HW 2, EECS 207A. Fall 2004. UCI.
First problem.

by Nasser Abbasi

■ Problem

The following table displays the pixel intensity values of a macro block of a 1-dimensional image
60, 75, 86, 200, 235, 255, 46, 34
a) compute the DCT coefficients of this function.
b) compute the original function from the DCT using the IDCT algorithm
c)Ignore the last 4 bytes and recompute the original function, how much error is introduced?

extra work: I also did an edditional analysis on this problem. I showed how the error in data changes as a function of number of terms dropped from the DCT table

■ Part(a)

Define the DCT and IDCT functions

In[923]:=

```
Clear["Global`*"];

buildDCT[i_, data_, c_] := Module[{L = Length[data]},
```
$$c[[i+1]] \sum_{n=0}^{L-1} \left(data[[n+1]] \, Cos\left[\frac{(2n+1)\, i\, \pi}{2L}\right]\right)$$
```
]

buildIDCT[i_, dct_, c_, endIndex_] := Module[{},
```
$$\sum_{n=0}^{endIndex} c[[n+1]] \, dct[[n+1]] \, Cos\left[\frac{(2i+1)\, n\, \pi}{2L}\right]$$
```
]
```

Define the data and create the C table

In[1005]:=

```
(*originalData={128,88,40,0,0,40,88,128};*)
(*originalData={0,0,255,255,255,0,0,0};*)
(*originalData={190,184,186,182,167,123,63,38};*)
originalData = {60, 75, 86, 200, 235, 255, 46, 34};

L = Length[originalData];
```

$$c = \text{Table}\left[\sqrt{\frac{2}{L}}, \{L\}\right];$$

$$c[[1]] = \sqrt{\frac{1}{L}};$$

Create the DCT values, and print them

In[1009]:=

```
dct = Table[buildDCT[i, originalData, c], {i, 0, L - 1}];
Print["DCT values = ", TableForm[Chop[N[dct]], TableDirections -> {Row, Column}]]

Print["Original data = ", TableForm[originalData, TableDirections -> {Row, Column}]]

Print["C constants table= ",
 TableForm[c, TableDirections -> {Row, Column}, TableSpacing -> {1, 6}],
 "\nC constants table= ",
 TableForm[N[c], TableDirections -> {Row, Column}, TableSpacing -> {1, 6}]]
```

DCT values = 350.371 −25.5532 −199.617 100.579 23.6881 −38.0348 36.3792 −58.615

Original data = 60 75 86 200 235 255 46 34

C constants table= $\frac{1}{2\sqrt{2}}$ $\frac{1}{2}$ $\frac{1}{2}$ $\frac{1}{2}$ $\frac{1}{2}$ $\frac{1}{2}$ $\frac{1}{2}$ $\frac{1}{2}$
C constants table= 0.353553 0.5 0.5 0.5 0.5 0.5 0.5 0.5

■ Part(b)

Use IDCT to recompute the original data, i.e. use all DCT points.

In[1013]:=

```
endIndex = Length[dct] - 1;
dataNew = Table[buildIDCT[i, dct, c, endIndex], {i, 0, L - 1}];
Print["data back from IDCT = ",
 TableForm[Chop[N[dataNew]], TableDirections -> {Row, Column}, TableSpacing -> {1, 6}]]
```

data back from IDCT = 60. 75. 86. 200. 235. 255. 46. 34.

■ Part(c)

In[1016]:=

```
bytesToIgnore = 4;
endIndex = Length[dct] - bytesToIgnore - 1;
dataNewShort = Table[buildIDCT[i, dct, c, endIndex], {i, 0, L - 1 }];
Print["data back from IDCT = ", TableForm[Chop[N[dataNewShort]],
  TableDirections -> {Row, Column}, TableSpacing -> {1, 6}]]
error = dataNew - dataNewShort;
Print["Absolut error       = ",
 TableForm[ N[error] , TableDirections -> {Row, Column}, TableSpacing -> {1, 6}]]
Print["% error             = ",
 TableForm[ N[Abs[error]/dataNew 100] , TableDirections -> {Row, Column}, TableSpacing -> {1, 6}]]
```

```
data back from IDCT = 60.9473  65.2456  105.648  185.654  246.518  218.492  106.114  2.38102

Absolut error       = -0.947258  9.75437  -19.6484  14.3461  -11.5178  36.5084  -60.1144  31.619

% error             = 1.57876  13.0058  22.847  7.17306  4.90121  14.317  130.683  92.997
```

■ Extra work. Generate table showing how much error (in percentage) in the data recomputed as we drop more terms in the DCT. Try from 1 to 7 terms dropped.

In[1023]:=

```
L = Length[dct];
data = Table[i, {i, 1, L - 1}, {j, 1, L}];
For[n = 1, n ≤  L - 1,  n = n + 1,
 {
  endIndex = L - n;
  data⟦n, All⟧ = Table[buildIDCT[i, dct, c, endIndex], {i, 0, Length[dct] - 1 }];
 }
]
```

Data recomputed, each row shows the data based on dropping as many terms from DCT as the row number-1.
For example, the first row shows the data recomputed if we dropped ZERO terms from DCT.
The second row shows the data recomputed if dropped ONE term from DCT, etc...

In[1130]:=

```
Print["Original input data\n",
 TableForm[originalData, TableDirections -> {Row, Column}],
 "\nTable showing the recomputed data values as more DCT terms are dropped\n"
  MatrixForm[Chop[N[data⟦All, All⟧]]]]
```

```
Original input data
 60  75  86  200  235  255  46  34
Table showing the recomputed data values as more DCT terms are dropped
```

60.	75.	86.	200.	235.	255.	46.	34.
65.7176	58.7176	110.368	171.256	263.744	230.632	62.2824	28.2824
58.7567	75.5226	93.5633	178.216	270.705	213.827	79.0874	21.3215
69.3223	56.8706	97.2734	194.029	254.893	210.117	97.7394	10.756
60.9473	65.2456	105.648	185.654	246.518	218.492	106.114	2.38102
19.133	75.0566	154.972	213.593	218.578	169.168	96.3034	44.1953
111.344	113.252	116.777	121.382	126.368	130.973	134.498	136.406

In[1144]:=

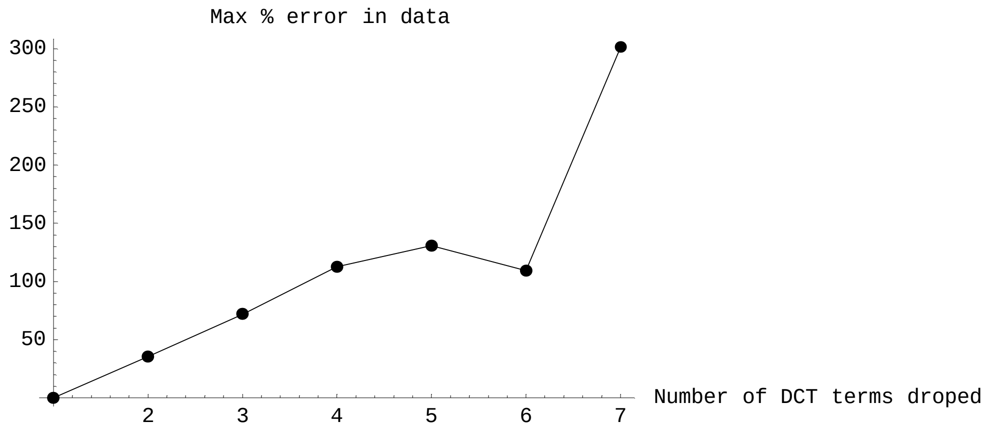
```
(* now compute the error in % *)
error = Table[i, {i, 1, L - 1}, {j, 1, L}];
maxerror = Table[i, {i, 1, L - 1}];
averageError = Table[i, {i, 1, L - 1}];

For[n = 1, n ≤ L - 1, n = n + 1,
 {

  ϵ = Chop[N[originalData - data[[n, All]]]];
  error[[n, All]] = N[Abs[ ϵ ]/originalData 100];
  maxerror[[n]] = Max[error[[n, All]]];
  averageError[[n]] = (Sum[error[[n, k]], {k, 1, L}])/L;
 }
]
Print["Table showing the % erorr in each data term as more DCT terms are dropped",
  MatrixForm[error]];

Print["Average error in data as more DCT terms are dropped       ]\n",
 TableForm[ N[averageError] , TableSpacing -> {1, 6}]]

g1 = ListPlot[maxerror, PlotStyle → PointSize[0.02],
   PlotLabel -> "Max % error in data ", AxesLabel → {"Number of DCT terms droped", ""},
   PlotJoined → True, DisplayFunction → Identity, PlotRange → All];
g2 = ListPlot[maxerror, PlotStyle → PointSize[0.02],
   PlotRange → All, PlotLabel -> "Max % error in data ",
   AxesLabel → {"Number of DCT terms droped", ""} , DisplayFunction → Identity];
Show[{g1, g2}, DisplayFunction → $DisplayFunction]


g1 = ListPlot[averageError, PlotStyle → PointSize[0.02], PlotLabel ->
    "Average % error in data ", AxesLabel → {"Number of DCT terms droped", ""},
   PlotJoined → True, PlotRange → All, DisplayFunction → Identity];
g2 = ListPlot[averageError, PlotStyle → PointSize[0.02], PlotLabel ->
    "Average % error in data ", AxesLabel → {"Number of DCT terms droped", ""},
   PlotJoined → False, PlotRange → All, DisplayFunction → Identity];
Show[{g1, g2}, DisplayFunction → $DisplayFunction]
```

Table showing the % erorr in each data term as more DCT terms are dropped

0.	0.	0.	0.	0.	0.	0.	0.
9.52935	21.7098	28.3352	14.3722	12.2317	9.5562	35.3965	16.8165
2.07208	0.696835	8.79453	10.8918	15.1937	16.1464	71.9291	37.2896
15.5371	24.1725	13.1086	2.98556	8.46504	17.6013	112.477	68.3647
1.57876	13.0058	22.847	7.17306	4.90121	14.317	130.683	92.997
68.1116	0.0755142	80.1996	6.79663	6.98789	33.6595	109.355	29.9861
85.5731	51.0022	35.7868	39.3088	46.2266	48.6379	192.388	301.194

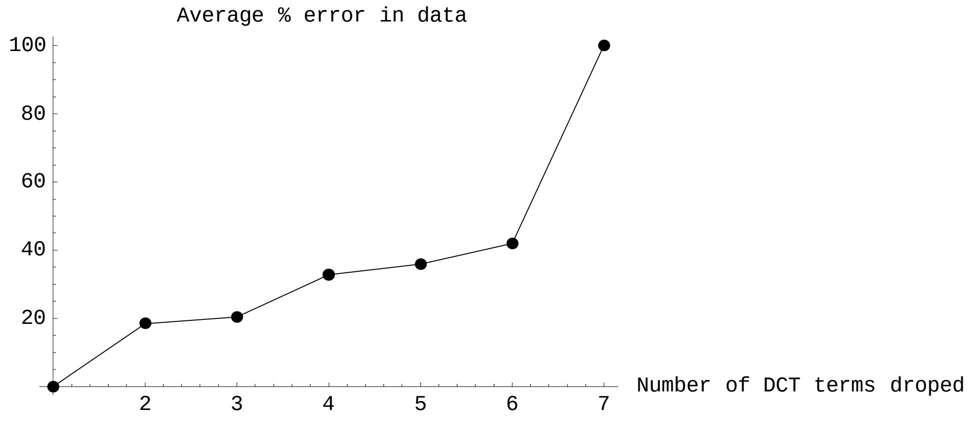
Average error in data as more DCT terms are dropped

0.
18.4934
20.3768
32.839
35.9379
41.8965
100.015



Out[1152]=

- Graphics -



Out[1155]=

- Graphics -

Conclusion

The Max error in the recomputed data array increases as more DCT terms are dropped. But that is not always true each time. In this example, as we dropped the 6th term in the DCT table, the max error was actually smaller, but we see that the average error in data is always increasing as expected. I am not sure now why the max error in data did not go down every time we dropped an additional term from the DCT. This needs more investigation. I have tried this on another data input and saw the same result.