ECE 63700 Laboratory:

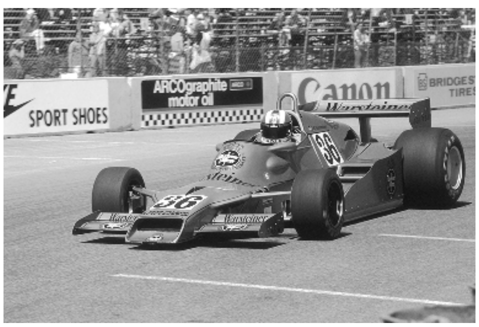
Pointwise Operations and Gamma

Shang Hsuan (Sean) Lee

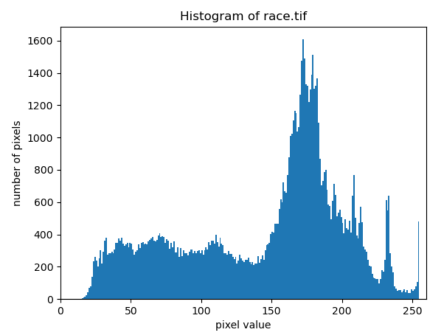
February 4, 2024

1 Histogram of an Image

1.1 Histogram of *race.tif* and *kids.tif*

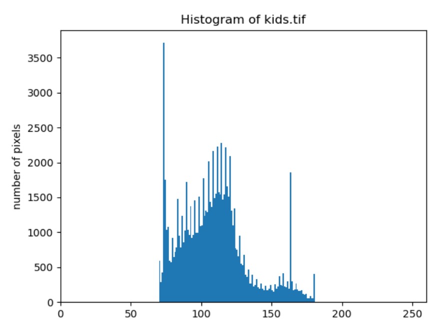


(a) race.tif



(b) Histogram of race.tif

(c) kids.tif



(d) Histogram of kids.tif

2 Histogram Equalization

2.1 Code Listing

Function *equalizer(X)*

```
def equalizer(X):
    Z = np.zeros(np.shape(X))
    F_hat = np.zeros(256)
```

```
h = np.histogram(X.flatten(), bins=np.linspace(0, 255, 256))

for i in range(0, 256):
    F_hat[i] = np.sum(h[0][0:i+1]/np.sum(h[0]))

Ymax = np.max(F_hat)
Ymin = np.min(F_hat)

for i in range(np.shape(X)[0]):
    for j in range(np.shape(X)[1]):
        Z[i, j] = 255 * ((F_hat[X[i, j]] - Ymin) / (Ymax - Ymin))

return Z, F_hat
```

2.2 Plot of $\hat{F}_x(i)$ for the image *kids.tif*

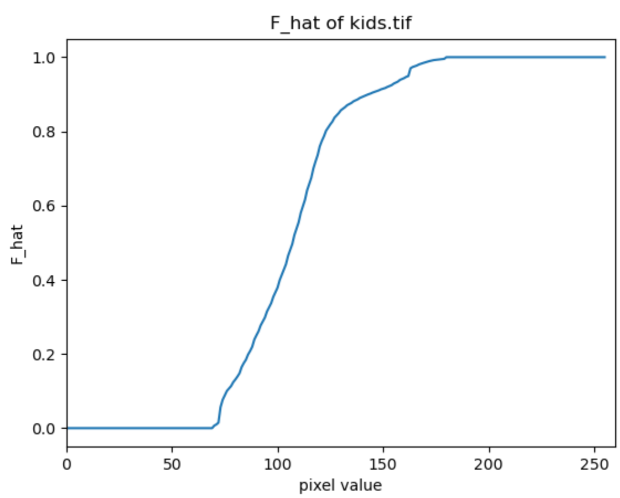


Figure 2: $\hat{F}_x(i)$ of *kids.tif*

2.3 Histogram of equalized *kids.tif*

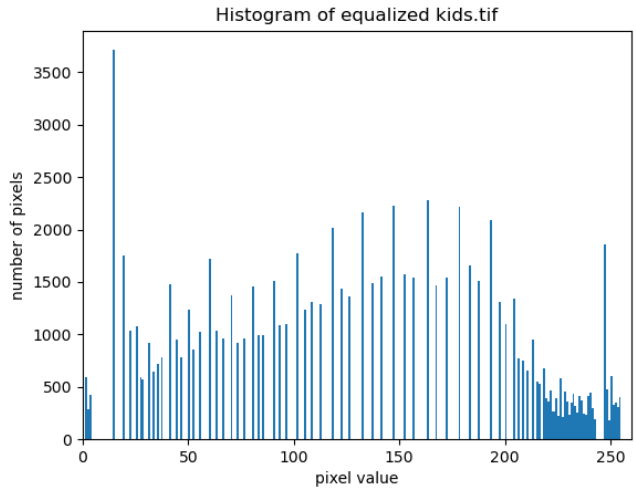


Figure 3: Histogram of equalized *kids.tif*

2.4 Equalized Image

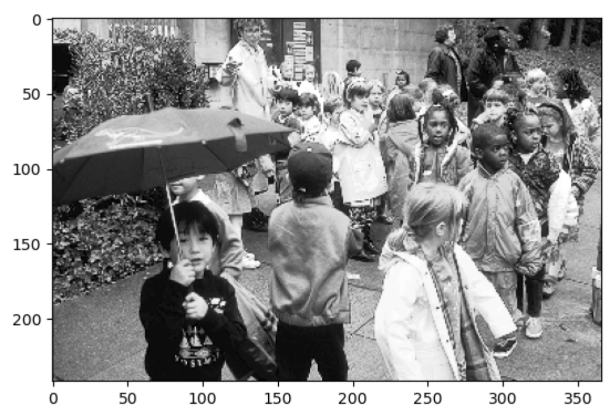


Figure 4: Equalized *kids.tif*

3 Contrast Stretching

3.1 Code Listing

Function *stretch(X, T1, T2)*

```
def stretch(X, T1, T2):
    output = np.zeros(np.shape(X))
```

```
for i in range(np.shape(X)[0]):
    for j in range(np.shape(X)[1]):
        if X[i, j] <= T1:
            output[i, j] = 0
        elif X[i, j] >= T2:
            output[i, j] = 255
        else:
            output[i, j] = (255 / (T2-T1)) * (X[i, j] - T1)

return output
```

3.2 Histogram of stretched *kids.tif*

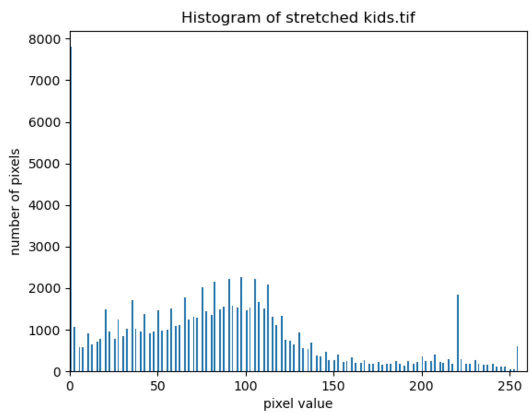


Figure 5: Histogram of stretched *kids.tif*

3.3 Stretched Image

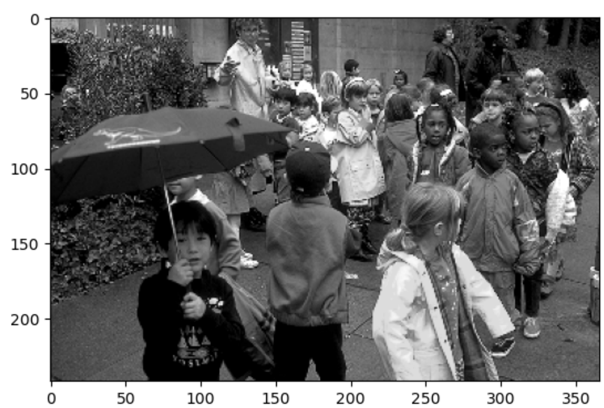


Figure 6: Stretched *kids.tif*

4 Gamma (γ)

4.1 Setting the Black Level and Picture of Your Monitor

No reports needed.

4.2 Determining the Gamma of Your Computer Monitor

4.2.1 Checkerboard pattern

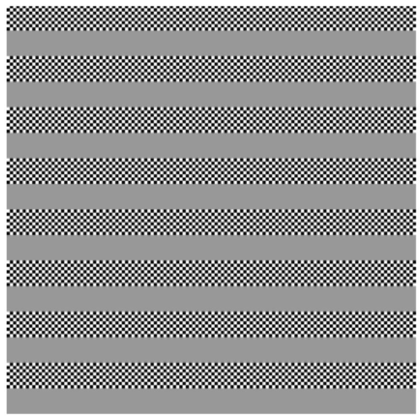


Figure 7: Checkerboard pattern with $g = 152$

4.2.2 Derivation of γ

Given:

$$I_g = I_c = \frac{I_{255} + 0}{2}$$

$$I_g = I_{255}(\frac{g}{255})^\gamma$$

We can derive γ:

$$\begin{aligned}
\frac{I_{255}}{2} &= I_{255}(\frac{g}{255})^\gamma \\
\frac{1}{2} &= (\frac{g}{255})^\gamma \\
log(\frac{1}{2}) &= log((\frac{g}{255})^\gamma) = \gamma log(\frac{g}{255}) \\
\gamma &= \frac{log(\frac{1}{2})}{log(\frac{g}{255})} \\
&= \frac{log(1) - log(2)}{log(g) - log(255)} \\
&= \frac{-log(2)}{log(g) - log(255)}
\end{aligned}$$

4.2.3 Values of the measured gray level and the measured γ

With the measured gray level (g) 152, we can calculate γ:

$$\gamma = \frac{-log(2)}{log(152) - log(255)} = 1.3398$$

4.3 Gamma Correction

4.3.1 Original and corrected *linear.tif* ($\gamma = 1.3398$)

(a) Original image *linear.tif*

(b) Corrected image of *linear.tif*

4.3.2 Formula used for transformation

Knowing that:

$$y = 255 * (\frac{x}{255})^{\gamma}$$
$$\gamma = 1.3398$$

We can derive the formula for transformation:

$$y = 255 * (\frac{x}{255})^{\gamma}$$
$$\frac{y}{255} = (\frac{x}{255})^{\gamma}$$
$$(\frac{y}{255})^{\frac{1}{\gamma}} = \frac{x}{255}$$
$$x = 255 * (\frac{y}{255})^{\frac{1}{\gamma}}$$

4.3.3 Reproduce *gamma15.tif*

Figure 9: Reproduced *gamma15.tif*

4.3.4 Procedure used for gamma correction

Knowing that the given image was encoded with $\gamma_1 = 1.5$:

$$y = 255 * (\frac{x}{255})^{\gamma_1}$$

To reproduce the image with $\gamma_2 = 1.3398$:

$$\begin{aligned} z &= 255 * (\frac{y}{255})^{\frac{1}{\gamma_2}} \\ &= 255 * (\frac{255 * (\frac{x}{255})^{\gamma_1}}{255})^{\frac{1}{\gamma_2}} \\ &= 255 * (\frac{x}{255})^{\frac{\gamma_1}{\gamma_2}} \end{aligned}$$