HW3, Computer problem, part (b)
EECS 203A. UCI. FALL 2004

by Nasser Abbasi

This below shows the solution to this problem.
First initliaize session and load package needed to read/write binary files.

```
In[3]:= Remove["Global`*"];
        Install["binary"];
        << Graphics`Graphics`;
        << DiscreteMath`Combinatorica`;

In[7]:=
        SetDirectory[ToFileName[Extract[
            "FileName" /. NotebookInformation[EvaluationNotebook[]], {1}, FrontEnd`FileName]]];
```

Read the image file and display it.

```
In[8]:= process[fileName_, outFileName_] := Module[{},
          nRow = 480;
          nCol = 640;
          maskWidth = 5;  (*mask width/height*)
          data = FastBinaryFiles`ReadListBinary[fileName, Byte];
          nPixels = Length[data];
          Print["Dimensions of data read is =", Dimensions[data]];
          Print["Filter used is =", filter = Table[1, {maskWidth}, {maskWidth}]];

          data = Reverse[Partition[data, nCol]];

          hOriginalData = Histogram[Flatten[data],
            PlotLabel → {"Histogram of ", fileName, " BEFORE. mean=" , N[Mean[Flatten[data]]],
              "\nmedian=", N[Median[Flatten[data]]]}, DisplayFunction → Identity];

          Print["Dimensions of data after partition is =", Dimensions[data]];

          (*Display the image before processing *)
          ListDensityPlot[data, Mesh → False, Frame → False, ImageSize → {nRow, nCol},
           PlotRange → All, AspectRatio → Automatic, PlotLabel → {fileName,
             " Before. Number of pixels =" , nPixels, "\nResolution=", nRow, " by ", nCol}];

          ϵ = Floor[maskWidth / 2];
          firstCol = ϵ + 1;
          firstRow = ϵ + 1;
          lastRow = nRow - ϵ ;
          lastCol = nCol - ϵ ;

          newImage = Table[0, {nRow - 2 ϵ}, {nCol - 2 ϵ}];

          ii = 0;
          For[i = firstRow, i <= lastRow, i = i + 1,
```

```
  {
   ii = ii + 1;
   jj = 0;
   For[j = firstCol, j <= lastCol, j = j + 1,
    {
     jj = jj + 1;
     d = Take[data, {i - ϵ, i + ϵ}, {j - ϵ, j + ϵ}];
     newImage[[ii, jj]] = Median[Flatten[d]];
    }
   ]
  }
 ];

 (* normalize the new image gray level
   by dividing by number of coefficients in the filter *)
 (*newImage=Round[Chop[N[newImage/(maskWidth^2 )]]];*)

 Print["New image dimensions=", {r, c} = Dimensions[newImage]];

 ListDensityPlot[newImage, Mesh → False, Frame → False,
  ImageSize → {r, c}, PlotRange → All, AspectRatio → Automatic,
  PlotLabel → {fileName , " After Median filtering. Number of pixels =",
    Length[Flatten[newImage]], "\nResolution=", r, "by", c}];

 (* Now write the new image to file *)
 Print["Writing new image to file ", outFileName];

 strm = FastBinaryFiles`OpenWriteBinary[outFileName];
 FastBinaryFiles`WriteBinary[strm, Round[Chop[N[newImage]]], Byte];
 Close[strm];

 hNewImage = Histogram[Flatten[newImage], PlotLabel → {"Histogram of",
     fileName, " After Median filtering. mean=", N[Mean[Flatten[newImage]]],
     "\nmedian=", N[Median[Flatten[newImage]]]}, DisplayFunction → Identity];

 Print["display histogram of original image"];
 Show[hOriginalData, PlotRange → {{0, 255}, All},
  Frame → True, DisplayFunction → $DisplayFunction];

 Show[hNewImage, PlotRange → {{0, 255}, All},
  Frame → True, DisplayFunction → $DisplayFunction];


]
```

General::spell1 : Possible spelling error: new symbol name "nRow" is similar to existing symbol "Row". More…

In[9]:=

```
process["cat.raw", "cat_median_averaged.raw"];
process["triangle.raw", "triangle_median_averaged.raw"];
```

Dimensions of data read is ={307211}

Filter used is $=\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

Dimensions of data after partition is ={480, 640}

```
{cat.raw,  Before. Number of pixels =, 307211,
Resolution=, 480,  by , 640}
```

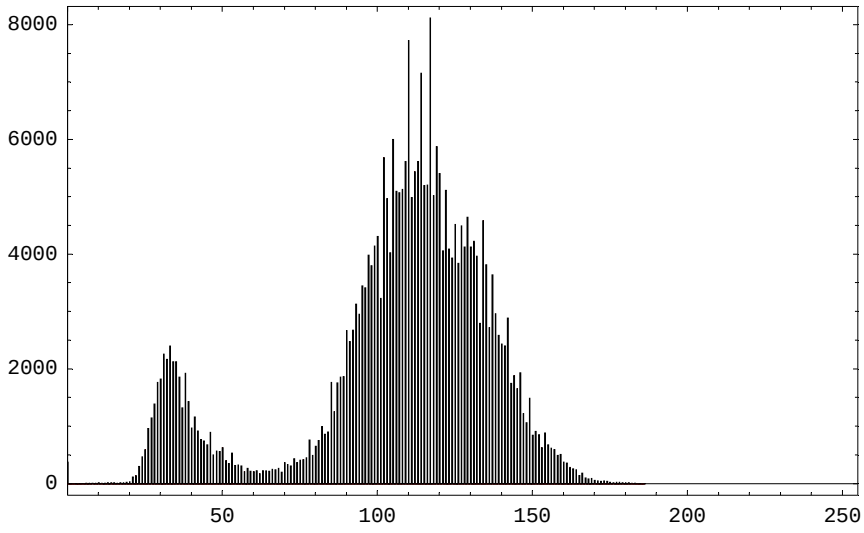
New image dimensions={476, 636}

```
{cat.raw,  After Median filtering. Number of pixels =, 302736,
Resolution=, 476, by, 636}
```

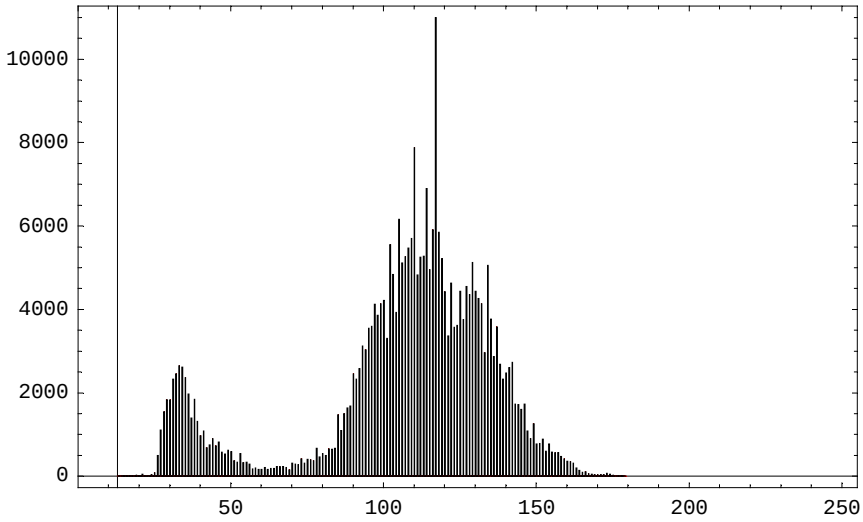
Writing new image to file cat_median_averaged.raw

display histogram of original image

{Histogram of , cat.raw, BEFORE. mean=, 105.799, median=, 112.}



{Histogram of, cat.raw, After Median filtering. mean=, 106.134, median=, 112.}

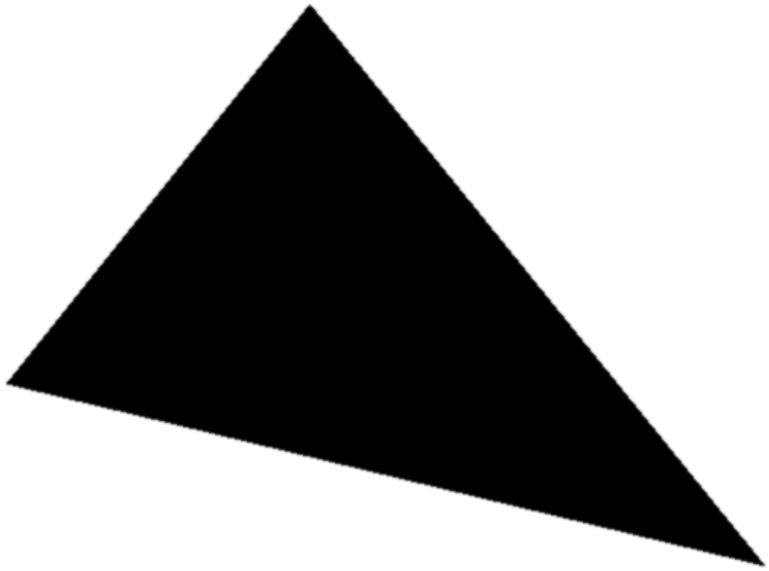


Dimensions of data read is ={307200}

Filter used is $=\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

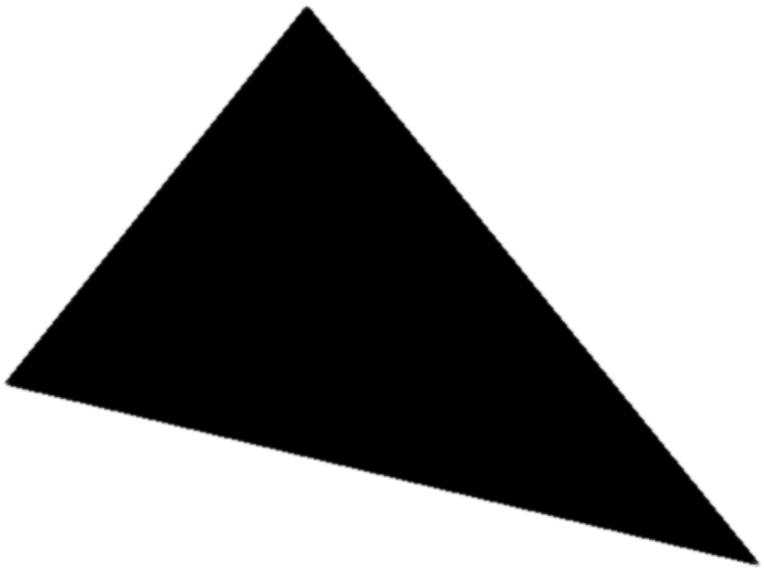
Dimensions of data after partition is ={480, 640}

```
{triangle.raw,  Before. Number of pixels =, 307200,
Resolution=, 480,  by , 640}
```



New image dimensions={476, 636}

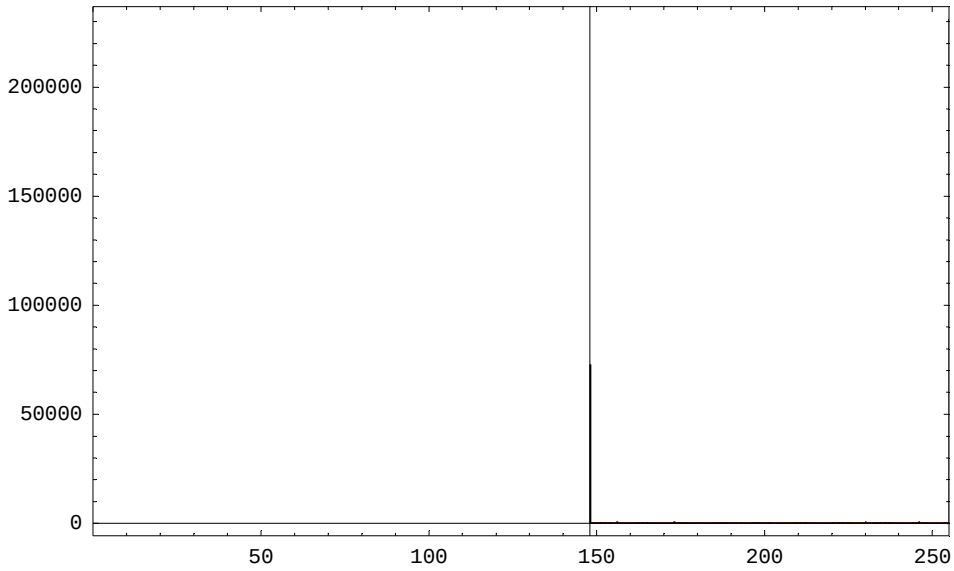
```
{triangle.raw,  After Median filtering. Number of pixels =, 302736,
Resolution=, 476, by, 636}
```



Writing new image to file triangle_median_averaged.raw

display histogram of original image

{Histogram of , triangle.raw, BEFORE. mean=, 229.063,
median=, 255.}



{Histogram of, triangle.raw, After Median filtering. mean=, 228.684,
median=, 255.}