

LỜI CẢM ƠN

Khoa học kỹ thuật luôn đóng vai trò quan trọng trong đời sống con người, là nền tảng và động lực của công cuộc công nghiệp hóa, hiện đại hóa đất nước. Muốn vậy chúng ta phải có những kiến thức nhất định về kỹ thuật, công nghệ để có thể tiếp thu được những tiến bộ khoa học kỹ thuật và từ đó có thể vận dụng những kiến thức đã học vào thực tiễn một cách có hiệu quả.

Vào học kỳ cuối, việc hoàn thành luận văn tốt nghiệp là nhiệm vụ và cũng là cơ hội thử thách năng lực của sinh viên. Với việc thực hiện đề tài: “Điều khiển mô hình xe gấp khối vật sử dụng Computer Vision”, em đã có cơ hội vận dụng những kiến thức về chuyên ngành đã được học tại trường cũng như học hỏi nhiều kiến thức mới, kiến thức trong thực tế để giải quyết một vấn đề cụ thể.

Em xin cảm ơn tất cả quý Thầy/Cô trong Khoa Cơ Khí cũng như quý Thầy/Cô ở Trường Đại học Bách Khoa TP. HCM đã trang bị những kiến thức quý báu cũng như đã giúp đỡ chúng em trong suốt quá trình học tập tại trường. Đặc biệt em xin bày tỏ lòng biết ơn sâu sắc đối với Thầy Đoàn Thế Thảo, người đã tận tình chỉ dạy cho em phương pháp học tập, làm việc, nghiên cứu khoa học, cung cấp rất nhiều kiến thức chuyên sâu để thực hiện đề tài. Trong quá trình làm luận văn em đã tiếp thu từ Thầy rất nhiều điều quý báu là hành trang cho em bước vào công việc và cuộc sống sau này.

Sinh viên

Nguyễn Đình Huy

TÓM TẮT LUẬN VĂN

Robot nói chung và mobile robot nói riêng là một ngành khoa học phát triển từ giữa thế kỷ XX, và có tốc độ phát triển ngày càng nhanh chóng nhờ vào các thành tựu của khoa học kỹ thuật. Với sự phát triển đa dạng và ngày càng hoàn thiện, mobile robot được ứng dụng trong rất nhiều các lĩnh vực như trong công nghiệp, y tế, quân sự, thám hiểm, phục vụ sinh hoạt trong gia đình.

Việc tích hợp Computer Vision (CV) vào mobile robot đem lại cho robot khả năng nhận biết môi trường làm việc rất tốt, nhờ đó, tăng khả năng linh hoạt trong khi hoạt động. Khi đó mobile robot có thể hoạt động, đáp ứng tốt hơn với sự thay đổi của môi trường, có thể tránh được vật cản, thực thi nhiệm vụ định vị và gấp đối tượng, phát hiện chuyển động và theo dõi giám sát...

Đề tài “Điều khiển mô hình xe gấp khối vật sử dụng Computer Vision” thực hiện giải quyết những nhiệm vụ như sau: Tìm hiểu về mobile robot sử dụng CV, cơ sở lý thuyết về đo khoảng cách sử dụng CV, thiết kế và điều khiển mobile robot sử dụng CV để nhận diện vật thể, tìm vật thể cần gấp, xác định vị trí của vật (sử dụng hệ 2 camera để xác định khoảng cách đến vật, xác định hướng di chuyển của robot), điều khiển mobile robot di chuyển và gấp vật. Loại robot này không những có nhiều ứng dụng trong nghiên cứu khoa học, khám phá, đời sống... mà những nội dung nghiên cứu được trong quá trình nghiên cứu robot này có thể được ứng dụng trong nhiều loại robot, hệ thống hữu ích khác.

MỤC LỤC

Lời cảm ơn.....	i
Tóm tắt luận văn	ii
Mục lục.....	iii
Danh sách hình vẽ	vi
Danh sách bảng biểu.....	viii
Chương 1: Tổng quan	1
1.1 Giới thiệu về mobile robot	1
1.2 Giới thiệu Computer Vision và mobile robot sử dụng Computer Vision.....	3
1.2.1 Thị giác máy tính-Computer Vision-CV	3
1.2.2 Mobile robot sử dụng CV	5
1.3 Mục tiêu, nhiệm vụ và phạm vi đề tài.....	5
1.3.1 Mục tiêu và nhiệm vụ đề tài	6
1.3.2 Phạm vi đề tài	6
1.4 Nội dung và tổ chức luận văn.....	7
1.4.1 Nội dung.....	7
1.4.2 Tổ chức luận văn	7
Chương 2: Phương án thiết kế	9
2.1 Giới thiệu.....	9
2.2 Lựa chọn thư viện thị giác máy tính	9
2.3 Lựa chọn hình dáng, kích thước giới hạn của robot.....	10
2.4 Lựa chọn phương án di chuyển của robot.....	10
2.4.1 Robot sử dụng bánh xe	10
2.4.2 Robot dùng chân.....	12
2.4.3 Lựa chọn phương án di chuyển	13
2.5 Lựa chọn phương án thiết kế tay máy.....	14
2.5.1 Số bậc tự do của tay máy	15
2.5.2 Cấu trúc động học	15
2.5.3 Khâu cuối	16
2.5.4 Phương án dẫn động của tay máy.....	17
2.6 Kết luận	19

Chương 3: Mô hình camera, đo khoảng cách sử dụng CV	20
3.1 Giới thiệu	20
3.2 Mô hình camera, và các thông số của quá trình calib camera	20
3.2.1 Mô hình camera	20
3.2.2 Thông số camera	21
3.2.3 Lens Distortions	23
3.2.4 Calibrate camera	25
3.3 Đo khoảng cách sử dụng CV	28
3.3.1 Triangulation	29
3.3.2 Epipolar Geometry	32
3.3.3 Stereo Calibration, Rectification	33
3.3.4 Stereo Correspondence	38
3.4 Xác định vị trí của vật thể	39
3.5 Kết luận	41
Chương 4: Tính toán và thiết kế phần cứng	43
4.1 Giới thiệu	43
4.2 Yêu cầu thiết kế	43
4.3 Thiết kế khung xe	44
4.4 Lựa chọn, thiết kế các nhóm chức năng	45
4.4.1 Động cơ di chuyển	45
4.4.2 Hệ 2 camera	46
4.4.3 Tay máy	47
4.4.4 Laptop, pin và mạch điện	50
4.5 Bố trí các nhóm chức năng	51
4.6 Kết luận	52
Chương 5: Thiết kế hệ thống điều khiển	53
5.1 Giới thiệu	53
5.2 Sơ đồ khối hệ thống điều khiển của mobile robot	53
5.3 Thiết kế mạch điều khiển và mạch công suất của robot	54
5.3.1 Mạch vi điều khiển	55
5.3.2 Mạch nguồn	56

5.3.3 Mạch Opto cách ly.....	57
5.3.4 Mạch driver cho động cơ di chuyển	58
5.4 Kết luận	59
Chương 6: Động học robot và giải thuật điều khiển	60
6.1 Giới thiệu.....	60
6.2 Phương trình động học của mobile robot.....	61
6.3 Phương trình động học tay máy.....	63
6.4 Giải thuật điều khiển	65
6.5 Kết luận	69
Chương 7: Thực nghiệm và đánh giá kết quả	70
7.1 Giới thiệu.....	70
7.2 Đo khoảng cách sử dụng OpenCV	70
7.3 Điều khiển tay máy	75
7.4 Điều khiển mobile robot di chuyển và gấp vật.....	76
7.5 Kết luận	77
Chương 8: Tổng kết và hướng phát triển đề tài.....	78
8.1 Đánh giá kết quả đạt được.....	78
8.2 Hướng phát triển của đề tài	79
Tài liệu tham khảo	80

DANH SÁCH HÌNH VẼ

Hình 1.1: Một số loại mobile robot trong thực tế	2
Hình 1.2: Ứng dụng CV trong robot tự hành- CVL, Linkkoping University	4
Hình 1.3: Một số loại mobile robot sử dụng thị giác	5
Hình 2.1: Các loại bánh dẫn hướng thường được sử dụng	11
Hình 2.2: Một số dạng mobile robot dùng chân	13
Hình 2.3: Robot 2 bánh kết hợp với 1 bánh caster để giữ thăng bằng	14
Hình 2.4: Sơ đồ nguyên lý tay máy	16
Hình 2.5: Cơ cấu khâu cuối tay máy	17
Hình 3.1: Mô hình camera Pinhole [1]	20
Hình 3.2: Hệ tọa độ camera [1]	21
Hình 3.3: Ma trận R , vector \vec{T} [1]	23
Hình 3.4: Radial Distortion [1]	24
Hình 3.5: Tangential Distortion bị gây ra do lens không hoàn toàn song song với mặt phẳng ảnh [1]	25
Hình 3.6: Thực hiện calibrate cho camera- Hiển thị các corner của chessboard	27
Hình 3.7: Lưu đồ giải thuật đo khoảng cách	29
Hình 3.8: Mô hình 2 camera đã được undistorted, rectified. O_l, O_r là tâm phép chiếu; P là điểm thực cần được đo khoảng cách Z [1]	30
Hình 3.9: Hệ tọa độ được OpenCV sử dụng [1]	31
Hình 3.10: Epipolar plane là mặt phẳng tạo bởi điểm P và hai tâm chiếu của hai camera; hai điểm epipole là các giao điểm của đường thẳng nối hai tâm chiếu O_l, O_r với hai mặt phẳng ảnh [1]	32
Hình 3.11: Mối tương quan giữa hai camera được thể hiện trong ma trận E- chứa các thông tin về ma trận Rotation và Translation [1]	34
Hình 3.12: Tiến hành stereo calibration cho hệ camera	36
Hình 3.13: Hình ảnh trước và sau khi thực hiện bước calibration và rectification	37
Hình 3.14: Lưu đồ các bước thực hiện stereo calibration	38
Hình 3.15: Tiến hành chọn khoảng giá trị HSV để phân ngưỡng ảnh	39
Hình 3.16: Mô hình hệ camera	40
Hình 4.1: Ví dụ về các loại khung xe mobile robot	44

Hình 4.2: Động cơ DC Leison SG555123000	46
Hình 4.3: Thiết kế của hệ camera	47
Hình 4.4: Động cơ RC servo TowerPro M995	48
Hình 4.5: Phương án thiết kế khâu cuối- cơ cấu kẹp sử dụng bánh răng để truyền động.....	49
Hình 4.6: Thiết kế tay máy	50
Hình 4.7: Bố trí các nhóm chức năng	51
Hình 4.8: Thiết kế 3D của mobile robot	52
Hình 4.9: Mô hình mobile robot thực tế và mô phỏng	52
Hình 5.1: Sơ đồ khối hệ thống điều khiển	54
Hình 5.2: Mạch Arduino Uno	55
Hình 5.3: Sơ đồ nguyên lý mạch ổn áp 5V	56
Hình 5.4: Sơ đồ nguyên lý mạch ổn áp 12V	57
Hình 5.5: Pin và mạch ổn áp 5V và 12V	57
Hình 5.6: Mạch Opto cách ly	58
Hình 5.7: Mạch driver động cơ di chuyển	58
Hình 5.8: Sơ đồ mạch điện của mobile robot	59
Hình 6.1: Mô hình hóa robot	61
Hình 6.2: Coordinate frames của tay máy robot	63
Hình 6.3: Lưu đồ giải thuật điều khiển trên máy tính	67
Hình 6.4: Lưu đồ giải thuật điều khiển trên vi điều khiển	68
Hình 7.1: Sử dụng mặt phẳng chuẩn để cân chỉnh hệ camera bằng tay	71
Hình 7.2: Các file chứa thông tin thu được sau quá trình calib	71
Hình 7.3: Giao diện chương trình tìm khoảng phân ngưỡng- Hình ảnh thu được trước và sau khi phân ngưỡng	72
Hình 7.4: Khoảng cách Z và disparity quan hệ tỉ lệ nghịch ($T = 105\text{ mm}, f = 900$)	73
Hình 7.5: Độ chính xác của hệ camera (mm/pixel) với $T = 105\text{ mm}, f = 900$	73
Hình 7.6: Hệ tọa độ tay máy, hệ camera	74
Hình 7.7: Thực nghiệm đo khoảng cách sử dụng CV	75
Hình 7.8: Thực nghiệm điều khiển tay máy gấp vật thể	76
Hình 7.9: Mobile robot di chuyển và gấp vật thể	76

DANH SÁCH BẢNG BIỂU

Bảng 2.1: Một số dạng robot 2 bánh	12
Bảng 2.2: Phương án thiết kế mobile robot	19
Bảng 3.1: Kết quả calibrate camera trái và phải	28
Bảng 4.1: Các thông số của động cơ RC servo TowerPro 995	48
Bảng 5.1: Các mạch chức năng trên robot	54
Bảng 6.1: Các thông số D-H của tay máy	63

Chương 1: Tổng quan

1.1 Giới thiệu về mobile robot:

Robot là một ngành khoa học phát triển từ giữa thế kỷ XX, và có tốc độ phát triển ngày càng nhanh chóng nhờ vào các thành tựu của khoa học kỹ thuật. Với sự phát triển đa dạng và ngày càng hoàn thiện nên robot được ứng dụng trong rất nhiều các lĩnh vực như trong công nghiệp, y tế, quân sự, thám hiểm, phục vụ sinh hoạt trong gia đình...

Mobile robot là một loại robot có thể tự di chuyển và vận động trong môi trường để thực hiện các công việc đã được lập trình. Nó có vai trò quan trọng trong ngành khoa học robot, những năm gần đây với các thành tựu trong lĩnh vực cơ khí và điện tử, mobile robot phát triển ngày càng mạnh mẽ, đa dạng về chủng loại, hoàn thiện về chức năng nên chúng ngày càng đóng một vai trò quan trọng trong cuộc sống, được sử dụng trong nhiều lĩnh vực đặc biệt là trong quân sự, cứu hộ, thám hiểm, và phục vụ sinh hoạt hằng ngày. Hiện nay, mobile robot đang được ứng dụng rất nhiều trong thực tế, từ robot vận chuyển hàng hóa trong các tòa nhà, nhà máy, cửa hàng, sân bay, thư viện; robot thực hiện việc quét dọn đường phố, robot làm việc trong môi trường nguy hiểm như núi lửa, vùng bị phóng xạ, nhà máy hạt nhân; robot an ninh, canh gác, robot do thám dùng trong quân sự; robot thực hiện việc khám phá vũ trụ, di chuyển và lấy mẫu phân tích trên các hành tinh; robot phục vụ, hỗ trợ người già, người tàn tật và robot phục vụ sinh hoạt trong gia đình...

Mobile robot rất đa dạng và ngày càng phổ biến. Có nhiều cách phân loại mobile robot như sau:

+Theo môi trường hoạt động: trên mặt đất (Hình 1.1 a, d, e, f), trong môi trường nước (Hình 1.1 c), trong không trung (Hình 1.1 b)...

+Theo phương thức di chuyển: bánh xe (Hình 1.1 a, f), chân (Hình 1.1 d), bánh xích (Hình 1.1 e)...



(a)



(b)



(c)



(d)



(e)



(f)

Hình 1.1: Một số loại mobile robot trong thực tế

1.2 Giới thiệu Computer Vision và mobile robot sử dụng Computer Vision:

1.2.1 Thị giác máy tính-Computer Vision-CV:

Thị giác máy tính là một lĩnh vực đã và đang phát triển mạnh mẽ trên thế giới hiện nay. Khái niệm về CV có liên quan tới nhiều ngành học và có nhiều hướng nghiên cứu khác nhau. Đây là một trong những hướng nghiên cứu rộng, đa dạng và có nhiều ứng dụng trong thực tế, có thể kể đến như xử lý ảnh y khoa, robot tự hành, xử lý ảnh viễn thám, phát giác và theo dõi chuyển động, an ninh, quân sự (phát giác chuyển động, nhận dạng mặt, vân tay, võng mạc, xe, chiến hạm...) cũng như áp dụng trong các ngành nghiên cứu khác...

Trong thế giới kỹ thuật số ngày nay, hình ảnh và video phổ biến khắp mọi nơi. Không những thế, với sự phát triển nhanh chóng của khoa học công nghệ, các thiết bị xử lý, tính toán trở nên không những mạnh mẽ hơn mà giá thành cũng hợp lý hơn. Điều này tạo điều kiện thuận lợi rất lớn cho việc tạo ra các ứng dụng về xử lý hình ảnh, video. Từ các loại máy tính để bàn, đến laptop, và cả các loại điện thoại di động thông minh nhỏ bé đều có sức mạnh để xử lý các tác vụ về hình ảnh, video. Do đó, thị giác máy tính là một hướng nghiên cứu đầy tiềm năng, là xu hướng nghiên cứu của nhiều kỹ sư, nhà nghiên cứu, nhà khoa học trên thế giới.

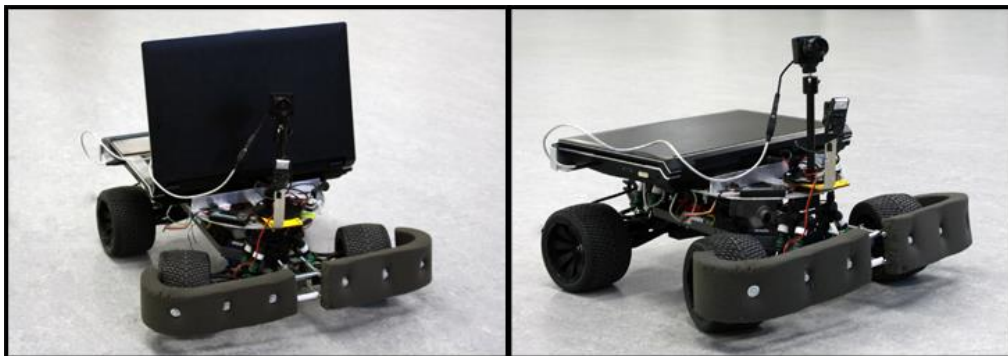
Hơn nữa, robot ngày càng đóng vai trò quan trọng trong công nghiệp và gia đình. Chúng sẽ thực hiện những công việc nguy hiểm hoặc lặp đi lặp lại, nhàm chán, và những công việc mà yêu cầu tốc độ xử lý và độ chính xác vượt quá khả năng của con người. Khi người máy ngày càng trở nên tinh vi hơn, thị giác máy tính đóng vai trò ngày càng quan trọng. Người ta sẽ đòi hỏi người máy không những phát hiện, nhận dạng các đối tượng đã biết, mà còn “hiểu” được những gì chúng “thấy” và đưa ra những hành động phù hợp. Ngày nay, phần lớn các robot hoạt động trong môi trường làm việc ít thay đổi và ứng dụng của robot bị hạn chế khi phải hoạt động trong các môi trường không thể mô tả, nắm rõ trước. Hạn chế này là do thiếu các cảm biến thích hợp để nhận biết môi trường, các yếu tố bên ngoài... Thị giác (vision) trong trường hợp này là một loại cảm biến linh động và có nhiều ưu thế lớn so với các loại cảm biến thông thường khác, nó có thể cung cấp rất nhiều thông tin về môi trường hoạt động cho robot. Robot có thị giác có ưu thế lớn trong việc nhận biết môi trường làm việc, do đó

nó được sử dụng phổ biến trong nhiều lĩnh vực như công nghiệp, quốc phòng và nghiên cứu. Đặc biệt là các hệ thống cần quan sát, theo dõi mục tiêu; hệ thống phân loại sản phẩm trong công nghiệp, những hệ thống thông minh hỗ trợ lái xe tự động, các hệ thống giám sát và theo dõi chuyển động...

Hệ thống thị giác có thể được phân thành 2 loại, theo như cấu trúc của hệ, đó là hệ có camera gắn cố định và hệ có camera gắn trực tiếp trên tay máy hoặc trên thân robot:

- + Hệ có camera gắn cố định: camera được gắn cố định so với hệ toạ độ thực, thu thập ảnh của cả mục tiêu và cả môi trường. Mục tiêu của hệ này là cung cấp tín hiệu điều khiển sao cho tay máy, robot đạt được vị trí mong muốn. Hoặc ứng dụng trong các loại hệ thống phân loại sản phẩm, đếm sản phẩm...

- + Hệ có camera gắn trực tiếp trên tay máy hoặc trên thân robot: mục đích của cấu trúc này là điều khiển tay máy (robot) sao cho ảnh của mục tiêu cố định hay di động luôn được duy trì ở vị trí mong muốn trên mặt phẳng ảnh thu được. Trên cơ sở ảnh thu được từ camera, được số hoá và tích hợp trong vòng điều khiển phản hồi, hệ thống điều khiển tay máy, robot.

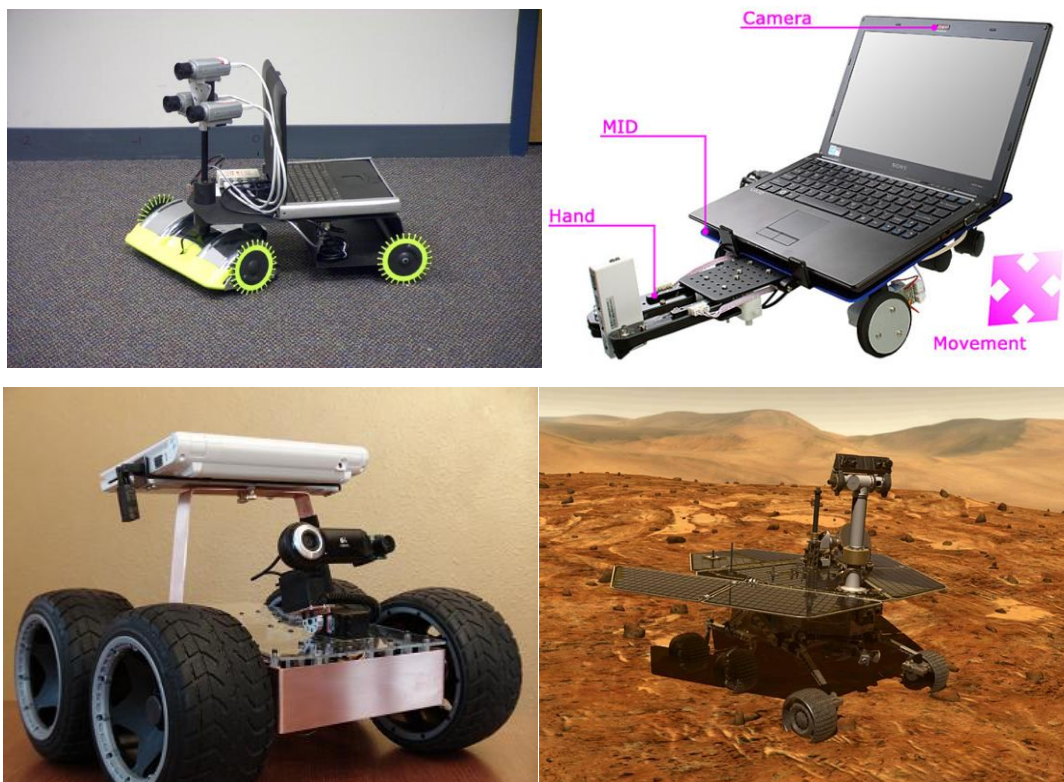


Hình 1.2: Ứng dụng CV trong robot tự hành-CVL, Linköping University

CV là một lĩnh vực mới mẻ đối với các nhà nghiên cứu ở Việt nam, nó chỉ mới xuất hiện gần đây ở trong các tổ chức quân đội và một số viện nghiên cứu, trường đại học khác. Việc áp CV vào các ứng dụng nghiên cứu khoa học được coi là một khởi đầu cho chặng đường phát triển của thị giác máy ở các trường đại học hiện nay.

1.2.2 Mobile robot sử dụng CV:

Như đã trình bày ở trên, việc tích hợp CV vào robot đem lại cho robot khả năng nhận biết môi trường làm việc rất tốt, nhờ đó, tăng khả năng linh hoạt trong khi hoạt động. Khi đó mobile robot có thể hoạt động, đáp ứng tốt hơn với sự thay đổi của môi trường, có thể tránh được vật cản, thực thi nhiệm vụ định vị và gặp đối tượng, phát hiện chuyển động và theo dõi giám sát... Các ứng dụng này chủ yếu dành cho mobile robot tự hành thông minh, sử dụng trong nghiên cứu khoa học, khám phá và thám hiểm...



Hình 1.3: Một số loại mobile robot sử dụng thị giác

1.3 Mục tiêu, nhiệm vụ và phạm vi đề tài:

Qua giới thiệu tổng quan về mobile robot cũng như ứng dụng CV vào mobile robot, có thể thấy được tầm quan trọng và mức độ phổ biến ngày càng cao của loại robot này. Đề tài “Điều khiển mô hình xe gấp khối vật sử dụng Computer Vision” được chọn để thực hiện. Mobile robot sử dụng CV để nhận diện khối vật màu, tìm khối vật màu cần gấp, xác định vị trí của vật (sử dụng hệ 2 camera để xác định khoảng cách đến vật, xác định hướng di chuyển của robot), từ đó truyền tín hiệu để điều khiển robot di chuyển và gấp vật. Loại robot này không những có nhiều ứng dụng trong

ngiên cứu khoa học, khám phá, đời sống... mà những nội dung nghiên cứu được trong quá trình nghiên cứu robot này có thể được ứng dụng trong nhiều loại robot, hệ thống hữu ích khác.

1.3.1 Mục tiêu và nhiệm vụ đề tài:

Mục tiêu và nhiệm vụ của đề tài có thể chia thành ba phần như sau:

Thứ nhất: Thiết kế và chế tạo phần cứng robot gồm khung xe và tay máy gấp vật

Thứ hai: Thiết kế và gia công mạch điều khiển và công suất cho mobile robot

Thứ ba: Xây dựng phần mềm cho robot sử dụng OpenCV, nhận dạng, đo khoảng cách đến vật, xác định hướng di chuyển robot, điều khiển robot di chuyển và gấp vật.

Như vậy, công việc chính của đề tài là:

- Tìm hiểu tổng quan về mobile robot gấp khối vật sử dụng camera
- Thiết kế mô hình mobile robot có gắn tay máy 2 DOFs
- Tìm hiểu openCV và lý thuyết điều khiển robot sử dụng 2 camera
- Thiết kế mạch điều khiển và chương trình điều khiển
- Thực nghiệm và kết luận

1.3.2 Phạm vi đề tài:

Mục tiêu của đề tài là tìm hiểu lý thuyết và thiết kế, thực hiện mô hình robot-với những yêu cầu cụ thể như sau:

- Kích thước tổng thể mô hình mobile robot khoảng $400 \times 400 \times 400 \text{ mm}$, mobile robot tìm khối vật, sử dụng Computer Vision để nhận diện vật thể, đo khoảng cách đến khối vật, xác định hướng di chuyển của robot và điều khiển mobile robot di chuyển, gấp khối vật.

- Đề tài được giới hạn trong việc thao tác với khối vật đơn giản- khối trụ màu có $R=10 \text{ mm}$, $h=100\text{mm}$, được đặt trên mặt phẳng di chuyển của robot, khối lượng vật gấp được bỏ qua.

1.4 Nội dung và tổ chức luận văn:

1.4.1 Nội dung:

Lựa chọn phương án thiết kế robot

Tính toán, thiết kế kết cấu phần đế robot và tay máy

Thiết kế khối mạch điều khiển, công suất cho robot

Tìm hiểu về thư viện OpenCV, tìm hiểu và giải quyết bài toán tìm vật, đo khoảng cách, xác định hướng cần di chuyển sử dụng OpenCV

Lập trình điều khiển, viết chương trình cho vi điều khiển và máy tính

1.4.2 Tổ chức luận văn:

Với nội dung đã nêu ở trên, luận văn được trình bày thành 8 chương chính

- Chương 1: Tổng quan

Trước hết, luận văn đi vào tìm hiểu tổng quan về mobile robot, thị giác máy tính và mobile robot ứng dụng thị giác máy tính. Đồng thời, trong chương này cũng đưa ra yêu cầu, nhiệm vụ, mục tiêu và phạm vi của đề tài

- Chương 2: Phương án thiết kế

Dựa trên yêu cầu, mục tiêu, nhiệm vụ và phạm vi của đề tài, chương 2 sẽ trình bày phân tích, lựa chọn phương án thiết kế mobile robot gấp vật sử dụng CV. Gồm lựa chọn thư viện CV, lựa chọn kích thước và hình dáng robot, lựa chọn phương án di chuyển, phương án thiết kế tay máy.

- Chương 3: Mô hình camera, đo khoảng cách sử dụng CV

Chương 3 sẽ giới thiệu về mô hình camera, các thông số của camera và trình bày lý thuyết đo khoảng cách, xác định vị trí vật thể sử dụng CV. Ở đây, ta sử dụng thư viện OpenCV để hỗ trợ giải quyết các bài toán này. Bên cạnh đó, các bước tiến hành và các thông số cần xử lý, tính toán, các hàm hỗ trợ trong thư viện OpenCV để đo khoảng cách đến khối vật màu cũng sẽ được trình bày.

- Chương 4: Tính toán và thiết kế phần cứng

Mobile robot cần thiết kế gồm hai phần chính là khung xe robot và tay máy gấp vật. Sau khi đưa ra các yêu cầu thiết kế mobile robot, chương này sẽ trình bày tính toán, thiết kế khung xe, tay máy và các nhóm chức năng của robot gồm động cơ di chuyển, hệ camera, tay máy...

- Chương 5: Thiết kế hệ thống điều khiển

Nhiệm vụ của chương này là phân tích, lựa chọn thiết kế hệ thống điều khiển cho mobile robot. Nội dung của chương 5 bao gồm thiết kế mạch điều khiển, driver và mạch nguồn cho mobile robot

- Chương 6: Động học robot và giải thuật điều khiển

Chương này tập trung vào tính toán động học của mobile robot 2 bánh dẫn động, động học tay máy. Đồng thời, đưa ra giải thuật điều khiển của mobile robot gồm giải thuật trên máy tính và giải thuật cho vi điều khiển.

- Chương 7: Thực nghiệm và đánh giá kết quả

Sau khi đã thiết kế phần cứng, mạch điện, tính toán động học robot và đưa ra giải thuật điều khiển cho robot, tiến hành thực nghiệm và đánh giá kết quả hoạt động thực nghiệm. Trong chương này, nội dung được trình bày gồm hiện thực hóa giải thuật đo khoảng cách, xác định hướng cần di chuyển của mobile robot sử dụng OpenCV, thực hiện truyền nhận qua module RS232 để điều khiển tay máy và mobile robot di chuyển.

- Chương 8: Tổng kết và hướng phát triển đề tài

Chương 8 tổng kết, đánh giá kết quả đạt được và những hạn chế còn tồn tại của luận văn, bên cạnh đó, nêu ra hướng phát triển của đề tài này.

Chương 2: Phương án thiết kế

2.1 Giới thiệu:

Ở chương này, ta đi vào phân tích lựa chọn phương án thiết kế dựa trên cơ sở mục tiêu, nhiệm vụ và giới hạn luận văn đã trình bày ở chương trước. Mục đích nhằm xác định những thông số, nền tảng cơ bản ban đầu của thiết kế, bao gồm lựa chọn thư viện ứng dụng để xử lý thị giác máy tính, lựa chọn phương án di chuyển của robot, hình dáng robot, tay máy và số bậc tự do của tay máy...

2.2 Lựa chọn thư viện thị giác máy tính:

Vì sao lại sử dụng thư viện thị giác máy tính? Thị trường thị giác máy đang ngày càng mở rộng và liên tục phát triển. Việc sử dụng thư viện tiêu chuẩn sẽ giúp ta có thể tạo ra các ứng dụng mới và các giải pháp xử lý dễ dàng hơn, giảm thời gian cho người lập trình. Nhờ đó việc phát triển, tạo ra các mẫu ứng dụng bằng các thư viện- được cập nhật và phát triển thường xuyên- trở nên dễ dàng và thuận tiện hơn. Lĩnh vực thị giác là một lĩnh vực rộng và phức tạp, nhờ có việc chia sẻ mã nguồn mở và cung cấp thư viện như thế này đã góp phần rất lớn đẩy mạnh sự phát triển của thị giác máy tính trong những năm gần đây. Nhờ nó mà việc thực hiện được các thuật toán rất phức tạp như dò tìm khuôn mặt, nhận diện vật thể... trở nên đơn giản hơn bao giờ hết.

Có rất nhiều thư viện thị giác máy mã nguồn mở, mỗi thư viện có một nét riêng của nó mà chúng ta có thể khai thác sử dụng để áp dụng cho ứng dụng thị giác của mình. Hiện nay trên thế giới có rất nhiều thư viện mã nguồn mở cho chúng ta sử dụng, cụ thể là:

- Intel OpenCV (<http://sourceforge.net/projects/opencvlibrary>).
- CMVision (<http://www.cs.cmu.edu/~coral/minnow/platform.html>).
- ImLib3D.
- Computer Vision System Toolbox-Matlab

Có rất nhiều thư viện thị giác máy mã nguồn mở như đã giới thiệu ở trên, mỗi thư viện có một nét riêng của nó mà chúng ta có thể khai thác. Ở đây chúng ta nghiên

cứu, sử dụng thư viện OpenCV của Intel vì tính phổ biến và những khả năng của nó có thể phục vụ cho thực hiện xử lý ảnh, video.

2.3 Lựa chọn hình dáng, kích thước giới hạn của robot:

Mobile robot, tùy theo yêu cầu ứng dụng và môi trường làm việc cụ thể mà được thiết kế kích thước, hình dáng cho phù hợp. Vì sử dụng thị giác máy tính có chức năng như 1 cảm biến nhận dạng, đo khoảng cách đến vật thể, mobile robot cần được gắn camera và chờ theo máy tính hoặc truyền tín hiệu ảnh từ camera về máy tính từ xa để xử lý. Ở đây, phương án chờ theo máy tính để trực tiếp xử lý thông tin nhận về từ camera được lựa chọn vì phương án này đơn giản, hiệu quả, không cần phải tập trung vào việc giải quyết truyền nhận ảnh phức tạp như phương án thứ hai. Bên cạnh đó, do môi trường làm việc không quá phức tạp, hình dáng robot có thể được thiết kế đơn giản.

Như vậy, robot chờ theo máy tính-laptop và được gắn camera để thực hiện việc thu nhập, xử lý ảnh và điều khiển robot hoạt động, kích thước robot có thể được giới hạn trong 400x400x400 mm (kể cả kích thước tay máy). Hình dáng robot có thể tròn, chữ nhật, tam giác... nhưng ở đây, để thuận tiện cho việc chế tạo và chờ laptop, robot được thiết kế theo hình dáng chữ nhật.

2.4 Lựa chọn phương án di chuyển của robot:

2.4.1 Robot sử dụng bánh xe:

Robot dạng bánh xe là robot di chuyển bằng các bánh xe, chúng là loại được dùng phổ biến, khá đa dạng về kích thước bánh xe, số lượng bánh xe, loại bánh xe và cách bố trí bánh xe. Trong đó, robot hai, ba bánh, bốn bánh chiếm ưu thế hơn cả vì chúng có kết cấu cơ khí khá đơn giản, khả năng cân bằng tốt, độ trượt khi chuyển động ít và việc điều khiển không quá khó khăn. Nói chung, robot có cấu trúc dạng bánh có ưu điểm là đơn giản trong thiết kế và thực thi, chi phí chế tạo thấp và lựa chọn phong phú nó phù hợp với nhiều ứng dụng khác nhau.

*Robot dùng 2 bánh:

- Ưu điểm: Sử dụng ít động cơ, dễ thiết kế, chế tạo.
- Nhược điểm: Phải giải quyết bài toán cân bằng khi di chuyển.

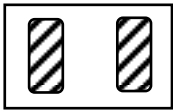

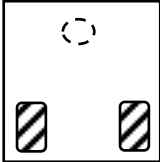
Để cân bằng, có thể sử dụng thêm các loại cảm biến như gyro, gia tốc... nhưng điều này làm tăng giá thành sản phẩm, kết cấu và điều khiển phức tạp.

Một cách khác để cân bằng là sử dụng bánh dẫn hướng phương án này đơn giản, dễ chế tạo mà vẫn đảm bảo di chuyển linh hoạt, dễ dàng đổi hướng nhờ các bánh dẫn động. Nhược điểm của phương án này là di chuyển khó khăn trong các địa hình phức tạp, gồ ghề. Tuy nhiên, rõ ràng môi trường khảo sát đặt ra ở đầu bài là địa hình bằng phẳng, không quá phức tạp, do đó nhược điểm này có thể bỏ qua.

- Đối với bánh chủ động, tùy theo môi trường làm việc mà đường kính bánh xe khác nhau. Thông thường, ở địa hình gồ ghề, phức tạp, người ta sử dụng bánh xe có đường kính lớn vì nó giúp robot dễ di chuyển qua những vùng nhấp nhô hơn và không bị mắc kẹt giữa các rãnh, hố. Đối với địa hình bằng phẳng, bánh nhỏ thường được sử dụng nhiều vì khối lượng nhẹ và không cần tải bằng các động cơ công suất lớn. Ngoài ra, trong môi trường làm việc có bề mặt trơn nhẵn, nên chọn bánh xe có diện tích tiếp xúc bề mặt lớn và chất liệu tạo ma sát tốt để tránh hiện tượng trượt.



Hình 2.1: Các loại bánh dẫn hướng thường được sử dụng

	<p>Cơ cấu với hai bánh hai bên, sử dụng bánh xe cố định và có gắn bộ truyền động</p>
	<p>Giống với dạng thứ nhất nhưng có thêm hai bánh xe caster để cân bằng</p>
	<p>Giống với dạng thứ nhất nhưng thêm một bánh xe caster để cân bằng</p>

Bảng 2.1: Một số dạng robot 2 bánh

*Robot dùng 3 bánh trở lên:

-Ưu điểm: Nhờ có sử dụng nhiều bánh, việc di chuyển được thực hiện linh hoạt hơn, dễ xoay sở, đổi hướng khi di chuyển

-Nhược điểm: Tuy nhiên, nhược điểm của cách thiết kế này là phải sử dụng và điều khiển nhiều động cơ, cần cung cấp nhiều năng lượng, chi phí thiết kế, chế tạo cao hơn.

Để phù hợp hơn khi di chuyển trên các bề mặt không bằng phẳng, độ mập mô cao, hay kết cấu bề mặt không chặt, tăng tải trọng có thể mang thì nên lựa chọn các dạng robot nhiều bánh hơn. Vì có nhiều bánh nên diện tích tiếp xúc với mặt đất lớn, trọng lực phân bố đều vì vậy nó có độ ổn định cao, hiệu suất truyền động cao hơn. Tuy nhiên, khả năng di chuyển linh hoạt là kém hơn các cấu trúc khác.

2.4.2 Robot dùng chân:

Robot chân sử dụng chân để di chuyển bằng cách mô phỏng chuyển động của con người hay một số loài vật, loại này thường có hai, bốn, sáu hay tám chân. Mỗi chân sử dụng ít nhất hai bậc tự do. Chúng có ưu điểm là gần gũi với tự nhiên và khả năng di chuyển linh hoạt trên nhiều loại địa hình như trên các bề mặt gồ ghề, trong

hang... Tuy nhiên, loại robot này rất phức tạp về kết cấu cơ khí cũng như điện tử, việc lập trình điều khiển phức tạp và có chi phí cao.

Mobile robot sử dụng chân có nhiều loại, tùy theo đối tượng mô phỏng mà chúng có thể có hai, bốn, hay sáu chân...

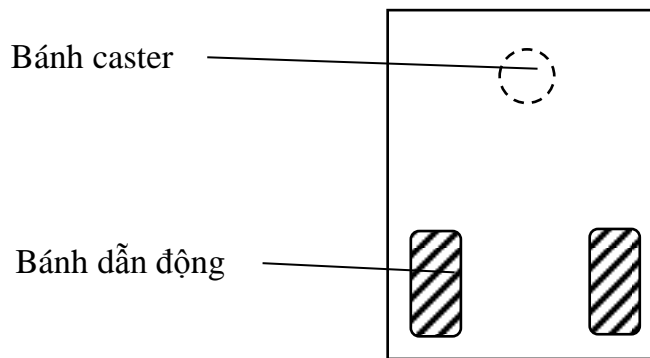


Hình 2.2: Một số dạng mobile robot dùng chân

2.4.3 Lựa chọn phương án di chuyển:

Qua quá trình phân tích, tìm hiểu trên, đã cung cấp sơ lược phần nào các loại mobile robot với các phương án di chuyển rất đa dạng và phong phú. Bên cạnh đó, mỗi phương án đều có các ưu, nhược điểm khác nhau, mà tùy theo ứng dụng, yêu cầu của robot mà ta chọn phương án di chuyển cho phù hợp. Đề tài thực hiện yêu cầu robot di chuyển trên bề mặt bằng phẳng, không quá phức tạp trong điều khiển, chế tạo. Đáp ứng được yêu cầu đó, mobile robot sử dụng 2 bánh dẫn động kết hợp với bánh dẫn hướng caster lại đơn giản, dễ chế tạo, các phụ tùng, bộ phận của robot có thể dễ dàng tìm kiếm.

Như vậy mobile robot sử dụng 2 bánh truyền động kết hợp với 1 bánh caster để giữ thăng bằng. Dựa vào hình dáng, và kích thước ban đầu, ta có bố trí các bánh xe như sau:



Hình 2.3: Robot 2 bánh kết hợp với 1 bánh caster để giữ thăng bằng

2.5 Lựa chọn phương án thiết kế tay máy:

Tay máy robot là kết cấu cơ khí được lập trình với những chức năng tương tự một cánh tay người, tay máy có thể là một robot hoàn chỉnh hoặc có thể là một bộ phận của một robot phức tạp hơn. Nó bao gồm các khâu liên kết với nhau bằng các khớp, cho phép thực hiện chuyển động quay (Rotation-R) hoặc tịnh tiến (Translation-T) để có thể tạo nên những chuyển động cơ bản của robot.

Nguồn động lực của tay máy là các động cơ điện (DC, RC, Step...) các hệ thống xy lanh khí nén, thủy lực... để tạo động lực cho tay máy hoạt động.

Tay máy rất phong phú đa dạng, có thể phân loại theo nhiều cách khác nhau:

- Theo kết cấu: robot theo kiểu tọa độ Đề các, kiểu tọa độ trụ, kiểu tọa độ cầu, kiểu SCARA...
- Theo hệ thống truyền động: Có các dạng hệ truyền động phổ biến là điện, thủy lực, khí nén.
- Theo ứng dụng: Dựa vào ứng dụng của robot có robot sơn, robot hàn, robot lắp ráp, robot chuyên phôi...
- Theo cách thức và đặc trưng của phương pháp điều khiển: Robot có thể điều khiển hở, hoặc điều khiển kín- sử dụng cảm biến, mạch phản hồi để tăng độ chính xác và mức độ linh hoạt khi điều khiển.

Như vậy, qua những giới thiệu cơ bản trên về tay máy, có những yếu tố cần phân tích lựa chọn phương án trước khi thiết kế một tay máy bao gồm: số bậc tự do của tay máy, hệ truyền động, loại khớp nối, khâu cuối...

2.5.1 Số bậc tự do của tay máy:

Muốn xác định một vật thể trong không gian, ta cần phải biết được phương hướng, vị trí tọa độ của vật thể mà ta cần tiếp cận. Bậc tự do là số thông số độc lập cần thiết để xác định hoàn toàn vị trí của cơ cấu, nó cũng là số khả năng chuyển động độc lập đối với một hệ quy chiếu của cơ cấu đó trong không gian. Công thức tính bậc tự do của một cơ cấu trong không gian là:

$$W = 6n - \left(\sum_{k=1}^5 kp_k - R_0 \right)$$

Trong đó: n - số khâu động; p_k - số khớp loại k ($k = 1, 2, \dots, 5$: số bậc tự do bị hạn chế); R_0 - số ràng buộc trùng

Để định vị và định hướng khâu chấp hành cuối một cách tùy ý trong không gian 3 chiều, robot cần có 6 bậc tự do, trong đó 3 bậc tự do để định vị và 3 bậc tự do để định hướng. Đối với các ứng dụng đơn giản như nâng hạ, sắp xếp... có thể yêu cầu số bậc tự do ít hơn.

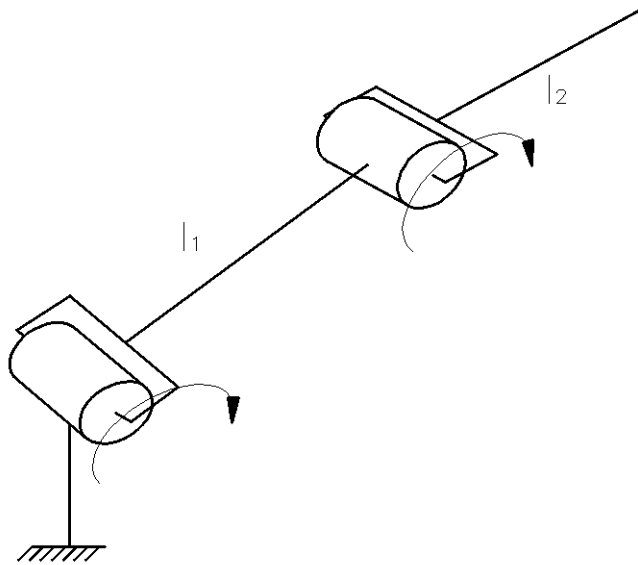
Đối với mobile robot cần thiết kế, tác vụ gấp vật tay máy gấp vật chỉ thực hiện việc gấp, nâng khối vật đơn giản. Tay máy được đặt trên mobile robot, nhờ đó có thể di chuyển xoay xở trong mặt phẳng chuyển động của robot- điều này tương đương với tay máy cố định có 3 bậc tự do. Do đó, tay máy được thiết kế đơn giản, chỉ gồm 2 bậc tự do. Như vậy, toàn bộ mobile robot có 5 bậc tự do. Với số bậc tự do này robot hoàn toàn có thể thực hiện được các tác vụ gấp khối vật trên mặt phẳng như yêu cầu.

2.5.2 Cấu trúc động học:

Thông thường, khớp động trên robot là khớp động học loại 5: khớp quay, khớp tịnh tiến. Cơ hệ của tay máy là cơ hệ hở, không có ràng buộc trùng. Do đó, theo công thức tính bậc tự do đã nêu ở trên, ta cần 2 khâu (2 khâu động) với 2 khớp loại 5. Tùy thuộc vào không gian làm việc, yêu cầu robot mà ta sẽ lựa chọn các khâu, các khớp cho hợp lý.

Robot 2 khâu, 2 khớp loại 5 có nhiều cách bố trí và lựa chọn các khâu, khớp như sau: 2 khớp tịnh tiến; 1 khớp tịnh tiến và 1 khớp xoay; 2 khớp xoay.

Mỗi phương án bố trí đều có những ưu, nhược điểm riêng, tuy nhiên với các bố trí sử dụng hai khớp xoay giúp tay máy có tầm vươn khá lớn, hơn nữa khớp xoay dễ điều khiển vì để tạo chuyển động tròn có thể sử dụng các loại động cơ điện- có giá thành rẻ, phổ biến và có thể dễ dàng áp dụng các thuật toán điều khiển phức tạp.



Hình 2.4: Sơ đồ nguyên lý tay máy

2.5.3 Khâu cuối:

Như đã trình bày ở trên, tùy theo yêu cầu, ứng dụng cụ thể của tay máy mà ta lựa chọn khâu cuối sao cho phù hợp. Để gấp được khối vật như đề bài đã nêu, có nhiều phương án để thiết kế:

-Phương án dùng giác hút chân không:

Phương án này có ưu điểm là có thể hút được mọi vật liệu, có hình dáng bất kỳ. Tuy nhiên, cần phải có thiết bị tạo lực hút, điều khiển phức tạp và tốn kém

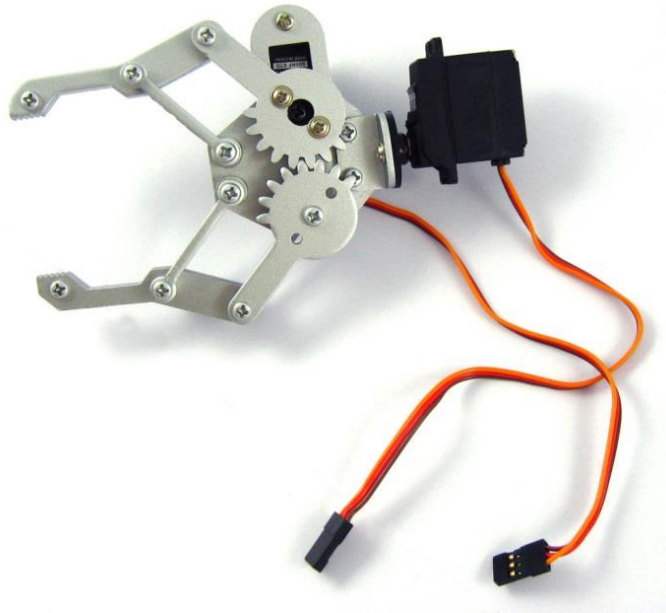
-Phương án dùng nam châm điện:

Phương án này có ưu điểm là có thể gấp được vật có bất kỳ hình dáng nào có khối lượng phù hợp. Nhưng nhược điểm của phương án này là có chi phí cao, vật hút được phải có từ tính.

-Phương án dùng cơ cấu kẹp cơ khí:

Phương án này có nhiều cách thiết kế để đạt được mục đích theo yêu cầu, có thể dễ dàng thiết kế, chế tạo với chi phí thấp.

Một trong các cơ cấu cơ khí được sử dụng để gắp vật có thể thỏa mãn yêu cầu là:



Hình 2.5: Cơ cấu khâu cuối tay máy

2.5.4 Phương án dẫn động của tay máy:

Như đã trình bày ở phần trước, có thể dẫn động tay máy bằng thủy lực, khí nén, động cơ điện. Tuy nhiên, vì khớp của tay máy đã chọn là khớp xoay, hơn nữa, việc sử dụng động cơ điện có nhiều ưu điểm như cơ cấu tác động nhanh, chính xác; có khả năng áp dụng các kỹ thuật điều khiển phức tạp; giá thành không cao; phổ biến...

Tiếp theo, ta sẽ phân tích lựa chọn loại động cơ cụ thể để sử dụng cho tay máy:

Vì là mobile robot, nguồn cung cấp cho tay máy thông thường là pin của robot, do đó các loại động cơ phổ biến có thể được sử dụng cho ứng dụng này là: DC servo, RC servo, Step

- Động cơ Step:

Trong điều khiển chuyển động kỹ thuật số, động cơ bước là một cơ cấu chấp hành đặc biệt hữu hiệu bởi nó có thể thực hiện các lệnh đưa ra dưới dạng số. Chúng được sử dụng ngày càng rộng rãi trong các hệ thống tự động, điều khiển và nhiều thiết bị điện tử khác.

Ưu điểm của động cơ này là có thể điều khiển chính xác góc quay, giá thành thấp. Tuy nhiên, nhược điểm lớn của loại động cơ này là hiện tượng trượt bước khi

quá tải gây sai lệch trong việc điều khiển. Do đó, động cơ bước thường được sử dụng trong các hệ thống điều khiển vòng hở đơn giản, tải trọng tĩnh...

-Động cơ DC servo:

DC servo là loại động được thiết kế cho những hệ thống hồi tiếp vòng kín. Ưu điểm của loại động cơ này là không có hiện tượng trượt bước như ở động cơ bước, tốc độ làm việc cao. Tuy nhiên, để điều khiển loại động cơ này cần mạch điều khiển và giải thuật điều khiển phức tạp.

-Động cơ RC servo:

Bên trong động cơ RC servo gồm một động cơ, bộ giảm tốc, một mạch điều khiển và một vôn kế. Động cơ và vôn kế nối với mạch điều khiển tạo thành mạch hồi tiếp vòng kín. Để quay động cơ, tín hiệu số được gửi tới mạch điều khiển. Tín hiệu này khởi động động cơ, thông qua chuỗi bánh răng, nối với vôn kế. Vị trí của trục vôn kế cho biết vị trí trục ra của servo. Khi vôn kế đạt được vị trí mong muốn, mạch điều khiển sẽ tắt động cơ.

RC servo được thiết kế để quay có giới hạn chứ không phải quay liên tục như động cơ DC hay động cơ bước, công dụng chính của động cơ servo là đạt được góc quay chính xác trong khoảng từ 90° – 180° . Hiện nay, góc quay này đã được mở rộng, có thể thực hiện quay với góc lớn hơn hoặc đủ một vòng.

Vì việc điều khiển động cơ rất đơn giản, RC servo thường được có thể ứng dụng để lái robot, di chuyển các tay máy lên xuống hoặc quay một cảm biến để quét khắp phòng... Tuy nhiên, động cơ có công suất nhỏ, thường được ứng dụng trong các loại mô hình, robot...

2.6 Kết luận:

Đến đây, từ đầu bài đã đặt ra, qua quá trình phân tích, ta lựa chọn phương án thiết kế như sau:

Thông số	Phương án thiết kế
Thư viện thị giác máy tính	OpenCV
Hình dáng robot	Hình chữ nhật
Kích thước	400x400x400 mm
Phương án di chuyển	2 bánh dẫn động kết hợp với bánh caster
Phương án thiết kế tay máy	Tay máy 2 bậc tự do, gồm các khớp xoay Dẫn động bằng động cơ RC Khâu cuối có kết cấu cơ khí

Bảng 2.2: Phương án thiết kế mobile robot

Chương 3: Mô hình camera, đo khoảng cách sử dụng CV

3.1 Giới thiệu:

Một trong những tính năng quan trọng của robot là khả năng sử dụng CV để xác định vật và tính toán vị trí của vật trong không gian để từ đó thực hiện tính toán, điều khiển robot di chuyển, gấp vật. Ở đây, ta sử dụng thư viện OpenCV để hỗ trợ giải quyết các bài toán này.

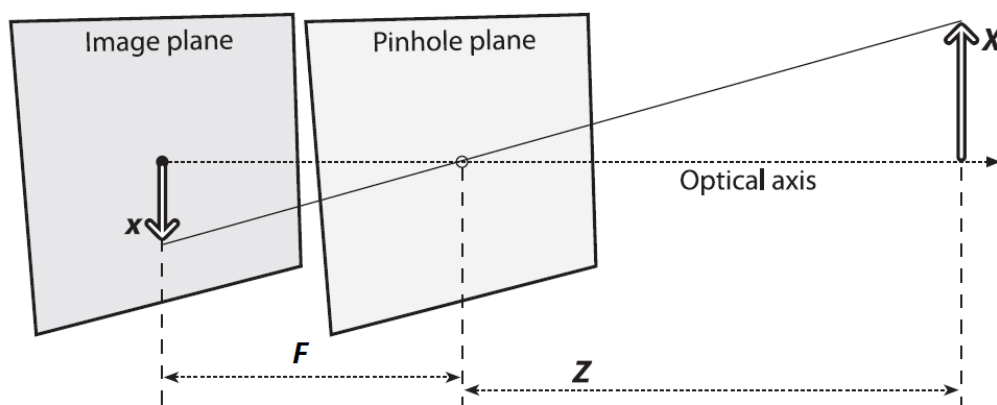
Bên cạnh đó, phương pháp đo khoảng cách sử dụng camera cần rất nhiều kiến thức nền tảng ban đầu, do đó, trước hết ta sẽ đi tìm hiểu về mô hình camera, các thông số của camera cũng như các nội dung liên quan đến quá trình calibration— một trong những bước đầu tiên, quan trọng trong quá trình đo khoảng cách. Sau đó, cơ sở lý thuyết, giải thuật đo khoảng cách, các bước tiến hành và các thông số cần xử lý, tính toán trong quá trình đo khoảng cách cũng sẽ được trình bày.

Sau khi đã biết được khoảng cách từ hệ camera đến vật thể, sử dụng các kiến thức về mô hình camera, các lý thuyết về đo khoảng cách đã xây dựng ở trên, ta có thể tính toán được các thông tin khác về vị trí của vật thể. Việc này giúp cung cấp thêm thông tin về vật thể để điều khiển robot thực hiện di chuyển và gấp vật.

3.2 Mô hình camera, và các thông số của quá trình calib camera:

3.2.1 Mô hình camera:

Mô hình pinhole camera là mô hình đơn giản nhất. Về mặt hình học, mô hình này bao gồm một mặt phẳng ảnh I (Image plane) và một tâm điểm (Eyepoint) nằm trên mặt phẳng tiêu cự (Focal plane-Pinhole plane), mô hình như sau:



Hình 3.1: Mô hình camera Pinhole [1]

Tuy nhiên, với mô hình như trên hình 3.1, ảnh của vật bị đảo ngược, do đó, ta chuyển sang một mô hình tương đương nhưng sẽ giúp dễ dàng hơn trong quá trình tính toán. Trên hình 3.2, đổi vị trí hai mặt phẳng Image plane và Pinhole plane.

Một đặc điểm rất cơ bản của mô hình trên hình 3.2 là tất cả những điểm ảnh q trên mặt phẳng ảnh I được tạo thành bởi điểm vật Q qua O sẽ thẳng hàng với O và Q , có nghĩa là ba điểm q , O , Q nằm trên một đường thẳng. Điểm O còn được gọi là tâm chiếu (Center of projection), đường thẳng qua O vuông góc với I và mặt phẳng tiêu cự được gọi là trục quang (Optical axis), trục quang cắt mặt phẳng ảnh I tại c – được gọi là Principal point.

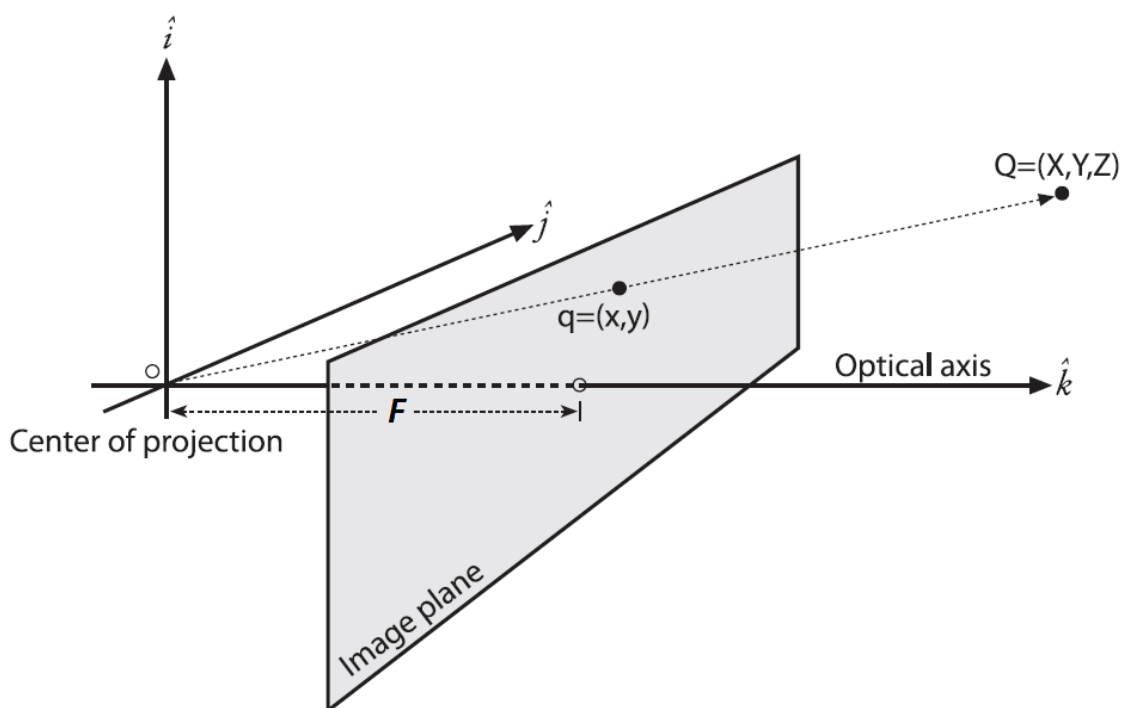
3.2.2 Thông số camera:

Thông số camera bao gồm các thông số nội (intrinsics) và các thông số ngoại (extrinsics):

a. Thông số nội

Gọi $O(X, Y, Z)$ là hệ tọa độ camera, $C(x, y)$ là hệ tọa độ của các điểm ảnh trên mặt phẳng ảnh. Ta có thể nhận thấy:

$$\frac{x}{X} = \frac{y}{Y} = \frac{F}{Z}$$



Hình 3.2: Hệ tọa độ camera [1]

Trong thực tế, do principal point c không bao giờ trùng với tâm ảnh mà có vị trí so với tọa độ gốc quy ước là $[c_x \ c_y]$. Do đó, một điểm $\mathbf{Q} = [X \ Y \ Z]^T$ ở tọa độ thực có tọa độ pixel khi được chiếu lên mặt phẳng ảnh là $[x \ y]^T$

$$x = f_x \left(\frac{X}{Z} \right) + c_x$$

$$y = f_y \left(\frac{Y}{Z} \right) + c_y$$

Ở đây, ta có hai giá trị khác nhau của tiêu cự theo hai phương. Nguyên nhân là với một camera thông thường, imager luôn là hình chữ nhật chứ không phải hình vuông tuyệt đối. Với s_x, s_y là số pixel trên một đơn vị dài milimet; F là chiều dài tiêu cự thực tế theo milimet, thì $f_x = F s_x$, $f_y = F s_y$.

Các thông số f_x , c_x , f_y , c_y còn được gọi là các thông số nội. Chúng thường được biểu diễn trong ma trận thông số nội \mathbf{M}

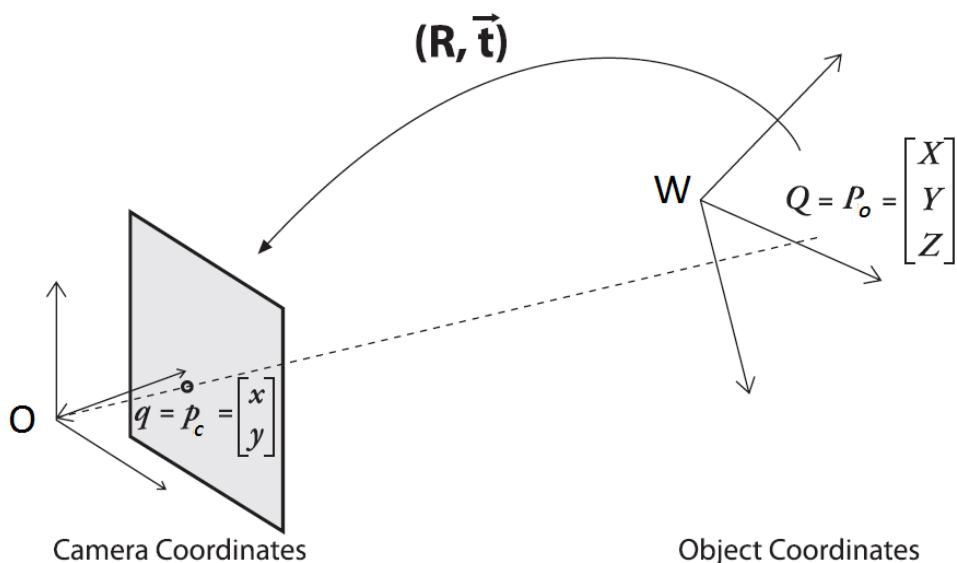
Khi đó, ta có mối liên quan giữa tọa độ điểm ảnh với tọa độ điểm thực như sau:

$$\mathbf{q} = \mathbf{M}\mathbf{Q}$$

$$\text{Với } \mathbf{q} = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, \mathbf{M} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{Q} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

b. Thông số ngoại:

Một vật thể trong không gian thường được biểu diễn bởi một hệ tọa độ bất kỳ - không phải hệ tọa độ camera. Do đó, để tính toán được trên các hệ tọa độ này, cách đơn giản nhất là ta chuyển nó về hệ tọa độ camera. Giả sử rằng một điểm bất kỳ có tọa độ \mathbf{P}_W trong hệ tọa độ W (Object/World Coordinates) và có tọa độ \mathbf{P}_O trong hệ tọa độ camera. Khi đó, \mathbf{P}_W có thể chuyển về \mathbf{P}_O bằng một phép quay và một phép tịnh tiến.



Hình 3.3: Ma trận R , vector T [1]

$$P_o = RP_w + T$$

Ma trận R và vector T mô tả hướng, vị trí tương đối giữa hệ tọa độ thực world và hệ tọa độ camera. Ma trận R chứa 3 phép quay và vector T chứa 1 phép tịnh tiến theo vector T . Những thông số chứa trong R và T được gọi là những thông số ngoại của camera.

3.2.3 Lens Distortions:

Trong thực tế, không có thấu kính nào là tuyệt đối hoàn hảo. Điều này là do hạn chế không thể tránh khỏi trong quá trình sản xuất các thấu kính. Và ngoài ra, cũng rất khó để lắp đặt các thấu kính thẳng hàng một cách chính xác tuyệt đối. Do đó, ở đây, chúng ta mô tả hai distortions chính đó là Radial distortion (gây ra do hình dáng của lens kính) và Tangential Distortion (do quá trình lắp đặt hệ thống bên trong camera)

Chúng ta bắt đầu với Radial distortion. Các thấu kính của camera thực tế thường gây ra các biến dạng ở rìa của imager. Điều này thường dẫn đến hệ quả gây ra hiện tượng “barrel” hoặc “fish-eye”. Hình 3.4 cho ta gợi ý tại sao Radial Distortion xảy ra. Những tia sáng càng xa trung tâm thấu kính càng bị bẻ cong hơn so với những tia đi qua gần tâm.

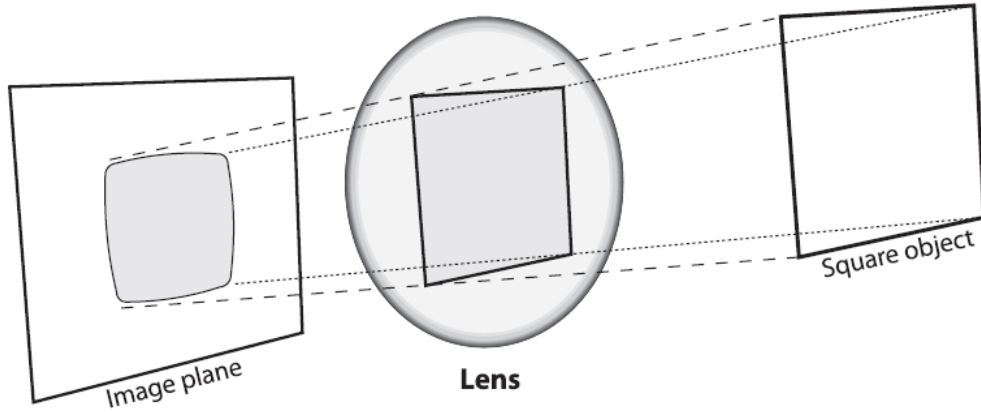
Đối với radial distortion, distortion bằng 0 ở tâm của ảnh (optical) và tăng dần khi ra dần phía ngoài. Trong thực tế, distortion này nhỏ và có thể được mô tả nhờ khai

triển Taylor quanh $r = 0$. Thông thường, chúng ta sử dụng 2 hệ số đầu tiên k_1, k_2 . Với một số loại camera có biến dạng lớn, ta có thể sử dụng đến hệ số thứ 3 k_3 .

$$x_{corrected} = x(1 + k_1r^2 + k_2r^4 + k_3r^6)$$

$$y_{corrected} = y(1 + k_1r^2 + k_2r^4 + k_3r^6)$$

Trong đó, (x, y) là điểm trên ảnh, và $(x_{corrected}, y_{corrected})$ là tọa độ mới sau khi đã bù trừ distortion



Hình 3.4: Radial Distortion [1]

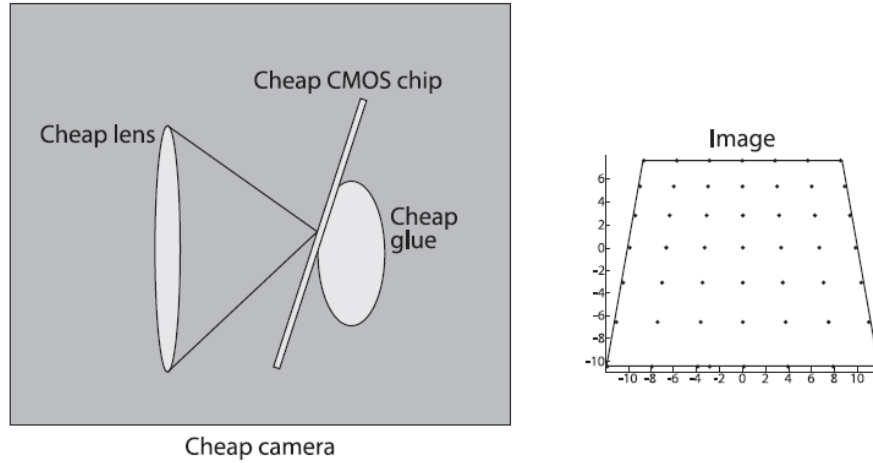
Distortion phổ biến thứ hai đó là Tangential Distortion, được gây ra do lỗi quá trình sản xuất làm cho lens không định vị song song hoàn toàn so với mặt phẳng ảnh-
Hình 3.5

Tangential Distortion được mô tả theo phương trình sau [1]:

$$x_{corrected} = x + (2p_1y + p_2(r^2 + 2x^2))$$

$$y_{corrected} = y + (2p_2x + p_1(r^2 + 2y^2))$$

Vậy Có tổng cộng 5 hệ số ta cần để mô tả lens distortions. Năm hệ số này được lưu trong 1 vector 5x1 khi tính toán $(k_1, k_2, p_1, p_2, k_3)$ hoặc $(k_1, k_2, k_3, k_4, k_5)$ (theo đúng thứ tự này)



Hình 3.5: Tangential Distortion bị gây ra do lens không hoàn toàn song song với mặt phẳng ảnh [1]

3.2.4 Calibrate camera:

Thực hiện quá trình calibrate camera thực chất là tìm các thông số của camera gồm 4 thông số nội (f_x, c_x, f_y, c_y), 3 thông số ma trận xoay, 3 thông số của vector tịnh tiến (T_x, T_y, T_z) và 5 hệ số distortion (k_1, k_2, p_1, p_2, k_3).

Quá trình calib thực chất là đi tìm các thông số này khi ta có các điểm $\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_n$ đã biết vị trí trong hệ tọa độ vật, và ảnh $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n$ của chúng trên mặt phẳng ảnh. Thông thường, để tạo các điểm này ta thường hay sử dụng chessboard (hình 3.6).

Giả sử bỏ qua các hệ số distortion, có 10 thông số cần tìm. Như vậy cần xác định ít nhất 5 điểm trên chessboard (mỗi điểm có tọa độ x, y, do đó có thể thiết lập được 2 phương trình từ 1 điểm)

Giả sử có các điểm trong tọa độ thực là $\mathbf{Q}_i = [X_i \ Y_i \ Z_i]^T$ có ảnh $\mathbf{q}_i = [x_i \ y_i]^T$ ($i = 1, 2, \dots, n$)

Đặt $\tilde{\mathbf{Q}}_i = [X_i \ Y_i \ Z_i \ 1]^T$, $\tilde{\mathbf{q}}_i = [x'_i \ y'_i \ w]^T$

Với $x_i = x'_i/w$ và $y_i = y'_i/w$

Với $\mathbf{M} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$ là ma trận thông số nội camera

\mathbf{R}, \mathbf{T} là các ma trận xoay, tịnh tiến của camera

Theo quan hệ giữa các hệ tọa độ, có biểu thức sau:

$$\tilde{\mathbf{q}}_i = \mathbf{M} [\mathbf{R} \ \mathbf{T}] \widetilde{\mathbf{Q}}_i$$

Đặt $\mathbf{H} = \mathbf{M} [\mathbf{R} \ \mathbf{T}]$

\mathbf{H} là ma trận 3×4 , \mathbf{H} có thể được viết dưới dạng các vector theo hàng như sau với $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$ là các vector 1×4

$$\mathbf{H} = [\mathbf{h}_1 \ \mathbf{h}_2 \ \mathbf{h}_3]^T$$

Suy ra

$$\tilde{\mathbf{q}}_i = \mathbf{H} \widetilde{\mathbf{Q}}_i$$

$$\begin{bmatrix} x'_i \\ y'_i \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{h}_1 \widetilde{\mathbf{Q}}_i \\ \mathbf{h}_2 \widetilde{\mathbf{Q}}_i \\ \mathbf{h}_3 \widetilde{\mathbf{Q}}_i \end{bmatrix}$$

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{h}_1 \widetilde{\mathbf{Q}}_i}{\mathbf{h}_3 \widetilde{\mathbf{Q}}_i} \\ \frac{\mathbf{h}_2 \widetilde{\mathbf{Q}}_i}{\mathbf{h}_3 \widetilde{\mathbf{Q}}_i} \end{bmatrix}$$

Suy ra:

$$\begin{cases} x_i = \frac{\mathbf{h}_1 \widetilde{\mathbf{Q}}_i}{\mathbf{h}_3 \widetilde{\mathbf{Q}}_i} \\ y_i = \frac{\mathbf{h}_2 \widetilde{\mathbf{Q}}_i}{\mathbf{h}_3 \widetilde{\mathbf{Q}}_i} \end{cases}$$

$$\begin{cases} x_i(\mathbf{h}_3 \widetilde{\mathbf{Q}}_i) - \mathbf{h}_1 \widetilde{\mathbf{Q}}_i = 0 \\ y_i(\mathbf{h}_3 \widetilde{\mathbf{Q}}_i) - \mathbf{h}_2 \widetilde{\mathbf{Q}}_i = 0 \end{cases}$$

Do đó, được hệ phương trình:

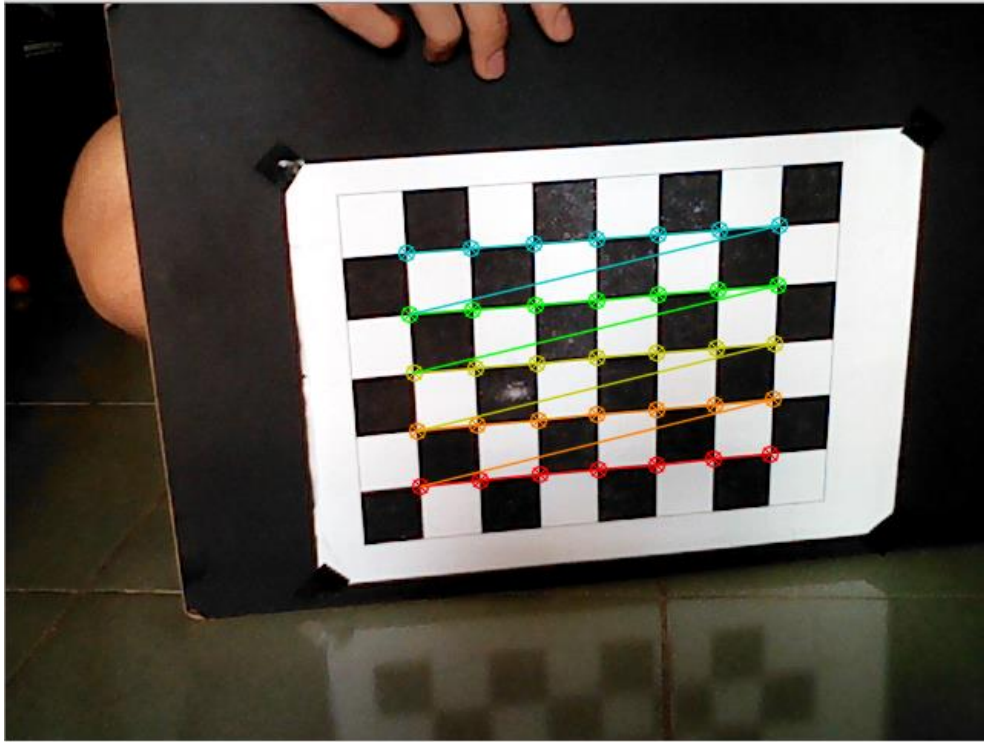
$$\begin{cases} -x_1(\mathbf{h}_3 \widetilde{\mathbf{Q}}_1) + \mathbf{h}_1 \widetilde{\mathbf{Q}}_1 = 0 \\ -y_1(\mathbf{h}_3 \widetilde{\mathbf{Q}}_1) + \mathbf{h}_2 \widetilde{\mathbf{Q}}_1 = 0 \\ \vdots \\ -x_i(\mathbf{h}_3 \widetilde{\mathbf{Q}}_i) + \mathbf{h}_1 \widetilde{\mathbf{Q}}_i = 0 \\ -y_i(\mathbf{h}_3 \widetilde{\mathbf{Q}}_i) + \mathbf{h}_2 \widetilde{\mathbf{Q}}_i = 0 \\ \vdots \\ -x_n(\mathbf{h}_3 \widetilde{\mathbf{Q}}_n) + \mathbf{h}_1 \widetilde{\mathbf{Q}}_n = 0 \\ -y_n(\mathbf{h}_3 \widetilde{\mathbf{Q}}_n) + \mathbf{h}_2 \widetilde{\mathbf{Q}}_n = 0 \end{cases}$$

$$\text{Đặt } \mathbf{P} = \begin{bmatrix} \widetilde{\mathbf{Q}}_1^T & 0^T & -x_1 \widetilde{\mathbf{Q}}_1^T \\ 0^T & \widetilde{\mathbf{Q}}_1^T & -y_1 \widetilde{\mathbf{Q}}_1^T \\ \dots & \dots & \dots \\ \widetilde{\mathbf{Q}}_n^T & 0^T & -x_n \widetilde{\mathbf{Q}}_n^T \\ 0^T & \widetilde{\mathbf{Q}}_n^T & -y_n \widetilde{\mathbf{Q}}_n^T \end{bmatrix}_{2n \times 12} \quad \mathbf{m} = \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \end{bmatrix}_{12 \times 1}$$

Hệ phương trình trở thành dạng $\mathbf{P}\mathbf{m} = 0$ (hệ đồng nhất tuyến tính). Hệ này có thể giải nhờ sử dụng phương pháp SVD (Singular Value Decomposition)[1].

Sau khi giải hệ phương trình này ta được ma trận $\mathbf{H} = \mathbf{M} [\mathbf{R} \mathbf{T}]$. Khi đó các thông số của camera hoàn toàn có thể được xác định.

Thư viện OpenCV cung cấp một công cụ để ta tiến hành calib camera đó là hàm `void cvCalibrateCamera2(CvMat* object_points, CvMat* image_points, int* point_counts, CvSize image_size, CvMat* intrinsic_matrix, CvMat* distortion_coeffs, CvMat* rotation_vectors, CvMat* translation_vectors, int flags);`



Hình 3.6: Thực hiện calibrate cho camera- Hiển thị các corner của chessboard

Sau đây là kết quả của quá trình calib từng camera của hệ camera được sử dụng:

Camera	f_x	f_y	c_x	c_y	k_1	k_2	p_1	p_2	k_3
Trái	901	903	341	273	0.039	-1.016	0.002	0.007	4.595
Phải	901	902	343	273	0.071	-1.719	0.003	0.002	5.906

Bảng 3.1: Kết quả calibrate camera trái và phải

3.3 Đo khoảng cách sử dụng CV:

Ở phần này, sẽ đi vào trình bày việc sử dụng 2 camera để đo khoảng cách. Thông thường, phương pháp được sử dụng để trích xuất thông tin 3D là từ nhiều hình ảnh, hay còn gọi là phương pháp multiple images. Trong đó, phương pháp dễ dàng và đơn giản hơn cả là sử dụng hai camera để tái tạo lại một cảnh 3D. Để xác định thông tin về khoảng cách, trước hết, các đặc điểm đặc trưng của vật thể trong một ảnh (hoặc nhiều hơn) được tìm tương ứng trong ảnh khác (các ảnh chụp vật thể cùng lúc từ các camera riêng biệt). Sau đó, sự sai khác các đặc điểm của vật thể trong hai ảnh sẽ được sử dụng để tính toán thông tin về khoảng cách. Chúng ta sử dụng disparity effect (triangulation) để tính toán khoảng cách (sẽ được trình bày kỹ hơn ở phần sau). Để có cái nhìn tổng quan hơn, trong thực tế, có 4 bước cần thực hiện khi sử dụng 2 camera để xác định khoảng cách là:

-Bước 1: Loại bỏ radial và tangential distortion nhờ tính toán; bước này còn được gọi là undistortion (đã trình bày ở phần trước). Sau bước này, kết quả cho ra là hình ảnh đã được undistorted

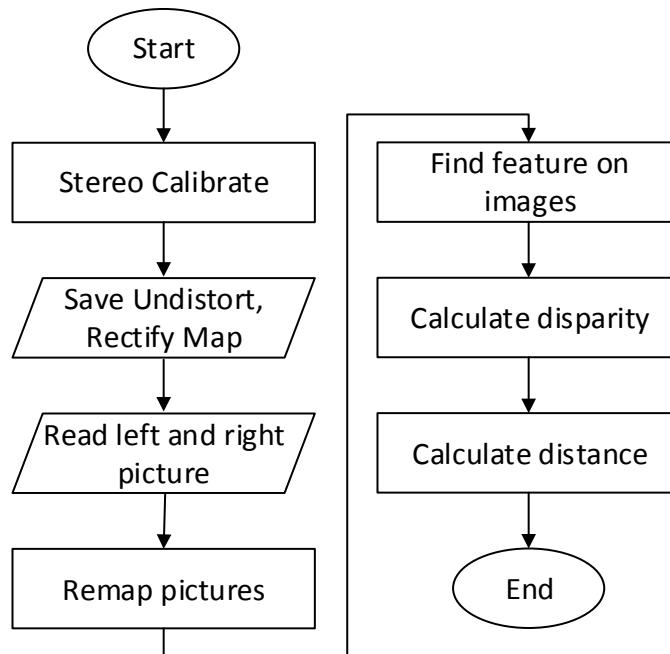
-Bước 2: Điều chỉnh góc và khoảng cách giữa các camera, còn gọi là rectification. Bước này cho kết quả là các ảnh “row- aligned” (2 mặt phẳng ảnh đồng phẳng, ảnh có cùng hướng và cùng tọa độ y)

-Bước 3: Tìm các điểm đặc trưng tương ứng ở hình trái và phải, còn gọi là bước correspondence. Kết quả của bước này là disparity của điểm đặc trưng của vật cần đo khoảng cách (khái niệm này sẽ trình bày rõ hơn ở sau)

-Bước 4: Khi đã biết được mối quan hệ hình học giữa các camera với nhau, ta có thể sử dụng kết quả disparity ở bước 3 để tính toán ra khoảng cách- sử dụng triangulation. Bước này còn được gọi là reprojection.

Như vậy, để trích xuất thông tin về khoảng cách từ các ảnh cần phải trải qua rất nhiều bước phức tạp. Thực chất, bước 1 và 2 là những bước đầu tiên để calib, chỉnh sửa hình ảnh sao cho ảnh được sử dụng để tính toán có độ chính xác cao nhất, ít bị sai lệch nhất. 2 bước sau cùng là cách làm cụ thể để trích xuất được thông tin về khoảng cách từ hình ảnh nhận được sau 2 bước đầu tiên.

Sau đây là sơ đồ khối giải thuật đo khoảng cách sử dụng CV:



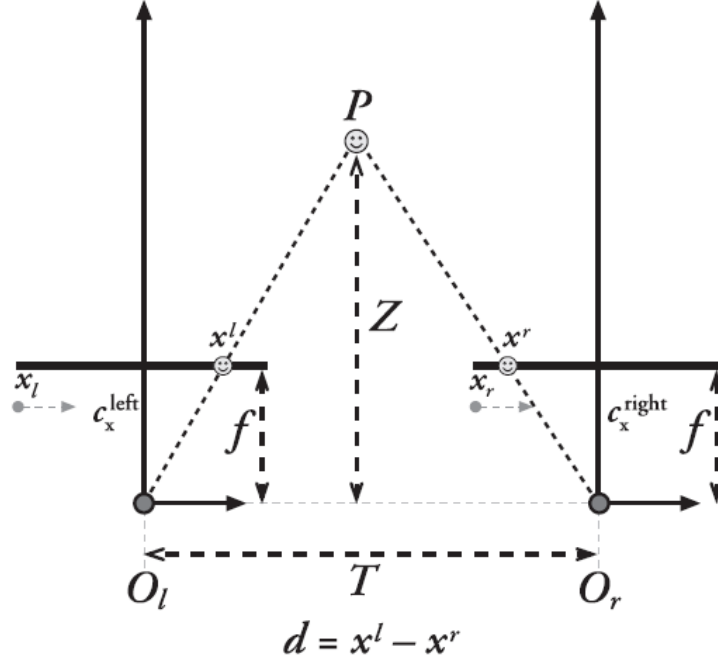
Hình 3.7: Lưu đồ giải thuật đo khoảng cách

Các phần sau đây sẽ giải thích rõ về phương pháp đo khoảng cách này và cũng như giải thích áp dụng cụ thể giải thuật trên đối với đối tượng mà đầu đề đã đặt ra.

3.3.1 Triangulation:

Để hiểu rõ về phương pháp đo khoảng cách này, trước hết ta tìm hiểu phương pháp triangulation. Giả sử rằng, sau bước 1 và 2, hình ảnh thu được đã được undistorted, aligned như hình 3.8, hai camera có 2 mặt phẳng ảnh đồng phẳng, 2 optical axis song song với nhau (optical axis là đường thẳng từ tâm phép chiếu O qua điểm principal c –đã trình bày ở phần mô hình camera), khoảng cách của 2 trục này đã

biết, chiều dài tiêu cự của 2 camera như nhau: $f_l = f_r$. Đồng thời, cũng giả sử rằng 2 điểm principal c^{left} và c^{right} có cùng tọa độ pixel tương ứng trong ảnh trái, phải.



Hình 3.8: Mô hình 2 camera đã được undistorted, rectified. O_l, O_r là tâm phép chiếu; P là điểm thực cần được đo khoảng cách Z [1]

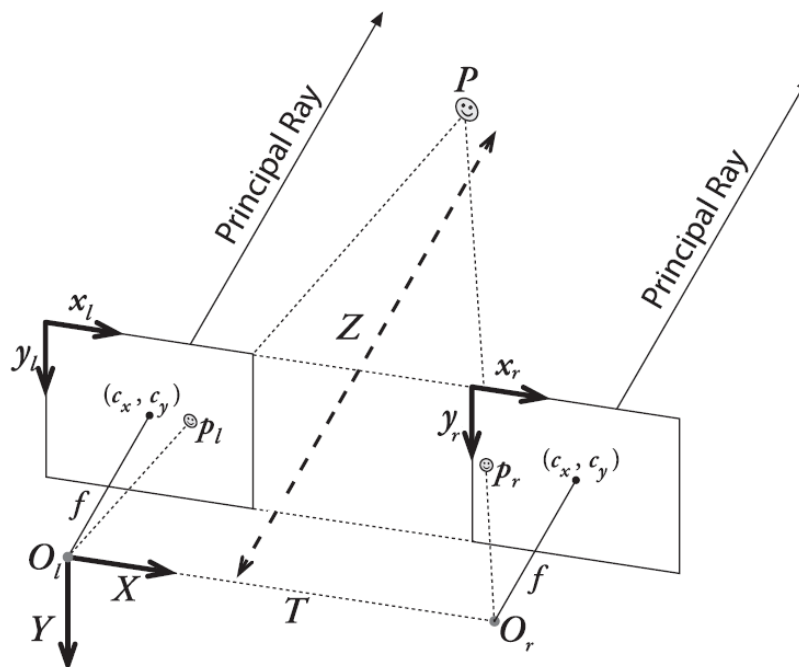
Giả sử rằng ảnh trái và ảnh phải row-aligned sao cho mỗi pixel trên một ảnh có cùng một tọa độ y với pixel tương ứng trên ảnh kia— có nghĩa là ảnh của điểm thực P trên ảnh trái và ảnh phải nằm trên cùng một hàng, cùng tọa độ y— còn gọi là frontal parallel. Đồng thời, cũng giả sử rằng điểm P nằm trong vùng phù hợp sao cho ảnh của P xuất hiện trong ảnh trái và phải là : p_l, p_r , có tọa độ x tương ứng là x^l, x^r .

Trong trường hợp đơn giản này, tọa độ x^l, x^r sẽ cho phép chúng ta tính toán khoảng cách, thông qua việc sử dụng disparity (ký hiệu là d , với $d = x^l - x^r$). Từ hình 3.8 sử dụng tam giác đồng dạng ta được:

$$\frac{T - (x^l - x^r)}{T} = \frac{Z - f}{Z}$$

$$Z = \frac{fT}{x^l - x^r} = \frac{fT}{d}$$

Hình 3.9 cho ta biết về các hệ tọa độ 2D và 3D được sử dụng trong hệ hai camera (đây cũng chính là các hệ tọa độ được OpenCV sử dụng khi tính toán trong hệ hai camera)



Hình 3.9: Hệ tọa độ được OpenCV sử dụng [1]

Trong thực tế, camera không bao giờ có thể được lắp đặt một cách hoàn hảo để đạt được frontal parallel như hình 3.8. Thay vào đó, chúng ta thường phải tính toán, tìm các phép chiếu, các distortion để có thể chỉnh sửa– rectify– các ảnh trái, phải sao cho chúng đạt được frontal parallel. Do đó, khi thiết lập hệ thống camera, yêu cầu đặt ra là phải làm cho các camera này càng đồng phẳng, càng thẳng hàng càng tốt. Điều này giúp cho việc tính toán trở nên nhanh chóng và dễ dàng hơn. Hơn nữa, để kết quả được tốt, hai camera cũng cần được đồng bộ sao cho chúng chụp hình gần như cùng lúc, đặc biệt là trong trường hợp vật di động hoặc các camera di chuyển. Nếu không được đồng bộ, phương pháp này chỉ thích hợp cho xử lý các ảnh tĩnh, cảnh tĩnh.

Ngoài ra, từ công thức tính khoảng cách Z, ta có một kết luận quan trọng như sau:

$$Z = \frac{fT}{x^l - x^r} = \frac{fT}{d}$$

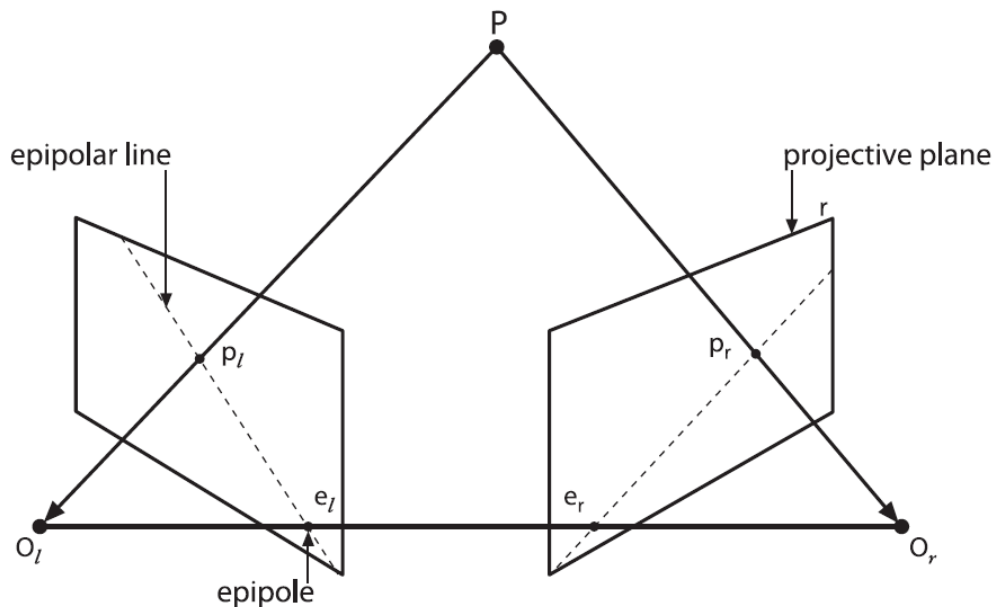
$$\Delta Z = \frac{Z^2}{fT} \Delta d$$

Công thức trên cho phép ta xác định khoảng cách T , lựa chọn camera để thu được độ chính xác của phép đo khoảng cách như mong muốn.

3.3.2 Epipolar Geometry:

Thông số hình học cơ bản của hệ thống stereo imaging quan hệ mật thiết với epipolar geometry. Đây là sự kết hợp của mô hình pinhole (cho mỗi camera) và các điểm epipole. Tìm hiểu nội dung này sẽ cho chúng ta hiểu về thông số hình học cũng như cách để giới hạn việc tìm kiếm điểm đặc trưng trong hình của hệ thống stereo 2 camera.

Trước hết, ta tìm hiểu một số ký hiệu và khái niệm liên quan:



Hình 3.10: Epipolar plane là mặt phẳng tạo bởi điểm P và hai tâm chiếu của hai camera; hai điểm epipole là các giao điểm của đường thẳng nối hai tâm chiếu O_l, O_r với hai mặt phẳng ảnh [1]

Có 2 tâm phép chiếu là O_l, O_r và hai mặt phẳng chiếu tương ứng là Π_l, Π_r . Điểm thực P có ảnh trên hai mặt phẳng chiếu là p_l, p_r . Điểm epipole e_l (hay e_r) trên mặt phẳng Π_l (hay Π_r) được định nghĩa là ảnh của tâm chiếu của camera phải O_r (hay của camera trái O_l). Mặt phẳng tạo bởi điểm thực P và hai điểm epipole e_l, e_r (hoặc hai điểm tâm phép chiếu O_l, O_r) được gọi là epipolar plane. Các đường thẳng $p_l e_l, p_r e_r$ (từ ảnh của vật đến điểm epipole) được gọi là epipolar line.

Một điểm thực P có ảnh p_r trên mặt phẳng ảnh phải (hoặc trái), P có thể nằm tại bất kì điểm nào trên đường thẳng đi qua điểm O_r và p_r (hoặc qua điểm O_l và p_l). Điều này là do với 1 camera, chúng ta không thể biết khoảng cách đến điểm mà ta quan sát. Như vậy, đối với camera phải, đường thẳng qua O_r và p_r chứa điểm P và tất nhiên cũng chứa rất nhiều điểm khác. Do đó, điều quan trọng là ta phải tìm xem đường thẳng này có ảnh như thế nào khi được chiếu trên mặt phẳng ảnh bên trái. Thực tế, ảnh của đường thẳng này là đường epipolar $p_l e_l$ (vì p_l là ảnh của P , e_l là ảnh của O_r trên mặt phẳng ảnh trái).

Chúng ta có một số kết luận như sau:

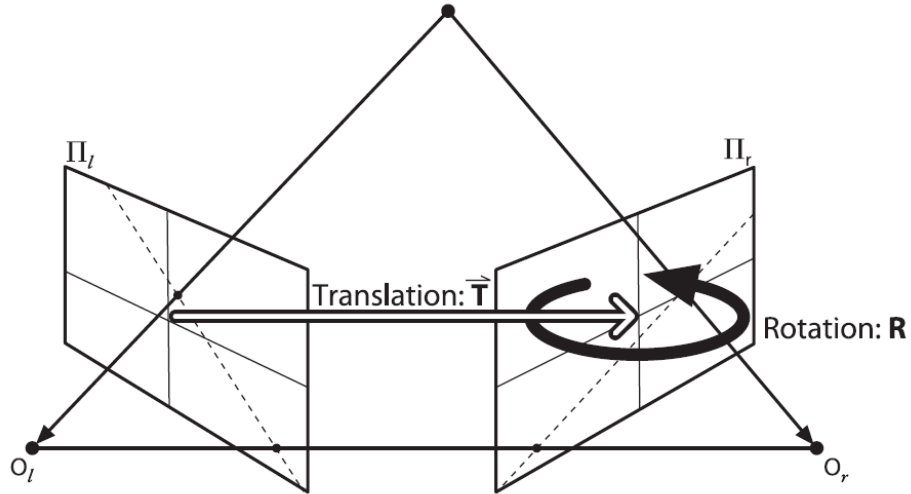
- Mỗi điểm trong thế giới thực 3D đều nằm trong một epipolar plane và mặt phẳng này cắt mỗi mặt phẳng ảnh tại epipolar line.

- Cho một điểm đặc trưng nằm trong một mặt phẳng ảnh (ví dụ là ảnh trái), ta có thể tìm ra điểm tương ứng của điểm đó trong mặt phẳng ảnh còn lại (ảnh phải) chắc chắn nằm trên epipolar line. Điều này được gọi là epipolar constraint.

- Việc có epipolar constraint giúp việc tìm kiếm điểm đặc trưng tương ứng trên các ảnh được giảm xuống từ 2 chiều (tìm trên toàn bộ ảnh) còn 1 chiều (tìm trên epipolar line). Điều này có ý nghĩa rất lớn trong phương pháp đo khoảng cách này vì không những tiết kiệm rất nhiều thời gian tính toán mà còn giúp ta loại bỏ được rất nhiều điểm có thể làm ta nhầm tưởng là điểm đặc trưng trong hình (hay còn gọi là các điểm nhiễu), việc này giúp việc tìm kiếm trở nên nhanh và chính xác hơn.

3.3.3 Stereo Calibration, Rectification:

Như đã trình bày, trước hết cần tiến hành calibration và rectification. Mục tiêu của bước này là tìm được các thông số của mỗi camera và các ma trận Rotation, Translation- thể hiện mối tương quan giữa hai camera



Hình 3.11: Mối tương quan giữa hai camera được thể hiện trong ma trận E –chứa các thông tin về ma trận Rotation và Translation [1]

Stereo calibration là quá trình tính toán mối quan hệ giữa hai camera trong không gian thực. Các mối quan hệ này được biểu diễn thông qua các ma trận Essential E và ma trận Fundamental F . Ma trận E chứa thông tin về Translation và Rotation giữa hai camera trong không gian thực. Ma trận F chứa thông tin tương tự như ma trận E nhưng có bao gồm cả các thông số nội của hai camera liên quan trong hệ tọa độ pixel. Để thấy rõ sự khác nhau giữa hai ma trận này. Ma trận E có 5 thông số, ma trận này chỉ cung cấp thông tin, mối quan hệ về vị trí, tọa độ thực của ảnh của điểm P trên ảnh trái và ảnh phải (p_l, p_r). Do đó, ma trận E không thể cung cấp cho ta biết về mối quan hệ giữa hai camera trong hệ tọa độ pixel cũng như không thể tìm các epipolar line tương ứng bằng cách sử dụng ma trận E . Vì thế, ở đây, chúng ta thêm vào các thông số nội, sử dụng ma trận F (gồm có 7 thông số) để chuyển các thông tin cần thiết từ ma trận E sang hệ tọa độ pixel.

Xét trong hệ tọa độ camera trái gốc là O_l , điểm P có tọa độ P_l . Điểm P có tọa độ trong hệ tọa độ camera phải là P_r . Với ${}^r_lR, {}^r_lT$ là các ma trận xoay, chuyển vị chuyển từ hệ tọa độ camera trái sang hệ tọa độ camera phải; ${}^l_rR, {}^l_rT$ là các ma trận xoay, chuyển vị chuyển từ hệ tọa độ camera phải sang hệ tọa độ camera trái.

Ta có mối quan hệ sau: $P_l = {}^l_rR P_r + {}^l_rT$

Suy ra: $(P_l - {}^l_rT) = {}^l_rR^{-1} P_r = {}^r_lR^T P_r$

Vì \mathbf{P}_l và ${}^l_r\mathbf{T}$ cùng nằm trong mặt phẳng epipolar, do đó

$$(\mathbf{P}_l - {}^l_r\mathbf{T})^T ({}^l_r\mathbf{T} \times \mathbf{P}_l) = 0$$

$$({}^r_l\mathbf{R}^T \mathbf{P}_r)^T ({}^l_r\mathbf{T} \times \mathbf{P}_l) = 0$$

Với ${}^l_r\mathbf{T} = [T_x \quad T_y \quad T_z]$, $\mathbf{S} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}$ (\mathbf{S} là ma trận Skew symmetric)

$${}^l_r\mathbf{T} \times \mathbf{P}_l = \mathbf{S} \mathbf{P}_l$$

Khi đó ta được $({}^r_l\mathbf{R}^T \mathbf{P}_r)^T (\mathbf{S} \mathbf{P}_l) = 0$

$$(\mathbf{P}_r)^T {}^r_l\mathbf{R} \mathbf{S} \mathbf{P}_l = 0$$

Vì $\mathbf{p}_l = F_l \mathbf{P}_l / Z_l$, $\mathbf{p}_r = F_r \mathbf{P}_r / Z_r$, chia hai vế phương trình trên cho $Z_l Z_r / (F_l F_r)$

(Với F_l, F_r lần lượt là chiều dài tiêu cự thực tế của camera trái, phải)

$$(\mathbf{p}_r)^T {}^r_l\mathbf{R} \mathbf{S} \mathbf{p}_l = 0$$

Ma trận $\mathbf{E} = {}^r_l\mathbf{R} \mathbf{S}$ là ma trận essential cần tìm

$$(\mathbf{p}_r)^T \mathbf{E} \mathbf{p}_l = 0$$

Để tìm ma trận Fundamental \mathbf{F} , ta chuyển các tọa độ thực $\mathbf{P}_l, \mathbf{P}_r$ của điểm P về tọa độ ảnh (pixel) ($\mathbf{q}_l = \mathbf{M}_l \mathbf{P}_l$, $\mathbf{q}_r = \mathbf{M}_r \mathbf{P}_r$):

$$(\mathbf{P}_r)^T \mathbf{E} \mathbf{P}_l = 0$$

$$\mathbf{q}_r^T (\mathbf{M}_r^{-1})^T \mathbf{E} \mathbf{M}_l^{-1} \mathbf{q}_l = 0$$

Với $\mathbf{F} = (\mathbf{M}_r^{-1})^T \mathbf{E} \mathbf{M}_l^{-1}$

\mathbf{F} thỏa biểu thức sau:

$$\mathbf{q}_r^T \mathbf{F} \mathbf{q}_l = 0$$

Như vậy, đến đây, ta đã biết làm thế nào để tính các ma trận \mathbf{E}, \mathbf{F} . Tiếp theo, ta tìm hiểu cách tính các ma trận ${}^r_l\mathbf{R}, {}^l_r\mathbf{T}$ của hệ camera.

Với một điểm P bất kỳ, trong hệ tọa độ camera trái (camera phải) P có tọa độ $\mathbf{P}_l = {}^l_w\mathbf{R} \mathbf{P}_w + {}^l_w\mathbf{T}$ ($\mathbf{P}_r = {}^r_w\mathbf{R} \mathbf{P}_w + {}^r_w\mathbf{T}$). Trong đó ${}^l_w\mathbf{R}, {}^l_w\mathbf{T}$ (hay ${}^r_w\mathbf{R}, {}^r_w\mathbf{T}$) là các ma trận chuyển từ hệ tọa độ thực (object/world) sang hệ tọa độ camera trái (hay phải).

Lại có

$$P_r = {}^r_l R P_l + {}^r_l T$$

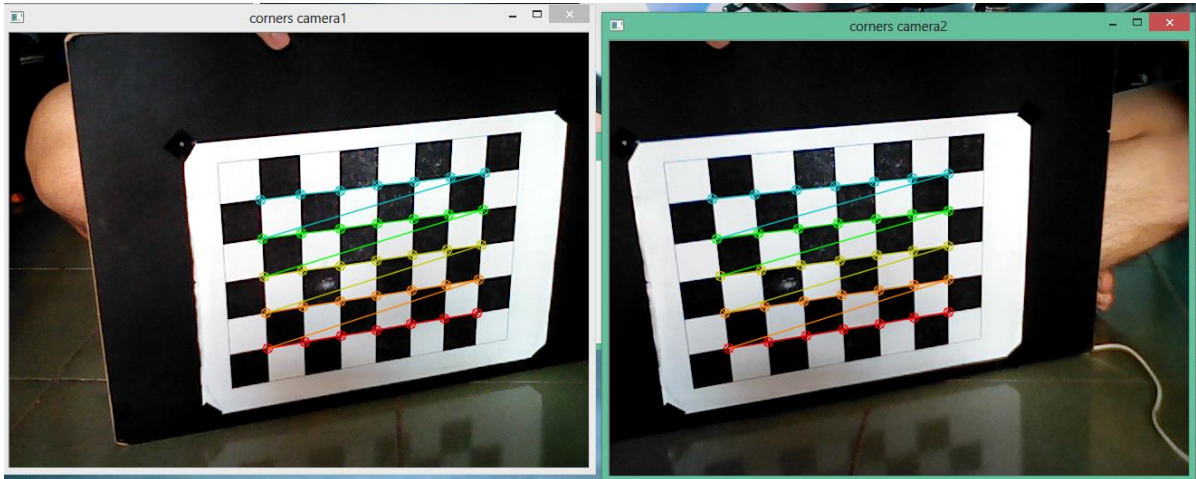
$$P_l = {}^r_l R^T (P_r - {}^r_l T)$$

Thay $P_l = {}^l_w R P_w + {}^l_w T$ và $P_r = {}^r_w R P_w + {}^r_w T$ vào phương trình trên, ta tìm được ${}^r_l R$ và ${}^r_l T$ như sau:

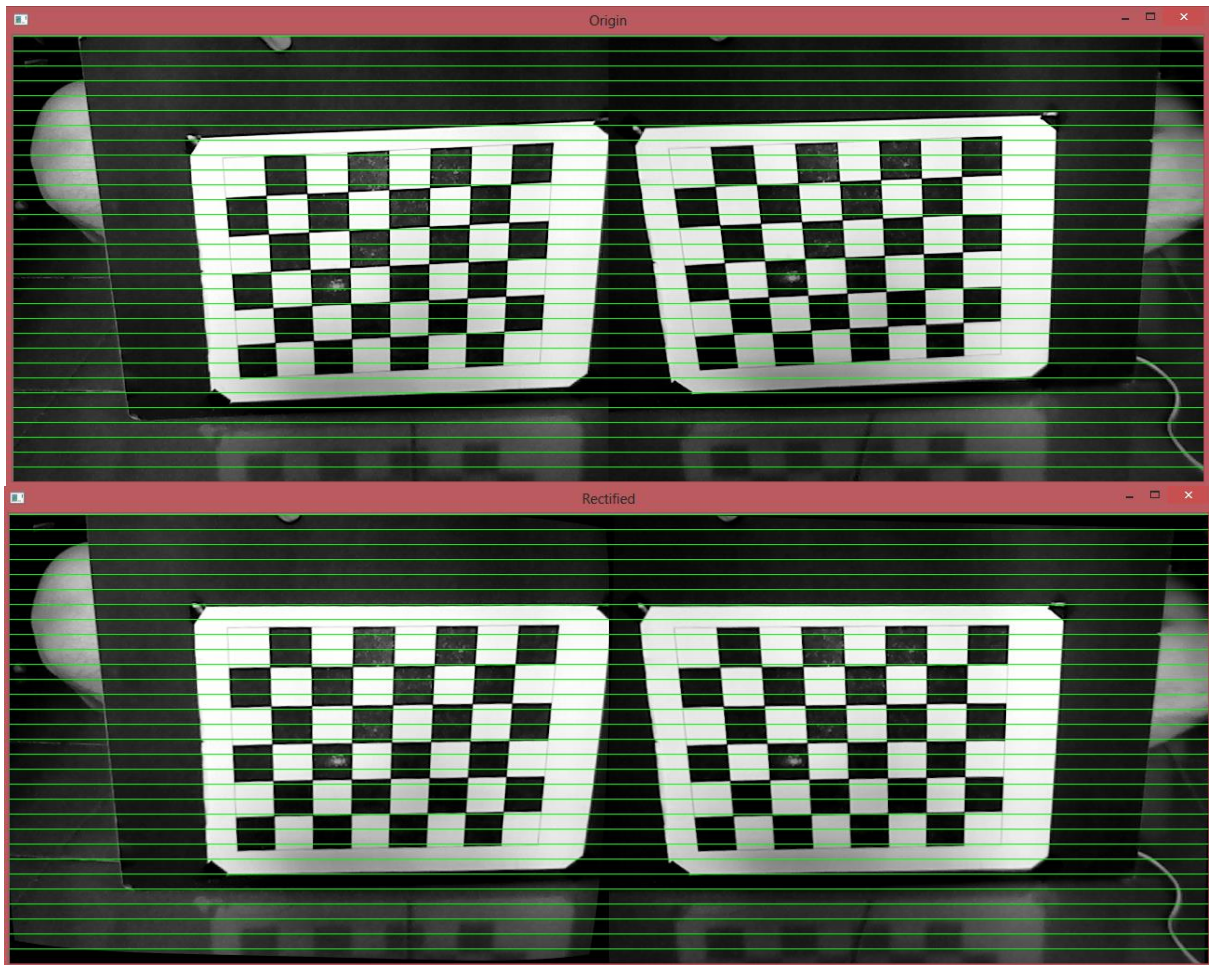
$${}^r_l R = {}^r_w R ({}^l_w R)^T$$

$${}^r_l T = {}^r_w T - {}^r_l R {}^l_w T$$

Như vậy, ${}^r_l R$ và ${}^r_l T$ có thể tìm được thông qua việc calibrate từng camera trái, phải. Công việc này được OpenCV hỗ trợ tính toán nhờ hàm `cvStereoCalibrate()`. Kết quả tìm được được sử dụng để tính toán ma trận Fundamental F . Sử dụng kết quả này, chúng ta có thể thực hiện bước tiếp theo là rectification để “chỉnh sửa” hai hình ảnh từ hai camera sao cho chúng trở nên như được chụp từ hai camera row-aligned, các epipolar line nằm trên cùng một hàng (cùng y) đối với cả hai hình. Nhờ bước rectification mà hai trục quang (principal ray) của hai camera sẽ song song (giao nhau ở vô cùng).



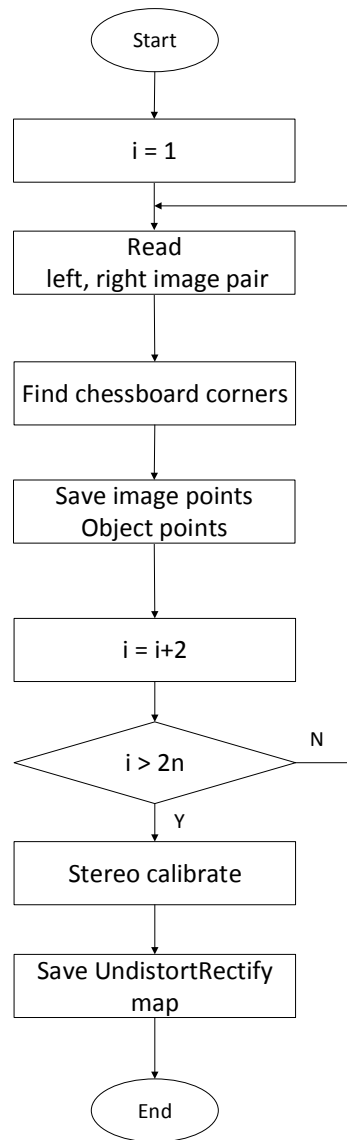
Hình 3.12: Tiến hành stereo calibration cho hệ camera



Hình 3.13: Hình ảnh trước và sau khi thực hiện bước calibration và rectification

Nhận xét: Như trên hình, sau khi đã được rectification các điểm đặc trưng (ví dụ: các corner của chessboard...) nằm trên cùng một đường epipolar line—các đường gạch ngang—cùng tọa độ y . Nhờ đó, khi tìm kiếm các đặc trưng của vật cần nhận dạng, ta có thể giới hạn việc tìm kiếm nằm chỉ trên các epipolar line tương ứng.

Sử dụng thông tin sau khi đã calib được gồm các thông số nội, các distortion của hai camera và ma trận \mathbf{R}, \mathbf{T} , ta tạo ra một “map” để sử dụng khi chuyển từ ảnh ban đầu sang ảnh được rectified.



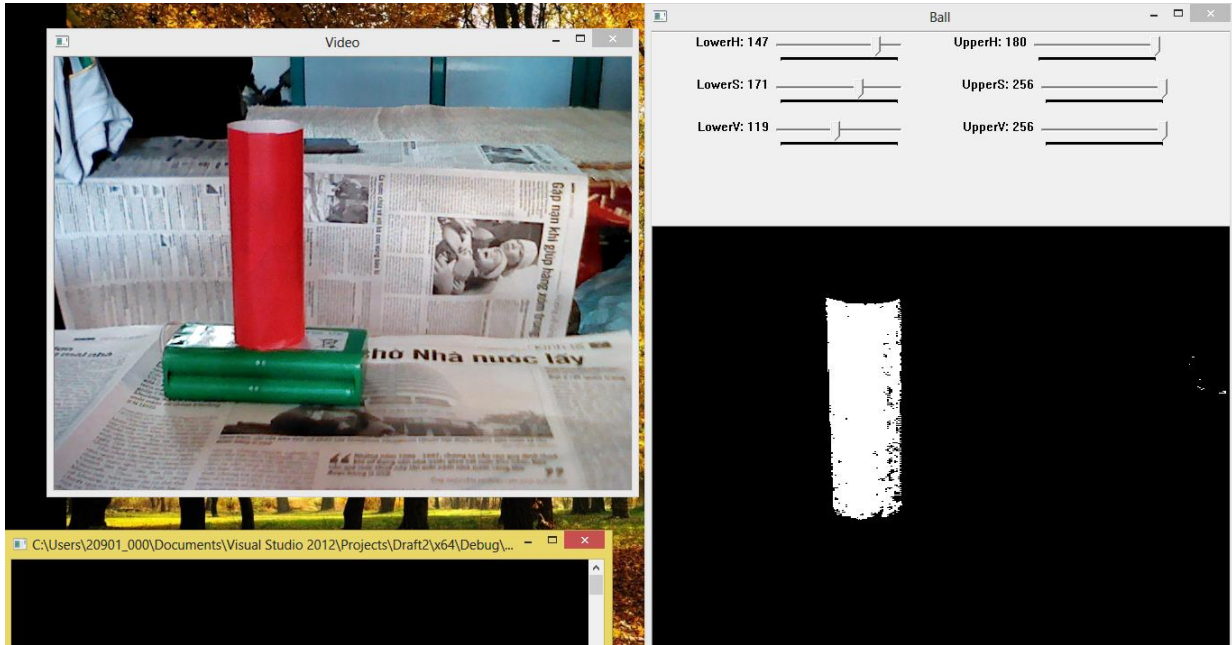
Hình 3.14: Lưu đồ các bước thực hiện stereo calibration

3.3.4 Stereo Correspondence:

Stereo correspondence là bước thứ 3 trong quá trình tính khoảng cách. Ở bước này, ta tìm điểm đặc trưng tương ứng ở hai ảnh để từ đó tính toán tìm disparity. Theo đầu đề, khối vật cần gấp là khối trụ màu.

Từ hình ảnh nhận được sau khi tiến hành calibration và rectification, tiến hành tìm đặc trưng của vật. Đặc trưng của vật được sử dụng để nhận diện là ở đây là màu sắc. Phương pháp này đơn giản nhưng có nhược điểm là môi trường phải không có (hoặc rất ít) nhiễu- vật khác có cùng màu sắc với đối tượng cần gấp.

Một trong những cách đơn giản nhất để nhận diện vật màu là sử dụng threshold để phân ngưỡng ảnh. Trước hết, ảnh ban đầu (dạng RGB) được chuyển về dạng HSV, sau đó tiến hành phân ngưỡng dựa trên đặc trưng các thông số HSV màu sắc của vật. Sau khi đã phân ngưỡng ảnh, kết quả thu được là ảnh binary, các bit có màu đỏ của vật cần gấp được gán giá trị 1, các bit khác là 0. Để loại bớt các nhiễu trong hình, dùng thêm mặt nạ Gaussian 3x3 Kernel để smooth các ảnh binary thu được.



Hình 3.15: Tiến hành chọn khoảng giá trị HSV để phân ngưỡng ảnh

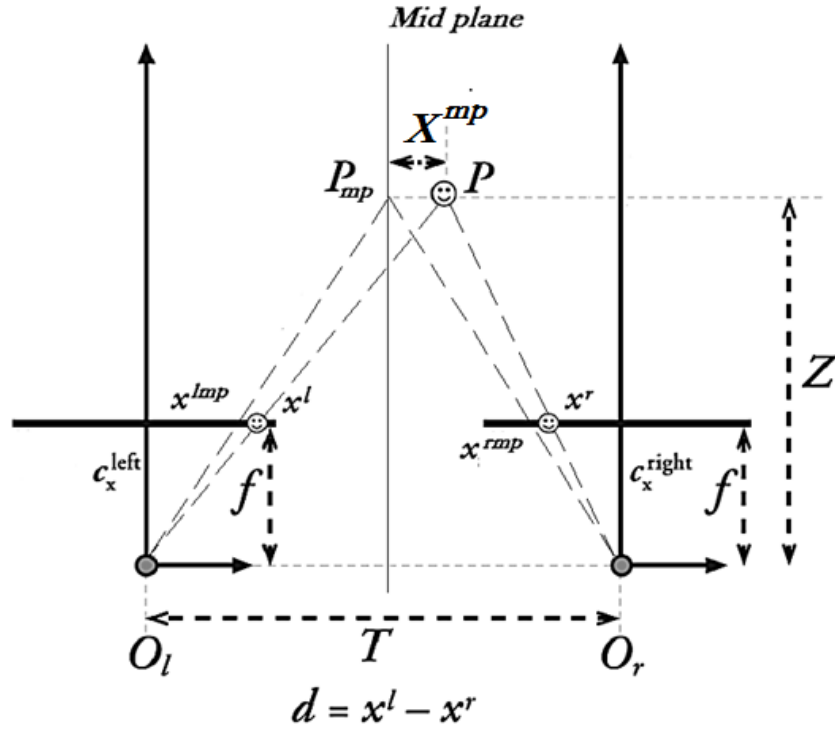
Sau khi đã phân ngưỡng, sử dụng hàm `CvMoments` để xác định vùng trắng (vật) và xác định tọa độ trọng tâm của vùng này trên hình (đây chính là tọa độ của vật trong hình). Một cách để loại bớt nhiễu trong hình là kiểm tra diện tích của vùng trắng nhận diện được. Ở đây, theo thực nghiệm đã tiến hành, chọn điều kiện diện tích:

$$S > 1000 \text{ (pixel} \times \text{pixel)}$$

Như vậy, vị trí tọa độ của vật trong hai hình trái và phải đã được xác định. Từ đó, việc tính disparity trở nên vô cùng đơn giản với $d = x^l - x^r$. Đến đây, ta sử dụng triangulation như đã trình bày để tính toán khoảng cách.

3.4 Xác định vị trí của vật thể:

Như đã trình bày ở trên, ta đã có thể sử dụng 2 camera để đo khoảng cách đến vật thể. Tiếp theo, từ thông tin đã có về khoảng cách của vật thể, ta tiếp tục tìm hiểu cách tính toán các thông tin khác về vị trí của vật thể.



Hình 3.16: Mô hình hệ camera

Xét điểm P_{mp} nằm trên A (Mid plane- mặt phẳng đối xứng của hệ camera) có cùng khoảng cách Z với điểm P . Điểm P_{mp} có tọa độ x của các điểm ảnh trên các mặt phẳng ảnh lần lượt là x^{lmp} và x^{rmp} .

Vì P_{mp} nằm trên mặt phẳng A nên $x^{lmp} + x^{rmp} = \text{image width}$ (bề rộng của ảnh tính theo phương x, đơn vị: pixel)

Ngoài ra, P_{mp} và P có cùng khoảng cách Z nên $x^{lmp} - x^{rmp} = d$

Do đó, giải hệ hai phương trình trên ta được:

$$x^{lmp} = \frac{\text{image width} + d}{2}; x^{rmp} = \frac{\text{image width} - d}{2}$$

Gọi X^{mp} là khoảng cách từ P đến mặt phẳng A. Sử dụng tam giác đồng dạng, dễ dàng suy ra biểu thức sau:

$$X^{mp} = \frac{(x^l - x^{lmp}) Z}{f}$$

Ở trên, ta đã xét trường hợp điểm P nằm bên phải mặt phẳng A. Hoàn toàn tương tự, ta có khoảng cách từ điểm P đến mặt phẳng A khi P nằm bên trái mặt phẳng A như sau (trong trường hợp này $x^{lmp} > x^l$):

$$X^{mp} = \frac{(x^{lmp} - x^l) Z}{f}$$

Như vậy, khoảng cách Z và X^{mp} đã được xác định, ta sử dụng các kết quả này để điều khiển robot di chuyển đến vật cần gấp và thực hiện việc gấp vật.

Vì vật thể được đặt trên mặt phẳng di chuyển của robot và kích thước khối vật đã được xác định trước nên ta không cần thiết phải xác định vị trí vật thể theo phương đứng (phương y của ảnh). Tuy nhiên, sau khi đã xác định được khoảng cách Z đến vật thể, các thông tin khác (kích thước, vị trí theo phương đứng của vật thể) có thể hoàn toàn được xác định một cách dễ dàng nhờ sử dụng các biểu thức và tam giác đồng dạng tương tự như trên.

3.5 Kết luận:

Chương 3 đã giới thiệu về mô hình camera, các thông số của camera cũng như làm quen với các thuật ngữ về camera. Đồng thời, đến đây luận văn cũng đã hoàn thành xây dựng nền tảng kiến thức, giải thuật cho phương pháp đo khoảng cách, xác định vị trí của vật thể sử dụng hệ thống stereo 2 camera.

Quá trình đo khoảng cách trải qua các bước chính sau:

- Calib từng camera để lấy các thông số cần thiết của các camera
- Tiến hành stereo calibration, tạo “map” để loại bỏ radial và tangential distortion nhờ tính toán, điều chỉnh bằng tính toán vị trí tương đối giữa các camera.
- Tìm các điểm đặc trưng tương ứng ở hình trái và phải sau khi đã được rectification (nhân diện khối màu và xác định tọa độ vật trên hình), tìm disparity, còn gọi là bước correspondence
- Tính khoảng cách- sử dụng triangulation, bước này còn được gọi là reprojection.

Mối quan hệ giữa disparity và khoảng cách Z cần đo được thể hiện trong công thức sau:

$$Z = \frac{fT}{x^l - x^r} = \frac{fT}{d}$$

Khoảng cách từ vật thể đến mặt phẳng đối xứng A của hệ camera là:

$$X^{mp} = \frac{|x^l - x^{lmp}| Z}{f}$$

$$\text{Với } x^{lmp} = \frac{\text{image width} + d}{2}$$

Vật nằm bên phải mặt phẳng A khi $x^l > x^{lmp}$, nằm bên trái mặt phẳng A khi $x^l < x^{lmp}$

Chương 4: Tính toán và thiết kế phần cứng

4.1 Giới thiệu:

Đến chương này, nhiệm vụ tiếp theo cần giải quyết là đưa ra được thiết kế phần cứng của mobile robot sao cho nó có thể thực hiện được những yêu cầu đã đặt ra ban đầu. Trong chương này, luận văn đi sẽ vào trình bày thiết kế phần cứng của mobile robot.

Mobile robot cần thiết kế gồm hai phần chính là khung xe robot và tay máy gấp vật. Phần khung xe là nơi để lắp ráp các chi tiết của robot sao cho bảo đảm được vị trí tương quan của các bộ phận trong hệ thống dẫn động. Phần tay máy có 2 bậc tự do, khâu cuối có kết cấu cơ khí làm nhiệm vụ gấp vật.

4.2 Yêu cầu thiết kế:

Việc thiết kế phần cứng mobile robot phải đạt được những yêu cầu sau:

- Phần khung gầm có nhiều phương án thiết kế khác nhau, song chúng phải có chung nhiệm vụ: đảm bảo vị trí tương đối giữa các bộ phận, chi tiết của robot, tiếp nhận tải trọng do các chi tiết lắp trên robot truyền đến, bảo vệ được hệ thống điện bên trong robot.

Về tải trọng thì mobile robot được thiết kế phải có thể tải được một khối lượng khoảng 6kg bao gồm khối lượng của tay máy, camera, laptop, pin, mạch điện...

Mạch điện, camera và tay máy được gắn cố định trên Mobile robot, do đó khung xe phải được thiết kế nơi gắn các bộ phận này.

- Phần tay máy gồm 2 bậc tự do, cần phải có vùng làm việc và khả năng gấp vật tốt.

Vì bỏ qua khối lượng vật được gấp (khối lượng nhỏ) nên chỉ cần tính toán, thiết kế đảm bảo khả năng chịu tải từ chính tải trọng của tay máy

Yêu cầu chỉ tiêu cơ bản của mô hình mobile robot là độ cứng vững tốt và khối lượng nhỏ

Các chi tiết phải được lắp ráp đúng yêu cầu kỹ thuật và phải đảm bảo vị trí tương đối giữa các chi tiết.

4.3 Thiết kế khung xe:

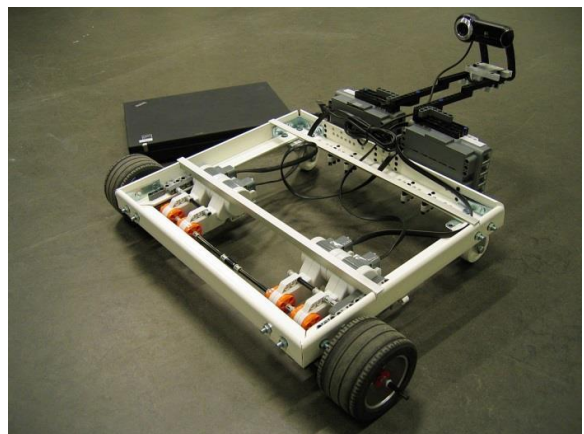
Như đã nêu trên, phần khung xe là nơi lắp ráp các chi tiết của robot. Yêu cầu cơ bản của khung là độ cứng vững và độ chính xác về vị trí tương đối của các bề mặt lắp ghép.

Để thiết kế phần khung của robot ta có một số phương án sau:

-Phương án 1: Khung được chế tạo bằng thép chữ V, và được gắn kết với nhau bằng phương pháp hàn hoặc bulong. Để lắp ráp các chi tiết lên khung ta cần xác định chính xác vị trí tương quan của chúng, sau đó khoan lỗ tại các vị trí đã xác định để lắp ráp chúng. Phương pháp này cho khung xe có độ cứng vững cao nhưng khối lượng robot lớn, gia công cần chính xác cao và khó căn chỉnh khi lắp ráp.

-Phương án 2: Khung mobile robot được chế tạo bằng nhôm định hình, các thanh nhôm được lắp với nhau bằng các bulong. Vì sử dụng nhôm định hình làm mobile robot có khối lượng nhẹ, nhôm định hình có các gân tăng cứng nên khả năng chịu tải đảm bảo. Tuy nhiên, việc bố trí nơi gắn tay máy, camera, mạch điện gặp nhiều khó khăn.

-Phương án 3: Sử dụng thép hoặc nhôm tấm. Để lắp các chi tiết lên khung ta cần xác định vị trí tương quan giữa chúng, sau đó khoan lỗ và lắp ráp các chi tiết bằng bulong đai ốc. Việc sử dụng tấm có thể đảm bảo khối lượng nhỏ, gia công đơn giản, việc bố trí nơi lắp ráp các chi tiết dễ dàng và dễ căn chỉnh trong quá trình lắp ráp. Trọng lượng cần tải của mobile không lớn nên sử dụng tấm có thể đảm bảo độ bền mà khối lượng robot không quá lớn.



Hình 4.1: Ví dụ về các loại khung xe mobile robot

Nhôm tấm có khối lượng nhẹ mà vẫn có thể bảo đảm khả năng chịu tải, do đó, khung mobile robot được thiết kế sử dụng nhôm tấm có bề dày 3mm

4.4 Lựa chọn, thiết kế các nhóm chức năng:

Mobile robot gồm các nhóm chức năng chính như sau: hệ 2 camera, tay máy, laptop, động cơ di chuyển, pin và mạch điện.

4.4.1 Động cơ di chuyển:

Động cơ điện ảnh hưởng trực tiếp đến khả năng làm việc, độ tin cậy và tính an toàn của robot. Việc lựa chọn cần đảm bảo để robot hoạt động đạt công suất yêu cầu, đảm bảo khả năng di chuyển của robot, tiết kiệm điện năng cũng như tăng tuổi thọ cho robot.

- Tính toán tốc độ động cơ:

Theo yêu cầu thiết kế, mobile robot không yêu cầu về tốc độ di chuyển, tuy nhiên, để dễ dàng lựa chọn động cơ và tính toán điều khiển, lựa chọn tốc độ di chuyển của mobile robot là 0.15 m/s

Bánh xe được sử dụng có bán kính $r = 35 \text{ mm}$

Tốc độ quay cần thiết của trục đầu ra động cơ:

$$n_{dc} = 0.15 \times r \times \frac{60}{2\pi} = 50 \frac{\text{vòng}}{\text{phút}}$$

- Tính công suất động cơ:

Giả sử khối lượng m của robot phân bố đều trên ba bánh, phản lực trên mỗi bánh là $mg/3$, xét trong trường hợp đơn giản khi robot di chuyển thẳng, bỏ qua lực ma sát theo phương vuông góc với bánh xe.

Để mobile robot di chuyển, lực phát động do moment động của động cơ tạo ra phải thỏa mãn

$$F = \frac{T}{r} > F_{ms} = \mu \frac{mg}{3}$$

Bánh xe robot làm bằng nhựa, môi trường hoạt động trên nền gạch, do đó, một cách gần đúng chọn hệ số ma sát $\mu = 0.3$

$$T_1 > 0.3 \times 6 \times \frac{9.8}{3} \times 35 = 205.8 \text{ Nmm}$$

Suy ra công suất tối thiểu động cơ:

$$P = \frac{Tn_{đc}}{9.88 \times 10^3} = 1 \text{ W}$$

- Dựa vào các thông số yêu cầu đã tính toán, trong mô hình này, lựa chọn động cơ Leison SG555123000 với các thông số sau:

Điện áp hoạt động 12VDC, động cơ có tích hợp sẵn hộp giảm tốc có tốc độ trục đầu ra 60 RPM, công suất 6W, dòng không tải < 140 mA, dòng tải tối đa < 1 A.



Hình 4.2: Động cơ DC Leison SG555123000

4.4.2 Hệ 2 camera:

Hệ thống camera là một trong những bộ phận quan trọng của mobile robot. Việc thiết kế hệ camera cần phải đảm bảo các yêu cầu sau:

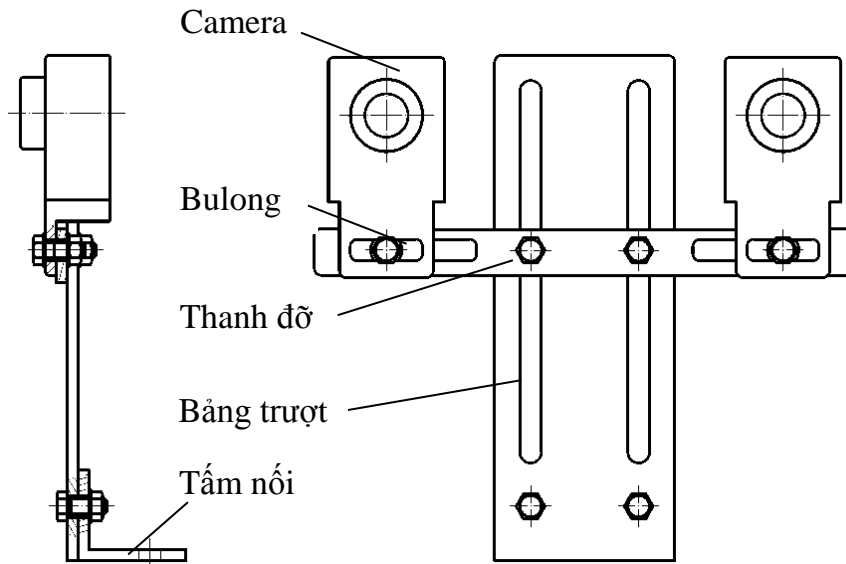
-Có thể thay đổi khoảng cách giữa hai camera.

Như đã trình bày, khoảng cách giữa hai camera ảnh hưởng đến việc xử lý dữ liệu, đo khoảng cách, vùng quan sát, do đó hệ camera phải điều chỉnh được khoảng cách giữa các camera.

-Có khả năng điều chỉnh được vị trí tương đối giữa các camera để giảm bớt sai số của hệ camera. Hệ hai camera phải đạt được độ đồng phẳng nhất định. Như đã đề cập

ở chương trước, việc bố trí các camera phải đảm bảo sao cho đạt được độ đồng phẳng (frontal parallel, row-aligned)

Với yêu cầu đặt ra như vậy, hệ camera được thiết kế như sau:



Hình 4.3: Thiết kế của hệ camera

Với thiết kế ở trên, camera có thể được điều chỉnh khoảng cách, độ cao cho phù hợp nhờ thay đổi vị trí các bulong trong các rãnh.

Ngoài ra, bulong được sử dụng là M4, các rãnh (hoặc lỗ) có chiều rộng (đường kính) là 6mm. Nhờ đó, vị trí tương đối của các camera có thể được điều chỉnh trong một phạm vi nhất định để tăng độ chính xác khi calib camera.

4.4.3 Tay máy:

Như đã trình bày trong chương 2- Lựa chọn phương án thiết kế tay máy, tay máy cần thiết kế có hai bậc tự do, động cơ truyền động là động cơ RC, sử dụng khâu cuối là kết cấu cơ khí truyền động bằng bánh răng.

Yêu cầu đặt ra khi thiết kế tay máy là đảm bảo không gian làm việc và khâu cuối gặp được khối vật đơn giản, yêu cầu về khối lượng vật được bỏ qua.

Đế của tay máy được gắn trực tiếp lên khung xe, do đó, phần đế cần được thiết kế không tốn quá nhiều diện tích, tay máy ít che khuất tầm nhìn của các camera,

Các khâu cần được thiết kế đơn giản, giảm trọng lượng tay máy.

- Lựa chọn động cơ RC servo:

Động cơ sử dụng cho tay máy phải đáp ứng một số yêu cầu sau: khối lượng và không quá lớn để đảm bảo yêu cầu về khối lượng và kích thước tay máy, momen xoắn trung bình 6-9 kg/cm, tốc độ quay nhanh.

Động cơ được sử dụng trong tay máy là động cơ RC servo TowerPro MG995. Đây là loại động cơ phổ biến, có chất lượng tương đối tốt, các thông số thỏa mãn các yêu cầu đặt ra đối với nhiều ứng dụng, giá thành chấp nhận được .

Các thông số cơ bản của loại động cơ này là:

RC servo TowerPro 995			
Kích thước	$40.7 \times 19.7 \times 42.9 \text{ mm}$	Khối lượng	55g
Tốc độ quay	$0.20s/60^\circ (4.8V)$ $0.16s/60^\circ (6V)$	Điện áp hoạt động	4.8 – 7.2V
Momen xoắn	$8.5kg/cm(4.8V), 10kg/cm(6V)$	Loại bánh răng	Kim loại

Bảng 4.1: Các thông số của động cơ RC servo TowerPro 995



Hình 4.4: Động cơ RC servo TowerPro M995

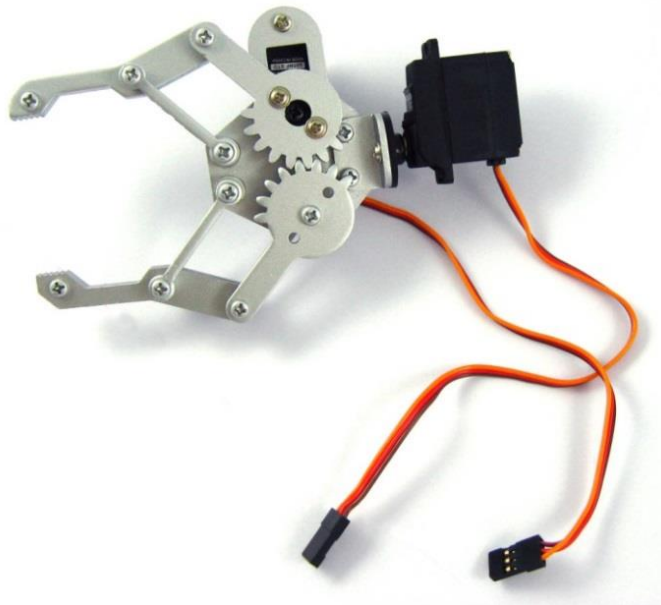
- Lựa chọn vật liệu và phương pháp gia công:

Hiện nay, tay máy có thể được chế tạo từ các vật liệu phổ biến như thép, nhôm, nhựa, mica... Đối với các vật liệu như thép, nhôm việc gia công chế tạo khó khăn và làm gia tăng khối lượng của tay máy.

Do đó, vật liệu được sử dụng với tay máy này là mica, sử dụng các bulong để lắp các bộ phận với nhau. Hiện nay, việc gia công mica sử dụng các máy cắt laser khá phổ biến, nhờ đó, độ chính xác gia công cao, giá thành rẻ và thuận tiện.

-Lựa chọn thiết kế khâu cuối:

Khâu cuối của tay máy là bộ phận thực hiện thao tác gấp vật sau khi tay máy đã di chuyển tới vị trí cần thiết. Thiết kế khâu cuối của tay máy được chọn như sau vì đảm bảo khả năng gấp vật, dễ dàng chế tạo và lắp ráp.



Hình 4.5: Phương án thiết kế khâu cuối- cơ cấu kẹp sử dụng bánh răng để truyền động

Việc truyền động sử dụng động cơ RC servo TowerPro M995. Động cơ quay, truyền động cho bánh răng và làm các khâu khác chuyển động theo. Để đảm bảo hoạt động mượt mà của khâu cuối, bánh răng cần có biên dạng thân khai, đảm bảo khả năng ăn khớp giữa hai bánh răng.

Qua quá trình phân tích, lựa chọn phương án, tay máy được thiết kế như sau:



Hình 4.6: Thiết kế tay máy

-Kiểm tra bền và khả năng tải của tay máy:

Tay máy được thiết kế theo quy trình thiết kế kết cấu sau đó kiểm tra bền và tải trọng các chi tiết cần kiểm tra. Vì trọng lượng bản thân tay máy nhẹ (khoảng 0.8 kg), hơn nữa khối lượng vật được gấp được bỏ qua, do đó với vật liệu và loại động cơ đã chọn, thiết kế tay máy có thể đảm bảo khả năng chịu tải và độ bền, không cần thiết phải tiến hành kiểm tra bền và tải trọng.

4.4.4 Laptop, pin và mạch điện:

Robot được thiết kế phải có khả năng chở theo laptop để xử lý thông tin nhận về từ camera. Bên cạnh đó, robot cần được thiết kế bộ phận để gắn pin và mạch điều khiển robot. Ở trên, khung xe được thiết kế từ nhôm tấm, do đó, ta cũng sử dụng nhôm tấm để thiết kế bộ phận chở laptop.

Để tiện cho việc quan sát kết quả hoạt động trên màn hình máy tính và cân chỉnh khi robot hoạt động, ta lựa chọn phương án bố trí theo tầng. Laptop được đặt lên trên cùng, khoang dưới chứa pin và mạch điện

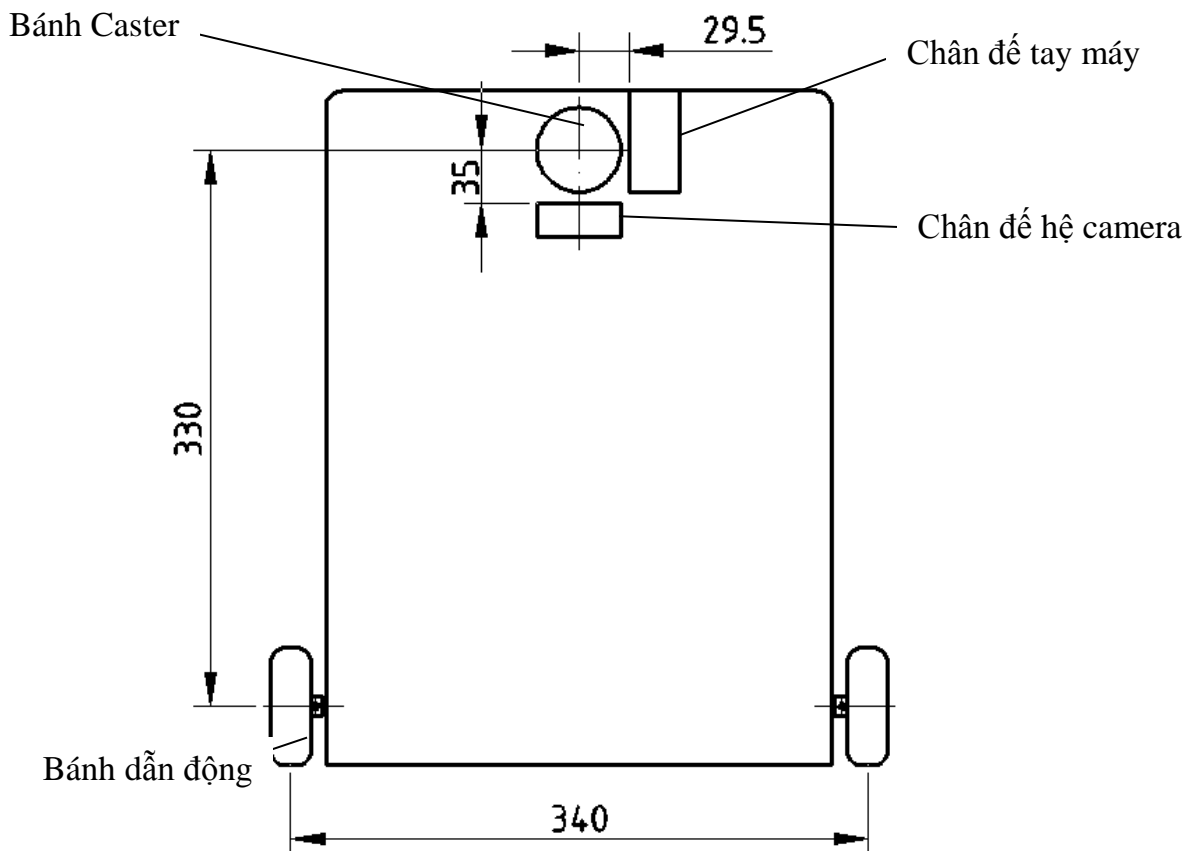
4.5 Bố trí các nhóm chức năng:

Sau khi đã chọn phương án thiết kế khung xe, ta cần bố trí các nhóm chức năng để thực xác định vị trí tương quan của các bộ phận này, sau đó tiến hành gia công, lắp ráp.

-Bố trí động cơ, bánh xe: Khoảng cách các bánh xe dẫn động ảnh hưởng đến quá trình điều khiển di chuyển của robot , do đó khoảng cách này cần được xác định khi thiết kế.

-Bố trí hệ camera, tay máy: vùng quan sát của camera là một trong những yếu tố quan trọng cần được chú ý khi bố trí các bộ phận. Hệ camera và tay máy cần được bố trí sao cho mobile robot có vùng quan sát rộng, ít bị che khuất; tay máy không bị vướng khi gấp vật. Theo kích thước và mô hình đã thiết kế của hệ camera, tay máy, hai bộ phận này được bố trí như hình dưới.

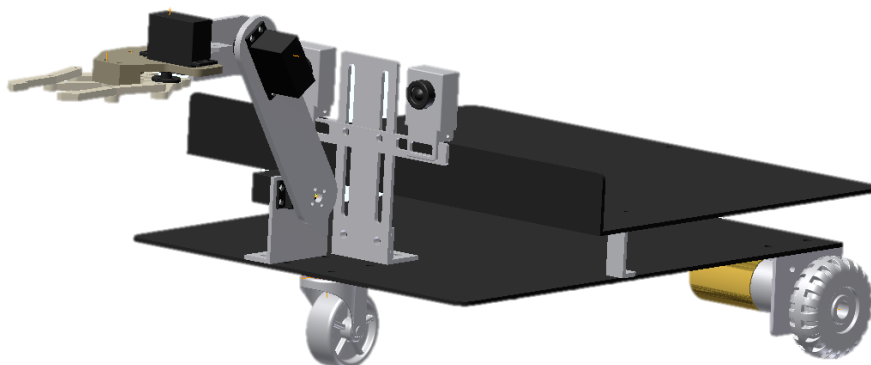
Các nhóm chức năng của mobile robot được bố trí như sau:



Hình 4.7: Bố trí các nhóm chức năng

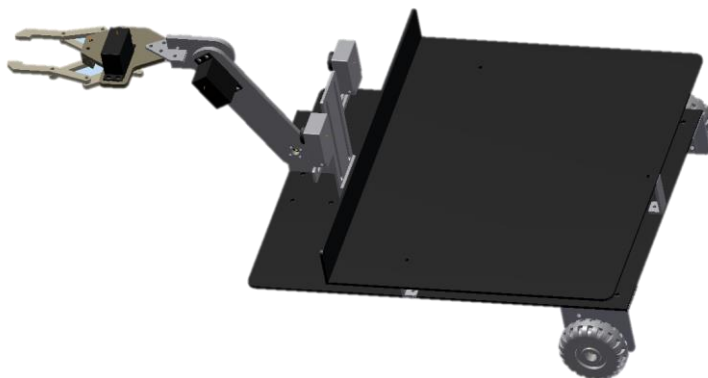
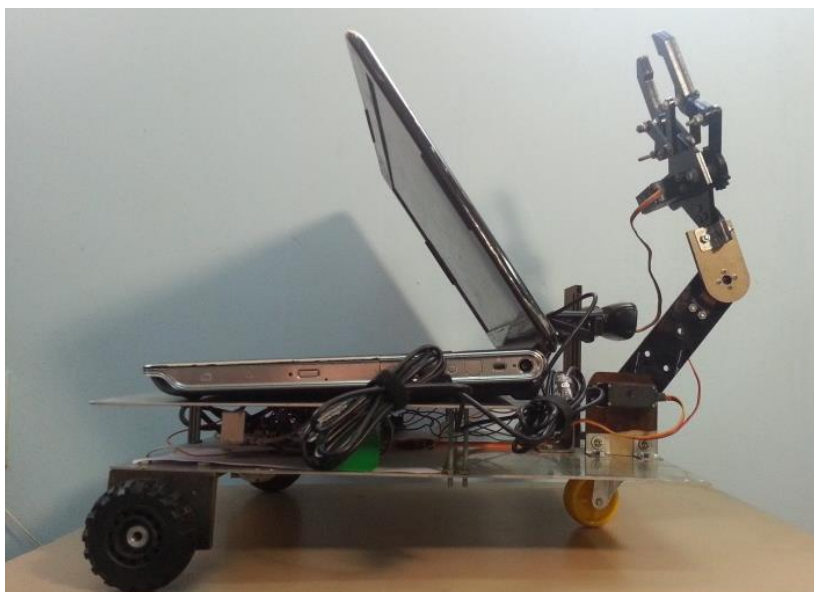
4.6 Kết luận:

Sau quá trình phân tích và thiết kế, robot được thiết kế như sau:



Hình 4.8: Thiết kế 3D của mobile robot

Sau quá trình thiết kế, tiến hành gia công, chế tạo mô hình thực tế của mobile robot



Hình 4.9: Mô hình mobile robot thực tế và mô phỏng

Chương 5: Thiết kế hệ thống điều khiển

5.1 Giới thiệu:

Hệ thống điều khiển được xem là “bộ não” của robot, đảm nhận nhiệm vụ xử lý thông tin, truyền nhận tín hiệu, điều khiển robot thực hiện các thao tác đúng theo yêu cầu. Robot phải đáp ứng linh hoạt và chính xác trong quá trình hoạt động. Do đó, phải tiến hành chọn các phương pháp, giải thuật, công cụ và cách xử lý sao cho đáp ứng được yêu cầu của đề bài đồng thời phù hợp với những điều kiện khách quan cũng như chủ quan khi thực hiện luận văn này. Hệ thống điều khiển phải thỏa mãn một số yêu cầu sau:

- Mạch điện được thiết kế hợp lý đảm bảo công suất, hoạt động ổn định.
- Mobile robot đáp ứng nhanh và chính xác theo yêu cầu đã lập trình sẵn.
- Tiết kiệm năng lượng, robot có thể hoạt động trong thời gian lâu.
- Có khả năng mở rộng và phát triển thêm sau này.

Ở chương này, luận văn sẽ trình bày thiết kế hệ thống điều khiển, bao gồm trình bày sơ đồ khối của mobile robot, bên cạnh đó cũng trình bày thiết kế các mạch điều khiển, mạch công suất cho mobile robot.

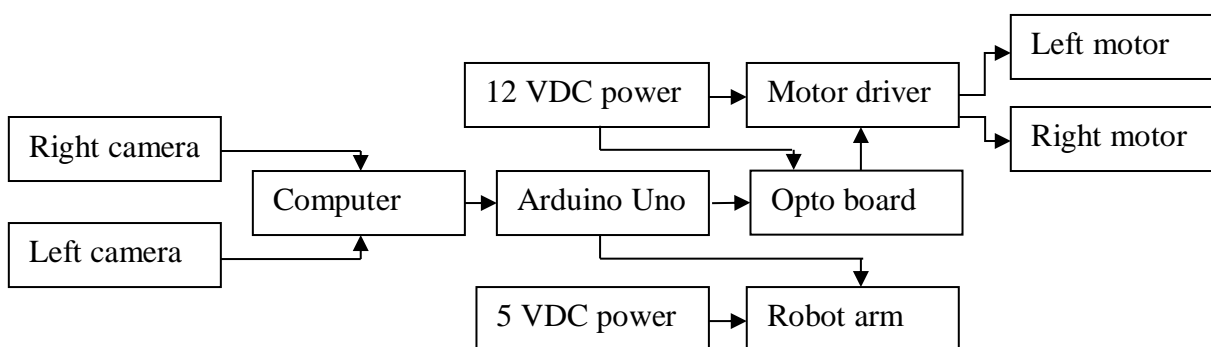
5.2 Sơ đồ khối hệ thống điều khiển của mobile robot:

Từ yêu cầu trên và qua quá trình phân tích lựa chọn phương án ở chương 2, hệ thống điều khiển được xác định gồm những thành phần chính sau:

Một laptop được sử dụng để nhận tín hiệu và xử lý thông tin từ hai camera (Vì CV là lĩnh vực cần có tốc độ và khả năng xử lý cao nên sử dụng laptop- là thiết bị có khả năng xử lý mạnh mẽ mà vẫn đảm bảo được tính linh hoạt của mobile robot.) Sau khi nhận và xử lý thông tin, laptop truyền tín hiệu điều khiển xuống vi điều khiển thông qua giao thức RS232

Vi điều khiển: Sử dụng module mạch Arduino Uno (vi điều khiển ATMEGA328P-PU) để truyền nhận tín hiệu giao tiếp với máy tính và xuất tín hiệu điều khiển tay máy, động cơ di chuyển hoạt động.

Sơ đồ khối của bộ điều khiển mobile robot như sau:



Hình 5.1: Sơ đồ khối hệ thống điều khiển

5.3 Thiết kế mạch điều khiển và mạch công suất của robot:

Theo thiết kế đã đưa ra, để robot có thể thực hiện các tính năng đã thiết kế, robot gồm các cụm mạch chức năng sau: mạch vi điều khiển trung tâm, mạch nguồn và mạch driver động cơ dẫn động.

Mạch	Mô tả
Vi điều khiển trung tâm	Mạch chứa vi điều khiển, được lập trình làm nhiệm vụ truyền nhận tín hiệu RS232 từ laptop, sau đó tính toán, xử lý dữ liệu và xuất tín hiệu điều khiển các bộ phận đầu ra (động cơ di chuyển, động cơ tay máy)
Mạch nguồn	Mạch nguồn làm nhiệm vụ biến điện áp 14.4 V của pin thành nguồn 5V cung cấp cho các động cơ của tay máy, đồng thời mạch nguồn cũng chuyển điện áp của pin thành 12V để cung cấp cho mạch driver điều khiển động cơ di chuyển
Mạch driver động cơ di chuyển	Mạch driver chuyển tín hiệu từ vi điều khiển ở mức điện áp 5V thành tín hiệu cho động cơ hoạt động 12V từ nguồn. Có chức năng cách ly nguồn điện giữa mạch điều khiển và mạch nguồn- do hai mạch này sử dụng mức điện áp khác nhau.
Mạch Opto	Mạch sử dụng opto cách ly nguồn điều khiển với nguồn động lực

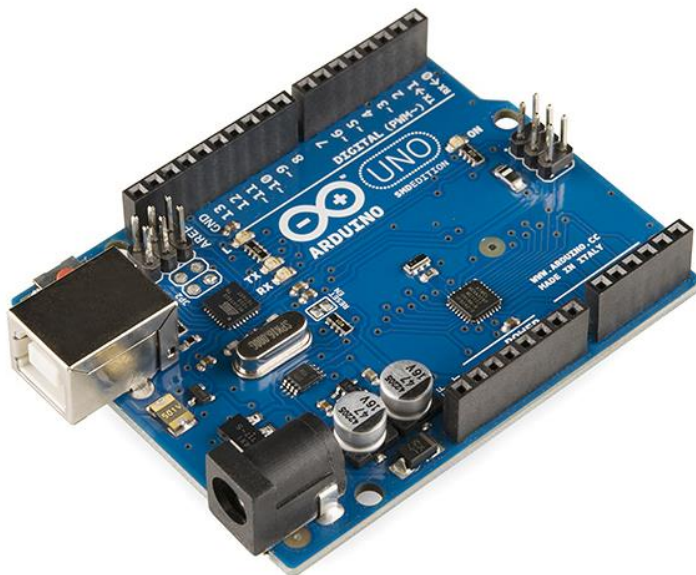
Bảng 5.1: Các mạch chức năng trên robot

5.3.1 Mạch vi điều khiển:

Với những chức năng đã phân tích ở trên, mạch vi điều khiển được thiết kế cần đáp ứng được những yêu cầu sau:

- Mạch vi điều khiển có phải đảm bảo được chức năng kết nối, giao tiếp được với máy tính bằng giao tiếp RS232
- Mạch có tốc độ xử lý nhanh, hoạt động ổn định
- Mạch có ít nhất 5 chân có thể tạo xung PWM để điều khiển các động cơ hoạt động (3 động cơ của tay máy và 2 động cơ di chuyển)
- Ngoài ra, mạch cần có một số port I/O phục vụ việc xuất tín hiệu hiển thị trạng thái hoạt động của robot để tiện lợi cho quá trình kiểm tra lập trình điều khiển và mở rộng chức năng robot (xuất LED, LCD...)

Với yêu cầu như vậy, mạch Arduino Uno dựa trên nền tảng là vi điều khiển ATMEGA328P-PU là mạch thích hợp cho robot này. Arduino Uno là bo mạch vi điều khiển linh hoạt, dễ sử dụng, có thể thực hiện các chức năng như giao tiếp với máy tính, điều khiển các thiết bị ngoại vi như LED, động cơ, đèn, đọc tín hiệu từ cảm biến analog...



Hình 5.2: Mạch Arduino Uno

Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, các thiết bị, mạch hỗ trợ được module hóa nên việc phát triển, mở rộng cho

robot rất dễ dàng. Ngoài ra, đây là mạch đã được thương mại hóa nên có thể đảm bảo tin cậy, hoạt động ổn định.

Mạch Arduino Uno có 14 chân digital input/output (trong đó có 6 chân có thể xuất PWM), 6 chân analog input. Mạch Uno có tích hợp chip chuyển đổi kết nối USB-serial, do đó, khi kết nối với máy tính, máy tính sẽ nhận cổng kết nối là một cổng COM serial port (thường là COM4). Mạch Arduino Uno có thể sử dụng nguồn từ kết nối USB để hoạt động.

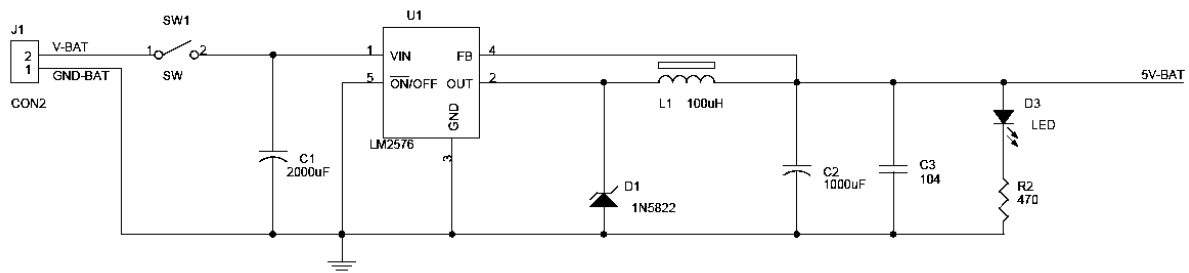
Với những đặc điểm như trên, mạch Arduino Uno được chọn sử dụng để điều khiển mobile robot đã thiết kế.

5.3.2 Mạch nguồn:

Trước khi phân tích thiết kế mạch nguồn cho mobile robot cần xác định nguồn cấp và các nguồn điện cần cung cấp cho robot:

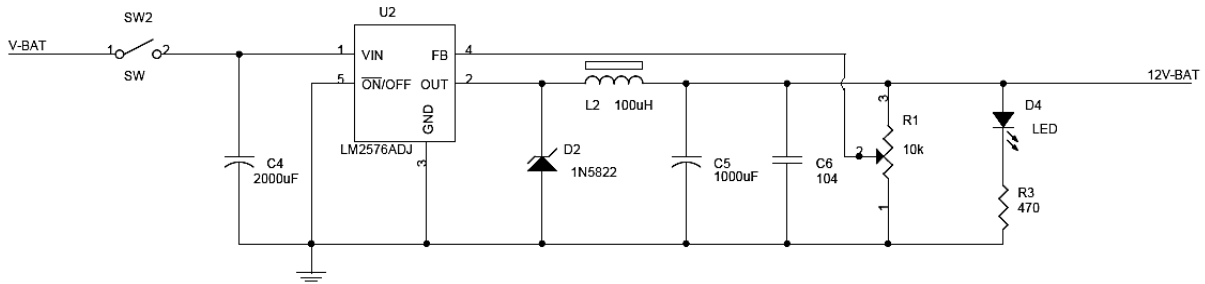
Robot sử dụng pin 14.4V. Ta cần cung cấp điện áp 5V cho động cơ RC và 12V cho động cơ di chuyển hoạt động. Vì mạch Arduino Uno có thể được cấp nguồn từ kết nối USB với máy tính, do đó, không cần thiết kế mạch cấp nguồn cho mạch vi điều khiển này.

Như vậy, trước hết, cần tạo điện áp ngõ ra 5V với điện áp vào là 14.4V. Để thực hiện việc này có rất nhiều IC chuyên dụng như 7805, LM317, LM2672, LM2674, LM2576. Vì điện áp 5V được sử dụng cho động cơ RC hoạt động, do đó sử dụng IC ổn áp có dòng cung cấp lớn để tránh gặp hiện tượng dòng không đủ lớn. IC LM2576 là một trong những IC đáp ứng được các yêu cầu kỹ thuật trên (dòng cấp tối đa là 3A). Sau đây là sơ đồ nguyên lý mạch ổn áp 5V



Hình 5.3: Sơ đồ nguyên lý mạch ổn áp 5V

Tiếp theo, cần tạo ra điện áp 12V để cung cấp cho 2 động cơ di chuyển. Trong các IC ổn áp, IC LM2576-ADJ là một IC nằm trong họ LM2576, có thể tạo điện áp ngõ ra từ 1.23V đến 37V mà dòng cấp tối đa có thể đạt đến 3A. Sau đây là sơ đồ nguyên lý mạch ổn áp 12V, việc điều chỉnh điện áp ra được thực hiện nhờ điều chỉnh biến trở R1.



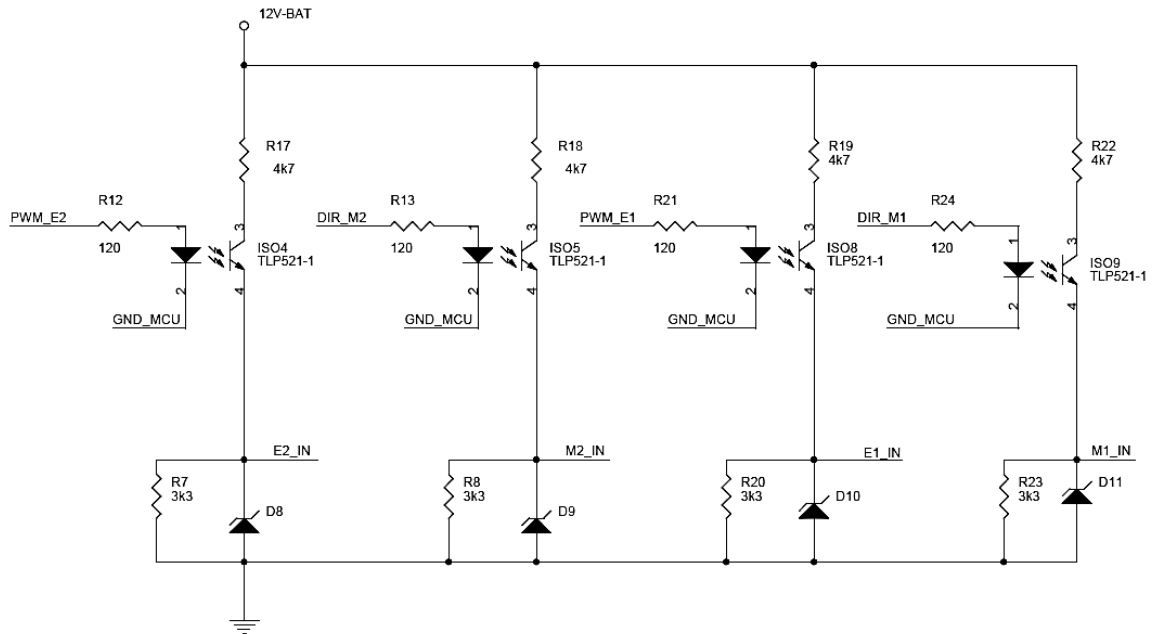
Hình 5.4: Sơ đồ nguyên lý mạch ổn áp 12V



Hình 5.5: Pin và mạch ổn áp 5V và 12V

5.3.3 Mạch Opto cách ly:

Nhiệm vụ của mạch Opto là cách ly nguồn động lực và nguồn điều khiển. TLP521-1 là loại opto thông dụng, dễ tìm thấy trên thị trường có thể đáp ứng được yêu cầu trên. Để thực hiện nhiệm vụ cách ly, mạch được thiết kế như sau:

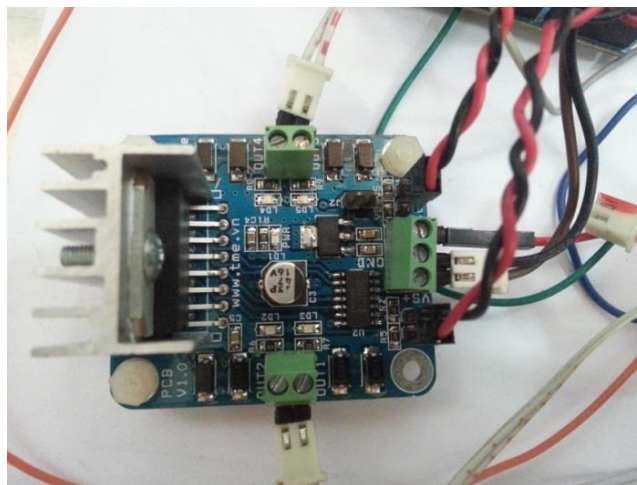


Hình 5.6: Mạch Opto cách ly

5.3.4 Mạch driver cho động cơ di chuyển:

Hai động cơ có điện áp định mức 12V, dòng không tải < 140mA và dòng tối đa có tải là 1A. Do đó, mạch driver được thiết kế phải có khả năng đáp ứng các đặc điểm trên.

L298N là một driver chip tích hợp sẵn hai mạch cầu H bên trong với chuẩn điều khiển TTL, không có diode nội bảo vệ Mosfet. Chịu tải tối đa trên mỗi cầu là 2A, điện áp 40V. Hai mạch cầu có thể được sử dụng một cách độc lập, do đó một driver chip L298N có thể được dùng để điều khiển 2 động cơ cùng lúc. Như vậy, driver chip L298N hoàn toàn thỏa mãn yêu cầu đặt ra cho mạch driver.

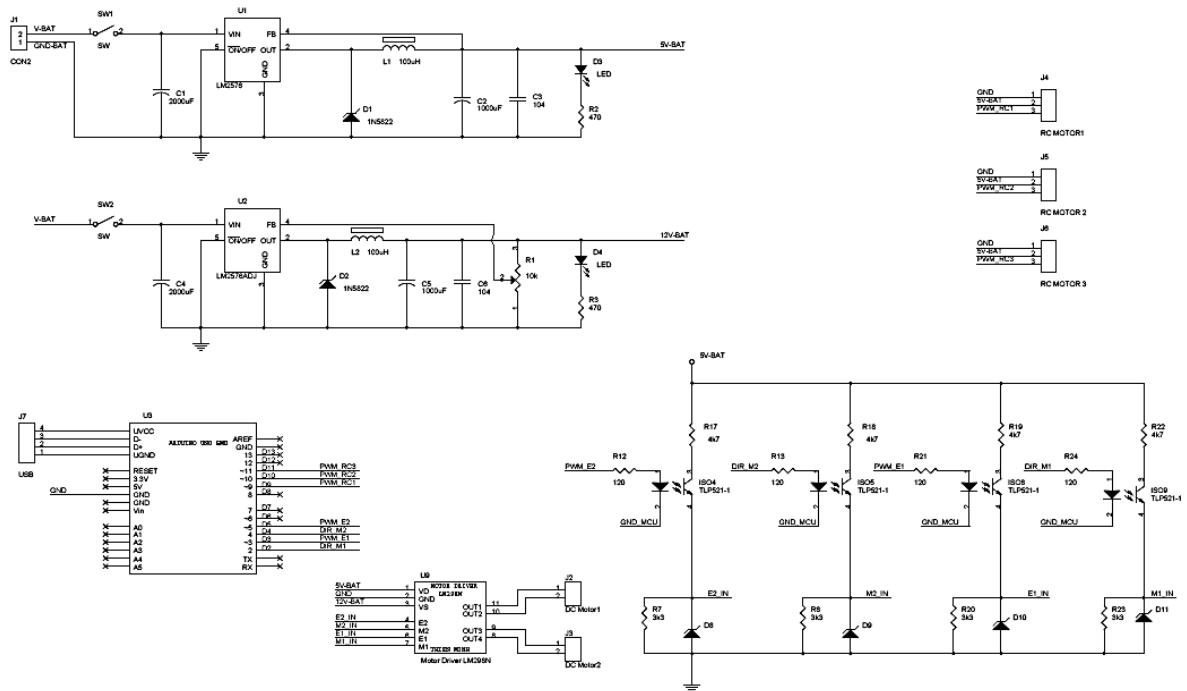


Hình 5.7: Mạch driver động cơ di chuyển

5.4 Kết luận:

Như vậy, chương 5 đã hoàn thành lựa chọn, thiết kế hệ thống điều khiển cho robot.

Mobile robot bao gồm mạch vi điều khiển Arduino được kết nối với máy tính qua giao tiếp serial RS232, mạch opto cách ly, mạch driver cho các động cơ DC dẫn động, mạch nguồn 12V và mạch nguồn 5V để cung cấp cho mobile robot hoạt động.



Hình 5.8: Sơ đồ mạch điện của mobile robot

Chương 6: Động học robot và giải thuật điều khiển

6.1 Giới thiệu:

Sau khi đã hoàn thành xong thiết kế cơ khí và mạch điều khiển, mạch động lực cho mobile robot, đến đây luận văn phân tích trình bày giải thuật và lập trình điều khiển mobile robot hoạt động như đầu đề đã đặt ra.

Yêu cầu hoạt động của robot là sử dụng CV tìm vật, đo khoảng cách, xác vị trí vật thể và điều khiển mobile robot di chuyển đến vị trí mong muốn và gấp vật.

Các công việc mobile robot cần thực hiện để đáp ứng yêu cầu trên bao gồm:

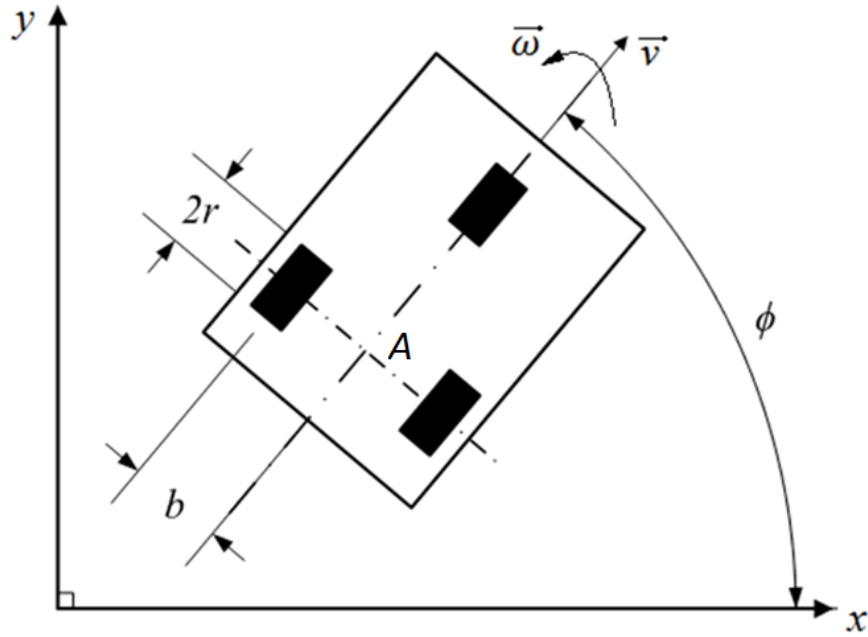
- Đầu tiên, laptop nhận và xử lý tín hiệu từ hai camera xác định vị trí của vật. Nếu không có vật trong vùng quan sát của hệ camera, robot thực hiện thao tác xoay tại chỗ để tìm vật.

- Tiếp theo, ta cũng cần tiến hành đo khoảng cách Z đến vật cần gấp. Sau đó, khoảng cách Z này được sử dụng để điều khiển mobile robot di chuyển đến vị trí sao cho vật nằm trong không gian làm việc của tay máy. Khi vật đã nằm trong không gian làm việc của tay máy, khoảng cách Z tiếp tục được sử dụng để điều khiển tay máy gấp vật.

Lưu ý: Theo thiết kế, khâu cuối tay máy có mặt phẳng đối xứng trùng với mặt phẳng đối xứng của hệ camera. Vì tay máy có hai bậc tự do, chỉ thực hiện được thao tác gấp khi trục đối xứng vật nằm trên mặt phẳng đối xứng của khâu cuối tay máy. Do đó để gấp được vật, phải cho robot luôn luôn điều chỉnh sao cho đảm bảo vị trí của vật thể đối với mobile robot như đã nói.

Như vậy, để thực hiện điều khiển mobile robot, trước hết cần tiến hành tính toán động học cho mobile robot có 2 bánh dẫn động 1 bánh caster, tính động học ngược để điều khiển tay máy 2 DOF. Sau đó, đưa ra thiết kế giải thuật điều khiển mobile robot di chuyển, gấp vật.

6.2 Phương trình động học của mobile robot:



Hình 6.1: Mô hình hóa robot

Giả sử các bánh xe lăn không trượt. Phương trình động học của mobile robot:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (6.1)$$

Với $A(x, y)$ là tọa độ giao điểm trục 2 bánh xe với trục của robot, ϕ là góc tạo bởi trục robot với trục x , v và ω lần lượt là tốc độ dài và tốc độ góc của robot.

Mối quan hệ giữa v, ω với tốc độ góc của 2 bánh xe chủ động là:

$$\begin{bmatrix} \omega_{rw} \\ \omega_{lw} \end{bmatrix} = \begin{bmatrix} 1/r & b/r \\ 1/r & -b/r \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (6.2)$$

Với ω_{rw}, ω_{lw} là tốc độ góc của bánh phải và bánh trái, $2b$ là khoảng cách giữa hai bánh xe, r là bán kính của bánh xe.

Từ (6.2) ta có:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \omega_{rw} \\ \omega_{lw} \end{bmatrix} \quad (6.3)$$

Kết hợp (6.1), (6.3), ta được:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \omega_{rw} \\ \omega_{lw} \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} (r \cos \phi)/2 & (r \cos \phi)/2 \\ (r \sin \phi)/2 & (r \sin \phi)/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \omega_{rw} \\ \omega_{lw} \end{bmatrix} \quad (6.4)$$

Các chuyển động cơ bản cần thực hiện của mobile robot gồm chuyển động thẳng và chuyển động xoay tại chỗ.

Để mobile robot di chuyển thẳng, góc ϕ tạo bởi trục robot với trục x không đổi, suy ra $\dot{\phi} = 0$. Từ phương trình động học mobile robot trên,

$$\dot{\phi} = \frac{r(\omega_{rw} - \omega_{lw})}{2b}$$

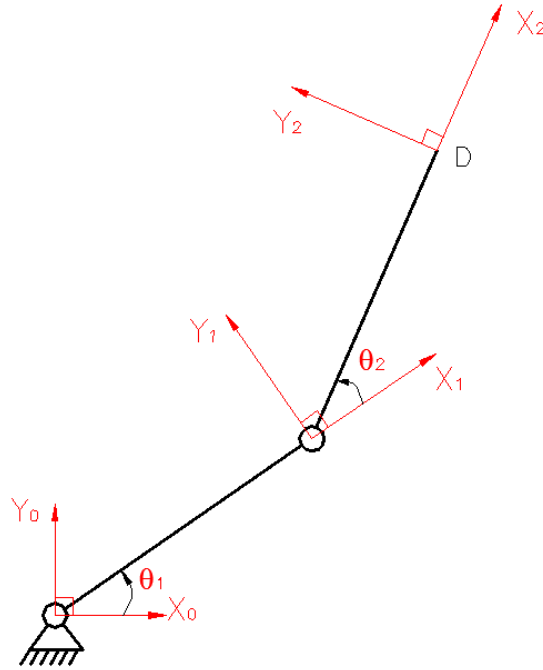
Suy ra, để robot di chuyển thẳng ta phải điều khiển sao cho $\omega_{rw} = \omega_{lw}$, có nghĩa là hai bánh trái và phải có cùng tốc độ góc, tùy thuộc vào tốc độ đi thẳng mong muốn mà chọn giá trị tốc độ góc bánh trái và phải cho hợp lý với $v = r\omega_{rw} = r\omega_{lw}$

Để thực hiện chuyển động xoay tại chỗ quanh tâm A , tọa độ $A(x, y)$ không đổi, tức là $\dot{x} = 0, \dot{y} = 0$. Theo (6.4),

$$\begin{aligned} \dot{x} &= \frac{(\omega_{rw} + \omega_{lw})(r \cos \phi)}{2} \\ \dot{y} &= \frac{(\omega_{rw} + \omega_{lw})(r \sin \phi)}{2} \end{aligned}$$

Do đó, để thực hiện chuyển động xoay tại chỗ $\omega_{rw} = -\omega_{lw}$, tức là hai bánh trái và phải phải có xoay cùng tốc độ, ngược chiều nhau. Tốc độ góc của robot trong trường hợp này là $\omega = \omega_{rw}r/b$

6.3 Phương trình động học tay máy:



Hình 6.2: Coordinate frames của tay máy robot

Ta gán các hệ tọa độ cho tay máy như hình trên. Từ sơ đồ động của tay máy, lập bảng các thông số D-H:

i	α_i	a_i	d_i	θ_i
1	0	L_1	0	θ_1
2	0	L_2	0	θ_2

Bảng 6.1: Các thông số D-H của tay máy

Suy ra, các ma trận transformation:

$${}^0_1T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & L_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & L_1\sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1_2T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

-Tính động học thuận của tay máy:

$${}^0_2T = {}^0_1T {}^1_2T = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & L_2 \cos(\theta_1 + \theta_2) + L_1 \cos\theta_1 \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 & L_2 \sin(\theta_1 + \theta_2) + L_1 \sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Suy ra:

$${}^0_Dx = L_2 \cos(\theta_1 + \theta_2) + L_1 \cos\theta_1$$

$${}^0_Dy = L_2 \sin(\theta_1 + \theta_2) + L_1 \sin\theta_1$$

$${}^0_Dz = 0$$

-Tính động học ngược của tay máy:

Từ động học thuận của tay máy

$$\begin{cases} {}^0_Dx - L_1 \cos\theta_1 = L_2 \cos(\theta_1 + \theta_2) \\ {}^0_Dy - L_1 \sin\theta_1 = L_2 \sin(\theta_1 + \theta_2) \end{cases}$$

Suy ra

$${}^0_Dx^2 + {}^0_Dy^2 + L_1^2 - 2L_1 {}^0_Dx \cos\theta_1 - 2L_1 {}^0_Dy \sin\theta_1 = L_2^2$$

$${}^0_Dx \cos\theta_1 - {}^0_Dy \sin\theta_1 = \frac{{}^0_Dx^2 + {}^0_Dy^2 + L_1^2 - L_2^2}{2L_1}$$

Đặt

$$A = {}^0_Dx^2 + {}^0_Dy^2 + L_1^2 - L_2^2$$

$$\cos\phi = \frac{{}^0_Dx}{\sqrt{{}^0_Dx^2 + {}^0_Dy^2}}, \sin\phi = \frac{{}^0_Dy}{\sqrt{{}^0_Dx^2 + {}^0_Dy^2}}$$

$$\phi = \text{atan2}({}^0_Dy, {}^0_Dx)$$

Phương trình trên trở thành:

$$\cos(\theta_1 - \phi) = \frac{A}{2L_1 \sqrt{{}^0_Dx^2 + {}^0_Dy^2}}$$

$$\sin(\theta_1 - \phi) = \pm \sqrt{\frac{4L_1^2 ({}^0_Dx^2 + {}^0_Dy^2) - A^2}{4L_1^2 ({}^0_Dx^2 + {}^0_Dy^2)}}$$

$$\text{Do đó: } \theta_1 = \text{atan2}\left(\pm \sqrt{(4L_1^2 ({}^0_Dx^2 + {}^0_Dy^2) - A^2)}, A\right) + \phi$$

$$\theta_1 = \text{atan2}\left(\pm \sqrt{(4L_1^2 ({}^0_Dx^2 + {}^0_Dy^2) - A^2)}, A\right) + \text{atan2}({}^0_Dy, {}^0_Dx)$$

Tìm θ_2 :

$$\text{Đặt: } \begin{cases} a = {}^0_Dx - L_1 \cos \theta_1 \\ b = {}^0_Dy - L_1 \sin \theta_1 \end{cases} \quad \begin{cases} k_1 = L_2 \cos \theta_1 \\ k_2 = L_2 \sin \theta_1 \end{cases}$$

Từ động học thuận tay máy, suy ra:

$$\begin{cases} a = L_2 \cos(\theta_1 + \theta_2) = k_1 \cos \theta_2 - k_2 \sin \theta_2 \\ b = L_2 \sin(\theta_1 + \theta_2) = k_2 \cos \theta_2 + k_1 \sin \theta_2 \end{cases}$$

Vì a, b, k_1, k_2 đã xác định, giải hệ phương trình trên được:

$$\begin{cases} \cos \theta_2 = \frac{ak_1 + bk_2}{k_1^2 + k_2^2} \\ \sin \theta_2 = \frac{bk_1 - ak_2}{k_1^2 + k_2^2} \end{cases}$$

Do đó,

$$\theta_2 = \text{atan2}(bk_1 - ak_2, ak_1 + bk_2)$$

Vậy với giá trị ${}^0_Dx, {}^0_Dy$ đã biết, giá trị các góc quay được xác định:

$$\begin{aligned} \theta_1 &= \text{atan2}\left(\pm \sqrt{(4L_1^2({}^0_Dx^2 + {}^0_Dy^2) - A^2)}, A\right) + \text{atan2}({}^0_Dy, {}^0_Dx) \\ \theta_2 &= \text{atan2}(bk_1 - ak_2, ak_1 + bk_2) \end{aligned}$$

Với

$$A = {}^0_Dx^2 + {}^0_Dy^2 + L_1^2 - L_2^2$$

$$\begin{cases} a = {}^0_Dx - L_1 \cos \theta_1 \\ b = {}^0_Dy - L_1 \sin \theta_1 \end{cases}$$

$$\begin{cases} k_1 = L_2 \cos \theta_1 \\ k_2 = L_2 \sin \theta_1 \end{cases}$$

6.4 Giải thuật điều khiển:

Dựa vào sơ đồ khối bộ điều khiển của robot, sơ đồ giải thuật điều khiển được chia làm 2 phần như sau:

-Phần 1: Máy tính nhận tính hiệu từ camera, xác định vị trí vật thể, đo khoảng cách, dựa vào các thông tin này để truyền lệnh điều khiển robot di chuyển, sau đó sử dụng khoảng cách đến vật đo được để tính động học ngược và truyền lệnh gập vật

-Phần 2: Vi điều khiển đảm nhận việc xử lý tín hiệu nhận được từ máy tính để điều khiển các động cơ hoạt động.

Vì máy tính và vi điều khiển giao tiếp với nhau nhờ serial port RS232, do đó dễ dàng trong việc truyền nhận, ta đặt ra một số quy định về mã lệnh được truyền nhận như sau:

- “G “: Bắt đầu bằng ký tự ‘G’ kết thúc bằng ký tự ‘ ‘. Robot di chuyển thẳng về phía trước, 2 bánh quay cùng tốc độ

- “S “: Bắt đầu bằng ký tự ‘S’ kết thúc bằng ký tự ‘ ‘. Robot đứng yên, vận tốc 2 bánh bằng 0.

- “P $\theta_1\theta_2$ “: Bắt đầu bằng ký tự ‘P’ kết thúc bằng ký tự ‘ ‘, giá trị các góc gồm 3 chữ số ví dụ: “P030120 “ nghĩa là giá trị các góc $\theta_1 = 30^\circ$, $\theta_2 = 120^\circ$. Robot đứng yên, bắt đầu gấp vật, sử dụng hai giá trị θ_1 và θ_2 để điều khiển góc quay của các động cơ RC servo.

- “R “: Bắt đầu bằng ký tự ‘R’ kết thúc bằng ký tự ‘ ‘. Robot xoay tại chỗ sang phải

- “L “: Bắt đầu bằng ký tự ‘L’ kết thúc bằng ký tự ‘ ‘. Robot xoay tại chỗ sang trái

- “D” : Đây là tín hiệu từ vi điều khiển gửi về, thông báo đã nhận được lệnh

Đầu tiên, khởi tạo các tham số cần thiết như khai báo các biến chứa dữ liệu, các ma trận... Sau đó, tiến hành Stereo calibrate hệ camera (như sơ đồ giải thuật đã trình bày ở chương 3), kết quả thu được của quá trình này là các ma trận “remap”.

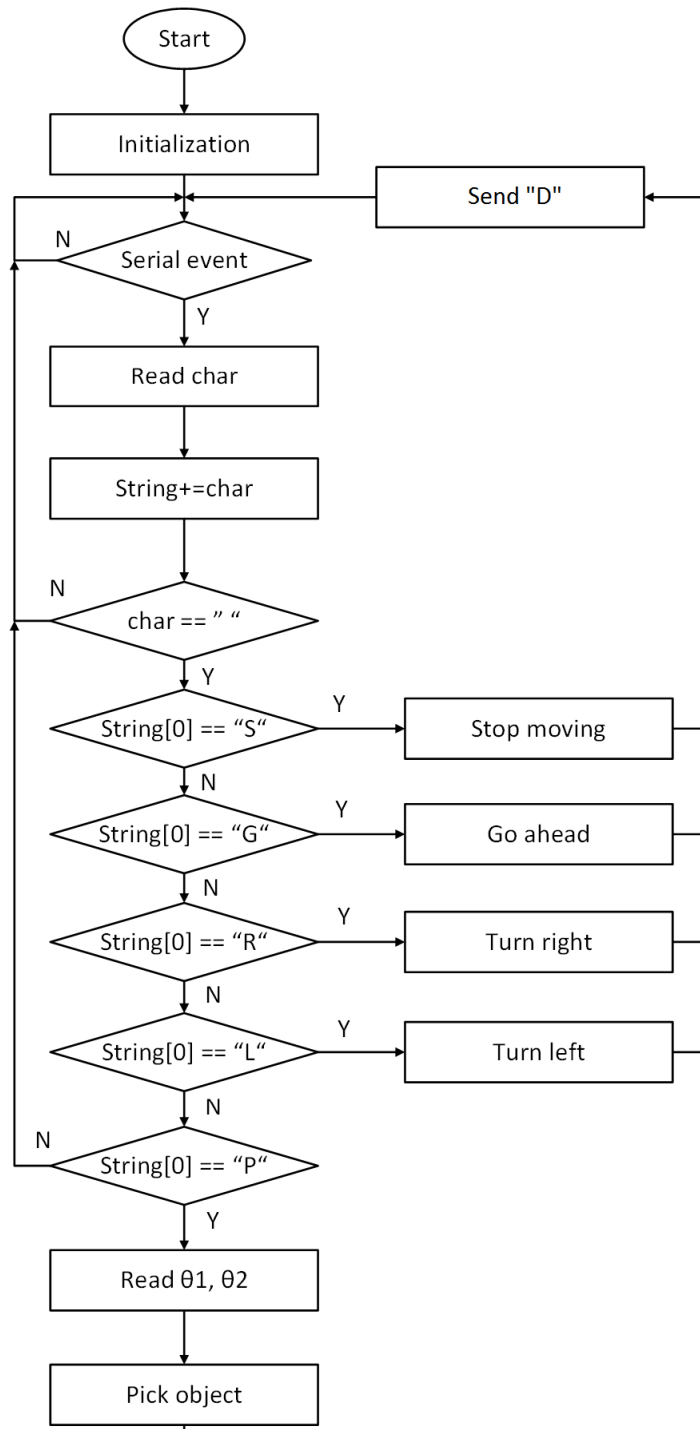
Khi người sử dụng muốn bắt đầu quá trình hoạt động của robot, nhấn “Start”. Hai ảnh từ hai camera trái và phải sẽ được đọc về (gần như cùng lúc). Tiến hành remap hai ảnh này để được các ảnh đã được undistorted và rectified.

Bước tiếp theo, tiến hành tìm vật thể trong các ảnh dựa trên đặc trưng đã đưa ra ban đầu (màu sắc).

Nếu vật không nằm trong vùng quan sát của hệ camera, cho mobile robot xoay để tìm vật thể. Khi đã phát hiện vật trong vùng quan sát, tiến hành tính khoảng cách Z đến vật và xác định vị trí của vật thể đối với mặt phẳng đối xứng của hệ camera X^{mp} . Điều khiển robot di chuyển, điều chỉnh sao X^{mp} và Z nằm trong khoảng giá trị cho phép (Với $\Delta_{X^{mp}} > 0$, $\Delta_Z > 0$ được định nghĩa trước, $Z < \Delta_Z$; $-\Delta_{X^{mp}} \leq X^{mp} \leq \Delta_{X^{mp}}$), tức là vật nằm trong vùng làm việc của tay máy. Quá trình điều khiển này là điều



-Hoạt động của vi điều khiển thực chất là giải mã các lệnh nhận được từ máy tính và thực thi các lệnh này, sau khi nhận và thực thi lệnh được nhận, vi điều khiển gửi tín hiệu về thông báo cho máy tính biết để tiếp tục xử lý. Sau đây là lưu đồ giải thuật điều khiển trên vi điều khiển



Hình 6.4: Lưu đồ giải thuật điều khiển trên vi điều khiển

6.5 Kết luận:

Như vậy, đến đây chương này đã hoàn thành việc xây dựng giải thuật điều khiển, tính toán động học mobile, động học ngược tay máy.

Mobile robot hoạt động như sau:

Trước hết, mobile robot sẽ tiến hành di chuyển tìm vật sao cho vật thể xuất hiện trên ảnh của cả camera trái và phải. Sau đó, khoảng cách Z và giá trị X^{mp} được tính toán và sử dụng để điều khiển mobile robot di chuyển đến vị trí mong muốn so với vật. Quá trình điều khiển này là điều khiển vòng kín. Hệ camera được sử dụng như một cảm biến của robot. Các thông tin tính toán được từ hệ camera (các giá trị X^{mp} và Z) được phản hồi về để điều khiển mobile robot di chuyển.

Khi xe đã di chuyển đến vị trí mong muốn so với vật, tiếp tục sử dụng CV để đo khoảng cách đến vật và sử dụng giá trị này để tính động học ngược tay máy. Giá trị các góc quay của các khớp của tay máy được truyền xuống cho vi điều khiển, tiến hành gấp vật.

Chương tiếp theo ta tiến hành thực nghiệm, lập trình điều khiển dựa trên giải thuật và tính toán đã trình bày ở trên.

Chương 7: Thực nghiệm và đánh giá kết quả

7.1 Giới thiệu:

Sau khi đã hoàn thành thiết kế cơ khí, mạch điện và giải thuật điều khiển cho mobile robot. Ở chương này sẽ trình bày quá trình thực nghiệm và đánh giá kết quả hoạt động của robot.

Các bước thực nghiệm bao gồm các hoạt động sau:

-Đo khoảng cách đến Z vật, tính X^{mp} sử dụng CV:

Để đo khoảng cách cần tiến hành thiết lập hệ camera, calibrate cho từng camera, tiến hành stereo Calibration, nhận dạng khối vật màu, tính toán disparity và khoảng cách.

Tính X^{mp} , xác định vị trí của vật so với mặt phẳng đối xứng của hệ camera và truyền nhận lệnh tương ứng giữa máy tính và vi điều khiển.

-Áp dụng động học ngược tay máy, điều khiển gấp vật.

Các công việc cần thực hiện gồm: tính động học ngược của tay máy trên máy tính, truyền nhận tín hiệu giữa máy tính và vi điều khiển.

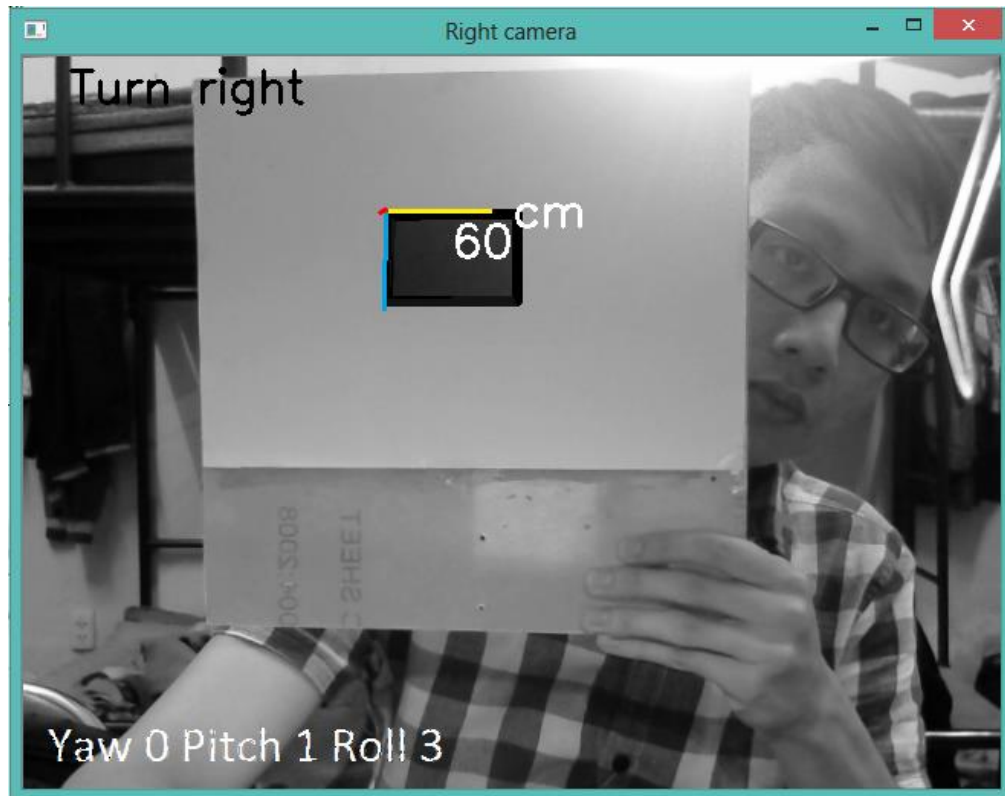
-Điều khiển robot di chuyển và tay máy gấp vật

7.2 Đo khoảng cách sử dụng OpenCV:

-Đầu tiên, cần tiến hành lắp đặt hệ camera. Như đã trình bày, hai camera cần được lắp đặt sao cho các mặt phẳng ảnh đạt frontal-parallel và row-aligned với nhau tối đa có thể.

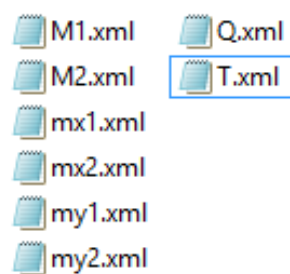
Để thực hiện điều này, ngoài dựa vào thiết kế có thể điều chỉnh được của hệ camera, ta sử dụng thêm công cụ CvFindExtrinsics của thư viện OpenCV để thực hiện điều chỉnh.

Hàm CvFindExtrinsics cho kết quả là các ma trận R và vector T mô tả hướng, vị trí tương đối giữa hệ tọa độ camera và hệ tọa độ thực tế. Do đó, ta có thể sử dụng một chessboard như là 1 mặt phẳng chuẩn để điều chỉnh (bằng tay) từng camera trong hệ sao cho đạt song song với mặt phẳng chuẩn này.



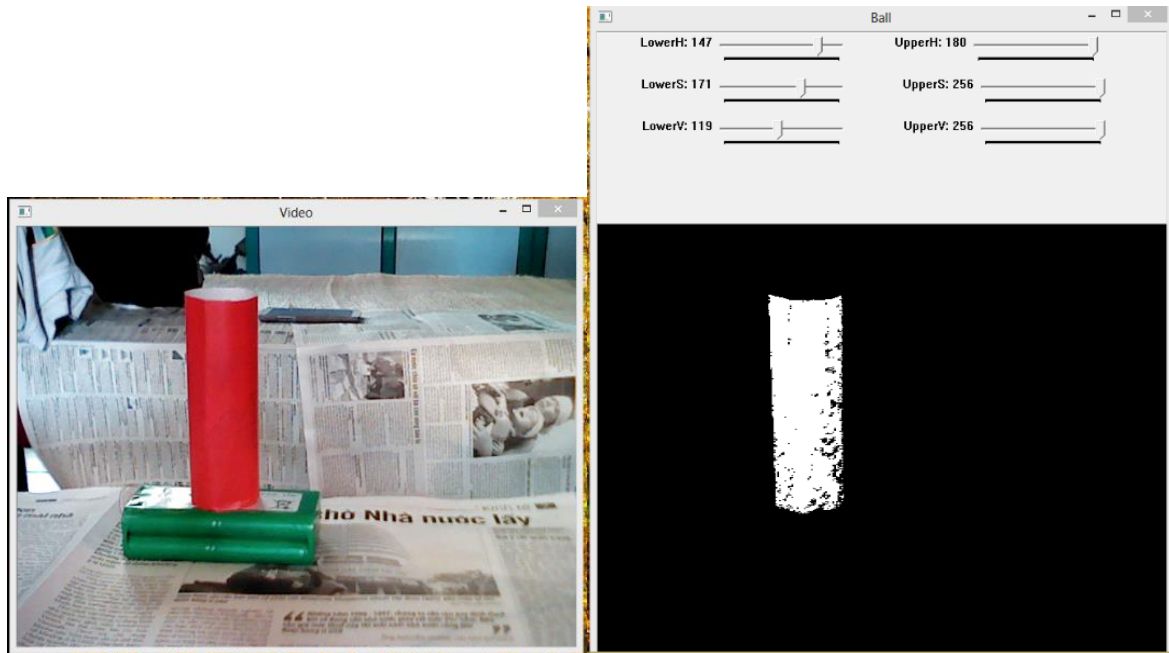
Hình 7.1: Sử dụng mặt phẳng chuẩn để cân chỉnh hệ camera bằng tay

-Bước tiếp theo, ta tiến hành calib từng camera và calib hệ stereo camera. Bước này đã được trình bày rõ ở chương 3. Vì quá trình calib thường tốn thời gian nên các ma trận remap và thông tin của hệ camera được lưu sẵn dưới dạng file *.xml để tiện sử dụng khi điều khiển robot.



Hình 7.2: Các file chứa thông tin thu được sau quá trình calib

-Nhận dạng khối vật màu: Khối vật được sử dụng là khối trụ màu đỏ. Tiến hành xác định khoảng phân ngưỡng HSV để phân ngưỡng ảnh thu được.



Hình 7.3: Giao diện chương trình tìm khoảng phân ngưỡng- Hình ảnh thu được trước và sau khi phân ngưỡng

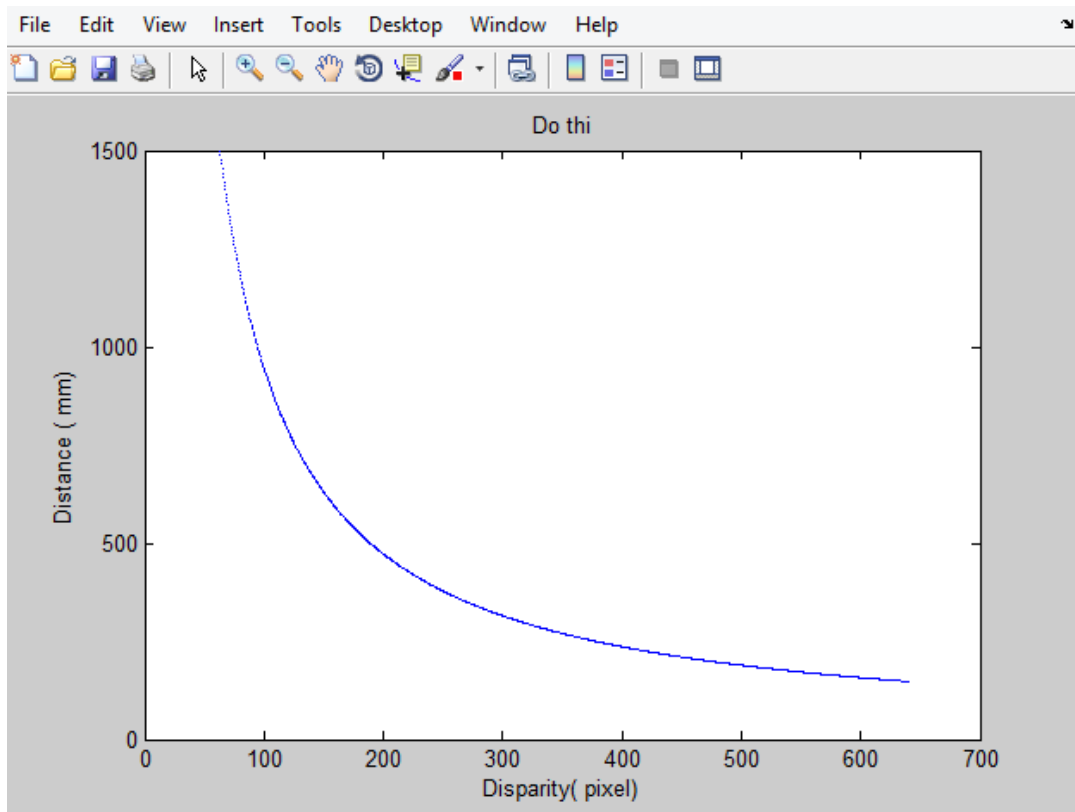
Kết quả thu được: khoảng phân ngưỡng để nhận diện khối trụ màu đỏ như sau:

$$H \in [147,180] , S \in [171,255], V \in [118,255]$$

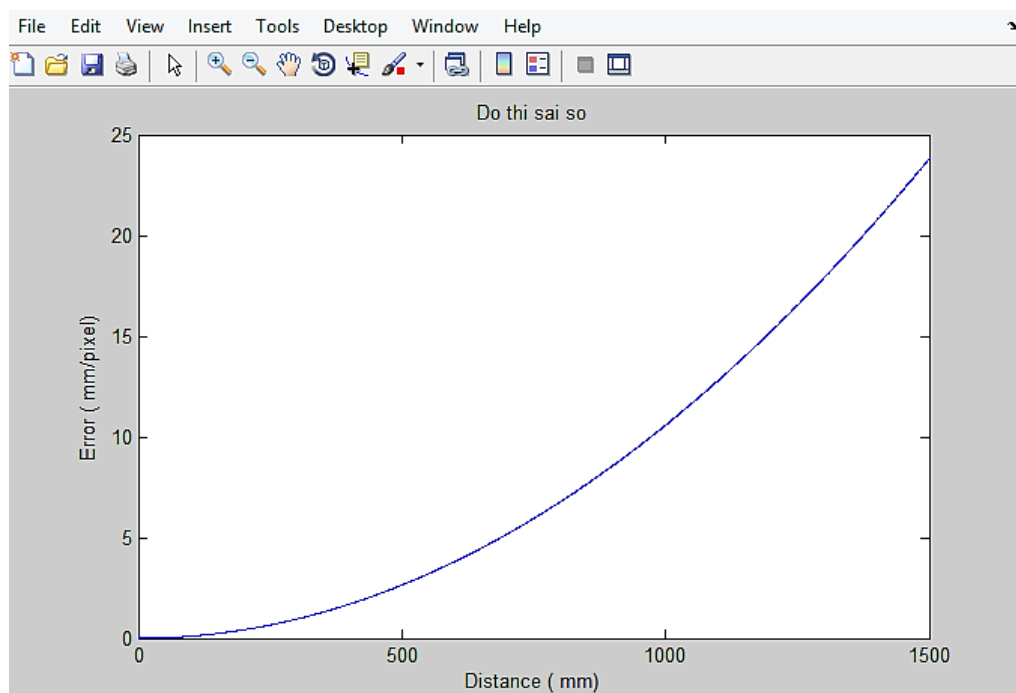
-Tính toán disparity và khoảng cách:

Qua thực nghiệm. khoảng cách giữa hai camera được chọn là 105 mm. (Cho vùng nhìn thấy của hai camera đủ rộng và đảm bảo độ chính xác của phép đo khoảng cách).

Để cân chỉnh khoảng cách T giữa hai camera đạt 105 mm, sử dụng stereo calibration để calib, tìm các thông số của hệ camera. Như đã trình bày, khoảng cách T chính là giá trị T_x của ma trận Translation (có được sau khi thực hiện stereo calibration).



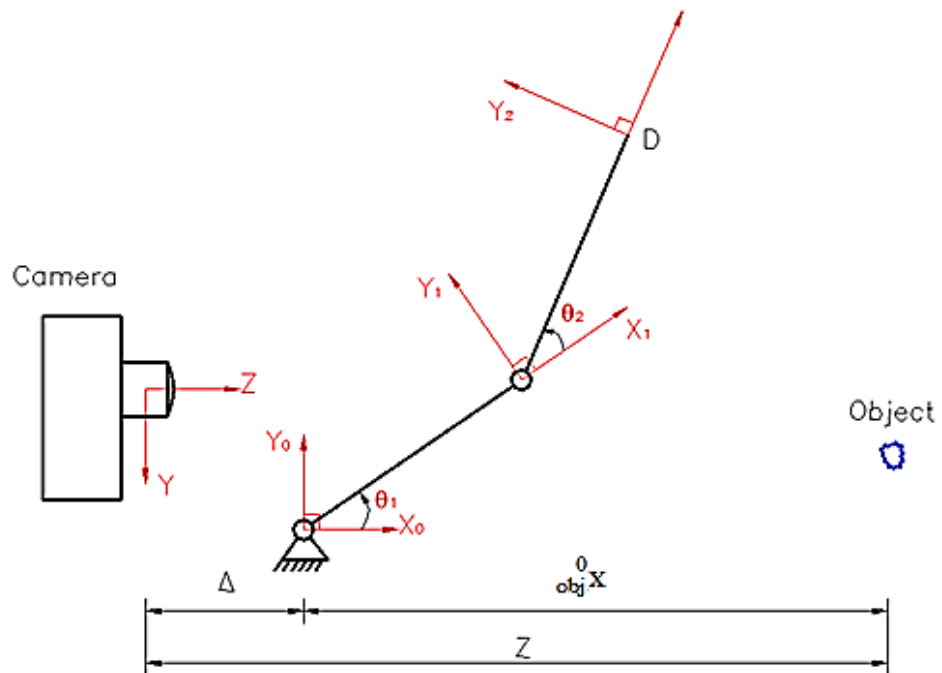
Hình 7.4: Khoảng cách Z và disparity quan hệ tỉ lệ nghịch ($T = 105 \text{ mm}, f = 900$)



Hình 7.5: Độ chính xác của hệ camera (mm/pixel) với $T = 105 \text{ mm}, f = 900$

Ngoài ra, hệ tọa độ camera là hệ tọa độ ảo (không biết chính xác vị trí gốc tọa độ). Khoảng cách Z đo được là từ vật đến tâm phép chiếu. Do đó, ta chuyển gốc tọa độ này về gốc tọa độ khác (vật lý) đã xác định để đảm bảo độ chính xác.

Vì tay máy sử dụng khoảng cách Z để tính động học ngược và gắp vật nên ta chọn gốc tọa độ 0 của tay máy để thuận tiện cho việc tính toán.



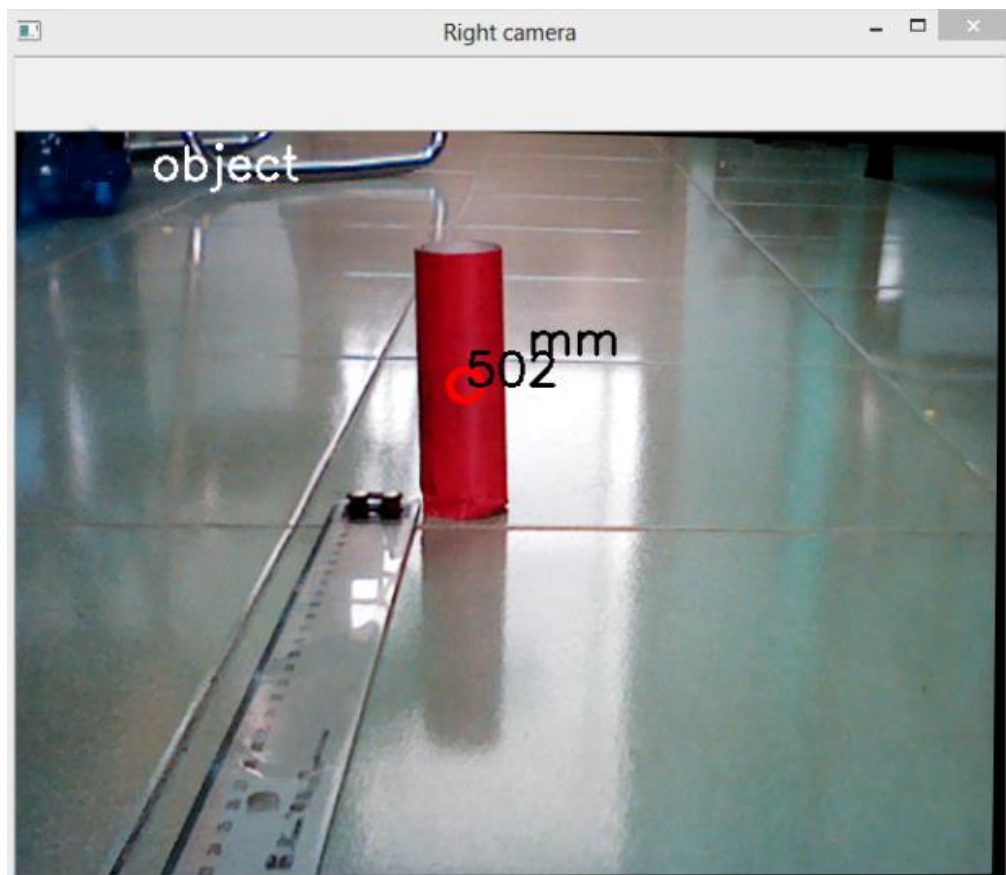
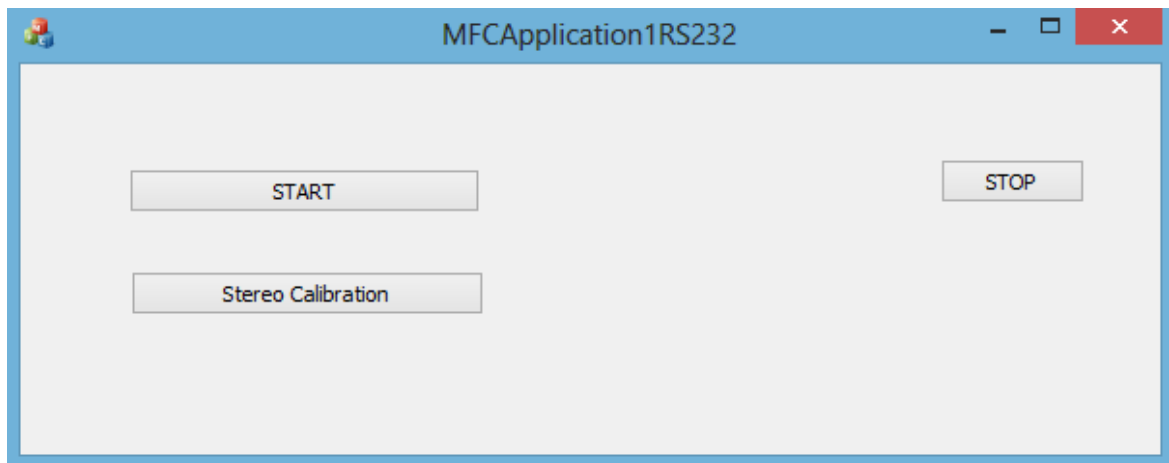
Hình 7.6: Hệ tọa độ tay máy, hệ camera

Lúc này, khoảng cách từ vật đến gốc tọa độ 0 của tay máy là ${}_{obj}^0x = Z - \Delta$

Để xác định Δ ta tiến hành như sau:

Đo khoảng cách từ vật đến camera sử dụng CV, đo khoảng cách từ vật đến gốc tọa độ tay máy (sử dụng dụng cụ đo). Lúc này $\Delta = Z - {}_{obj}^0x$

-Chương trình điều khiển trên máy tính được viết trên nền C++.



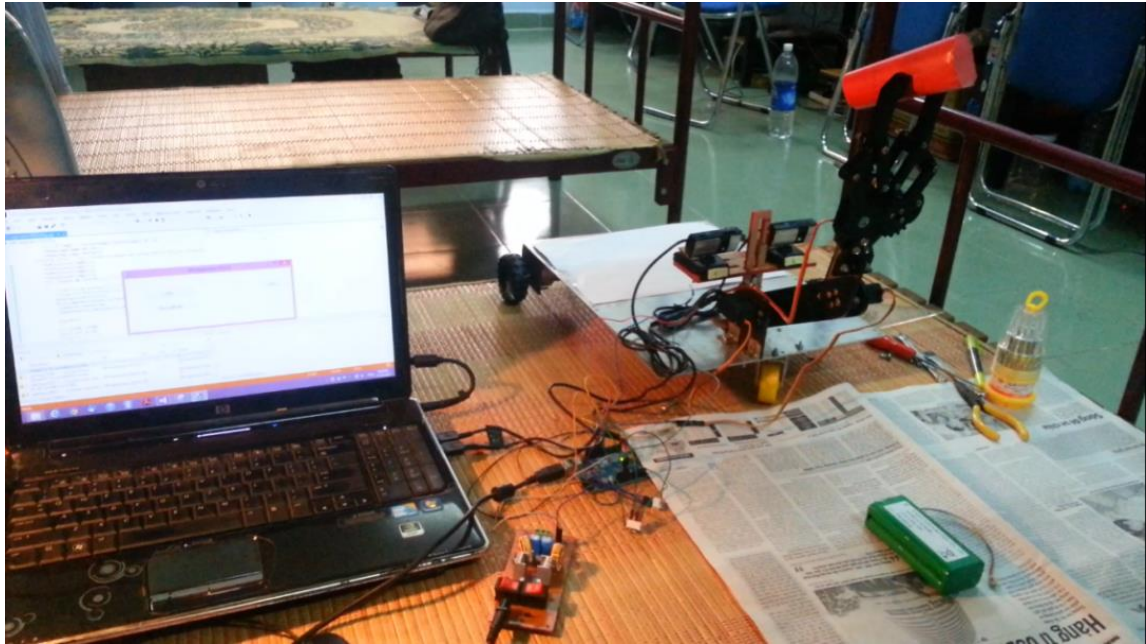
Hình 7.7: Thực nghiệm đo khoảng cách sử dụng CV

7.3 Điều khiển tay máy:

Động cơ RC được sử dụng có góc quay từ $0^{\circ} - 180^{\circ}$, do đó tay máy cần được lắp đặt sao cho phù hợp với hệ tọa độ tay máy đã được chọn.

Tiến hành thực nghiệm đo khoảng cách từ tay máy đến vật và tính động học ngược tay máy. Giá trị các góc quay được truyền qua serial port RS232.

Vì vật nằm trên mặt phẳng không đổi nên tọa độ y của vật thể được giữ không đổi.



Hình 7.8: Thực nghiệm điều khiển tay máy gấp vật thể

7.4 Điều khiển mobile robot di chuyển và gấp vật:

Sau khi đã tiến hành thực nghiệm đo khoảng cách, gấp vật, ta tiến hành điều khiển mobile robot di chuyển và gấp vật theo giải thuật đã đưa ra.



Hình 7.9 Mobile robot di chuyển và gấp vật thể

7.5 Kết luận:

Qua thực nghiệm, đã tiến hành điều khiển robot thành công. Mobile robot có thể tìm vật thể cần gấp, xác định vị trí của vật (sử dụng hệ 2 camera để xác định khoảng cách đến vật, xác định hướng di chuyển của robot), truyền tín hiệu để điều khiển robot di chuyển và gấp vật. Như vậy, luận văn đã hoàn thành yêu cầu và mục tiêu ban đầu đã đề ra.

Qua thực nghiệm cũng đã chứng minh tính đúng đắn của lý thuyết và phương pháp đo khoảng cách sử dụng CV, giải thuật điều khiển robot mà luận văn đã thiết kế.

Chương 8: Tổng kết và hướng phát triển đề tài

8.1 Đánh giá kết quả đạt được:

Luận văn đã hoàn thành được các nhiệm vụ sau:

- Tìm hiểu tổng quan về mobile robot gắn khối vật sử dụng camera
- Thiết kế mô hình mobile robot có gắn tay máy 2 DOFs
- Tìm hiểu OpenCV và lý thuyết điều khiển robot sử dụng 2 camera.
- Thiết kế mạch điều khiển và chương trình điều khiển
- Thực nghiệm kiểm nghiệm

Ngoài những nhiệm vụ đã được giải quyết ở trên, luận văn còn phát triển được phương pháp cân chỉnh (bằng tay) hệ camera, giúp quá trình đo đạc được chính xác hơn. Kết quả của phương pháp này còn có thể được áp dụng để xác định toàn hoàn toàn hướng và vị trí của vật thể trong không gian, giúp robot hoạt động tốt hơn.

Đề tài đã được chọn để tham dự **“The 2014 International Capstone Design Contest on Renewable Energy Technology” (CORE 2014)**, tổ chức từ ngày 08-01-2014 đến ngày 09-01-2014 tại Hàn Quốc.

Một số kết quả nghiên cứu được của luận văn đã được sử dụng để thực hiện bài báo cáo **“Gaze Estimation with a Single Camera based on an ARM-based embedded Linux platform”** tại hội nghị “International Symposium on Mechatronics and Robotics” ngày 10-12-2013

Tuy nhiên, luận văn vẫn còn một số hạn chế sau:

- Robot chỉ gắn vật trụ màu, chưa thể gắn vật có hình dạng phức tạp khác.
- Robot chỉ hoạt động trong môi trường không có (hoặc có rất ít) nhiễu- là các vật thể khác có cùng màu với khối vật cần gắn.
- Gia công cơ khí chưa đạt độ chính xác cao nên gây một số khó khăn cho việc điều khiển.

8.2 Hướng phát triển của đề tài:

Đề tài chỉ mới dừng lại ở việc xác định khoảng cách, vị trí của vật sử dụng CV và điều khiển mobile robot di chuyển gặp khối vật đơn giản. Tuy nhiên, kết quả đạt được của đề tài có thể được sử dụng làm nền tảng cho rất nhiều các ứng dụng và nghiên cứu khác.

Để đo khoảng cách các khối vật khác, phức tạp hơn, cần nâng cấp giải thuật nhận diện vật thể như sử dụng Haar-like hoặc mạng Nơ-ron để huấn luyện nhận diện vật thể.

Áp dụng giải thuật xác định hướng và vị trí tương đối giữa hệ tọa độ camera và hệ tọa độ vật như khi calib bằng tay hệ camera, hướng của vật thể hoàn toàn có thể được xác định. Khi đó, mobile robot có thể di chuyển và gặp vật thể bất kỳ, nằm ở vị trí bất kỳ trong vùng làm việc xung quanh robot đã được định nghĩa từ trước.

Giải thuật xác định khoảng cách và hướng của vật thể có thể được ứng dụng ở nhiều nơi, nhiều lĩnh vực như trong giao thông công cộng, trong các khu vực giám sát, các loại robot dân dụng, giải trí, robot công nghiệp, quân sự...

TÀI LIỆU THAM KHẢO

- [1] Gary Bradski, Adrian Kaehler (2008). *Learning OpenCV*, O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472
- [2] Richard Hartley, Andrew Zisserman (2004). *Multiple View Geometry in Computer Vision 2nd edition*, Cambridge university press, Cambridge
- [3] Robert Laganière (2011). *OpenCV 2 Computer Vision Application Programming Cookbook*, Packt Publishing Ltd., 32 Lincoln Road, Olton, Birmingham, B27 6PA, UK
- [4] Michael Margolis (2011). *Arduino Cookbook*, O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472
- [5] Reza N.Jazar (2010). *Theory of Applied Robotics-Kinematics, Dynamics, and Control 2nd*, RMIT University, Melbourne, Victoria, Australia, ISBN 978-1-4419-1749-2
- [6] PGS.TS Nguyễn Hữu Lộc (2011). *Cơ sở thiết kế máy*, NXB Đại học quốc gia TP Hồ Chí Minh
- [7] PGS.TS Nguyễn Hữu Lộc (2011). *Bài tập Chi tiết máy*, NXB Đại học quốc gia TP Hồ Chí Minh
- [8] PGS.TS Trần Hữu Quế (2006). *Vẽ kỹ thuật cơ khí (tập 1)*, NXB Giáo Dục