

Nghiên cứu phương pháp phát hiện va chạm của cánh tay robot cộng tác 6 bậc tự do

Nghiêm Văn Hưng^{1,2}, Đặng Văn Đức³, Nguyễn Văn Căn², Trịnh Hiền Anh³

¹ Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội, Việt Nam

² Trường Đại học Kỹ thuật - Hậu cần Công an nhân dân, Bộ Công an, Bắc Ninh, Việt Nam

³ Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội, Việt Nam

Tác giả liên hệ: Nghiêm Văn Hưng, ngiemvanhung1985@gmail.com

Ngày nhận bài: 08/04/2023, ngày sửa chữa: 29/07/2023, ngày duyệt đăng: 01/10/2023

Định danh DOI: 10.32913/mic-ict-research-vn.v2023.n2.1236

Tóm tắt: Robot cộng tác (Collaborative robot - Cobot) là những robot có thể tiếp xúc với con người một cách trực tiếp, phát huy đồng thời được các lợi thế của cả con người và robot để tăng hiệu suất công việc. Cobot hoạt động thân thiện và tương tác với con người vì được lập trình để phát hiện va chạm an toàn. Do đó yêu cầu phát hiện chính xác và nhanh chóng các va chạm của cánh tay cobot là một chủ đề thu hút được sự quan tâm của nhiều nhà nghiên cứu. Nghiên cứu của chúng tôi đề xuất hướng tiếp cận áp dụng hai kỹ thuật học máy có giám sát là SVMR (Support Vector Machine Regression) và 1D CNN (1D Convolutional Neural Network) cho kết quả phát hiện va chạm hiệu quả với cánh tay robot CURA6 trên bộ dữ liệu của Intema.

Từ khóa: Phát hiện va chạm, cánh tay robot cộng tác, 6-DoF.

Title: Collision Detection for 6-DoF Collaborative Robot Arm

Abstract: Cobots are robots that can directly contact humans, simultaneously promoting the advantages of both humans and robots to increase work efficiency. Cobots operate friendly and interactive with humans because they are programmed to detect collisions safely. Therefore, the need to accurately and quickly detect collisions of cobot arms is a topic that attracts the attention of many researchers. The our proposed technique using the SVMR model and the 1D CNN model is tested and gives good collision detection results with CURA6 cobot arm on the Intema's dataset.

Keywords: Collision detection, cobot arm, 6-DoF.

I. GIỚI THIỆU

Cobot là những robot có thể làm việc đồng thời với người vận hành. Theo ISO 10218 [7], cobot là robot có thể tiếp xúc với con người một cách trực tiếp trong một không gian làm việc cộng tác xác định. Khác với robot công nghiệp, cobot được thiết kế để bảo đảm tương tác an toàn với con người cũng như môi trường làm việc cộng tác. Cobot được chế tạo bằng vật liệu nhẹ và thiết kế với hình dạng mô phỏng tương tự cánh tay người. Cobot hoạt động phối kết hợp với con người để tăng hiệu năng quá trình sản xuất hoặc khắc phục các hạn chế kỹ thuật mà không cần đến các thiết bị vật lý bảo hộ như tấm chắn, lưới, hàng rào bảo vệ như robot công nghiệp [18].

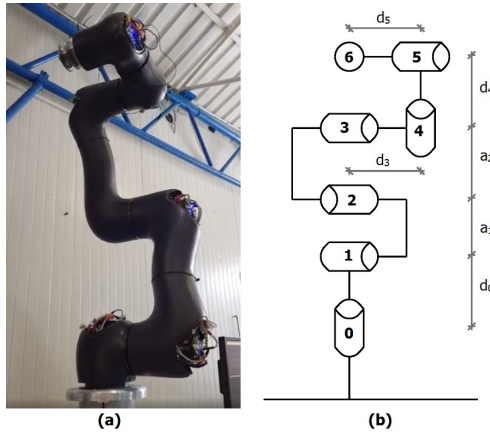
Đã có nhiều công bố về bài toán phát hiện va chạm với đối tượng nghiên cứu là cánh tay cobot [2, 8, 13, 20]. Điển hình là cánh tay cobot sáu bậc tự do 6-DoF (Six Degrees of Freedom) được thiết kế nhằm thực hiện các tác vụ chính

xác và an toàn với con người. Hình 1 mô tả cánh tay robot CURA6 (Cooperative Universal Robotic Assistant 6) của Intema [2].

Các phần tiếp theo của bài báo gồm: Mục II giới thiệu các công trình nghiên cứu có liên quan; mục III khái quát về động lực học robot; mục IV mô tả phương pháp đề xuất của chúng tôi. Các kết quả thử nghiệm trên bộ dữ liệu Intema được trình bày trong mục V. Cuối cùng là Mục VI đưa ra kết luận và định hướng cho các nghiên cứu phát triển trong thời gian tới.

II. CÁC NGHIÊN CỨU LIÊN QUAN

S. Haddadin và các cộng sự [5] công bố một khảo cứu khá toàn diện về bài toán xử lý va chạm của robot. Bài toán xử lý va chạm được chia thành những tác vụ chính là: phát hiện (detection) va chạm, nhận dạng (identification) va chạm, phân loại (classification) va chạm và phản ứng



Hình 1. (a) Hình chụp robot CURA6 tại phòng thí nghiệm của Intema; (b) Mô hình robot CURA6

(reaction) và chạm. Trọng tâm nghiên cứu của chúng tôi là bài toán phát hiện và chạm, đây chính là bước đầu tiên, có tính chất định hướng các tác vụ khác trong toàn bộ hệ xử lý và chạm. Chúng tôi phân chia các phương pháp phát hiện và chạm trên cánh tay cobot thành hai nhóm: (i) nhóm phương pháp sử dụng học máy và (ii) nhóm phương pháp không sử dụng học máy. Nhóm phương pháp (ii) thường dựa vào các cảm biến bề mặt xúc giác (sensor skins) bao phủ bên ngoài robot [1, 17]. Tuy nhiên, các cảm biến bề mặt xúc giác thường có chi phí cao, không bền và có thể bị hỏng khi bị va chạm mạnh hay va chạm nhiều lần. Các phương pháp dựa trên học máy [2, 5, 13] thường ước tính các mômen xoắn khớp bên ngoài do va chạm, sau đó so sánh với một tập giá trị ngưỡng cho trước để kiểm tra xem va chạm có xảy ra hay không. Đo mômen xoắn trực tiếp sẽ là lý tưởng, nhưng cảm biến mômen xoắn có thể tốn kém, giải pháp thay thế là ước tính mômen xoắn khớp từ các phép đo dòng điện tại các bộ truyền động khớp.

Hiện nay, việc áp dụng các phương pháp học máy để phát hiện va chạm đã trở nên phổ biến. Với điều kiện thu thập đủ lượng dữ liệu đa dạng, chất lượng tốt, các phương pháp học máy mở ra khả năng có thể khắc phục được sự không chắc chắn (uncertainty) của các tham số trong mô hình động lực học cũng như các hiệu ứng không được mô hình hóa như ma sát và nhiễu đo lường.

A. N. Sharkawy và cộng sự [14] đã sử dụng M-FNN (Multilayer Feedforward Neural Network) với sai số vị trí khớp và vận tốc làm đầu vào (input), đồng thời ước tính mômen xoắn khớp bên ngoài mà không cần sử dụng động lực học robot (nhưng cũng yêu cầu điều chỉnh ngưỡng va chạm thủ công cho từng khớp), trong khi [15] xem xét các mômen khớp đo được và vectơ lực hấp dẫn của tay máy. Tuy nhiên, các phương pháp này chỉ được thực nghiệm với các chuyển động hai và ba khớp của một tay máy robot 7-DoF. Tác giả Z. Zhang và các cộng sự [19] huấn luyện

một FNN với các tín hiệu đầu vào 36 chiều được trích xuất từ cả các đặc điểm miền thời gian và tần số của các phép đo cảm biến mômen xoắn khớp. Đối với các robot có hình dạng đầy đủ giống như con người, nhóm nghiên cứu của K. Narukawa [11] đề xuất cải tiến kỹ thuật máy vectơ hỗ trợ một lớp (OC-SVM), trong đó đầu vào được coi là vectơ điểm mômen của robot hình người và các phép đo lực và mômen xoắn cho từng chi.

Gần đây, Y. J. Heo và các cộng sự [6] đề xuất phương pháp phát hiện nhanh và chậm của cánh tay robot 6-DoF sử dụng mạng neural tích chập một chiều (1D CNN) với kích thước là $W \times 66$ (với W biểu thị kích thước cửa sổ thời gian) trong đó có các mômen xoắn khớp bên ngoài ước tính làm đầu vào; đầu ra là kết quả phát hiện va chạm dạng nhị phân (*True, False*). K. M. Park và các cộng sự [13] tập trung nghiên cứu giải quyết vấn đề phát hiện va chạm của cánh tay robot Doosan M0609. Trong khi M. Czubenko và nhóm nghiên cứu [2] đề xuất kết hợp một số mạng neural khác nhau như AR, RNN, CNN-LSTM và MC-LSTM nhằm phát hiện chính xác va chạm của cánh tay robot CURA6. Tuy nhiên, hạn chế chung của các phương pháp do K. M. Park và M. Czubenko đề xuất là sự phụ thuộc vào ma sát của động cơ, đặc biệt là ma sát Coulomb.

Trong bài báo này, nhóm nghiên cứu của chúng tôi đề xuất áp dụng hai phương pháp học có giám sát là SVMR (Support Vector Machine Regression) và 1D CNN (1D Convolutional Neural Network) để phát hiện va chạm của cánh tay robot CURA6 trên cơ sở các phép đo dòng điện cùng với mô hình động lực học của robot. Hướng nghiên cứu của chúng tôi có ưu điểm là không yêu cầu bất kỳ cảm biến bên ngoài bổ sung nào khác, so với [19] yêu cầu cảm biến mômen xoắn khớp tại các khớp. Hai phương pháp của chúng tôi không yêu cầu bù ma sát. Chi phí tính toán trong phương pháp đề xuất ít hơn đáng kể cho cả huấn luyện và suy luận (đầu vào 4 chiều cho SVMR và 6 chiều cho 1D CNN) trong khi phương pháp của Y. J. Heo [6] có đầu vào 66 chiều.

III. ĐỘNG LỰC HỌC ROBOT

1. Cấu tạo của cánh tay robot

Các bộ phận chủ yếu của một robot CURA6 bao gồm: đế, các khớp (Joint) quay hoặc tịnh tiến, các liên kết (Link) giữa các khớp và bàn tay robot còn được gọi khác là khâu chấp hành cuối (End of Effect). Cánh tay robot CURA6 của Intema có đặc trưng nhỏ gọn với trọng tải 5000 g, bán kính hoạt động rộng 1200 mm. Robot CURA6 có thể thực hiện được các nhiệm vụ phức tạp với yêu cầu tính chính xác cao như thao tác xử lý, tiếp xúc đối với những vật có đặc tính dễ biến dạng hoặc dễ rách, vỡ trong những điều kiện vận hành với yêu cầu kỹ thuật rất khắt khe.

Bảng I thể hiện bảng tham số động học D-H (Denavit

Bảng I
BẢNG THAM SỐ D-H CỦA ROBOT CURA6

i	Khoảng cách a_i	Góc quay α_i	Khoảng cách d_i	Mômen τ_i	Góc quay θ_i
0	0	$\frac{\pi}{2}$	0.105	τ_0	θ_0
1	0.4	0	-	τ_1	θ_1
2	0.4	0	-	τ_2	θ_2
3	0	$\frac{\pi}{2}$	0.220	τ_3	θ_3
4	0	$-\frac{\pi}{2}$	0.200	τ_4	θ_4
5	0	0	0.140	τ_5	θ_5

- Hartenberg) của robot CURA6 [2]. Ký hiệu trong bảng I như sau:

a_i là tham số khoảng cách tính từ trục Z_{i-1} tới trục Z_i đo dọc theo trục X_i .

α_i là tham số góc quay trục Z_{i-1} xung quanh trục X_i tới song song hoặc trùng với trục Z_i .

d_i là tham số khoảng cách giữa 2 trục X_{i-1} và trục X_i đo dọc theo trục Z_{i-1} .

θ_i là tham số góc quay của trục X_{i-1} xung quanh trục Z_{i-1} tới song song với trục X_i .

2. Phương trình động lực học

Phương trình động lực học được định nghĩa cho robot CURA6 là:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + \tau_F = \tau_m \quad (1)$$

với $q(t) = [q_0(t), q_1(t), \dots, q_{n-1}(t)] \in \mathbb{R}^n$ bao gồm vị trí của mỗi khớp (với $n = 6$), $\dot{q}(t) \in \mathbb{R}^n$ và $\ddot{q}(t) \in \mathbb{R}^n$ lần lượt biểu thị vectơ vận tốc và gia tốc. $M(q) \in \mathbb{R}^{n \times n}$ là ma trận quán tính robot, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ là ma trận Coriolis, $g(q) \in \mathbb{R}^n$ là lực hấp dẫn, $\tau_F \in \mathbb{R}^n$ biểu thị lực ma sát và $\tau_m \in \mathbb{R}^n$ là vectơ mômen khớp. Lưu ý: $M(q)$ là ma trận đối xứng; $\tau_m = K_i i_m$ với $K_i \in \mathbb{R}^{n \times n}$ là ma trận khuếch đại và $i_m \in \mathbb{R}^n$ là dòng điện động cơ. Khi va chạm xảy ra, phương trình (1) trở thành:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + \tau_F = \tau_m + \tau_{ext} \quad (2)$$

với $\tau_{ext} \in \mathbb{R}^n$ là mômen xoắn khớp ngoài do va chạm hoặc lực bên ngoài khác tác động lên robot.

Ma trận $\dot{M}(q) - 2C(q, \dot{q})$ là đối xứng xiên, với $\dot{M}(q)$ là ký hiệu đạo hàm của $M(q)$ theo thời gian (chi tiết trong [10]).

$$\dot{M}(q) = C(q, \dot{q}) + C^T(q, \dot{q}) \quad (3)$$

Động lượng tổng quát của robot xác định như sau:

$$\rho = M(q)\dot{q} \in \mathbb{R}^n \quad (4)$$

Đạo hàm của ρ theo thời gian được biểu diễn là:

$$\dot{\rho} = \dot{M}(q)\dot{q} + M(q)\ddot{q} \quad (5)$$

hay

$$\dot{\rho} = \tau_m + \tau_{ext} - \tau_F + C^T(q, \dot{q})\dot{q} - g(q) \quad (6)$$

Ký hiệu $\beta(q, \dot{q}) = g(q) - C^T(q, \dot{q})\dot{q}$ và ma trận $K_O = \text{diag}(k_{O,i}) \in \mathbb{R}^{n \times n}$, động lực học của bộ quan sát động lượng được định nghĩa như sau:

$$\dot{\hat{\rho}} = \tau_m - \tau_F - \hat{\beta}(q, \dot{q}) + r_m \quad (7)$$

và

$$\dot{r}_m = K_O(\dot{\rho} - \dot{\hat{\rho}}) \quad (8)$$

Đầu ra $r_m(t) \in \mathbb{R}^n$ của bộ quan sát động lượng được mô tả như sau [13]:

$$r_m = K_O \left(\rho(t) - \int_0^t (\tau_m - \tau_F - \hat{\beta} + r_m) ds - \rho(0) \right) \quad (9)$$

với $\rho = \hat{M}(q)\dot{q}$ và $\hat{\beta} = \hat{g}(q) - \hat{C}^T(q, \dot{q})\dot{q}$, trong đó $\hat{M}$, $\hat{C}$ và $\hat{g}$ biểu thị các tham số mô hình. Trong các điều kiện lý tưởng khi $\hat{M} = M$ và $\hat{\beta} = \beta$, mối quan hệ động lực học giữa $r_m(t)$ và mômen xoắn khớp ngoài τ_{ext} là:

$$\dot{r}_m = K_O(\tau_{ext} - r_m) \quad (10)$$

sao cho $r_m(t)$ đóng vai trò ước tính cho τ_{ext} .

Mômen xoắn khớp ngoài ước tính được so sánh theo từng thành phần với ngưỡng cố định do người dùng xác định $\epsilon_{cd} \in \mathbb{R}^n$, sao cho hàm xác định va chạm $cd : r_m(t) \rightarrow \{True, False\}$ được định nghĩa là:

$$cd(r_m(t)) = \begin{cases} True & \text{if } |r_m(t)| > \epsilon_{cd} \\ False & \text{if } |r_m(t)| \leq \epsilon_{cd} \end{cases} \quad (11)$$

Hàm trả về kết quả dạng nhị phân cho biết có xảy ra va chạm hay không [3, 5].

IV. PHƯƠNG PHÁP ĐỀ XUẤT

1. Mô tả bộ dữ liệu

Có 15 lát (slices) chuyển động ngẫu nhiên của robot đã được chuẩn bị để làm dữ liệu huấn luyện, mỗi lát chứa 10000 mẫu (07 phút chuyển động). Tốc độ của mỗi khớp được bộ giải thiết kế an toàn nằm trong khoảng 25% tốc độ tối đa của động cơ.

Có 03 lát được thu thập mà robot không có tải trọng và 12 lát còn lại có tải (đơn vị tính bằng gam) từ danh sách sau {782, 1016, 1282, 2298, 2757, 4039}, với 02 lát cho mỗi tải (phạm vi tải xấp xỉ là từ 500 g đến 4000 g). Trong đó, dữ liệu học (90%) và dữ liệu xác thực (10%). Bộ dữ liệu này được tải về từ trang Gitlab (<https://gitlab.com/intema-gdansk/cura6-dataset/-/tree/main>).

Các tín hiệu cần được chuẩn hóa trong phạm vi [0,1] trước khi được xử lý làm đầu vào. Đầu ra không xét đến ma sát được mô tả bằng phương trình:

$$r = K_O \left(\rho(t) - \int_0^t (\tau_m - \hat{\beta} + r) ds - \rho(0) \right) \quad (12)$$

với

$$\dot{\rho} = \tau_m - \hat{\beta}(q, \dot{q}) + r \quad (13)$$

và

$$\dot{r} = K_O (\dot{\rho} - \dot{\rho}) \quad (14)$$

Trong các điều kiện lý tưởng, mối quan hệ động giữa $r(t)$ và mômen xoắn khớp ngoài τ_{ext} là:

$$\dot{r} = K_O (\tau_{ext} - \tau_F - r) \quad (15)$$

2. Phương pháp phát hiện va chạm

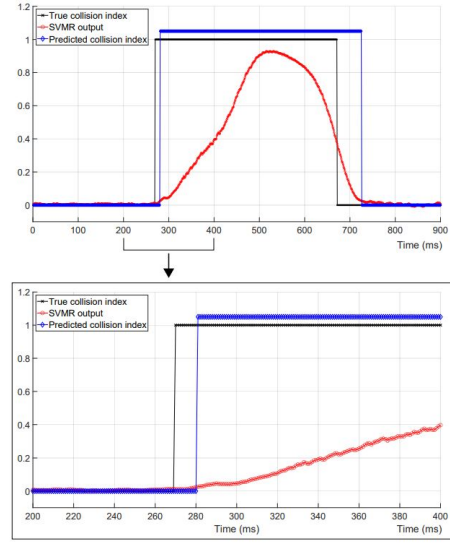
Trên cơ sở phương pháp SVMR [16] và 1D CNN [9, 12] là nền tảng, chúng tôi nghiên cứu phát triển mô hình phát hiện va chạm cho cánh tay robot CURA6. Quy ước: Một tín hiệu f -chiều thay đổi theo thời gian $s(t) \in \mathbb{R}^f$ được lấy mẫu tại khoảng thời gian lấy mẫu t_I trên một cửa sổ thời gian có kích thước t_W ; số lượng mẫu là $N+1 = \frac{t_W}{t_I} + 1 \in \mathbb{N}$.

a) Phát hiện va chạm với SVMR

SVMR lấy giá trị tuyệt đối của đầu ra (12) làm đầu vào.

- Thiết kế vectơ đặc trưng: Thiết kế vectơ đặc trưng bao gồm chuyển tín hiệu thành dạng vectơ và gán nhãn cho các đặc trưng. Để lấy mẫu theo thời gian, chúng tôi đặt $t_W = 80$ ms và $t_I = 8$ ms. Sau khi lấy mẫu đầu ra của bộ quan sát động lượng và biến đổi thành ma trận $R \in \mathbb{R}^{n \times (N+1)}$, giá trị các hàng 1, 2, 3 và 4 của R tạo thành một vectơ duy nhất, dẫn đến kết quả vectơ đặc trưng $x(t)$ như sau:

$$x = [r_{J_1} \quad r_{J_2} \quad r_{J_3} \quad r_{J_4}]^T \in \mathbb{R}^{4(N+1)} \quad (16)$$



Hình 2. Đầu ra của SVMR và chỉ số va chạm tương ứng

trong đó $r_{J_i} \in \mathbb{R}^{1(N+1)}$ biểu thị giá trị của hàng có thứ tự i của R . Lưu ý rằng hàng thứ i của R biểu thị đầu ra quan sát động lượng được lấy mẫu của khớp i . So với việc sử dụng tín hiệu của tất cả các khớp, kích thước của vectơ đặc trưng x này nhỏ hơn, giảm lượng tính toán cho SVMR.

- Huấn luyện SVMR: Vì hàm mũ yêu cầu tính toán nhiều hơn các hàm đa thức, nên chúng tôi sử dụng hàm SRQ [4] thay vì hàm RBF:

$$K_{SRQ}(x, z) = \frac{1}{\left(1 + \frac{\|x-z\|^2}{\sigma^2}\right)^2} \quad (17)$$

với các tham số $C = 1$, $\varepsilon = 0.02$ (dựa theo [16]) và $\sigma = 3$.

- Điều chỉnh độ nhạy phát hiện va chạm: Đầu ra của SVMR được huấn luyện là một giá trị vô hướng được chuẩn hóa thành một đơn vị được mô tả như trong Hình 2 sau đó được đặt ngưỡng và chuyển thành chỉ số va chạm nhị phân tương ứng theo (11).

Giá trị đầu ra gần bằng 0 khi không có lực bên ngoài tác dụng và tăng khi mômen xoắn bên ngoài ước tính của khớp 1-4 (16) tăng và giảm khi mômen xoắn bên ngoài giảm. Mômen xoắn bên ngoài ước tính càng lớn và số lượng khớp bị tác động của ngoại lực càng nhiều thì giá trị đầu ra SVMR càng lớn.

b) Phát hiện va chạm với 1D CNN

- Thiết kế đầu vào mạng: Tín hiệu được lấy mẫu trong cửa sổ thời gian $t_W=50$ ms ở khoảng thời gian lấy mẫu $t_I=10$ ms và được chuyển đổi thành ma trận $R \in \mathbb{R}^{n \times (N+1)}$. Ma trận đặc trưng đầu vào mạng $Z(t)$ được định nghĩa là:

$$Z = \begin{bmatrix} | & | & & | \\ |r_{J_1}^T| & |r_{J_2}^T| & \dots & |r_{J_6}^T| \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times (N+1)} \quad (18)$$

trong đó $|r_{J_i}| \in \mathbb{R}^{1 \times (N+1)}$ biểu thị giá trị của hàng thứ i của ma trận R .

- Thiết kế kiến trúc: Kiến trúc CNN được mô tả trong Hình 3 chứa bốn lớp tích chập 1-D, một lớp làm phẳng, một lớp fully connected và một bộ lọc đầu ra. Hai trong số các lớp tích chập 1-D sử dụng cùng một phần đệm, trong khi hai lớp còn lại sử dụng phần đệm hợp lệ. Sau khi đi qua các lớp làm phẳng và được kết nối đầy đủ và bộ lọc đầu ra, giá trị chỉ số va chạm nhị phân được xuất ra, loại bỏ quá trình điều chỉnh ngưỡng phát hiện. Tất cả các lớp đều sử dụng hàm kích hoạt *relu*.

c) Lọc đầu ra

Chúng tôi thực hiện lọc đầu ra để giảm dương tính giả (cảnh báo sai). Va chạm chỉ được cảnh báo nếu đầu ra của phương pháp phát hiện va chạm liên tục là *True* trong khoảng thời gian t_c ms. Bộ lọc đầu ra loại bỏ các lỗi (cảnh báo sai với thời gian kéo dài trong phạm vi $1 - t_c$ ms) giúp phát hiện va chạm hiệu quả hơn. Chúng tôi đặt $t_c=3$ ms cho phương pháp SVMR và $t_c=4$ ms cho CNN.

V. THỬ NGHIỆM VÀ KẾT QUẢ

Thử nghiệm của chúng tôi dùng *PRAUC* (Area Under the Precision-Recall Curve) làm tiêu chí để đánh giá hiệu suất phát hiện va chạm. *Precision* và *Recall* được định nghĩa như sau:

$$Precision = \frac{TP}{TP + FP} \quad (19)$$

và

$$Recall = \frac{TP}{TP + FN} \quad (20)$$

với TP biểu thị số lượng dương tính thật, FP biểu thị số lượng dương tính giả và FN biểu thị số lượng âm tính giả (va chạm được coi là dương tính). *Precision* cao thể hiện tỉ lệ dương tính giả thấp trong khi *Recall* cao thể hiện tỉ lệ âm tính giả thấp. *PRAUC* cao thể hiện cả *Precision* và *Recall* cao.

Trước tiên, chúng tôi đánh giá *PRAUC* của SVMR với các ngưỡng phát hiện khác nhau (α); đánh giá *PRAUC* của CNN với các tham số bộ lọc liên tục đầu ra khác

nhau (t_c). Sau đó, đối với mỗi phương pháp, với ngưỡng cố định và tham số bộ lọc đầu ra cố định, chúng tôi đo độ trễ phát hiện trung bình và số lần phát hiện không thành công đối với các va chạm. Đối với SVMR, trong khi tăng dần ngưỡng phát hiện α từ $\alpha = -0.08$ (tất cả các nhãn được dự đoán là 1) lên $\alpha = 0.98$ (tất cả các nhãn được dự đoán là 0), chúng tôi tính toán cặp *Precision*, *Recall* cho từng giá trị ngưỡng và vẽ đường cong như trong Hình 4.

Tiếp theo, đánh giá *PRAUC* của CNN với tham số bộ lọc đầu ra t_c thay đổi từ $t_c = 1$ ms (không áp dụng lọc) đến $t_c = 619$ ms (tất cả các nhãn được dự đoán là 1 được lọc thành 0). Chúng tôi lưu ý rằng do quá trình lọc liên tục chỉ có thể biến đổi các giá trị 1 được dự đoán thành 0 (điều ngược lại là không thể; không thể loại bỏ các phủ định sai hiện có), nên *Recall* tối đa không phải là 1. Như minh họa trong Hình 4, SVMR có *PRAUC* là 80.95% trong khi CNN có *PRAUC* là 82.12%. Như vậy, cả hai phương pháp học máy có giám sát đều phát hiện hiệu quả các va chạm và không va chạm đối với nhiều ngưỡng phát hiện và tham số bộ lọc đầu ra.

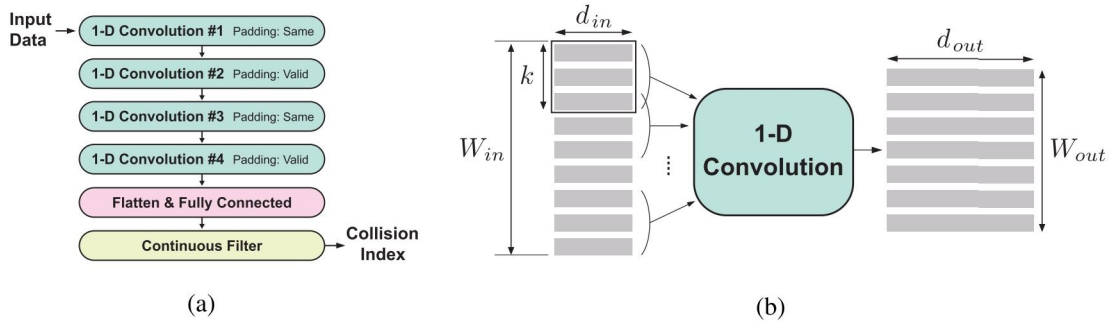
Chúng tôi lập trình thử nghiệm bằng ngôn ngữ Python. Các phương pháp đề xuất đã khắc phục được những trở ngại về sự không chắc chắn của các tham số trong mô hình động lực học và các hiệu ứng không được mô hình hóa (ma sát, nhiễu đo cảm biến).

VI. KẾT LUẬN VÀ ĐỊNH HƯỚNG NGHIÊN CỨU PHÁT TRIỂN

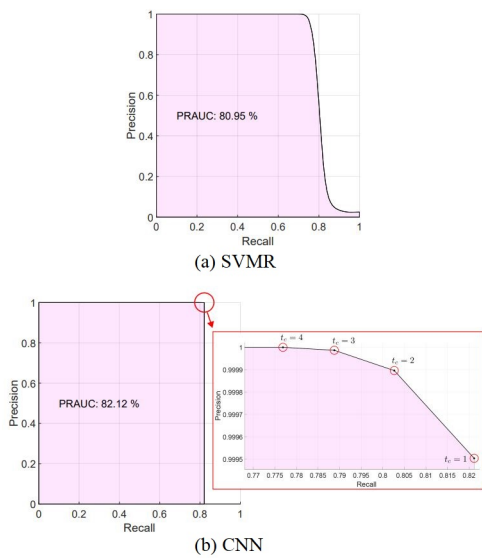
Phương pháp phát hiện va chạm dựa trên SVMR và 1D CNN do chúng tôi phát triển đã phát hiện va chạm của cánh tay robot 6-DoF CURA6 khá hiệu quả. Ưu thế nổi bật trong phương pháp đề xuất là chỉ yêu cầu phép đo các cảm biến dòng điện của động cơ cùng với mô hình động lực học của robot; không cần mô hình hóa hoặc xác định các mômen ma sát trong các khớp.

Qua thử nghiệm trên bộ dữ liệu của robot CURA6, phương pháp dựa trên SVMR chỉ yêu cầu điều chỉnh một tham số không đổi, trong khi phương pháp dựa trên 1D CNN không yêu cầu bất kỳ điều chỉnh nào. Chúng tôi so sánh và nhận thấy rằng phương pháp dựa trên 1D CNN phát hiện va chạm cho kết quả nhanh phương pháp dựa trên SVMR hơn khoảng 9 ms; trong khi phương pháp dựa trên SVMR yêu cầu về dữ liệu va chạm ít hơn để huấn luyện so với phương pháp dựa trên 1D CNN. Phương pháp dựa trên 1D CNN thích hợp với ngữ cảnh dữ liệu huấn luyện phong phú, còn phương pháp dựa trên SVMR phù hợp với ngữ cảnh dữ liệu hạn chế.

So với kết quả nghiên cứu trong [2] và [6], phương pháp của chúng tôi chiếm ưu thế trong ngữ cảnh xử lý tác động



Hình 3. Kiến trúc mạng CNN



Hình 4. (a) PRAUC của SVMR; (b) PRAUC của CNN

của các tải trọng khác nhau. Tuy nhiên, việc xác thực cho các tải trọng ngẫu nhiên không xác định vẫn là một bài toán nan giải. Hơn nữa, bài toán phát hiện va chạm của hai hay nhiều cánh tay robot rất phức tạp và cần được nghiên cứu chuyên sâu hơn. Đối với các robot cộng tác được sản xuất hàng loạt, một vấn đề thực tế khác là các quy trình cần thiết để phát hiện va chạm hiệu quả phải được nhân rộng trên quy mô lớn. Đây sẽ là những chủ đề mở cho nghiên cứu trong tương lai của chúng tôi.

VII. LỜI CẢM ƠN

Để hoàn thành bài báo này, chúng tôi xin chân thành cảm ơn sự hỗ trợ của nhóm nghiên cứu Intema, các tác giả M. Czubenko, K. M. Park và cộng sự.

TÀI LIỆU THAM KHẢO

- [1] A. Cirillo, F. Ficuciello, C. Natale, S. Pirozzi, and L. Villani, "A conformable force/tactile skin for physical human-robot

- interaction", *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 41-48, Jan. 2016.
- [2] M. Czubenko and Z. Kowalczyk, "A simple neural network for collision detection of collaborative robots", *Sensors*, vol. 21, no. 12, pp. 4235-4254, 2021.
- [3] C. Gaz, E. Magrini, and A. De Luca, "A model-based residual approach for human-robot collaboration during manual polishing operations", *Mechatronics*, vol. 55, pp. 234-247, 2018.
- [4] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective", *J. Mach. Learn. Res.*, vol. 2, no. Dec, pp. 299-312, 2001.
- [5] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification", *IEEE Trans. Robot.*, vol. 33, no. 6, pp. 1292-1312, Dec. 2017.
- [6] Y. J. Heo, D. Kim, W. Lee, H. Kim, J. Park, and W. K. Chung, "Collision detection for industrial collaborative robots: A deep learning approach", *IEEE Robot. and Autom. Lett.*, vol. 4, no. 2, pp. 740-746, Apr. 2019.
- [7] ISO 10218-1/2:2011, Robots and robotic devices - Safety requirements for industrial robots, *International Organization for Standardization*, Geneva, Switzerland, 2011.
- [8] Y. Jiang, Y. Wu, X. Wu and B. Zhao, "Research on Collision Detection of Collaborative Robot using improved Momentum-based Observer", *IEEE IAS Global Conference on Emerging Technologies (GlobConET)*, London, United Kingdom, pp. 1-6, May. 2023.
- [9] Y. LeCun et al. "Gradient-based learning applied to document recognition", *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.
- [10] K. M. Lynch and F. C. Park, *Modern Robot*, Cambridge University Press, 2017.
- [11] K. Narukawa, T. Yoshiike, K. Tanaka, and M. Kuroda, "Real-time collision detection based on one class SVM for safe movement of humanoid robot", *Proc. IEEE-RAS 17th Int. Conf. Humanoid Robot*, pp. 791-796, Nov. 2017.
- [12] D. Palaz, R. Collobert, and M. M. Doss, "Estimating phoneme class conditional probabilities from raw speech signal using convolutional neural networks", *Proc. Interspeech*, pp. 1766-1770, 2013.
- [13] K. M. Park, J. Kim, J. Park, and F. C. Park, "Learning-Based Real-Time Detection of Robot Collisions Without Joint Torque Sensors", *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 103-110, Jan. 2021.
- [14] A.-N. Sharkawy, P. N. Koustoumpardis, and N. Aspragathos, "Neural network design for manipulator collision detection based only on the joint position sensors", *Robotica*, vol. 38, no. 10, pp. 1737-1755, 2020.

- [15] A.-N. Sharkawy, P. N. Koustoumpardis, and N. Aspragathos, "Human-robot collisions detection for safe human-robot interaction using one multi-input-output neural network", *Soft Comput.*, vol. 24, no. 9, pp. 6687-6719, 2020.
- [16] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression", *Statist. Comput.*, vol. 14, no. 3, pp. 199-222, 2004.
- [17] M. Strohmayer, H. Wörn, and G. Hirzinger, "The DLR artificial skin Step I: Uniting sensitivity and collision tolerance", *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 1012-1018, 2013.
- [18] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications", *Mechatronics*, vol. 55, pp. 248-266, 2018.
- [19] Z. Zhang, K. Qian, B. W. Schuller, and D. Wollherr, "An online robot collision detection and identification scheme by supervised learning and bayesian decision theory", *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 3, pp. 1144-1156, 2021.
- [20] D. Zurlò, T. Heitmann, M. Morlock and A. De Luca, "Collision Detection and Contact Point Estimation Using Virtual Joint Torque Sensing Applied to a Cobot", *IEEE International Conference on Robotics and Automation (ICRA)*, London, United Kingdom, pp. 7533-7539, 2023.

SƠ LƯỢC VỀ TÁC GIẢ



Nguyễn Văn Hưng tốt nghiệp Cử nhân ngành Công nghệ thông tin tại Trường Đại học Sư phạm Đà Nẵng năm 2008, Thạc sĩ ngành Hệ thống thông tin tại Học viện Kỹ thuật quân sự năm 2012. Nghiên cứu sinh ngành Hệ thống thông tin tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Hiện công tác tại Khoa Công nghệ thông tin, Trường Đại học Kỹ thuật – Hậu cần Công an nhân dân.

Lĩnh vực nghiên cứu: Đa phương tiện, Mô phỏng, Thực tại ảo, Hệ thống thông tin.

Điện thoại: 0946.555.189

E-mail: nghiemvanhung1985@gmail.com



Nguyễn Văn Căn nhận học vị Tiến sĩ chuyên ngành Cơ sở toán học cho tin học năm 2016. Hiện công tác tại Trường Đại học Kỹ thuật – Hậu cần Công an nhân dân.

Lĩnh vực nghiên cứu: Công nghệ nhận dạng, Thị giác máy tính, Xử lý ảnh, Đa phương tiện, Học máy, An ninh mạng, An

toàn thông tin.

Điện thoại: 0986.919.333

E-mail: nguyenvancan@gmail.com



Trịnh Hiền Anh tốt nghiệp Thạc sĩ ngành Công nghệ phần mềm tại Đại học Công nghệ-ĐHQG Hà Nội năm 2011. Nghiên cứu sinh tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Hiện công tác tại Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Lĩnh vực nghiên cứu quan: Đa phương tiện, Công nghệ mô phỏng, Thực tại ảo, Hệ thống thông tin.

Điện thoại: 0989.197.288

E-mail: hienanh@ioit.ac.vn



Đặng Văn Đức nhận học vị Tiến sĩ năm 1996, nhận chức danh Phó Giáo sư năm 2002. Hiện công tác tại Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Lĩnh vực nghiên cứu: Đa phương tiện, GIS, Công nghệ phần mềm, Phân tích và thiết kế hệ thống thông tin.

Điện thoại: 0912.223.163

E-mail: dvduc@ioit.ac.vn