framedecfile

February 9, 2017

1 Program - framedecfile

Program to decode a video from files framedim.txt, y00enc.bin, y01enc.bin, and y10enc.bin, on the Y component, using 2-bit DCT coefficient values - Gerald Schuller, Dec. 2015 * Import relevant modules:

```
In [1]: import cv2
        import numpy as np
        import scipy.fftpack as sft
        #import our file functions:
        from writereadbits import *
        import blockdct

        N=8
```

writereadbits.py

- Read DC values of the DCT's from file:

```
In [2]: #indices00=readbitsfile('y00enc.bin')
        codestring00=readbinaryfile('y00enc.bin')
        indices00=codestring2data(codestring00)
```

- Subtract smallest index value to obtain original value range:

```
In [3]: indices00=indices00+2;
```

- Read AC values from file:

```
In [4]: #indices01=readbitsfile('y01enc.bin')
        codestring01=readbinaryfile('y01enc.bin')
        indices01=codestring2data(codestring01)

        #indices10=readbitsfile('y10enc.bin')
        codestring10=readbinaryfile('y10enc.bin')
        indices10=codestring2data(codestring10)
```

- Reshape back into 2-D frame with rindex rows and cindex solumns:

```
In [5]: #load dimensions from info file:
        [r,c]=np.loadtxt('framedim.txt')
        rindex=r/8;
        cindex=c/8;
        indices00=np.reshape(indices00,(-1,cindex))
        indices01=np.reshape(indices01,(-1,cindex))
        indices10=np.reshape(indices10,(-1,cindex))

        #print('De-Quantisieren')
```

- de-quantization in the decoder:

```
In [6]: Xrek=np.zeros((r,c));
```

- Number of bits per pixel

```
In [7]: bits=2
```

- Resulting quantization step size for 2^bits steps:

```
In [8]: #Stufen fuer unterschiedliche Ortsfrequenzen:
        #DC Indices mit range 0...5:
        quantstufeDC=5.0/(2**bits-1)

        #Zwei AC Koeffizienten, mit range 0.5-(-0.5)
        quantstufeAC=1.0/(2**bits-1)
```

- DC values de-quantization:

```
In [9]: Xrek[0::N,0::N]=indices00*quantstufeDC
```

- 2 AC values de-quantization:

```
In [10]: Xrek[0::N,1::N]=indices01*quantstufeAC
         Xrek[1::N,0::N]=indices10*quantstufeAC
```

- The rest of the DCT coefficients is not transmitted and set to zero.

```
In [11]: #Inverse 2D DCT:
         x=blockdct.invdct8x8(Xrek)
```

```
In [12]: while(True):
             cv2.imshow('Decoder mit De-Quantizer und Inverse 2D DCT, mit 0.095 bits/Pixel! ', x

             #Keep window open until key 'q' is pressed:
             if cv2.waitKey(1) & 0xFF == ord('q'):
                 break
```

- When everything done, release the capture:

```
In [13]: #cap.release()
         cv2.destroyAllWindows()
```