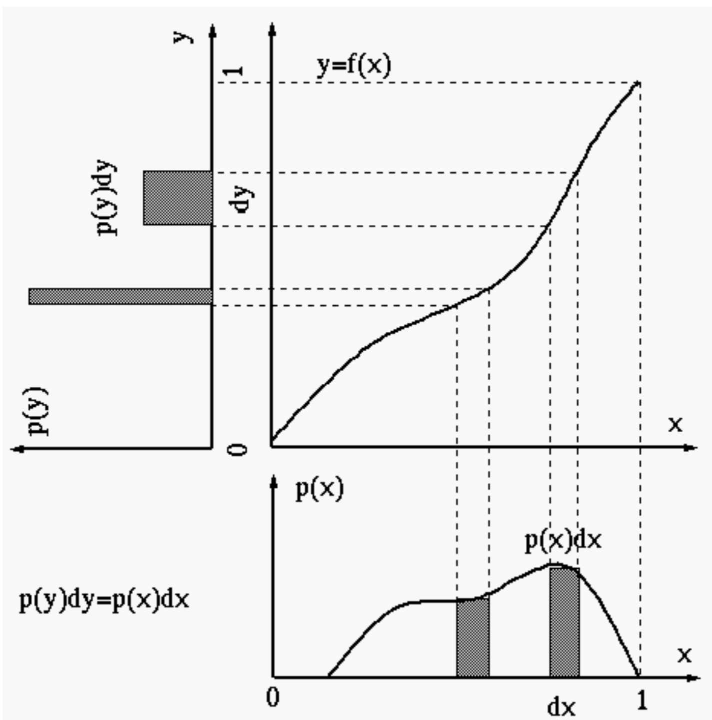
# Histogram Equalization

To transfer the gray levels so that the histogram of the resulting image is equalized to be a constant:

$$h[i] = \text{constant}, \quad \text{for all } i$$

The purposes:

- To equally use all available gray levels;
- For further histogram specification.




This figure shows that for any given mapping function y = f(x) between the input and output images, the following holds:

$$p(y)\,dy = p(x)\,dx$$

i.e., the number of pixels mapped from x to y is unchanged. To equalize the histogram of the output image, we let p(y) be a constant.

In particular, if the gray levels are assumed to be in the ranges between 0 and 1 ($0 \leq x \leq 1, 0 \leq y \leq 1$), then p(y) = 1. Then we have:

$$dy = p(x)dx, \quad \text{or} \quad \frac{dy}{dx} = p(x)$$

i.e., the mapping function y = f(x) for histogram equalization is:

$$y = f(x) = \int_0^x p(u)du = P(x) - P(0) = P(x)$$

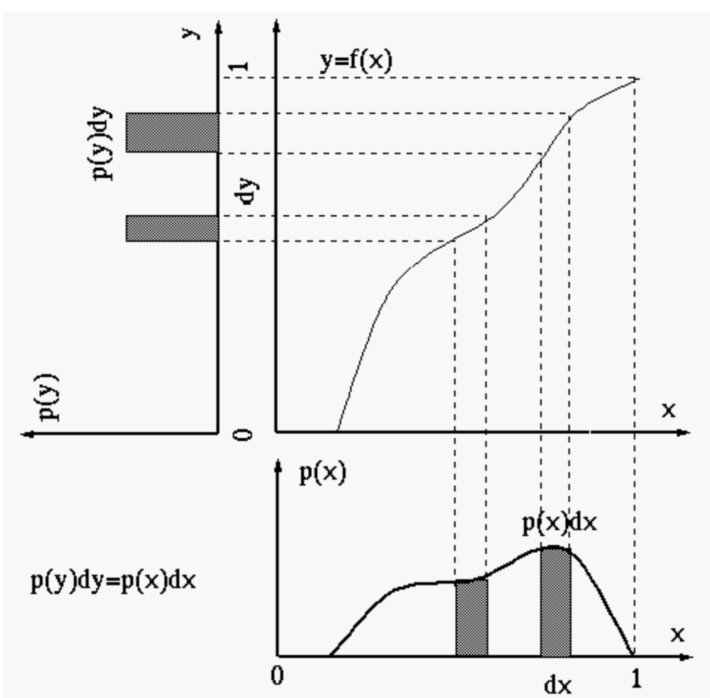
where

$$P(x) = \int_0^x p(u)du, \qquad P(0) = 0$$

is the cumulative probability distribution of the input image, which monotonically increases.

Intuitively, histogram equalization is realized by the following:

- If p(x) is high, P(x) has a steep slope, dy will be wide, causing p(y) to be low to keep p(y) dy = p(x) dx;
- If p(x) is low, P(x) has a shallow slope; dy will be narrow, causing p(y) to be high.



For discrete gray levels, the gray level of the input x takes one of the L discrete values: $x \in \{0, 1, \cdots, L-1\}$ and the continuous mapping function becomes discrete:

$$y' = f[x] \stackrel{\triangle}{=} \sum_{i=0}^{x} h[i] = H[x]$$

where h[i] is the probability for the gray level of any given pixel to be i ($0 \le i \le L-1$):

$$h[i] = \frac{n_i}{\sum_{i=0}^{L-1} n_i} = \frac{n_i}{N} \quad \text{and} \quad \sum_{i=0}^{L-1} h[i] = 1$$

Here h[i] is the histogram of the image and H[i] is the cumulative histogram.

The resulting function y' is in the range $0 \le y' \le 1$ and it needs to be converted to the gray levels $0 \le y \le L-1$ by either of the two ways:

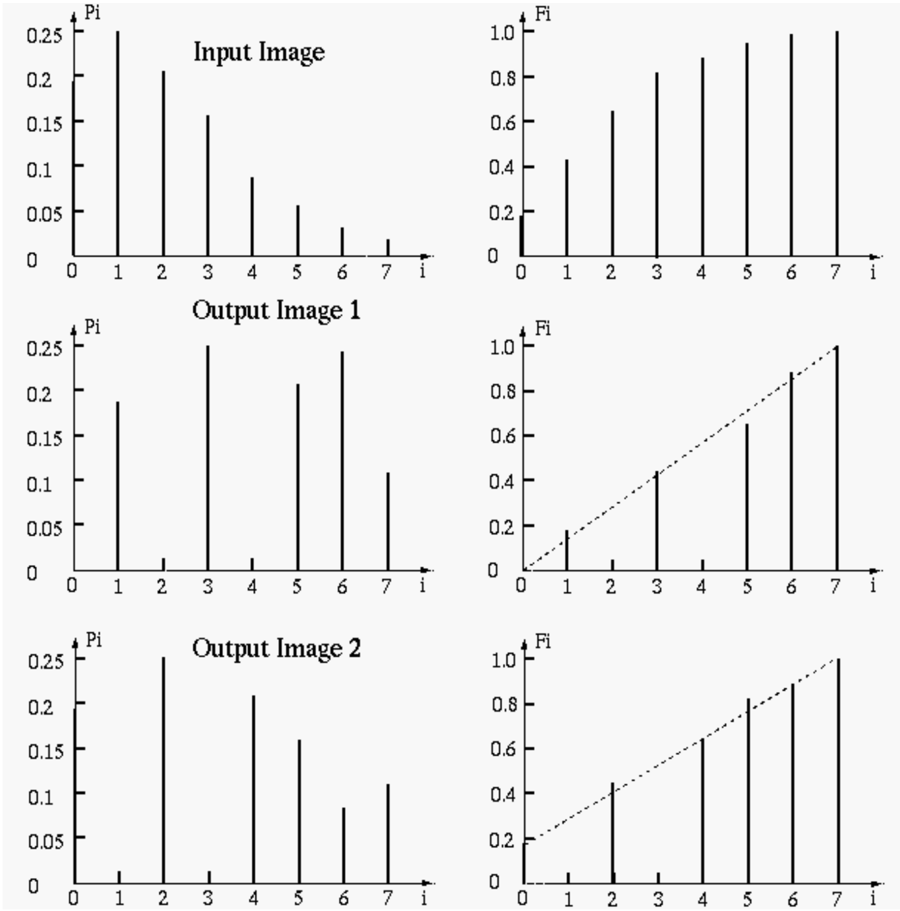
1. $$y = \lfloor y'(L-1) + 0.5 \rfloor$$

2. $$y = \lfloor \frac{y' - y'_{min}}{1 - y'_{min}}(L-1) + 0.5 \rfloor$$

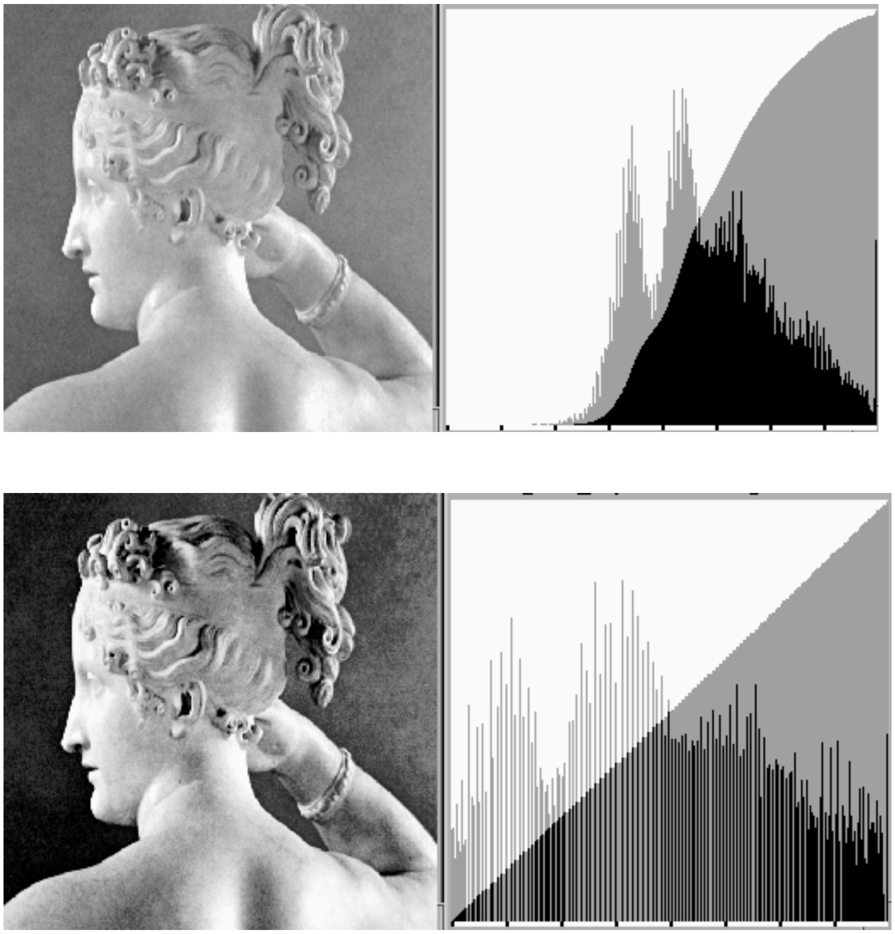
Here $\lfloor x \rfloor$ is the floor, or the integer part of a real number x, and adding 0.5 is for proper rounding. Note that while both conversions map $y'_{max} = 1$ to the highest gray level L-1, the second conversion also maps $y'_{min}$ to 0 to stretch the gray levels of the output image to occupy the entire dynamic range $0 \le y \le L-1$.

**Example:** Assume the images have 64 x 64 = 4096 pixels in 8 gray levels. The following table shows the equalization process corresponding to the two methods:

| $x_i$ | $n_i$ | $h[i] = n_i/N$ | $y' = H[i]$ | $y_j^1$ | $h^1[j]$ | $H^1[j]$ | $y_j^2$ | $h^2[j]$ | $H^2[j]$ |
|---|---|---|---|---|---|---|---|---|---|
| 0/7 | 790 | 0.19 | 0.19 | 1/7 | 0.19 | 0.19 | 0/7 | 0.19 | 0.19 |
| 1/7 | 1023 | 0.25 | 0.44 | 3/7 | 0.25 | 0.44 | 2/7 | 0.25 | 0.44 |
| 2/7 | 850 | 0.21 | 0.65 | 5/7 | 0.21 | 0.65 | 4/7 | 0.21 | 0.65 |
| 3/7 | 656 | 0.16 | 0.81 | 6/7 | | | 5/7 | 0.16 | 0.81 |
| 4/7 | 329 | 0.08 | 0.89 | 6/7 | 0.24 | 0.89 | 6/7 | 0.08 | 0.89 |
| 5/7 | 245 | 0.06 | 0.95 | 7/7 | | | 7/7 | | |
| 6/7 | 122 | 0.03 | 0.98 | 7/7 | | | 7/7 | | |
| 7/7 | 81 | 0.02 | 1.00 | 7/7 | 0.11 | 1.00 | 7/7 | 0.11 | 1.00 |




In the following example, the histogram of a given image is equalized. Although the resulting histogram may not look constant, but the cumulative histogram is a exact linear ramp indicating that the density histogram is indeed equalized. The density histogram is not guaranteed to be a constant because the pixels of the same gray level cannot be separated to satisfy a constant distribution.



**Programming Hints:**

- Find histogram of given image:

```
d=1.0/M/N;
for (i=0;  i < 256;  i++)  h[i] = 0;
for (i=0;  i < M;  i++)
   for(j = 0;  j < N;  j++)
      h[x[i][j]]+ = d;
```

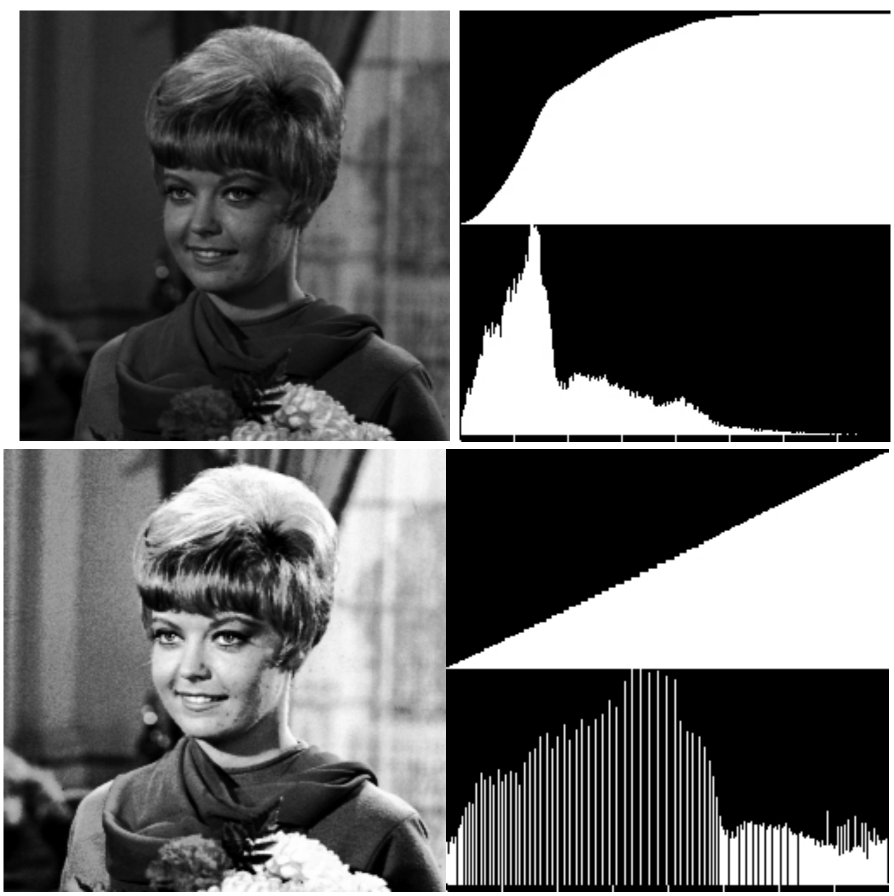
- Build lookup table:

```
sum=0.0;
for (i=0;  i < 256;  i++)  {
    sum += h[i];
    lookup[i] = sum * 255 + 0.5;
}
```

- Image Mapping:

```
for (i=0;  i < M;  i++)
  for(j = 0;  j < N;  j++)
    y[i][j] = lookup[x[i][j]];
```

**Example:**



# Histogram Specification

Here we want to convert the image so that it has a particular histogram as specified. First consider equalization transform of the given image x:

$$y = f(x) = \int_0^x p_x(u)du$$

If the desired image $z$ were available, it could also be equalized:

$$y' = g(z) = \int_0^z p_z(u)du$$

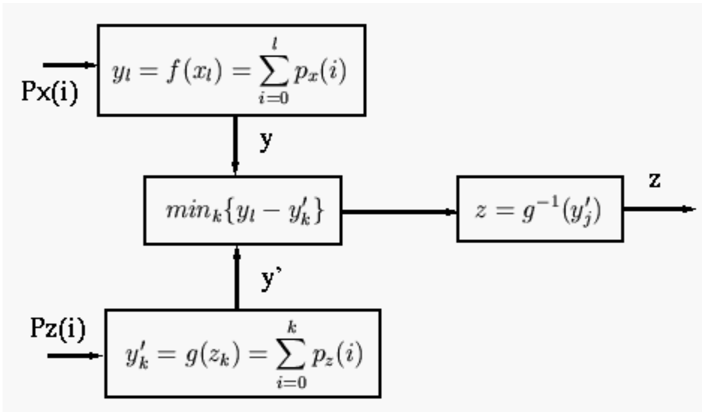
Here $p_z(u)$ is the histogram of the output image, which is specified. The inverse of the above transform is

$$z = g^{-1}(y')$$

Since images y and y' have the same equalized histogram, they are actually the same image; i.e., y = y', and the overall transform from the given image x to the desired image z can be found as:

$$z = g^{-1}(y') = g^{-1}(y) = g^{-1}(f(x))$$

where both f and g can be found from the histogram of the given image x and the desired histogram, respectively.




- **Step 1:** Find histogram of input image $p_x(i)$, and find histogram equalization mapping:

$$P_x(j) = \sum_{i=0}^{j} p_x(i)$$

- **Step 2:** Specify the desired histogram $p_z(i)$, and find histogram equalization mapping:

$$P_z(j) = \sum_{i=0}^{j} p_z(i)$$

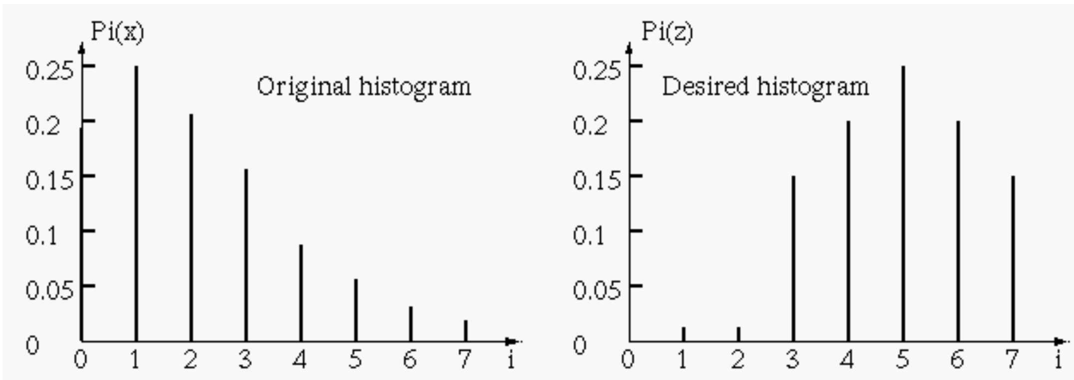
**Step 3:** Build lookup table: For each gray level $i$, find $P_x(i)$ and then find a j level so that $P_z(j)$ best matches $F_x(i)$:

$$|P_x(i) - P_z(j)| = min_k |P_x(i) - P_z(k)|$$

Setup a lookup entry lookup[i] = j.

**Example:**

The histogram of the given image and the histogram desired are shown below:




- **Step 1:** Equalize $p_x$ to get y = g(x) (save as previous example).

| $x_i$ | $y_j$ | $n_j$ | $h_y[j]$ | $H_y[j]$ |
|---|---|---|---|---|
| $x_0$ | $y_1 = 1/7$ | 790 | 0.19 | 0.19 |
| $x_1$ | $y_1 = 3/7$ | 1023 | 0.25 | 0.44 |
| $x_2$ | $y_1 = 5/7$ | 850 | 0.21 | 0.65 |
| $x_3, x_4$ | $y_1 = 6/7$ | 985 | 0.24 | 0.89 |
| $x_5, x_6, x_7$ | $y_1 = 7/7$ | 448 | 0.11 | 1.00 |

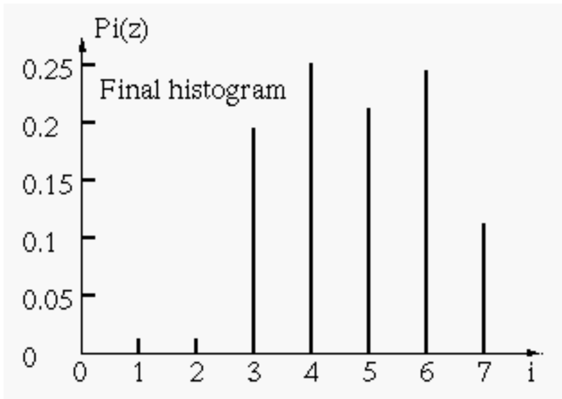
- **Step 2:** Equalize $p_z$, $y' = h(z)$.

| $z_i$ | $p_k(z)$ | $y'_j$ |
|---|---|---|
| 0/7 | 0.0 | 0.0 |
| 1/7 | 0.0 | 0.0 |
| 2/7 | 0.0 | 0.0 |
| 3/7 | 0.15 | 0.15 |
| 4/7 | 0.20 | 0.35 |
| 5/7 | 0.30 | 0.65 |
| 6/7 | 0.20 | 0.85 |
| 7/7 | 0.15 | 1.0 |

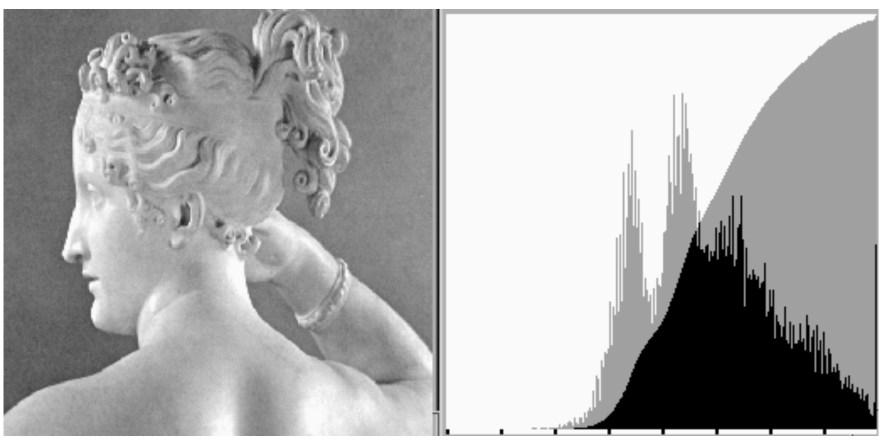
- **Step 3:** Obtain overall mapping, $x \to y \to y' \to z$

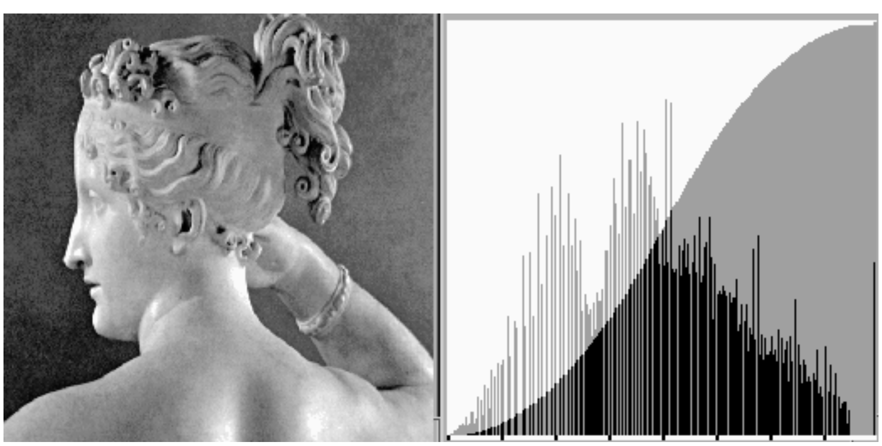
| $x_i$ | $y_j$ | $y'_j$ | $z_j$ |
|---|---|---|---|
| $x_0$ | $y_1 = 1/7 = 0.14$ | $y'_3 = 0.15$ | $z_3$ |
| $x_1$ | $y_1 = 3/7 = 0.43$ | $y'_4 = 0.35$ | $z_4$ |
| $x_2$ | $y_1 = 5/7 = 0.71$ | $y'_5 = 0.65$ | $z_5$ |
| $x_3, x_4$ | $y_1 = 6/7 = 0.86$ | $y'_6 = 0.85$ | $z_6$ |
| $x_5, x_6, x_7$ | $y_1 = 7/7 = 1.0$ | $y'_7 = 1.0$ | $z_7$ |

This is the histogram of the resulting image:




In the following example, the desired histogram is a triangle with linear increase in the lower half of the gray level range, and linear decrease in the upper half. Again the cumulative histogram shows indeed the density histogram is such a triangle.





**Programming issues:**

j=0;
for $(i = 0;\ \ i < 256;\ \ i++)\ \{$
$\quad if\ \ (Hx[i] \leq Hz[j])\ \ lookup[i] = j;$
$\quad else\ \ \{$
$\qquad while(Hx[i] > Hz[j])\ \ j++;$
$\qquad if\ \ (Hz[j] - Hx[i] > Hx[i] - Hz[j-1])\ \ \ lookup[i] = j--;$
$\qquad else\ \ \ lookup[i] = j;$
$\quad \}$
$\}$