

Image and video compression

Chapter 3 - Quantization

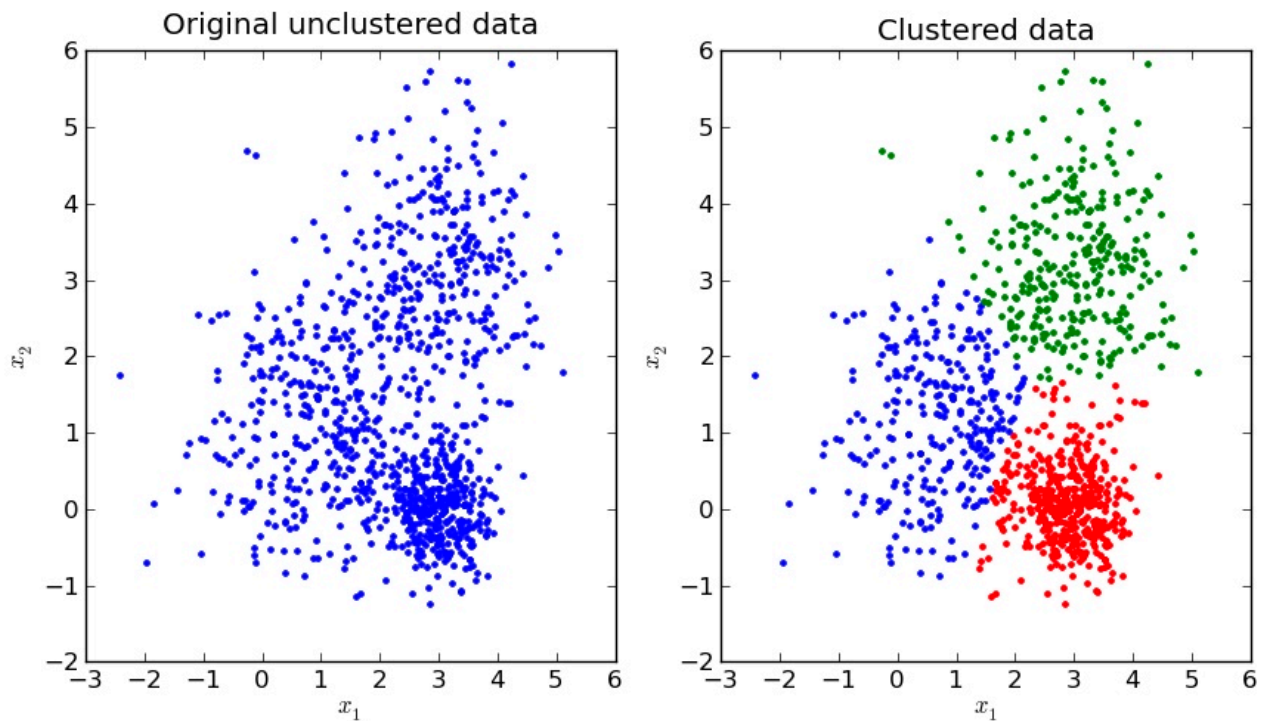
The target of chapter 3 is to change the content of the picture. try to use less code to describe the image.

UniQuant

将一张图片按照平均分配减少尺寸

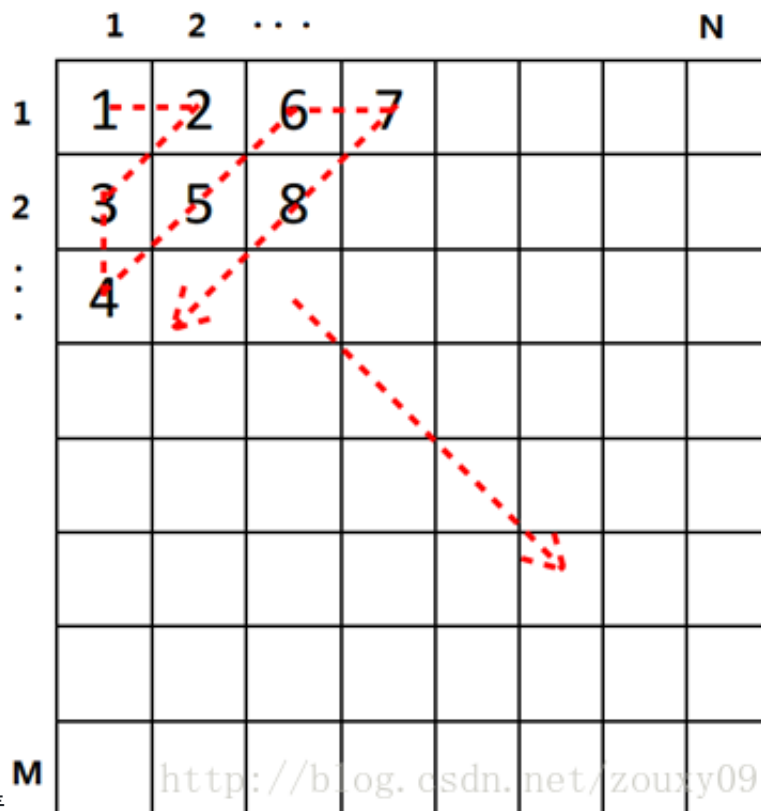
LloydMax

按照Machine Learning的方式将图片分类，每一次改变cluster， untill cluster is unchanged.



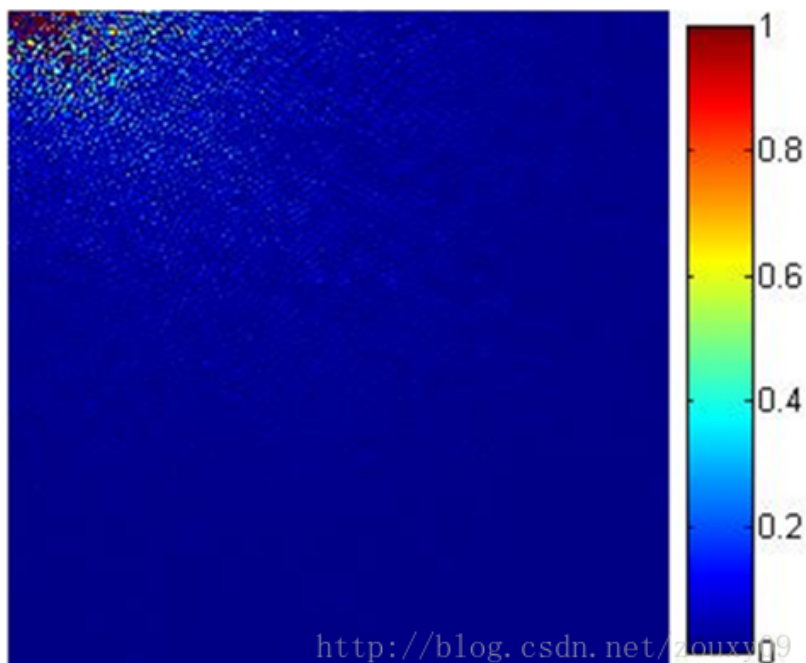
Chapter 4 Transform Coding

Zigzag



zigzag模式提取矩阵元素

这个东西用在哪呢？用在离散余弦变换的系数提取里面。离散余弦变换（DCT）是种图像压缩算法，JPEG-2000好像就是用它来进行图像压缩的。DCT将图像从像素域变换到频率域。然后一般图像都存在很多冗余和相关性的，所以转换到频率域之后，只有很少的一部分频率分量的系数才不为0，大部分系数都为0（或者说接近于0），这样就可以进行高效的编码，以达到压缩的目的。下图的右图是对lena图进行离散余弦变换（DCT）得到的系数矩阵图。从左上角依次到右下角，频率越来越高，由图可以看到，左上角的值比较大，到右下角的值就很小很小了。换句话说，图像的能量几乎都集中在左上角这个地方的低频系数上面了。



DCT

DCT又称离散余弦变换，是一种块变换方式，只使用余弦函数来表达信号，与傅里叶变换紧密相关。常用于图像数据的压缩，通过将图像分成大小相等（一般为8*8）的块，利用DCT对其进行变换，得到更加简洁的数据。因为图像像素间存在较大的空间相关性，DCT可以大大减小这些相关性，使图像能量集中在左上角区域，从而利于数据压缩。变换后得到的数据称为DCT系数。这一过程是无损的。

二维DCT变换

$$F(u, v) = c(u)c(v) \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i, j) \cos \left[\frac{(i+0.5)\pi}{N} u \right] \cos \left[\frac{(j+0.5)\pi}{N} v \right]$$

$$c(u) = \begin{cases} \sqrt{\frac{1}{N}}, & u = 0 \\ \sqrt{\frac{2}{N}}, & u \neq 0 \end{cases}$$

$c(u)$ 和 $c(v)$ 为添加的系数，主要作用为使DCT变换矩阵为正交矩阵。 $F(u,v)$ 即为DCT变换系数，可以通过矩阵形式来表示：

$$F = AfA^T$$

$$A(i, j) = c(i) \cos \left[\frac{(j+0.5)\pi}{N} i \right]$$

A即为正交矩阵，通过F和A逆变换即可恢复图像数据。

IntraEncode IntraDecode

```
function dst = IntraEncode(image, qScale)
% Function Name : IntraEncode.m
% Input : image (Original RGB Image)
% qScale(quantization scale)
% Output : dst (sequences after zero-run encoding, 1xN)
imageYUV = ictRGB2YCbCr(image);
EOB = 1000;
%DCT Transform
imageYUV_DCT = blockproc(imageYUV, [8, 8], @(block_struct) DCT8x8(block_struct.data));
imageYUV_quant = blockproc(imageYUV_DCT, [8, 8], @(block_struct)
Quant8x8(block_struct.data, qScale));
imageYUV_zz = blockproc(imageYUV_quant, [8, 8], @(block_struct)
ZigZag8x8(block_struct.data));
```

```
dst = ZeroRunEnc_EoB(imageYUV_zz(:), EOB);
```

```
end
```

将一个图片的上述的高频能量数取出

IntraDecode和之相反，通过k值反向计算出图片

Chapter 5 Motion Compensation

这一章试图计算每一个block的运动趋势。从而简化传输的数据