Colorful Image Colorization

Method

CNN architecture

Training details

- LAB space
- Quantize ab output space: bins with grid size = 10, Q = 313 (Number of quantized ab pairs)
- For a given input X, predict color distribution $\hat{Z}$:

$$\hat{Z} = G(X) \text{ where } \hat{Z} \in [0,1]^{H \times W \times Q}$$

- From $\hat{Z}$ (a distribution) to $\hat{Y}$ (a point in ab space):
  - **Mode**: vibrant but strange details
  - **Mean**: desaturated color, similar to Euclidean loss
  - **Annealed mean** (scaled softmax): $H(Z_{h,w}) = E[f_T(Z_{h,w})]$, $f_T(z) = \frac{e^{log(z)/T}}{\sum_q e^{log(z_q)/T}}$, $T = .38$.
- Ground truth color Y is converted to distribution Z using **soft encoding**:

  Find **5** nearest neighbours to Y in output space, weight them $\propto$ distance from Y using Gaussian kernel with $\sigma = 5$
- **Multinomial cross entropy loss**:

$$L_{cl}(\hat{Z}, Z) = -\sum_{h,w} v(Z_{h,w}) \sum_q Z_{h,w,q} log(\hat{Z}_{h,w,q})$$

  - $v(Z_{h,w})$ class rebalancing:
    * low ab values dominate natural images (grayish, due to clouds, pavement, dirt, walls, etc.)
    * Increase importance of rare colors:
      1. Estimate empirical probability distribution of colors in quantized ab space $p \in \Delta Q$.
      2. Smooth p to $\tilde{p}$ with Gaussian kernel G_σ, $\sigma = 5$.
      3. Mix $\tilde{p}$ with a uniform distribution $\frac{1}{Q}$ (tones down importance of rare colors slightly), then take reciprocal (rare colors importance > frequent colors): $w \propto ((1-\lambda)\tilde{p} + \lambda \times \frac{1}{Q})^{-1}$, $\lambda = .5$.
      4. Normalize w so that $E[w] = \sum_q \tilde{p}_q w_q = 1$

Experiments

- Data:
  - Training: 1.3M ImageNet training set
  - Validation: 10k ImageNet validation set
  - Test: 10k ImageNet validation set
- Details:
  - Initialization: k-means (checkout *data dependent initialization* paper!)
  - ADAM solver
  - 450k iterations
  - $\beta_1 = .9, \beta_2 = .99$, weight decay $= 10^{-3}$.
  - Initial learning rate $= 3 \times 10^{-5}$, dropped to 10^{-5} (~200k iteration) and then 3×10^{-6} (~375k iterations) when loss plateaued
- More tests:
  - *task generalization*: freeze network parameters and training an object classifier on seen data from features from each conv layer
  - *dataset generalization*: object classification on unseen data
  - colorize legacy black and white images

Evaluations

1. Perceptual realism: find people to evaluate
2. Semantic interpretability: feed colorized results to state-of-the-art ImageNet classifiers (compare different ones?)
3. AuC (Raw accuracy):
   - without class rebalance: count number of pixels within a threshold L2 distance from the ground truth in *ab* space, $threshold \in [0, 150]$, integrate the area under the curve and normalize.
   - class rebalance: use w function defined above with $\lambda = 0$.