

---

## HW2, Computer problem, Part (b)

### Histogram equalization

#### EECS 203A, Digital Image Processing. UCI. Fall 2004

#### Nasser Abbasi

Read the image file, cat.raw, and generate the histogram table, then the probability density function PDF table, then the cumulative density function CDF table. Then scan the image and generate new pixel gray level values by a lookup to the CDF.

```
In[495]:=
  Clear["Global`*"];
```

Now set current directory to where the image is (same folder as this note book) and read the image file

```
In[496]:=
  SetDirectory[ToFileName[Extract[
    "FileName" /. NotebookInformation[EvaluationNotebook[]], {1}, FrontEnd`FileName]]];

  fileName = "cat.raw";
  L = 256;
  rows = 480;
  cols = 640;
  data = FastBinaryFiles`ReadListBinary[fileName, Byte];
  nPixels = Length[data];
  Print["Max pixel value in data=", Max[data]];
  Print["Min pixel value in data=", Min[data]];

  Max pixel value in data=186

  Min pixel value in data=0
```

Now that the image is read into data, we display it before processing it.

In[505]:=

```
ListDensityPlot[Reverse[Partition[data, cols]], Mesh → False, Frame → False,  
ImageSize → {rows, cols}, PlotRange → All, AspectRatio → Automatic];
```



```
makeHistogram[data_] := Module[{h, i, r, nPixels, L},
  L = 256;
  Print["nPixels=", nPixels = Length[data]];
  h = Table[{0, 0}, {i, L}];
  Print["h table size=", Dimensions[h]];
  For[i = 1, i ≤ nPixels, i = i + 1,
    {
      (*If[Mod[i,10000]==0,Print["data["i,"]=",data[[i]]];*)
      r = data[[i]] + 1;
      h[[r, 2]] = h[[r, 2]] + 1;
    }
  ];
  Return[h];
];

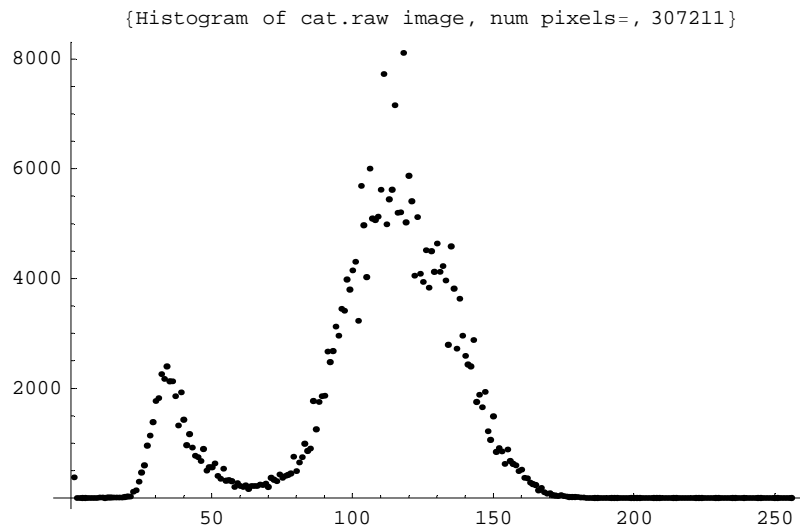
makePDF[h_, nPixels_] := Module[{i, pdf, L},
  L = Length[h];
  pdf = Table[{0, 0}, {i, L}];
  For[i = 1, i ≤ L, i = i + 1,
    pdf[[i, 2]] =  $\frac{h[[i, 2]]}{nPixels}$ ;
  ];
  Return[pdf];
];

makeCDF[pdf_] := Module[{cdf, L, i},
  L = Length[pdf];
  cdf = Table[{0, 0}, {i, L}];
  cdf[[1, 1]] = pdf[[1, 1]];

  For[i = 2, i ≤ L, i = i + 1,
    cdf[[i, 2]] = pdf[[i, 2]] + cdf[[i - 1, 2]];
  ];
  Return[cdf];
];
```

```
In[511]:=
```

```
(*Print the histogram*)  
(*TableForm[h]*)  
h = makeHistogram[data];  
ListPlot[h[[All, 2]],  
  PlotLabel -> {"Histogram of cat.raw image, num pixels=", Length[data]}]  
  
npixels=307211  
  
h table size={256, 2}
```



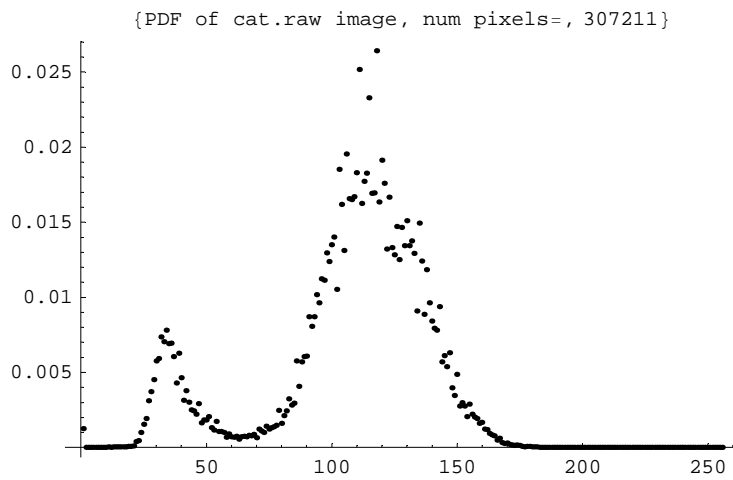
```
Out[512]=
```

```
- Graphics -
```

```
In[513]:=
```

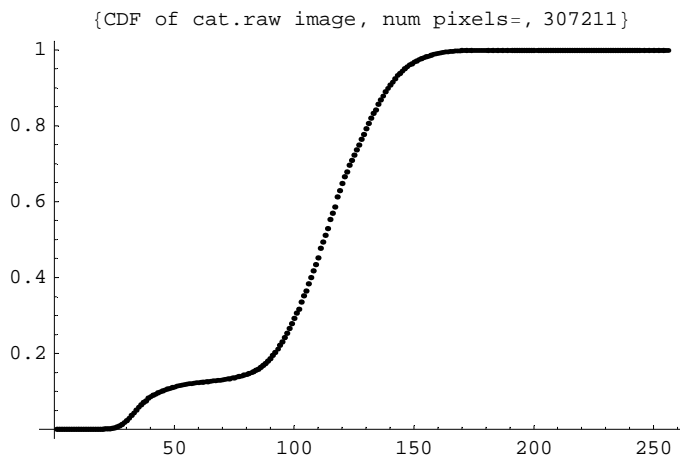
```
pdf = makePDF[h, Length[data]];
(*Print the PDF*)
ListPlot[pdf[[All, 2]], PlotLabel -> {"PDF of cat.raw image, num pixels=", nPixels}]

cdf = makeCDF[pdf];
(*Print the CDF*)
ListPlot[cdf[[All, 2]], PlotLabel -> {"CDF of cat.raw image, num pixels=", nPixels}]
```



```
Out[514]=
```

- Graphics -



```
Out[516]=
```

- Graphics -

In[517]:=

```
(*Now generate new image*)
nPixels = Length[data];
s = Table[0, {i, nPixels}];
For[i = 1, i ≤ nPixels, i = i + 1,
  r = data[[i]] + 1;
  s[[i]] = Round[ cdf[[r, 2]] (L - 1) ];
]
```

In[520]:=

```
Print["Max pixel value in s=", Max[s]];
Print["Min pixel value in s=", Min[s]];
```

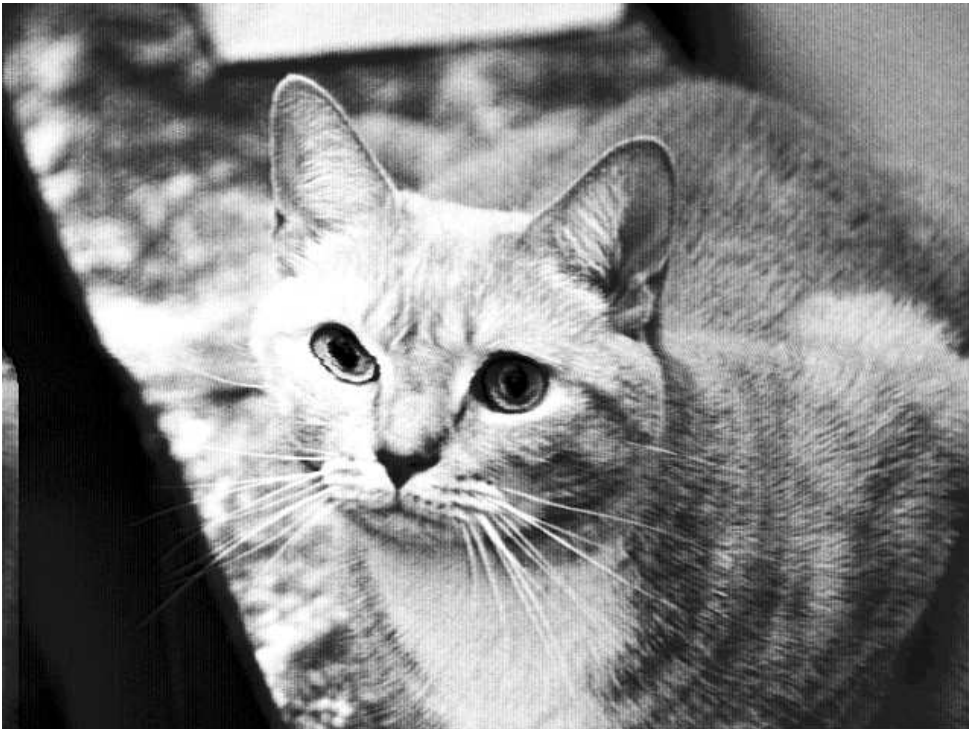
Max pixel value in s=255

Min pixel value in s=0

In[522]:=

```
(*Now display the new image*)
```

```
ListDensityPlot[Reverse[Partition[s, cols]], Mesh → False, Frame → False,
  ImageSize → {rows, cols}, PlotRange → All, AspectRatio → Automatic];
```

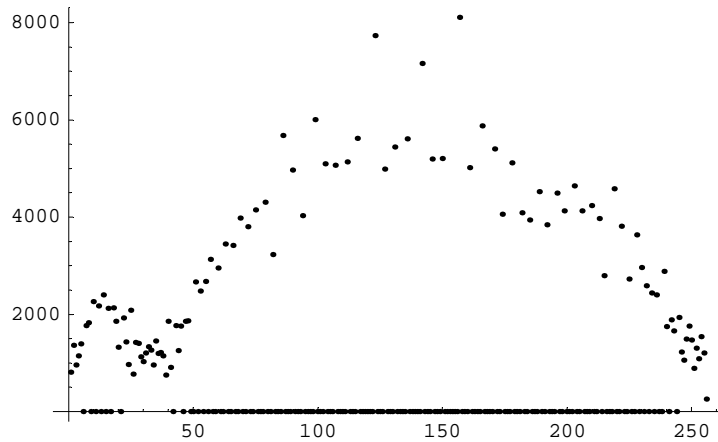


```
In[523]:=
(*Generate histogram and CDF for new image and plot to compare*)
h = makeHistogram[s];
ListPlot[h[[All, 2]],
  PlotLabel -> {"Histogram of transformed cat.raw image, num pixels=", Length[s]}]

npixels=307211

h table size={256, 2}
```

```
{Histogram of transformed cat.raw image, num pixels=, 307211}
```



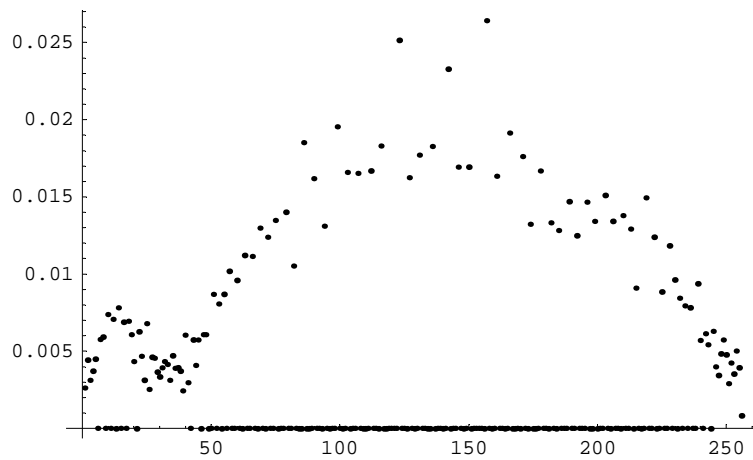
```
Out[524]=
```

```
- Graphics -
```

```
In[525]:=
```

```
pdf = makePDF[h, Length[s]];
ListPlot[pdf[[All, 2]],
PlotLabel -> {"PDF of transformed cat.raw image, num pixels=", nPixels}]
```

```
{PDF of transformed cat.raw image, num pixels=, 307211}
```

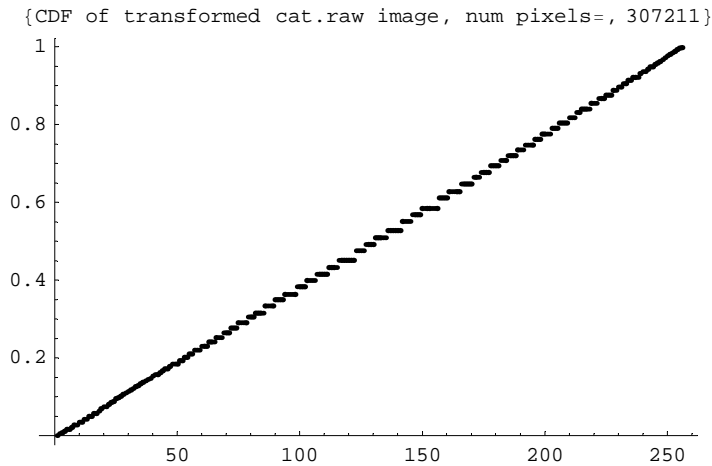


```
Out[526]=
```

```
- Graphics -
```



```
In[527]:=  
cdf = makeCDF[pdf];  
ListPlot[cdf[[All, 2]],  
PlotLabel -> {"CDF of transformed cat.raw image, num pixels=", nPixels}]
```



```
Out[528]=  
- Graphics -
```

```
In[533]:=  
  
(* Now write the new image to file *)  
fileName = "cat_histogram_equalized.raw";  
strm = FastBinaryFiles`OpenWriteBinary[fileName];  
FastBinaryFiles`WriteBinary[strm, s, Byte];  
Close[strm];  
  
(*That is all folks.... *)
```

---

## Conclusion

With histogram equalization applied, we see that new image contains more gray levels than the original level. Looking at generated plots of the new image histogram and PDF and CDF we see from the CDF that the distribution of gray levels is uniform in the new image.