**PHÂN ĐOẠN TÍN HIỆU THÀNH TIẾNG NÓI VÀ KHOẢNG LẶNG**

**DỰA VÀO NĂNG LƯỢNG NGẮN HẠN CỦA TÍN HIỆU**

**Nguyễn Nhật Tùng, Ninh Hải Hoàng**

Khoa Công nghệ thông tin, Trường Đại học Bách Khoa, Đại học Đà Nẵng

*nhattungnguyen.2kgl@gmail.com, ninhhaihoang@gmail.com*

**Nhóm X, lớp HP: 18N15**

|  |  |  |  |
| --- | --- | --- | --- |
| **Điểm** | **Bảng phân công nhiệm vụ** | | **Chữ ký của SV** |
|  | Nguyễn Nhật Tùng (nhóm trưởng) | Viết báo cáo (tr. 1, 3), viết báo cáo kết quả thực nghiệm và kết luận (tr. 7-14). |  |
|  | Ninh Hải Hoàng | Viết báo cáo (tr. 1, 3), đọc tài liệu, viết báo cáo về cơ sở lý thuyết,mã cài đặt thuật toán và tài liệu tham khảo (tr. 3-7) |  |

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— Tự động phân đoạn tín hiệu thành tiếng nói và khoảng lặng là công việc cơ bản mà quan trọng trong lĩnh vực xử lý tín hiệu số, đặc biệt là trong việc nhận diện giọng nói, giúp xác định các khoảng có tiếng nói và khoảng không có tiếng nói. Thuật toán được trình bày sau đây chia tín hiệu âm thanh đầu vào thành các đoạn ngắn, sau đó dựa vào năng lượng của tín hiệu trên từng đoạn đem so sánh với giá trị năng lượng chuẩn làm mốc để phân đoạn tín hiệu đầu vào thành các đoạn tiếng nói và khoảng lặng. Kết quả thử nghiệm với 4 mẫu tín hiệu cho thấy các biên được tìm tự động có giá trị gần đúng với giá trị biên tìm bằng phương pháp thủ công, với sai số trung bình là: 0.0533 giây, hoặc 0.0420 giây (khi dùng phương pháp chuẩn hóa về thang giá trị [0;1]).

Từ khóa— xử lý tín hiệu số, phân biệt tiếng nói và khoảng lặng, năng lượng ngắn hạn.

Mục lục

[I. ĐẶT VẤN ĐỀ 3](#_Toc55335758)

[II. CƠ SỞ LÝ THUYẾT VÀ CÁC THUẬT TOÁN 3](#_Toc55335759)

[A. Sơ đồ khối thuật toán 3](#_Toc55335760)

[B. Thuật toán tính năng lượng ngắn hạn 3](#_Toc55335761)

[C. Thuật toán chuẩn hóa 4](#_Toc55335762)

[*1.* Chuẩn hóa sử dụng phân phối chuẩn tắc 4](#_Toc55335763)

[*2.* Chuẩn hóa về thang giá trị [0;1] 4](#_Toc55335764)

[D. Thuật toán tìm giá trị biên 4](#_Toc55335765)

[E. Lỗi biên ảo 4](#_Toc55335766)

[*1.* Vấn đề 4](#_Toc55335767)

[*2.* Giải pháp đề xuất và kết quả 5](#_Toc55335768)

[III. MÃ CHƯƠNG TRÌNH CÀI ĐẶT CÁC THUẬT TOÁN 6](#_Toc55335769)

[A. Hàm tính năng lượng mỗi frame: 6](#_Toc55335770)

[B. Hàm chuẩn hoá tín hiệu X sử dụng phân phối chuẩn tắc: 6](#_Toc55335771)

[C. Hàm chuẩn hoá tín hiệu X về dạng [0;1]: 6](#_Toc55335772)

[D. Hàm tính các giá trị biên: 6](#_Toc55335773)

[E. Hàm lọc biên ảo 7](#_Toc55335774)

[F. Hàm tổng hợp (sử dụng phương pháp chuẩn hóa dùng phân phối chuẩn tắc) 7](#_Toc55335775)

[IV. KẾT QUẢ THỰC NGHIỆM 7](#_Toc55335776)

[A. Dữ liệu mẫu 7](#_Toc55335777)

[B. Kết quả thu được 8](#_Toc55335778)

[C. Sử dụng thuật toán chuẩn hóa về thang giá trị [0;1] 11](#_Toc55335779)

[V. KẾT LUẬN 14](#_Toc55335780)

[VI. TÀI LIỆU THAM KHẢO 14](#_Toc55335781)

# ĐẶT VẤN ĐỀ

Kể từ khi các kỹ sư có thể dựa vào các đặc trưng âm thanh để phân biệt các âm tiết, người ta đã nảy ra nhiều ý tưởng để nhận biết các loại âm tiết khác nhau, từ các âm tiết cơ bản tới các âm tiết phức tạp, nhằm mục đích nhận biết các từ ngữ dựa vào tín hiệu âm thanh được ghi âm.

Tuy nhiên để nhận biết được chính xác hơn, hệ thống nhận biết từ ngữ đó cần phải sử dụng các đoạn âm thanh chỉ chứa tiếng nói (hay các tín hiệu âm thanh có giá trị). Vì vậy việc tự động phân đoạn tín hiệu thành tiếng nói và khoảng lặng giúp xác định các khoảng có tiếng nói và khoảng không có tiếng nói từ đó giúp hệ thống khác trích xuất tín hiệu giọng nói và dùng cho việc nhận diện giọng nói sau đó.

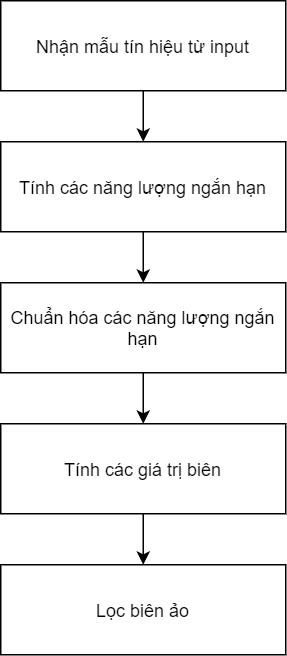
Thuật toán được trình bày sau đây thực hiện việc tự động phân đoạn tín hiệu thành tiếng nói và khoảng lặng dựa vào năng lượng ngắn hạn (short-time energy) của tín hiệu âm thanh. Cụ thể: Thuật toán này chia tín hiệu âm thanh đầu vào thành các đoạn ngắn, sau đó dựa vào năng lượng của tín hiệu trên từng đoạn đem so sánh với giá trị năng lượng làm mốc xác định để phân đoạn tín hiệu đầu vào thành các đoạn tiếng nói và khoảng lặng.

Bài viết có bố cục như sau: Phần II trình bày tổng quan về cơ sở lý thuyết liên quan tới tín hiệu âm thanh, nguyên lý của các thuật toán, những vấn đề phát sinh trong thuật toán và cách khắc phục. Phần III ghi mã nguồn cách cài đặt thuật toán bằng Matlab. Phần IV trình bày kết quả thu được và các đánh giá kết quả đó khi áp dụng thuật toán lên các dữ liệu mẫu; so sánh thuật toán chuẩn hóa hiện tại với thuật toán chuẩn hóa khác. Cuối cùng là kết luận rút ra được trình bày ở phần V.

# CƠ SỞ LÝ THUYẾT VÀ CÁC THUẬT TOÁN

## Sơ đồ khối thuật toán

Nội dung thuật toán được tóm tắt bằng sơ đồ khối trong hình sau.



Hình 1. Sơ đồ khối cho thuật toán.

## Thuật toán tính năng lượng ngắn hạn

Năng lượng của một tín hiệu rời rạc x[n] được tính bằng công thức:

Như vậy năng lượng của một đoạn tín hiệu x[n] giới hạn bởi mẫu thứ nL và mẫu thứ nR (1 ≤ n­L ≤ nR) là:

Thuật toán được trình bày bên dưới quy ước khung tín hiệu có độ dài 10ms. Như vậy nếu tín hiệu vào có tần số lấy mẫu là 16 kHz thì một khung chứa 160 mẫu tín hiệu từ đó ta có công thức tính năng lượng ngắn hạn:

Với và Ek là năng lượng ngắn hạn của khung tín hiệu thứ k.

## Thuật toán chuẩn hóa

### Chuẩn hóa sử dụng phân phối chuẩn tắc

Với biến ngẫu nhiên X có phân phối chuẩn: X ~ N(µ,) thì dạng phân phối chuẩn tắc của nó được tính bằng công thức:

### Chuẩn hóa về thang giá trị [0;1]

Với biến ngẫu nhiên X có giá trị lớn nhất là , giá trị nhỏ nhất là thì dạng chuẩn hóa về thang giá trị [0;1] của nó được tính bằng công thức:

## Thuật toán tìm giá trị biên

Giá trị biên được xác định bằng cách so sánh từng mức năng lượng ngắn hạn đã chuẩn hóa với ngưỡng năng lượng chuẩn, với ngưỡng năng lượng chuẩn (E0) là hằng số được xác định từ trước giúp phân biệt khung năng lượng nào chứa tiếng nói và khung năng lượng nào không có tiếng nói.

Hai phép so sánh được sử dụng để xác định giá trị biên: để tìm biên tiếng nói và để tìm biên khoảng lặng. Tuy nhiên thuật toán khi cài đặt đã rút gọn thành 1 phép so sánh bằng cách sử dụng biến nhớ *t* được mô tả như sau:

Phép so sánh được rút gọn thành:

Và giá trị t được cập nhật sau mỗi lần tìm được biên: *t = -t*.

## Lỗi biên ảo

### Vấn đề

Trong thực tế, các tín hiệu giọng nói thường có những lúc ngưng nghỉ ngắn làm xuất hiện các khung năng lượng nhỏ hơn ngưỡng năng lượng chuẩn, hoặc những lúc môi trường đột ngột tạo tiếng động lớn làm xuất hiện một hoặc một vài khung năng lượng có giá trị cao bất thường dẫn tới thuật toán phát hiện sai giá trị biên.

Các biên ảo do thuật toán tìm được cho 1 tín hiệu mẫu được minh họa bằng hình bên dưới.



Hình 2. Lỗi xác định biên xuất hiện các biên ảo.

### Giải pháp đề xuất và kết quả

Ta thêm một điều kiện là chiều dài tối thiểu của khoảng lặng để loại bỏ những biên ảo. Cụ thể, ta quy ước chiều dài tối thiểu của khoảng lặng là 200ms, khi đó nếu khoảng cách giữa 2 biên nhỏ hơn 200ms thì chương trình sẽ nhận định đó là biên ảo và loại bỏ nó.

Kết quả sau khi lọc biên ảo đem so sánh với khi chưa lọc được minh họa trong hình bên dưới.



Hình 3. Kết quả khắc phục lỗi xuất hiện các biên ảo.

# MÃ CHƯƠNG TRÌNH CÀI ĐẶT CÁC THUẬT TOÁN

Mã chương trình ở đây viết trên ngôn ngữ Matlab.

## Hàm tính năng lượng mỗi frame:

function [ ek ] = seframes(y, F, flen)

% Short-time Energy frames from y and F, where

% Inputs: y: input signal

% F: sampling rate

% flen: frame length

% ek: vector of energy in each frame

N=length(y); % length of y (samples)

alen = N\*1000/F; % audio length (milliseconds)

frames = ceil(alen/flen); % number of frames

elms = flen\*F/1000; % number of samples in each frame

ek=zeros(frames,1); % initializing ek

for k=1:frames

rightB = k\*elms; % calculating left and right limit

leftB = rightB-elms+1;

if(rightB>N)

rightB=N;

end

ek(k) = log(sum(y(leftB:rightB).^2))/log(10); % take log base 10 of calculated short-time energy

end

## Hàm chuẩn hoá tín hiệu X sử dụng phân phối chuẩn tắc:

function [Y] = stdnormalize(X)

xtb = mean(X);

devi = std(X); % standard deviation

Y=(X-xtb)/devi; % normalizing

end

## Hàm chuẩn hoá tín hiệu X về dạng [0;1]:

function [Y] = datanormalize(X)

% Normalize X to [0;1] value range

xmax = max(X);

xmin = min(X);

N=length(X);

Y=zeros(1,N);

for i=1:N

Y(i)=(X(i)-xmin)/(xmax-xmin); % normalizing

end

end

## Hàm tính các giá trị biên:

function b = svboundaries(nsenergy,E0)

% svboudnaries: return vector of silence - voiced boundaries based on the signal

% short-time energy frames

% Inputs: nsenergy: vector of normalized short-time energy frames

% E0: E0 used to distinguish silence or voiced

L = length(nsenergy); % length (samples) of nsenergy vector

b=[]; % initialize as vector zero

b=[b,1];

t=1; % helper variable

if(nsenergy(1)>=E0) % initializing t

t=-1;

end

for i=2:L

if(t\*nsenergy(i)>=t\*E0) % compare each short-time energy with E0

b=[b,i];

t=-t;

end

end

b=[b,L+1];

end

## Hàm lọc biên ảo

function b = vbfilter(vb,cond)

% vbfilter: returns vector of silence - voiced boundaries with virtual

% boundaries filtered.

% Inputs: vb: vector of boundaries including virtual boundaries

% cond: minimum length in frames of a silence span

b = []; % intialize b as vector zero

for i=1:2:length(vb)-1

if(vb(i+1)-vb(i)>cond) % verify condition

b=[b,vb(i),vb(i+1)]; % include 2 boundaries in b

end

end

end

## Hàm tổng hợp (sử dụng phương pháp chuẩn hóa dùng phân phối chuẩn tắc)

function b = svfilterstd(y, F)

% Silence - Voiced filter: Returns vector of boundaries using standard distribution normalization

%Inputs: y: Audio signal to find boundaries

% F: Sampling frequency

flen = 10; % frame length in ms

cond = 20; % minimum length (frames) for a span to be silence span

E0 = 0.4; % threshold for standard distribution normalization

senergy = seframes(y,F,flen); % find short-time energy of input signal

nsenergy = stdnormalize(senergy); % normalize short-time energy using standard distribution

vb = svboundaries(nsenergy, E0); % find silence - voiced boundaries

b = vbfilter(vb, cond); % filter out virtual boundaries

end

# KẾT QUẢ THỰC NGHIỆM

## Dữ liệu mẫu

Dữ liệu mẫu được sử dụng để đánh giá thuật toán là 4 tín hiệu giọng nói được thu âm bởi 2 người khác nhau trong 2 môi trường khác nhau, được lấy mẫu với tần số lấy mẫu là 16 kHz, độ dài trung bình là 8 giây.



Hình 4. Các tín hiệu mẫu được sử dụng để đánh giá thuật toán.

## Kết quả thu được

Áp dụng thuật toán cho từng dữ liệu mẫu với ngưỡng năng lượng chuẩn E0=0.4, ta thu được kết quả như trong các hình bên dưới.



Hình 5. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “lab-male”.

C:\Users\nhatt\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result_lab-female_std.eps

Hình 6. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “lab-female”.



Hình 7. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “studio-male”.



Hình 8. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “studio-female”.

Sai số trung bình (Root mean square error – RMSE) giữa biên chuẩn và biên tìm bằng thuật toán cho từng dữ liệu mẫu được thể hiện trong hình dưới đây.



Hình 9. Sai số trung bình (RMSE) của thuật toán cho từng dữ liệu mẫu.

Như vậy sai số trung bình của thuật toán trong 4 dữ liệu mẫu là 5.3267 khung (0.0533 giây).

Các tham số ảnh hưởng đến sai số của thuật toán: Mức độ chênh lệch về năng lượng giữa tiếng ồn từ môi trường và tiếng nói trong tín hiệu, và ngưỡng năng lượng chuẩn E0 được chọn làm mốc. Cụ thể, sai số của thuật toán càng thấp khi mức độ chênh lệch năng lượng giữa tiếng ồn môi trường và tiếng nói càng cao (như khi áp dụng thuật toán cho tín hiệu “lab-female”), hoặc khi ngưỡng năng lượng chuẩn E0 có giá trị càng nhỏ sao cho E0 lớn hơn năng lượng tối đa của tiếng ồn môi trường.

## Sử dụng thuật toán chuẩn hóa về thang giá trị [0;1]

Bằng cách sử dụng thuật toán chuẩn hóa khác, đưa năng lượng ngắn hạn về thang giá trị [0;1], ta cũng thu được kết quả tương đối chính xác như các hình bên dưới.



Hình 10. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “lab-male” (chuẩn hóa về [0;1]).



Hình 11. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “lab-female” (chuẩn hóa về [0;1]).



Hình 12. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “studio-male” (chuẩn hóa về [0;1]).



Hình 13. So sánh biên tìm được với biên chuẩn cho tín hiệu mẫu “studio-female” (chuẩn hóa về [0;1]).

Sai số của thuật toán sử dụng cách chuẩn hóa này được thể hiện trong hình sau.



Hình 14. Sai số trung bình (RMSE) của thuật toán cho từng dữ liệu mẫu (sử dụng phương pháp chuẩn hóa về [0;1]).

Như vậy sai số trung bình là 4.2041 khung (0.0420 giây).

Qua 4 dữ liệu mẫu, ta thấy sử dụng thuật toán chuẩn hóa này đem lại sai số nhỏ hơn so với cách chuẩn hóa đưa về phân phối chuẩn tắc. Sai số của thuật toán phân đoạn tiếng nói-khoảng lặng sử dụng 2 phương pháp chuẩn hóa khác nhau được thể hiện trong hình sau đây.



Hình 15. So sánh sai số trung bình (RMSE) của thuật toán cho từng dữ liệu mẫu khi dùng từng cách chuẩn hóa.

# KẾT LUẬN

Nhóm đã cài đặt thành công thuật toán tự động phân đoạn tiếng nói và khoảng lặng dựa vào năng lượng của tín hiệu theo 2 phương pháp chuẩn hóa khác nhau là đưa về phân phối chuẩn tắc và chuẩn hóa về thang giá trị [0;1].

Kết quả thử nghiệm với 4 mẫu tín hiệu cho thấy các biên được tìm tự động có giá trị gần đúng với giá trị biên tìm bằng phương pháp thủ công, với sai số trung bình là: 0.0533 giây, hoặc 0.0420 giây (khi dùng phương pháp chuẩn hóa về thang giá trị [0;1]). Kết quả đó cũng cho thấy sử dụng phương pháp chuẩn hóa đưa về thang giá trị [0;1] giúp giảm sai số của thuật toán.

Trong tương lai nhóm sẽ thử nghiệm thuật toán với nhiều mẫu dữ liệu khác để tìm ra ngưỡng năng lượng chuẩn chính xác hơn giúp giảm sai số của thuật toán.

# TÀI LIỆU THAM KHẢO

1. Matthieu Hodgkinson, “CS425 Audio and Speech Processing”, National University of Ireland, Maynooth, pp.30-48, 2012.
2. Link: https://vi.wikipedia.org/wiki/Ph%C3%A2n\_ph%E1%BB%91i\_chu%E1%BA%A9n