ESTIMATING COLORFULNESS OF AN IMAGE

Eric Chu :: Erin Hsu :: Sandy Yu

Psych221

March 19, 2007

Introduction

- What is the problem?
  - Lack of quantitative metric for colorfulness of image
  - Hard to simulate whether image will appear colorful
- Past work
  - Actual Colorfulness of image (not perceptual)
  - Spatial CIELAB
  - Colorfulness Metric (Hasler, Süsstrunk)

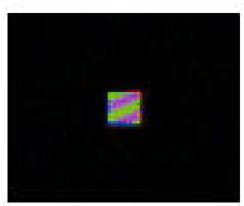
$$\begin{aligned}\hat{M}^{(3)} &= \sigma_{rgyb} + 0.3 \cdot \mu_{rgyb}, \\ \sigma_{rgyb} &:= \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2}, \\ \mu_{rgyb} &:= \sqrt{\mu_{rg}^2 + \mu_{yb}^2},\end{aligned}$$

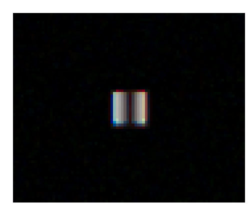
Methods - Overall

- Collect behavioral data using ISET
- Use CIELAB, sCIELAB, Hasler metric to calculate errors
  - Varies with distance, frequency, angle
- Find function that best separates colorfulness from non-colorfulness
- Apply different classifier algorithms to predict colorfulness
- Compare with frequency-orientation image

Collecting Behavioral Data

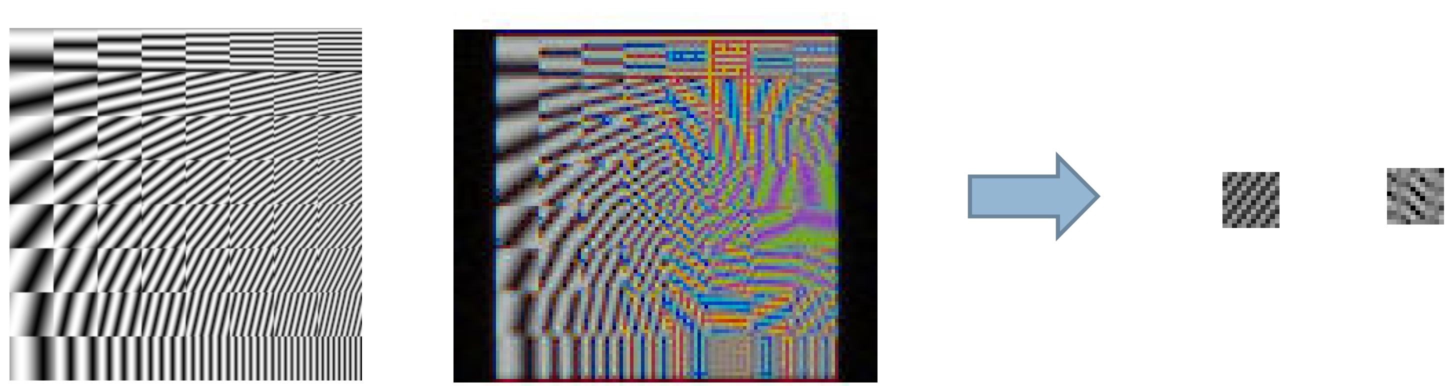
- 100 images of different harmonics
  - Varied orientation from 0-90°
  - Frequency varied from 1-10
- Tested distances of 6, 12, 24, 36 inches from screen
- Total: “colorfulness” determined for 20 sets of 100 images
  - Binary: Color = 1, No color = 0





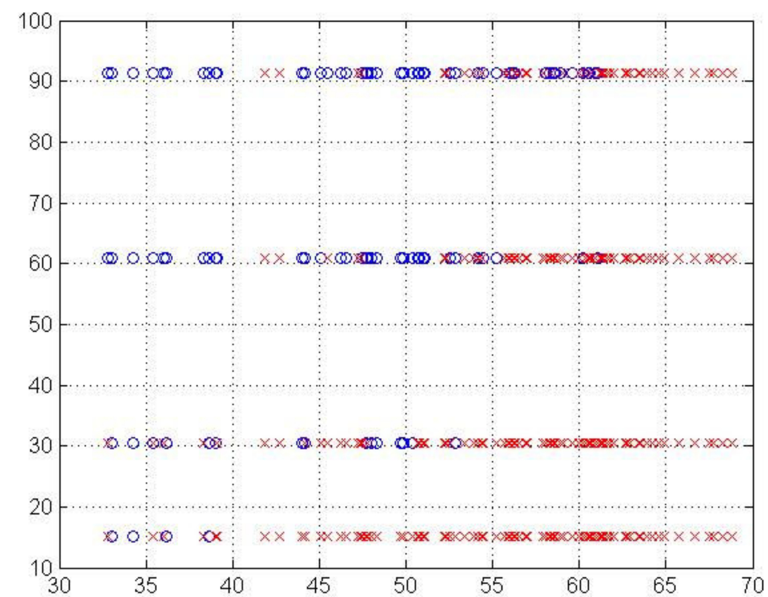
Creating Reference Images

- For individual ‘harmonic’ scenes, generate reference image with specific parameters
- For ‘frequencyOrientation’ scene, interpolate reference image to allow one-to-one comparison
  - Divide into 64 12x12 blocks



Separating Color Data

- Optimizing CIELAB and sCIELAB values
  - Ideally, “aggregate” delta Eab values should be large for “colored” images
  - Used Gaussian blur alongside $\mu + 3*\sigma$



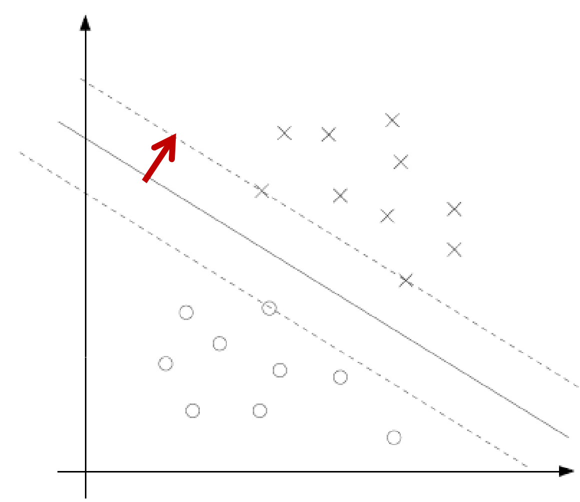
- Hasler Colorfulness Metric
  - Measures “natural” colorfulness without a reference image
  - Uses sRGB values

Classifiers

- Logistic Regression
  - Binary Classifier
  - Maximizes the likelihood of P(colored | parameters) by finding a weighted sum of the parameters
    - Weighted sum gives a separating hyper-plane
  - Quick to compute (iterative)
  - Easy to use for classification (store weights)
  - Our data appears linearly separable (accurate)

Classifiers

- SVM – Support Vector Machine
  - Binary Classifier
  - Maximizes margin between colored (or not colored) samples and the separating hyper-plane



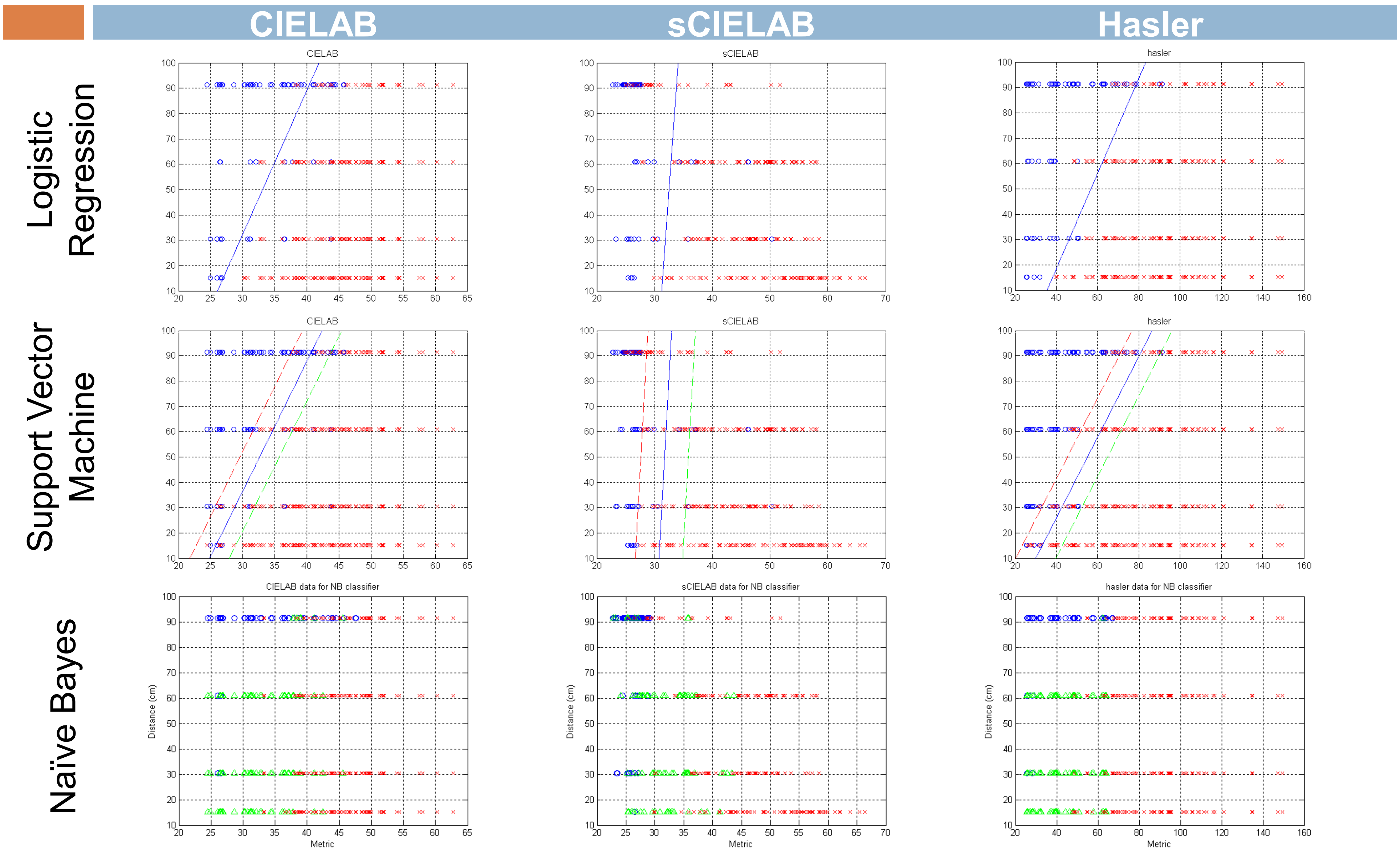
  - Pre-packaged program (SVM_light from Cornell)
  - Easy to use for classification (store support vector)
  - Can use nonlinear separating "hyper-planes"

Classifiers

- Naïve Bayes
  - Simple, Multivariable probabilistic classifier
  - Applies Bayes' theorem with strong independence assumptions
  - Conditional model over dependent class variable C with a small number of outcomes, conditional on feature variables F_1 through F_n
  - Supervised learning session for training

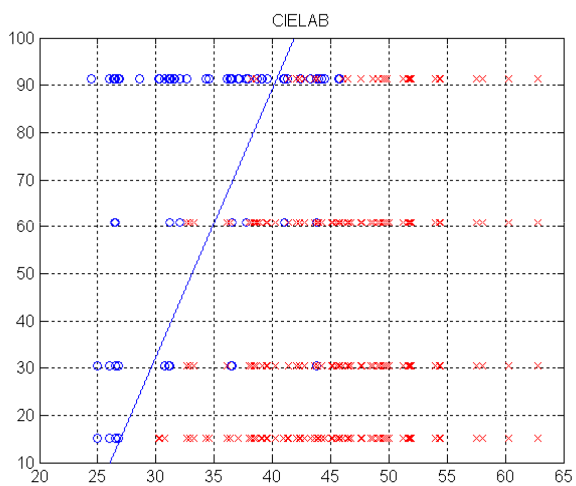
$$\text{classify}(f_1, \ldots, f_n) = \text{argmax}_c\ P(C = c) \prod_{i=1}^{n} p(F_i = f_i | C = c)$$

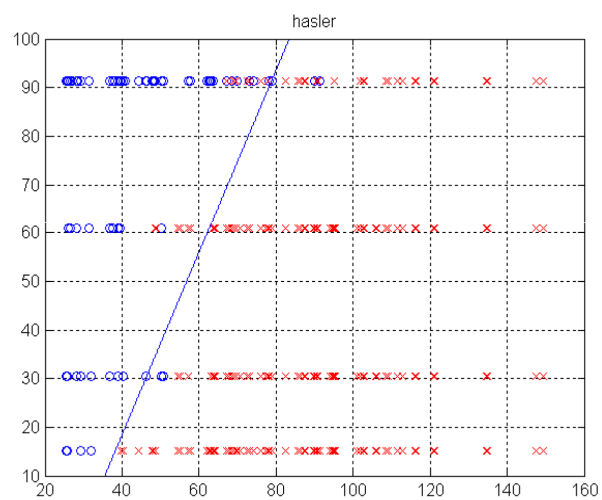
Results – Classifier vs. Metric

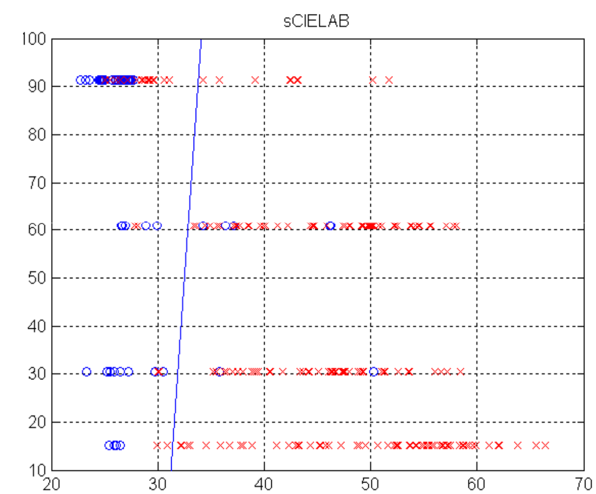


Classifier Results

- Logistic Regression







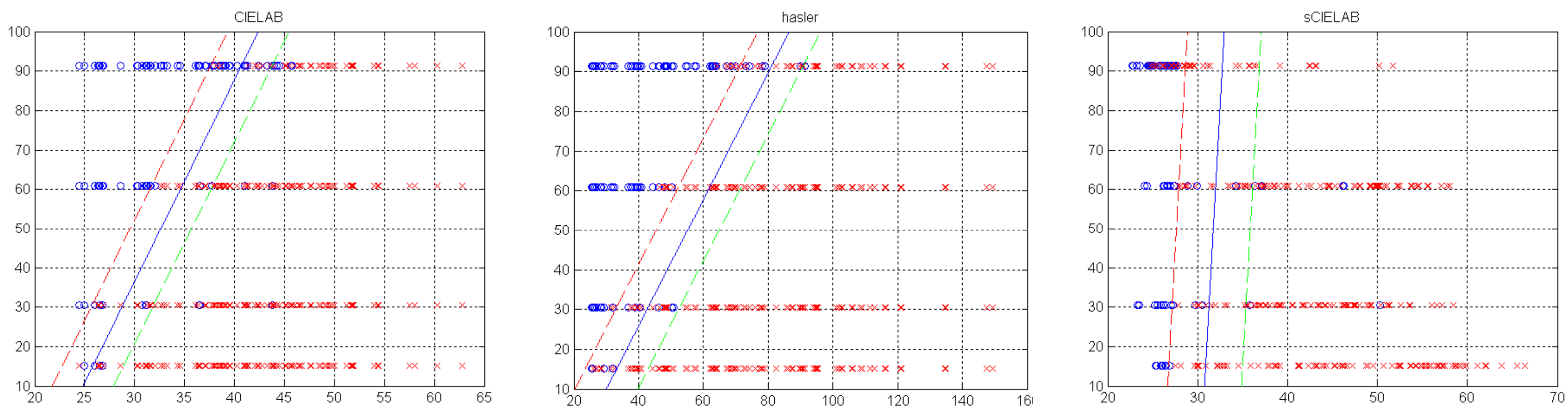
Best Separating Hyper-plane (using sCIELAB metric):

$$\text{Set } dist^2 + freq^2 + orien^2 + error^2 = 1, \text{ then :}$$

$$-3.7197 * dist + 19.7184 * freq + 0.1108 * orien + 2.399 * error + 0.0944 = 0$$

Classifier Results

- Support Vector Machine



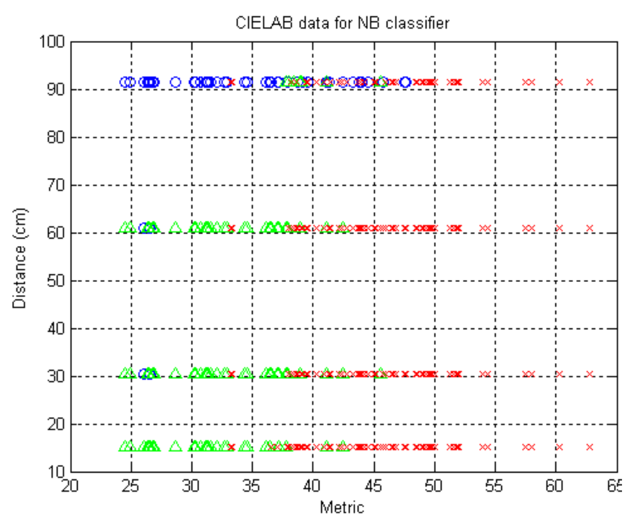
Best Separating Hyper-plane (using sCIELAB metric):

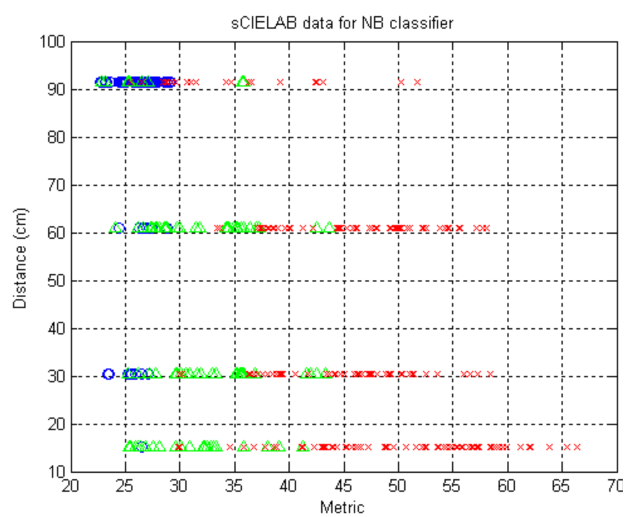
Set $dist^2 + freq^2 + orien^2 + error^2 = 1$, then :

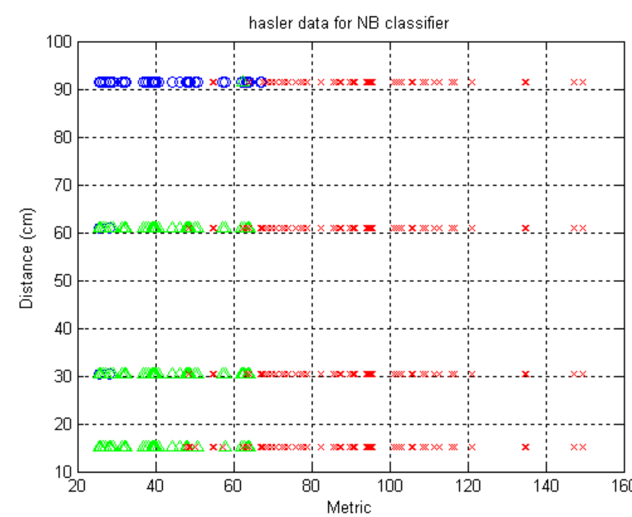
$$-2.7202 * dist + 2.1072 * freq + 0.0475 * orien + 4.01 * error - 0.2472 = 0$$

Classifier Results

- Naïve Bayes
  - 3 Bins:
    - Uncolored ○
    - Unknown △
    - Colored ✕







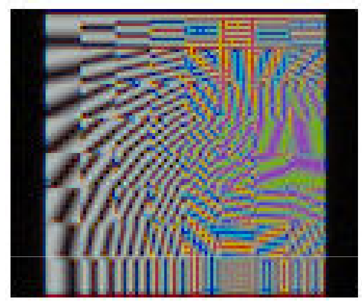
Metric Results

- Comparisons using Training Data
  - CIELAB
    - LR: 0.1025
    - SVM: 0.1025
    - NB: 0.12
    - Avg: 0.1083
  - sCIELAB
    - LR: 0.1125
    - SVM: 0.1525
    - NB: 0.13
    - Avg: 0.1317
  - Hasler
    - LR: 0.065
    - SVM: 0.08
    - NB: 0.12
    - Avg: 0.0883
- CIELAB
  - Sensitive to spatial frequency
- Spatial CIELAB
  - Considers spatial frequency, most ideal
- Hasler metric
  - Not necessarily perceptually uniform

Frequency-Orientation Test Results

- Original "Truth" Tables

```
[ 0  0  0  1  1  1  1  1;
  0  0  0  1  1  1  1  1;
  0  0  0  0  1  1  1  1;
  0  0  0  0  1  1  1  1;
  0  0  0  1  1  1  1  1;
  0  0  0  1  1  1  1  1;
  0  0  0  1  1  1  1  1;
  0  0  0  1  1  1  1  1 ]
```



```
[ 0  0  0  2  2  2  2  2;
  0  0  1  2  2  2  2  2;
  0  0  1  1  2  2  1  2;
  0  0  1  1  2  2  2  2;
  0  0  1  1  2  2  2  2;
  0  0  1  1  2  2  2  2;
  0  0  1  2  2  2  2  2;
  0  0  1  2  2  2  2  1 ]
```

- Error Values
  - CIELAB
    - LR: 0.0313
    - SVM: 0.0625
    - NB: .1797
  - sCIELAB
    - LR: 0.0313
    - SVM: 0.0938
    - NB: 0.1250
  - Hasler
    - LR: 0.0625
    - SVM: 0.0938
    - NB: 0.1719

Overall Results

- Logistic Regression Classifier appears to be the most accurate
  - **LR: 0.0417 avg error**
  - SVM: 0.0834 avg error
  - NB: 0.1589 avg error
- Spatial CIELAB Gaussian-filtered metric appears to be the most accurate
  - **sCIELAB: 0.0834 avg error**
  - CIELAB: 0.0912 avg error
  - Hasler: 0.1094 avg error

Sources of Error

- Skewed behavioral data
  - Small sample size
  - Effect of different displays
  - Aliasing in spatial frequency
- Imprecise reference images
- Assumption of linear relationships between errors and distance, frequency, orientation

Future Steps

- Apply to any black and white scene
  - Take 2DFT of image
  - Use metric on the harmonics
- Areas of improvement
  - Improve sCIELAB
  - Improve classifier
    - More dimensions – FOV of scene, brightness, etc
    - Nonlinear classifier

Conclusion

- B&W harmonics can be quantitatively classified
  - 90% correlation with perceived color
  - Fast computation
- Theoretically, spatial CIELAB is best metric for measuring color perception
- Experimentally, CIELAB, sCIELAB, Hasler comparable
  - Depends on application