PHÂN ĐOẠN TÍN HIỆU TIẾNG NÓI VÀ KHOẢNG LẶNG DÙNG CÁC PHƯƠNG PHÁP STE&ZCR, MA VÀ HAI KỸ THUẬT CHUẨN HÓA NĂNG LƯỢNG

Nguyễn Huy Cổn, Trần Trọng Bảo, Bùi Phan Minh Hưng, Phạm Ngọc Hiếu

Nhóm 3, lớp HP: 19.99

Điểm	Bảng phân công nhiệm vụ		Chữ ký của SV
	Nguyễn Huy Cổn	Đọc tài liệu, cài đặt, và viết báo cáo về thuật toán chuẩn hóa năng lượng 1[5] do giảng viên hướng dẫn, viết báo cáo về thuật toán chuẩn hóa năng lượng 1, phần kết quả thực nghiêm, cùng Bảo lập bảng biểu thống kê về kết quả thu được	
	Trần Trọng Bảo	Đọc tài liệu, cài đặt, và viết báo cáo về thuật toán chuẩn hóa năng lượng 2[5] do giảng viên hướng dẫn, viết báo cáo về thuật toán chuẩn hóa năng lượng 2, phần kết quả thực nghiêm, cùng Cổn lập bảng biểu thống kê về kết quả thu được	
	Bùi Phan Minh Hưng	Đọc tài liệu, cài đặt, và viết báo cáo về thuật toán Mean Average trong tài liệu tham khảo [1](tr. 32-40), viết báo cáo về thuật toán MA, đặt vấn đề, phần kết quả thực nghiêm, kết luận, và làm slide PowerPoint	
	Phạm Ngọc Hiếu (nhóm trưởng)	Đọc tài liệu, cài đặt, và viết báo cáo về thuật toán kết hợp giữa Short-Time Energy và Zero-crossing Rate trong tài liệu tham khảo [2](tr. 117-135), viết báo cáo về thuật toán STE & ZCR, phần kết quả thực nghiêm, so sánh độ hiệu quả của từng thuật toán	

Lời cam đoan: Chúng tôi, gồm các sinh viên có chữ ký ở trên, cam đoan rằng báo cáo này là do chúng tôi tự viết dựa trên các tài liệu tham khảo liệt kê ở cuối báo cáo. Các số liệu thực nghiệm và mã nguồn chương trình nếu không chỉ dẫn nguồn tham khảo đều do chúng tôi tự làm. Nếu vi phạm thì chúng tôi xin chịu trách nhiệm và tuân theo xử lý của giáo viên hướng dẫn.

TÓM TẮT— Bài toán phân đoạn tín hiệu tiếng nói và khoảng lặng là bài toán phổ biến trong việc xử lý tiếng nói. Bài báo cáo này dựa trên việc thực hiện cài đặt thuật toán phân loại tiếng nói và khoảng lặng dựa vào các đặc điểm đặc biệt của tín hiệu âm thanh chính là Short-Time Energy, Zero-crossing Rate và Magnitude Average. Các thử nghiệm đều dựa trên các file ghi âm tiếng nói trong các điều kiện môi trường khác nhau trong phòng thí nghiệm và trong studio. Từ đó đánh giá được các thuật toán là đúng sai và có những sai khác trong những môi trường nhất định. Môi trường càng nhiều tạp âm, càng nhiều nhiễu thì việc phân loại càng trở nên khó khăn và đòi hỏi thuật toán phải tốt ưu hơn.

Từ khóa— Short-Time Energy(STE), Zero-crossing Rate(ZCR), Mean Average(MA), Tiếng nói(Speech), khoảng lặng(Slience), chuẩn hóa năng lượng.

Mục lục

I. ĐẶT VẤN ĐỀ

Tiếng nói là một phương tiện giao tiếp bằng âm thanh của con người với mục đích trao đổi thông tin cũng như tâm tư tình cảm của con người. Chính vì lẽ đó, việc nghiên cứu về xử lý tiếng nói đóng vài trò quan trong cuộc sống của chúng ta. Thông qua nghiên cứu về xử lý tiếng nói, chúng ta có thể cải thiện chất lượng tiếng nói, nhận dạng được nhiều thông tin về nhiều mặt của một người như quốc tịch, giới tính,… thông qua việc xử lý giọng nói của họ, mã hóa tiếng nói để tăng độ bảo mật,… Và bài tập nhóm về phân đoạn tiếng nói/khoảng lặng dung các phương pháp khác nhau là một bài tập bổ ích giúp chúng ta có được cái nhìn sơ khai về việc xử lý tín hiệu tiếng nói.

Phân đoạn tiếng nói/khoảng lặng giống như tên gọi, là phân đoạn những khoảng có tín hiệu tiếng nói và những chỉ không có tín hiệu tiếng nói (nhưng có thể có rè, nhiễu). Dựa trên các đặc trưng của tín hiệu như Năng lượng, Zero-crossing Rate, Magnitude Average,..gần như không thay đổi trong những khoảng thời gian ngắn hạn 10 – 30ms. Do vậy, chúng ta chia tín hiệu thành các khung như nhau có độ dài 10 – 30ms để xử lý, tìm các đặc trưng của mỗi khung sau đó tổng hợp lại thành các hàm đặc trưng biến thiên theo thời gian.

Phụ thuộc vào điều kiện môi trường (yên tĩnh hay có nhiều rè nhiễu) cũng như khả năng của thiết bị ghi âm mà các tiếng nói – khoảng lặng có độ chính xác khác nhau. Môi trường càng nhiễu thì càng khó xử lý đòi hỏi một thuật toán tối ưu hơn. Do vậy, một thuật toán có thể đúng với tín hiệu này nhưng lại sai với tín hiệu khác. Việc cải thiện độ chính xác của thuật toán đúng trong nhiều trường hợp hơn là bài toán nan giải.

Báo cáo được phân ra thành các phần:

- Cơ sở lý thuyết
- Phương pháp chuẩn hóa năng lượng 1
- Phương pháp chuẩn hóa năng lượng 2
- Phương pháp Mean Average
- Phương pháp Short-Time Energy & Zero-Crossing Rate
- Kết quả thực nghiệm và nhận xét
- Kết luận

II. LÝ THUYẾT XỬ LÝ TÍN HIỆU TIẾNG NÓI VÀ CÁC THUẬT TOÁN

A. Xử lý tín hiệu tiếng nói

Tiếng nói của chúng ta được thể hiện ở dạng tín hiệu số khi được đưa vào trong máy tính, nên để xử lý tiếng nói của chúng ta thì ta phải "số hóa" tín hiệu rồi mới xử lý tín hiệu nhận được bằng các phương pháp khác nhau sau khi "số hóa" ở trên máy. Quá trình này chính là quá trình xử lý tín hiệu tiếng nói.

B. Phân đoạn tín hiệu

1. Phân đoạn tín hiệu

Phân đoạn tín hiệu là quá trình chia tín hiệu đang xét thành các khung tín hiệu nhỏ hơn tạo thuận lợi trong việc tính toán, xử lý rồi từ đó nâng cao việc nhận dạng tín hiệu. Ngoài ra ta có thể dung việc phân đoạn để xác định đầu và cuối của tín hiệu tiếng nói. Ở đây, ta phân đoạn tín hiệu đầu vào thành các khung chồng lên nhau.

2. Cách thức

Chia các tín hiệu thành các khung chồng có độ dài như nhau 10-30s. Tính các đặc trưng Zero-Crossing Rate, Short-Time Energy, Magnitude Average của từng khung sau đó tổng hợp các khung lại thành các hàm đặc trưng biến thiên theo thời gian.

3. Thực hiện

Chia tín hiệu thành một ma trận với kích thước 2*N-1 x f_size với N là số frame được chia theo chiều dài của tín hiệu và f_size là chiều dài (theo mẫu) của mỗi frame được chia (10-30ms).

C. Zero-Crossing Rate

Zero-crossing Rate(tốc độ băng qua 0) là tốc độ vượt qua điểm 0 của tín hiệu(từ dương sang 0 sang âm hoặc từ âm sang 0 sang dương)[3].

Công thức toán học :

$$Z_n = \sum_{m=-\infty}^{+\infty} \left|\operatorname{sgn}(x[m]) - \operatorname{sgn}(x[m-1])\right| \times w[n-m] \text{ [2]}$$

Trong đó:

$$sgn(x[n]) = \begin{cases} 1, & x[n] \geq 0 \\ -1, & x[n] < 0 \end{cases}$$

$$w[n] = \begin{cases} \frac{1}{2N}, & 0 \leq x \leq N-1 \\ x, & còn\ lại \end{cases}$$

$$N: \text{độ } dài\ khung\ tín\ hiệu(mẫu)$$

Các âm hữu thanh là kết quả của sự rung động tuần hoàn của dây thanh và thường cho thấy tốc độ băng qua 0 thấp, trong khi các âm vô thanh lại không có sự rung động của dây thanh và thường cho thấy tốc độ băng qua 0 cao.[4]

D. Short-Time Energy

Short-Time Enegy(năng lượng ngắn hạn) là năng lượng của tín hiệu khi xét trong một khoảng thời gian ngắn.

Công thức toán học:

$$E_n = \sum_{m=-\infty}^{+\infty} (x[m]w[n-m])^2 \text{ [2]}$$

Trong đó:

$$w[n] = \begin{cases} \frac{1}{2N}, & nếu\ 0 \leq n \leq N-1 \\ 0, & còn\ lại \end{cases}$$

Các âm hữu thanh thường có độ lớn biên độ lớn dẫn tới năng lượng lớn. Ngược lại, các âm vô thanh thường có độ lớn biên độ khá nhỏ nên năng lượng cũng khá nhỏ so với năng lượng của âm hữu thanh.

E. Mean Average

Magnitude Average là cường độ trung bình của tín hiệu trong khoảng thời gian ngắn [1] và được tính theo công thức :

$$MA[n] = \sum_{m=-\infty}^{+\infty} |x[n-m]|\, w[m], [1]$$

Trong khi :

$$w[n] = \begin{cases} 1, n \in [0, \quad N] \\ 0, n \notin [0, \quad N] \end{cases}$$

Các âm hữu thanh có biên độ lớn, dẫn đến cường độ tin hiệu trung bình cũng rất lớn và ngược lại, các âm vô thanh có biên độ tương đối nhỏ, đồng nghĩa với việc có cường độ trung bình của tín hiệu thấp hơn nhiều so với các âm hữu thanh. Từ đó có thể dùng để tìm ra ngưỡng trung bình để phân đoạn tiếng nói và khoảng lặng[1].

Cách xây dựng thuật toán Mean Average dựa trên việc nếu giá trị Magnitude Average của tín hiệu tiếng nói cao hơn so với tín hiệu khoảng lặng thì sẽ có một ngưỡng phân đoạn T để $MA_{speech} \geq T$ và $MA_{silence} < T$. Việc tìm kiếm ngưỡng phân đoạn này có thể được thực hiện bằng việc sử dụng thuật toán được đề cập trong tài liệu [1] :

- Từ khung chồng của tín hiệu ta phân ra thành f chứa các khung tín hiệu có thứ tự lẻ được biết là khung tín hiệu tiếng nói, và g chứa các khung tín hiệu có thứ tự chẵn được biết là khung tín hiệu khoảng lặng.
- Đặt T_{min}, T_{max} là giá trị cực đại và cực tiểu của khung chồng tín hiệu.
- Đặt ngưỡng hiên tại là $T = \frac{1}{2}(T_{min} + T_{max})$.
- Đặt i và p là lượng khung tín hiệu f và g nhỏ hơn và lớn hơn T, $i = \sum_n f[n] < T, p = \sum_n g[n] > T$.
- Đặt j = q = -1.
- Lặp lại các thao tác tiếp theo nếu i khác j và p khác q.
- Tín hệ thức $\frac{1}{i}\sum_1^i \max(f[n] - T, 0) - \frac{1}{p}\sum_1^p \max(T - g[n], 0)$.
- Nếu kết quả là số dương thì đặt $T_{min} = T$, ngược lại nếu kết quả âm đặt $T_{max} = T$.
- Đặt lại ngưỡng hiên tại là $T = \frac{1}{2}(T_{min} + T_{max})$.
- Đặt j = i, và q = p.
- Đặt lại i và p, $i = \sum_n f[n] < T, p = \sum_n g[n] > T$.

Qua các bước trên ta có thể đặt ra ngưỡng tương đối để phân đoạn tín hiệu tiếng nói và khoảng lặng.

F. Chuẩn hóa năng lượng

Do dải động của $\log(E_k)$ biến thiên khá rộng với mỗi file tín hiệu,để tiến hành quan sát, tính toán, phân đoạn cần chuẩn hóa $\log(E_k)$ của các tín hiệu bằng 1 trong 2 cách.

1. Cách 1 :

Chuẩn hóa $\log(E_k)$ về dải (0,1) để dễ tiến hành quan sát, tính toán:

$$X_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)} [5]$$

Với X_{norm} : là giá trị sau khi chuẩn hóa.

x : là giá trị $\log(E_k)$ trước khi chuẩn hóa.

x_{min}, x_{max}: là giá trị min, max của dải x trước khi chuẩn hóa.

2. Cách 2 :

a) Giá trị trung bình :

Là trung bình cộng các giá trị của tín hiệu, như vậy nó biểu diễn giá trị mà người ta "mong đợi".

$$\mu = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$$

Với μ : giá trị trung bình của tín hiệu x[n];

N: độ dài của tín hiệu x[n].

b) Phương sai :

Là 1 đại lượng dùng để đo sự phân tán thống kê của tín hiệu, nó hàm ý cách giá trị của tín hiệu cách giá trị trung bình bao xa.

$$\sigma^2 = \frac{1}{N-1}\sum_{i=0}^{N-1} (x[n] - \mu)^2$$

Với σ^2 :giá trị phương sai của tín hiệu x[n];

μ : giá trị trung bình của tín hiệu x[n];

N: độ dài của tín hiệu x[n].

c) Hàm chuẩn hóa năng lượng bằng phương pháp dựa vào giá trị trung bình và độ lệch chuẩn

Chuẩn hóa log(E_k) về dải (0,1) để dễ tiến hành quan sát, tính toán:

$$X_{norm} = \frac{x - \bar{x}}{\sigma x} [5]$$

Với X_{norm} : là giá trị sau khi chuẩn hóa.

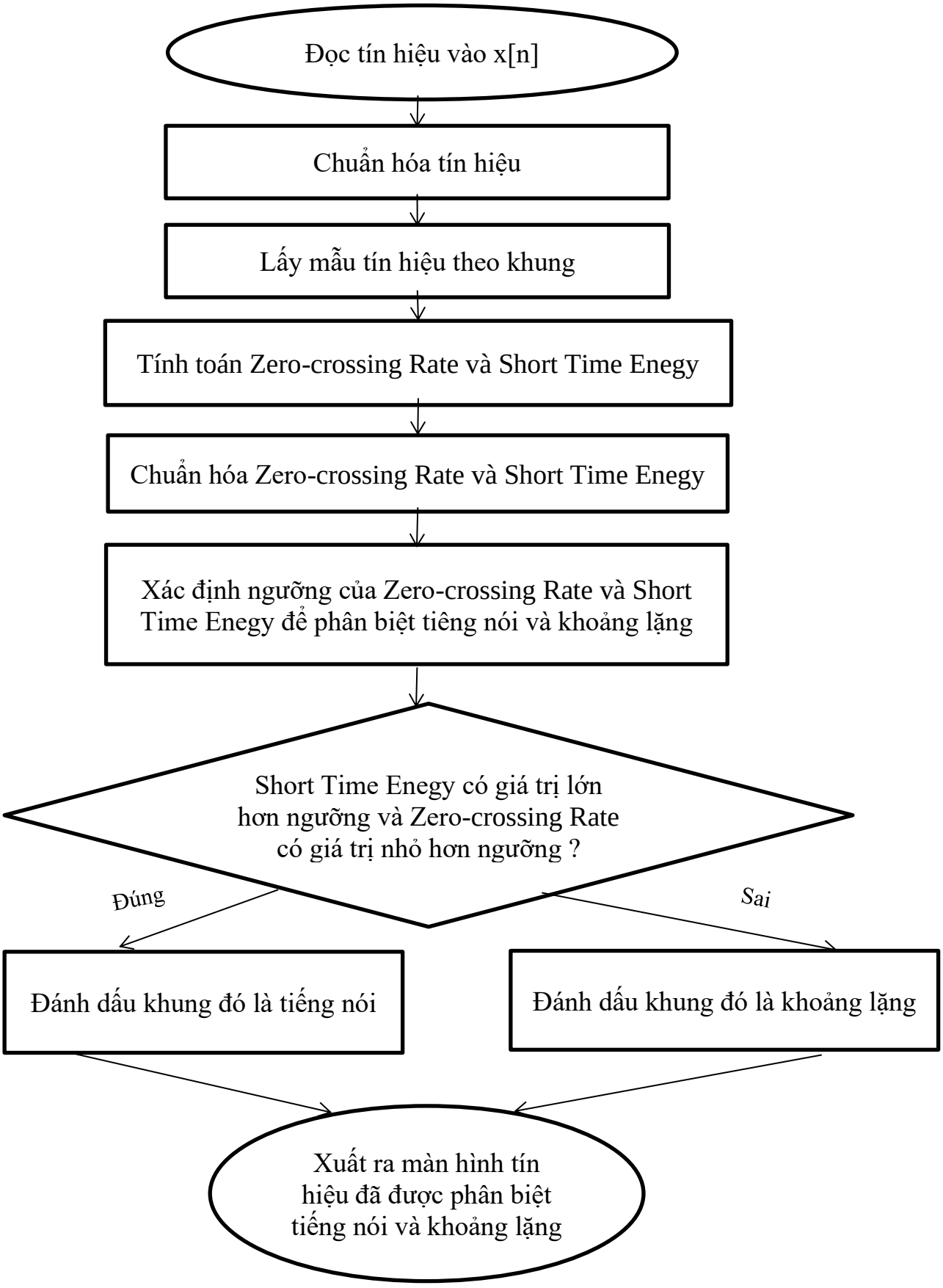
x : là giá trị log(E_k) trước khi chuẩn hóa.

$\bar{x}$: là giá trị trung bình của tín hiệu x trước khi chuẩn hóa.

σx: $là\ phương\ sai\ của\ tín\ hiệu\ x\ trước\ khi\ chuẩn\ hóa.$

G. Phân đoạn tín hiệu giọng nói/khoảng lặng bằng phương pháp kết hợp Short-Time Energy & Zero-Crossing Rate

1. Sơ đồ khối :


Đọc tín hiệu vào x[n]
Chuẩn hóa tín hiệu
Lấy mẫu tín hiệu theo khung
Tính toán Zero-crossing Rate và Short Time Enegy
Chuẩn hóa Zero-crossing Rate và Short Time Enegy
Xác định ngưỡng của Zero-crossing Rate và Short Time Enegy để phân biệt tiêng nói và khoảng lặng
Short Time Enegy có giá trị lớn hơn ngưỡng và Zero-crossing Rate có giá trị nhỏ hơn ngưỡng ?
Đúng
Sai
Đánh dấu khung đó là tiếng nói
Đánh dấu khung đó là khoảng lặng
Xuất ra màn hình tín hiệu đã được phân biệt tiếng nói và khoảng lặng

2. Các tham số ảnh hưởng đến thuật toán:

Độ dài của khung tín hiệu: Nếu khung tín hiệu quá dài thì các đặc trưng không thể xem như không đổi, nếu khung quá ngắn thì sẽ không thể hiện được rõ các đặc trưng.

Tần số lấy mẫu Fs: nếu Fs lớn, các đặc trưng của tín hiệu sẽ rõ ràng và chính xác hơn so với Fs nhỏ.

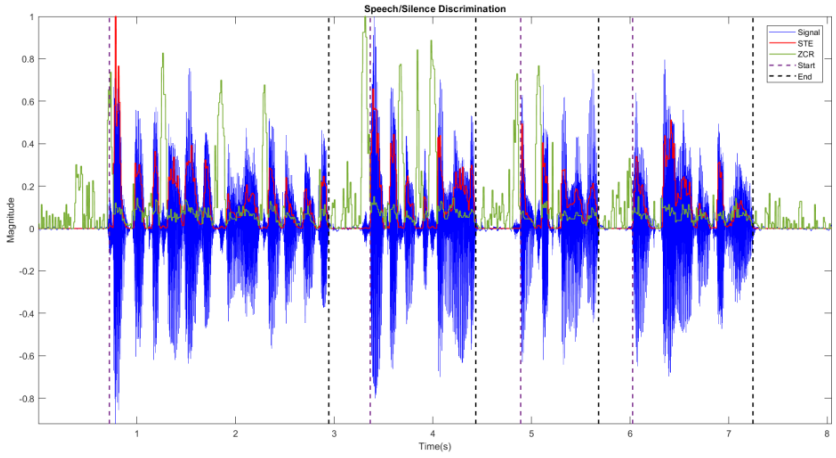
3. Các vấn đề phát sinh và cách giải quyết:

Vấn đề:

Muốn vẽ được tín hiệu, tốc độ băng qua 0, năng lượng lên cùng một hình thì độ dài các vector phải bằng nhau, trong khi đó, các vector năng lượng và tốc độ băng qua 0 lại có độ dài khác với vector tín hiệu(vì đó là vector lưu các giá trị tương ứng của một khung tín hiệu, không phải tại một điểm nhất định).

Cách giải quyết:

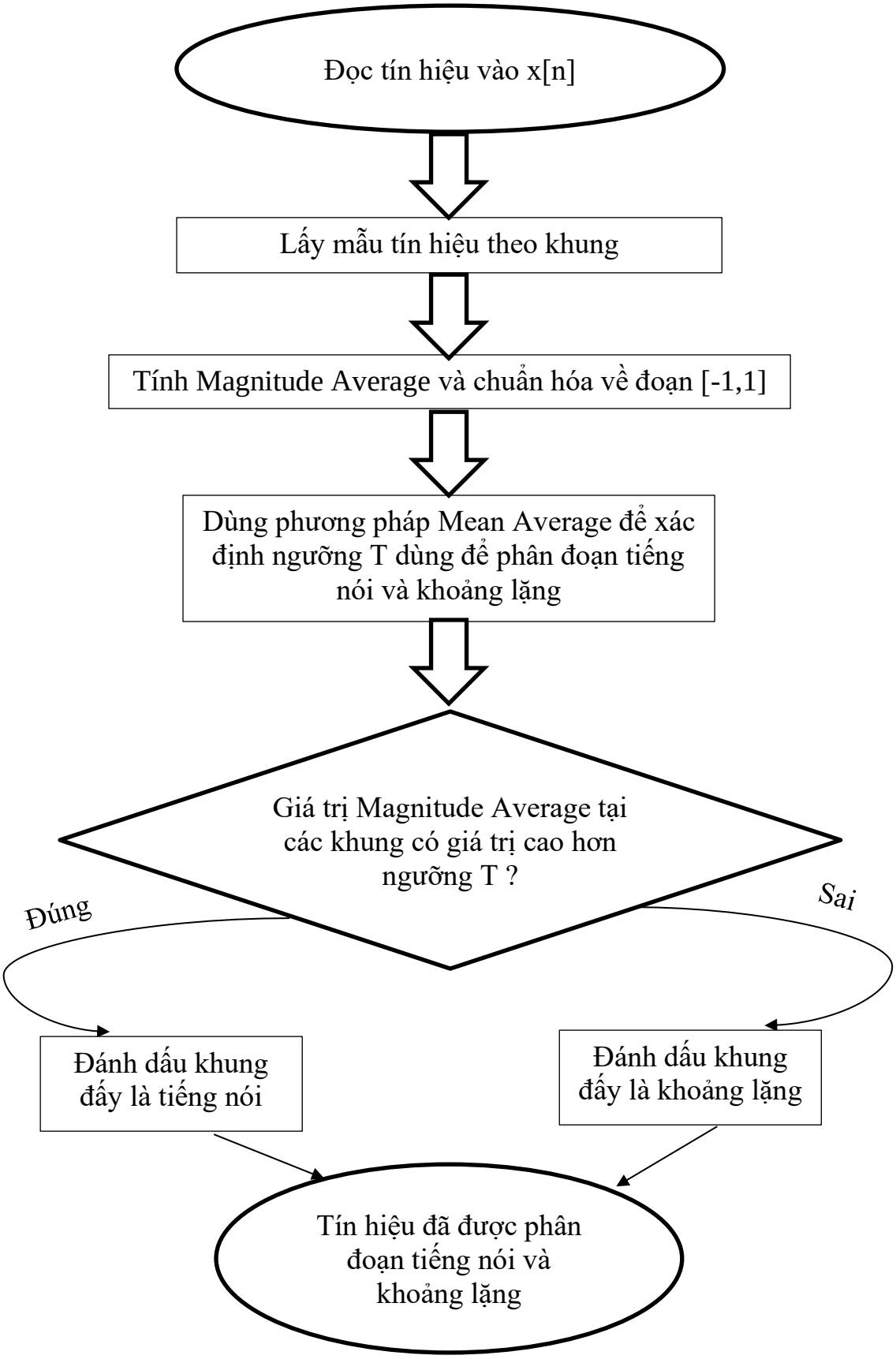
Đối với năng lượng và tốc độ băng qua 0, tạo một vector mới có độ dài tương ứng với độ dài của vector tín hiệu, với các giá trị thuộc một khung thì gán cho năng lượng bằng năng lượng của cả khung đó.Hình 1 minh họa việc vẽ cả ba yếu tố lên trên một đồ thị.



Hình 1. Minh họa về việc vẽ cả ba yếu tố trên cùng một đồ thị

H. Phân đoạn tín hiệu giọng nói/khoảng lặng bằng phương pháp Mean Average

1. Sơ đồ khối


Đọc tín hiệu vào x[n]
Lấy mẫu tín hiệu theo khung
Tính Magnitude Average và chuẩn hóa về đoạn [-1,1]
Dùng phương pháp Mean Average để xác định ngưỡng T dùng để phân đoạn tiếng nói và khoảng lặng
Giá trị Magnitude Average tại các khung có giá trị cao hơn ngưỡng T ?
Đúng
Sai
Đánh dấu khung đấy là tiếng nói
Đánh dấu khung đấy là khoảng lặng
Tín hiệu đã được phân đoạn tiếng nói và khoảng lặng

2. Các tham số ảnh hưởng đến thuật toán :

Độ dài khung tín hiệu : Nếu khung tín hiệu quá dài thì các đặc trưng của tín hiệu sẽ không được biểu thị rõ rang và ngược lại nếu khung quá ngắn thì sẽ không thể hiện rõ được các đặc trưng.

Tần số lấy mẫu ban đầu của tín hiệu : Tần số lấy mẫu càng lớn thì các đặc trưng của tín hiệu sẽ càng được biểu thị rõ rang hơn.

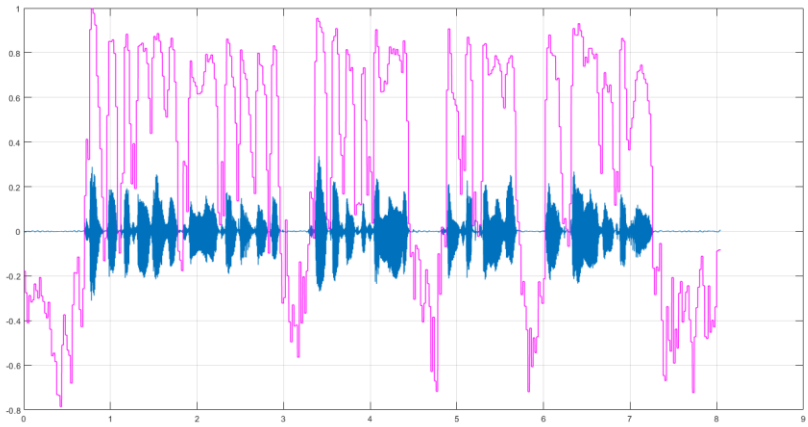
Khoảng lặng tối thiểu : Khoảng lặng có độ dài tối thiểu là 200ms, nếu < 200ms thì nhận định đó là tiếng nói.

3. Vấn đề phát sinh :

Muốn vẽ được tín hiệu và cường độ trung bình lên trên 1 đồ thị thì độ dài các vector phải bằng nhau nhưng vector của cường độ trung bình có độ dài nhỏ hơn nhiều so với chiều dài tín hiệu(vì các vector MA lưu các giá trị của mỗi khung tín hiệu).

4. Hướng giải quyết :

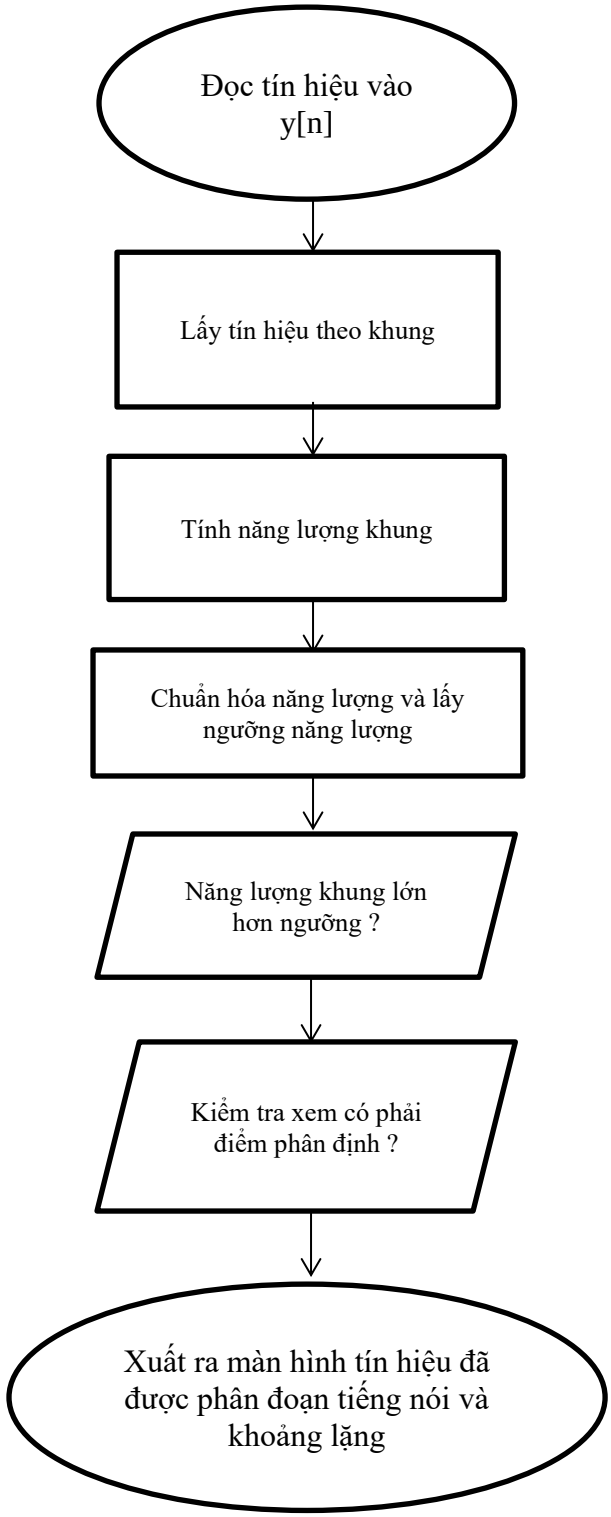
Đối với cường độ tín, tạo một vector mới có độ dài tương ứng với độ dài của vector tín hiệu, với các giá trị thuộc một khung thì gán cho cường độ bằng cường độ của cả khung đó. Hình 2 minh họa việc vẽ cả hai yếu tố lên trên một đồ thị.



Hình 2. Minh họa khi vẽ tín hiệu và cường độ trung bình trên cùng một đồ thị

I. Phân đoạn tính hiệu tiếng nói/khoảng lặng bằng 2 phương pháp chuẩn hóa năng lượng

1. Sơ đồ khối :


Đọc tín hiệu vào
y[n]
Lấy tín hiệu theo khung
Tính năng lượng khung
Chuẩn hóa năng lượng và lấy
ngưỡng năng lượng
Năng lượng khung lớn
hơn ngưỡng ?
Kiểm tra xem có phải
điểm phân định ?
Xuất ra màn hình tín hiệu đã
được phân đoạn tiếng nói và
khoảng lặng

2. Vấn đề phát sinh và hướng giải quyết :

Vì độ dài của vector năng lượng khung khác so với độ dài tín hiệu, nên sau khi tính toán vị trí phân đoạn trên vector năng lượng khung,ta nhân với tỉ lệ độ dài số mẫu tương ứng với một khung năng lượng. Để tính toán ra vị trí phân đoạn trên tín hiệu.

III. MÃ CHƯƠNG TRÌNH CÀI ĐẶT CÁC THUẬT TOÁN

A. Hai phương pháp chuẩn hóa năng lượng

1. Main

```
% [y,fs] = audioread('studio_female.wav');i= 1;%doc tin hieu dau vao
% [y,fs] = audioread('studio_male.wav');i = 2;
% [y,fs] = audioread('lab_femaleth1.wav');i = 3;
[y,fs] = audioread('lab_female.wav');i = 4;
y = y';
f_d = 0.01;%do dai cua 1 khung
frames = framing(y,fs,f_d);%chia khung tin hieu
ch = energy_log(frames);%tinh log nang luong cac khung
ch = chuanhoa1(ch); %chuan hoa
% ch = chuanhoa2(ch) % chuan hoa 2
figure(i);
val = 0.62;%gia tri nguong doi voi chuan hoa 0,1
%val = (max(ch)+min(ch))/2;%gia tri nguong doi voi chuan hoa phan bo chuan
%val = 1,6 % gia tri nguong doi voi cach 2 theo lab_male
plot_cut(y,fs,ch,val,f_d);%tinh toan, ve chia khung
```

2. Framing

```
function y = framing(x,fs,f_d)
% Ham nay dung de chia tin hieu thanh cac khung chong nhau
%-------------------------------------------------------------------------
% y = framing(x,fs,f_d)
% y: mang hai chieu voi so hang bang so khung va so cot bang do dai cua
% khung
% x: tin hieu can chia khung
% fs: tan so lay mau cua tin hieu
% f_d: do dai cua 1 khung
f_s = floor(f_d*fs);                          %Do dai cua khung(mau)
fs1 = floor(f_s/2);                           %Do dai cua mot nua khung(mau)
N = floor(length(x)/f_s);                     %So khung co the chia duoc
y = zeros(2*N-1,f_s);                         %Khoi tao y
%Chia khung chong cho tin hieu
for i = 1:N
    y(i*2-1,:) = x(1,(i-1)*f_s+1:i*f_s);
end
for i = 2:N
    y(2*(i-1),:) = x(1,(i-1)*f_s+1-fs1:i*f_s-fs1);
end
end
```

3. Medium Function

```
function medium = Medium_function(x)
%ham tinh gia tri trung binh
%-------------------------------------------------------------------------
% medium la gia tri trung binh sau khi tra ve
% x la tin hieu truyen vao
medium = 0;  %khoi tao gia tri trung binh
for i = 1:length(x)%Tinh gia tri trung binh
    medium = medium+x(i);
end
medium = medium/length(x);
end
```

4. Variance Function

```
 function variance = Variance_function(x)
%ham tinh phuong sai
%--------------------------------------------------------------------------
% variance la phuong sai sau khi tra ve
% x = tin hieu truyen vao
medium = Medium_function(x); %lay gia tri trung binh
variance = 0;   %khoi tao gia tri phuong sai
N = length(x); %do dai tin hieu
for i = 1:N
    variance = variance + (x(i)-medium)*(x(i)-medium);
end
variance = variance/(N-1);
end
```

5. Energy log

```
function y = energy_log(frames)%ham nay tinh nang luong khung
[r,c] = size(frames);%lay chi so cua khung
y = zeros(1,r);%tao mang luu ket qua
frames = frames.^2;%binh phuong bien do
for i = 1:r
    s = 0;
    for j=1:c
       s = s+frames(i,j);%tong nang luong cua mot khung
    end
    y(i) = log(s);%lay log cua khung
end

end%ket thuc ham
```

6. Chuẩn hóa 1

```
function y = chuanhoa1(x)%ham nay de chuan hoa dau vao ve dai [0,1]
mi = min(x);%tim gia tri min
an = max(x)-min(x);%tinh hieu cua max va min
y = (x-mi)/(an); %tien hanh chuan hoa
end
%ket thuc ham
```

7. Chuẩn hóa 2

```
function y = chuanhoa2(x)

le = length(x); % do dai tin hieu
y = zeros(1,le); %  khoi tao gia tri cho y
dolechchuan = sqrt(var(x)); %
for i=1:le
    y(i) = (x(i)-mean(x))/dolechchuan;
end
end
```

8. Plot cut

```
function plot_cut(y,fs,ch,val,f_d)
%ham nay tinh toan va ve phan doan khoang lang tieng noi
%ham nay dung  cho ca hai ch??ng trinh su dung chuan hoa nang luong
%phan doan nang luong dua vao dua vao nang luong khung da chuan hoa
%va dieu kien moi khoang lang dai hon 200ms
%--------------------------------------------------------------------------
le = length(y);      %lay do dai tin hieu
so = 0.2*fs;         %so mau ung voi 200ms
laytile = f_d*fs/2; %lam tron so mau tin hieu ung voi 1 khung nang luong
dem = so/laytile;    %so khung nang luong ung voi 200ms
loc = [];            %vector luu cac vi tri phan doan cua tin hieu
sta = [];            %vector luu trang thai vi tri(1:start,-1:end)
```

```
%-------------------------------------------------------------------------
%tinh toan phan doan tin hieu theo nang luong khung
%-------------------------------------------------------------------------
v1 = val*ones(1,length(ch));%vector nguong gioi han khoang lang tieng noi
if(ch(1)>=val)
    loc = [loc 1];
    sta = [sta 1];
end
for i=2:dem
    if((ch(i)>=val)&&(max(ch(1:i-1))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta 1];
    end
    if((ch(i)>=val)&&(max(ch(i+1:i+dem))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta -1];
    end
end
for i=dem+1:length(ch)-dem-1
    if((ch(i)>=val)&&(max(ch(i-dem:i-1))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta 1];
    end
    if((ch(i)>=val)&&(max(ch(i+1:i+dem))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta -1];
    end
end
for i=length(ch)-dem:length(ch)-1
    if((ch(i)>=val)&&(max(ch(i-dem:i-1))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta 1];
    end
    if((ch(i)>=val)&&(max(ch(i+1:end))<val))
        loc = [loc (i-1)*laytile+1];
        sta = [sta -1];
    end
end
if(ch(end)>=val&&(max(ch(end-dem:end-1)<val)))
    loc = [loc (i-1)*laytile+1];
     sta = [sta 1];
end
%sinh truc thoi gian cho tin hieu
subplot(3,1,1)
plot(y);%ve tin hieu
title('Tin hieu'); %tieu de
ylabel('Bien do');%bien do tin hieu
xlabel('Chieu dai tin hieu');%
axis([1 length(y) min(y) max(y)]);
subplot(3,1,2)
T = 1:length(ch);
plot(T,ch);%ve nang luong khung da chuan hoa
hold on;
p3=plot(v1);%ve nguong xac dinh khoang lang, tieng noi
hold off;
axis([1 length(ch) min(ch) max(ch)]);
title('Chuan hoa nang luong'); %tieu de
ylabel('Bien do');
xlabel('Khung');
legend([p3],'moc xac dinh');%dat ten
subplot(3,1,3)
plot(y);
axis([1 length(y) min(y) max(y)]);
```

```
hold on;

for i=1:length(loc)%ve chia khung voi cac chi so tim duoc
    if(sta(i) == 1)
        p1=plot([loc(i),loc(i)],[min(y) , max(y)],'r--','Linewidth',0.5);%ve
duong bat dau
    else p2=plot([loc(i),loc(i)],[min(y) , max(y)],'k--','Linewidth',0.5);%ve
duong ket thuc
    end
end
hold off;
title('Phan doan tin hieu khoang lang, tieng noi');
xlabel('Chieu dai tin hieu');
legend([p1,p2],'Bat dau','Ket thuc');%dat ten
end
%ket thuc ham
```

B. Phương pháp Mean Average

1. Main

```
%doc tin hieu dau vao
[x,Fs] = audioread('lab_male.wav');
%[x,Fs] = audioread('lab_female.wav');
%[x,Fs] = audioread('studio_male.wav');
%[x,Fs] = audioread('studio_female.wav');
N = length(x);                                          %do dai tin hieu dau
vao(mau)
f_d = 0.02;                                             %do dai 1 khung(s)
f_size = floor(f_d*Fs);                                 %do dai 1 khung(mau)
frames = framing_function(x,Fs,f_d);                    %Chia khung chong
MA = Mean_Average_function(frames);                     %Tinh cuong do tin hieu
trung binh
MA = Normally_function(MA);                             %chuan hoa tin hieu ve
doan [-1,1]
T = Threshold_setting_function(frames,MA);              %tim gia tri bien chuan
mark = Discriminated_function(T,x,Fs,f_d,MA);           %tim vi tri khoang lang và
tieng noi
Plot_Discrimination_function(x,Fs,f_d,mark,T,MA);       %Ve do thi xac dinh khoang
lang và tieng noi
```

2. Normally Function

```
function [x_nor] = Normally_function(x)
%Ham chuan hoa tin hieu ve doan [-1,1]
%-------------------------------------------------------------------------
%x_nor : tin hieu da duoc chuan hoa
%x : tin hieu dau vao
x_nor = x/abs(max(x));
end
```

3. Framing Function

```
function [frames] = framing_function(x,fs,f_d)
%Ham chia khung chong
%-------------------------------------------------------------------------
%[frames] = framing_function(x,fs,f_d)
%frames : mang luu khung da chia
%x : tin hieu dau vao
%f_d : do dai khung tin hieu
f_size = floor(f_d * fs);    %so luong mau trong 1 khung
l_s = length(x);             %do dai tin hieu
n_f = floor(l_s/f_size);     %so luong khung duoc chia theo do dai tin hieu
temp = 0;
%khung le chua cac khung duoc chia tu 0
for i = 1 : n_f
```

```
    frames(2*i-1,:) = x(temp + 1 : temp + f_size);
    temp = temp + f_size;
end
%khung chan chua cac khung duoc chia tu 1 nua do dai 1 khung
temp = f_size/2;
for i = 2 : n_f
    frames(2*(i-1),:) = x(temp + 1 : temp + f_size);
    temp = temp + f_size;
end
```

4. Mean Average Function

```
function [MA] = Mean_Average_function(frames)
%Ham tinh cuong do tin hieu trung binh
%-------------------------------------------------------------------------
%[MA] = Mean_Average_function(frames)
%MA : cuong do tin hieu trung binh
%frames : cac khung chong da duoc chia
frames = abs(frames);   %lay gia tri tuyet doi voi moi phan tu cua moi khung
[r,c] = size(frames);   %lay kich thuoc cua mang luu cac khung
%Tinh toan cuong do tin hieu trung binh cho tung frames
MA = zeros(1,r);
for i = 1:r
    for j = 1:c
        MA(i) = MA(i) + frames(i,j);
    end
end
MA = log(MA);           %lay log cua cuong do tin hieu trung binh
```

5. Threshold Setting Function

```
function T = Threshold_setting_function(frames, MA)
%Ham tinh toan bien chuan giua tieng noi va khoang lang
%-------------------------------------------------------------------------
%T = Threshold_setting_function(frames, MA)
%T : Bien chuan cua tin hieu
%frames : khung chong da chia
%MA : cuong do tin hieu trung binh
N = floor(length(MA)/2);
%%
f = zeros(1,N);
g = zeros(1,N);
for i = 1:N
    f(i) = MA(2*(i-1)+1);
end
for i = 1:N-1
    g(i) = MA(2*i);
end
%%
Tmin = min(f);                          %Bien thap nhat cua MA
Tmax = max(g);                          %Bien cao nhat cua MA
%%
T = (1/2)*(Tmin + Tmax);                %Bien chuan ban dau
%%
i = 0;
p = 0;
mark = g > T;                           %Luu cac diem MA lon hon bien chuan
g1 = frames(mark,:);                    %Dung lai cac khung co bien cao hon bien
chuan
mark = f < T;                           %Luu các diem MA nho hon biên chuan
f1 = frames(mark,:);                    %Dung lai cac khung co bien nho hon bien
chuan
%So luong khung lon hon bien chuan
for k = 1:N
    if f(k) < T
```

```
        i = i + 1;
    end
end
%So luong khung nho hon bien chuan
for h = 1:N
    if g(h) > T
        p = p + 1;
    end
end
%%
j = -1;                                 %Khoi tao j
q = -1;                                 %Khoi tao q
%%
left1 = 0;
left2 = 0;
%Lap den khi i va p la bat bien
%cung he thuc (1/i)*sum(f-T,0) - (1/p)*sum(T-g,0) = 0
while  i ~= j || p ~= q
    for k = 1:i
        if f1(k)-T > 0
            left1 = left1 + max(f1(k)-T);
        end
    end
    left1 = (1/i)*left1;
    for h = 1:p
        if T - g1(k) > 0
            left2 = left2 + max(T-g1(h));
        end
    end
    left2 = (1/p)*left2;
    if left1 - left2 > 0
        Tmin = T;
    else
        Tmax = T;
    end
    T = (1/2)*(Tmin + Tmax);            %dinh lai gia tri bien chuan hien tai
    j = i;
    q = p;
    mark = g > T;
    g1 = frames(mark,:);
    mark = f < T;
    f1 = frames(mark,:);
    i = 0;
    p = 0;
    %So luong khung lon hon bien chuan hien tai
    for k = 1:N
        if f(k) < T
            i = i + 1;
        end
    end
    %So luong khung nho hon bien chuan hien tai
    for h = 1:N
        if g(h) > T
            p = p + 1;
        end
    end
end
end
```

6. Discriminated Function

```
function mark3 = Discriminated_function(T,x,Fs,f_d,MA)
%Ham tim tieng noi và khoang lang
```

```
%-------------------------------------------------------------------------
%mark3 = Discrimination_function(T,x,Fs,f_d,MA)
%mark3 : mang danh dau khung nao là tieng noi/khoang lang
%T : gia tri bien chuan
%x : tín hieu dau vào
%Fs : tan so cua tín hieu
%f_d : do dai 1 khung(mau)
%MA : cuong do tin hieu trung binh
f_s = floor(f_d*Fs);            %Chieu rong cua 1 khung
N1 = floor(length(x)/f_s);      %So luong khung duoc chia theo chieu dai tin
hieu
mark = zeros(1,2*N1-1);         %mang danh dau khung tieng noi/khoang lang
%%
%danh dau khung nào là tieng noi/khoang lang
for i = 1:2*N1-1
    if MA(i)>T
        mark(i) = 1;
    else
        mark(i) = -1;
    end
end
%%
i = 1;
%Neu khoang lang nho hon 200ms thì danh dau la giong noi
while i <= (2*N1-1)
    if(mark(i)==(-1))
        Silence_size = 0;
        j = i;
        while mark(i) == (-1)
            Silence_size = Silence_size+1;
            i = i+1;
            if i >(2*N1-1)
                break;
            end
        end
        if Silence_size <= 10
            for k = j:(j+Silence_size-1)
                mark(k) = 1;
            end
        end
    else
        i = i + 1;
    end
end
%%
%Luu vi tri cac diem chuyen tu khoang lang qua giong noi va nguoc lai
mark1 = [];
for i = 1:2*N1-2
    if mark(i)*mark(i+1)<0
        mark1 = [mark1,i];
    end
end
%%
%So lan chuyen tu khoang lang qua giong noi va nguoc lai
N2 = length(mark1);
%Luu do dai cac doan giong noi
mark2 = zeros(1,N2);
mark2(1) = mark1(1)*mark(mark1(1));
for i = 2:N2
    mark2(i) = (mark1(i) - mark1(i-1))*mark(mark1(i));
end
%Neu do dai giong noi ngan hon 60ms thi dat lai thanh khoang lang
for i = 2:N2
```

```
    if mark2(i)>0&&mark2(i)<=3
        for k = mark1(i-1) + 1 : mark1(i)
            mark(k) = -1;
        end
    end
end
mark3 = mark;
end
```

7. Plot Discrimination Function

```
function Plot_Discrimination_function(x,fs,f_d,mark,T,MA)
%Ham ve do thi xac dinh khoang lang và tieng noi
%-------------------------------------------------------------------------
%Plot_Discrimination_function(x,fs,f_d,mark,T,MA)
%x : tin hieu dau vao
%fs : tan so dau vào
%f_d : do dai 1 khung tin hieu
%mark : vi tri tieng noi/khoang lang
%T : gia tri bien chuan
%MA : cuong do tin hieu trung binh
    N = length(x);              %do dai tin hieu(mau)
    t = 1/fs:1/fs:N/fs;         %Sinh truc thoi gian
    f_size = floor(f_d*fs);     %So mau cua 1 khung
    f_s1 = floor(f_size/2);     %1 nua mau cua 1 khung
    n_f = floor(N/f_size);      %So luong khung chia theo chieu dai tin hieu
    %luu cac vi tri chuyen tu khoang lang sang am thanh va nguoc lai
    mark1 = [];
    for i = 1:2*n_f-2
        if mark(i)*mark(i+1)<0
            mark1 = [mark1,i];
        end
    end
    %%
    plot(t,x,'Linewidth',1);        %ve tin hieu dau vao
    hold on;                        %bat ve chen
    grid on;                        %bat chia toa do
    yline(T,'--k','Linewidth',1);               %ve bien chuan
    %%
    %ve cuong do tin hieu trung binh
    MA_sample = zeros(1,N);
    temp = 0;
    %Chi ve 1 nua khung duoc chia theo do dai tin hieu
    for i = 1:n_f-1
        MA_sample(temp+1:temp + f_size) = MA((2*(i-1))+1);
        temp = temp + f_size;
    end
    MA_sample((n_f-1)*f_size:end) = MA(2*n_f-1);
    plot(t,MA_sample,'m','Linewidth',1);
    hold on;                                    %bat ve chen
    %%
    %Xac ding cac vi tri va ve vach phan dinh tieng noi/khoang lang
    for i = 1:length(mark1)
        if mod(mark1(i),2) == 1
            marking = (((mark1(i)+1)/2)*f_size+1)/fs;
        else
            marking = ((mark1(i)/2)*f_size + f_s1 + 1)/fs;
        end
            plot([marking,marking],[min(x),max(x)],'--r','Linewidth',1.5);
    end
    xlabel('Time(s)');                                  %truc thoi gian
    ylabel('Amplitude');                                %truc biên do
    title('Speech/Silence Discrimination')              %ten do thi
    legend('Base Signal','Threshold','MA','Speech');  %chu thich
```

```
end
```

C. Phương pháp kết hợp STE & ZCR

1. Main

```
[y,fs] = audioread('lab_female.wav');       %Doc tin hieu am thanh

y = y';                                     %Doi gia tri hang va cot cua ma tran

y = normalized(y);                          %Chuan hoa tin hieu
                                            %Ham mormalized la ham tu code

f_d = 0.01;                                 %Do dai cua 1 khung(giay)

frames = framing(y,fs,f_d);                 %Chia tin hieu thanh cac khung chong
                                            %nhau 50%

ste = STE(frames);                          %Tinh toan nang luong ngan han cua
                                            %moi khung
                                            %Ham STE la tu code

ste = normalized(ste);                      %Chuan hoa nang luong ngan han


zcr = ZCR(frames);                          %Tinh toan toc do bang qua 0 cua
                                            %moi khung
                                            %Ham ZCR la ham tu code

zcr = normalized(zcr);                      %Chuan hoa toc do bang qua 0

mark = discriminate(y,fs,f_d,ste,zcr);      %Danh dau cac khung tin hieu(tieng
                                            %noi danh dau la 1, khoang lang
                                            %danh dau la -1)

%Ve tin hieu, nang luong ngan han, toc do bang qua 0, va cac duong phan
%biet tieng noi va khoang lang
figure(1);
plot_signal(y,fs,f_d,ste,zcr,mark);
```

2. Framing

```
function y = framing(x,fs,f_d)
% Ham nay dung de chia tin hieu thanh cac khung chong nhau
%-------------------------------------------------------------------------
% y = framing(x,fs,f_d)
% y: mang hai chieu voi so hang bang so khung va so cot bang do dai cua
% khung
% x: tin hieu can chia khung
% fs: tan so lay mau cua tin hieu
% f_d: do dai cua 1 khung

f_s = floor(f_d*fs);                        %Do dai cua khung(mau)

fs1 = floor(f_s/2);                         %Do dai cua mot nua khung(mau)

N = floor(length(x)/f_s);                   %So khung co the chia duoc

y = zeros(2*N-1,f_s);                       %Khoi tao y

%Chia khung chong cho tin hieu
for i = 1:N
```

```
    y(i*2-1,:) = x(1,(i-1)*f_s+1:i*f_s);
end
for i = 2:N
    y(2*(i-1),:) = x(1,(i-1)*f_s+1-fs1:i*f_s-fs1);
end
end
```

3. Nomallized

```
function y = normalized(x)
%Ham nay dung de chuan hoa tin hieu
%-------------------------------------------------------------------------
%[y] = normalized(x)
%y: tin hieu sau khi chuan hoa
%x: tin hieu can chuan hoa

y = x./max((abs(x)));                       %Chuan hoa tin hieu

end
```

4. STE

```
function y = STE(frames)
% Ham nay tra ve nang luong ngan han cho moi khung
%-------------------------------------------------------------------------
% y = STE(frames)
% y: mang cac gia tri nang luong cua moi khung
% frames: mang 2 chieu voi so hang bang so khung va so cot bang do dai mot
% khung(mau)

[r,c] = size(frames);                %Xac dinh kich thuoc cua mang cac khung

y = zeros(1,r);                      %Khoi tao y

frames = frames.^2;                  %Mang frames voi moi phan tu da duoc binh
                                     %phuong

%Tinh toan nang luong cua moi khung
for i = 1:r
    for j = 1:c
        y(i) = y(i)+ frames(i,j);
    end
end
end
```

5. ZCR

```
function y = ZCR(frames)
% Ham nay tra ve toc do bang qua 0 cho moi khung
%-------------------------------------------------------------------------
% y = ZCR(frames)
% y: mang cac gia tri toc do bang qua 0 cua moi khung
% frames: mang 2 chieu voi so hang bang so khung va so cot bang do dai mot
% khung(mau)

[r,c] = size(frames);               %Xac dinh kich thuong cua mang cac khung

sgn = zeros(r,c);                   %Ham sgn[n] cua x[n] (bang 1 neu x[n]>=0,
                                    %bang -1 neu nguoc lai)

y = zeros(1,r);                     %Khoi tao y

%Tinh toan sgn[n]
for i = 1:r
```

```
    for j = 1:c
       if frames(i,j)>=0
           sgn(i,j) = 1;
       else
           sgn(i,j) = -1;
       end
    end
end

%Tinh toan toc do bang qua 0 cua moi khung
for i = 1:r
    for j = 1:c-1
        y(i) = y(i) + abs(sgn(i,j)-sgn(i,j+1));
    end
    y(i) = y(i)/(2*c);
end
end
```

6. Discriminated

```
function mark = discriminate(y,fs,f_d,ste,zcr)
% Ham nay dung de danh dau cac khung tin hieu
%--------------------------------------------------------------------------
% mark = discriminate(y,fs,f_d,ste,zcr)
% mark: mang danh dau cac khung(neu tieng noi danh dau la 1, khoang lang
% danh dau -1)
% y: tin hieu vao
% fs: tan so lay mau cua tin hieu
% f_d: do dai cua 1 khung(giay)
% ste: nang luong ngan han cua cac khung
% zcr: toc do bang qua 0 cua cac khung

f_s = floor(f_d*fs);                        %Do dai cua mot khung tin hieu(mau)

N1 = floor(length(y)/f_s);                  %So khung chia duoc

mark = zeros(1,2*N1-1);                     %Khoi tao mang danh dau

%Danh dau
for i = 1:2*N1-1
    if (ste(i)>=0.008 && zcr(i)<=0.71)
        mark(i) = 1;
    else
        mark(i) = -1;
    end
end

%Neu do dai cua khoang lang nho hon 200ms thi bo qua
i = 1;
while i <= (2*N1-1)
    if(mark(i)==(-1))
        dem = 0;
        j = i;
        while mark(i) == (-1)
            dem = dem+1;
            i = i+1;
            if i >(2*N1-1)
                break;
            end
        end
        if dem <= 40
            for k = j:(j+dem-1)
                mark(k) = 1;
```

```
            end
        end
    else
        i = i + 1;
    end
end
end
```

7. Plot Signal

```
function plot_signal(y,fs,f_d,ste,zcr,mark)
% Ham nay dung de ve tin hieu, nang luong ngan han, toc do bang qua 0, va
% cac duong phan biet tieng noi va khoang lang
%--------------------------------------------------------------------------
% plot_signal(y,fs,f_d,ste,zcr,mark)
% y: tin hieu vao
% fs: tan so lay mau cua tin hieu
% f_d: do dai cua 1 khung(giay)
% ste: nang luong ngan han cua cac khung
% zcr: toc do bang qua 0 cua cac khung
% mark: mang danh dau cac khung(neu tieng noi danh dau la 1, khoang lang
% danh dau -1)

N = length(y);                          %Do dai cua tin hieu(mau)
t = 1/fs:1/fs:N/fs;                     %Vector thoi gian roi rac
f_s = floor(f_d*fs);                    %Do dai cua mot khung(mau)
f_s1 = floor(f_s/2);                    %Do dai cua mot nua khung
N1 = floor(length(y)/f_s);              %So luong khung chia duoc(la 2*N1+1)

ste1 = zeros(1,N);                      %Mang nay dung de ve nang luong
                                        %ngan han

%Tinh toan ste1
for i = 1:N1-1
    ste1((i-1)*f_s+1:i*f_s) = ste((2*(i-1))+1);
end
ste1((N1-1)*f_s:end) = ste(2*N1-1);


zcr1 = zeros(1,N);                      %Mang nay dung de ve toc do bang
                                        %qua 0;

%Tinh toan zrc1
for i = 1:N1-1
    zcr1((i-1)*f_s+1:i*f_s) = zcr((2*(i-1))+1);
end
zcr1((N1-1)*f_s:end) = zcr(2*N1-1);

mark1 = [];                             %Mang danh dau cac vi tri giao nhau
                                        %giua tieng noi va khoang lang

%Tinh toan mark1
for i = 1:2*N1-2
    if mark(i)*mark(i+1)<0
        mark1 = [mark1,i];
    end
end

%Ve tin hieu
plot(t,y,'b','Linewidth',0.5);
axis([min(t),max(t),min(y),max(ste)]);
hold on;
```

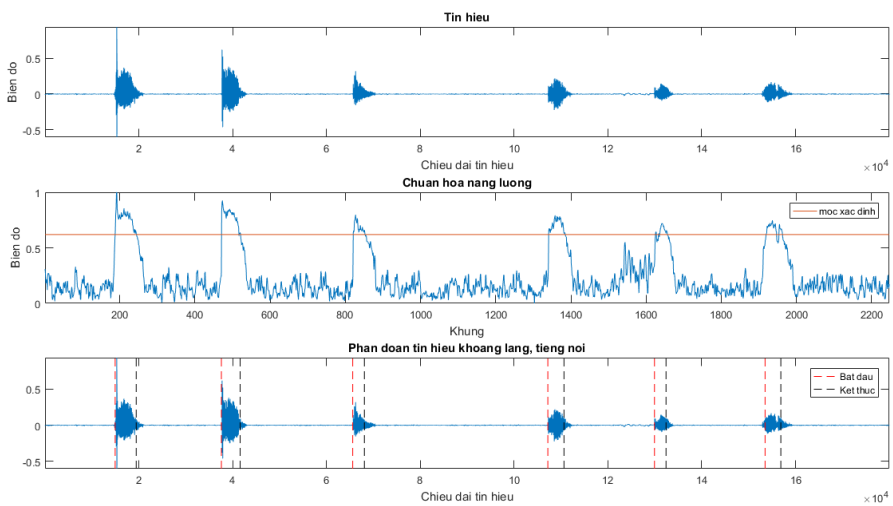
```
%Ve nang luong ngan han va toc do bang qua 0
plot(t,ste1,'Color','r','Linewidth',1);
plot(t,zcr1,'Color',[0.4660 0.6740 0.1880],'Linewidth',1);

%Ve duong phan cach giua tieng noi va khoang lang
for i = 1:length(mark1)
    if mod(mark1(i),2) == 1
        tam = (((mark1(i)+1)/2)*f_s+1)/fs;
    else
        tam = ((mark1(i)/2)*f_s + f_s1 + 1)/fs;
    end

    if mark(mark1(i)) == -1
        plot([tam,tam],[-1,1],'--','Color',[0.4940 0.1840
0.5560],'Linewidth',1.5);
    else
        plot([tam,tam],[-1,1],'--','Color','k','Linewidth',1.5);
    end
end

%Dieu chinh cac tham so
xlabel('Time(s)');
ylabel('Magnitude');
title('Speech/Silence Discrimination')
legend('Signal','STE','ZCR','Start','End');
end
```

IV. KẾT QUẢ THỰC NGHIỆM

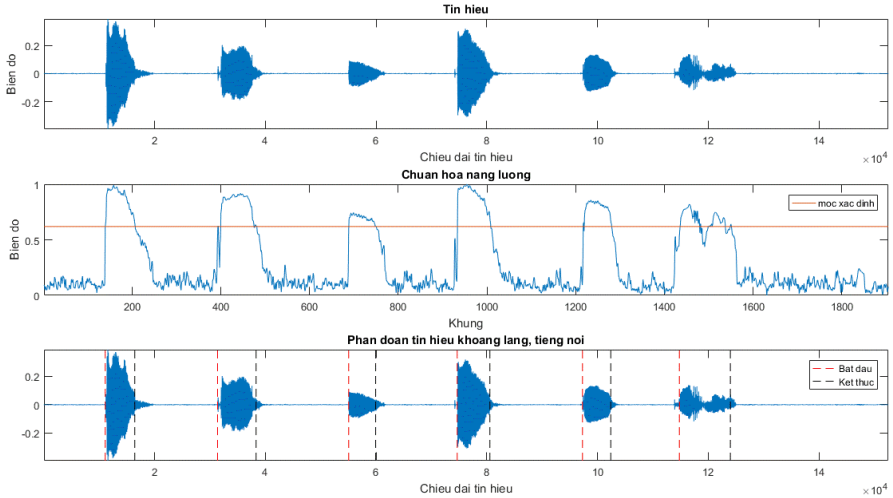
Dữ liệu dùng để đánh giá thuật toán là 4 file: “lab_male.wav”, “lab_female”, “studio_male”, “studio_female” trong thư mục tín hiệu mẫu do giảng viên cung cấp.

A. Hình vẽ

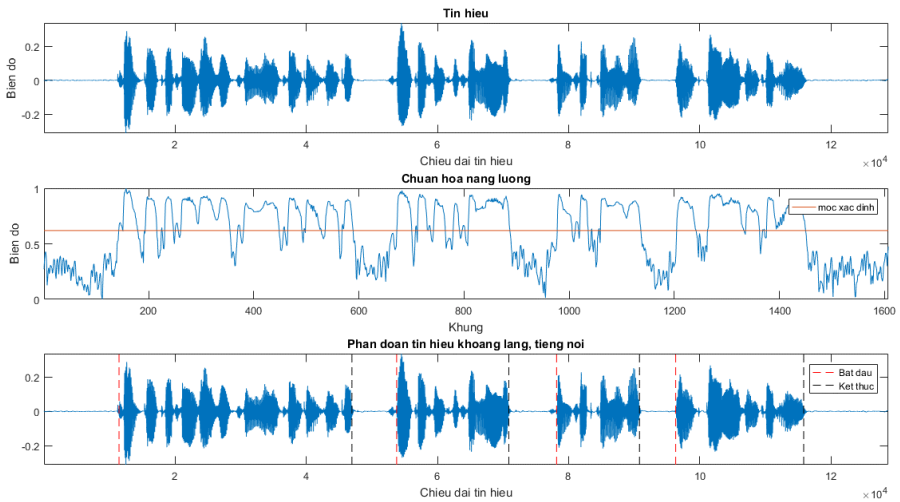
1. Phương pháp chuẩn hóa năng lượng 1



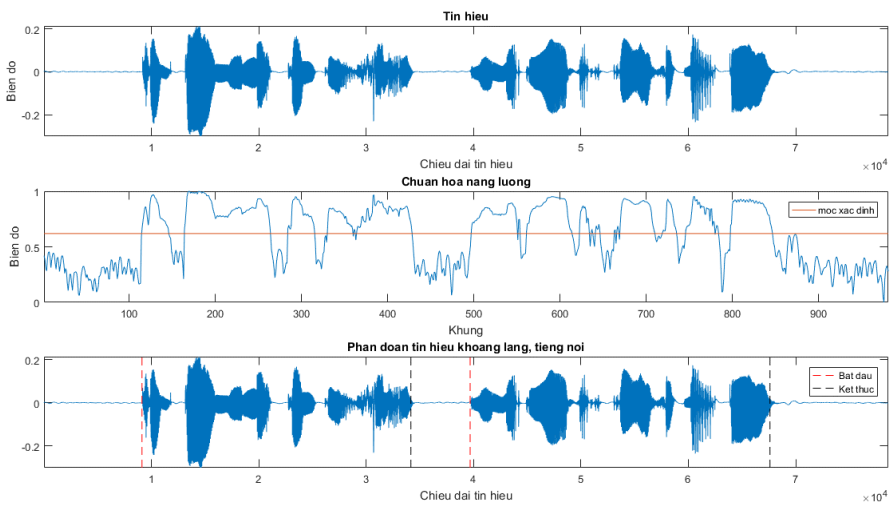
Hình 3. Phân đoạn tín hiệu “lab_male” bằng phương pháp chuẩn hóa năng lượng 1



Hình 4. Phân đoạn tín hiệu "lab_female" bằng phương pháp chuẩn hóa năng lượng 1



Hình 5. Phân đoạn tín hiệu "studio_male" bằng phương pháp chuẩn hóa năng lượng 1



Hình 6. Phân đoạn tín hiệu "studio_female" bằng phương pháp chuẩn hóa năng lượng 1

a) Nhận xét

Nhận xét chung cả 4 file tín hiệu:

Nhìn chung trong mỗi tín hiệu năng lượng của khoảng tiếng nói lớn hơn rất nhiều so với khoảng lặng. Năng lượng ở phần đầu và phần cuối của mỗi khoảng tiếng nói nhỏ hơn nhiều so với năng lượng lớn nhất của mỗi khoảng tiếng nói. Mặt khác, để lấy mốc phân định có thể áp dụng được nhiều tín hiệu nhất nên phải chọn ngưỡng khá lớn để có thể loại bỏ được hết nhiễu, dẫn đến các khoảng tiếng nói bị lấy mốc đánh ở hai đầu sâu hơn so với lấy mốc bằng phương pháp thủ công.

Nhận xét riêng:

File "lab_female.wav"(Hình 4) có nhiễu ít nhất, năng lượng khoảng lặng của file này nhỏ nhất so với năng lượng khoảng lặng của các tín hiệu còn lại, năng lượng của khoảng lặng gần như không đáng kể so với năng lượng của các khoảng tiếng nói. Ở khoảng tiếng nói thứ nhất mốc đánh dấu phía sau bị lấy vào rất sâu.

Đối với file "lab_male.wav"(Hình 3) ở giữa đoạn tiếng nói thứ 5 và thứ 6 có nhiễu lớn, dẫn đến xuất hiện những khung có năng lượng lớn trong đoạn khoảng lặng.

Đối với file "studio_male.wav"(Hình 5) và "studio_female.wav"(Hình 6) năng lượng của các đoạn tiếng nói tương đối đồng đều, đều gần đạt max, dẫn đến lấy mốc đánh dấu tương đối chính xác, độ sai lệch nhỏ hơn so với lấy mốc 2 file "lab_male.wav" (Hình 3) và "lab_female.wav"(Hình 4).

Trong file "studio_female.wav"(Hình 6) ở đoạn gần cuối của tín hiệu có nhiễu lớn, xuất hiện các khung có năng lượng lớn, đây là các khung của khoảng lặng có năng lượng lớn nhất trong cả 4 file, so với năng lượng khoảng tiếng nói trong cùng tín hiệu thì không có ảnh hưởng gì lớn, nhưng ảnh hưởng rất lớn đến việc lấy mốc phân định cho cả 4 tính hiệu. Giá trị ngưỡng phân định được lấy lớn hơn giá trị năng lượng đó.

b) Ưu điểm và nhược điểm

Ưu điểm

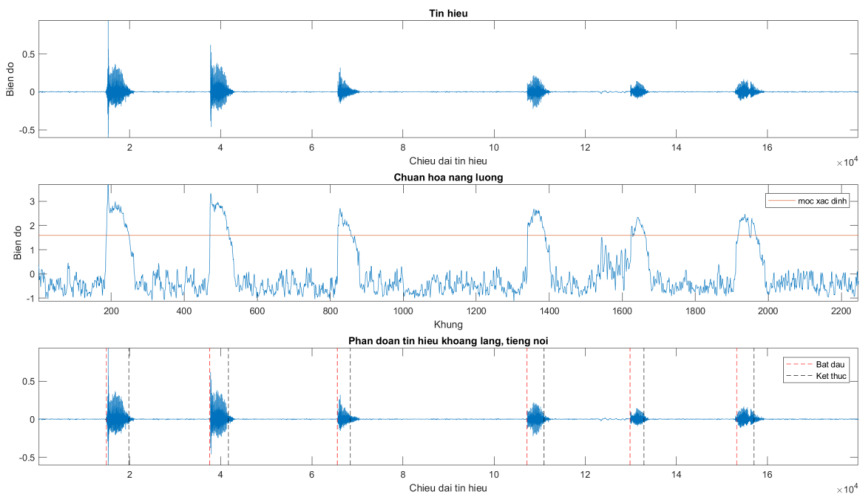
Xây dựng thuật toán chuẩn hóa năng lượng đơn giản, dễ tính toán, dễ thực hiện,xử lý nhanh .

Phân định tương đối đúng đối với những tín hiệu âm thanh đơn giản, rõ ràng, môi trường sạch, ít nhiễu, biên độ lớn

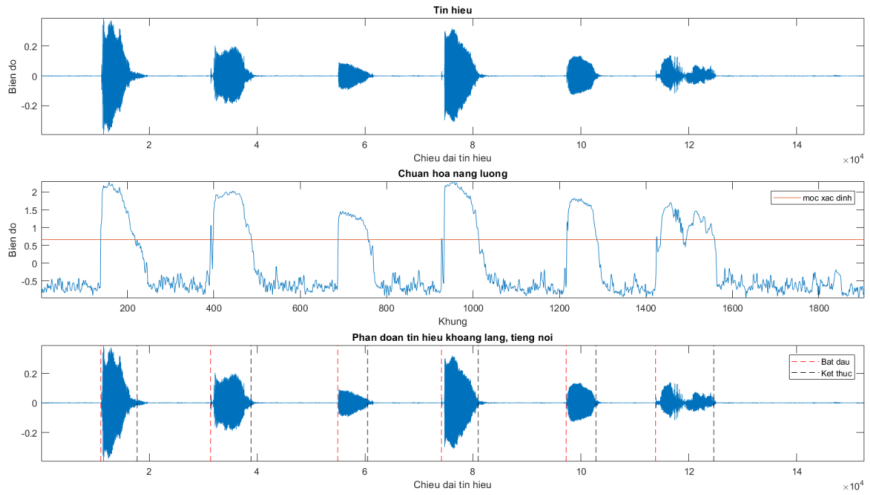
Nhược diểm

Đối với những tín hiệu âm thanh có nhiễu tương đối lớn, biên độ tín hiệu nhỏ chương trình phân định chưa chính xác tiếng nói và khoảng lặng, lấy sai mốc phân định ở phần đầu và cuối cuối của tiếng nói.

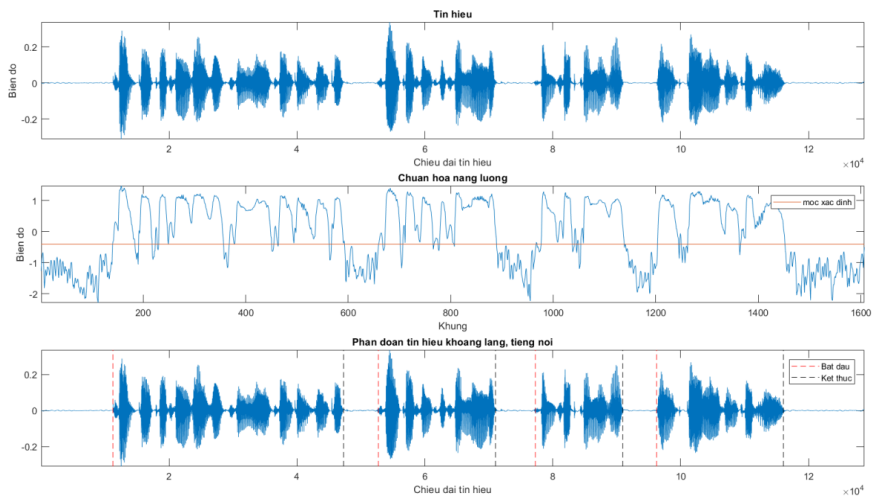
2. Phương pháp chuẩn hóa năng lượng 2



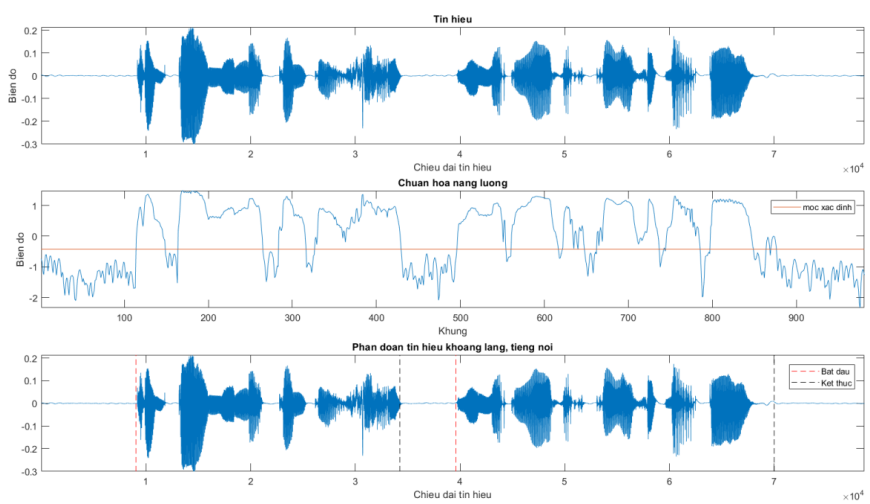
Hình 7. Phân đoạn tín hiệu "lab_male" bằng phương pháp chuẩn hóa năng lượng 2



Hình 8. Phân đoạn tín hiệu "lab_female" bằng phương pháp chuẩn hóa năng lượng 2



Hình 9. Phân đoạn tín hiệu "studio_male" bằng phương pháp chuẩn hóa năng lượng 2



Hình 10. Phân đoạn tín hiệu "studio_female" bằng phương pháp chuẩn hóa năng lượng 2

a) Nhận xét

Đối với ngưỡng $z_i = \frac{x_i - \bar{x}}{s}$ có độ chính xác tương đối ổn định đối với các file . Ngoại trừ file tín hiệu "lab_male.wav"(Hình 7) có độ nhiễu môi trường lớn nên ngưỡng có độ chính xác thấp . Nên đối tín hiệu "lab_male.wav"(Hình 7) với tín hiệu bị nhiễu lớn muốn chính xác ta có thể đo ngưỡng bằng mắt thường.

b) Ưu điểm và nhược điểm

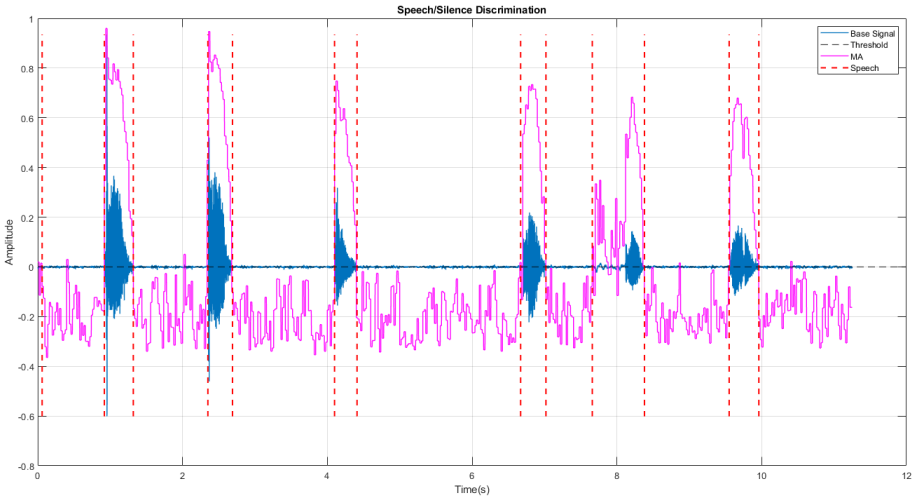
Ưu điểm

Độ chính xác tương đối ổn định.

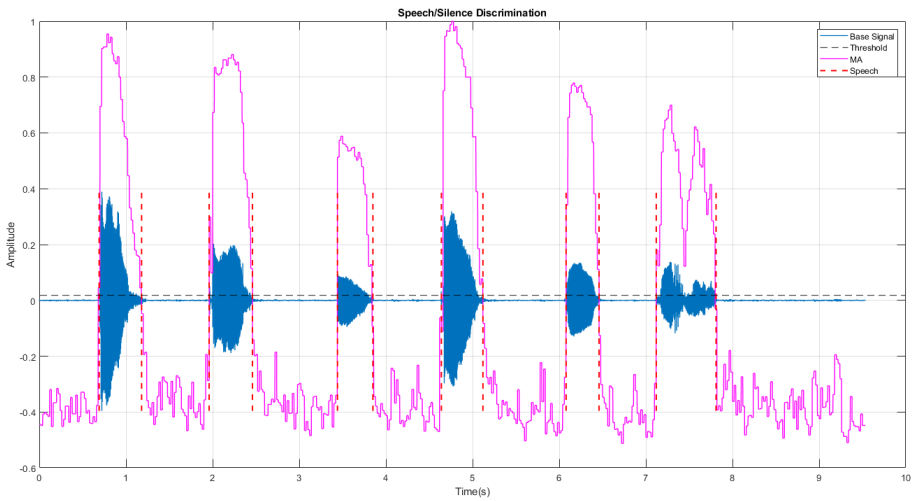
Nhược diểm

Dễ sai sót đối với tín hiệu có độ nhiễu lớn.

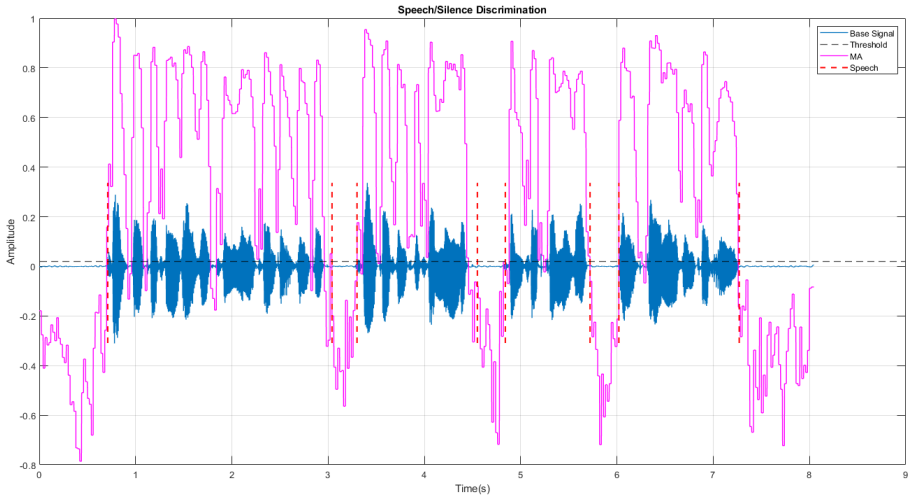
3. Phương pháp Mean Average



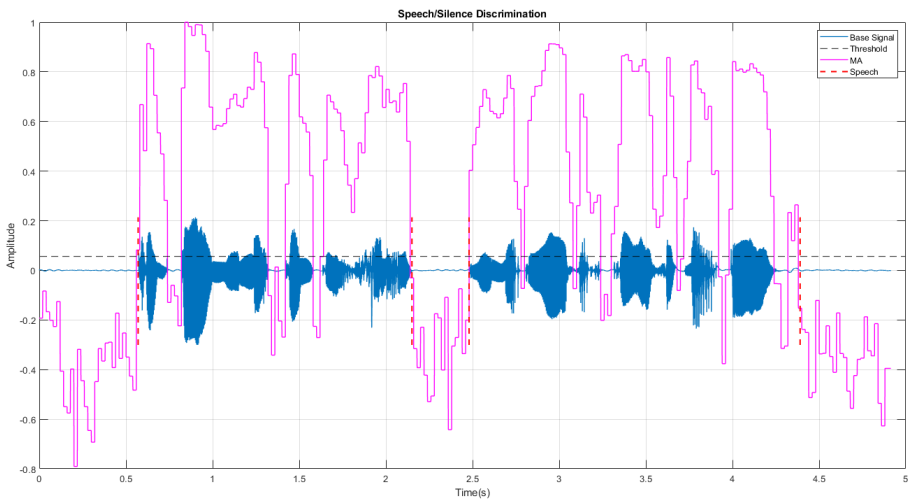
Hình 11. Phân đoạn tín hiệu "lab_male" bằng phương pháp Mean Average



Hình 12. Phân đoạn tín hiệu "lab_female" bằng phương pháp Mean Average



Hình 13. Phân đoạn tín hiệu "studio_male" bằng phương pháp Mean Average



Hình 14. Phân đoạn tín hiệu "studio_female" bằng phương pháp Mean Average

a) Nhận xét

Nhìn tổng quan sau khi xử lý cả 4 tín hiệu, ta thấy tín hiệu tiếng nói và khoảng lặng được phân đoạn ở file "lab_male.wav"(Hình 11) và file "studio_female.wav"(Hình 14) có một số sai sót. Dễ dàng nhận thấy bên trong khoảng tiếng nói đã được phân đoạn có lẫn nhiễu ở bên trong vì những nhiễu này có Magnitude Average khá lớn, có thể coi như là Magnitude Average của tín hiệu giọng nói. Dù đã thử nhiều cách xử lý nhưng việc phân đoạn nhiễu và tiếng nói có vẻ là vấn đề nan giải. Còn ở file "lab_female.wav"(Hình 12) và file "studio_male.wav"(Hình 13) thì tín hiệu giọng nói được phân đoạn tương đối chính xác.

b) Ưu điểm và nhược điểm

Ưu điểm

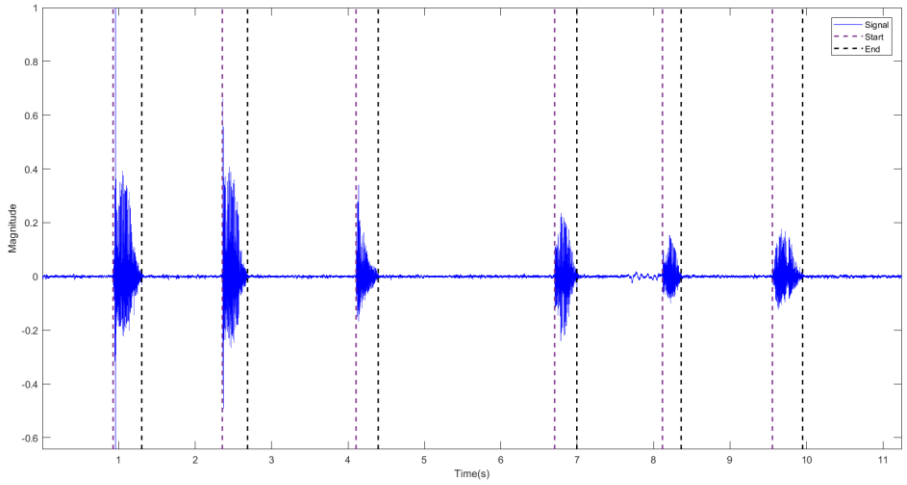
Magnitude Average của khoảng lặng và giọng nói giúp dễ dàng phân đoạn trên đồ thị.

Phương pháp Mean Average giúp ta có thể tìm ngưỡng đối với các tín hiệu khác nhau.

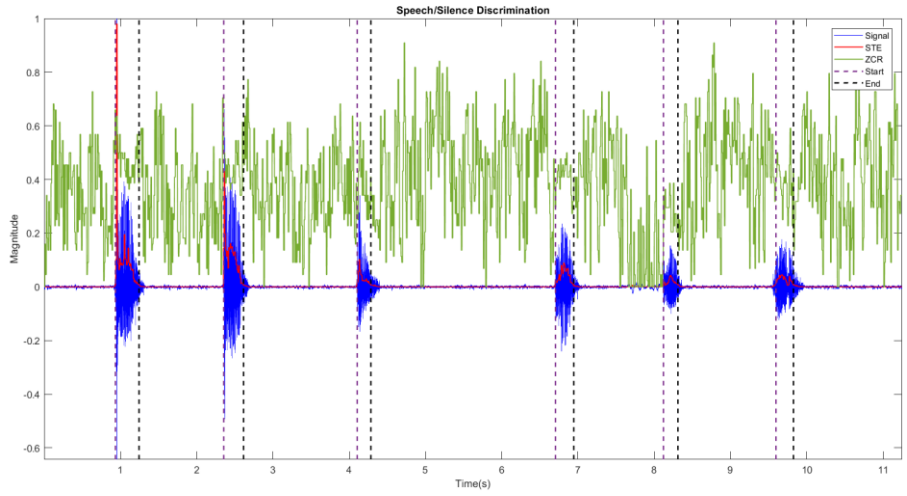
Nhược diểm

Phương pháp Mean Average vẫn chưa có thể phân đoạn hiệu quả giữa tín hiệu nhiễu và tín hiệu tiếng nói.

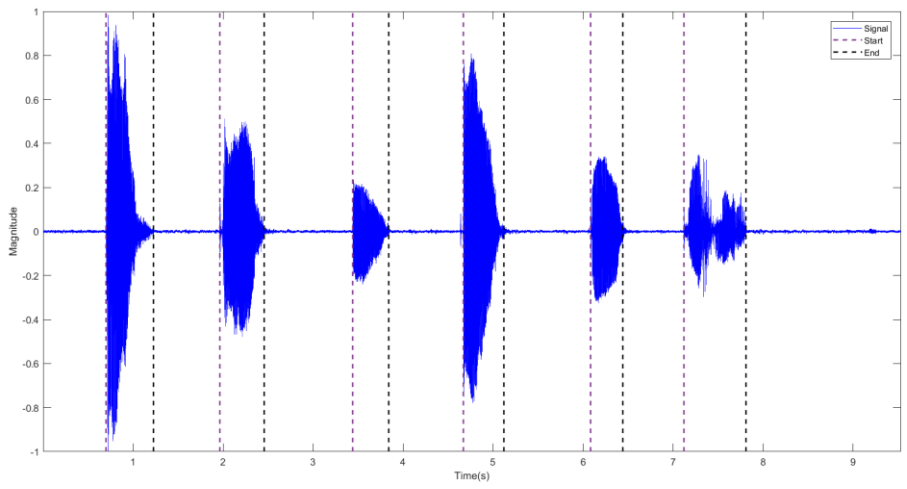
4. Phương pháp kết hợp STE & ZCR



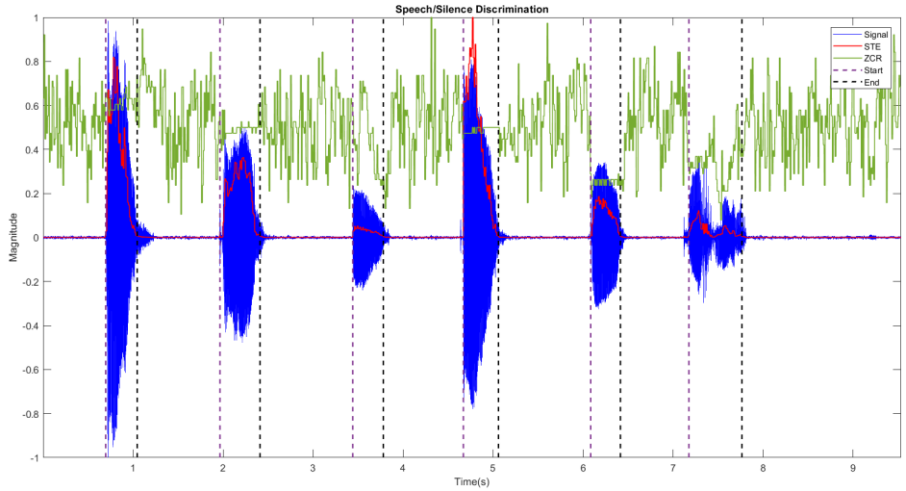
Hình 15. Phân biệt tiếng nói và khoảng lặng của file "lab_male" bằng cách thủ công



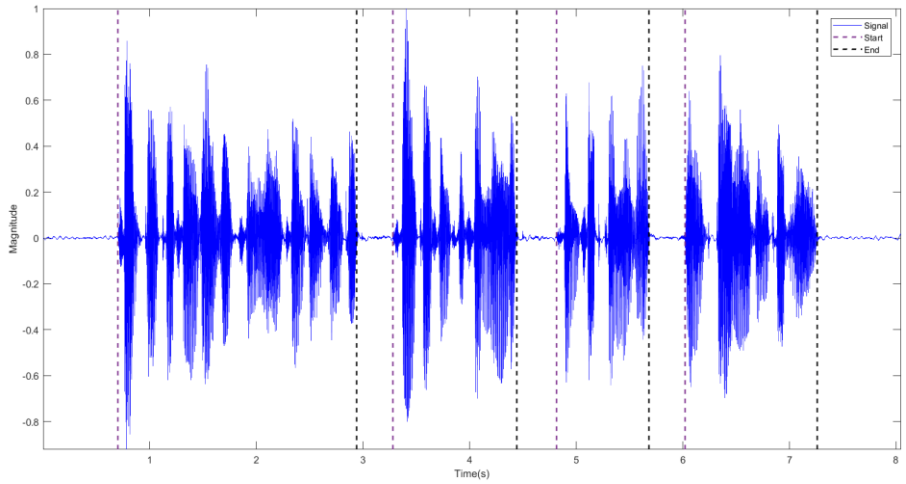
Hình 16. Phân biệt tiếng nói và khoảng lặng của file "lab_male" bằng thuật toán kết hợp giữa STE và ZCR



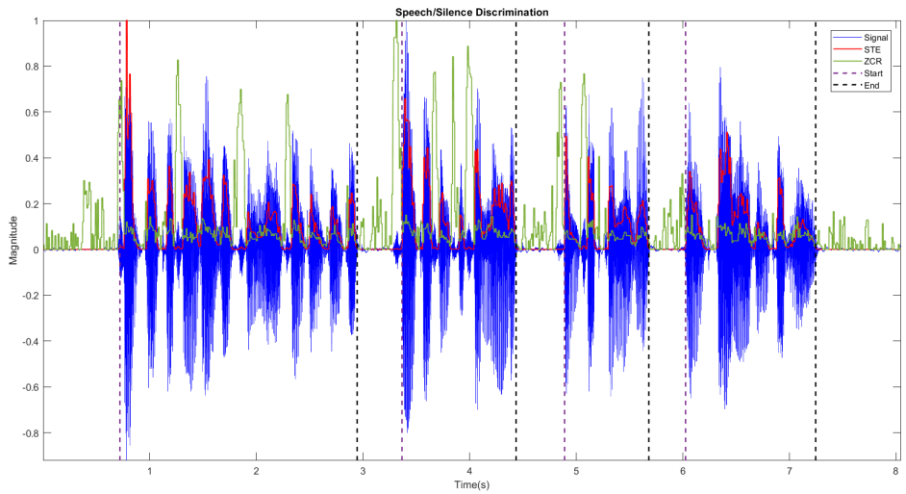
Hình 17. Phân biệt tiếng nói và khoảng lặng của file "lab_female" bằng cách thủ công



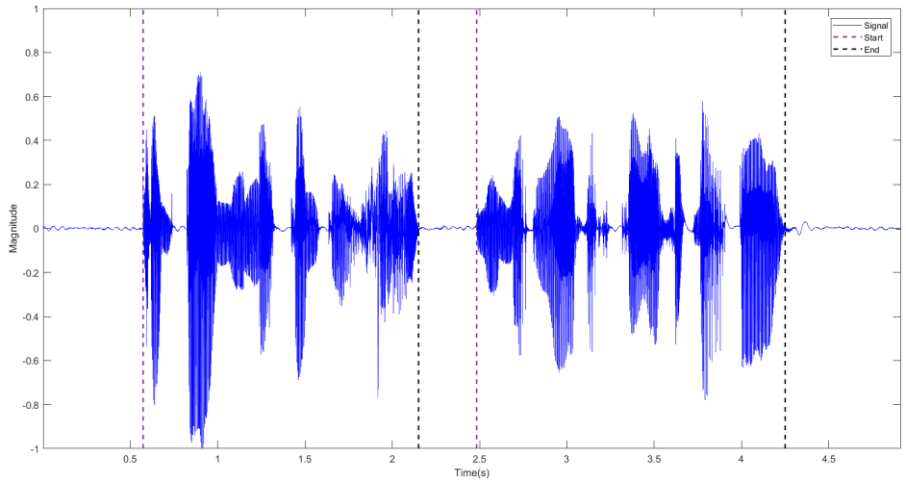
Hình 18. Phân biệt tiếng nói và khoảng lặng của file "lab_female" bằng thuật toán kết hợp giữa STE và ZCR



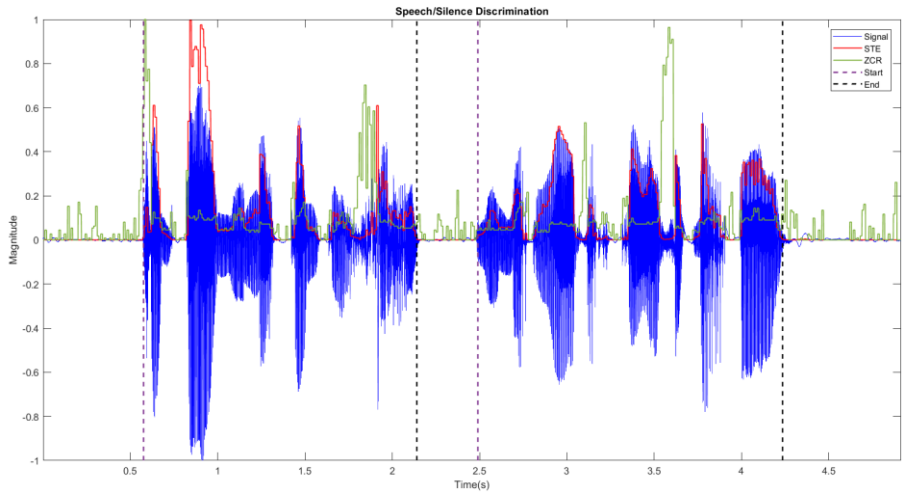
Hình 19. Phân biệt tiếng nói và khoảng lặng của file "studio_male" bằng cách thủ công



Hình 20. Phân biệt tiếng nói và khoảng lặng của file "studio_male" bằng thuật toán kết hợp giữa STE và ZCR



Hình 21. Phân biệt tiếng nói và khoảng lặng của file "studio_female" bằng cách thủ công



Hình 22. Phân biệt tiếng nói và khoảng lặng của file "studio_female" bằng thuật toán kết hợp giữa STE và ZCR

a) Nhận xét

Ở các file "lab_male.wav" và "lab_female.wav"(Hình 16 và Hình 18) , tốc độ băng qua 0 của khoảng lặng và tiếng nói ở một số chỗ xấp xỉ bằng nhau. Ở các file "studio_male" và "studio_female" (Hình 20 và Hình 22) mặc dù đã có sự khác nhau, nhưng vẫn chưa thực sự rõ ràng, ngoài ra, trong tiếng nói tồn tại một số vị trí có tốc độ băng qua 0 cao hơn rất nhiều so với khoảng lặng.

Có sự chênh lệch tốc độ băng qua 0 lớn giữa hai file "lab_male.wav" và "lab_female.wav"(Hình 16 và Hình 18) với "studio_male" và "studio_female" (Hình 20 và Hình 22).

Đối với các file "lab_male.wav" và "lab_female",năng lượng ở cuối tiếng nói khá nhỏ nên các vị trí kết thúc tiếng nói bị cắt vào sâu hơn một chút so với cách làm thủ công.

Đối với file "studio_male.wav", , ta có thể thấy tại khoảng tiếng nói thứ 2 và 3 vị trí bắt đầu tiếng nói bị mất đi một đoạn tín hiệu nhỏ do ở đó tốc độ băng qua 0 quá cao.

Đối với file "studio_female", thuật toán hoạt động khá chính xác, các khoảng chia hầu như trùng khớp với cách thủ công

b) Ưu diểm và nhược điểm của thuật toán

Ưu điểm:

Phân biệt khá chính xác tiếng nói và khoảng lăng.

Năng lượng thể hiện sự khác nhau rất rõ ràng giữa tiếng nói và khoảng lặng.

Nhược điểm:

Do năng lượng ở vị trí từ tiếng nói chuyển qua khoảng lặng khá nhỏ nên sẽ rất dễ mất những đoạn tín hiệu này, đặc biệt đối với môi trường nhiễu nặng .

Tốc độ băng qua 0 chưa có sự phân biệt rõ ràng giữa tiếng nói và khoảng lặng, đối với các tín hiệu khác nhau thì sai lệch khá lớn, dẫn đến không đóng góp nhiều vào việc làm tăng độ chính xác của thuật toán.

B. Bảng biểu

Bảng 1. Bảng so sánh các kết quả thu được và mốc tiếng nói trong wavesurfer

File	Mốc tiếng nói trong wavesurfer	Mốc chia theo chuẩn hóa 1	Mốc chia theo chuẩn hóa 2	Mốc chia theo pp Mean Average	Mốc chia theo pp ste và zcr
Lab_male	0.925-1.300	0.935-1.215	0.930-1.240	0.920-1.320	0.935-1.245
	2.355-2.685	2.350- 2.600	2.350- 2.605	2.350-2.690	2.355-2.615
	4.105-4.395	4.100-4.255	4.100-4.275	4.100-4.410	4.105-4.285
	6.705-6.995	6.700- 6.915	6.700- 6.935	6.670-7.020	6.705-6.945
	8.116-8.360	8.120-8.275	8.115-8.305	7.660-8.380	8.120-8.310
	9.555-9.950	9.595- 9.805	9.580- 9.815	9.550-9.960	9.595-9.825
Lab_female	0.698-1.226	0.690-1.025	0.690-1.110	0.690-1.180	0.695-1.045
	1.964-2.456	1.960-2.395	1.960-2.430	1.960-2.460	1.965-2.410
	3.440-3.840	3.440-3.745	3.435-3.780	3.440-3.850	3.440-3.780
	4.670-5.120	4.665-5.035	4.635-5.060	4.460-5.120	4.670-5.060
	6.083-6.441	6.080-6.400	6.080-6.425	6.080-6.460	6.085-6.415
	7.120-7.810	7.175-7.750	7.115-7.790	7.120-7.810	7.175-7.765
Studio_male	0.701-2.940	0.715-2.935	0.700-2.955	0.710-3.040	0.720-2.945
	3.280-4.442	3.360-4.430	3.295-4.440	3.300-4.550	3.365-4.435
	4.815-5.682	4.885-5.675	4.830-5.685	4.840-5.720	4.890-5.680
	6.020-7.260	6.020-7.240	6.015-7.255	6.020-7.270	6.025-7.245
Studio_female	0.573-2.150	0.570-2.135	0.565-2.140	0.570-2.150	0.575-2.140
	2.483-4.250	2.480-4.225	2.475-4.375	2.480-4.390	2.490-4.235

C. So sánh các thuật toán

Dựa vào bảng biểu ở trên(Bảng 1), ta thấy đối với hai file “lab_male.wav” va file ”lab_female.wav”, mặc dù sai lệch khá lớn ở đoạn tiếng nói thứ 4 nhưng những đoạn còn lại thuật toán Mean Average cho kết quả chính xác hơn những thuật toán còn lại, thuật toán chuẩn hóa năng lượng 1 cho kết quả kém chính xác nhất. Đối với file “studio_male.wav” thì thuật toán chuẩn hóa năng lượng 2 cho kết quả chính xác nhất, thuật toán chuẩn hóa năng lượng 1 lại cho kết quả kém chính xác nhất. Đối với file “studio_female.wav” ta có thuật toán kết hợp giữa Short Time Enegy và Zero-crossing Rate cho kết quả đúng nhất, thuật toán Mean Average cho kết quả tốt ở hai file đầu thì ở file này lại cho kết quả kém chính xác nhất.

V. KẾT LUẬN

Thông qua bài tập nhóm này, chúng em đã có được những kiến thức cơ bản trong lĩnh vực xử lý tín hiệu tiếng nói và âm thanh. Có thể phân đoạn tiếng nói và khoảng lặng tương đối chính xác thông qua các phương pháp khác nhau dựa trên việc phân tích các đặt trưng của tín hiệu âm thanh và đã phát triển được các kĩ năng hữu ích như làm việc nhóm, tìm hiểu tài liệu, lập trình trên Matlab, … Song, chúng em vẫn còn thiếu nhiều kinh nghiệm nên vẫn còn một số sai sót nhưng chúng em sẽ cố gắng cải thiện trong tương lai.

VI. TÀI LIỆU THAM KHẢO

[1] Matthieu Hodgkinson, “CS425 Audio and Speech Processing”, Department of Computer Science, National University of Ireland, Maynooth, April 25, 2012.
[2] Lawrence R. Rabiner and Ronal W.Schafer , “Digital Processing Of Speech Signals,Prentice Hall”, 1978.
[3] Bachu R.G, Kopparthi S, Adapa B, Barkana B.D,”Separation of Voiced and Unvoiced using Zero crossing rate and Energy of the Speech Signal ”,Electrical Engineering Department ,University of Bridgeport, 2015.
[4] Phạm Văn Sự, Lê Xuân Thành, “Bài giảng Xử lý tiếng nói”, Học viện Công nghệ Bưu chính Viễn thông,2010.
[5] Chuẩn hóa dữ liệu : “https://www.statisticshowto.com/probability-and-statistics/normal-distributions/normalized-data-normalization/”