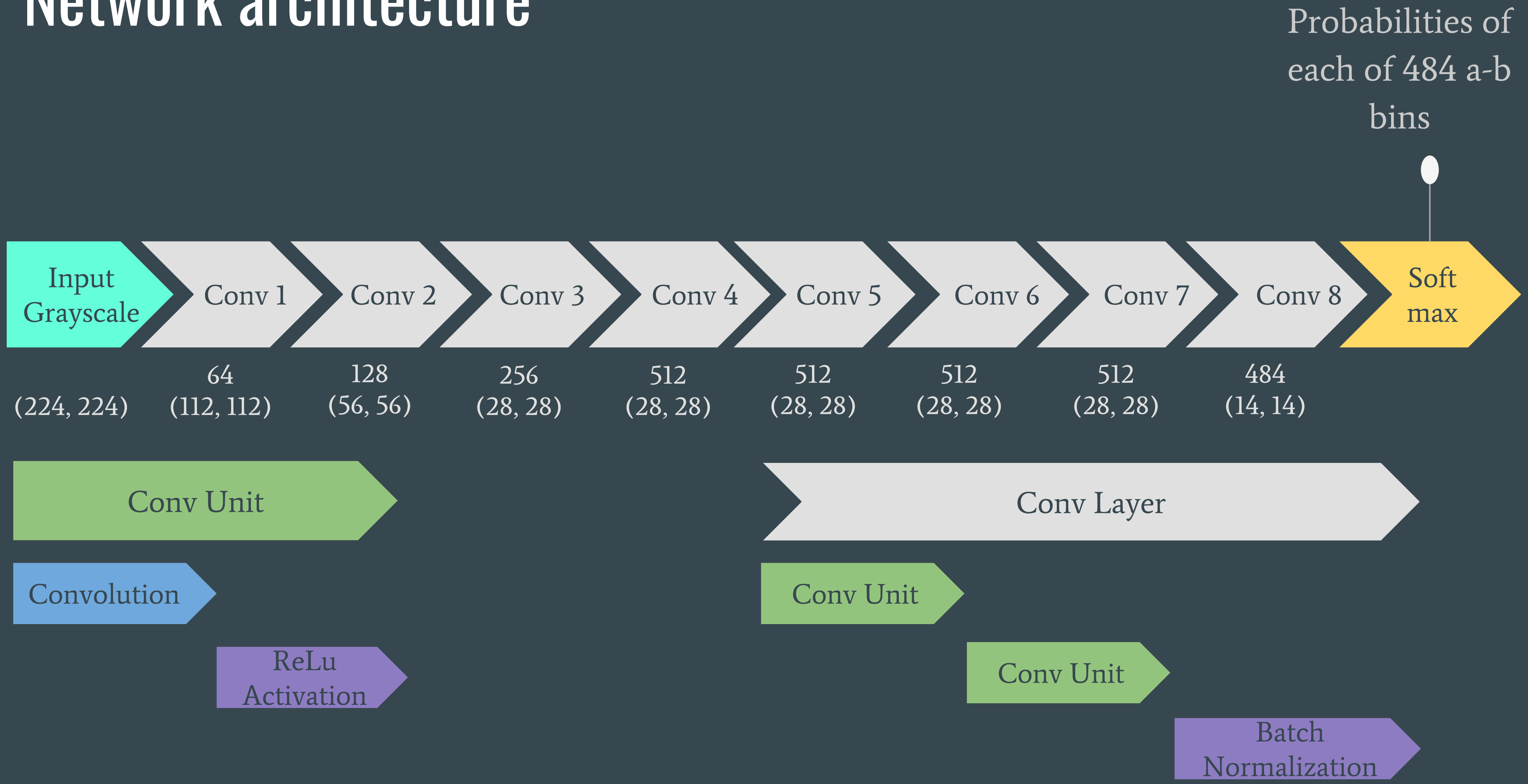
Design

Lab Color Space

- Euclidean distance in the a-b space corresponds to visually perceived differences
- L is the lightness channel
  - 0 - black, 100 - white
- a, b are color components
  - Range from -128 to +127 (or scaled to -100 to +100)

Our task : predict the channels a, b using input L

Network architecture

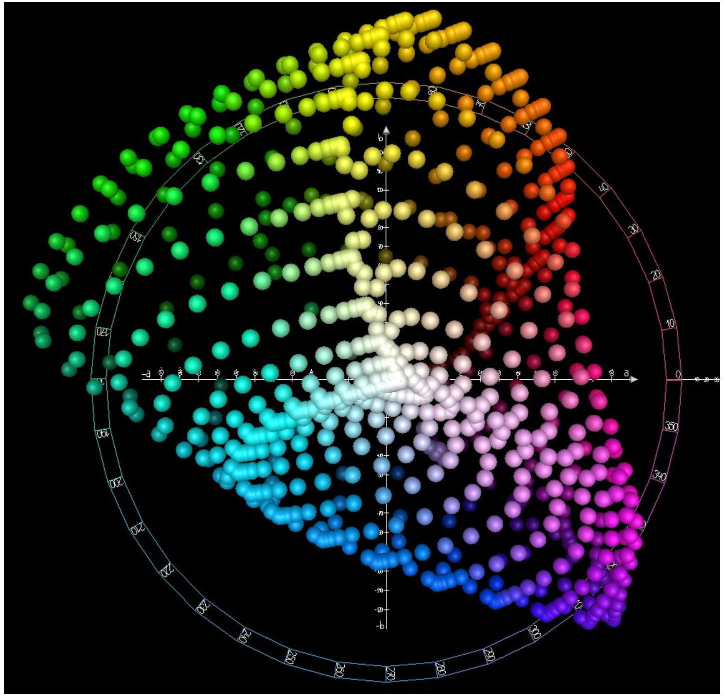


Loss Function
MSE

- Not robust to inherent ambiguity
- Predicts mean of possible colorizations
- Results in desaturated color predictions

Loss Function Multinomial classification

- Divide a, b space into K bins
- Predict probability distribution over these bins

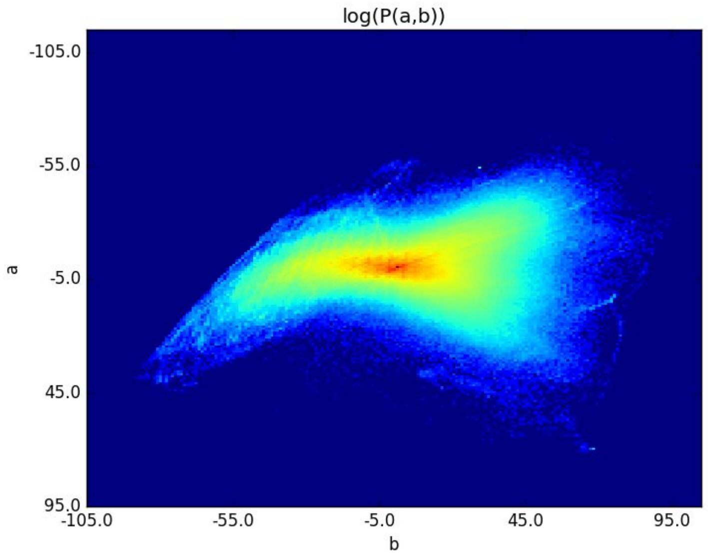


$$Loss(Z, \hat{Z}) = \sum_{p} \sum_{k=1}^{K} Z_{p,k} log(\hat{Z}_{p,k})$$

Problem : Desaturated output

Solution : Color Rebalancing

- Distribution of a,b values strongly biased towards low (desaturated) values
- Reweight each pixel loss based on pixel color rarity
- Integrated into the loss function



$$Loss(Z,\hat{Z}) = \sum_{p} w_{k^*} \sum_{k=1}^{K} Z_{p,k} log(\hat{Z}_{p,k})$$

$$w_k \propto (0.5 f_k + 0.5 \tfrac{1}{K})^{-1}$$

Desaturated output

Color rebalanced

Problem : Color patches

Solution : Segmentation

- Intuition: Neighboring pixels that have similar intensities tend to have similar colors
- Introduce the notion of spatial locality into the loss function

$$Loss \mathrel{+}= \lambda \left(\sum_{p} \left(\hat{a}_p - \sum_{q \in N(p)} w_{pq} \hat{a}_q\right)^2 \right)$$

$$Loss \mathrel{+}= \lambda \left(\sum_{p} \left(\hat{b}_p - \sum_{q \in N(p)} w_{pq} \hat{b}_q\right)^2 \right)$$

$$w_{pq} \propto e^{-(L_p - L_q)^2}$$

Color patches

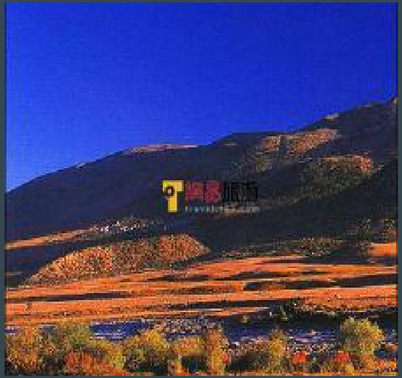
Segmented

Results













Failure cases

Future Work

Tune the trade-off of vibrancy vs. realism in color re-balancing weights.

Explicitly control the colors used in a picture

Loss function - ambiguity invariant

Visualize network’s intermediate layers

Applications to other underconstrained problems? E.g. Depth of field etc.

References

[1] Zhang, Richard, Phillip Isola, and Alexei A. Efros. "Colorful image colorization." European Conference on Computer Vision. Springer International Publishing, 2016.

[2] Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. European Conference on Computer Vision (2016)

[3] Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification. ACM transactions on Graphics (Proc. of SIGGRAPH 2016) 35(4) (2016)