**GIỚI THIỆU TỔNG QUAN**

Robot là phương tiện sản xuất thay thế con người dựa trên tính ổn định và sự hiệu quả của chúng. Chúng không bao giờ mệt, không bị thương cũng như không bị ảnh hưởng bởi cảm xúc.

Robot có những dạng như: Humanoid Robots, Collaborative Robots, Mobile Robots và Robot Manipulators,… Trong các dạng robot kể trên, cánh tay robot thường là mục tiêu nghiên cứu và phát triển trong các đồ án học thuật.

Một số sản phẩm đã được nghiên cứu và phát triển có liên quan đến cánh tay robot như Stretch - Mobile Robot được phát triển bởi Boston Dynamics, là một dạng cánh tay robot 7 bậc tự do với bệ có thể di chuyển và thực hiện nhiệm vụ không cần sự can thiệp từ con người, thường được vận hành trong nhà kho giúp bốc xếp thùng hàng, SWIFTI – Cobot của ABB, một cánh tay robot 7 bậc tự do với đặc điểm đạt tốc độ cao trong thao tác, cụ thể là tốc độ tối đa của TCP (Tool Center Point) của SWIFTI CRB 1300 là 6.2 m/s, nhanh gấp 6 lần các cobot cùng phân khúc v.v…..

Với tính dụng mạnh mẽ trong các lĩnh vực, em chọn đề tài thiết kế và điều khiển cánh tay robot 5 bậc tự do nhằm hiểu rõ lý thuyết và kết hợp với thuật toán nhận diện đặc trưng SURF phục vụ cho ứng dụng gắp vật dựa vào vật mẫu.

**NHIỆM VỤ ĐỀ TÀI**

Các nhiệm vụ đề tài em đề ra cho đồ án này bao gồm:

* Xây dựng lý thuyết động học và hoạch định quỹ đạo cho cánh tay robot.
* Xây dựng giải thuật xử lý ảnh kết hợp với SURF để nhận diện nắp chai giống với nắp chai mẫu.
* Kết nối và vận hành hệ thống.
* Đánh giá kết quả.

**SƠ LƯỢC HỆ THỐNG**

Tiếp đến, em xin đến với phần xây dựng hệ thống. Để mọi người có cái nhìn bao quát về những gì em sắp trình bày, em xin trước hết trình bày về sơ đồ sơ lược hệ thống.

Hệ thống của em gồm 2 phần chính là: phần xử lý ảnh được thực hiện trên Raspberry và phần điều khiển robot sẽ được thực hiện trên STM32.

Hai phần sẽ giao tiếp bằng giao thức truyền thông UART nhằm để truyền dữ liệu là cặp tọa độ (x, y) của nắp chai cần gắp.

Trên STM32, phần điều khiển robot sẽ gồm 2 task chính được chạy trên hệ điều hành RTOS là: UART Task dùng để xử lý dữ liệu nhận được từ Raspberry và phần Robot Task thực hiện các tác vụ chính liên quan tới robot.

Hai task sẽ được đặt độ ưu tiên ngang nhau và sẽ được quản lý quyền truy cập vào tài nguyên chung thông qua semaphore.

Ban đầu, Raspberry sẽ đọc tọa độ từ camera và đóng gói thành frame truyền và gửi đi qua UART đến STM32. Tại STM32, frame dữ liệu sẽ được đưa vào Message Queue và kích hoạt task UART hoạt động để tiến hành xử lý dữ liệu. Ở task này, frame truyền sẽ được xử lý và đưa ra kết quả là cặp tọa độ dưới dạng float cho task Robot thực hiện. Sau khi dữ liệu được xử lý xong, task UART sẽ giải phóng semaphore để nhường quyền hoạt động cho Robot Control Task. Robot Control Task khi này sẽ có quyền truy cập vào nguồn tài nguyên chung để lấy tọa độ về để tiến hành vận hành robot. Task này sẽ dùng tọa độ cần đến để tính toán động học nghịch ra các góc tới, từ đó hoạch định quỹ đạo cho góc tới và cấp xung các động cơ servo chạy. Sau khi chạy xong STM32 sẽ gửi tín hiệu yêu cầu Raspberry truyền tiếp tọa độ mới.

Và đó là phần sơ lược về hệ thống.

**CÁNH TAY ROBOT**

Tiếp đến em sẽ nói vào phần Robot. Từ cánh tay robot thực tế, em xác định giới hạn quay và hướng quay của các khớp thông qua các động cơ servo. Em sẽ vẽ lại và tiến hành đặt trục cho cánh tay robot. Sau đó em lập bảng để chuyển các góc theta khi tính xong lý thuyết thành các góc theta thực tế để truyền vào động cơ.

Ở bài toán này, em xác định hướng gắp của cánh tay robot như hình vẽ, từ đó em tính ra ma trận xoay của end-effector so với base.

Tiếp đến em sẽ tính toán không gian làm việc của robot, khớp thứ nhất có giới hạn quay chính bằng động cơ servo của khớp đó, nên em sẽ vẽ ra được không gian làm việc của robot khi chiếu lên mặt phẳng Oxy như hình.

Em đặt giới hạn góc quay cho theta 2 và theta 3 như sau:

Tuy nhiên khi z = 0 em có tính toán động học nghịch để ra góc của các khớp thì nhận thấy góc theta 2 không đạt được đến 180 độ vì vậy giới hạn của theta 2 sẽ được đặt lại như sau:

Em đặt một hệ tọa độ mới là DOz với D là căn của tổng x bình và y bình, D lấy dấu theo x. Ứng với mỗi giá trị của góc theta 2, theta 3 sẽ có ngưỡng từ 60 độ đến 150 độ. Vì vậy để vẽ không gian làm việc, em cố định góc theta 3 ở 150 độ và cho thay đổi giá trị theta 2 dần, em sẽ vẽ được đường màu đỏ, tương tự em cố định góc theta 3 ở 60 độ và cho thay đổi giá trị theta 2 dần, em sẽ vẽ được đường màu xanh. Không gian làm việc là miền được giới hạn bởi 2 đường cong và trục D.

Sau khi xác định không gian làm việc, em sẽ đến phần tính toán động học nghịch của cánh tay, từ hệ trục em đặt ban đầu em xác định được bảng DH dựa vào các quy tắc thành lập bảng. Và sau đó em dùng phương pháp hình học tính ra được các phương trình của các góc khớp.

Cuối cùng ở phần hoạch định quỹ đạo, em sẽ dùng LSPB bậc 3 để hoạch định quỹ đạo cho góc các khớp xoay gồm 3 giai đoạn, giai đoạn tăng tốc chậm dần đều, giai đoạn tăng đều và giai đoạn giảm chậm dần đều.

**PHẦN XỬ LÝ ẢNH**

Phần hai sẽ là giải thuật xử lý ảnh. Em cũng sẽ xin trình trình bày về sơ đồ giải thuật để giải thích sơ về phần này.

Sơ đồ bắt đầu khi camera bắt được 1 frame ảnh, thuật toán SURF sẽ được áp dụng lên frame để tìm kiếm các đặc trưng và khớp chúng lại với nhau, đặc trưng ở bài toán này chính là những đặc điểm về hình vẽ, màu sắc, hướng đặt của hình trên nắp chai, các cặp đặc trưng khớp tốt từ đó cho ra các good matches ( good matches tìm được như thế nào em sẽ giải thích ở phần sau), nếu tại một khu vực có số lượng good matches lớn hơn hoặc bằng một số lượng cố định nhằm để đánh giá vật đó có giống vật mẫu không, thì ta sẽ áp dụng homography để khoanh vùng và đánh dấu đó là ROI (Region of Interest) tức là vùng cần lưu tâm. Vùng này sau đó sẽ qua các giai đoạn xử lý ảnh để đưa về ảnh nhị phân để tìm đường viền nếu đó là hình tròn. Từ tập dữ liệu điểm của đường viền, em sẽ tìm ra tọa độ tâm và bán kính và kết thúc sơ đồ giải thuật.

Đầu tiên là SURF, SURF sẽ trả về 2 thông tin là keypoint tức điểm đặc trưng với thông tin về tọa độ, kích thước, hướng xoay, v.v…. và descriptors để mô tả vùng lân cận của điểm đặc trưng. Descriptor của frame ảnh sẽ được so với descriptor của ảnh mẫu, dựa trên phương pháp đánh giá của Lowe là Lowe’s ratio test, tỷ lệ khoảng cách giữa cặp đặc trưng tốt nhất so với cặp đặc trưng tốt thứ hai nếu nhỏ hơn hoặc bằng một giá trị cụ thể là 0.8 thì cặp đặc trưng tốt nhất sẽ là good\_matches.

Ví dụ, ta bốc một điểm đặc trưng trên ảnh mẫu là A, t khớp được A với B1 và B2, tỷ lệ khoảng cách A và B1 so với A và B2 bằng 0.8, vậy A và B1 là good\_matches.

Dạ đây là hình ảnh sử dụng thuật toán SURF để tìm đặc trưng. Tiếp đến là tìm Homography để đánh dấu ROI vùng cần lưu tâm vì ảnh mẫu là hình chụp trực diện khác với khi nhận hình ảnh qua live camera sẽ có nhiều hướng và góc nhìn, vì vậy cần ánh xạ ảnh gốc qua frame ảnh cần tìm để xác nhận ROI.

Dạ đây là hình nắp chai bị chênh lên bởi miếng kim loại nhưng thuật toán vẫn khoanh vùng được.

Khi có được ROI em tiến hành xử lý ảnh, từ ảnh gốc, em sẽ loại bỏ nhiễu bằng bộ lọc trung bình (Box Filter), rồi đến tăng độ tương phản để tách vật và nền, sau đó là phân ngưỡng dựa trên ngưỡng tĩnh để đưa ảnh về ảnh nhị phân.

Có ảnh nhị phân, em sẽ dễ dàng tìm được các tọa độ của đường viền nắp chai bằng hàm findContours trong opencv và tính tọa độ trọng tâm dựa trên các công thức như sau

**PHẦN ĐIỀU KHIỂN ĐỘNG CƠ**

Tiếp theo em xin nói đến phần điều khiển động cơ, động cơ của em được điều khiển theo vòng hở, và thứ tự điều khiển sẽ theo sơ đồ sau:

Khi muốn di chuyển đến vị trí mong muốn, 2 khớp cuối sẽ chuyển động trước rồi đến 3 khớp đầu tiên.

Khi di chuyển về vị trí home, khớp thứ hai sẽ chuyển động trước rồi đến khớp thứ nhất và khớp thứ ba, cuối cùng là hai khớp cuối.

**THIẾT KẾ PHẦN CỨNG**

Em sang nội dung thiết kế phần cứng, 6 động cơ sẽ được điều khiển thông qua PCA 9685, PCA sẽ giao tiếp I2C với STM32. STM32 và Raspberry sẽ giao tiếp bằng giao thức UART thông qua CP2102. Camera em sử dụng là Camera Module V3 của Raspberry với FOV là 75 độ. Hệ thống sẽ được vận hành với nguồn tổ ong 5V 20A.

**ĐÁNH GIÁ KẾT QUẢ**

Và đây là phần chạy mô hình của em,

Sau khi chạy mô hình em tiến hành đánh giá sai số dựa trên quy trình sau:

Sai số em cần đánh giá là sai số về tọa độ của nắp chai với đơn vị cm

* Em sẽ đo tọa độ nắp chai bằng thước ghi vào bảng tọa độ thực
* Camera sẽ đọc tọa độ và ghi vào tọa độ camera.
* Robot sẽ di chuyển đến tọa độ cung cấp và ghi vào cột tọa độ Robot.

Sau đó em tiến hành đánh giá sai số camera và robot.

Bảng giá trị các lần thực nghiệm em xin đính kèm phía sau quyển slide em gửi các thầy lúc đầu giờ ạ.

Sau đó là hiệu chỉnh, em hiệu chỉnh sai số camera bằng căn chỉnh lại góc nhìn camera bằng cách xê dịch chân tripod hoặc sẽ đo lại FOV của camera cho chính xác.

Đối với sai số robot, em tiến hành bù góc cho servo hoặc bù tọa độ trực tiếp vào code.

Sau khi hiệu chỉnh em có được bảng số liệu dựa trên quy trình đánh giá trước đó.

Từ bảng số liệu, sai số mới sẽ được tính như trên hình.

**KẾT LUẬN**

Hạn chế về mặt đề tài, điều khiển động cơ vòng hở.

Động cơ không có feedback dẫn đến sai số vẫn còn lớn, tỉ lệ gắp trúng chưa đạt như mong muốn. Việc này dẫn đến khi đánh giá sai số muốn hiệu chỉnh kết quả, phải bù trực tiếp tọa độ hoặc góc vào động cơ servo.

Phát triển đề tài, thêm camera để đọc feedback động cơ để có thể điều khiển vòng kín. Từ đó có thể phát triển các phương pháp điều khiển giúp chuyển động mượt mà hơn như MPC, Back Stepping.

Dùng camera quét hình ảnh 3D của vật và xác định trọng tâm, từ đó xác định hướng gắp phù hợp với vật.