Histograms Continued

CS/BIOEN 4640: Image Processing Basics

January 24, 2012
Given a particular image $I(u, v)$,

- What is $h(i)$?
  
  
  $h(i) = \text{the # of pixels in } I \text{ with intensity } i$

- What is $P(i)$?
  
  
  $P(i) = h(i) / wh$

- What is $E(I)$?
  
  
  $E(I) = \sum_{i=0}^{K-1} i \cdot P(i)$
What is $P(I \leq t)$?

$$P(I \leq t) = \frac{1}{wh} \sum_{i=0}^{t} h(i)$$
Cummulative Histogram

**Definition**

For a given image $I(u, v)$, the **cummulative histogram** is a function $H(i)$ that counts the number of pixels in $I$ that have intensity less than or equal to $i$.

$$H(i) = \sum_{j=0}^{i} h(j)$$

$$P(I \leq t) = H(t)/wh$$
Automatic Contrast Adjustment

- Find lowest and highest pixel intensities, $a_{\text{low}}$ and $a_{\text{high}}$.
- Use intensity windowing point operation to stretch $[a_{\text{low}}, a_{\text{high}}]$ to the full range, $[0, K - 1]$:

$$ f(p) = \begin{cases} 
0 & \text{if } p < a_{\text{low}} \\
(K - 1) \times \frac{p - a_{\text{low}}}{a_{\text{high}} - a_{\text{low}}} & \text{if } a_{\text{low}} \leq p \leq a_{\text{high}} \\
(K - 1) & \text{if } p > a_{\text{high}}
\end{cases} $$
Auto-Contrast With Saturation

- As we have seen, low and high intensities in an image may not be stable.
- So, we let the endpoints of the range, 0 and $K - 1$ become “saturated”.
- That is, we shift our interval inwards so that a small percentage of pixels are above and below.
- Choose a saturation level $\epsilon$. Compute an interval $[\hat{a}_{\text{low}}, \hat{a}_{\text{high}}]$ such that $P(I \leq \hat{a}_{\text{low}}) = \epsilon$ and $P(I \geq \hat{a}_{\text{high}}) = \epsilon$. Then intensity window using this interval.
Auto-Contrast With Saturation
The goal of histogram equalization is to apply a point operation that makes the histogram uniform (flat).
Histogram Equalization

- Histogram cannot be made exactly flat - peaks cannot be increased or decreased by point operations.
- The following point operation makes the histogram as close to flat as possible:

\[
feq(a) = \left\lfloor H(a) \cdot \frac{K - 1}{wh} \right\rfloor
\]
Histogram Equalization

\[ f_{eq}(a) = \left[ H(a) \cdot \frac{K - 1}{wh} \right] \]
Histogram Matching

- Given two images $I_A$ and $I_B$, we want to make their intensity profiles look as similar as possible.
- Do this by “matching” the cumulative histograms $H_A$ and $H_B$.
- Works well for images with similar content.
- Looks really horrible for images with different content.
Histogram Matching

$H_A(i)$

$H_B(i)$
Histogram Matching

\[ f(0) = 0 \]
Histogram Matching

\[ f(0) = 0 \]
\[ f(1) = 1 \]
Histogram Matching

\[
\begin{align*}
H_A(i) & = 0 \\
H_B(i) & = 0 \\
H_A(i) & = 1 \\
H_B(i) & = 1 \\
H_A(i) & = 3 \\
H_B(i) & = 3 \\
H_A(i) & = 3 \\
H_B(i) & = 3 \\
H_A(i) & = 6 \\
H_B(i) & = 6 \\
H_A(i) & = 6 \\
H_B(i) & = 6 \\
H_A(i) & = 7 \\
H_B(i) & = 7 \\
\end{align*}
\]

\[
\begin{align*}
f(0) & = 0 \\
f(1) & = 1 \\
f(2) & = 3 \\
\end{align*}
\]
Histogram Matching

$f(0) = 0$
$f(1) = 1$
$f(2) = 3$
$f(3) = 3$
Histogram Matching

\[ H_A(i) \quad H_B(i) \]

\[ f(0) = 0 \]
\[ f(1) = 1 \]
\[ f(2) = 3 \]
\[ f(3) = 3 \]
\[ f(4) = 3 \]
Histogram Matching

\[ f(0) = 0 \]
\[ f(1) = 1 \]
\[ f(2) = 3 \]
\[ f(3) = 3 \]
\[ f(4) = 3 \]
\[ f(5) = 3 \]
Histogram Matching

\[ f(0) = 0 \]
\[ f(1) = 1 \]
\[ f(2) = 3 \]
\[ f(3) = 3 \]
\[ f(4) = 3 \]
\[ f(5) = 3 \]
\[ f(6) = 6 \]
\[ f(7) = 6 \]
Continuous Point Operations

- Consider an image with continuous (floating point) pixel type
- For simplicity, let’s assume the range is in the interval $[0, 1]$
  \[ I : \Omega \rightarrow [0, 1] \]
- Now a point operation is a function
  \[ f : [0, 1] \rightarrow [0, 1] \]
“Curves” Operation

- Specify a continuous point operation function

\[ f : [0, 1] \to [0, 1] \]

- slope = 1, no contrast change
- slope < 1, contrast is decreased
- slope > 1, contrast is increased
Gamma Correction

- When forming an image, camera sensors convert light into an electrical signal.
- Different camera sensors will have different responses to light intensity and produce different electrical signals.
- Likewise display devices (monitors, projectors, printers) have to turn images into a physical representation (light, ink, etc)
- How do we make sure there is consistency in the images recorded by different cameras and in the images produced by different display devices?
Gamma Correction

Gamma correction is the continuous point operation

\[ f(a) = a^\gamma, \quad \gamma > 0 \]
Gamma Correction

Hardware manufacturers specify the gamma value that will correctly record or reproduce colors.