ĐIỀU KHIỂN ROBOT PIONEER P3-DX BẰNG TIẾNG NÓI VỚI ĐẶC TRƯNG MFCC VÀ GIẢI THUẬT NAÏVE BAYES NEAREST NEIGHBORS

Mã Trường Thành[1], Đỗ Thanh Nghị[2], Phạm Nguyên Khang[2], Châu Ngân Khánh[3]

[1]Khoa Kỹ thuật – Công nghệ, Trường CĐCĐ Sóc Trăng
[2]Khoa CNTT&TT, Trường Đại học Cần Thơ
[3]Trường Đại học An Giang

truongthanh1511@gmail.com,dtnghi@cit.ctu.edu.vn

TÓM TẮT *- Trong bài báo này, chúng tôi trình bày ý tưởng điều khiển robot Pioneer P3-DX bằng tiếng nói theo thời gian thực với giải thuật Naïve Bayes Nearest Neighbor (NBNN) sử dụng đặc trưng MFCC (Mel-scale Frequency Cepstral Coefficient). Tập dữ liệu cho quá trình huấn luyện và nhận dạng là các mẫu tiếng nói tương ứng với các lệnh điều khiển robot được thu âm từ 20 người đọc khác nhau. Bước xử lý tiếp theo là thực hiện rút trích 39 đặc trưng MFCC từ mỗi mẫu âm thanh của tập dữ liệu thu được. Chúng tôi đề xuất sử dụng giải thuật máy học NBNN để nhận dạng trực tiếp các tiếng nói là các lệnh điều khiển hoạt động robot từ các đặc trưng MFCC tương ứng không cần bất kỳ thao tác xử lý trung gian nào khác. Kết quả thực nghiệm cho thấy rằng phương pháp đề xuất (NBNN sử dụng đặc trưng MFCC) có thể nhận dạng chính xác tiếng nói là các lệnh điều khiển robot, đáp ứng thời gian thực. Giải thuật NBNN cho độ chính xác trong nhận dạng là 98.5%, cao hơn khi so sánh với giải thuật (Support vector machines - máy học véctơ hỗ trợ) SVM và mô hình túi từ với độ chính xác tương ứng là 97.14%, giải thuật (Dynamic time warping – xoắn thời gian động) DTW có độ chính xác tương ứng là 98.4%, và (Hidden Markov model - mô hình Markov ẩn) HMM có độ chính xác là 97.8%. Hơn nữa, phương pháp NBNN sử dụng MFCC đơn giản và có thời gian thực hiện nhanh hơn, đáp ứng được yêu cầu điều khiển robot thời gian thực.*

Từ khóa - *Nhận dạng âm thanh, Đặc trưng MFCC, Naive Bayes Nearest Neighbor, Điều khiển robot Pioneer P3-DX.*

I. GIỚI THIỆU

Nghiên cứu điều khiển robot là bài toán được các nhà khoa học quan tâm, nhằm phục vụ cho con người trong nhiều lĩnh vực ứng dụng như: robot khuân hàng hóa, robot dọn rác, lau nhà, đá bóng, dẫn dường, robot trong công nghiệp ôtô, thăm dò khai thác mỏ, robot thợ lặn, v.v. Do có tính ứng dụng cao nên các nhà nghiên cứu đã bắt tay vào phát triển robot thông minh hơn, phục vụ tốt cho nhu cầu phát triển kinh tế - xã hội. Để làm được điều đó, cần có sự kết nối giữa "bộ não thông minh" và robot để tạo nên những robot thông minh như ngày nay. Máy học chính là nền tảng giúp robot có thể thông minh, hoạt động tinh vi hơn. Trong các hướng nghiên cứu về lĩnh vực robot, điều khiển robot thông qua nhận dạng hình ảnh và nhận dạng tiếng nói là một trong những lĩnh vực được các nhiều nhà nghiên cứu quan tâm rất nhiều do tính khả thi và khả năng ứng dụng trong thực tiễn.

Hiện nay các nghiên cứu liên quan đến nhận dạng tiếng nói đã được thực hiện trên nhiều hướng phát triển, mục tiêu khác nhau và đạt hiệu quả cao. Tiêu biểu là hệ thống Desktop Via Voice của IBM hay hệ thống Speed Recognition Engine của Microsoft và bộ công cụ HTK dựa trên mô hình Markov ẩn của Đại học Cambridge hay Đại học Mellon với CMU Sphinx. Những sản phẩm *(công cụ)* đã được áp dụng nhiều trong thực tế, nhận dạng và xử lý âm thanh. Hệ thống nhận dạng tiếng nói bao gồm hai bước chính [5], [6], [7]: rút trích và biểu diễn đặc trưng, huấn luyện mô hình máy học nhận dạng. Rút trích và biểu diễn đặc trưng tín hiệu âm thanh thường được sử dụng [6] là MFCC (Mel-scale Frequency Cepstral Coefficient), LPC (Linear Prediction Coefficients), FFT (Fast Fourier Transform). Mô hình máy học thường được sử dụng có thể là mạng nơron nhân tạo [10], [12], mô hình Markov ẩn HMM [16], [20].

Các nghiên cứu trong thời gian gần đây [11], [15], [19] tập trung vào sử dụng đặc trưng MFCC [7] đạt được hiệu quả cao. Nhóm tác giả trong [11] đề xuất rút trích đặc trưng MFCC từ âm thanh, biểu diễn các đặc trưng MFCC theo mô hình túi từ với hỗ trợ của giải thuật gom cụm *k*means [13], huấn luyện mô hình máy học véctơ hỗ trợ SVM [18] để nhận dạng âm thanh. Nghiên cứu của [15], [19] cũng thực hiện rút trích đặc trưng MFCC nhưng sử dụng giải thuật xoắn thời gian động DTW (Dynamic Time Warping) để nhận dạng trực tiếp tiếng nói.

Trong bài viết này, chúng tôi đề xuất hệ thống nhận dạng tiếng nói để điều khiển robot Pioneer P3-DX theo thời gian thực. Hệ thống thực hiện rút trích đặc trưng âm thanh MFCC, không cần qua bước tiền xử lý và biểu diễn phức tạp, hệ thống sử dụng giải thuật NBNN (Naïve Bayes Nearest Neighbor [2]) để nhận dạng trực tiếp tiếng nói là các lệnh điều khiển. Kết quả thử nghiệm trên tập dữ liệu thu thập từ 20 người nói khác nhau cho thấy đề xuất của chúng tôi đạt được độ chính xác đến 98.5% nhưng vẫn đáp ứng được về thời gian nhận dạng để điều khiển robot theo thời gian thực.

Phần tiếp theo của bài báo được tổ chức như sau: Phần II giới thiệu về robot Pioneer P3-DX. Phần III trình bày việc điều khiển robot Pioneer P3-DX bằng tiếng nói thông qua giải thuật NBNN với đặc trưng MFCC được rút trích để nhận dạng. Phần IV trình bày kết quả thực nghiệm cũng như cách di chuyển của robot và khoảng cách thông qua Sonar và Laser tương ứng với vận tốc điều khiển robot trước khi kết luận và hướng phát triển được trình bày trong phần V.

II. SƠ LƯỢC VỀ ROBOT PIONEER P3-DX

Robot được sử dụng trong bài báo này là loại robot di động của hãng Adept Mobile Robot với dòng Pioneer P3-DX. Robot Pioneer [22] là một dòng sản phẩm robot được nhiều nhà chuyên gia, các nghiên cứu đánh giá cao và được

sử dụng phổ biến trong các nghiên cứu robot hiện nay. Robot này được tạo ra và cho phép người nghiên cứu hoàn toàn có thể "lập trình được".

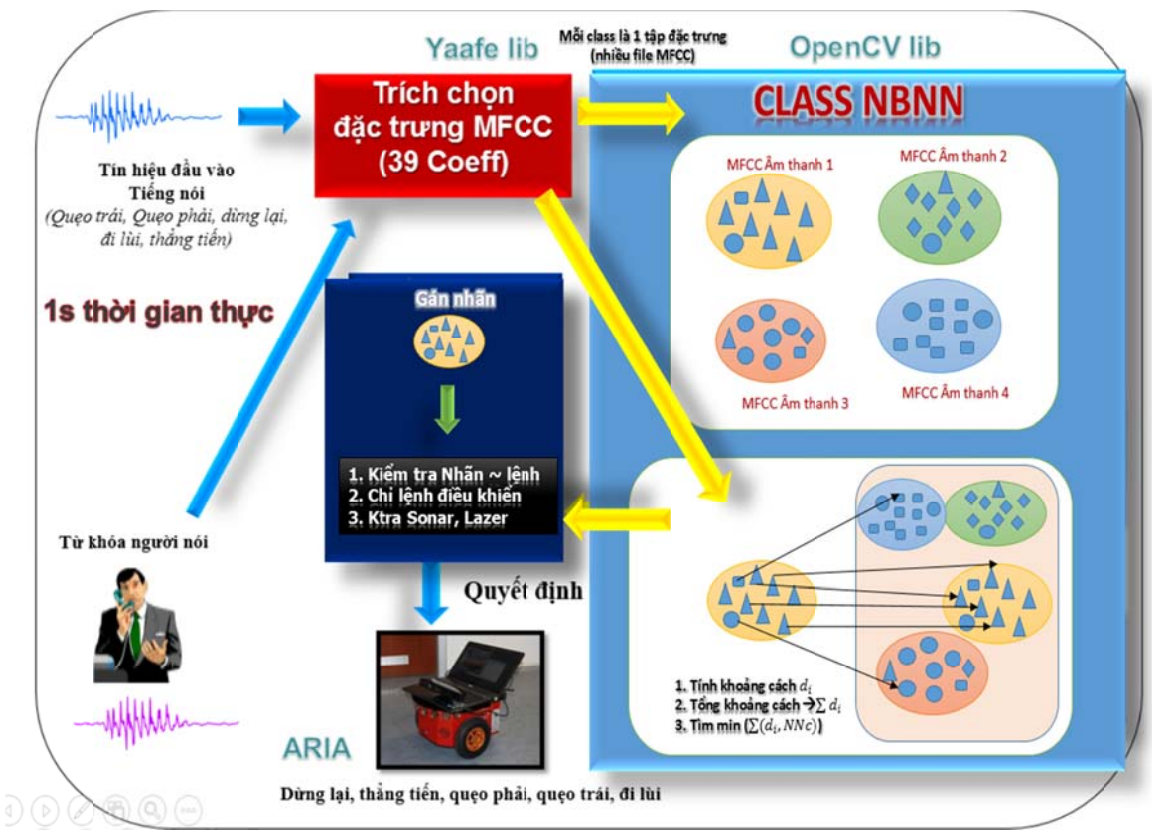
Robot Pioneer P3-DX [22] là loại robot di động nhỏ, trọng lượng nhẹ với 3 bánh xe (2 bánh trước chủ động còn gọi là bánh chính và bánh phụ sau di chuyển tự do, thực hiện nhiệm vụ cân bằng và rẽ), bánh xe có đường kính 19.5cm, thân được bao bọc bởi kim loại nhôm cứng cáp. Phía trước được trang bị 8 cảm biến Sonar – đây là dạng cảm biến siêu âm để dò tìm vật cản. Robot Pioneer P3-DX có 3 pin nhằm dự trữ và có thể thay đổi nhanh chóng.

Vì khả năng có thể lập trình được nên hãng Adept Mobile Robot đã hỗ trợ bộ SDK (software development kit) dành cho những nhà nghiên cứu robot, các loại máy tính có thể dễ dàng cài đặt và tích hợp vào để robot có thể hoạt động nhịp nhàng.

Cốt lõi của SDK mà robot Pioneer hoạt động chính là gói thư viện ARIA[21], đây là gói thư viện để phát triển phần mềm cho tất cả các nền tảng và các thiết bị, gói thư viện khá linh hoạt giúp cho nhiều nhà phát triển dễ dàng lựa chọn ngôn ngữ phù hợp cho mình với 3 ngôn ngữ hỗ trợ: C++, Java hoặc Python, gói thư viện có thể được phát triển trên đa nền (Windows và Linux).

Gói thư viện Aria sử dụng gcc trên Linux và Visual C++ trên Window để thực hiện viết các ứng dụng cũng như biên dịch gói thư viện. Aria là một gói thư viện mã nguồn mở được đính kèm theo giấy phép của GNU GPL (General Public License). Robot Pioneer P3-DX có thể di chuyển tới và di chuyển lui cũng như di chuyển rẽ với vận tốc tối đã là 1.2 m/s và có tải trọng đồ vậy lên đến 22kg.

Hình 1. Robot Pionner P3-DX [22]



Hình 2. Đề xuất mô hình hoạt động điều khiển robot Pioneer P3-DX

III. ĐIỀU KHIỂN ROBOT PIONEER P3-DX BẰNG TIẾNG NÓI

Xây dựng một hệ thống nhận dạng tiếng nói bao gồm hai bước chính [5], [6], [7]: rút trích và biểu diễn đặc trưng, huấn luyện mô hình máy học nhận dạng. Trong bài báo này, chúng tôi đề xuất mô hình nhận dạng dạng âm

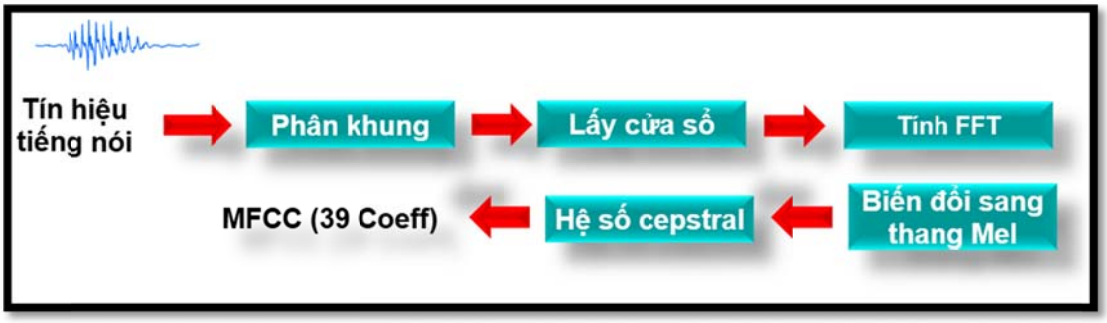
thanh trực tiếp bằng giải thuật NBNN [2] sử dụng đặc trưng MFCC để thực hiện điều khiển robot theo các chỉ thị lệnh âm thanh thu thập như mô tả trong hình 2. Để chuẩn bị cho pha huấn luyện và nhận dạng tiếng nói (chỉ thị lệnh âm thanh), trước hết chúng tôi tiến hành thu âm từ 20 người đọc khác nhau, mỗi chỉ thị lệnh âm thanh tương ứng với một lệnh. Có 5 lệnh điều khiển cơ bản là: thẳng tiến, đi lùi, quẹo trái, quẹo phải, dừng lại và lệnh khác (không thuộc 5 lệnh cơ bản).

Tín hiệu âm thanh ngoài đời thực là tín hiệu liên tục, hay tính hiệu tương tự trước khi thực hiện bất cứ bước xử lý nào, tín hiệu âm thanh cần được số hóa. Việc này được thực hiện tự động bởi các thiết bị thu âm, bằng cách lấy mẫu tín hiệu đầu vào. Như vậy, một tín hiệu âm thanh bất kỳ khi đã được đưa vào máy tính, là một tập các mẫu liên tiếp nhau, mỗi mẫu là giá trị biên độ của tín hiệu tại một thời điểm nhất định. Một tham số quan trọng trong việc lấy mẫu tín hiệu âm thanh là tần số lấy mẫu (Fs), là số mẫu được lấy trong một giây. Chúng tôi đề xuất sử dụng đặc trưng MFCC [7] để nghiên cứu cho hệ thống điều khiển này bởi vì MFCC là phương pháp trích đặc trưng được nhiều chuyên gia trong lĩnh vực âm thanh sử dụng và rất thành công trong việc nhận dạng tiếng nói.

3.1. Đặc trưng MFCC [7]

Kỹ thuật rút trích đặc trưng MFCC dựa trên việc thực hiện biến đổi để chuyển dữ liệu âm thanh đầu vào (đã được biến đổi Fourier dạng phổ) về thang đo tần số Mel, một thang đo diễn tả tốt hơn sự nhạy cảm của tai người đối với âm thanh. Kỹ thuật trích chọn đặc trưng này gồm các bước biến đổi liên tiếp, trong đó đầu ra của bước biến đổi trước sẽ là đầu vào của bước biến đổi sau. Đầu vào của quá trình đặc trưng này sẽ là một đoạn tín hiệu tiếng nói. Vì tín hiệu âm thanh sau khi được đưa vào máy tính đã được rời rạc hóa nên đoạn tín hiệu tiếng nói này bao gồm các mẫu liên tiếp nhau, mỗi mẫu là một giá trị thực, thể hiện giá trị biên độ của âm thanh tại một thời điểm nhất định.

Trong bài này, chúng tôi thực hiện lấy mẫu với tần số 16.000Hz (âm thanh nghe được), một đoạn mẫu với một số lượng nhất định tạo thành một frame, trích chọn đặc trưng MFCC cho ta tập đặc trưng cho mỗi frame tiếng nói. Kết quả là một tập trong trích chọn được sử dụng trong bài viết gồm 39 giá trị đặc trưng cho mỗi một frame tiếng nói. Trích xuất đặc trưng MFCC sẽ bao gồm 5 bước cơ bản như hình 3.



Hình 3. Trích đặc trưng MFCC

Bước 1: Phân khung (Frame Blocking)

Thực hiện chia tín hiệu đầu vào thành các đoạn nhỏ khoảng 20ms-30ms. Phân khung tín hiệu mỗi khung N mẫu, hai khung kề nhau lệch nhau M mẫu: $M=(1/2)N$ *(Biết N>M)*. Trong bước này, để hiệu quả cho âm thanh được nhận người ta thường tăng âm thanh (Pre-emphasis) trước khi thực hiện phân khung vì thực hiện tăng cường độ của những tần số cao lên nhằm làm tăng năng lượng ở vùng có tần số cao – vùng tần số của tiếng nói, một cách dễ hiểu là làm tiếng nói lớn hơn lên để ảnh hưởng của các âm thanh môi trường và nhiễu trở thành không đáng kể. Tăng cường độ tần số được thực hiện như công thức (1):

$$Y[n] = X[n] - 0.95 \times X[n-1] \quad (1)$$

Trong đó X là tín hiệu đầu vào trên từng khung mẫu; Y là cường độ tần số tăng (pre-emphasis after).

Bước 2: Lấy cửa sổ (Windowing)

Lấy cửa sổ nhằm giảm sự gián đoạn của tín hiệu ở đầu và cuối mỗi khung vừa được chia. Dùng cửa sổ Hamming (với $\alpha = 0.54$), theo công thức (2):

$$w[n] = \begin{cases} \propto -(1-\propto) cos\left(\frac{2\pi n}{L}\right) với\ 0 \le n \le N-1, N\ là\ số\ mẫu\ trong\ một\ frame \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad trường\ hợp\ khác \end{cases} \quad (2)$$

Trong đó: $w[n]$ là hệ số cho mẫu thứ n trong frame.

Trong loại cửa sổ Hamming, giá trị của tín hiệu sẽ giảm dần về 0 khi tiến dần ra hai biên của frame. Nói cách khác, nếu sử dụng cửa sổ Hamming để lấy ra các frame, năng lượng của mỗi frame sẽ tập trung ở giữa frame, một ưu điểm nữa là các giá trị biên của cửa sổ Hamming tiến dần về 0 sẽ làm bước biến đổi Fourier trở nên dễ dàng hơn.

Sử dụng một cửa sổ (window) chạy dọc tín hiện âm thanh và cắt ra các đoạn tín hiệu nằm trong cửa sổ đó. Một cửa sổ được định nghĩa bằng các thông số:

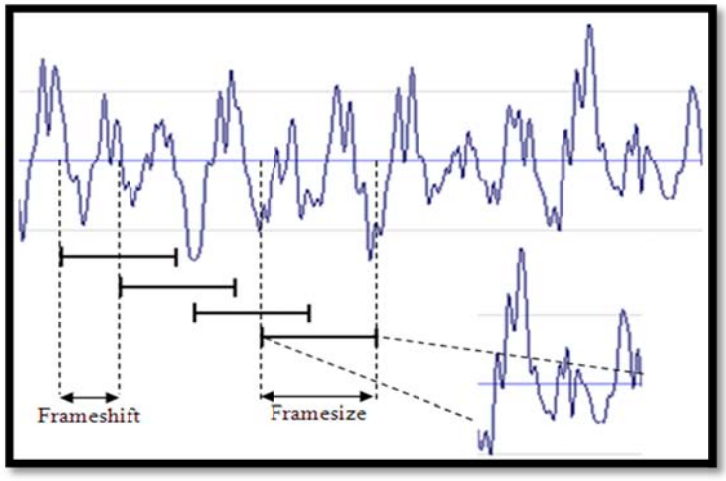
Frame size: độ rộng của cửa sổ, cũng là độ lớn của frame tín hiệu sẽ được cắt ra.

Frame shift: bước nhảy của cửa sổ, là độ dài đoạn mà cửa sổ sẽ trượt để cắt ra frame tiếp theo.

Mỗi frame sau đó sẽ được nhân với một hệ số, giá trị của hệ số này tùy thuộc vào từng loại cửa sổ.

$$Y[n] = w[n] * X[n] \quad (3)$$

Trong đó *X[n]* là giá trị của mẫu thứ *n* và *Y[n]* là giá trị của mẫu thứ *n* sau khi nhân với hệ số, *w[n]* là hệ số cho mẫu thứ *n* trong frame đó.



Hình 4. Chia cửa sổ *(Windowing)* dựa vào độ rộng và bước nhảy của cửa sổ

Bước 3: Biến đổi FFT (Fast Fourier Transform)

Bước biến đổi tiếp theo là thực hiện biến đổi Fourier rời rạc đối với từng mẩu tín hiệu đã được cắt ra. Qua phép biến đổi này, tín hiệu sẽ được đưa về không gian tần số.

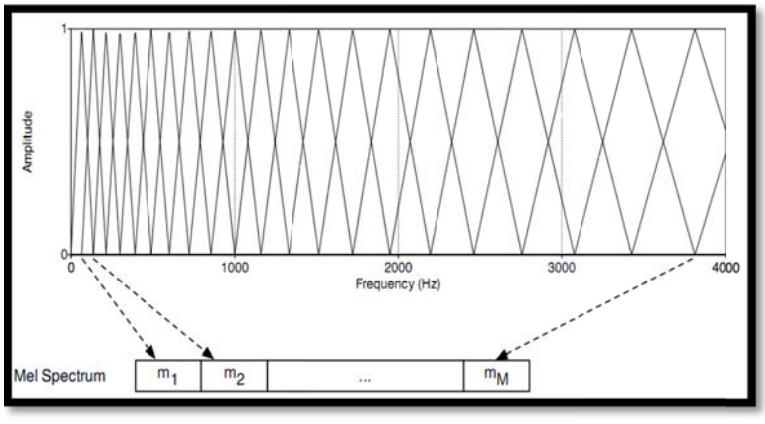
Công thức của biến đổi Fourier rời rạc như sau:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\frac{\pi}{N}kn} \quad (4)$$

Trong đó *x[n]* là giá trị của mẫu thứ *n* trong frame, *X[k]* là một số phức biểu diễn cường độ và pha của một thành phần tần số trong tín hiệu gốc, *N* là số mẫu trong một frame.

Tuy nhiên, để cải tiến cho chuyển đổi mỗi khung với *N* mẫu từ miền thời gian sang miền tần số được nhanh sẽ sử dụng FFT:

$$Y(\omega) = \int_{+\infty}^{-\infty} y(t) e^{-i\omega t} \quad (5)$$



Hình 5. Thang đo tần số Mel

Bước 4: Biến đổi sang thang đo Mel (Mel-frequency Wrapping)

Trong mô hình trích chọn đặc trưng MFCC, tần số sẽ được chuyển sang thang đo tần số Mel theo công thức:

$$f_{mel} = 2595 * \ln\left(1 + \frac{f}{700}\right) \quad (6)$$

Trong đó f là tần số ở thang đo thường, f_{mel} là tần số ở thang đo Mel. Người ta sử dụng các băng lọc để tính các hệ số Mel. Sử dụng bao nhiêu băng lọc thì sẽ cho ra bấy nhiêu hệ số Mel và các hệ số Mel này sẽ là đầu vào cho quá trình tiếp theo của trích chọn đặc trưng MFCC.

Bước 5: Hệ số Cepstrum (Cepstral Coefficients)

Bước tiếp theo của việc trích chọn đặc trưng MFCC là biến đổi Fourier ngược với đầu vào là các hệ số phổ Mel của bước trước, đầu ra sẽ là các hệ số cepstrum (MFCC – Mel Frequency Cepstrum Coefficients).

Kết quả của bước này là ta tính được hệ số MFCC theo công thức:

$$MFCC(i) = \frac{1}{Nfilters} \times \sum_{i=1}^{Nfilters} mfb(l) \times \cos(i(l - \frac{1}{2}) \times \frac{\pi}{Nfilter}) \quad (7)$$

Trích chọn đặc trưng MFCC sẽ thu được các đặc trưng sau đây:

- 12 giá trị đặc trưng phổ Mel được biến đổi Fourier ngược
- 12 giá trị delta phổ
- 12 giá trị double delta phổ
- 1 giá trị mức năng lượng
- 1 giá trị delta mức năng lượng
- 1 giá trị double delta mức năng lượng

Tổng cộng: 39 đặc trưng cho mỗi frame tiếng nói.

Việc rút trích đặc trưng MFCC từ một chỉ thị lệnh âm thanh cho ra tập hợp các véctơ đặc trưng khác nhau. Các giải thuật máy học (như mạng nơron hay SVM) thường cần dữ liệu đầu vào là bảng có cùng số chiều (cột, thuộc tính) để huấn luyện mô hình nhận dạng. Để có thể tạo cấu trúc bảng cho giải thuật học, cần phải biểu diễn lại các đặc trưng theo mô hình túi từ, như đã thực hiện trong các nghiên cứu [3], [11]. Sử dụng giải thuật *k*means [13] gom nhóm các véctơ MFCC vào các nhóm (cluster) và mỗi cluster tương ứng với một từ. Tập các cluster này tạo thành một từ điển. Sau cùng, mỗi véctơ MFCC trong chỉ thị lệnh âm thanh sẽ được gán vào cluster gần nhất (dựa vào khoảng cách mỗi véctơ đến các tâm của các cluster đại diện đã được định nghĩa trước đó. Tiếp theo, một chỉ thị lệnh âm thanh được biểu diễn bằng tần số của các từ trong chỉ thị lệnh âm thanh. Bước tiền xử lý này thường làm giảm độ chính xác khi nhận dạng tuy xử lý rất nhanh.

Các nghiên cứu của [15], [19] sử dụng giải thuật xoắn thời gian động DTW (Dynamic Time Warping) để nhận dạng trực tiếp tiếng nói mà không cần qua bước biểu diễn mô hình túi từ. Phương pháp tuy đơn giản nhưng độ chính xác cao hơn sử dụng mô hình túi từ. Khuyết điểm của phương pháp chính là thời gian thực thi khi nhận dạng rất lâu do việc so khớp theo giải thuật xoắn thời gian động bậc 2 so với số lượng véctơ đặc trưng MFCC.

Phương pháp chúng tôi đề xuất dựa trên giải thuật Naïve Bayes Nearest Neighbor (NBNN) [11], để nhận dạng trực tiếp các chỉ thị lệnh âm thanh mà cũng không cần có bước tiền xử lý để biểu diễn mô hình túi từ. Phương pháp có ưu điểm rất lớn do tính đơn giản, đạt được độ chính xác cao và thời gian nhận dạng nhanh hơn rất nhiều so với dùng giải thuật DTW.

3.2. Giải thuật NBNN

Giải thuật NBNN đã được đề xuất bởi O. Boiman [11] vào năm 2008. NBNN thực hiện tính toán trực tiếp khoảng cách từ *"ảnh đến lớp"*, để thực hiện phân lớp ảnh mà không cần phải qua bước tạo mô hình túi từ như thường thấy trong phân lớp ảnh [3]. NBNN là phương pháp phân loại ảnh rất thành công, được mở rộng xử lý trong phân lớp ảnh và các ứng dụng tương tự [1], [9], [14] và [17].

Chúng tôi đề xuất sử dụng phương pháp NBNN thực hiện nhận dạng tiếng nói như sau. Khi có chỉ thị lệnh âm thanh được đưa vào, thực hiện rút trích các đặc trưng MFCC, thu được các mô tả của âm thanh $d_1, ..., d_n$ (đặc trưng MFCC). Tương ứng với mỗi lớp C, cần tính tổng khoảng cách mỗi d_i đến láng giềng gần nhất của d_i trong lớp C, là nhỏ nhất.

$$sum = \sum_{i=1}^{n} \|d_i - NN_C(d_i)\|^2 \quad (8)$$

Trong đó $NN_C(d_i)$ là mô tả láng giềng gần nhất của d_i trong phân lớp C.

Ý tưởng NBNN là thực hiện tính mật độ xác suất $p(d|C)$ của mô tả d_i trong lớp C. Vì các mô tả trong cơ sở dữ liệu là rất lớn việc tính toán trở nên khó khăn hơn, nên một ước lượng mật độ xác suất Parzen cung cấp một xấp xỉ mật độ xác suất $p(d|C)$ làm cho việc tính toán nhẹ hơn. Cho d_1^C, d_2^C, …., d_L^C trong lớp C là các mô tả của tất cả các đặc trưng MFCC của âm thanh trong lớp C.

Sau đó ta ước tính Parzen của $p(d|C)$ ta được:

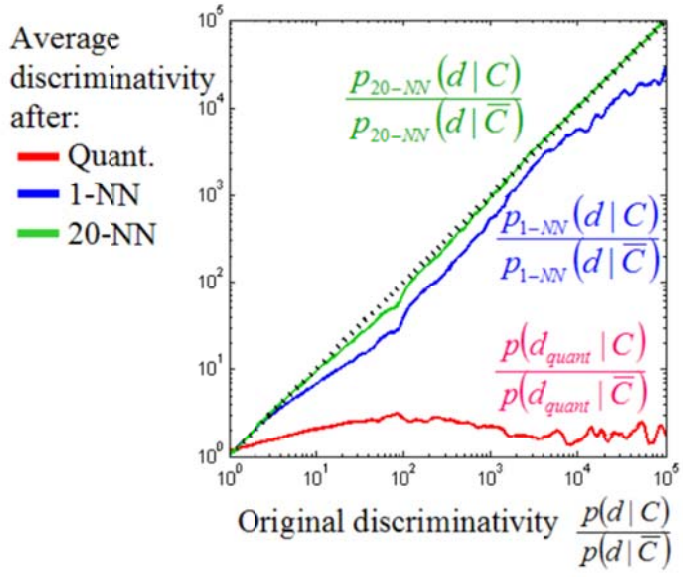
$$\hat{p}(d|C) = \frac{1}{L}\sum_{j=1}^{L} K\left(d - d_j^C\right) \quad (9)$$

Trong đó *K(*)* là một hàm của Parzen và hàm *K(*) > 0*.

Trong thực tiễn, K(*) là một hàm Gauss:

$$K(d - d_j^C) = \exp(-\frac{1}{2\sigma^2}\left\|d - d_j^C\right\|^2) \quad (10)$$

Như đã chỉ ra trong [2], khi L tiến đến vô cùng và σ giảm một cách phù hợp thì ước lượng $\hat{p}$ sẽ hội tụ xấp xỉ mật độ $p(d|C)$.



Hình 6. Biểu đồ so sánh 1-NN, 20-NN [2]

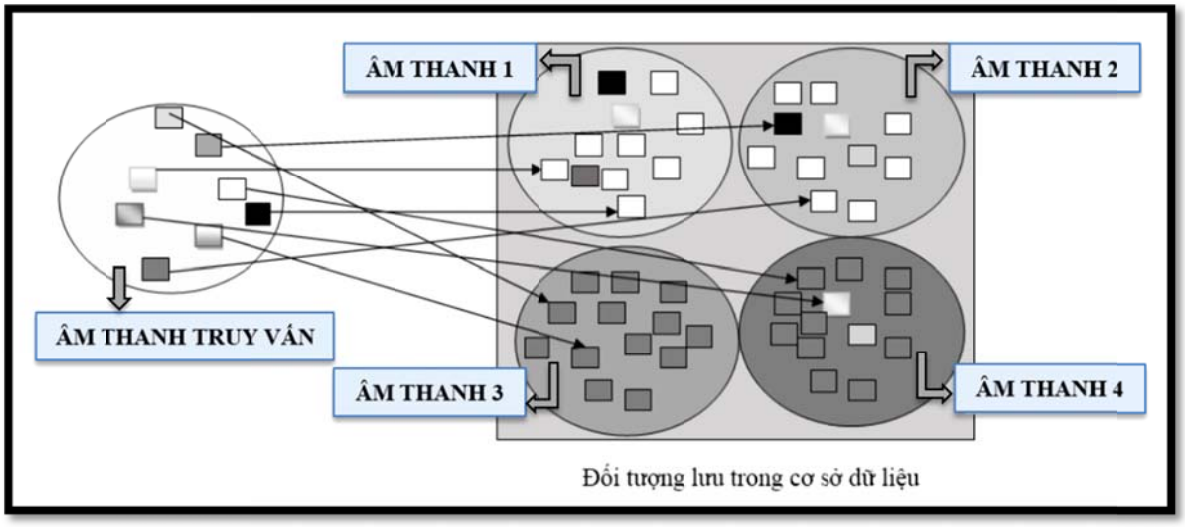
Để tính toán có được độ chính xác cao, tất cả các mô tả trong cơ sở dữ liệu cần được tính toán trong phương trình (9), việc tính toán này tốn rất nhiều thời gian vì nó đòi hỏi tính toán khoảng cách từ $d - d_j^C$ của tất cả các mô tả d_j^C ($j=1....L$). Gần như các mô tả của âm thanh đầu vào khá độc lập trong không gian mô tả, những mô tả này gần như khá xa so với hầu hết các mô tả trong cơ sở dữ liệu, chỉ có một số ít mô tả *d* ảnh hưởng đến phương trình (9). Do đó, có thể tính xấp xỉ phương trình (9) bằng cách sử dụng *r* láng giềng gần nhất của một mô tả $d \in Q$ trong mô tả của d_1^C, d_2^C, …., d_L^C trong lớp *C*.

$$p_{NN}(d|C) = \frac{1}{L}\sum_{d=1}^{r} K(d - d_{NN_j}^C) \quad (11)$$

Trong công thức (11), ngay cả khi *r=1* láng giềng gần nhất cho mỗi *d* trong lớp *C* thì xấp xỉ này cũng rất chính xác. Có thể xem điều này ở hình 6. Khi các mô tả *d* của âm thanh được lượng tử hóa (dquant) (biểu đồ đường màu đỏ) thì khả năng phân biệt mô tả *d* trong C và $\bar{C}$ là rất thấp. Trong khi đó, 1-NN *(1 láng giềng gần nhất)* và 20-NN *(20 láng giềng gần nhất)* thì khả năng phân biệt gần như có chính xác tương đương. Có thể nhận thấy không có sự khác biệt lớn khi *r* thay đổi (*r =1….1000*). Chọn *r=1* là thuận tiện nhất vì khi đó *log[p(d|C)]* được tính toán khá đơn giản. Khi *log[p(d|C)]* không còn phụ thuộc vào sự khác biệt của hàm Gauss thì *log[p(Q|C)]* được tính theo công thức (12):

$$\log p(Q|C) \propto -\sum_{i=1}^{n}\left\|d_i - NN_C(d_i)\right\|^2 \quad (12)$$

Trong đó, các mô tả d_i của tập tin âm thanh tương ứng với các đặc trưng MFCC được trích xuất. Hình 7 minh họa giải thuật NBNN tính khoảng cách đến láng giềng gần nhất từ một tập tin âm thanh truy vấn đến một lớp.



Hình 7. Láng giềng gần nhất từ một mô tả của truy vấn đến âm thanh đối tượng

IV. THỰC NGHIỆM

Để tiến hành đánh giá hiệu hiệu năng của hệ thống nhận dạng tiếng nói để điều khiển robot Pioneer P3-DX như đã đề xuất, giải thuật NBNN sử dụng đặc trưng MFCC (ký hiệu là MFCC-NBNN), chúng tôi đã cài đặt hệ thống chương trình bằng ngôn ngữ lập trình C/C++, sử dụng thư viện Yaafe [23] để trích đặc trưng MFCC. Chú ý trong giải thuật NBNN cần tìm láng giềng gần nhất của mỗi đặc trưng MFCC đến các đặc trưng MFCC của các chỉ thị lệnh lưu trữ trước đó, quá trình này mất nhiều thời gian nếu phải tính hết khoảng cách từ đặc trưng của truy vấn đến các đặc trưng MFCC trong cơ sở dữ liệu. Để tăng tốc quá trình này, chúng tôi đã sử dụng cấu trúc chỉ mục kd-tree [8], hỗ trợ cho NBNN có thể tính toán nhanh hơn khi phân lớp.

Chúng tôi cũng tiến hành so sánh mô hình chúng tôi đề xuất (MFCC-NBNN) với mô hình nhận dạng Markov ẩn HMM (tham khảo [16]), phương pháp tiếp cận MFCC và DTW (tham khảo [15], [19], MFCC-DTW), mô hình túi từ (BoAW) và máy học SVM (tham khảo [11], MFCC-BoAW-SVM). Chúng tôi đã cài đặt HMM, MFCC-DTW bằng ngôn ngữ lập trình Python. Với MFCC-BoAW-SVM, chúng tôi đã cài đặt bằng ngôn ngữ lập trình C/C++, sử dụng thư viện OpenCV [4].

Các thực nghiệm được tiến hành trên một máy tính cá nhân chạy hệ điều hành Linux với bản phân phối Ubuntu 14.04. Chúng tôi sử dụng robot Pioneer P3-DX của hãng Adept Mobile Robot [22] được trình bày ở phần II. Thông tin máy tính và ứng dụng cài đặt trong nghiên cứu như sau:

Loại	Tên và kích thước sử dụng	Ghi chú
Hiệu máy tính	Laptop HP DV6T-2000;	
Bộ xử lý (CPU)	Intel Core i7 CPU Q720 @1.60Ghz (8 CPUs), 1.6GHz;	
Bộ nhớ (RAM)	4096MB – DDR3;	
Hệ điều hành	Ubuntu 14.04 OS (64 bit);	
Voice	A4Tech – 16000Hz;	
Tốc độ Webcam	20 frame/s;	
Thư viện robot	SDK ARIA 2.7.2, MobileSim 0.5; [21], [22];	
Ngôn ngữ lập trình	C++, biên dịch g++ Linux;	
Thư viện trích đặc trưng	yaafe-v0.64.tgz [23];	

Thiết lập thông số cho robot trong thực nghiệm

Thông số robot sử dụng khoảng cách dựa vào Sonar và Lazer

- Vận tốc ban đầu của robot khi di chuyển là V0 = 200 mm/s;
- Khoảng cách dừng của robot trong khoảng L0 = (40 – 55) cm;
- Khoảng cách robot hoạt động:Lt = 56cm trở lên;

- Khoảng cách robot lùi Lb = (28 – 40) cm;
- Rẽ trái, rẽ phải liên tục với mỗi lần thực hiện một góc là 10 độ.

Chuẩn bị tập dữ liệu

Chuẩn bị tập dữ liệu chỉ thị lệnh âm thanh của robot, chúng tôi đã thu thập với số lượng 20 người đọc khác nhau (gồm 10 nam và 10 nữ), đọc các chỉ thị lệnh bao gồm 5 từ khóa được sử dụng là queo trái, queo phải, thẳng tiến, đi lùi, dừng lại, và lớp khác (lớp bù, không phải là 5 lớp từ khóa lệnh). Mỗi mẫu âm thanh chúng tôi thu trong một giây. Được tập dữ liệu với 3614 tập tin âm thanh, lưu theo định dạng tập tin wav.

Các tham số sử dụng trong rút trích đặc trưng MFCC

```
MFCC CepsIgnoreFirstCoeff=1 CepsNbCoeffs=39 FFTWindow=Hamming MelMaxFreq=6854.0 MelMinFreq=130.0
MelNbFilters=40 blockSize=1024 stepSize=512
```

Trong đó:

- CepsIgnoreFirstCoeff (default=1): 0 - Hệ số Coeff đầu tiên;
- CepsNbCoeffs (default=13): Số lượng Coeff được tạo ra. Bài báo đề xuất sử dụng Coeff là 39;
- FFTWindow (default=Hanning): Sử dụng loại khung cửa số trượt. Hanning|Hamming|None;
- MelMaxFreq (default=6854.0): Chỉ số mel frequency lớn nhất;
- MelMinFreq (default=130.0): Chỉ số mel frequency nhỏ nhất;
- MelNbFilters (default=40): Số lượng Mel;
- blockSize (default=1024): Kích thước của mỗi Block trong frames;
- stepSize (default=512): Kích thước giữa 2 frames.

Các tham số của MFCC-BoAW-SVM được sử dụng như sau:

- Sử dụng *k*means trong việc gom cụm BoAW với *k=1000* cụm (cluster) → 1000 từ
- Giải thuật SVM đã lựa chọn sử dụng hàm nhân RBF Radial Basis Function: $K(u, v) = \exp(\gamma \times \|u - v\|^2)$ với các thông số *gamma*, *C* tối ưu (*gamma=0.01*; *C=100000*);

Kết quả nhận dạng

Nghi thức kiểm tra trong thực nghiệm của chúng tôi là hold-out, lấy ngẫu nhiên 2/3 tập dữ liệu làm tập học (hay cơ sở dữ liệu đối tượng, dùng cho huấn luyện mô hình, điều chỉnh các tham số), và 1/3 tập dữ liệu còn lại làm tập kiểm tra để thông báo kết quả nhận dạng của mô hình. Kết quả nhận dạng (độ chính xác, thời gian xử lý) của các mô hình thu được như trình bày trong bảng 1.

Bảng 1. Kết quả nhận dạng của MFCC-NBNN, MFCC-DTW, BoAW-SVM, HMM

	Độ chính xác (%)	**Thời gian xử lý (phút)**
MFCC-NBNN	**98.50**	17.23
MFCC-DTW	98.40	720.57
BoAW-SVM	97.10	**5.41**
HMM	97.80	224.80

Hình 8. Biểu đồ so sánh độ chính xác và thời gian xử lý của các giải thuật

Nếu so sánh về độ chính xác, có thể thấy rằng đề xuất MFCC-NBNN cho kết quả nhận dạng tốt hơn so với các nghiên cứu trước đó. Do các giải pháp được cài đặt bởi các ngôn ngữ lập trình khác nhau, nên có sự chênh lệch rất xa

về thời gian xử lý. Nhưng với kết quả thu được, có thể thấy rằng MFCC-NBNN vẫn cho thời gian xử lý nhanh. Từ đó, Robot sẽ di chuyển và hoạt động đúng như yêu cầu đặt ra.

Phương pháp đề xuất có ưu điểm rất lớn do tính đơn giản trong giải thuật vì không cần qua bước xử lý trung gian nào mà đạt được độ chính xác cao và thời gian nhận dạng nhanh hơn rất nhiều so với dùng giải thuật DTW hay giải thuật HMM.

Điều khiển robot Pioneer P3-DX

Robot thực hiện di chuyển theo yêu cầu đặt ra với 5 từ khóa – thẳng tiến, đi lùi, quẹo trái, quẹo phải, dừng lại. Chúng tôi đã xây dựng hệ thống nhận dạng liên tục với mỗi giây sẽ thực hiện thu âm và nhận dạng âm thanh đó nhằm đảm bảo hoạt động tốt theo thời gian thực, để giúp cho robot có thể nhận dạng âm thanh giảm lỗi chúng tôi đã thêm vào phân lớp bù. Robot sẽ di chuyển và thực hiện tác vụ khi nhận được tín hiệu âm thanh của 5 từ khóa và thực hiện tín hiệu đó cho đến khi có một tín hiệu khác được chỉ định để thực hiện các tác vụ kế tiếp, khi nhận dạng được phân lớp bù thì robot sẽ ra lệnh tác vụ mới mà thực hiện tác vụ trước đó được chỉ định. Kết quả cho thấy robot đã nhận dạng tốt 5 từ khóa và mỗi khi không có tín hiệu âm thanh hay tín hiệu sai thì hệ thống xác định đúng phân lớp bù, hạn chế khả năng gây lỗi và đưa ra quyết định chính xác để điều khiển robot.

V. KẾT LUẬN

Chúng tôi trình bày ý tưởng điều khiển robot Pioneer P3-DX bằng tiếng nói theo thời gian thực với giải thuật NBNN sử dụng đặc trưng MFCC. Từ tập dữ liệu mẫu tiếng nói tương ứng với các lệnh điều khiển robot được thu âm từ 20 người đọc khác nhau, chúng tôi đề xuất sử dụng giải thuật máy học NBNN để nhận dạng trực tiếp các tiếng nói là các lệnh điều khiển hoạt động robot từ các đặc trưng MFCC tương ứng không cần bất kỳ thao tác xử lý trung gian nào khác. Kết quả thực nghiệm cho thấy rằng phương pháp đề xuất có thể nhận dạng chính xác tiếng nói là các lệnh điều khiển robot, đáp ứng thời gian thực, khi so sánh với các mô hình nhận dạng Markov ẩn HMM, MFCC-DTW, BoAW-SVM. Với khả năng nhận dạng nhanh với độ chính xác trên 98%, nên hệ thống điều khiển robot Pioneer P3-DX hoạt động hiệu quả.

Hướng nghiên cứu kế tiếp cho bài báo là nâng cao khả năng nhận dạng âm thanh cũng như kết hợp với hình ảnh (thông qua Webcam) nhằm giúp robot có thể vừa "nhìn" mà lại vừa "nghe". Nghiên cứu tập trung vào bước tiền xử lý các tạp âm, khử các tín hiệu nhiễu nhằm giúp khả năng nhận dạng được tốt hơn. Hiện robot đang di chuyển động không theo một bản đồ cụ thể, do đó có thể tích hợp thêm một số bản đồ tĩnh nhằm giúp robot có thể di chuyển trong một phạm vi nhất định.

Nghiên cứu các ứng dụng đáp ứng nhu cầu thực tế như: robot phục vụ cho trẻ em trong học tập và vui chơi, giải trí (robot hỗ trợ tra cứu từ điển, robot biết hát theo yêu cầu), robot làm phương tiện chuyên chở người khuyết tật (đối tượng được người khuyết tật sử dụng để robot di chuyển), robot lau dọn vệ sinh (đối tượng là các vết bẩn – kết hợp với ra lệnh bằng âm thanh và tìm kiếm vết bẩn thông qua hình ảnh).

VI. TÀI LIỆU THAM KHẢO

[1] R. Behmo, P. Marcombes, A. Dalalyan, and V. Prinet, "Towards optimal naive Bayes nearest neighbors", In European Conference on Computer Vision – ECCV, Lecture Notes in Computer Science, Vol. 6314, 2010, pp 171-184.

[2] O. Boiman, E. Shechtman, M. Irani, "In Defense of Nearest-Neighbor Based Image Classification". In Proc. of IEEE conference on Computer Vision and Pattern Recognition - CVPR, 2008, pp. 1-8.

[3] A. Bosch, A. Zisserman, X. Munoz, "Scene classification via pLSA", In: Proceedings of the European Conference on Computer Vision, 2006, pp. 517-530.

[4] G. Bradski and A. Kaehler. "*Learning OpenCV*". O'Reilly Media, 2012.

[5] S. Cassidy, "Speech Recognition", Department of Computing, Macquarie University, Australia, 2002.

[6] A-G. Chiţu, L-J-M. Rothkrantz, P. Wiggers, J-C. Wojdel, "Comparison between different feature extraction techniques for audio-visual speech recognition", *Journal on Multimodal User Interfaces*, Vol.1(1):7-20, 2007.

[7] S.B. Davis and P.Mermelstrin, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences", *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol.28(4): 357-366, 1980.

[8] J. Friedman, J. Bentley, R. Finkel, "An algorithm for finding best matches in logarithmic expected time", *ACM Transactions on Mathematical Software*, Vol.3(3):209-226, 1977.

[9] M. Fritz, T.Tuylelaars, T. Darrell, K. Saenko. "The NBNN kernel", IEEE International Conference on Computer Vision - ICCV, 2011, pp. 1824-1831.

[10] G. Hinton, L. Deng, D. Yu, G-E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T-N. Sainath, B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups", *Signal Processing Magazine*, IEEE, Vol. 29(6):82-97, 2012.

[11] S-M. Kamruzzaman, A-N-M. Rezaul Karim, S. Islam, E. Haque, "Speaker Identification using MFCC-Domain Support Vector Machine", *International Journal of Electrical and Power Engineering*, Vol. 1, pp. 274-278, 2007.

[12] Y. LeCun and Y. Bengio, "Convolutional Networks for Images, Speech, and Time-Series", in Arbib, M. A. (Eds), *The Handbook of Brain Theory and Neural Networks*, MIT Press, 1995.

[13] J. MacQueen, "Some methods for classification and analysis of multivariate observations", In proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, Vol. 1, 1967, pp. 281-297.

[14] S. McCann, D. Lowe**,** "Local Naive Bayes Nearest Neighbor for image classification", Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition - CVPR, 2012, pp. 3650-3656.

[15] L. Muda, M. Begam and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques", *Journal of computing*, Vol.2(3):138-143, 2010.

[16] L-R. Rabiner, "*A tutorial on hidden Markov models and selected applications in speech recognition*", Proceedings of the IEEE, 1989, Vol.77(2): 257-286.

[17] K. Rematas, M. Fritz, T.Tuylelaars, "The pooled NBNN Kernel: Beyond Image-to-Class and Image-to-Image", Proceedings of the 11th Asian Conference on Computer Vision – ACCV, Lecture Notes in Computer Science Vol. 7724, 2013, pp 176-189.

[18] V.Vapnik, *"The Nature of Statistical Learning Theory"*, Springer, NewYork, 1995.

[19] D-L. Vu, N-T. Vu. "Vietnamese Speech Recognition Using Dynamic Time Warping and Coefficient of Correlation", Proc. of The International Conference on Control, Automation and Information Sciences (ICCAIS), 2013, pp. 64-67.

[20] H-Q. Vu, K. Hoang, N-T. Pham, T-T. Lam, H-H. Nguyen, A-H. Nguyen, "A System for Recognizing Vietnamese Document Images Based on HMM and Linguistics", ICDAR 2001, pp. 627-630.

[21] A. WhitBrook, "*Programming Mobile robot with Aria and Player*", Springer, 2010.

[22] Adept MobileRobots, "Adept MobileRobots Community, Knowledge Base, and Support Site for Pioneer Research and Education Customers", 2014, http://robots.mobilerobots.com/wiki/Main_Page.

[23] B.Mathieu, S.Essid, T.Fillon, J.Prado, G.Richard, "YAAFE, an Easy to Use and Efficient Audio Feature Extraction Software", Proceedings of the 11th ISMIR conference, Utrecht, Netherlands, 2010.

VOICE CONTROLLED ROBOT PIONEER P3-DX THROUGH MFCC AND NAÏVE BAYES NEAREST NEIGHBORS

Ma Truong Thanh, Do Thanh Nghi, Pham Nguyen Khang, Chau Ngan Khanh

ABSTRACT - *In this paper, we propose to use MFCC (Mel-scale Frequency Cepstral Coefficient) and NBNN (Naïve Bayes Nearest Neighbor) for voice controlled robot Pioneer P3-DX. The dataset of six voice commands (turn left / right, go forward / backward, stop, other) is collected from 20 different readers. The next processing step is to extract 39 MFCC features from each voice command. And then, we propose to use the NBNN algorithm to directly recognize voice commands for controlling the robot without any complex task. Experimental results show that the proposed method (NBNN and MFCC) efficiently recognizes voice commands in real-time. The proposed approach achieves 98.5% in terms of accuracy, compared with state-of-the-art algorithms, including 97.14% obtained by support vector machines using the bag-of-audio-words, 98.4% given by dynamic time warping and 97.8% achieved by hidden Markov model. Moreover, NBNN using MFCC method is simple and fast to recognize voice commands in real-time.*

Keywords - *Speech recognition, MFCC features, Naïve Bayes Nearest Neighbor, Controlling robot Pioneer P3-DX.*