

**语音信号处理**

**实验报告**

**实验名称： 语音信号的频域分析**

**学生姓名： 赵展文 学号： 2024360809**

**教 师： 马 英 职称： 教授**

**院 系： 智能科学与工程学院**

**专业班级： 2024级集成电路工程**

**一、语音信号的短时频谱分析**

程序代码

%% 清除缓存

clear;

close all;

%% 读取音频文件

[file, path] = uigetfile('\*.wav', '选择WAV文件');

filename = fullfile(path, file);

[y, fs] = audioread(filename);

y = y(:, 1); % 取单通道音频

%% 设定帧参数

frame\_length = 512; % 帧长

frame\_shift = 128; % 帧移

%% 分帧处理

fn = fix((length(y) - frame\_length) / frame\_shift) + 1; % 计算帧数

frame\_time = (((1:fn)-1)\*frame\_shift + frame\_length/2)/fs; % 每帧中心对应的时间（秒）

frames = zeros(frame\_length, fn); % 预分配存储空间

for i = 1:fn

start\_idx = (i - 1) \* frame\_shift + 1; % 计算起始索引

end\_idx = start\_idx + frame\_length - 1; % 计算结束索引

frames(:, i) = y(start\_idx:end\_idx); % 逐帧提取

end

%% 计算 Hanning 窗

hanning\_win = hanning(frame\_length); % 计算 Hanning 窗

for i = 1:fn

frames(:, i) = frames(:, i) .\* hanning\_win; % 每帧加窗

end

%% 计算短时幅度谱

fft\_frames = fft(frames, frame\_length, 2);% 对每一帧进行 FFT

mag\_S = abs(fft\_frames).^2 + eps;

mag\_mean = mean(mag\_S, 1);

mag\_db = 20\*log10(mag\_mean(1:frame\_length/2 + 1));

freq = (0:frame\_length/2)\*fs/frame\_length;

%% 绘制波形

figure;

% 原始语音

subplot(2,1,1);

plot(y);

title('原始语音信号');

xlabel('时间/s');

ylabel('幅值');

grid on;

% 绘制幅度谱

subplot(2,1,2);

plot(freq, mag\_db, 'b-');

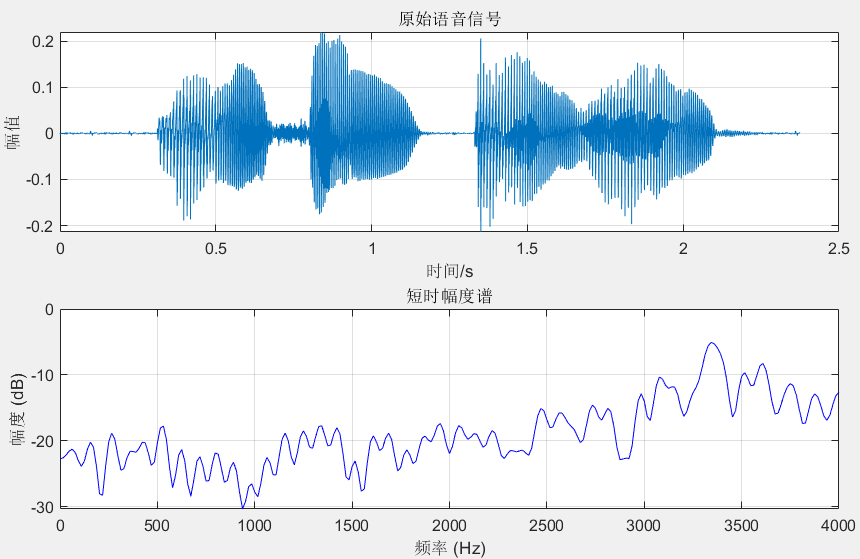
xlabel('频率 (Hz)');

ylabel('幅度 (dB)');

title('短时幅度谱');

grid on;

波形图



**二、语音信号的短时功率谱分析**

程序代码

%% 清除缓存

clear;

close all;

%% 读取音频文件

[file, path] = uigetfile('\*.wav', '选择WAV文件');

filename = fullfile(path, file);

[y, fs] = audioread(filename);

y = y(:, 1); % 取单通道音频

%% 设定帧参数

frame\_length = 512; % 帧长

frame\_shift = 128; % 帧移

%% 分帧处理

fn = fix((length(y) - frame\_length) / frame\_shift) + 1; % 计算帧数

frame\_time = (((1:fn)-1)\*frame\_shift + frame\_length/2)/fs; % 每帧中心对应的时间（秒）

frames = zeros(frame\_length, fn); % 预分配存储空间

for i = 1:fn

start\_idx = (i - 1) \* frame\_shift + 1; % 计算起始索引

end\_idx = start\_idx + frame\_length - 1; % 计算结束索引

frames(:, i) = y(start\_idx:end\_idx); % 逐帧提取

end

%% 计算 Hanning 窗

hanning\_win = hanning(frame\_length); % 计算 Hanning 窗

for i = 1:fn

frames(:, i) = frames(:, i) .\* hanning\_win; % 每帧加窗

end

%% 计算短时功率谱

[S, F, T] = spectrogram(y, frame\_length, frame\_shift, frame\_length, fs); % 短时傅里叶变换

P = abs(S).^2;

P\_mean = mean(P, 1);

%% 绘制波形

figure;

% 原始语音

subplot(3,1,1);

plot(y);

title('原始语音信号');

xlabel('时间/s');

ylabel('幅值');

grid on;

% 绘制短时功率谱

subplot(3, 1, 2);

plot(T, P\_mean, 'b-');

xlabel('时间/s');

ylabel('功率 (dB)');

title('短时功率谱');

grid on;

波形图

