A white number on a grey background

Description automatically generated

**PHÂN TÍCH TÍN HIỆU TIẾNG NÓI**

**VÀ ỨNG DỤNG XỬ LÝ TIẾNG NÓI**

**Họ và tên SV báo cáo 1: MSSV:**

**Họ và tên SV báo cáo 2: MSSV:**

**Họ và tên SV báo cáo 3: MSSV:**

**Nhóm lớp Tiểu nhóm: Ngày thí nghiệm:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Điểm đánh giá** | | | | **CBGD nhận xét và ký tên** |
| **Chuẩn bị lý thuyết** | **Báo cáo và kết quả TN** | **Kiểm tra** | **Kết quả** |  |

# **CHUẨN BỊ THÍ NGHIỆM**

* 1. Bên cạnh nội dung tiếng nói (từ được phát âm), hãy liệt kê những thông tin khác có thể biết được khi người nghe nghe một đoạn tín hiệu tiếng nói.

Giới tính, độ tuổi, trạng thái cảm xúc, ngữ điệu, tốc độ nói

* 1. Khi nghe một đoạn tiếng nói, người nghe có thể biết được tiếng nói này là của nữ hay nam không? Giải thích.

Có thể phân biệt. Dựa trên các đặc trưng sinh lý của giọng nói như tần số giọng nói, âm sắc, độ vang, ngữ điệu và cách phát âm cũng như là cách thở và nhịp điệu. Tuy nhiên, cũng có những trường hợp ngoại lệ khi giọng nói không phản ánh đúng giới tính, ví dụ như ở những người có giọng nói trung tính hoặc khi có các yếu tố đặc biệt ảnh hưởng đến giọng nói.

* 1. Tín hiệu tiếng nói khi được truyền trên kênh truyền điện thoại, băng thông của nó sẽ được giới hạn trong khoảng tần số nào? Cho biết lý do của việc giới hạn băng thông.

Tín hiệu tiếng nói khi được truyền qua kênh điện thoại thường được giới hạn trong khoảng tần số từ 300 Hz đến 3400 Hz. Lý do của việc này là để tiết kiệm băng thông, đảm bảo khả năng tương thích giữa các thiết bị và mạng, cũng như giúp các thuật toán mã hóa và giải mã hoạt động hiệu quả hơn. Khoảng tần số này đủ để truyền tải giọng nói một cách rõ ràng và dễ hiểu, mặc dù không đạt được chất lượng âm thanh cao như trong âm nhạc.

* 1. Việc thu tiếng nói trong Matlab có thể được thực hiện bằng lệnh y = wavrecord(N, Fs, DataType). Hãy cho biết ý nghĩa các thông số y, N, Fs, DataType? Vì sao cần phải xác định 3 thông số này trong quá trình thu tiếng nói.

Lệnh y = wavrecord(N, Fs, DataType) trong Matlab được dùng để thu tiếng nói, với các thông số: y là biến lưu trữ dữ liệu âm thanh, N là số lượng mẫu cần ghi lại, Fs là tần số lấy mẫu, và DataType là kiểu dữ liệu của các mẫu âm thanh. Việc xác định ba thông số này giúp đảm bảo chất lượng âm thanh thu được phù hợp và tối ưu hóa việc lưu trữ và xử lý tín hiệu âm thanh.

* 1. Việc phát tiếng nói trong Matlab có thể được thực hiện bằng lệnh wavplay(y, Fs). Fs là tần số lấy mẫu của y. Hãy cho biết nếu nhập tần số lấy mẫu không đúng thì kết quả tiếng nói được phát âm sẽ như thế nào?

Nếu nhập tần số lấy mẫu (Fs) không đúng khi sử dụng lệnh wavplay(y, Fs) trong Matlab, kết quả tiếng nói được phát ra sẽ bị biến đổi so với âm thanh gốc. Cụ thể:

* **Nếu tần số lấy mẫu nhập thấp hơn thực tế**: Âm thanh sẽ được phát chậm hơn và có tần số thấp hơn, làm cho giọng nói trở nên trầm và kéo dài hơn.
* **Nếu tần số lấy mẫu nhập cao hơn thực tế**: Âm thanh sẽ được phát nhanh hơn và có tần số cao hơn, làm cho giọng nói trở nên cao và ngắn hơn.

Cả hai trường hợp đều dẫn đến việc âm thanh bị méo mó và không giống với giọng nói gốc. Việc sử dụng tần số lấy mẫu chính xác là rất quan trọng để đảm bảo chất lượng và độ trung thực của âm thanh khi phát lại.

* 1. Tín hiệu tiếng nói y có tổng số 1000 mẫu. Biết rằng y được thu tại tốc độ lấy mẫu Fs=16000Hz. Cho biết tín hiệu y được thu trong bao nhiêu mili giây?...........................................................
  2. Tín hiệu tiếng nói y trong câu 6 được biết đổi Fourier rời rạc (Discrete Fourier Transform) tại N=1024 điểm. Biết rằng độ rộng giữa hai vạch phổ kế tiếp nhau là Δf. Tính Δf.

Δf = Hz

* 1. Vì sao việc triệt nhiễu trong tín hiệu tiếng nói là cần thiết? Cho biết một số cách có thể được dùng để đánh giá của hiệu quả của quá trình triệt nhiễu trong tiếng nói

Việc triệt nhiễu trong tín hiệu tiếng nói là cần thiết để cải thiện chất lượng âm thanh, làm cho giọng nói rõ ràng và dễ hiểu hơn. Một số cách đánh giá hiệu quả triệt nhiễu bao gồm: tỷ số tín hiệu trên nhiễu (SNR), độ méo tiếng, chỉ số chất lượng chủ quan (đánh giá từ người nghe), chỉ số chất lượng khách quan (PESQ, POLQA), và phân tích phổ tín hiệu. Những phương pháp này giúp đảm bảo quá trình triệt nhiễu đạt được kết quả mong muốn.

* 1. Dựa vào đặc trưng nào để nhận biết đoạn tín hiệu thực sự có tiếng nói trong toàn bộ đoạn tín hiệu ghi âm được. Giải thích ngắn gọn.

Để nhận biết đoạn tín hiệu có tiếng nói trong toàn bộ đoạn tín hiệu ghi âm, ta dựa vào mức năng lượng cao, tần số và phổ tín hiệu đặc trưng, tỷ số tín hiệu trên nhiễu (SNR) cao và các đặc trưng thống kê như zero-crossing rate và pitch. Những yếu tố này giúp phân biệt tiếng nói với các đoạn không có tiếng nói.

# **TIẾN HÀNH THÍ NGHIỆM**

**2.1 Ước lượng băng thông và khảo sát sự phân bố năng lượng**

1. Sử dụng hàm wavread trong Matlab để đọc file tiếng nói ‘CleanSpeech.wav’, gán tín hiệu tiếng nói trong file này là biến sp. Xác định tần số lấy mẫu fs của tín hiệu tiếng nói lưu trong file này.

fs= ……16000………mẫu/giây

1. Biết rằng đoạn tiếng nói trên được thu trong khoảng thời gian t giây. Từ giá trị fs tìm được ở trên, hãy xác định t.

t= ………10.2327…………giây

1. Sử dụng lệnh wavplay, phát và nghe lại đoạn tiếng nói trên tại ba tần số lấy mẫu khác nhau là f1=0.5fs, f2=fs, f3=2fs. Cho nhận xét về đoạn âm thanh nghe được trong mỗi trường hợp.

At f1 = 0.5fs . The audio will play at a slower speed and lower pitch.

This is because the sampling frequency is halved, causing the playback to be twice as long and the pitch to drop.

At f2 = fs. The audio will play at the original speed and pitch.

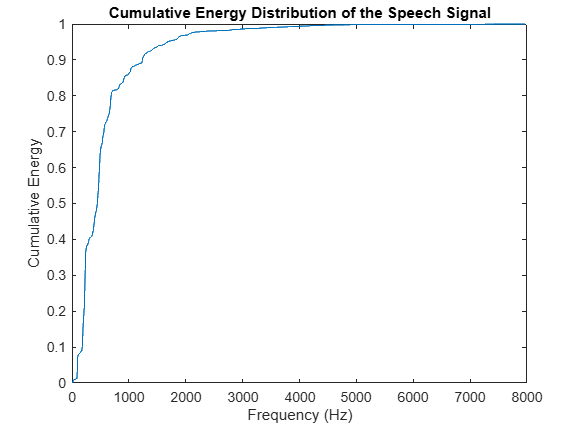
This is the normal playback scenario, where the sampling frequency matches the original recording frequency.

At f3 = 2fs. The audio will play at a faster speed and higher pitch.

Doubling the sampling frequency causes the playback to be twice as fast and the pitch to rise.

1. Sử dụng Matlab, vẽ dạng sóng tín hiệu tiếng nói sp trên với trục ngang biểu diễn theo đơn vị giây. Tính và vẽ phổ biên độ của sp trong khoảng tần số từ 0 đến fs/2 (sử dụng công cụ fft). Lưu ý: trục ngang của hình vẽ phổ biên độ phải được biểu diễn theo đơn vị Hz.

|  |
| --- |
| Dạng sóng tiếng nói |
| Phổ b iên độ của tiếng nói |

1. Dựa trên hình vẽ phổ biên độ này, hãy ước lượng băng thông của tiếng nói (độ rộng của đoạn tần số tính từ 0Hz, chứa khoảng hơn 95% năng lượng của tín hiệu tiếng nói).

Băng thông của tín hiệu tiếng nói là: BW = 1679 031 Hz

1. Dựa trên kết quả tính fft trong câu 6.1, hãy tính phần trăm năng lượng của tiếng nói phân bố trên các vùng tần số khác nhau có cùng độ rộng là 1kHz và hoàn thành Bảng 1.

Bảng 1 : Phần trăm năng lượng tiếng nói trên các băng tần khác nhau

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Băng Tần | | | | | | | |
| Băng tần | 0-1kHz | 1-2kHz | 2-3kHz | 3-4kHz | 4-5kHz | 5-6kHz | 6-7kHz | 7-8kHz |
| % năng lượng | 86.2595 | 10.6474 | 1.6921 | 0.78541 | 0.41226 | 0.05571 | 0.069496 | 0.078021 |

Cho biết vùng tần số nào chứa nhiều năng lượng tiếng nói nhất?...khoảng từ 0 đến 1 kHz............

1. Dựa vào kết quả trong Bảng 1 hãy giải thích vì sao tín hiệu tiếng nói truyền trên đường truyền điện thoại bị giới hạn tần số dưới 3.5kHz?

Kết quả trên cho thấy rằng phần lớn năng lượng của tín hiệu tiếng nói nằm trong các dải tần số thấp, cụ thể là:

0 - 1000 Hz: 86.2595% năng lượng

1000 - 2000 Hz: 10.6474% năng lượng

2000 - 3000 Hz: 1.6921% năng lượng

Tổng cộng, có 98.599% năng lượng của tín hiệu nằm trong khoảng từ 0 đến 3000 Hz. Giới hạn băng thông điện thoại dưới 3.5 kHz giúp tiết kiệm tài nguyên, đảm bảo tính tương thích, và duy trì chất lượng giọng nói đủ để giao tiếp rõ ràng.

**2.2 Xác định tần số lấy mẫu tối thiểu**

1. Từ kết quả trong câu 6.1, hãy ước lượng tần số lấy mẫu nhỏ nhất fsmin của tiếng nói để tín hiệu sau khi lấy mẫu không bị aliasing (chồng phổ).

fsmin=…16000……….mẫu/giây

1. Tín hiệu tiếng nói trong câu 6.1 có thể được giảm tần số lấy mẫu để tạo nên tín hiệu có tần số lấy mẫu nhỏ hơn theo dùng lệnh

spDownSampling=sp(1:N:end);

Trong đó N = 2, 3, 4…. là tốc độ giảm tần số lấy mẫu. Việc giảm tần số lấy mẫu của tín hiệu được thực hiện bằng cách trong mỗi đoạn N mẫu tín hiệu gốc liên tiếp nhau thì giữ lại một mẫu và bỏ đi N-1 mẫu còn lại để tạo thành tín hiệu mới.

1. Hãy cho biết lợi ích của việc giảm tần số lấy mẫu?

Giảm tần số lấy mẫu giúp tiết kiệm dung lượng lưu trữ, giảm băng thông truyền tải, tăng tốc độ xử lý và giảm yêu cầu về phần cứng. Điều này giúp tối ưu hóa việc lưu trữ và xử lý tín hiệu một cách hiệu quả hơn.

1. Hãy tiến hành giảm tần số lấy mẫu của tiếng nói gốc trong câu 6.1. Ứng với mỗi giá trị N, thực hiện các bước sau và hoàn thành Bảng 2.
2. Tính và vẽ phổ biên độ của tín hiệu tiếng nói spDownSampling trong khoảng tần số từ [0 fs/(2N)] Hz.
3. Vẽ dạng sóng tín hiệu tiếng nói trong 4 mili giây đầu tiên dùng lệnh stem(spDownSampling)
4. Nghe tín hiệu tiếng nói sau khi giảm tốc độ lấy mẫu theo lệnh sau: wavplay(spDownSampling,fs/N)

|  |  |
| --- | --- |
| Phổ biên độ tiếng nói với N=1 | Dạng sóng tiếng nói với N=1 |
| Phổ biên độ tiếng nói với N=2 | Dạng sóng tiếng nói với N=2 |
| Phổ biên độ tiếng nói với N=3 | Dạng sóng tiếng nói với N=3 |
| Phổ biên độ tiếng nói với N=4 | Dạng sóng tiếng nói với N=4 |
| Phổ biên độ tiếng nói với N=8 | Dạng sóng tiếng nói với N=8 |

Bảng 2: Giảm tần số lấy mẫu

|  |  |  |  |
| --- | --- | --- | --- |
| **Tốc độ giảm tần số lấy mẫu** | **Tần số lấy mẫu mới (kHz)** | **Tổng số mẫu của spDownSampling** | **Mức độ aliasing** |
| N = 1 | 16 kHz | 163 723 | No aliasing |
| N = 2 | 8 kHz | 81 862 | Low aliasing |
| N = 3 | 5.33 kHz | 54 575 | High aliasing |
| N = 4 | 4 kHz | 40 931 | High aliasing |
| N = 8 | 2 kHz | 20 466 | High aliasing |

1. Trong các giá trị N đã chọn trong Bảng 2, bắt đầu từ giá trị nào thì hiện tượng aliasing xảy ra? Giải thích nguyên nhân.

Hiện tượng aliasing bắt đầu xảy ra từ giá trị N = 3 với tần số lấy mẫu mới Fs = 5.33 kHz.

**Nguyên nhân:** Theo định lý Nyquist, để tránh hiện tượng aliasing, tần số lấy mẫu phải ít nhất gấp đôi tần số cao nhất của tín hiệu. Trong trường hợp này, tần số cao nhất của tín hiệu là 8000 Hz. Vậy tần số lấy mẫu nhỏ nhất phải là:

fsmin=2×8000Hz=16000Hz

Với giá trị N = 1 (Fs = 16.00 kHz) và N = 2 (Fs = 8.00 kHz), tần số lấy mẫu vẫn cao hơn hoặc bằng tần số Nyquist, nên không xảy ra aliasing hoặc chỉ xảy ra aliasing ở mức độ thấp.

Tuy nhiên, khi N = 3 với Fs = 5.33 kHz, tần số lấy mẫu thấp hơn tần số Nyquist, dẫn đến hiện tượng aliasing. Điều này có nghĩa là các thành phần tần số cao hơn một nửa tần số lấy mẫu (tức là 2.665 kHz) sẽ bị chồng phổ và gây ra hiện tượng aliasing.

Các giá trị N = 4 và N = 8 đều có tần số lấy mẫu thấp hơn nhiều so với tần số Nyquist, do đó hiện tượng aliasing sẽ xảy ra với mức độ cao.

**2.3 Triệt nhiễu nâng cao chất lượng tiếng nói**

Tín hiệu tiếng nói trong câu 6.1 bị ảnh hưởng của nhiễu cộng là một tín hiệu SIN tần số cao. Tiếng nói bị ảnh hưởng bởi nhiễu được lưu trong file ‘NoisySpeech.wav’.

Sử dụng hàm wavread trong Matlab để đọc file tiếng nói bị tác động bỡi nhiễu, gán tín hiệu tiếng nói trong file này là biến spNoisy. Sinh viên được yêu cầu ứng dụng bộ lọc để triệt nhiễu nhằm mục đích nâng cao chất lượng tiếng nói.

a. Tính và vẽ phổ biên độ của tín hiệu spNoisy trong khoảng tần số từ 0 đến fs/2. Lưu ý: trục ngang của hình vẽ phổ biên độ phải được biểu diễn theo đơn vị Hz. So

sánh với phổ của tín hiệu tiếng nói chưa bị tác động của nhiễu (sp) trong câu 6.1

để xác định tần số của tín hiệu nhiễu fNoise.

|  |
| --- |
| Phổ biên độ của tín hiệu tiếng nói bị nhiễu |

fNoise=……6697.9881….Hz

Tín hiệu spNoisy được truyền qua bộ lọc như Hình 8 nhằm triệt nhiễu để nâng cao chất lượng tiếng nói. Hãy cho biết để khử nhiễu hiệu quả, bộ lọc triệt nhiễu cần có đặc tính gì?

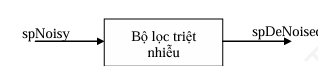
Loại bộ lọc (thông cao, thấp, dải, chắn dải)? thông thấp.......................................................

Tần số cắt của bộ lọc: 6500 Hz

Giải thích

Loại bộ lọc: Chúng ta sử dụng bộ lọc thông thấp vì nhiễu là tín hiệu SIN tần số cao. Bộ lọc thông thấp sẽ giúp loại bỏ các thành phần tần số cao hơn tần số cắt, trong khi giữ lại các thành phần tần số thấp hơn tần số cắt, là phần chính của tín hiệu tiếng nói.

Tần số cắt: Tần số cắt được chọn gần với tần số nhiễu (6697.9881 Hz), nhưng thấp hơn một chút (khoảng 6500 Hz) để đảm bảo loại bỏ hiệu quả nhiễu mà không làm mất mát nhiều các thành phần quan trọng của tín hiệu tiếng nói.



**Hình 48. Bộ lọc triệt nhiễu**

b. Sử dụng 03 bộ lọc thông thấp Chebyshev loại 1 khác nhau để triệt nhiễu trong tín hiệu spNoisy. Cả ba bộ lọc này đều có tần số cắt fc=4kHz và độ gợn dải thông Rp=0.2dB. Bậc của bộ lọc thay đổi như trong Bảng 3.

Sử dụng lệnh cheby1 trong Matlab để xác định các hệ số B và A của các bộ lọc 1, 2 và 3. Sau đó dùng lệnh filter để thực hiện việc lọc nhiễu trong tín hiệu spNoisy. Hãy thực hiện các cách sau để đánh giá hiệu quả của quá trình triệt nhiễu và hoàn thành Bảng 3.

1. Nghe tín hiệu trước và sau khi triệt nhiễu.
2. Vẽ và so sánh dạng sóng của tín hiệu trước và sau khi triệt nhiễu.
3. Vẽ và so sánh phổ biên độ của tín hiệu tiếng nói trước và sau khi triệt nhiễu.
4. Tính thông số Tỷ số tín hiệu trên nhiễu được ký hiệu là SNR (Signal to Noise Ratio) theo công thức sau:



Trong đó: *SpeechPower* là công suất tín hiệu tiếng nói chưa bị ảnh hưởng bởi nhiễu (sp). *NoisePower* là công suất của nhiễu. Nhiễu được xác định là hiệu số của tín hiệu sau khi xử lý và tín hiệu trước khi xử lý (sp). Dễ hiểu là SNR càng lớn thì chất lượng tín hiệu tiếng nói càng tốt. Lưu ý: Công suất tín hiệu có thể được tính là tổng bình phương tất cá các mẫu của tín hiệu đó. . Trong đó s[k] là biên độ của mẫu tín hiệu thứ k trong tổng số L mẫu của tín hiệu s.

Xác định SNR của tín hiệu tiếng nói trước khi triệt nhiễu spNoisy

= 0……….dB

Bảng 3: So sánh hiệu quả các bộ lọc triệt nhiễu

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Bậc bộ lọc** | **SNR sau khi triệt nhiễu (dB)** | **Chất lượng tiếng nói sau khi triệt nhiễu** |
| Bộ lọc 1 | N = 2 | 3.6577 | Cải thiện được chát lượng |
| Bộ lọc 2 | N = 4 | 3.5044 | Cải thiện nhưng ít hơn Bộ 1 |
| Bộ lọc 3 | N = 8 | 0.75035 | Cải thiện ít nhất |

c. Hãy vẽ dạng sóng và phổ biên độ của sp, spNoisy và spDeNoised hình vẽ sau.

|  |  |
| --- | --- |
| Dạng sóng tiếng nói gốc (chưa bị nhiễu) | Phổ biên độ tiếng nói gốc |
| Dạng sóng tiếng nói bị nhiễu | Phổ biên độ tiếng nói bị nhiễu |
| Dạng sóng tiếng nói sau triệt nhiễu dùng Bộ lọc 1 | Phổ biên độ tiếng nói sau triệt nhiễu dung Bộ lọc 1 |
| Dạng sóng tiếng nói sau triệt nhiễu dùng Bộ lọc 2 | Phổ biên độ tiếng nói sau triệt nhiễu dung Bộ lọc 2 |
| Dạng sóng tiếng nói sau triệt nhiễu dùng Bộ lọc 3 | Phổ biên độ tiếng nói sau triệt nhiễu dung Bộ lọc 3 |

d. Cho nhận xét và giải thích về hiệu quả triệt nhiễu của các bộ lọc 1, 2, 3.

Bộ lọc 1 (bậc 2) hiệu quả nhất khi cải thiện SNR từ 7.7666e-07 dB lên 3.6577 dB, nhờ vào việc ít gây biến dạng và giữ lại phần lớn tín hiệu gốc. Bộ lọc 2 (bậc 4) cũng cải thiện đáng kể với SNR đạt 3.5044 dB, tuy nhiên, sự phức tạp hơn của nó có thể gây ra biến dạng nhẹ. Bộ lọc 3 (bậc 8) kém hiệu quả nhất, với SNR chỉ đạt 0.75035 dB, do đặc tính phức tạp của bộ lọc này dẫn đến việc gây ra nhiều biến dạng hơn, làm giảm chất lượng tín hiệu.

**2.4 Tách tiếng nói ra khỏi đoạn tín hiệu thu được**

a. Hãy phân đoạn đoạn tín hiệu tiếng nói sp trong mục 6.1 theo hướng dẫn trong Mục 4 và hãy cho biết tổng số đoạn tiếng nói nhận được là …19…………..đoạn.

b. Viết một đoạn chương trình tính Năng lượng từng đoạn tiếng nói và vẽ thông số này trên cùng màn hình với dạng sóng tiếng nói. Lưu ý: Hình vẽ dạng sóng tiếng nói và Năng lượng phải được sắp xếp thẳng hàng. Trục ngang hình vẽ dạng sóng tiếng nói biểu diễn theo chỉ số mẫu, trục ngang của hình vẽ năng lượng biểu diễn theo chỉ số đoạn.

|  |
| --- |
| Dạng sóng tiếng nói |
| Năng lượng tiếng nói |

Từ kết quả nhận được hãy cho biết sự thay đổi của Năng lượng ứng với các vùng tiếng nói khác nhau

Vùng tiếng nói

Các vùng có tiếng nói thường có năng lượng cao hơn do sự hiện diện của âm thanh phát ra từ giọng nói. Trong các đoạn này, năng lượng sẽ tăng đáng kể, thể hiện qua đỉnh trong biểu đồ năng lượng.

Vùng không tiếng nói (im lặng hoặc nền nhiễu)

Các vùng không có tiếng nói hoặc chỉ có nhiễu nền thường có năng lượng thấp hơn nhiều. Năng lượng của các đoạn này sẽ giảm đáng kể, thể hiện qua các vùng thấp trong biểu đồ năng lượng.

Giao điểm giữa các vùng

Tại các điểm chuyển tiếp giữa vùng có tiếng nói và vùng không tiếng nói, năng lượng sẽ có sự thay đổi đột ngột. Điều này có thể thể hiện qua các sự tăng hoặc giảm nhanh chóng trong biểu đồ năng lượng.

c. Tiếng nói sau khi thu được sẽ bao gồm hai phần, phần có tín hiệu tiếng nói và phần không có tín hiệu tiếng nói (tương ứng với khoảng thời gian yên lặng trong quá trình phát âm). Việc trích đặc trưng trong các ứng dụng xử lý tiếng nói được thực hiện trong đoạn có tiếng nói. Vì vậy, việc nhận biết và tách đoạn tiếng nói ra khỏi đoạn tín hiệu thu được là cần thiết. Quá trình này được gọi là phát hiện tiếng nói (Voice Activity Detection VAD).

Từ sự quan sát về sự thay đổi của Năng lượng trong phần 6.4b, hãy đề nghị một giải thuật tách đoạn tín hiệu tiếng nói ra khỏi đoạn tín hiệu thu được sử dụng Năng lượng. Viết đoạn chương trình thực hiện giải thuật này bằng Matlab.

Giải thuật tách đoạn tín hiệu tiếng nói ra khỏi đoạn tín hiệu ghi âm được:

**Giải thuật:**

1. **Tính Năng lượng**: Tính năng lượng của từng đoạn tín hiệu như đã làm trong phần trước.
2. **Xác định ngưỡng**: Chọn một ngưỡng năng lượng để phân biệt giữa đoạn có tiếng nói và đoạn im lặng/nhiễu nền.
3. **Tách đoạn tiếng nói**: Duyệt qua các đoạn tín hiệu, nếu năng lượng của đoạn lớn hơn ngưỡng, coi đoạn đó là đoạn có tiếng nói.

Đoạn chương trình thực hiện VAD bằng Matlab

% Read the speech signal

[sp, fs] = audioread('CleanSpeech.wav');

% Window size (number of samples per segment)

windowSize = 8192; % Adjust this size as needed

% Number of segments

numSegments = floor(length(sp) / windowSize);

% Initialize the energy array

energy = zeros(1, numSegments);

% Calculate the energy of each segment

for i = 1:numSegments

segment = sp((i-1)\*windowSize + 1:i\*windowSize);

energy(i) = sum(segment.^2);

end

% Determine the energy threshold

energyThreshold = 0.01 \* max(energy);

% Initialize the array to store the speech segments

speechSegments = [];

% Extract the speech segments based on energy threshold

for i = 1:numSegments

if energy(i) > energyThreshold

speechSegments = [speechSegments; sp((i-1)\*windowSize + 1:i\*windowSize)];

end

end

Kết quả thực hiện

|  |
| --- |
| Dạng sóng của toàn bộ tín hiệu ghi âm |
| Dạng song tiếng nói tách được |

**2.5 Nhận dạng giới tính**

Lệnh pitch\_rapt được dùng để trích F0 trên tất cả các đoạn tiếng nói (chiều dài 25ms và chồng lấn 15ms giữa hai đoạn kế tiếp). Lệnh này được dùng như sau: [mf0]=pitch\_rapt(s,fs) trong đó mf0 là vector chứa giá trị F0 trên các đoạn tiếng nói của trong tín hiệu s, fs là tần số lấy mẫu của s.

a. Dùng lệnh pitch\_rapt hãy trích và vẽ F0 của 02 file tiếng nói, 1 file tiếng nói của nữ (‘Female.wav’), 1 file tiếng nói của nam (‘Male.wav’). Tính F0 trung bình của tiếng nói người nữ Mean\_F0nữ và người nam Mean\_F0nam. Lưu ý rằng F0 của những đoạn không có tiếng nói không xác định được và pitch\_rapt sẽ gán giá trị F0=0 cho những đoạn này. F0 trung bình chỉ tính cho những đoạn có F0>0.

|  |
| --- |
| Hình vẽ F0 của nữ, file ‘Female.wav’ |
| Hình vẽ F0 của nam, file ‘Male.wav’ |

Mean\_F0nữ =……172.0219……Hz

Mean\_F0nam =……98.6187……Hz

Nhận xét về khoảng dao động và giá trị trung bình của F0 người nữ và người nam.

Giọng nữ thường có giá trị F0 cao hơn và khoảng dao động lớn hơn so với giọng nam. Các giá trị trung bình và khoảng dao động thu được trong phân tích này là hợp lý và phù hợp với các đặc điểm sinh lý của giọng nói nam và nữ. Các giá trị F0 cao không nằm ngoài dự đoán và có thể phản ánh các đỉnh cao trong giọng nói cụ thể hoặc đặc thù của người nói.

b. Sinh viên được cung cấp 10 file tiếng nói của 10 người nói (speaker) lưu dưới dạng file .wav, thực hiện các bước sau và hoàn thành Bảng 4.

i. Hãy nghe 10 file tiếng nói này và sau khi nghe xác định giới tính người nói là nam hay nữ. Điền kết quả vào hàng (1).

ii. Sử dụng lệnh pitch\_rapt hãy xác định viết một đoạn chương trình để tính F0 trung bình của từng file tiếng nói. Từ đó so sánh với ngưỡng 150Hz để xác định giới tính người phát âm. Điền kết quả vào hàng (3) và (4).

Bảng 4: Kết quả nhận dạng giới tính

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Speaker | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1. Nghe | Nữ | Nữ | Nam | Nam | Nam | Nữ | Nữ | Nam | Nữ | Nam |
| 1. Kết luận | Nữ | Nữ | Nam | Nam | Nam | Nữ | Nữ | Nam | Nữ | Nam |
| 1. F0 (Hz) | 172.02 | 240.78 | 98.619 | 101.08 | 127.2 | 218.75 | 159.63 | 78.125 | 182.69 | 111.98 |
| 1. Chương trình | Nữ | Nữ | Nam | Nam | Nam | Nữ | Nữ | Nam | Nữ | Nam |
| 1. Kết luận | Nữ | Nữ | Nam | Nam | Nam | Nữ | Nữ | Nam | Nữ | Nam |

Đối chiếu kết quả nhận được với Giáo viên hướng dẫn để biết kết quả nhận dạng theo mỗi phương pháp là đúng hay sai và ghi vào ô tương ứng trong hàng (2)&(5) Bảng 4. Độ chính xác của quá trình nhận dạng theo phần trăm được tính như sau:

*Độ chính xác (%) = (Số lần nhận dạng đúng/Tổng số lần nhận dạng)\*100.*

c. Tính độ chính xác việc nhận dạng giới tính 10 file cho sẵn bằng cách lắng nghe …100…….%

Tính độ chính xác việc nhận dạng giới tính bằng chương trình Matlab……100…….%

d. Sử dụng wavread, đọc và thu vào máy tính dòng chữ: “Bộ môn Viễn thông, Khoa Điện, Trường Đại học Bách khoa Thành phố Hồ Chí Minh”. Tốc độ lấy mẫu fs=16kHz, datatype=’double’, gán tín hiệu thu được vào biến s trong Matlab. Dùng pitch\_rapt hãy tính F0 trung bình của chuỗi tiếng nói này. Hãy lặp lại việc này 5 lần và ghi nhận kết quả vào phần sau:

Giới tính của bạn………Nam…………

Lần 1: F01=………153.0296………Hz

Lần 2: F02=………136.9717………Hz

Lần 3: F03=………139.0295………Hz

Lần 4: F04=………134.0229………Hz

Lần 5: F05=…………134.0794……Hz

F0TrungBình=………139.4266………Hz

**2.6 Phần mở rộng**

Lưu ý: Phần này không bắt buộc sinh viên thực hiện. Nếu hoàn thành đúng phần này, sinh viên được cộng 2 điểm vào bài báo cáo.

Sử dụng Matlab, hãy viết một đoạn chương trình để khi người đọc đọc vào máy tính một đoạn tiếng nói (chậm và rời rạc), chương trình sẽ thực hiện một số yêu cầu sau:

a. Xác định tổng số từ trong đoạn tiếng nói thu vào.

b. Tách tiếng nói của mỗi từ riêng biệt và lưu vào các file ‘.wav’. Tên file sẽ là số thứ tự của từ tương ứng trong đoạn tiếng nói. Ví dụ ‘1.wav’, ‘2.wav’…

c. Tính chiều dài của tiếng nói mỗi từ theo đơn vị mili giây và chiều dài trung bình của tất cả các từ đọc vào