

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
BỘ MÔN TÍN HIỆU HỆ THỐNG



BÁO CÁO MÔN HỌC
MÔ PHỎNG HTTT

Giảng viên: Nguyễn Thị Thu Hằng

Họ và tên: Lương Đức Thuận

Mã sinh viên: B19DCVT399

STT: 54

Nhóm: 05

Hà Nội 5/2022

LỜI CẢM ƠN

Em xin chân thành cảm ơn cô Nguyễn Thị Thu Hằng đã tạo điều kiện cho chúng em làm bài tập lớn môn Mô phỏng hệ thống truyền thông. Đây là môn học quan trọng, giúp chúng em củng cố kiến thức từ các môn học cũ và rèn luyện tư duy lập trình thông qua ngôn ngữ Matlab.

Dù rất cố gắng để hoàn thiện bài tập lớn, nhưng trong quá trình làm bài không tránh khỏi nhiều thiếu sót, mong cô có thể châm chước cho em. Em hy vọng có cơ hội được học cô trong các kì học tới.

Em xin chân thành cảm ơn!

MỤC LỤC

I. NHIỆM VỤ 1	4
a. Mô tả phương pháp thực hiện chuyển đổi ảnh số được lưu trong tệp thành chuỗi nhị phân và ngược lại.	4
b. Viết chương trình MATLAB thực hiện chuyển đổi ảnh lưu trong tệp 2022PicVietnam.jpg thành chuỗi tín hiệu nhị phân và ngược lại. Có thể xây dựng dưới dạng hàm chuyển đổi để sử dụng trong các chương trình khác.	4
c. Xác định các thông số về kích thước ảnh, định dạng ảnh và độ dài chuỗi bit được chuyển đổi.	5
II. NHIỆM VỤ 2	5
a. Mô tả hệ thống mô phỏng bằng sơ đồ khối và xác định các tham số của hệ thống, có thể bao gồm cả các bộ lọc sử dụng để có được bộ thu tối ưu.	5
b. Bằng việc sử dụng MATLAB, viết chương trình mô phỏng hệ thống truyền dẫn số sử dụng kỹ thuật điều chế đã lựa chọn trên kênh AWGN với nguồn tín hiệu là tín hiệu thu được ở nhiệm vụ 1. Ước tính xác suất lỗi tại các mức tỉ số tín hiệu trên nhiễu SNR lần lượt bằng 5, 8 và 12 dB theo phương pháp Monte Carlo.	7
c. Biểu diễn biểu đồ chòm sao, dạng sóng tín hiệu, mẫu mắt và phổ của tín hiệu tại các điểm sau trên hệ thống: đầu ra bộ điều chế, sau khi truyền qua kênh AWGN tại SNR = 8dB, sau khi được xử lý và khôi phục tại bộ thu.	7
d. So sánh tệp đồ họa được khôi phục sau khi truyền qua hệ thống mô phỏng tại các mức SNR yêu cầu.	11
III. TÀI LIỆU THAM KHẢO	13
IV. PHỤ LỤC (CODE CHƯƠNG TRÌNH)	14
1. Code nhiemvu1	14
2. Code nhiemvu2	14

I. NHIỆM VỤ 1

- a. Mô tả phương pháp thực hiện chuyển đổi ảnh số được lưu trong tệp thành chuỗi nhị phân và ngược lại.
 - Bước 1: Dùng hàm **imread** đọc ảnh màu từ tệp đồ họa sẽ trả ra ma trận X có 3 chiều có kích thước 432x624x3, trong đó:
 - X(:, :, 1) chứa thông tin của màu đỏ cho từng điểm ảnh
 - X(:, :, 2) chứa thông tin của màu xanh lá cây cho từng điểm ảnh
 - X(:, :, 3) chứa thông tin của màu xanh lam cho từng điểm ảnh
 - Bước 2: Chuyển đổi ảnh sang chuỗi nhị phân, sử dụng lệnh **reshape**
 - Bước 3: Chuyển đổi chuỗi nhị phân về thành tệp ảnh 8 bit để hiển thị, sau đó dùng lệnh **imwrite**.
- b. Viết chương trình MATLAB thực hiện chuyển đổi ảnh lưu trong tệp 2022PicVietnam.jpg thành chuỗi tín hiệu nhị phân và ngược lại. Có thể xây dựng dưới dạng hàm chuyển đổi để sử dụng trong các chương trình khác.

```

1 %nhiemvul
2 function [kt, b] = nhienvul()
3 %Buoc 1
4 img = '2022PicVietnam.jpg';
5 y = imread(img); %Doc anh
6 %Buoc 2
7 %y = rgb2gray(x); %Convert to Gray
8 kt = size(y); %Kich thuoc anh
9 %buoc 3
10 y = de2bi(y); %Chuyen doi nhi phan
11 b = reshape(y, 1, []); %Chuoi nhi phan sau khi reshape
12 %buoc 4
13 y = reshape(b, kt(1)*kt(2)*kt(3), 8); %Reshape ve ma tran 8 bit
14 y = bi2de(y); %Chuyen ve thap phan
15 y = reshape(y, kt(1), kt(2), kt(3)); %Reshape ve kich thuoc hinh anh
16 imwrite(y, 'rs.jpg'); %Luu anh
17 imfinfo('rs.jpg')
18 end
    
```

Kết quả:



- c. Xác định các thông số về kích thước ảnh, định dạng ảnh và độ dài chuỗi bit được chuyển đổi.

struct with fields:

```

        Filename: 'D:\learn\K2N3\MPHTTT\bt1\rs.jpg'
    FileModDate: '14-May-2022 15:17:38'
    FileSize: 68867
    Format: 'jpg'
    FormatVersion: ''
    Width: 624
    Height: 432
    BitDepth: 24
    ColorType: 'truecolor'
    FormatSignature: ''
    NumberOfSamples: 3
    CodingMethod: 'Huffman'
    CodingProcess: 'Sequential'
    Comment: {}
    
```

- Ảnh có kích thước 432x624 sử dụng định dạng .JPG

```
>> length(b)
```

```
ans =
```

```
6469632
```

```
>> kt
```

```
kt =
```

```
432    624    3
```

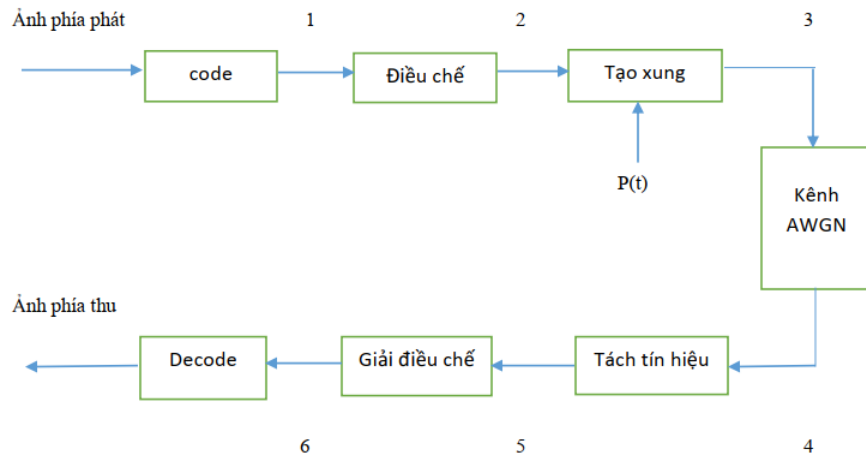
- Ảnh màu, độ dài chuỗi bit $b = 6469632$.

II. NHIỆM VỤ 2

- a. Mô tả hệ thống mô phỏng bằng sơ đồ khối và xác định các tham số của hệ thống, có thể bao gồm cả các bộ lọc sử dụng để có được bộ thu tối ưu.

- Lựa chọn tham số hệ thống:
 - Tốc độ dữ liệu: $R_b = 9 \text{ Mb/s}$
 - SNR mô phỏng tại: 5, 8, 12 dB
 - Bậc điều chế: $M = 64 \Rightarrow$ Điều chế: 64-QAM
 - Số bit trên 1 ký hiệu: $n_b = 6 \text{ bit}$
 - Số lượng ký hiệu: $n_{\text{syms}} = \text{length}(b)/n_b = 1078272$
 - Chu kỳ bit $T_b = 1/R_b$
 - Chu kỳ tín hiệu $T_s = n_b \cdot T_b$
 - Năng lượng mỗi ký hiệu: $E_s = 1$
 - Pha ban đầu: $\text{pha} = 0$

- Sơ đồ khối của hệ thống mã hóa và giải mã:



- 1 - Các chuỗi bit nhị phân
 - 2 – Chuỗi các giá trị $d(k)$ phức
 - 3 – Xung tín hiệu phát $s(t)$
 - 4 – Xung tín hiệu $s_r(t)$ sau khi thu
 - 5 – Chuỗi các giá trị $d_r(k)$ phức
 - 6 – Các bit nhị phân.
- Quá trình mô phỏng
 - Chuỗi tín hiệu là chuỗi bit thu được từ nhiệm vụ 1.
 - Điều chế 64-QAM chuyển chuỗi bit nhị phân thành chuỗi kí hiệu dạng phức d .
 - Thực hiện tạo xung cho các kí hiệu p_m , mỗi xung lấy 100 giá trị mẫu.

$$p(t) = \sqrt{\frac{2E_s}{T_{sym}}} \left[1 - \cos\left(\frac{2\pi t}{T_{sym}}\right) \right]$$

- Sau khi có giá trị các mức kí hiệu d và biểu diễn xung cho các kí hiệu p_m , tạo tín hiệu phát s dùng công thức:

$$s(t) = \left[\sum_{k=-\infty}^{\infty} d_k p(t - kT_{sym}) \right] e^{j\phi_0}$$

- Đưa chuỗi kí hiệu phức lên kênh awgn, dựa vào các giá trị đã cho của đề bài.
- Sau khi thu được tín hiệu s_r thì chia cho giá trị biên độ hàm xung để thu về lại các kí hiệu phức.
- Thực hiện giải điều chế QAMDEMOD các kí hiệu phức, thu được chuỗi bit nhị phân.
- Tính BER, vẽ biểu đồ.

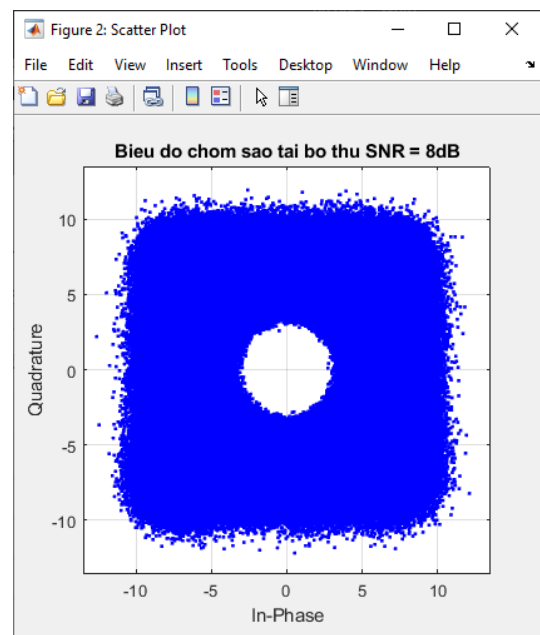
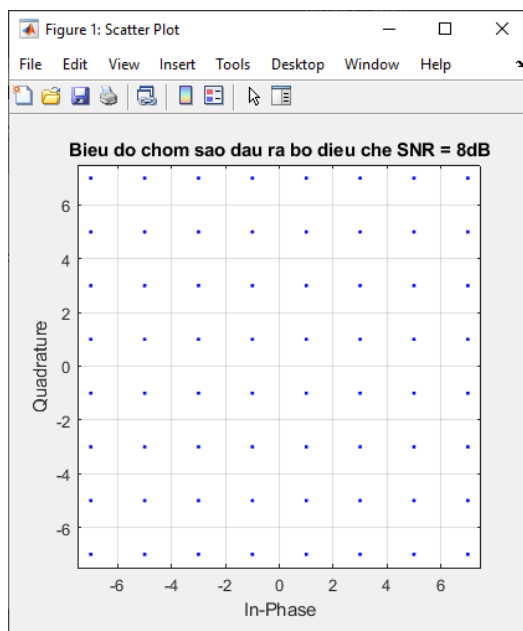
- b. Bằng việc sử dụng MATLAB, viết chương trình mô phỏng hệ thống truyền dẫn số sử dụng kỹ thuật điều chế đã lựa chọn trên kênh AWGN với nguồn tín hiệu là tín hiệu thu được ở nhiệm vụ 1. Ước tính xác suất lỗi tại các mức tỉ số tín hiệu trên nhiễu SNR lần lượt bằng 5, 8 và 12 dB theo phương pháp Monte Carlo.

- Xác suất lỗi tại các SNR là:

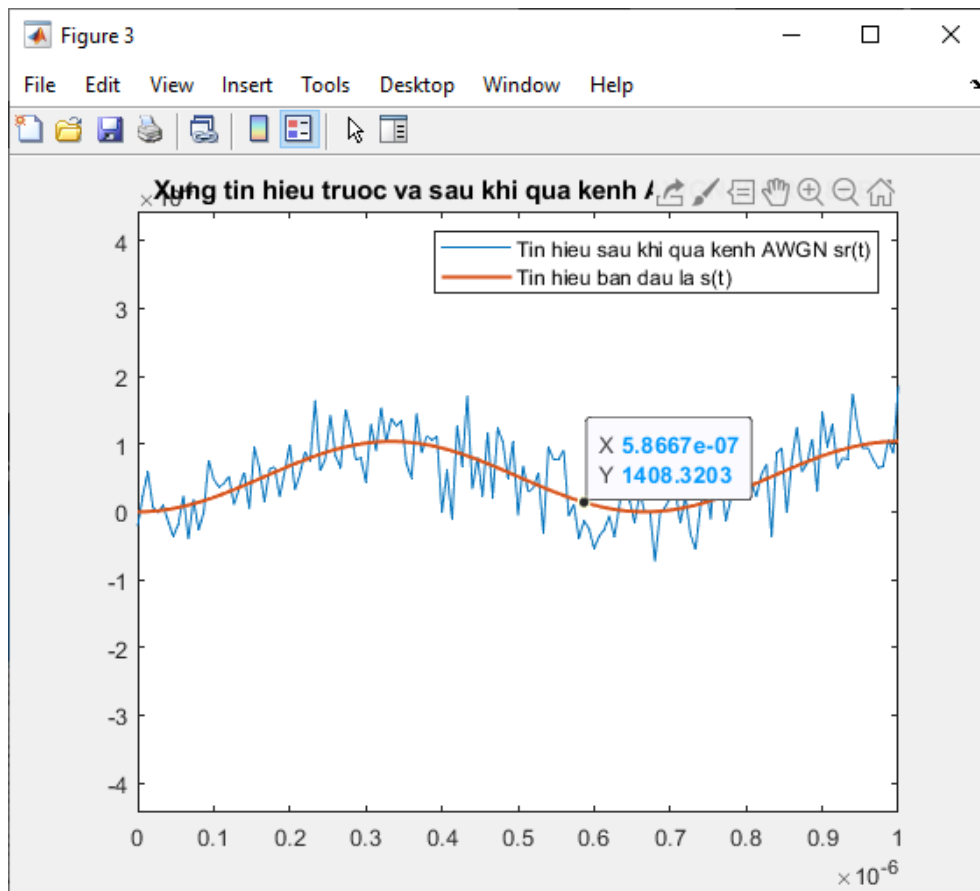
```
>> nhiemvu2
Tai SNR = 5, co BER = 0.250426
Tai SNR = 8, co BER = 0.175706
Tai SNR = 12, co BER = 0.098777
```

- Nhận xét: Tỉ số tín hiệu trên nhiễu càng cao thì xác suất lỗi bit BER càng thấp.

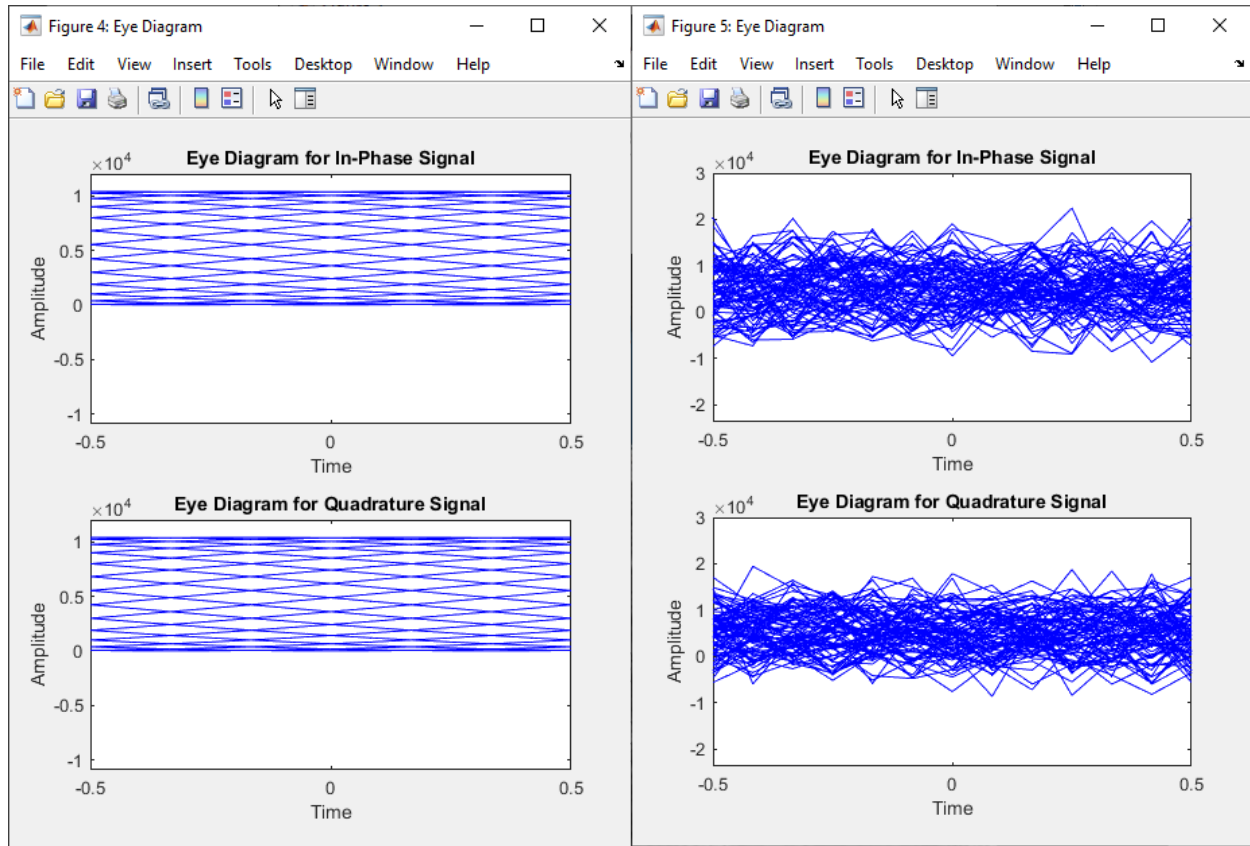
- c. Biểu diễn biểu đồ chòm sao, dạng sóng tín hiệu, mẫu mắt và phổ của tín hiệu tại các điểm sau trên hệ thống: đầu ra bộ điều chế, sau khi truyền qua kênh AWGN tại SNR = 8dB, sau khi được xử lý và khôi phục tại bộ thu.



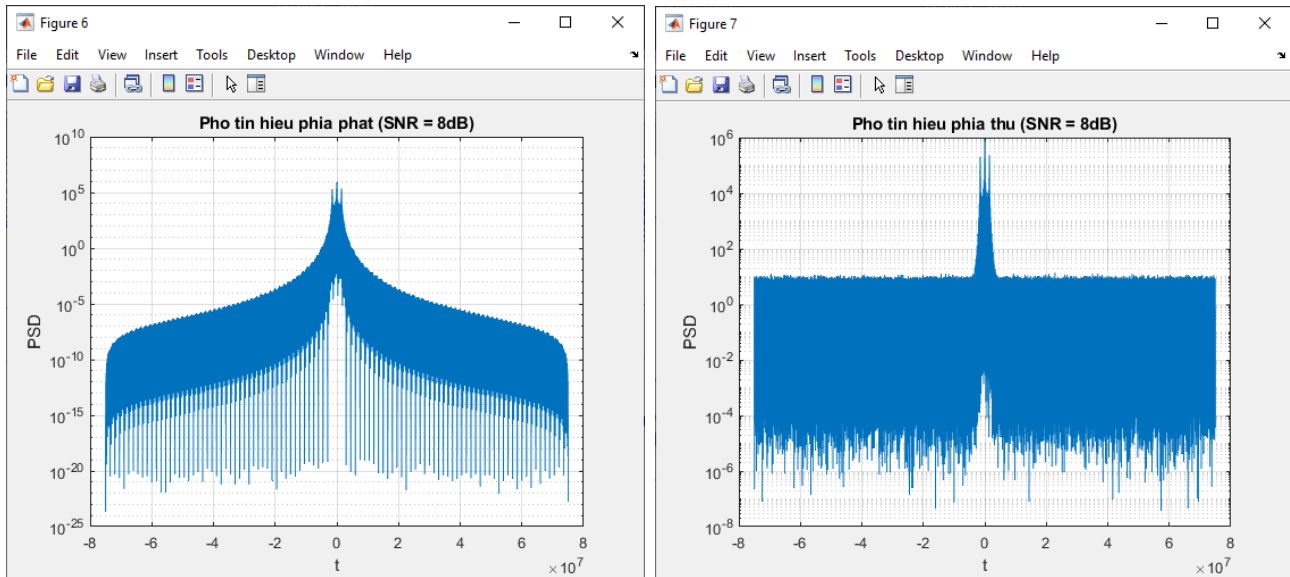
- Tín hiệu trước khi qua kênh gồm 64 chòm sao đều nhau tương ứng điều chế 64-QAM
- Ở phía thu tín hiệu bị ảnh hưởng bởi nhiễu nên các kí hiệu không còn đồng đều.



- Dạng sóng tín hiệu tại $\text{SNR} = 8$, xung tín hiệu trước và sau khi qua kênh awgn.
- Đường màu đỏ thể hiện cho tín hiệu ban đầu.
- Sau khi qua kênh awgn, tín hiệu bị ảnh hưởng bởi nhiễu nên xung tín hiệu bị biến dạng.

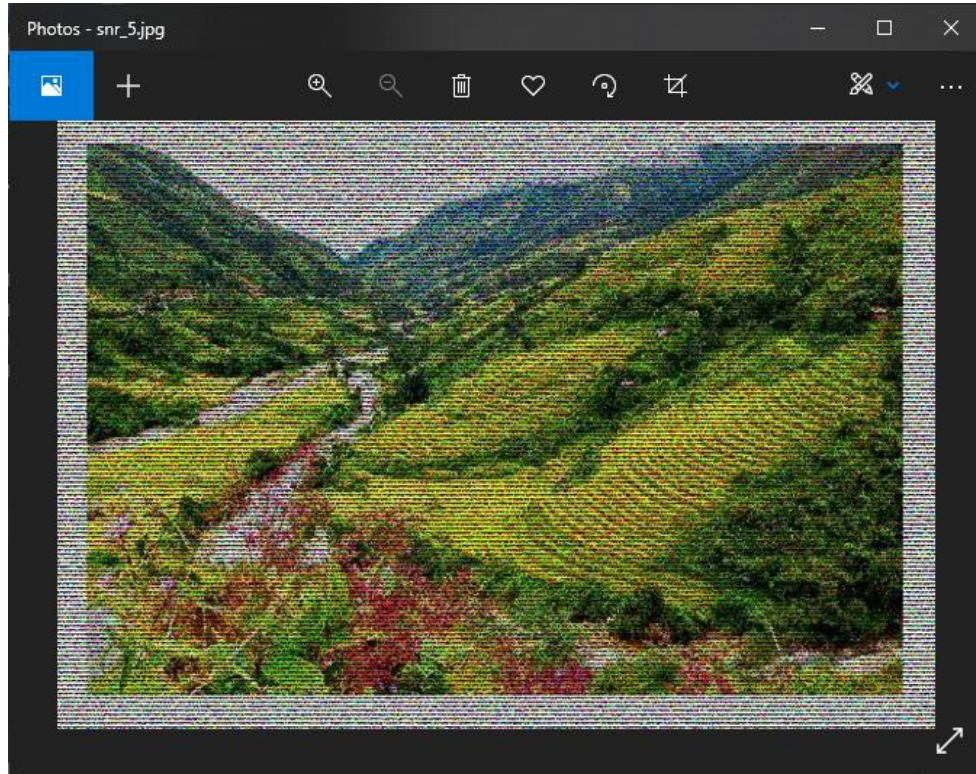


- Tín hiệu đầu vào chưa có ảnh hưởng của nhiễu nên sau khi qua một lượng chu kỳ nhất định biểu đồ mắt ổn định có quy luật và trật tự rõ ràng.
- Tín hiệu đi qua kênh awgn bị nhiễu kênh truyền. Do sự ảnh hưởng của nhiễu nên sau mỗi chu kỳ dạng sóng tín hiệu bị thay đổi nên mỗi chu kỳ đè lên nhau gây ra dạng mẫu mắt bị rối.

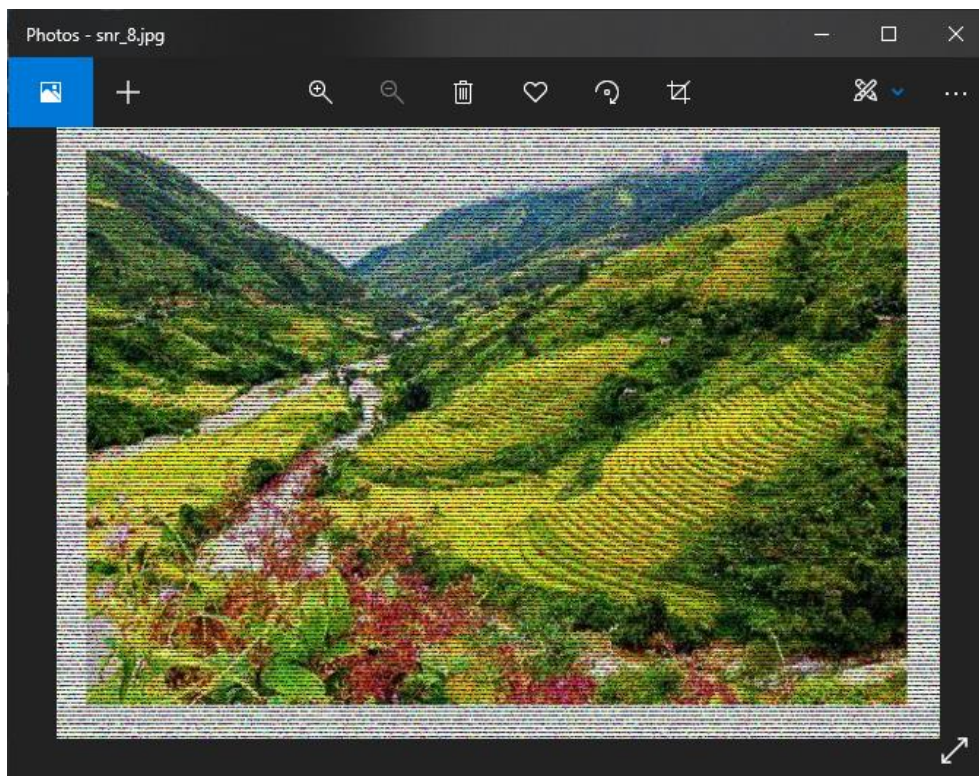


- Mật độ phổ công suất của tín hiệu sau khi có nhiễu đã tăng lên do ảnh hưởng của công suất nhiễu.
- Tại tần số trung tâm dạng phổ của tín hiệu khi không có tạp và tín hiệu có tạp thì dạng phổ này vẫn biểu hiện khá rõ và mật độ phổ công suất của nó không chênh lệch quá nhiều.

- d. So sánh tệp đồ họa được khôi phục sau khi truyền qua hệ thống mô phỏng tại các mức SNR yêu cầu.
- Ảnh được khôi phục tại mức SNR = 5 dB.



- Ảnh được khôi phục tại mức SNR = 8 dB.



- Ảnh được khôi phục tại $\text{SNR} = 12 \text{ dB}$.



- Nhận xét:
 - Khi SNR càng tăng thì hình ảnh càng rõ nét.
 - Do SNR là tỉ lệ tín hiệu trên tạp, nên khi SNR tăng thì ảnh hưởng của tạp nên tín hiệu càng thấp, ảnh được khôi phục càng giống với ảnh gốc.
 - Tăng độ khôi phục ảnh bằng cách tăng SNR.

III. TÀI LIỆU THAM KHẢO

- 1. SV-2014-ThuvienPTIT-Baigiang_MPHTTT*
- 2. Slide-MophongHTTT-Ch4*
- 3. Slide-MophongHTTT-Ch5*

IV. PHỤ LỤC (CODE CHƯƠNG TRÌNH)

1. Code nhiemvu1

```
%nhiemvu1
function [kt, b] = nhiemvu1()
%Buoc 1
img = '2022PicVietnam.jpg';
y = imread(img); %Doc anh
%Buoc 2
%y = rgb2gray(x); %Convert to Gray
kt = size(y); %Kich thuoc anh
%buoc 3
y = de2bi(y); %Chuyen doi nhi phan
b = reshape(y, 1, []); %Chuoai nhi phan sau khi reshape
%buoc 4
y = reshape(b, kt(1)*kt(2)*kt(3), 8); %Reshape ve ma tran 8 bit
y = bi2de(y); %Chuyen ve thap phan
y = reshape(y, kt(1), kt(2), kt(3)); %Reshape ve kich thuoc hinh anh
imwrite(y, 'rs.jpg'); %Luu anh
imfinfo('rs.jpg')
end
```

2. Code nhiemvu2

```
%Nhiemvu2
%MSV: B19DCVT399 => M = 64; N = 9
%Lay chuoai bit tu nhiemvu1
[kt, b] = nhiemvu1();

%Khai bao tham so
M = 64; %Bac dieu che
N = 9;
Rb = N*1e6; %Toc do bit
SNR_dB = [5 8 12]; %SNR theo dB
SNR = 10.^(SNR_dB/10); %SNR
nb = log2(M); %So bit bieu dien 1 ki hieu
n_syms = length(b)/nb; %Do dai chuoai ki hieu
Tb = 1/Rb; %Chu ki bit
Ts = nb*Tb; %Chu ki ki hieu

%Dieu che 64QAM chuyen doi chuoai nhi phan b sang chuoai ky tu d.
%Thuc hien tao dang xung p cho moi ki hieu, lay mau 100 lan.
%Tao tin hieu phat s
d = qammod(b', M, 'InputType', 'bit'); %Dieu che QAM phuc cho cac ki hieu
n_sample = 100; %So mau cho 1 tin hieu
Es = 1; %Nang luong moi ki hieu
pha = 0; %Pha ban dau
t = 0:Ts/n_sample:n_syms*Ts-Ts/100; %Thoi gian cho moi ki hieu
p_m = sqrt(2*Es/Ts)*(1-cos(2*pi*t/Ts)); %Tao xung p(t)
d = kron(d', ones(1, n_sample)); %Lap lai moi ki hieu 100 lan
s = d.*p_m.*exp(i*pha); %Xung tin hieu s(t)

%Dua chuoai s len canh AWGN theo SNR.
%Sau khi thu tin hieu, lay cac gia tri max cua tin hieu thu chia cho bien
%do xung ki hieu de thu duoc duoc chuoai ki hieu dr.
for i = 1:length(SNR)
    nMC = 10; %So vong lap cho Monte Carlo loop
    err = 0; %So loi bit
    BER = 0;
    for j = 1:nMC
        sr = awgn(s, SNR(i), 'measured', 'linear'); %Dua qua canh awgn

        dra = max(reshape(sr, n_sample, n_syms)); %Lay gia tri max cua tin hieu thu
        drb = max(p_m); %Lay gia tri max cua xung ki hieu
        dr = (dra/drb); %tin hieu/ bien do xung

        bits_output = qamdemod(dr', M, 'OutputType', 'bit'); %Giai dieu che 64-QAM
        err = err + length(b(b'~= bits_output)); %Tinh so luong bit loi
    end
    BER = err/(nMC*length(b));
end
```

```

%Ve do thi tai SNR = 8
if SNR_dB(i) == 8
    %Ve bieu do chom sao
    scatterplot(d); title('Bieu do chom sao dau ra bo dieu che SNR = 8dB'); grid;
    scatterplot(dr); title('Bieu do chom sao tai bo thu SNR = 8dB'); grid;

    %Ve xung tin hieu
    figure, plot(t, real(sr)); hold on;
    plot(t, real(s), 'LineWidth', 1.5); hold off;
    title('Xung tin hieu truoc va sau khi qua kenh AWGN (SNR = 8dB)');
    legend('Tin hieu sau khi qua kenh AWGN sr(t)', 'Tin hieu ban dau la s(t)');
    axis([0 0.000001 -inf inf]);

    %Ve mau mat
    eyediagram(s(1:1000), 2*nb); %Bieu do mat phat
    eyediagram(sr(1:1000), 2*nb); %Bieu do mat thu

    %Ve pho tin hieu
    [f, Xf] = spectrocal(t, s);
    figure, semilogy(f, Xf); grid; title('Pho tin hieu phia phat (SNR = 8dB)');
    xlabel('t'); ylabel('PSD');
    [f, Xf] = spectrocal(t, sr);
    figure, semilogy(f, Xf); grid; title('Pho tin hieu phia thu (SNR = 8dB)');
    xlabel('t'); ylabel('PSD');
end

%Ket qua tinh BER cho tung truong hop
fprintf('Tai SNR = %d, co BER = %f \n', SNR_dB(i), BER);

%Tep do hoa o cac muc SNR
bits_output = reshape(bits_output, [], 8);
bits_output = uint8(reshape(bi2de(bits_output), kt(1), kt(2), kt(3)));
imwrite(bits_output, "snr_" + num2str(SNR_dB(i)) + ".jpg");
end

```