

Wireless Networking - Fundamentals and Applications

第九組

Activity #2 CamCom Manchester

- 通訊協定
 - 定義 T 為 $\frac{1}{30}$ 秒
 - 開頭有 $4T$ 的 Header，其內容為 [0001]，這樣可以保證影片中可以以一個全暗的畫面和 01 訊號當開頭
 - 接下來把要傳輸的訊息編碼，每個 bit 佔 $2T$
 - 1 為 [01]
 - 0 為 [10]
- 程式步驟
 - encode.m
 - input：所要傳輸的訊息
 - tx_vec(1*N)：所要上傳的資料
 - decode.m
 - junk_frame：預留影片開頭亮度不穩而不處理的幀數
 - init：找到全暗的畫面後才開始處理後面的畫面
 - gmx1,gmi1,gmx2,gmi2：紀錄影片中上下兩排中分別最亮和最暗的灰階數
 - gap1,gap2：(gmx+gmi)/2，判斷一排為亮或暗的標準
 - output(1*N)：紀錄亮暗的序列
 - bin：影片所要傳遞的真正訊息
 - 流程
 1. 找出整支影片中亮度最小的一幀
 2. 將整支影片每一幀的灰階皆減去亮度最小的一幀的灰階數值
 3. 找出整支影片中最亮的一排跟最暗的一排的灰階數值，根據此二數值定出判斷一排為亮或暗的標準。
 4. 將每一幀的第一排和最後一排的亮暗紀錄成一個數列
 5. 如果數列不是 [AABBCCDD...] 的形式，或是 $A \neq B$ or $C \neq D$...，則一直將數列的第一項移除直到符合前面所述的形式形式
 6. 其中 AABB 為一個字元的訊息，CCDD 為二個字元的訊息...
 7. 移除數列的前四項(header部分)。
 8. 用 output(i*4) 即可取得原本的訊息。
- code
 - encode.m

```
%tx_vec : signal samples
input = '^' ;
bin=de2bi(uint8(char(input)));
%disp(bin);
input = zeros(1,8-size(bin,2));
for i = 1:size(bin,2)
```

```

        input=[input,bin(size(bin,2)-i+1)];
    end
    %bin=padarray(bin,8)
    disp(input);
    x = -pi:0.01:pi;
    tx_vec = sin(x);
    tx_vec = [];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,ones(1,6666)];
    for i = 1:8
        if input(i)
            tx_vec = [tx_vec,-ones(1,6666)];
            tx_vec = [tx_vec,ones(1,6666)];
        else
            tx_vec = [tx_vec,ones(1,6666)];
            tx_vec = [tx_vec,-ones(1,6666)];
        end
    end
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    tx_vec = [tx_vec,-ones(1,6666)];
    %save signal sample in the file
    fid = fopen('txdata.bin','w');
    %disp(real(tx_vec));
    fwrite(fid, [real(tx_vec); imag(tx_vec)], 'float');
    fclose(fid);

```

- decode.m

```

file_name='team9_v10_^.bin.mp4';
junk_frame=3;
v = VideoReader(file_name);
mx=[];
mxv=999999999999;
cnt=0;
while hasFrame(v)
    video = readFrame(v);
    signal = sum(rgb2gray(video),2);
    cnt=cnt+1;
    if cnt<junk_frame
        continue;
    end
    if sum(signal)==0
        continue;
    end
end

```

```

        if sum(signal)<mxv
            mxv=sum(signal);
            mx=signal;
        end
    end

v = VideoReader(file_name);
init=0;
output=[];
gap=-100000;
gmx1=-9999999;
gmi1=9999999;
gmx2=-9999999;
gmi2=9999999;
cnt=0;
while hasFrame(v)
    video = readFrame(v);
    signal = sum(rgb2gray(video),2);
    cnt=cnt+1;
    if cnt<junk_frame
        continue;
    end
    if sum(signal)==0
        continue;
    end
    signal=signal-mx;
    if sum(signal)<100000000
        init=1;
    end
    if init==0
        continue;
    else
        gmx1=max(signal(1),gmx1);
        gmx2=max(signal(end),gmx2);
        gmi1=min(signal(1),gmi1);
        gmi2=min(signal(end),gmi2);
    end
end

v = VideoReader(file_name);
init=0;
output=[];
%gap=(gmx+gmi+gmi)/3;
gap1=(gmx1+gmi1)/2;
gap2=(gmx2+gmi2)/2;
%gap=5000;
%disp(["gap1=",gap1]);
%disp(["gap2=",gap2]);
cnt=0;
while hasFrame(v)
    video = readFrame(v);
    signal = sum(rgb2gray(video),2);

```

```

        cnt=cnt+1;
        if cnt<junk_frame
            continue;
        end
        if sum(signal)==0
            continue;
        end
        signal=signal-mx;
        if signal(1)<gap1 && signal(end)<gap2
            init=1;
        end
        if init==0
            continue;
        else
            output=[output,signal(1)>gap1];
            output=[output,signal(end)>gap2];
            %disp([signal(1)>gap1,signal(end)>gap2])
        end
        %plot(signal);
        %axis([0, 1200, -1000000, 1000000]);
        %pause();
    end
    %disp(output);
    while 1
        ok=1;
        for i = 1:16
            if output(i*2-1) ~= output(i*2)
                ok=0;
            end
        end
        for i = 1:8
            if output(i*4-2) == output(i*4-1)
                ok=0;
            end
        end
        if ok==0 || output(1) ~= 0
            output = output(2:end);
        else
            break;
        end
    end
    output = output(5:end);
    bin=[];
    for i = 1:8
        bin=[bin,output(i*4)];
    end
    bin=fliplr(bin);
    disp(bin);
    disp(char(bin2de(bin)));

```