Computer Programming

Dr. Deepak B Phatak
Dr. Supratik Chakraborty
Department of Computer Science and Engineering
IIT Bombay

Session: Histogram Equalization Program

Quick Recap

- We discussed the concept of associative arrays, and saw how it could be used to efficiently calculate a histogram

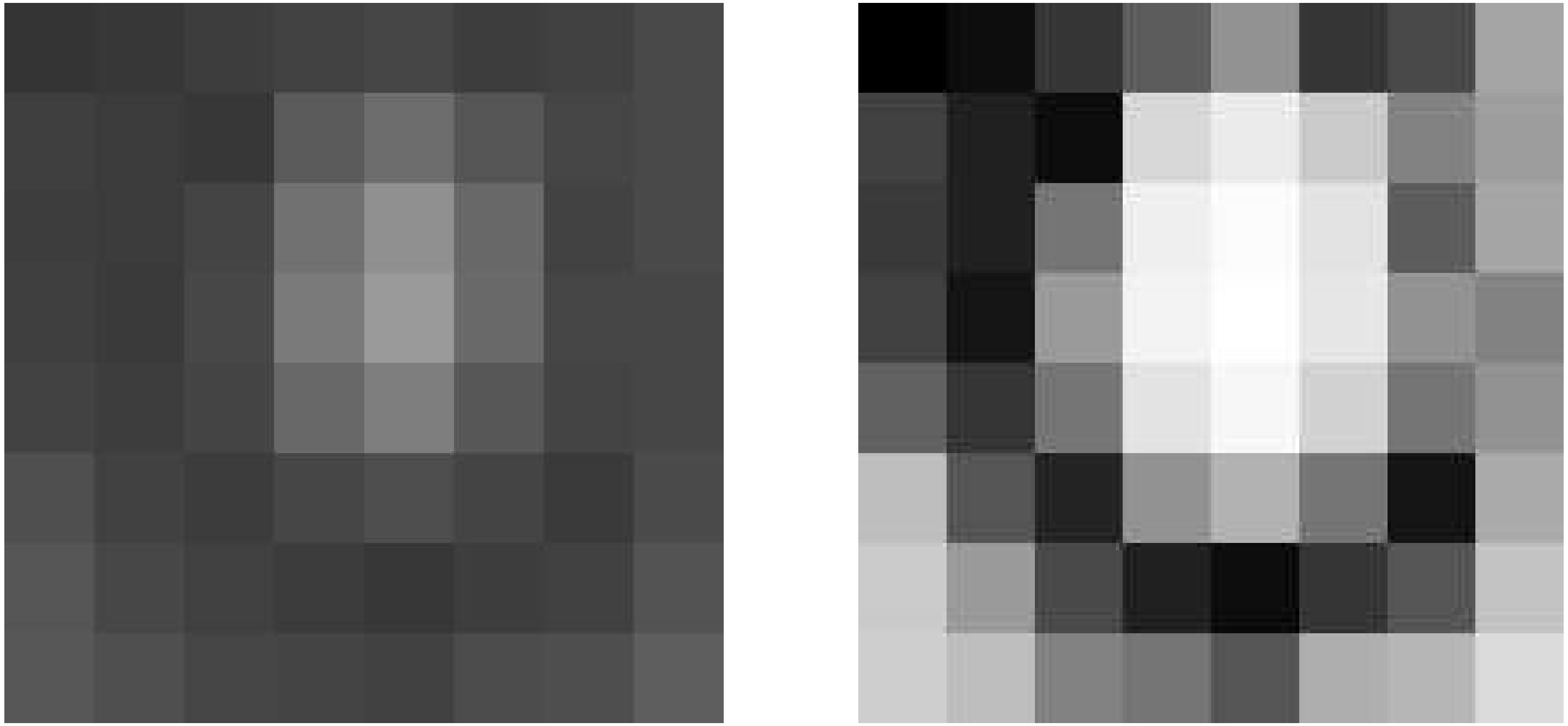
Overview

- We will use the formulae for histogram equalization, and write a program to improve image contrast

  [Note: The histogram equalization technique described here, and the digital images used are directly based on a wikipedia article:

  http://en.wikipedia.org/wiki/Histogram_equalization]

Original and contrast-enhanced pictures



Pixel values for the image

$$
\begin{bmatrix}
52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\
63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\
62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\
63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\
67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\
79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\
85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\
87 & 79 & 69 & 68 & 65 & 76 & 78 & 94
\end{bmatrix}
$$

Histogram values (shown for non-zero pixels)

Val	n	Val	n	Val	n	Val	n	Val	n
52	1	64	2	72	1	85	2	113	1
55	3	65	3	73	2	87	1	122	1
58	2	66	2	75	1	88	1	126	1
59	3	67	1	76	1	90	1	144	1
60	1	68	5	77	1	94	1	154	1
61	4	69	3	78	1	104	2		
62	1	70	4	79	2	106	1		
63	2	71	2	83	1	109	1		

Histogram Equalization

- The equalization formula to calculate new value for any existing pixel value v

$$h(v) = \mathrm{round}\left(\frac{cdf(v) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L-1) \right)$$

- “Equalization” formula for example image
  - L = 256, M = N = 8, minimum cdf is 1

$$h(v) = \mathrm{round}\left(\frac{cdf(v) - 1}{63} \times 255 \right)$$

Program: enhance_contrast.cpp

```
 /*   Program: enhance_contrast.cpp
   A program which reads the pixel intensities for a grayscale image,
   calculates the histogram and the cumulative distribution function,
   and finally recalculates the pixel values, such that the histogram is
   equalized. This procedure gives us an image with better contrast.
   Procedure and formula is based on the material given on wikipedia
*/
```

Program: enhance_contrast.cpp …

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){
  int i, j, min=0, M, N;
  int image[500][500], newimage[500][500];
  int histogram[256], cdf[256], equalizer[256];
  // Read Image data
  cout << " Give the image size: Height M and width N"<<endl;
  cin >> M >> N;
  cout<<M<<"\t"<<N<<endl;
```

enhance_contrast.cpp … (Read Original Image)

```
for(i=0;i<M;i++){
  for(j=0;j<N;j++) {
     cin>>image[i][j];
  }
}
 cout<<endl<<"Original Image:"<<endl;
 for(i=0;i<M;i++){
    for(j=0;j<N;j++){
        cout << image[i][j]<<"\t";   // output the image
    }
    cout<<endl;
 }
```

enhance_contrast.cpp … (Initialize all array elements)

```
for(i=0;i<256;i++){
  // Initialize all array elements of histogram, cdf, equalizer to 0
    histogram[i]=0;
    cdf[i]=0;
    equalizer[i]=0;
}
```

enhance_contrast.cpp … (Calculating Histogram)

```
/* calculate histogram table entries  */
for (i=0; i<M; i++) {
    for (j=0;j<N;j++){
        // principle of associative array is used
        // Value of the pixel itself is the "key" or index in the histogram table
        // indicates the element which must be incremented
         histogram[ image[i][j] ]++;
     }
 }
```

enhance_contrast.cpp … (Calculate CDF)

```
/* calculate cdf table entries */
cdf[0] = histogram[0];
for(i=1;i<256;i++){
    cdf[i]= cdf[i-1] + histogram[i];
}

/* Find the minimum nonzero value in cdf table */
min =255;
for (i=0; i < 256; i ++){
     if (cdf[i] < min && cdf[i] != 0) min =cdf[i];
}
```

enhance_contrast.cpp … (Calculate Equalizer)

IIT Bombay

```
/*  Calculate entries in the equalizer table */
for (i=0; i<256;i++){
  equalizer[i]=round((float)(cdf[i]-min)/(M*N-min)*(256-1));
}
```

enhance_contrast.cpp … (Compute New Image)

```
/*   Calculate entries in the newimage array */
for(i=0;i<M;i++){
    for(j=0;j<N;j++){
        newimage[i][j] = equalizer[ image[i][j] ];
    }
}
```

enhance_contrast.cpp … (Output New Image)

```
cout<<endl<<"New Image: "<<endl;
for(i=0;i<M;i++){
    for(j=0;j<N;j++){
        cout << newimage[i][j]<<"\t";   // output the image
    }
    cout<<endl;
 }
 return 0;
}
```

Summary

- We wrote a program to enhance the contrast of a black-and-white image
- The program enhance_contrast.cpp is available in the courseware