In this project, we implement an auto-enhancement framework that can learn user preferences to enhance images in a personalized way. Our method finds a maximally representative training subset (20 images) out of a large dataset, allowing for efficient training. The parameters chosen in the training phase can then be applied accordingly to other images in the dataset, automatically creating an entire library of personally customized images.

**Dataset**

- Dataset consists of 500 images
- Selected photos to represent a typical user photo library (landscapes, faces, urban life, etc).

**Enhancement Parameters**

- Four enhancement parameters to learn:
  - \( \lambda \) and \( a \) in 5-curve formula:
    \[
    x = \begin{cases} 
    a - a \left( 1 - \frac{y}{y_{\text{max}}} \right)^3, & \text{if } y \leq y_{\text{max}} \\
    a + \left( 1 - a \right) \left( 1 - \frac{y}{y_{\text{max}}} \right)^3, & \text{otherwise}
    \end{cases}
    \]
  - \( \lambda \) and \( a \) relate to amount of contrast in an image, and \( x \) is input pixel and \( y \) is output pixel.
  - Color Temperature (\( T \)) and tint (\( h \)), where changes in \( T \) and \( h \) can result in color correction.

**User Study Results**

- **Personalized vs. Original**
  - Preferred Personalization: 54.3%
  - Preferred Original: 27.1%
  - No Preference: 18.6%

- **Personalized vs. Google Photos**
  - Preferred Personalization: 28.6%
  - Preferred Google: 50.0%
  - No Preference: 21.4%

- **Personalized vs. Photoshop**
  - Preferred Personalization: 30.0%
  - Preferred Photoshop: 40.0%
  - No Preference: 30.0%

**Optimal Distance Metric** - Linear combination of 25 image distances. The weights of the linear combination are such that the difference between the distances in the image space and the parameter space is minimized.

**Training Set** - Using the distance metric found, we find the distance between each pair of images in the dataset. A sensor placement optimization scheme [2] is then used to rank the images and choose 20 images that are maximally informative of the dataset.

**Optimal Parameter Selection** - 3 values for 4 parameters = 81 combinations. Using the same optimization method, we find the 8 parameter combinations that maximally represent the parameter space.

**Training** - For each training image, a user selects the most desired parameter combination.

After training, an image is enhanced using the learned parameters via the following pipeline:

- **Input image**
- **\( y \) (linearize)**
- **Auto-enhance**
- **Find the closest training image**
- **Personalized enhance**
- **\( 1/y \) (delinearize)**

**Image Processing Pipeline**

**References**

