

# 多媒体Project1说明文档

## 1、对代码的简单说明

均匀量化：

```
function [a_quan]=u_pcm(a,n)
%U_PCM    uniform PCM encoding of a sequence
%         [A_QUAN]=U_PCM(A,N)
%         a=input sequence. 输入信号
%         n=number of quantization levels (even).量化级数
%         a_quan=quantized output before encoding.

% todo:
a_max = max(abs(a));    % 获取采样点的最大值
a_quan = a ./ a_max;    % 将采样点映射到 (-1, 1)
for i = -1 : 2 / n : 1 % 将(-1, 1)分成n段，即量化级数
    % 获取在(i, i + 2 / n)范围内的采样点
    a_quan_seg = a_quan(a_quan >= i & a_quan < (i + 2 / n));

    % 将这些采样点设为(最大值 + 最小值) / 2
    a_quan(a_quan >= i & a_quan < (i + 2 / n)) = a_max * (max(a_quan_seg) +
min(a_quan_seg)) / 2;
end

end
```

把输入范围等分为n份，步长为 $2/n$ ，在这个区间范围内的点值都设为这个区间的最大值和最小值之和的一半

$\mu$ 律非均匀量化

```
function [z]=ulaw(y,u)
% u-law nonlinearity for nonuniform PCM
% X=ULAW(Y,U).
% Y=input vector.

% todo:
%  $\mu$ 律压缩函数
z = sign(y) .* log(1 + u * abs(y ./ max(abs(y)))) ./ log(1 + u);
end
```

这里直接对于整个输入向量采用 $\mu$ 律非均匀量化公式

```
function [a_quan]=ula_pcm(a,n,u)
%ULA_PCM u-law PCM encoding of a sequence
%         [A_QUAN]=MULA_PCM(X,N,U).
```

```

%      X=input sequence.
%      n=number of quantization levels (even). 量化级数
%      a_quan=quantized output before encoding.
%      U the parameter of the u-law  $\mu$ 律参数

% todo:
x = ulaw(a, u); %  $\mu$ 律压缩函数
y = u_pcm(x, n); % 均匀量化
a_quan = inv_ulaw(y, u); %  $\mu$ 律扩张函数( $\mu$ 律压缩函数的反函数)

end

```

然后还是经过一遍上面的均匀量化，最后实施信号扩张，还原原来的信号量

```

function x=inv_ulaw(y,u)
%INV_ULAW the inverse of u-law nonlinearity
%X=INV_ULAW(Y,U) X=normalized output of the u-law nonlinearity.

% todo:
x = sign(y) .* (((1 + u) .^ (abs(y)) - 1) ./ u); % 公式, ulaw的反函数

end

```

## 2、阐述使用非均匀量化的优点

非均匀量化是根据信号的不同区间来确定量化间隔的。对于信号取值小的区间，其量化间隔也小；反之，量化间隔就大。与均匀量化相比，它有两个主要的优点：

- 当输入量化器的信号具有非均匀分布的概率密度时，非均匀量化器的输出端可以较高的平均信号量化噪声功率比
- 非均匀量化时，量化噪声功率的均方根值基本上与信号抽样值成比例，也就是在信号抽样值越小，其量化噪声功率的均方根值越小，其信噪比越大，所以非均匀量化在处理小信号时，可以得到较好的量化信噪比。而在均匀量化中，量化误差的最大瞬时值等于量化间隔的一半，这对于小信号来说可能会比较大，因此小信号并不适合均匀量化，而是适合非均匀量化。

所以综合而言，非均匀量化能大大减少量化信号所需的编码位数。