OFDM Final Project

EE-442: Wireless Device Algorithms

20.12.2022

Outline

Introduction

- System Design
- Implementations

Demonstration

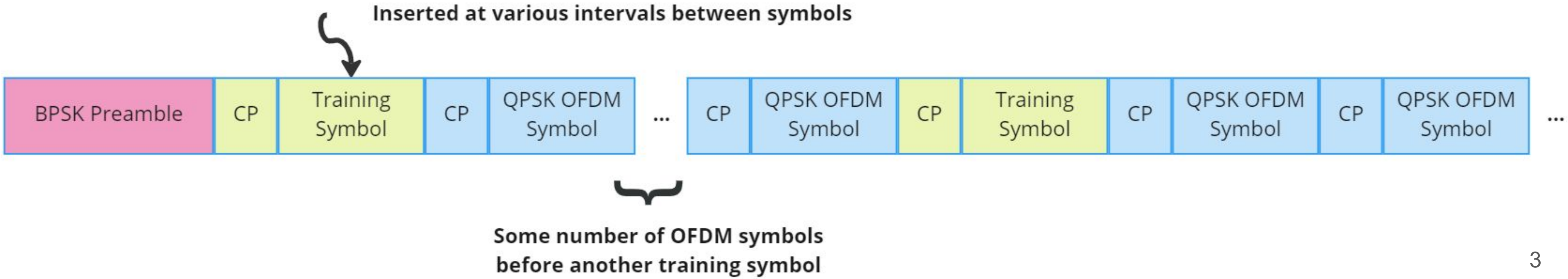
- Image Transmission
- Problems Encountered & Solutions

Analysis

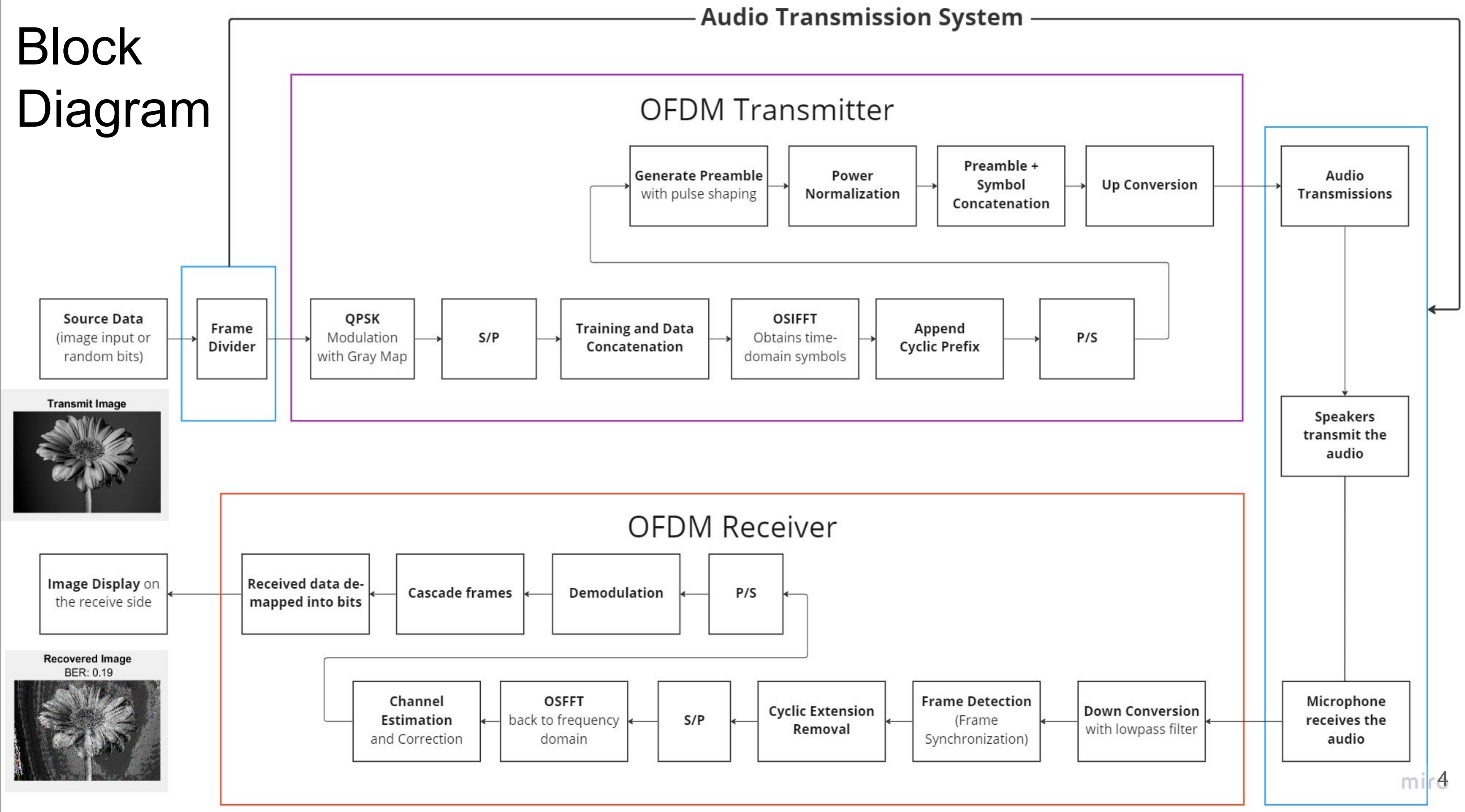
- Experiment Results
- Model Evaluations
- Future Work & Areas of Improvement

Introduction

- MATLAB-based implementation of an OFDM (Orthogonal Frequency Division Multiplexing) acoustic transmission system
- Two audio speakers + microphone

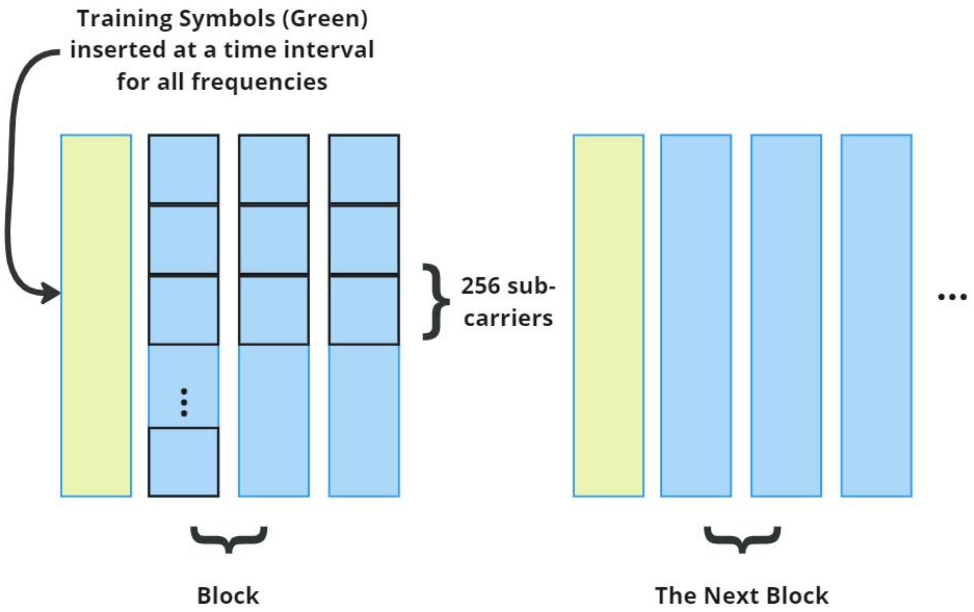


Block Diagram



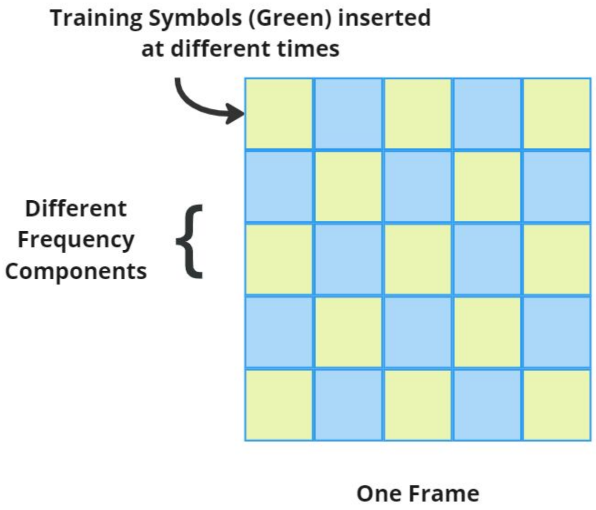
Training Types

Block Pilot Training



$$\widehat{h_l} = argmin_{\widehat{h_l}} \{|y_l - H_l T_l|^2\} = \frac{y_l}{T_l}$$

Comb Pilot Training

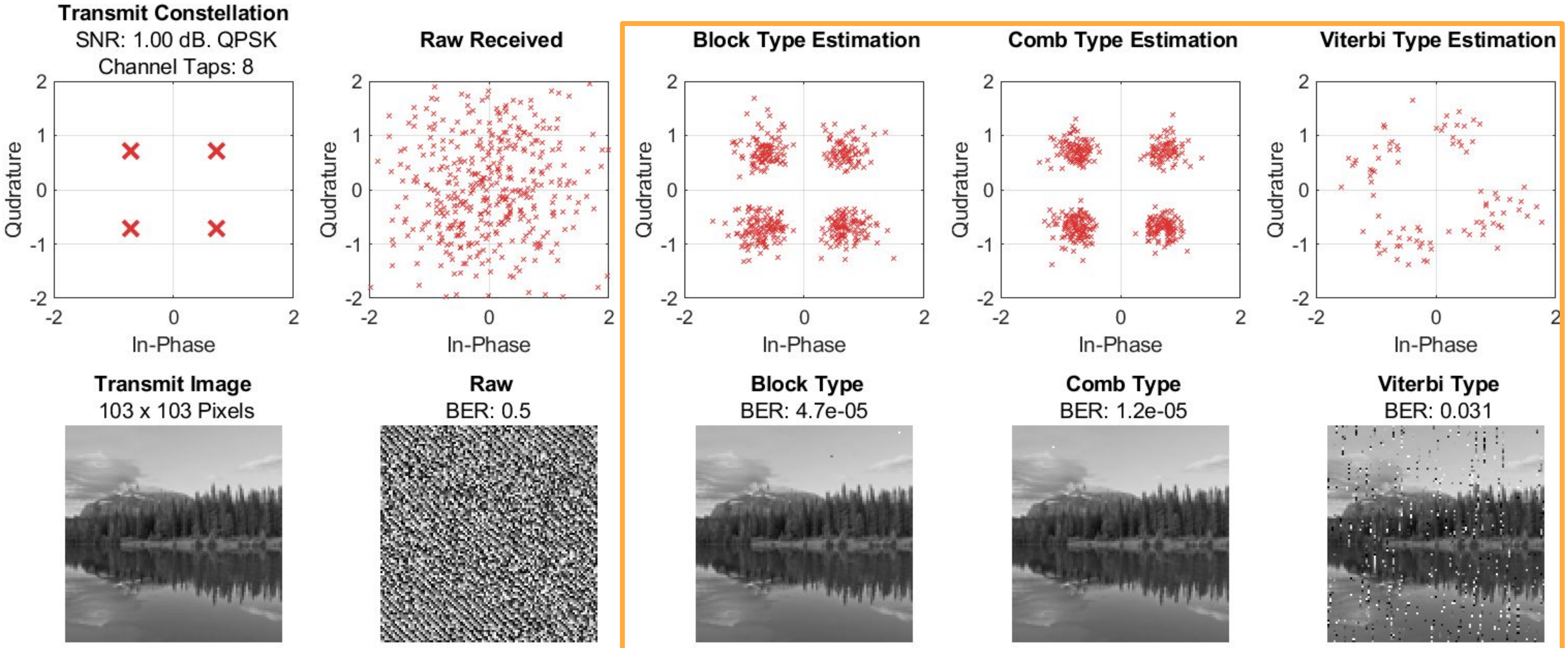


$$\hat{h} = (TF_{N\times L})^{\dagger} Y = (F^H_{N\times L} T^H T F_{N\times L})^{-1} F^H_{N\times L} T^H Y$$

Exp Fading + AWGN

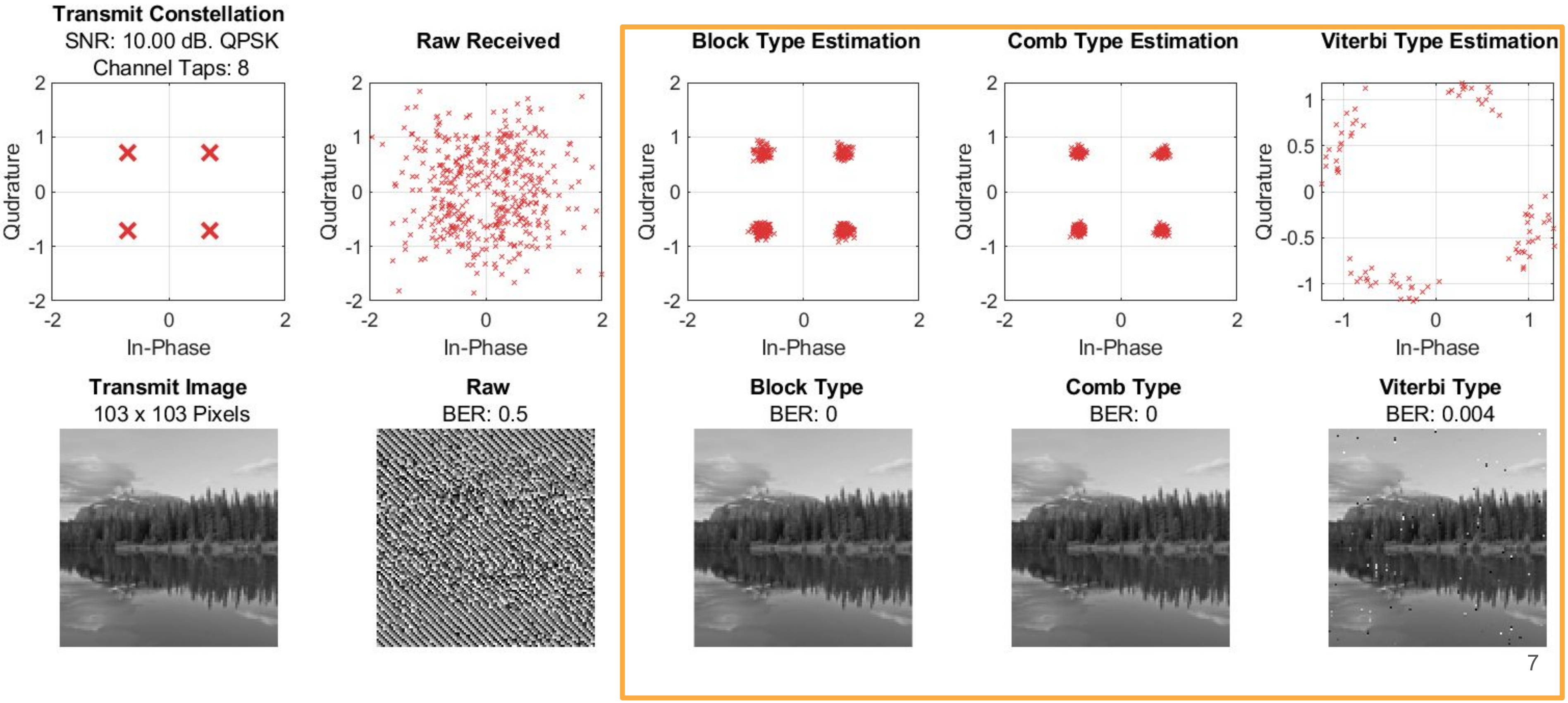
SNR = **1** dB, #taps = 8

```
% Apply fading channel
g = exp(-(0:n_taps-1));
g = g/norm(g);
x_s_noise_fading = conv(x_s_noise,g,'same');
```



Exp Fading + AWGN

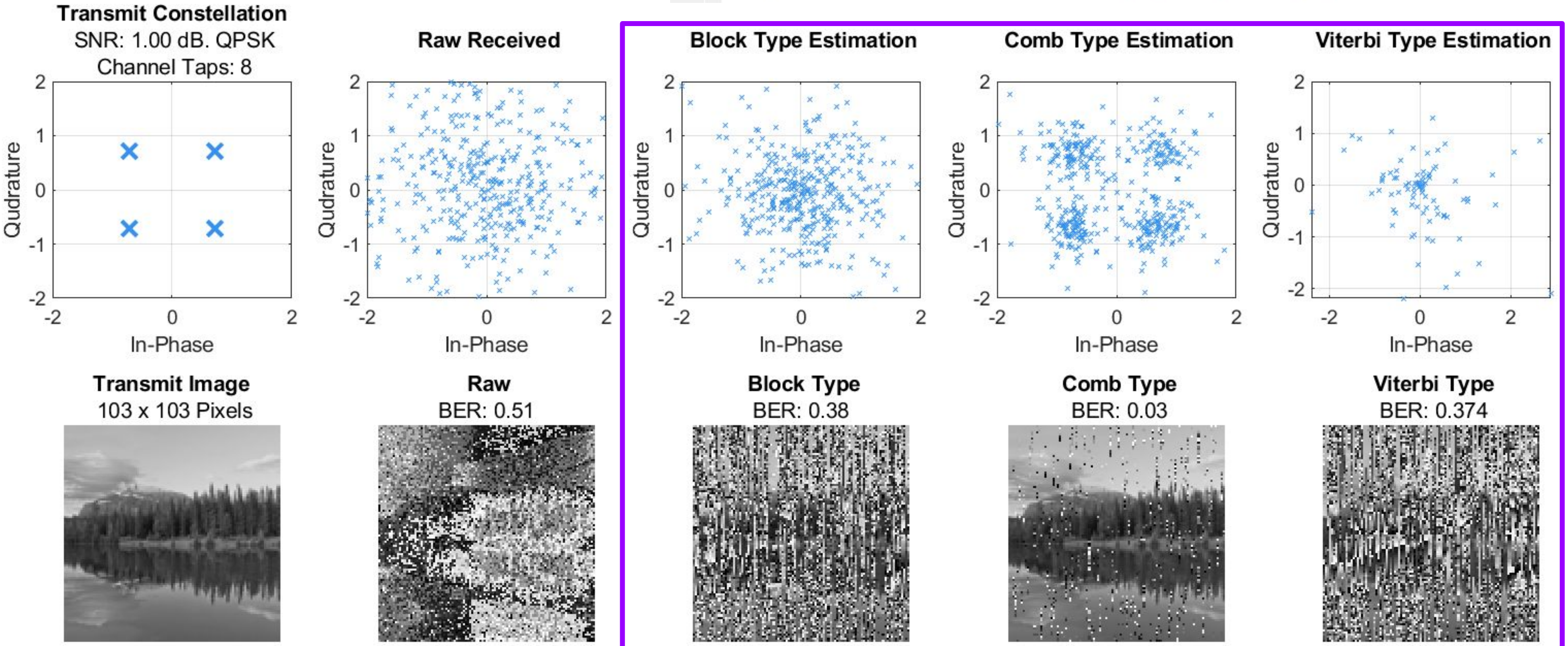
SNR = **10** dB, #taps = 8



Rayleigh Fading + AWGN

SNR = 1 dB, #taps = 8

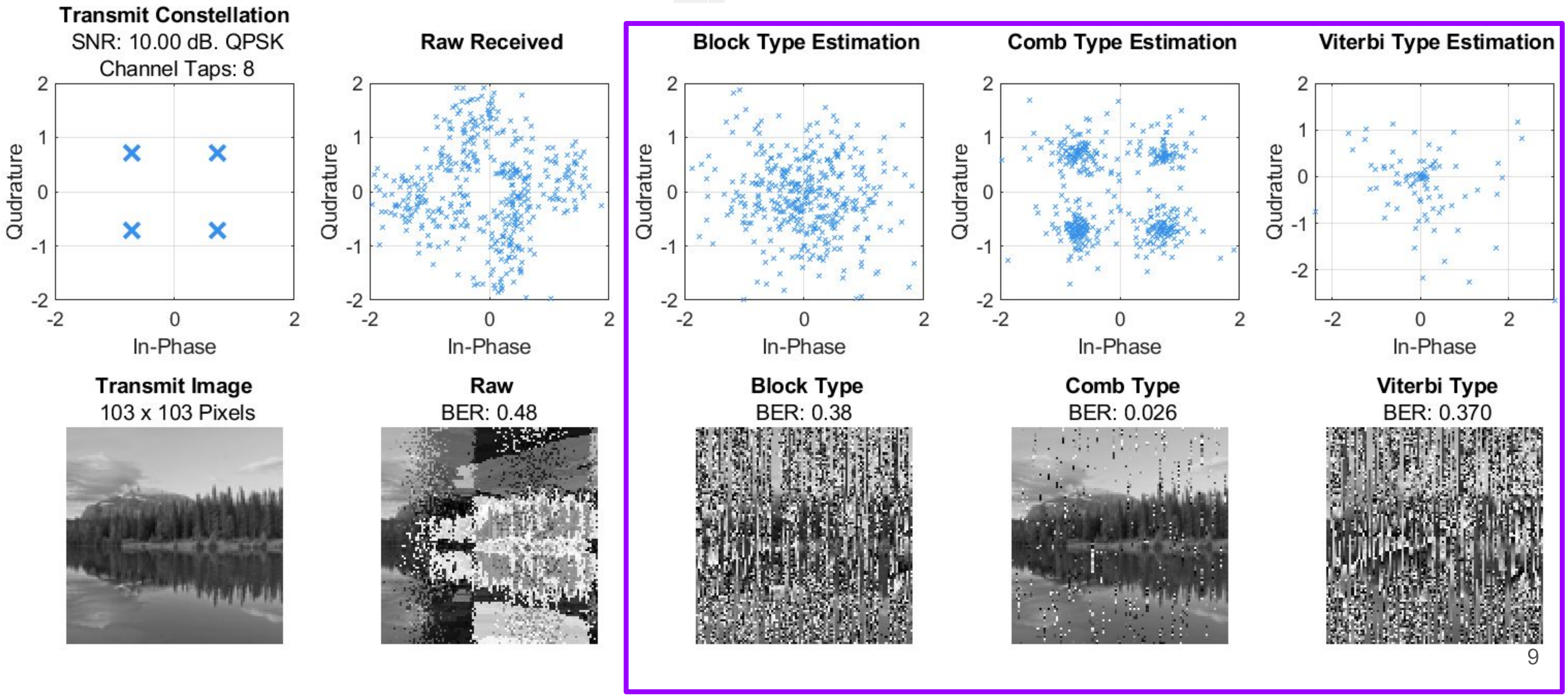
```
% Rayleigh Channel initialization
sampleRate   = 48000;      % Sample rate of 20K Hz
maxDopplerShift  = 1; % Max Doppler shift of diffuse components (Hz)
delayVector = [1 2]*1e-8; % Discrete delays of four-path channel (s)
gainVector  = [-1 -1];  % Average path gains (dB)
```



Rayleigh Fading + AWGN

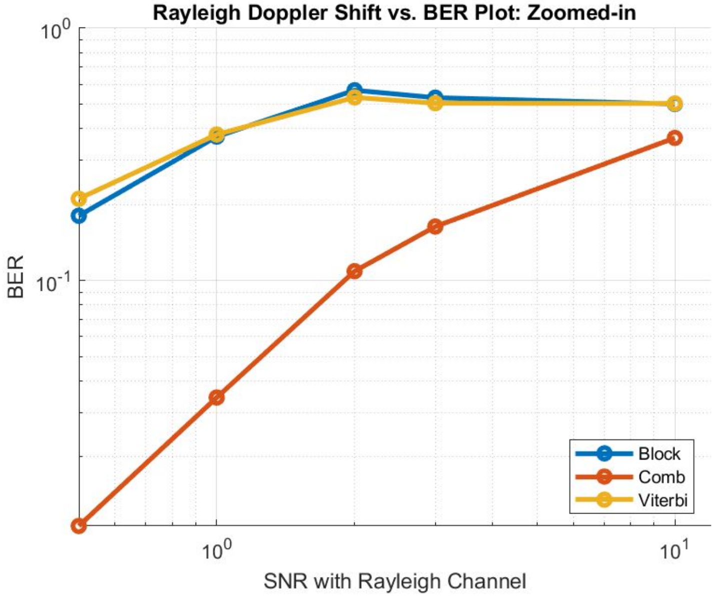
SNR = 10 dB, #taps = 8

```
% Rayleigh Channel initialization
sampleRate   = 48000;      % Sample rate of 20K Hz
maxDopplerShift   = 1; % Max Doppler shift of diffuse components (Hz)
delayVector = [1 2]*1e-8; % Discrete delays of four-path channel (s)
gainVector  = [-1 -1];  % Average path gains (dB)
```



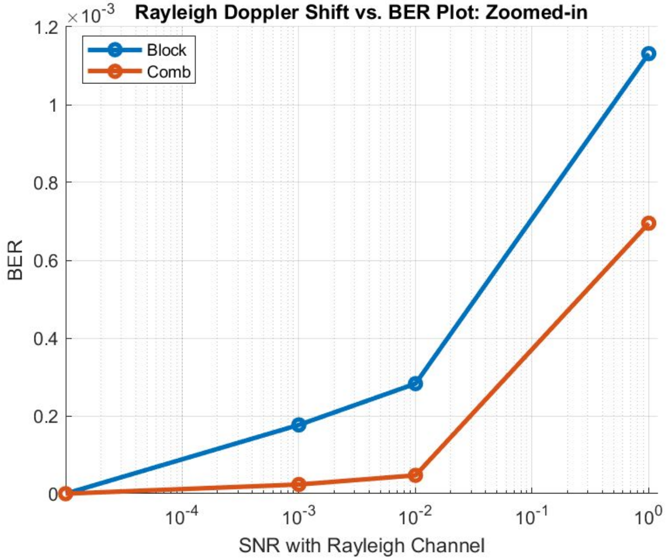
Model Comparisons: Rayleigh Fading Strength vs. BER

Doppler Shift (Hz) vs. BER



Delay(s) vs. BER

Comb Pilot is better than block pilot.

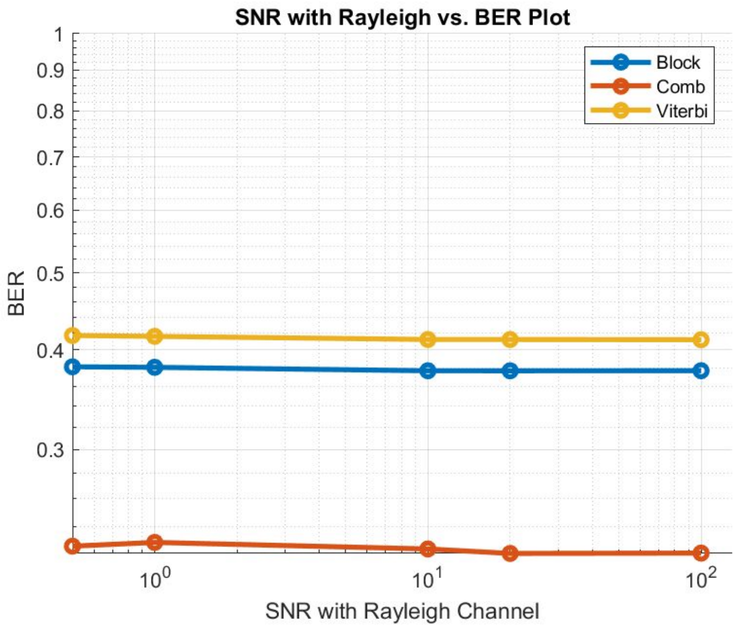


Model Comparisons: With Rayleigh

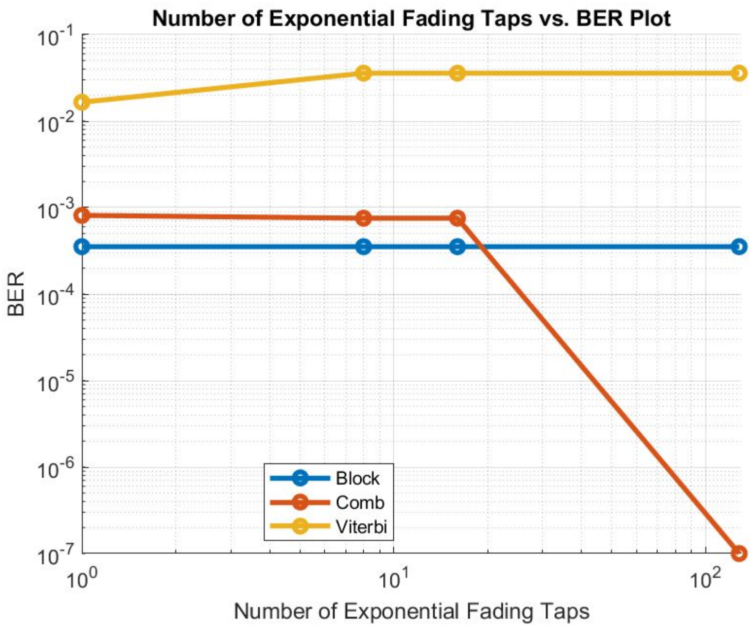
Rayleigh Fading Channel Parameters:

```
sampleRate = 48000;
maxDopplerShift = 1;
delayVector = [0.01 0.02];
gainVector  = [-1 -1];
```

SNR vs. BER



Number of Taps vs. BER



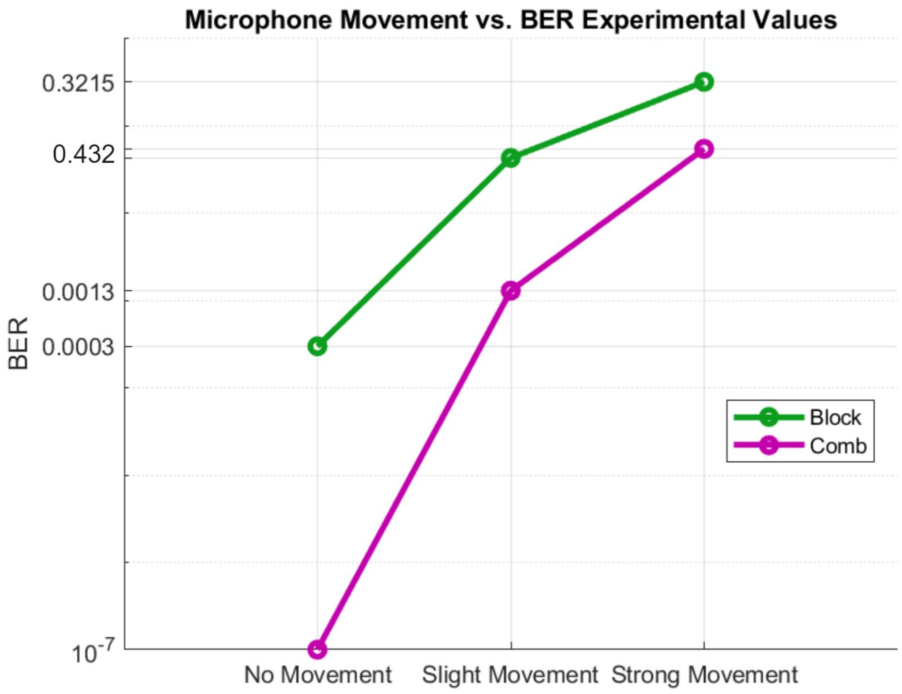
Demonstration: Microphone Transmission

Input: 44 x 51, gray-scaled uint8 image

Audio duration: around 25 seconds

Noise : a repeated piece of music

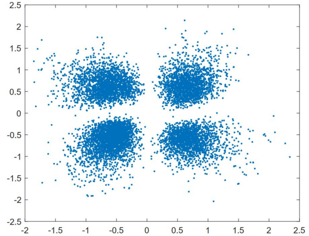
Microphone	BER (Block)	BER (Comb)
No Movement	0.0003	0.0
Slight Movement	0.0432	0.0013
Strong Movement	0.3215	0.0549

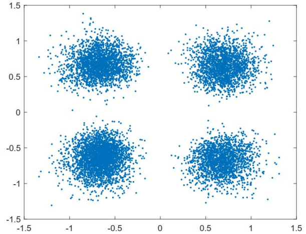


Demonstration: Channel Correction Ability

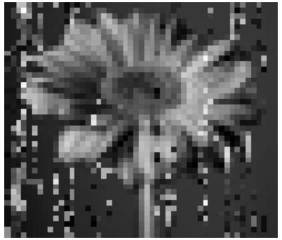
Block Pilot Training | Comb Pilot Training

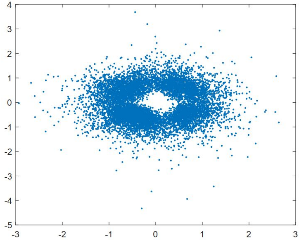
No Movement

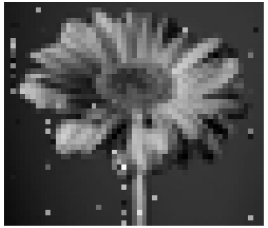


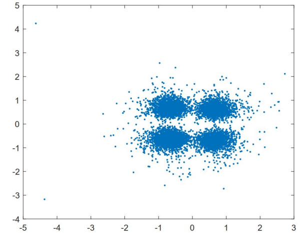


Slight Movement

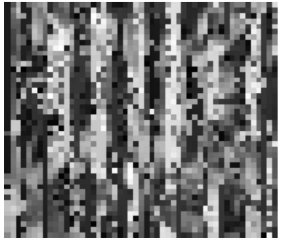


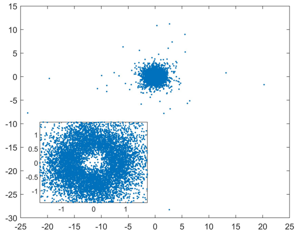


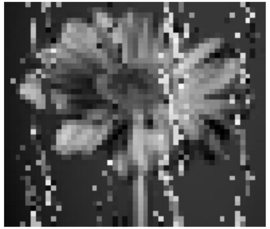


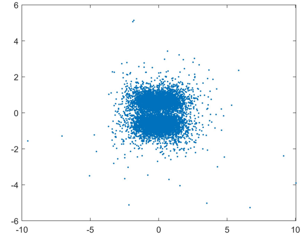


Strong Movement







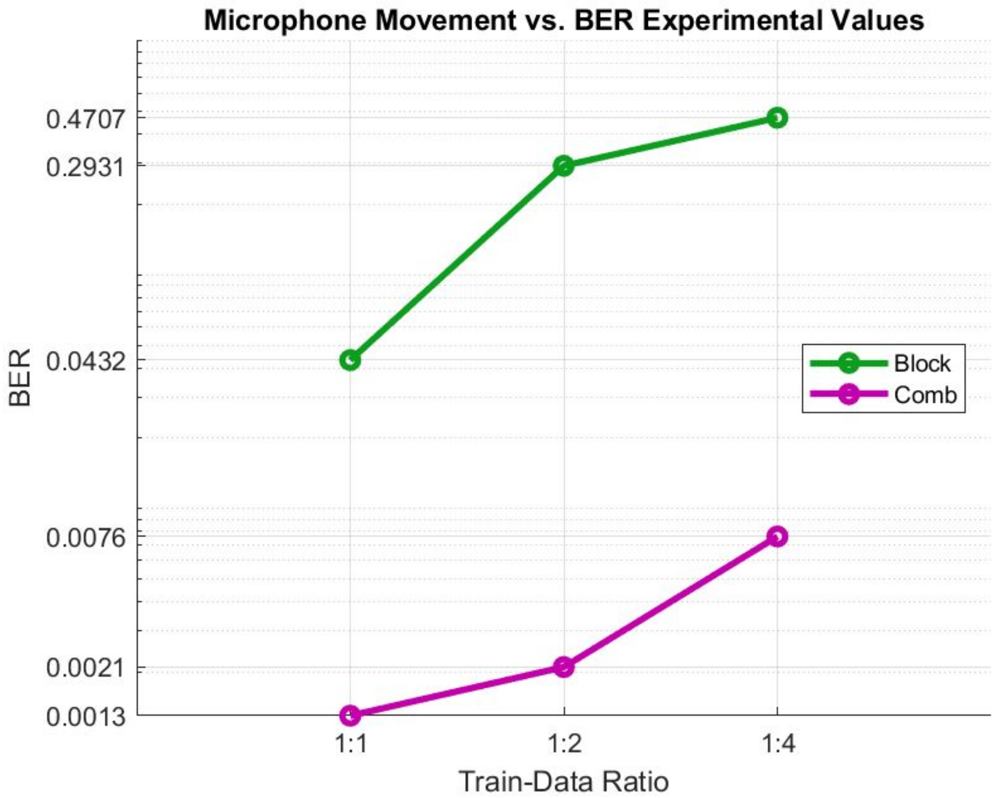


Demonstration: Distribution of Training Symbols

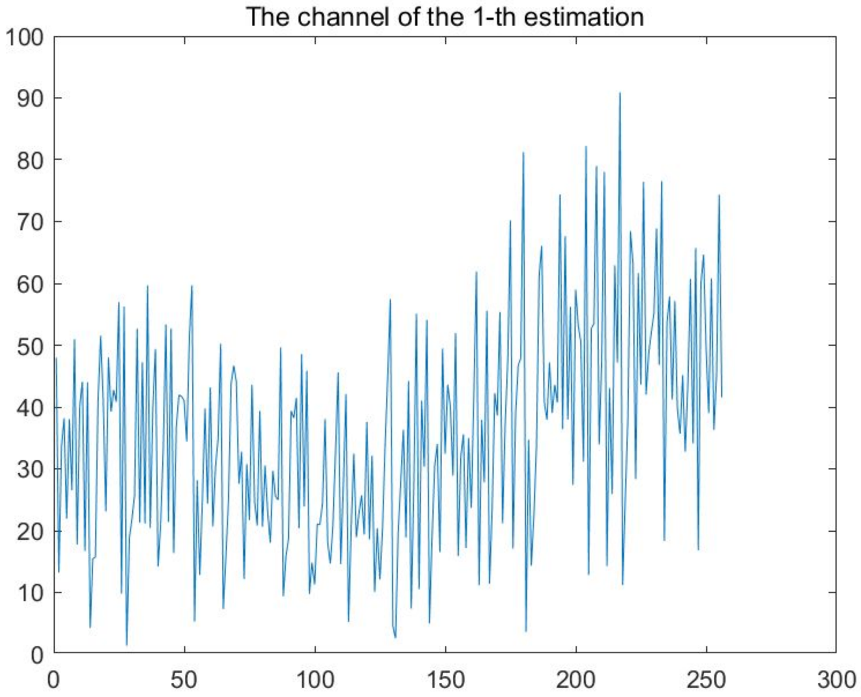
Train-Data Ratio: (number of train sub-carriers : data sub-carriers)

Microphone were slightly moving

Train-Data Ratio	BER (Block)	BER (Comb)
1:1	0.0432	0.0013
1:2	0.2931	0.0021
1:4	0.4707	0.0076

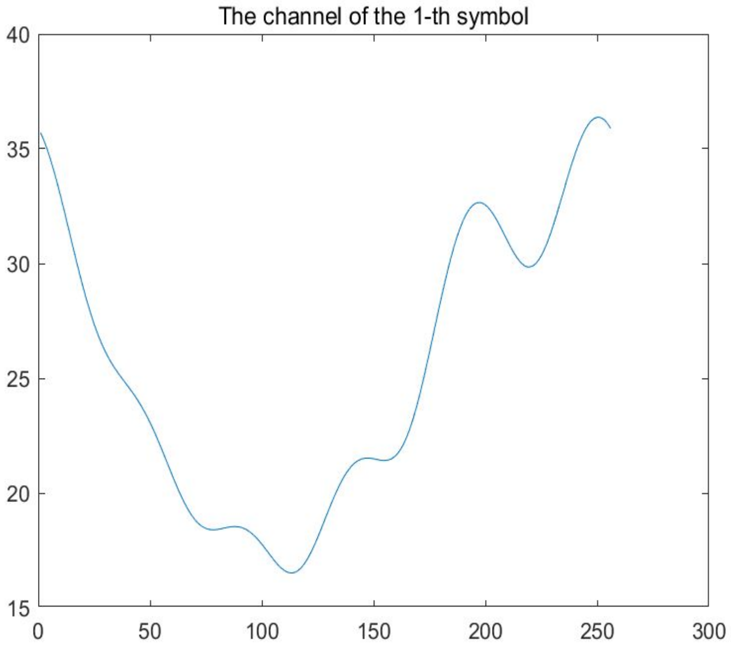


Demonstration: Block Pilot Channel Estimation

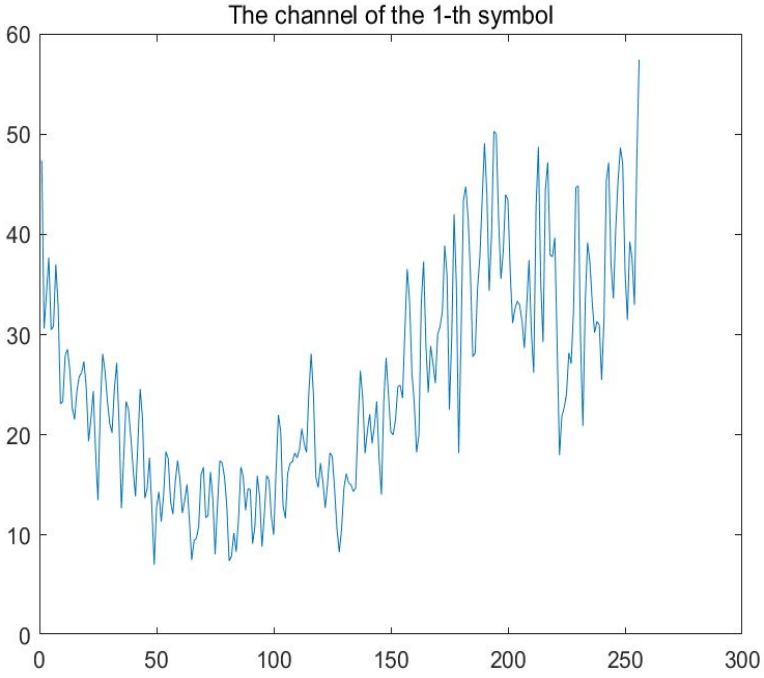


Block Pilot

Demonstration: Comb Pilot Channel Estimation



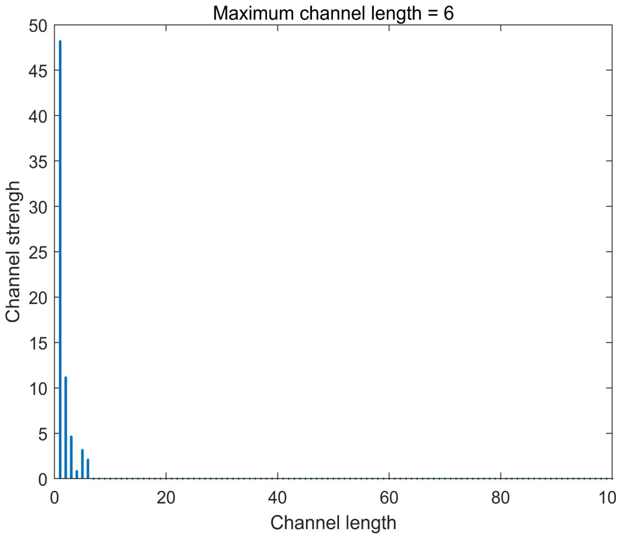
Comb pilot with channel length = 6

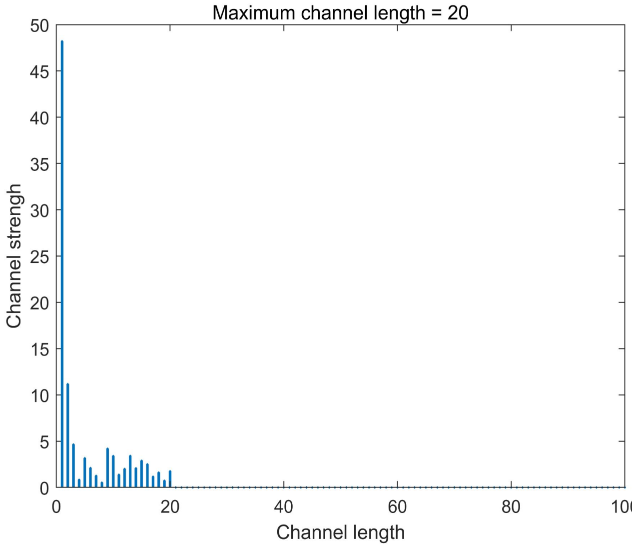


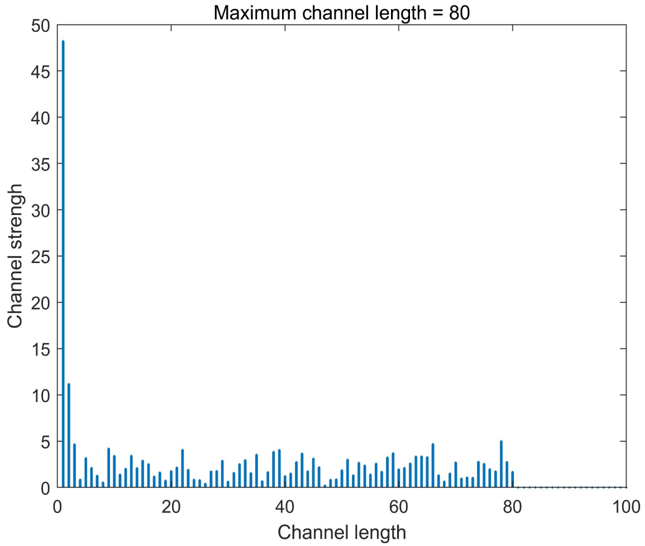
Comb pilot with channel length = 80

Demonstration: Channel length

RMS delay spread: $\sigma_\tau = \sqrt{\frac{\sum(\tau_i - \overline{\tau})^2 |h_i|^2}{\sum |h_i|^2}}$ $\overline{\tau} = \frac{\sum \tau_i |h|_i}{\sum |h_i|}$ $\Delta\tau = \frac{1}{N \cdot T} = 0.00078$ $x_\tau = \sigma_\tau \cdot v_s$







σ_τ = 0.0006 s x_τ = 0.21 m

σ_τ = 0.0032 x_τ = 1.10 m

σ_τ = 0.0238 x_τ = 8.16 m

Problems Encountered & Solutions

AGWN and exponential PDP fading are not sufficient

- Add Rayleigh fading channel to analyze performance

Long computation times for tx() and rx() Solution

- Only use small (50 x 50) gray-scale images to shorten the input data length

Padding and data chunks not dividing evenly

- Rigorous debugging

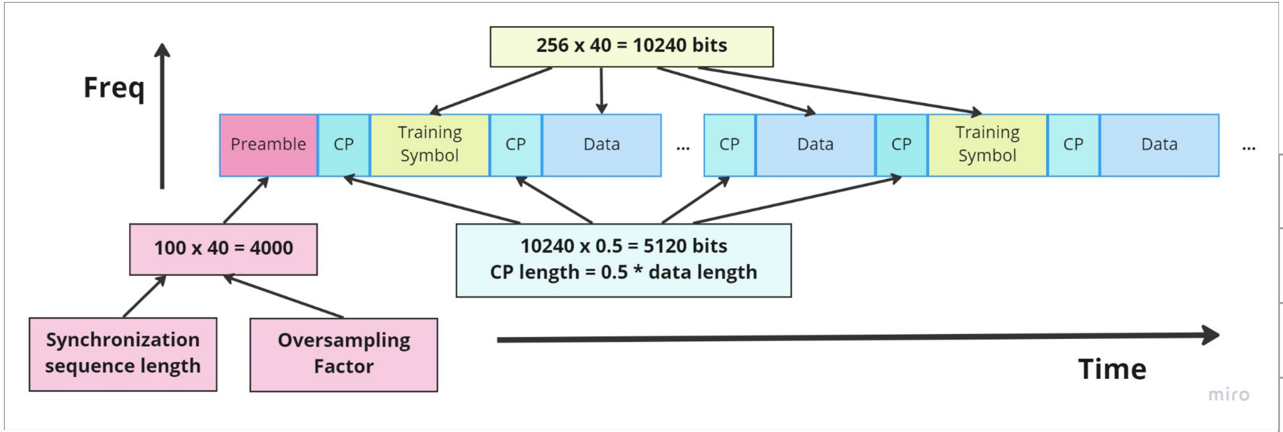
Difficult to control variables in real acoustic systems

- Modulate SNR vs. BER with Rayleigh fading

Problems with Viterbi Viterbi

- More debugging

Optimal Model & Efficiency Calculations



picture bits in total = 103 * 103 * 8 = 84872 bits

$$\frac{(\frac{picture\ bits\ in\ total}{256*2})}{((\frac{picture\ bits\ in\ total}{256*2}*\frac{1}{3})+\frac{picture\ bits\ in\ total}{256*2})*1.5+4000}$$

$$=\frac{165}{(165*\frac{1}{3}+165)*1.5+100}$$

$$=0.384=38.4\%$$

Default Parameters for audiotrans.m

Sampling Rate	48000
Symbol Rate	100
Modulation Order	QPSK
Carrier Frequency	8000
# Sub-carriers	**256**
Sample Spacing	**4.6875**
Oversampling Factor	40
Cyclic Prefix Length	5120
Comb Training Insertion Interval	2