**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA ĐIỆN – ĐIỆN TỬ

**BỘ MÔN ĐIỆN TỬ**

---------------o0o---------------

****

**ĐỀ CƯƠNG LUẬN VĂN TỐT NGHIỆP**

**THIẾT BỊ HỖ TRỢ CHO XE GẮN MÁY**

**GVHD: TS. Nguyễn Lý Thiên Trường**

**SVTH: Trương Văn Thịnh**

**MSSV: 1713335**

**TP. HỒ CHÍ MINH, THÁNG 1 NĂM 2020**

***LỜI CẢM ƠN***

Đầu tiên, em gửi lời cảm ơn chân thành nhất đến giáo viên hướng dẫn, thầy **Nguyễn Lý Thiên Trường**, người đã tận tình trực tiếp hướng dẫn em hoàn thành đề cương luận văn tốt nghiệp và sẽ hướng dẫn luận văn tốt nghiệp. Nhờ những sự chỉ bảo, định hướng, góp ý tận tình của thầy, em mới có thể hoàn thành được đề cương luận văn tốt nghiệp.

Tiếp theo, em xin gửi lời cảm ơn các cán bộ giảng viên Khoa điện-điện và trường Đại học Bách Khoa Thành Phố Hồ Chí Minh, vì đã không những truyền dạy học tập giá trị và còn những kinh nghiệm sống, những câu chuyện vô cùng quý báu trong thời gian học tập tại trường để em thành càng hoàn thiện về kiến thức và con người trong.

Và Em xin chân thành cảm ơn đến bạn bè, anh chị trong bộ môn điện tử và trường Đại học Bách Khoa Thành Phố Hồ Chí Minh trong thời gian qua đã cùng chia sẽ kiến thức, giúp đỡ nhau trong học tập để tất cả cùng tiến lên và hoàn thành đề tài.

Cuối cùng, con xin cảm ơn ba me và gia đình đã đồng hành và ủng hộ con trong suốt quá trình học tập từ bé đến ngày hôm nay.

Trong quá trình thực hiện đề tài, những thiếu sót, hạn chế trong kiến thức và kỹ năng,em đã cố gắng trong khả năng của mình nên sẽ không tránh khỏi những sai sót. Em rất mong nhận được sự đóng góp, ý kiến từ bạn bè và thầy cô để em có thể hoàn thiện đề tài trong thời gian sắp tới.

*Tp. Hồ Chí Minh, ngày 14 tháng 1 năm 2020 .*

**Sinh viên**

**Trương Văn Thịnh**

**MỤC LỤC**

[1. GIỚI THIỆU 1](#_Toc61358290)

[1.1 Tổng quan 1](#_Toc61358291)

[1.2 Tình hình nghiên cứu trong và ngoài nước 2](#_Toc61358292)

[1.3 Mục tiêu đề tài 4](#_Toc61358293)

[2. NỘI DUNG ĐỀ TÀI 4](#_Toc61358294)

[3. GIẢI PHÁP THỰC HIỆN 5](#_Toc61358295)

[3.1 LÝ THUYẾT : 5](#_Toc61358296)

[3.1.1 Xử lý ảnh và Deep learning: 5](#_Toc61358297)

[3.1.2 Python: 6](#_Toc61358298)

[3.1.3 Hệ điều hành Raspian : 7](#_Toc61358299)

[3.1.4 OpenCV: 8](#_Toc61358300)

[3.1.5 TensorFlow : 9](#_Toc61358301)

[3.1.6 Keras : 9](#_Toc61358302)

[3.1.7 Convolution neural networks: 9](#_Toc61358303)

[3.1.8 Tổng quan về vi xử lí ARM Cortex M3 : 15](#_Toc61358304)

[3.1.9 Ngôn ngữ C trong lập trình ARM: 17](#_Toc61358305)

[3.1.10 Chuẩn giao tiếp UART: 17](#_Toc61358306)

[3.1.11 Chuẩn giao tiếp I2C : 18](#_Toc61358307)

[3.1.12 Hiệu ứng Hall : 19](#_Toc61358308)

[3.2 PHẦN CỨNG: 19](#_Toc61358309)

[3.2.1 Raspberry Pi 4 Model B: 19](#_Toc61358310)

[3.2.2 STM32F103C8T6 21](#_Toc61358311)

[3.2.3 Cảm biến từ trường Hall 23](#_Toc61358312)

[3.2.4 Module điều khiển động cơ L298 23](#_Toc61358313)

[3.2.5 LCD 16x2 25](#_Toc61358314)

[3.2.6 Module I2C-LCD 27](#_Toc61358315)

[3.2.7 Động cơ DC giảm tốc GA25 Encoder : 28](#_Toc61358316)

[3.2.8 Camera pi v1 28](#_Toc61358317)

[3.3 THIẾT KẾ: 29](#_Toc61358318)

[3.3.1 PHẦN CỨNG 29](#_Toc61358319)

[3.3.2 PHẦN MỀM 33](#_Toc61358320)

[4. DỰ KIẾN KẾT QUẢ ĐẠT ĐƯỢC 39](#_Toc61358321)

[4.1 Kết quả sơ khởi đã đạt được : 39](#_Toc61358322)

[4.2 Kết quả dự kiến đạt được 42](#_Toc61358323)

[5. KẾ HOẠCH THỰC HIỆN: 42](#_Toc61358324)

[6. TÀI LIỆU THAM KHẢO 43](#_Toc61358325)

DANH SÁCH HÌNH

[Hình 1‑1 Xe máy lưu thông trên đường 1](#_Toc61370202)

[Hình 1‑2 Minh họa cho bài toán nhận dạng vật thể 2](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370203)

[Hình 1‑3 Hệ thống trí tuệ nhân tạo trên ô tô 3](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370204)

[Hình 1‑4 Hệ thống hỗ trợ thông minh cho xe máy của BOSCH 5](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370205)

[Hình 3‑1 Tổng quan về các lĩnh vực trong trí tuệ nhân tạo 7](#_Toc61370206)

[Hình 3‑2 Sự khác biệt giữa Machine Learning và Deep Learning 8](#_Toc61370207)

[Hình 3‑3 Mô hình cơ bản của một mạng neural 11](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370208)

[Hình 3‑4 Những thành phần cơ bản của mạng neural tích chập 13](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370209)

[Hình 3‑5 Minh họa cho kết quả sử dụng maxpooling với filer 2x2 14](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370210)

[Hình 3‑6 Minh họa cho đâu ra sau khi qua lớp fully connected 15](#_Toc61370211)

[Hình 3‑7 Activation Function 15](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370212)

[Hình 3‑8 Minh họa cho những trường hợp của model sau khi huấn luyện xong 16](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370213)

[Hình 3‑9 Minh họa sau khi mạng neural được áp dụng Drop out 17](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370214)

[Hình 3‑10 Cấu trúc vi xử lý ARM Cortex M3 18](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370215)

[Hình 3‑11 Cấu trúc lỗi ARM Cortex M3 19](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370216)

[Hình 3‑12 Minh họa 2 thiết bị giao tiếp UART 20](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370217)

[Hình 3‑13 Minh họa các thiết bị giao tiếp I2C 21](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370218)

[Hình 3‑14 Hướng giao tiếp của giao thức I2C 22](#_Toc61370219)

[Hình 3‑15 Dòng điện chạy trong thanh Hall 22](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370220)

[Hình 3‑16 Xuất hiện điện áp 2 bên thanh hall khi ta đặt từ trường vào 23](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370221)

[Hình 3‑17 Kit Raspberry Pi 4B+ 24](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370222)

[Hình 3‑18 Sơ đồ GPIO Raspberry Pi 26](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370223)

[Hình 3‑19 Kit STM32F103C8T6 Blue Pill 27](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370224)

[Hình 3‑20 Module cảm biến từ trường hall 29](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370225)

[Hình 3‑21 Sơ đồ khối của cảm biến hall 29](#_Toc61370226)

[Hình 3‑22 Dạng song đầu ra của chân Analog và Digital trên cảm biến Hall 30](#_Toc61370227)

[Hình 3‑23 Module L298 30](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370228)

[Hình 3‑24 Mạch cầu H 31](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370229)

[Hình 3‑25 Module LCD-I2C 34](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370230)

[Hình 3‑26 LCD được kết nối module lcd-i2c 35](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370231)

[Hình 3‑27 Động cơ encoder G25 36](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370232)

[Hình 3‑28 Camera pi V1.3 37](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370233)

[Hình 3‑29 Sơ đồ khối hệ thống 38](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370234)

[Hình 3‑30 STM32F103 Skematic 39](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370235)

[Hình 3‑31 Chương trình chính 41](#_Toc61370236)

[Hình 3‑32 Các mẫu biển báo giao thông sử dụng 43](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370237)

[Hình 3‑33 Lưu đồ nhận dạng ảnh 44](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370238)

[Hình 3‑34 Lưu đồ huấn luyện dữ liệu 44](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370239)

[Hình 3‑35 Lưu đồ chế độ hiển thị 45](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370240)

[Hình 3‑36 Lưu đồ ngắt ngoài khi nhắn nút 45](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370241)

[Hình 3‑37 Lưu đồ ngắt ngoài khi có từ trường đặt lên cảm biến 46](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370242)

[Hình 3‑38 Lưu đồ ngắt timer 46](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370243)

[Hình 4‑1 Độ chính xác của model trong quá trình huấn luyện 47](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370244)

[Hình 4‑2 Sai số của model trong quá trình huấn luyện 47](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370245)

[Hình 4‑3 Kết quả minh họa biển báo 20km/h 48](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370246)

[Hình 4‑4 Kết quả minh họa biển báo 100km/h 48](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370247)

[Hình 4‑5 Kết quả minh họa biển báo “chỉ đi thẳng” 49](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370248)

[Hình 4‑6 Kết quả minh họa biển báo 20km/h 49](file:///C:\Users\Thinh\Desktop\DE%20CUONG_CHINH%20THUC.docx#_Toc61370249)

**DANH SÁCH BẢNG**

[Bảng 3‑1 Thông số kĩ thuật Raspberry Pi 4 25](#_Toc61370751)

[Bảng 3‑2 Cấu hình STM32F103C8T6 28](#_Toc61370752)

[Bảng 3‑3 Bảng giá trị hoạt động của mạch cầu H 31](#_Toc61370753)

[Bảng 3‑4 Thông số kĩ thuật module L298 32](#_Toc61370754)

[Bảng 3‑5 Thông số kĩ thuật LCD1602 33](#_Toc61370755)

[Bảng 3‑6 Chức năng các chân trên LCD 34](#_Toc61370756)

[Bảng 3‑7 Thông số kỹ thuật module i2c-lcd 35](#_Toc61370757)

[Bảng 3‑8 Sơ đồ chân module i2c-lcd 35](#_Toc61370758)

[Bảng 3‑9 Thông số kĩ thuật động cơ encoder G25 36](#_Toc61370759)

[Bảng 3‑10 Thông số kĩ thuật camera pi v1.3 37](#_Toc61370760)

[Bảng 3‑11 Cấu hình Convolution Neural Network để huấn luyện 42](#_Toc61370761)

[Bảng 3‑12 Thứ tự biển báo 43](#_Toc61370762)

[Bảng 5‑1 Kế hoạch luận văn 51](#_Toc61370763)

# GIỚI THIỆU

## Tổng quan

Xe máy là phương tiện di chuyển chủ yếu của người Việt, chiếm hơn 85% tổng số phương tiện. Tính đến cuối năm 2011, Việt Nam có khoảng 33,4 triệu xe máy đang lưu thông, theo thống kê của Bộ Giao thông Vận tải. Đến cuối năm 2019, con số này đã gần chạm ngưỡng 60 triệu xe. Hiện tại, Việt Nam đang xếp thứ 4 trong danh sách các quốc gia tiêu thụ xe máy nhiều nhất thế giới, đứng sau Ấn Độ, Trung Quốc và Indonesia.

Đối với một nước có mức thu nhập khá thấp so với thế giới. Đối với người Việt, xe máy không chỉ là phương tiện để đi lại mà còn là phương tiện để kiếm sống.

Theo thống kê của các nhà sản xuất xe máy ở Việt Nam, cứ mỗi ngày lại có thêm khoảng 9000 xe máy lăn bánh ra đường. Do nhu cầu quá lớn, lưu thông bằng xe máy là một trong những nguyên lớn nhất gây ảnh hưởng đến tình trạng giao thông và đặc biệt là gây ra tai nạn giao thông.

Hình ‑ Xe máy lưu thông trên đường

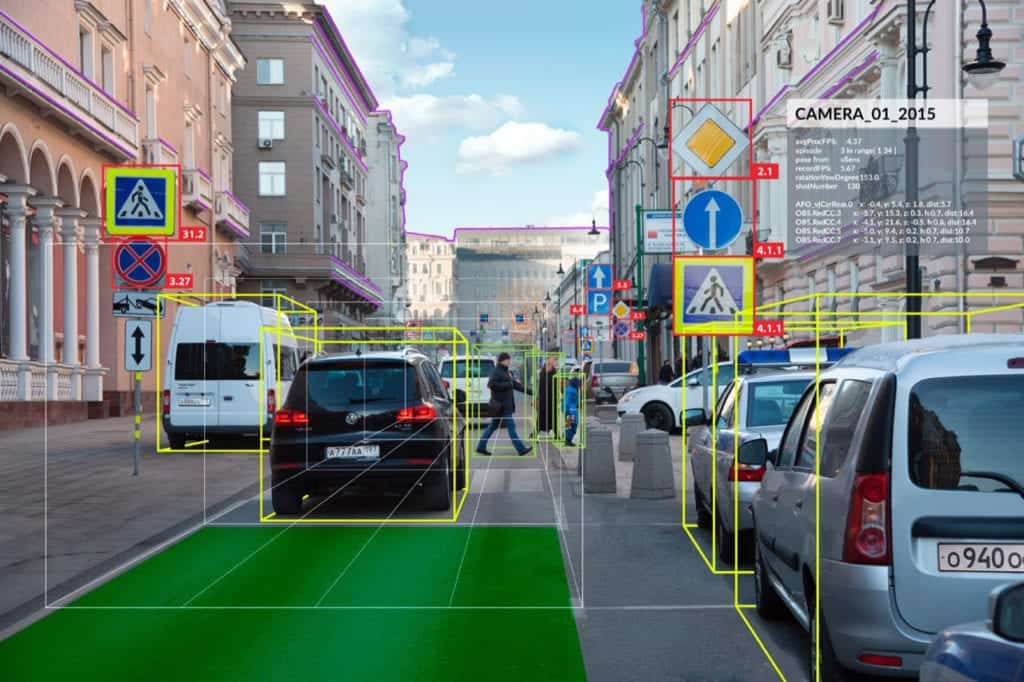
Theo Ủy ban An Toàn Giao Thông quốc gia - nêu con số thống kê trong 5 năm (tính từ ngày 15-10-2015 đến 14-10-2020), cả nước xảy ra 94.024 vụ tai nạn giao thông, làm chết 39.917 người, bị thương 77.477 người. Trong đó Xe máy, gồm cả xe máy điện, là phương thức có tỷ lệ hành khách bị tử vong do tai nạn giao thông cao nhất. Theo thống kê của Ủy ban An toàn giao thông (ATGT) Quốc gia, xe máy liên quan đến 70% tổng số vụ an toàn giao thông. Trong đó, các vi phạm do người điều khiển xe máy gây ra tới 70-80% các vụ tai nạn, các lỗi vi phạm chủ yếu : phóng nhanh ( 15,3% ), vượt ẩu ( 13,4% ), chạy quá gần ( 9,2 % ), sử dụng chất kích thích, sử dụng điện thoại di động trong lúc lái xe (chiếm 4,3% )…

Trong đó có nhiều nguyên nhân dẫn đến tai nạn như cơ sở hạ tần giao thông nghèo nàn, quản lí giao thông còn yếu kém trong khi lưu lượng giao thông dày đặc..Tuy nhiên nguyên nhân chủ yếu và khách quan nhất vẫn là ý thức của người tham gia giao thông.

Ngoài ra, hiện nay các phương tiện giao thông tại Việt Nam chưa được chú trọng trang bị những thiết bị hỗ trợ tài xế lái xe thông minh, hạn chế những pha va chạm không đáng có xảy ra và đưa ra những cảnh báo khi người điều khiển thi hành sai . Đối với những nước phát triển thì việc lắp đặt các thiết bị hoặc thiết kế những tính năng thông minh trên xe đã được áp dụng và được coi trọng.

Bùng nổ xe máy cũng mang đến nhiều hệ lụy. Ùn tắc giao thông ở các thành phố lớn, tỷ lệ tai nạn giao thông gia tăng và còn ảnh hưởng đến thiệt hại kinh tế.

Dựa trên tình hình thực tế như trên, Đề tài “ Thiết bị hỗ trợ lái xe máy “ ra đời để hỗ trợ và đưa ra những cảnh báo cho người tham gia giao thông giúp trách những tai nạn không đáng có và cải thiện tình hình giao thông ở Nước ta.

 Với tốc độ phát triển mạnh mẽ của ngành trí tuệ nhân tạo ( Artificial Intelligence ) đã thúc đẩy cho sự phát triển của các thuật toán thông minh và tự động trong các thiết bị thực tế và đời sống .Và một trong các phía cạnh đó là bài toán nhận dạng vật thể ( Object detection ) . Có nhiều thuật toán và phương pháp ra đời để phụ vụ cho việc phát hiện đối tượng hiện đại và chính xác cao như R-CNN, Fast-RCNN, Faster-RCNN, RetinaNet và nhanh hơn nhưng rất chính xác như SSD và YOLO..

Hình ‑ Minh họa cho bài toán nhận dạng vật thể

Trong đề tài sẽ sử dụng các thuật toán trí tuệ nhân tạo để nhận dạng được các biển báo giao thông kết hợp với khả năng điều khiển ngoại vi của các dòng vi điều khiển hiện tại để tạo ra một thiết bị có chức năng đưa ra những cảnh báo và chỉ dẫn cho người điều khiển về tốc độ và hướng đi của phương tiện. Ngoài ra, hệ thống sẽ có thêm một camera phía sau cảnh báo xe lớn để tránh những tai nạn không lường trước.

## Tình hình nghiên cứu trong và ngoài nước

Tình hình nghiêm cứu trong nước :

-Với sự phát triển ngày càng mạnh mẽ của thuật toán Deep learning về bài toán nhận dạng vật thể đã thúc đẩy cho việc nghiêm cứu và ứng dụng thuật toán này vào trong rất nhiều lĩnh vực như giải trí, kinh tế…và giao thông cũng không ngoại lệ. Nhiều đề tài nghiên cứu khoa học đã thực hiện các nhận dạng biển báo giao thông, đèn giao thông, biển số xe, làn đường bằng các thuật toán như R-CNN, Fast-CNN, Yolo, SSD…Đó là tiền đề để xuất hiện nhiều hơn và cải thiện tốt hơn ứng dụng vào thực tế.

-Ngày nay, khi nền công nghiêp ô tô đang ở thời kì đỉnh cao. những công nghệ thông minh ngày càng phát triển hơn nữa. Ô tô ngày nay có rất nhiều hiệu năng và hệ thống thông minh như nhận dạng biển báo giao thông , cảnh báo vật cản, tự vận hành…Tuy nhiên, đối với xe máy dừng như lại bị tụt lại phía sao và các ứng dụng này khá nghèo nàn, thậm chí không được đề xuất do kinh tế và ý thức sử dụng của người dân.

Hình ‑ Hệ thống trí tuệ nhân tạo trên ô tô

-Tập đoàn Vingroup đã thành lập công ty tránh nhiệm hữu hạn sản xuất và kinh doanh VinFast vào 2007, chuyên về lĩnh vực ô tô và máy xe của Việt Nam. VinFast đã đẩy mạnh tốc độ về sản xuất và các hiệu năng an toàn trên ô tô. Còn về xe máy, năm 2018 VinFast đã cho ra mắt mẫu xe máy điện thông minh Klara, sản phẩm có nhiều tính năng thông minh như hệ thống điều khiển tốc độ của xe, chống trộm..Tuy sản phẩm có nhiều cải tiến so với các xe mặt bằng chung nhưng khía cạnh cảnh báo và ứng dụng trí tuệ nhân tạo để đảm bảo an toàn cho người lái xe vẫn chưa xuất hiện.

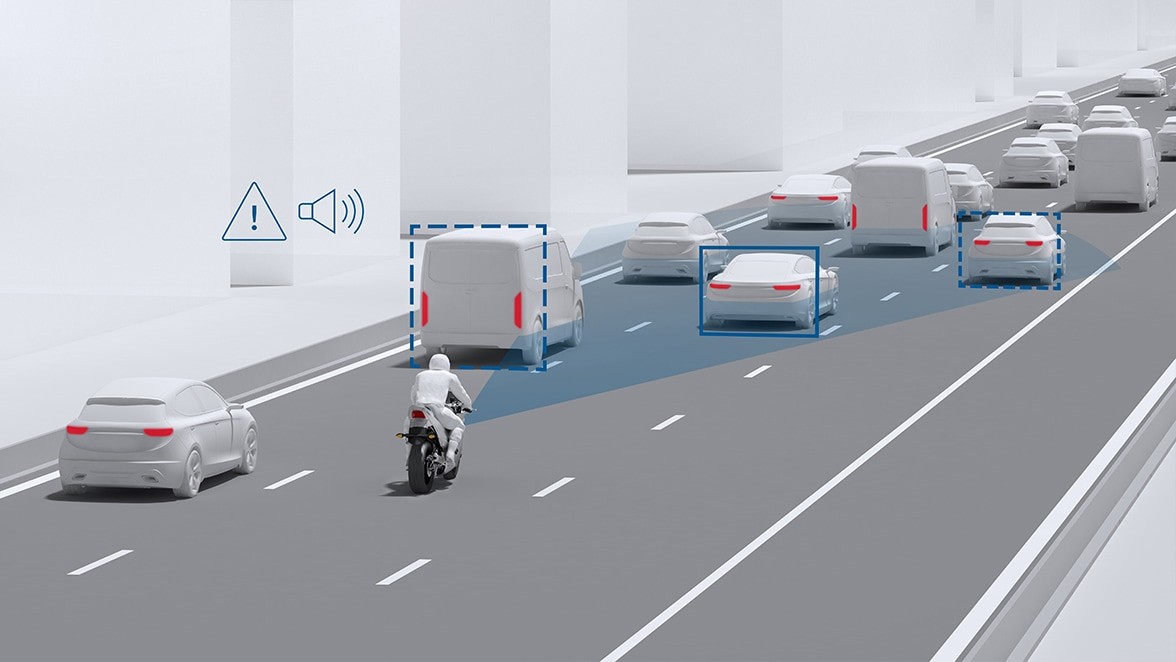
-Trong nước ta, hằng năm luôn tổ chức các công trình nghiên cứu về hành vi gây tai nạn giao thông và các giải pháp giúp tăng sự an toàn cho người tham gia giao thông.

-Đối với xe máy,ở nước ta cũng đã xuất hiện các ứng dụng, sản phẩm để an toàn cho người điều khiển xe máy như ứng dụng MotorSafe trên hệ điều hành Android giúp cài đăc sẵn tốc độ mặc định và định vị GPS …Hằng năm, trong các cuộc thi sáng tạo về khoa học kĩ thuật cũng đã xuất hiện nhiều đề tài về thiết bị hỗ trợ cho người đi xe máy như đo nồng độ cồn, cảnh báo ngủ gật…nhưng vẫn chưa được áp dụng vào thức tế vì tính hiệu quả chưa cao.

-Những thiết bị hỗ trợ dành cho xe máy tương đối hạn chế và chỉ hầu hết là những phụ kiện như bộ định vị chống trộm, khóa thông minh, đèn trợ sang, thiết bị cảnh báo nồng độ cồn..Vẫn chưa có một thiết bị an toàn kết hợp trí tuệ nhân tạo đúng nghĩa nào trên thị trường.

Tình hình nghiêm cứu ngoài nước :

-Công nghệ giao thông kết hợp với trí tuệ nhân tạo đã trở nên rất phổ biến và ngày càng phát triển ở nước ngoài, đặc biệt là các nước công nghệ cao như Mỹ, Đức, Nhật…Đặc biệt là hãng xe công nghệ nổi tiếng nhất hiện nay Tesla. Trong các năm gần đây, Tesla đã đưa ra nhiều mẫu ô tô với các tính năng thông minh và đang đẩy mạnh việc hệ thống xe tự hành kết hợp trí tuệ nhân tạo cho việc nhận diện biển báo, làn đường…Điều này làm cơ sở cho việc sử dụng trí tuệ nhân tạo vào các phương tiện giao thông không chỉ riêng cho ô tô.

-Nhiều công ty công nghê trên thế giới đã cho ra mắt các hệ thống thông minh trên xe máy như tự giữa thăng bằng khi có sự cố, cảnh báo an toàn khu có tay lái có chuyển động bất thường và đưa ra phản hồi rất nhanh cho người dùng…Nhưng phần là các hệ thống được tích hợp sẵn trên xe.

Hình ‑ Hệ thống hỗ trợ thông minh cho xe máy của BOSCH

-Các thiết bị ứng dụng trí tuệ nhân tạo kết hợp với các cảm biến radar cũng đang được nghiên cứu và phát triển.

## Mục tiêu đề tài

Mục tiêu 1 : Tìm hiểu kiến thức về xử lý ảnh, deep learning, các thuật toán nhận dạng như CNN,YOLO,SSD...Từ đó áp dụng cho việc nhận dạng các biển báo giao thông bằng các thuật toán học sâu. Lựa chọn thuật toán tối ưu, có thể chạy trên các thiết bị có cấu hình thấp như raspberry pi.

Mục tiêu 2 : Phán đoán được chuyển động bất thường của các xe tải, ô tô lớn phía sau bằng việc sử dụng các phương pháp xử lý ảnh, trí tuệ nhân tạo thông qua thu hình camera kết hợp cảm biến hồng ngoại, radar…để đưa ra dự đoán.

Mục tiêu 3 : Đo vận tốc của xe máy bằng phương pháp áp dụng hiệu ứng hall thông qua cảm biến từ trường hall. Hiển thị tốc độ, quảng đường, thời gian..

Mục tiêu 4 : Xây dựng một hệ thống phần cứng để kết hợp điều khiển các ngoại vi ,hiển thị chỉ dẫn, cảnh báo. Mô hình gọn nhỏ, có thể lắp ráp trên xe máy.

Mục tiêu 5 : Xây dựng giải thuật tối ưu hệ thống, hạn chế độ trễ tối đa.

# NỘI DUNG ĐỀ TÀI

**Nội dung 1**: Tìm hiểu nguyên lý, lý thuyết về …:

-Ngôn ngữ Python trong lập trình xử lý ảnh và các ứng dụng deep learning.

-Thư viện OpenCV, Keras, TensorFlow để xây dựng mô hình deep learning nhận dạng vật thể.

-Các thuật toán nhận dạng như R-CNN, YOLO...

-Tìm hiểu về cấu trúc, tính năng của vi điều khiển ARM.

-Ngôn ngữ C/C++ trong việc lập trình vi điều khiển ARM.

-Các chuẩn giao tiếp UART, I2C.

-Ứng dụng hiệu ứng Hall cho việc đo tốc độ động cơ.

-Hệ điều hành thời gian thực (RTOS).

**Nội dung 2**: Tìm hiểu về cảm biến, vi xử lý, bo mạch phát triển, …

-Máy tính nhúng Raspberry Pi 4, hệ điều hành raspbian.

-Kit phát triển stm32f103c8t6.

-Cảm biến từ trường hall.

-Cảm biến siêu âm, radar.

**Nội dung 3:** Thiết kế bộ điều khiển … sử dụng giải thuật …

-Thiết kế bộ điều khiển động cơ.

-Bộ điều khiển trung tâm giao tiếp với Raspberry và các ngoại vi như LCD, nút nhấn, loa, led, cảm biến.

-Sử dụng giải thuật sử dụng theo luồng dữ liệu và hợp với chương trình ngắt.

# GIẢI PHÁP THỰC HIỆN

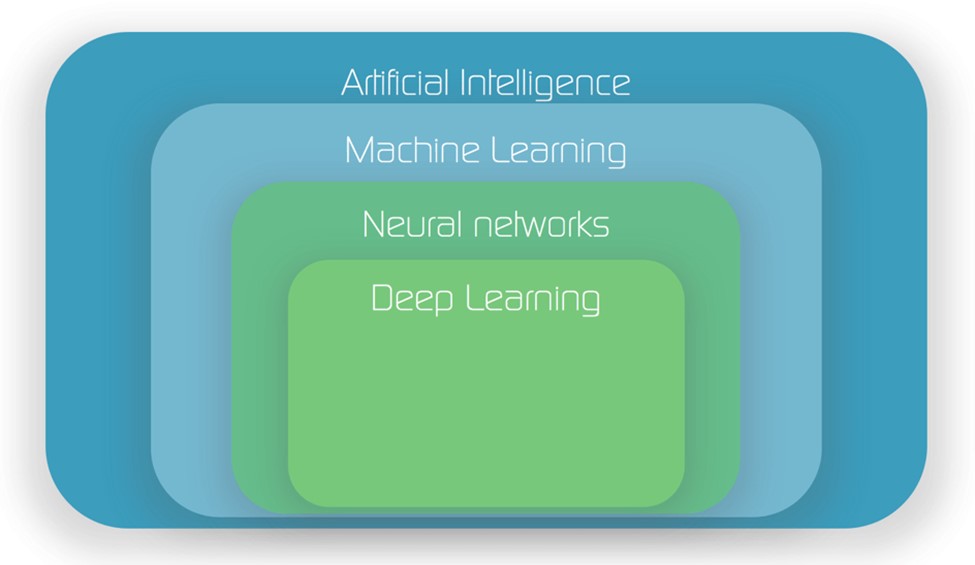
## LÝ THUYẾT :

### Xử lý ảnh và Deep learning:

-Xử lí ảnh là một dạng xử lý số tín hiệu với tín hiệu xử lý đầu vào là ảnh và đầu ra là ảnh đã được xử lý. Đây là một ngành khoa học mới rất phát triển trong những năm gần đây. Xử lí ảnh bao gồm 4 lĩnh vực chính: xử lý nâng cao chất lượng ảnh, nhận dạng ảnh, nén ảnh và truy vấn ảnh.Sự phát triển của xử lý ảnh đem lại rất nhiều lợi ích cho con người. Ngày nay xử lý ảnh đã áp dụng rất rộng rãi trong đời sống như photoshop, nén ảnh, nhận dạng vật thể…

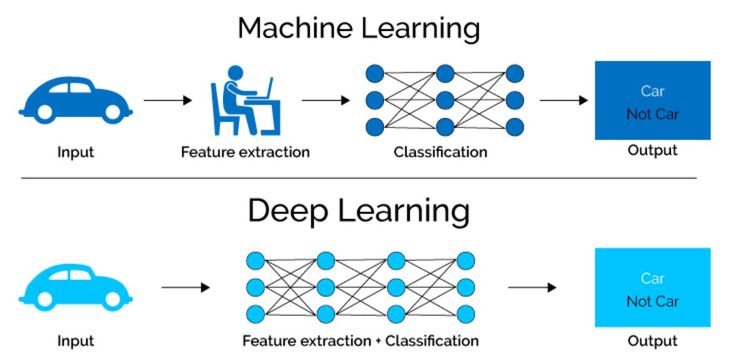
-Ảnh là tập hợp hữu hạn các điểm ảnh với mức xám phù hợp, Pixel ( điểm ảnh ) là đơn vị cơ bản của mỗi bức ảnh, càng nhiều pixel thì bức ảnh càng chi tiết..Gồm 3 loại ảnh : Ảnh màu RGB với 3 kênh màu mỗi kênh màu 8bit, ảnh xám hay còn gọi là ảnh đơn sắc với 1 kênh màu 8bit, ảnh nhị phân là ảnh ảnh được biểu diễn chỉ bằng 2 giá trị trắng và đen.

-Deep learning là một tập hợp con của Machine Learning xuất phát từ thuật toán neural network nằm trong lĩnh vực trí tuệ nhân tạo sử dụng mạng lưới thần kinh với nhiều lớp lấy cảm hứng từ neural thần kinh của con người , một kĩ thuật máy học dạy cho thiết bị hoạt động một cách logic. Nó liên quan đến việc đi sâu vào một số lớp của mạng, bao gồm cả một lớp ẩn để trích xuất ra những thông tin. Thông qua Deep learning, thiết bị có thể nhận biết các họa tiết để có thể phân loại chúng thành các loại khác nhau.



Hình ‑ Tổng quan về các lĩnh vực trong trí tuệ nhân tạo

- So với Machine learning phải thông qua các bước lập trình để thực hiện nhận diện thì Deep learning nhận dạng vật thể một cách tự động dựa vào các phương pháp lặp để dạy máy móc bắt chước trí thông minh của con người. Một mạng lưới thần kinh nhân tạo thực hiện phương pháp lặp này thông qua một số cấp độ phân cấp. Giống như con người học hỏi từ kinh nghiệm, một thuật toán Deep Learning có thể thực hiện một nhiệm vụ nhiều lần, mỗi lần điều chỉnh nó để cải thiện kết quả.



Hình ‑ Sự khác biệt giữa Machine Learning và Deep Learning

-Ngày nay, khi hệ thống cơ sở dữ liệu ngày càng phát triển đã tạo tiền đề mạnh mẽ để giúp Deeplearning được phát triển và ứng dụng nhiều trong đời sống.

### Python:

-Python là ngôn ngữ lập trình bậc cao cho các mục đích lập trình đa năng, do Guido Van Rossum thiết kế vào năm 1980 và lần đầu ra mắt vào năm 1999. Python dễ dàng tìm hiểu và nổi lên như một trong những ngôn ngữ lập trình nhập môn tốt nhất cho người lần đầu tiếp xúc với ngôn ngữ lập trình. Python hoàn toàn tạo kiểu động và sử dụng cơ chế cấp phát bộ nhớ tự động. Python có cấu trúc dữ liệu cấp cao mạnh mẽ và cách tiếp cận đơn giản nhưng hiệu quả đối với lập trình hướng đối tượng. Cú pháp lệnh của Python là điểm cộng vô cùng lớn vì sự rõ ràng, dễ hiểu và cách gõ linh động làm cho nó nhanh chóng trở thành một ngôn ngữ lý tưởng để viết script và phát triển ứng dụng trong nhiều lĩnh vực, ở hầu hết các nền tảng.

-Ban đầu python được phát triển chạy trên nền tảng Unix . Nhưng theo thời gian Python mở rộng sang hầu hết các hệ điều hành : Windows, Linux, Mac OS…

-Python cung cấp rất nhiều tín năng nổi bật : miễn phí, mã nguồn mã, khả năng mở rộng và có thể nhúng với các ngôn ngữ lập trình khác.

-Hiện nay, Python là một trong những ngôn ngữ lập trình phổ biến nhất thế giới. Với tính ứng dụng cực kì cao có khắp trong lĩnh vực nghiêm cứu kĩ thuật : lập trình web, data analysis, lập trình robot, lập trình game, thị giác máy tính, bảo mật mạng và máy tính…

-Trong lĩnh Deeplearning, Python hỗ trợ có nhiều thư viện hữu ít như Keras, TensorFlow, Scikit-learn…

### Hệ điều hành Raspian :

-Raspberry Pi có rất nhiều hệ điều hành hỗ trợ, trong đó có Raspbian là hệ điều hành chính thức của Raspberry Pi Foundation, ngoài ra có 7 hệ điều hành khác (Ubuntu Mate, Snappy Core Ubuntu, Window 10 Iot Core, Osmc, OpenElec. Pinet, RiscOS) được xác nhận hỗ trợ và kha khá hệ điều hành do các nhà phát triển tự tối ưu.

-Raspbian là một hệ điều hành cơ bản dễ sử dụng, phổ biến nhất và do chính Raspberry Pi Foundation cung cấp. Hơn thế nữa, Raspbian là phiên bản không chính thức của Debian Wheezy – một phiên bản Linux có tiếng. Raspbian rất đơn giản và quen thuộc. Nó cũng được hãng khuyến cáo sử dụng, nhất là cho người mới bắt đầu làm quen với RPI. Raspbian có dung lượng sau khi giải nén là khoảng gần 4GB, cần tối thiểu 1 cái thẻ 4GB để có thể sử dụng Raspbian.

-Raspbian hướng tới người dung có mục đích :

+Sử dụng Raspberry Pi như máy tính văn phòng để lướt web, soạn văn bản, check mail và thi thoảng nghe nhạc/xem phim.

+Nghiên cứu phát triển các thiết bị điều khiển tự động.

+Sử dụng như một máy chủ cung cấp các dịch vụ như web, file server, printer server...

+Raspbian hoạt động rất ổn định, tốc độ nhanh (đặc biệt là trên Raspberry Pi 3). Tốc độ khởi động thử nghiệm Raspbian với Raspberry Pi 3, sử dụng thẻ Toshiba Exceria 48MB/s thì chỉ 7s. Nhược điểm của nó là giao diện đơn giản, cổ điển và không hào nhoáng.

### OpenCV:

-OpenCV được bắt đầu vào năm 1999 tại Intel bởi Gary Bradsky và phát hành vào năm 2000. OpenCV  là một thư viện mã nguồn mở cho các ứng dụng xử lý ảnh, máy học ,thị giác máy tính và các tính năng tăng tốc GPU trong hoạt động thời gian thực.

-Hiện tại, OpenCV hỗ trợ rất nhiều thực toán liên quan đến Thị giác máy tính và machine learning. OpenCV hỗ trợ nhiều ngôn ngữ lập như C++,Python…có sẵn trên các nền tảng như Windows, Linux, OS X, Androi, IOS…

-OpenCV được thiết kế để để tính toán hiệu quả và với sự tập trung nhiều vào các ứng dụng thời gian thực. Được viết bằng tối ưu hóa C/C++, thư viện có thể tận dụng lợi thế của xử lý đa lõi. Được sử dụng trên khắp thế giới, OpenCV có cộng đồng hơn 47 nghìn người dùng và số lượng download vượt quá 6 triệu lần. Phạm vi sử dụng từ nghệ thuật tương tác, cho đến lĩnh vực khai thác mỏ, bản đồ trên web hoặc công nghệ robot.

-Những tính năng nổi bật của OpenCV là khả năng xử lý, phân tích, hiển thị hình ảnh, video. Hỗ trợ nhiều thuật toán trong việc nhận dạng, trích xuất đặc trưng của đối tượng, có tính năng CUDA acceleration hay con được biết đến là GPU.

### TensorFlow :

-TensorFlow là một mã nguồn mở và được sử dụng phổ biến trong lĩnh vực trí tuệ nhân tạo. TensorFlow được tạo ra bởi Google với mục đích phục vụ cho việc nghiên cứu và học tập trong lĩnh vực trí tuệ nhân tạo.

-TensorFlow cung cấp khả năng xử lí tính toán số học dựa trên biểu đồ mô tả sự thay đổi dữ liệu, trong đó các nút là các phép toán học, các cạnh biểu thị luồng dữ liệu.

### Keras :

-Keras là một giao diện lập trình ứng dụng được viết bằng ngôn ngữ python, một thư viện mở cho mạng neural trong Deeplearning . Keras có thể sử dụng chung với các thư viện Deeplearning nổi tiếng như TensorFlow.

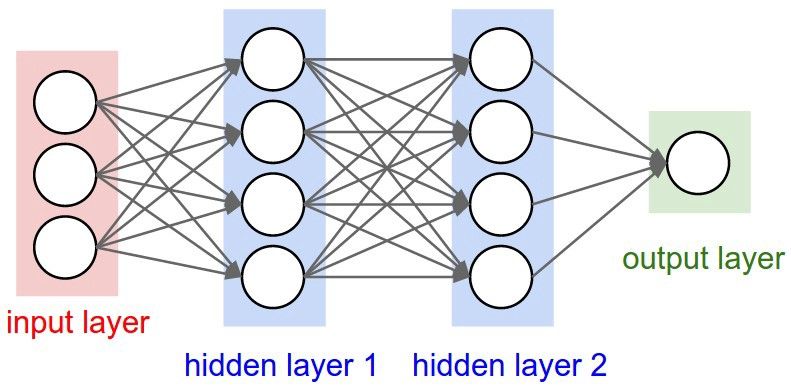
-Keras có nhiều ưu điểm là dễ sử dụng trong việc xây dựng mô hình, xử lí dữ liệu trong mạng neural tích chập (CNN),, có thể chạy trên cả CPU và GPU…

### Convolution neural networks:

**Giới thiệu về Neral networks :**

-Neural là tính từ của neuron (nơ-ron), network chỉ cấu trúc đồ thị nên neural network (NN) là một hệ thống tính toán lấy cảm hứng từ sự hoạt động của các nơ-ron trong hệ thần kinh, có khả năng cảm ứng, phát sinh xung thần kinh và dẫn truyền xung điện này. Một cách ngắn gọn nhất thì Neural Network là mô hình toán học mô phỏng nơ-ron trong hệ thống thần kinh con người. Tuy nhiên, Neural Network chỉ là lấy cảm hứng từ não bộ và cách thức hoạt động, chứ không phải bắt chước toàn bộ các chức năng của nó. Việc chính của chúng ta là dùng mô hình đấy đi giải quyết bài toán chúng ta cần.

-Neural Network mô phỏng theo cách thức hoạt động của các nơ-ron, là cấu thành chính yếu của Deep Learning. Neural Network bao gồm các lớp (layer) chứa một nhóm các nút tròn (node) đại diện cho nơ-ron ở não người. Các node ở layer sau sẽ liên kết với tất cả các node ở layer liền trước đó còn gọi để trao đổi dữ liệu.



Hình ‑ Mô hình cơ bản của một mạng neural

-Layer đầu tiên là input layer nhận các dữ liệu đầu vào, các layer ở giữa được gọi là hidden layer (lớp ẩn) thực hiện tính toán cho các đầu vào, layer cuối cùng được gọi là output layer với mục đích xuất kết quả đầu ra. Mỗi mô hình luôn có 1 input layer, 1 output layer, có thể có hoặc không các hidden layer. Tổng số layer trong mô hình được quy ước là (số layer - 1) bởi không tính input layer. Lấy ví dụ trực tiếp từ mô hình trên, nó có tất cả là 4 layer, bao gồm 1 input layer, 2 hidden layer và 1 output layer. Tuy nhiên, ta không xét input layer nên chỉ coi số lượng layer của mô hình này là 3.

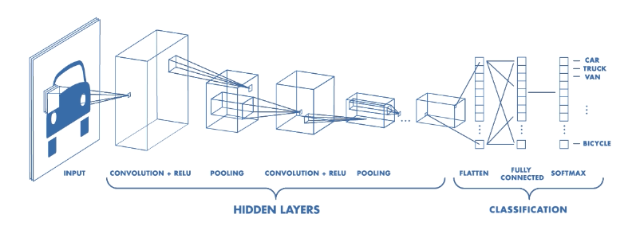
-Đối với Deep Learning, thử thách lớn nhất trong việc tạo Neural Network là quyết định số lượng các hidden layer, cũng như số các node cho mỗi layer. Từ "Deep" ở đây ý muốn nhận mạnh Deep Learning là phải có độ sâu qua nhiều hơn một hidden layer. Nói đơn giản hơn, ta có thể hiểu Deep Learning chính là Neural Network với nhiều **hidden layer.** Như vậy, mỗi hidden layer sẽ có một nhiệm vụ, và output của layer này sẽ là input của layer sau, đầu ra cuối cùng sẽ là output layer.

**Convolution neural networks ( mạng nơ-ron tích chập )**

-Được phát triển từ các mạng neural thông thường ban đầu. Tuy nhiện mạng neural tích chập được cải tiến rất nhiều so với các mạng neural thông thường và được áp dụng vào nhiều ứng dụng phức tạm hơn mà mạng neural thông thường không làm được.

-Đặc điểm khác biệt của mạng Convolution neural so với mạng neural thông thường là mỗi neural sẽ chỉ liên kết một phần với nhỏ của input ( local connectivity ) và mỗi neural trong cùng một layer sẽ có cùng một trọng số ( shared weights ) đều này làm sẽ kích thước các trọng số và khối lượng tính toán trong mạng đi rất nhiều mà độ chính xác cũng cao hơn.

-Hiện nay, Convolution neural networks ( CNN )là một trong những mô hình deeplearning phổ biến nhất và có ảnh hưởng nhất trong cộng đồng. CNN được sử dụng trong nhiều bài toán phân loại, nhận dạng ảnh.

**Cấu tạo của một mạng Convolution neural networks :**

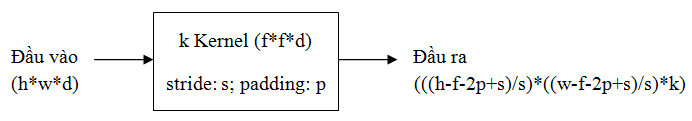
Hình ‑ Những thành phần cơ bản của mạng neural tích chập

**-**Convolution layer :

**+)**Có thể nói đây là một lớp cực cì quan trong CNN, bởi ở lớp này sẽ thực hiện mọi phép tính toán. Lớp này có chức năng chính là phát hiện đặt trưng. Trong phần này có 4 đối tượng chính : ma trận đầu vào, bộ filters, và receptive field, feature map. Convolution layer nhận đầu vào là một ma trận 3 chiều và một bộ filters cần phải học.

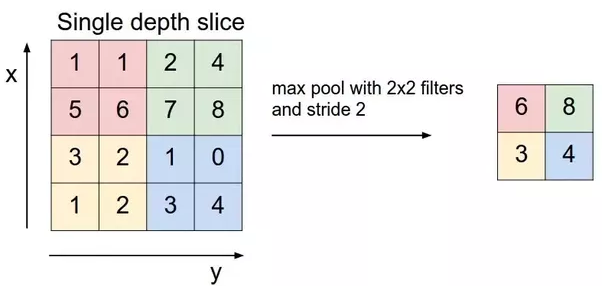
+)Nguyên lý : Bộ filters sẽ trượt qua từng vị trí trên bức ảnh để tính tích chập (convolution) giữa bộ filter và phần tương ứng trên bức ảnh. Phần tưng ứng này trên bức ảnh gọi là receptive field, tức là vùng mà một neuron có thể nhìn thấy để đưa ra quyết định, và mà trận cho ra bới quá trình này được gọi là feature map.

+) Một cách tổng quát, nếu đầu vào được cho là một tensor có kích thước (h\*w\*d) thì Kernel sẽ có kích thước (f\*f\*d), với f là một số lẻ, chọn Stride = s và Padding = p. Ở đây, Kernel có cùng độ sâu với đầu vào (cùng bằng d) nên phép tính Convolution cũng được thực hiện tương tự như đối với ma trận. Mỗi Kernel sẽ cho ra kết quả là một ma trận (((h-f-2p+s)/s)\*((w-f-2p+s)/s)) thì nếu dùng k Kernel để học được nhiều đặc trưng thì kích thước của kết quả đầu ra sẽ là (((h-f-2p+s)/s)\*((w-f-2p+s)/s)\*k).



-Pooling layer :

+)Pooling layer thường được dùng giữa các convolutional layer, để giảm kích thước dữ liệu nhưng vẫn giữ được các thuộc tính quan trọng. Kích thước dữ liệu giảm giúp giảm việc tính toán trong model.

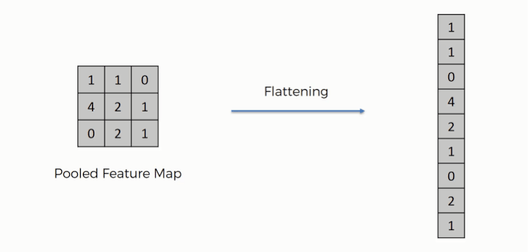
+)Pooling gồm các loại : Max pooling , Average pooling, Sum pooling.

Hình ‑ Minh họa cho kết quả sử dụng maxpooling với filer 2x2

+)Ý tương đằng sau tầng pooling là vị trí tuyết đối của những đặc trưng trong không gian ảnh không còn cần cần thiết, thay vào đó vị trí tương đối giữ các đặc trưng đã đủ để phân loại đối tượng. Hơn giảm tầng pooling có khả năng giảm chiều cực kì nhiều, làm hạn chế overfit, và giảm thời gian huấn luyện tốt.

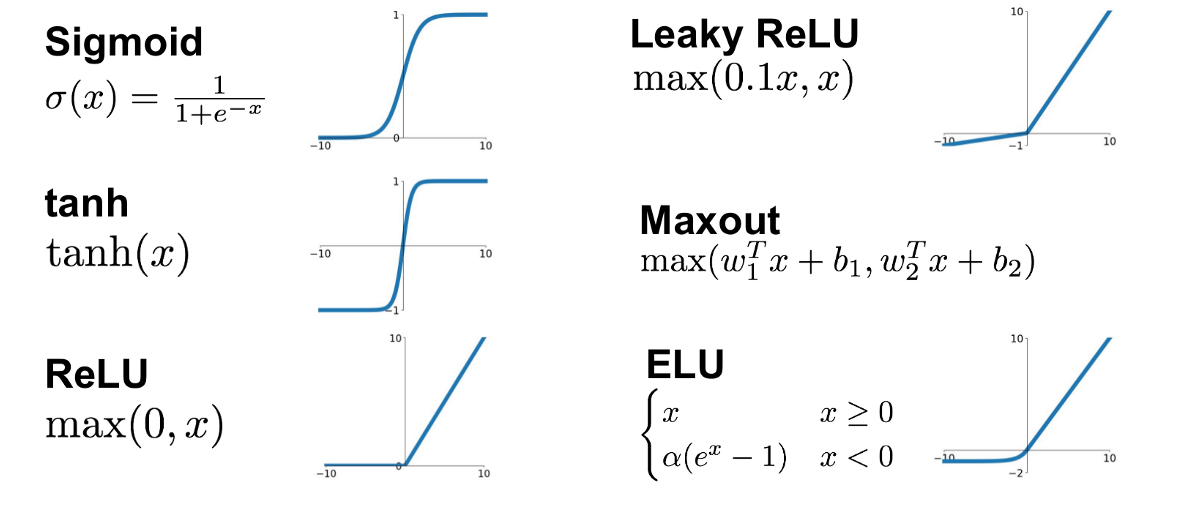
-Fully connected layer :

+)Được dùng để đưa ra kết quả. Sau khi các lớp convolutional layer và pooling layer đã nhận được hình ảnh truyền qua nó, ta thu được một model chứa nhiều thông tin về ảnh. Để liên kết các đặc điểm đó lại và cho ra output ta dùng fully connected layer. Tầng này có chức năng chuyển ma trận đặc trưng ở tầng trước thành vector chứa xác suất của các đối tượng cần được dự đoán.



Hình ‑ Minh họa cho đâu ra sau khi qua lớp fully connected

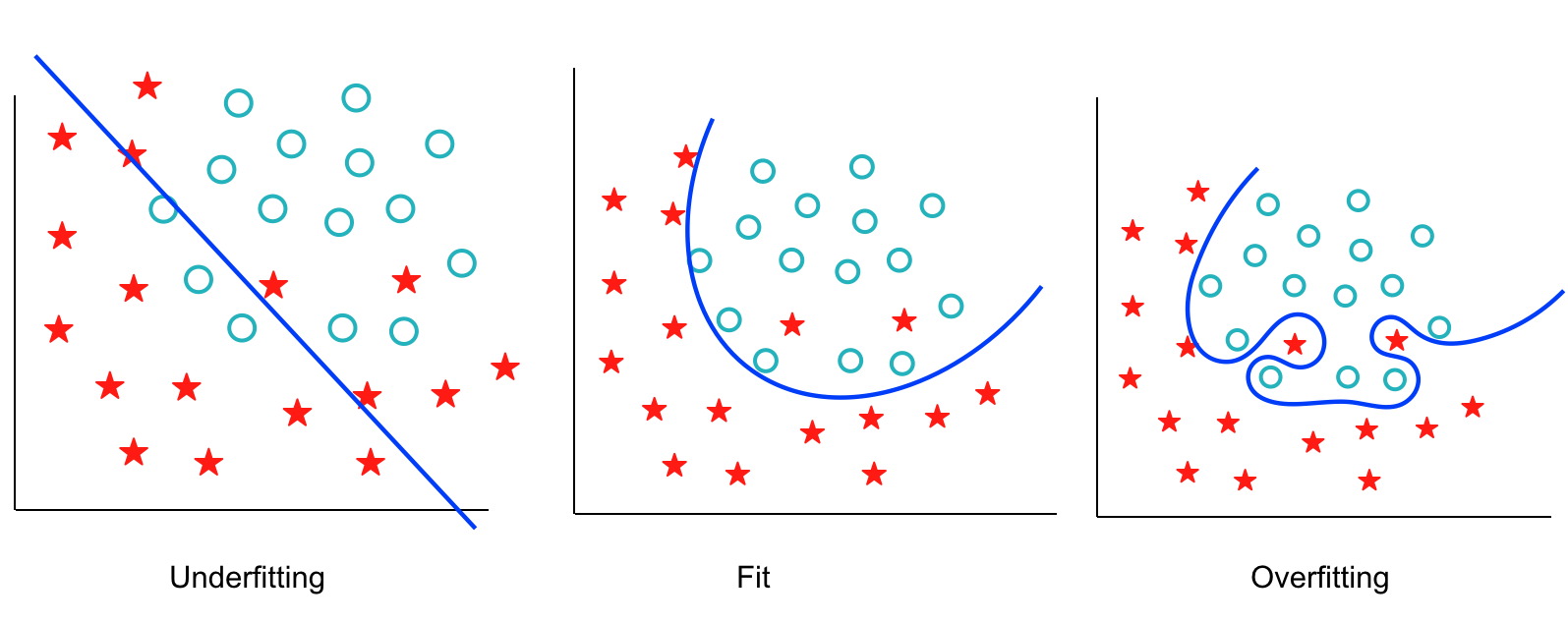
-Activation Function :

+)Activation functions là những hàm phi tuyến được áp dụng vào đầu ra của các nơ-ron trong tầng ẩn ( hidden layer) của một mô hình mạng, và được sử dụng làm input data cho tầng tiếp theo.

Hình ‑ Activation Function

+)Về các cấu trúc mạng cụ thể, các activation khác nhau sẽ được sử dụng, tuỳ vào độ sâu của mạng, output mong muốn, thậm chí là dữ liệu của bài toán.

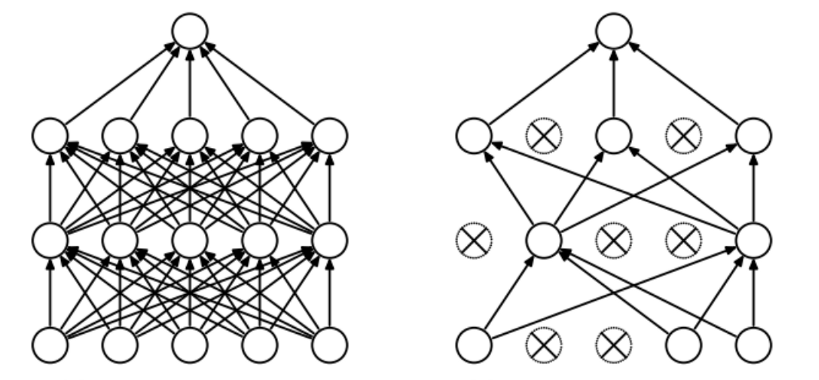
-Drop out :

+)Một số vấn đề thường gặp trong xử lí network là Overfitting. Overfitting là hiện tượng mô hình rất hợp lý, rất khớp với tập huấn luyện nhưng khi đem ra dự đoán với dữ liệu mới thì lại không phù hợp. Nguyên nhân có thể do ta chưa đủ dữ liệu để đánh giá hoặc do mô hình của ta quá phức tạp. Mô hình bị quá phức tạp khi mà mô hình của ta sử dụng cả những nhiễu lớn trong tập dữ liệu để học, dấn tới mất tính tổng quát của mô hình.

Hình ‑ Minh họa cho những trường hợp của model sau khi huấn luyện xong

+)Drop out là kĩ thuật sẽ bỏ qua một vài đơn vị ( neural ) trong quá trình huấn luyện mô hình, những neural bị bỏ qua sẽ được lựa chọn một cách ngẫu nhiên.Nghĩa là, tại mỗi giai đoạn huấn luyện, mỗi nút trong mạng neural tích chập bị bỏ qua là với xác xuất là 1-p và được chọn là p.

+)Vai trò của Drop out : Khi chúng ta sử dụng full connected layer, các neural sẽ phụ thuộc “mạnh” lẫn nhau trong suốt quá trình huấn luyện, điều này làm giảm sức mạng cho mỗi neural và dẫn đến việc over-fitting mà mất đi tính chung của dữ liệu.



Hình ‑ Minh họa sau khi mạng neural được áp dụng Drop out

+)Một số đặc điểm khi mô hình có sử dụng drop out :

* Dropout ép mạng neural phải tìm ra nhiều robust features hơn, với đặc điểm là chúng phải hữu ích hơn, tốt hơn khi kết hợp với nhiều neuron khác.
* Với H unit trong mô hình, mỗi unit đều có xác xuất bị bỏ qua hoặc được chọn, chúng ta sẽ có 2^H mô hình có thể có. Trong pha test, toàn bộ network được sử dụng và mỗi hàm activation được giảm đi với hệ số p.
* Người ta thường dùng hệ số dropout là 0.5. Việc sử dụng giảm 50% của dropout giúp kết quả đạt được là tốt nhất so với các phương pháp chuẩn hoá khác.

### Tổng quan về vi xử lí ARM Cortex M3 :

-Vi xử lý lõi ARM Cortex dựa trên 3 cấu hình của kiến trúc ARMv7:

+Cấu hình A : cho các ứng dụng Application, yêu cầu cao chạy trên các hệ điều hành mở và phức tạp như Linux, Android…

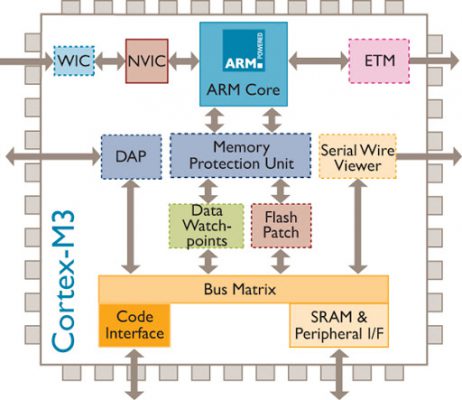
+Cấu hình R : cho phép chạy ứng dụng thời gian thực.

+Cấu hình M : cho các ứng dụng vi điều khiển.

-Bộ vi xử lý ARM Cortex-M3 là bộ vi xử lý ARM đầu tiên dựa trên kiến trúc ARMv7-M và được thiết kế đặc biệt để đạt được hiệu suất cao trong các ứng dụng nhúng cần tiết kiệm năng lượng và chi phí, chẳng hạn như các vi điều khiển, hệ thống cơ ô tô, hệ thống kiểm soát công nghiệp và hệ thống mạng không dây. Thêm vào đó là việc lập trình được đơn giản hóa đáng kể giúp kiến trúc ARM trở thành một lựa chọn tốt cho ngay cả những ứng dụng đơn giản nhất.

-Cấu trúc bộ vi xử lý ARM Cortex M3 :

+Bộ vi xử lý ARM Cortex-M3 dựa trên kiến trúc ARMv7-M có cấu trúc thứ bậc. Nó tích hợp lõi xử lý trung tâm, với các thiết bị ngoại vi hệ thống tiên tiến để tạo ra các khả năng như kiểm soát ngắt, bảo vệ bộ nhớ, gỡ lỗi và theo vết hệ thống.

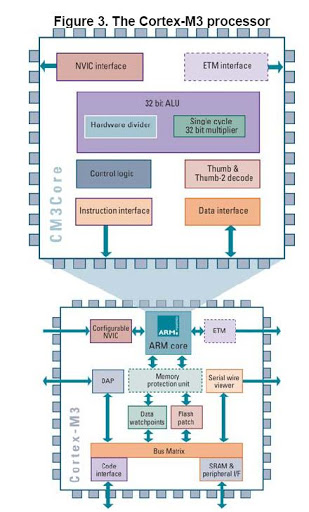
+ARM Cortex M3 dựa trên cấu trúc Havard, được đặc trưng bằng sự tách biệt giữa vùng nhớ dữ liệu và chương trình. Vì có thể đọc cùng lúc lệnh và dữ liệu từ bộ nhớ, bộ vi xử lý ARM Cortex-M3 có thể thực hiện nhiều hoạt động song song, tăng tốc thực thi ứng dụng.

Hình ‑ Cấu trúc vi xử lý ARM Cortex M3

-Cấu trúc lỗi ARM Cortex M3 :

+Lõi ARM Cortex có cấu trúc đường ống gồm 3 tầng: Instruction Fetch (nhận lệnh ), Instruction Decode (giải mã ) và Instruction Execute ( thực thi ). Khi gặp một lệnh nhánh, tầng decode chứa một chỉ thị nạp lệnh suy đoán có thể dẫn đến việc thực thi nhanh hơn. Bộ xử lý nạp lệnh dự định rẽ nhánh trong giai đoạn giải mã. Sau đó, trong giai đoạn thực thi, việc rẽ nhánh được giải quyết và bộ vi xử lý sẽ phân tích xem đâu là lệnh thực thi kế tiếp. Nếu việc rẽ nhánh không được chọn thì lệnh tiếp theo đã sẵn sàng. Còn nếu việc rẽ nhánh được chọn thì lệnh rẽ nhánh đó cũng đã sẵn sàng ngay lập tức, hạn chế thời gian rỗi chỉ còn một chu kỳ

+Lõi Cortex-M3 chứa một bộ giải mã cho tập lệnh Thumb truyền thống và Thumb-2 mới, một bộ phân chia logic ALU(arithmetic logic unit) tiên tiến hỗ trợ nhân chia phần cứng, điều khiển logic, và các giao tiếp với các thành phần khác của bộ xử lý.

+Bộ vi xử lý Cortex-M3 là một bộ vi xử lý 32-bit, với độ rộng của đường dẫn dữ liệu 32 bit, các dải thanh ghi và giao tiếp bộ nhớ. Có 13 thanh ghi đa dụng, hai con trỏ ngăn xếp, một thanh ghi liên kết, một bộ đếm chương trình và một số thanh ghi đặc biệt trong đó có một thanh ghi trạng thái chương trình.

Hình ‑ Cấu trúc lỗi ARM Cortex M3

### Ngôn ngữ C trong lập trình ARM:

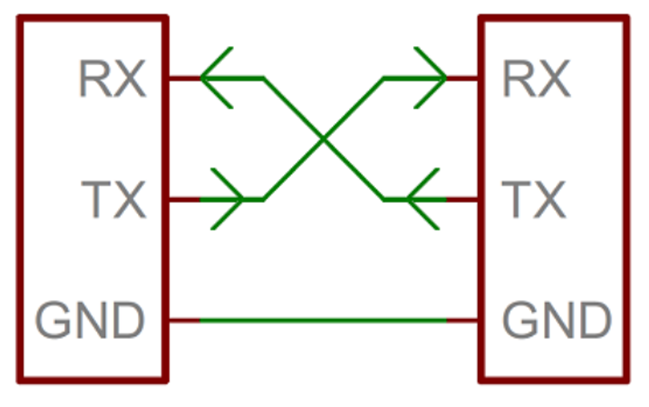
-Ngôn ngữ C được dùng để viết hệ điều hành UNIX bởi Dennis Ritchie vào năm 1971.

-C không bị ràng buộc bởi một phần cứng hay một hệ thống cụ thể đều này giúp cho chương trình chạy mà không cần thay đổi cho từng hệ thống riêng biệt.

- Là một ngôn ngữ tầm trung bởi nó kết hợp giữa ngôn ngữ bậc cao và cũng có thể truy cập rất sâu vào hệ thống như ngôn ngữ bậc thấp (chẳng hạn thông qua con trỏ hay các hàm Assembly nhúng vào chương trình).

-Ngày nay, ngôn ngữ C được sử dụng rất phổ biến trong lập trình nhúng vì dễ dàng tiếp cận và sử dụng.

### Chuẩn giao tiếp UART:

-UART (Universal Asynchronous Receiver – Transmitter ) là một mạch tích hợp được sử dụng trong việc truyền dẫn dữ liệu nối tiếp giữa máy tính và các thiết bị ngoại vi. Có chức năng chính là truyền dữ liệu nối tiếp. Trong UART, giao tiếp giữa hai thiết bị có thể được thực hiện theo hai phương thức là giao tiếp dữ liệu nối tiếp và giao tiếp dữ liệu song song.

Hình ‑ Minh họa 2 thiết bị giao tiếp UART

-Trong UART có các thông số chính :

+Baud rate (tốc độ baud ): Khoảng thời gian để 1 bit được truyền đi. Phải được cài đặt giống nhau ở cả phần gửi và nhận

+Frame (khung truyền): Khung truyền quy định về mỗi lần truyền bao nhiêu bit

+Start bit: là bit đầu tiên được truyền trong 1 Frame. Báo hiệu cho thiết bị nhận có một gói dữ liệu sắp đc truyền đến. Đây là bit bắt buộc.

+Data: dữ liệu cần truyền. Bit có trọng số nhỏ nhất LSB được truyền trước sau đó đến bit MSB.

+Parity bit: kiểm tra dữ liệu truyền có đúng không

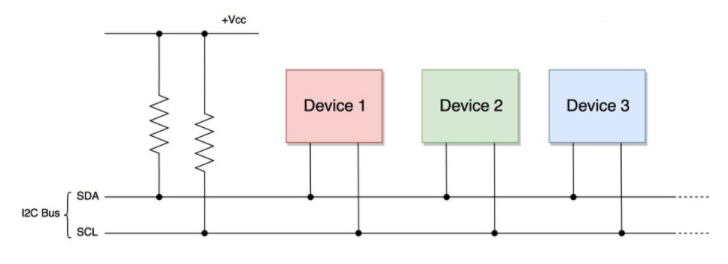
+Stop bit : là 1 hoặc các bit báo cho thiết bị rằng các bit đã được gửi xong. Thiết bị nhận sẽ tiến hành kiểm tra khung truyền nhằm đảm bảo tính đúng đắn của dữ liệu. Đây là bit bắt buộc

### Chuẩn giao tiếp I2C :

-I2C (Inter-Integrated Circuit ) là một loại bus nối tiếp được phát triển bởi Philips Semiconductors để truyền dữ liệu giữa một bộ xử lý trung tâm với nhiều IC trên cùng một board mạch chỉ sử dụng hai đường truyền tín hiệu. Ban đầu, loại bus chỉ ngày dùng trong các linh kiện điện tử Philips. Sau đó, do tính ưu việt và đơn giản của nó, I2C đã được chuẩn hóa và được dùng rộng dãi trong các module truyền thông nối tiếp của vi mạch tích hợp ngày này.

-Đây là một loại giao thức giao tiếp nối tiếp đồng bộ. Nghĩa là các bit dữ liệu được truyền từng bit một theo các khoảng thời gian đều đặn được thiết lập bởi một tín hiệu đồng hồ tham chiếu.

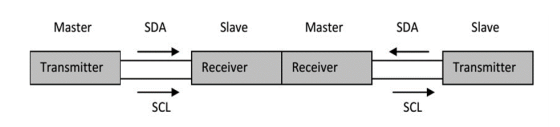
-I2C hoạt động dựa trên chế độ một Master và có thể có một hoặc nhiều thiết bị Slave. I2C sử dụng hai đường truyền tín hiệu :



Hình ‑ Minh họa các thiết bị giao tiếp I2C

+Một đường xung nhịp đồng hồ ( SCL ) chỉ do Master phát đi ( thông thường ở 100kHz và 400kHz, mức cao nhất là 1Mhz và 3.4MHz ).

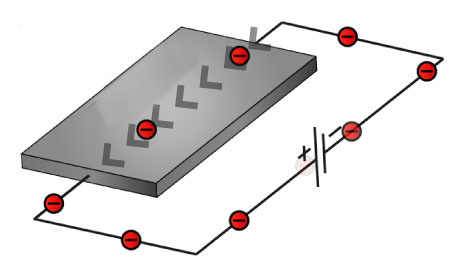
+Một đường dữ liệu ( SDA ) theo hai hướng.



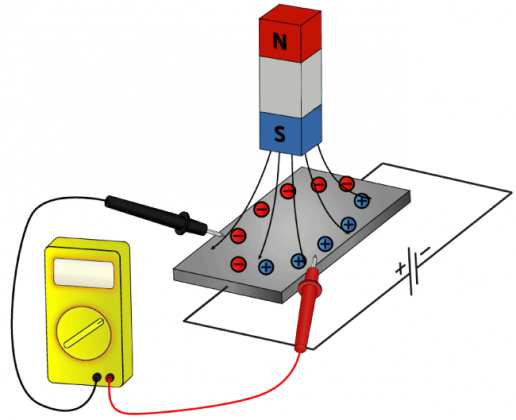
Hình ‑ Hướng giao tiếp của giao thức I2C

- Một thiết bị hay một IC khi kết nối với bus I2C, ngoài một địa chỉ (duy nhất) để phân biệt, nó còn được cấu hình là thiết bị chủ hay tớ. Thiết bị chủ nắm vai trò tạo xung đồng hồ cho toàn hệ thống, khi giữa hai thiết bị chủ-tớ giao tiếp thì thiết bị chủ có nhiệm vụ tạo xung đồng hồ và quản lý địa chỉ của thiết bị tớ trong suốt quá trình giao tiếp. Thiết bị chủ giữ vai trò chủ động, còn thiết bị tớ giữ vai trò bị động trong việc giao tiếp.

### Hiệu ứng Hall :

-Hiệu ứng Hall là một hiệu ứng vật lý được thực hiện khi áp dụng một từ trường vuông góc lên một bản làm bằng kim loại hay chất bán dẫn hay chất dẫn điện nói chung (thanh Hall) đang có dòng điện chạy qua. Lúc đó người ta nhận được hiệu điện thế (hiệu thế Hall) sinh ra tại hai mặt đối diện của thanh Hall. Hiệu ứng này được khám phá bởi Edwin Herbert Hall vào năm 1879.

Hình ‑ Dòng điện chạy trong thanh Hall

-Hiệu ứng Hall được giải thích dựa vào bản chất của dòng điện chạy trong vật dẫn điện. Dòng điện này chính là sự chuyển động của các điện tích (ví dụ như electron trong kim loại). Khi chạy qua từ trường, các điện tích chịu lực Lorentz bị đẩy về một trong hai phía của thanh Hall, tùy theo điện tích chuyển động đó âm hay dương. Sự tập trung các điện tích về một phía tạo nên sự tích điện trái dầu ở 2 mặt của thanh Hall, gây ra hiệu điện thế Hall.

Hình ‑ Xuất hiện điện áp 2 bên thanh hall khi ta đặt từ trường vào

-Ứng dụng của hiệu ứng Hall : đo cường độ dòng điện, đo công suất điện, xác định vị trí và chuyển động…

## PHẦN CỨNG:

### Raspberry Pi 4 Model B:

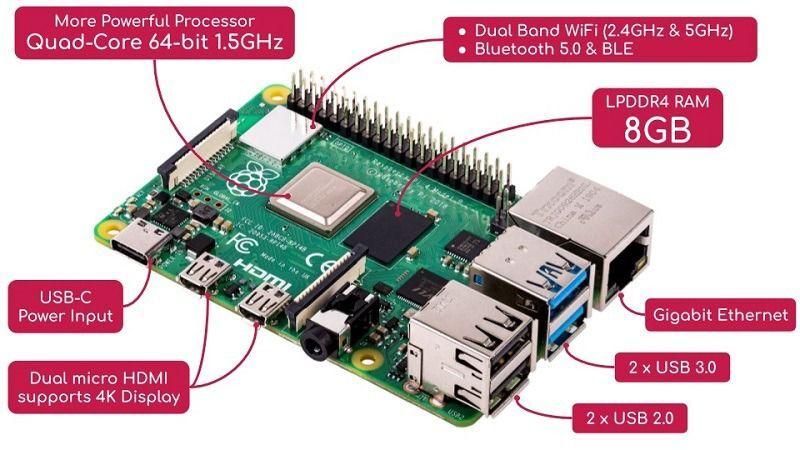
**Giới thiệu :**

-Raspberry Pi là máy tính chỉ có một board mạch ( hay còn gọi là máy tính nhúng ) kích thước chỉ bằng một cái thẻ tín dụng, tích hợp nhiều phần cứng mạnh mẽ đủ khả năng chạy hệ điều hành và cài đặt được nhiều ứng dụng trên nó. Được phát triển tại Anh bới Raspberry Pi Foundation với mục đích ban đầu là thúc đẩy việc giảng dạy về khoa học máy tính cơ bản trong các trường đại học và cái nước đang phát triển.

-Raspberry Pi có thể đáp ứng được hầu hết các yêu cầu cơ bản của một chiếc máy tính bình như thường như lướt web, xem phim HD, lập trình, nghe nhạc…Cho đến những ứng dụng trong hệ thống nhúng như điều khiển robot, nhà thông minh, ứng dụng nhận dạng...Raspberry tiêu thụ ít điện năng và có thể sử dụng liên tục trong 24/24.

-Đặc tính của Raspberry Pi sử dụng chip Broadcom BCM2805 SoC ( System on Chip ) có chứa bộ xử lí ARM1176JZF-S 700Mhz, GPU Video Core IV, bộ nhớ RAM 256MB sau nâng cấp lên 512MB. Nó không có ổ cứng hay SSD đi kèm mà sử dụng thẻ SD để lưu dữ và khởi động hệ điều hành.

-Raspberry Pi sử dụng hệ điều hành linux và không thể cài đặc hệ điều hành windows vì bộ vi xử lí SoC Broadcom BCM2805 dựa trên cấu trúc ARM nên không hỗ trợ mã x86/x64 nhưng có thể đáp ứng được mọi thứ mà một máy tính windows làm được.

-Raspberry Pi 4 Model B là phiên bản mới nhất của các dòng Raspberry đã được cải thiện và tối ưu về mọi mặt so với các phiên bản cũ như CPU mạnh hơn, tùy chọn Ram 1GB/2GB/4GB, dual-display Micro HDMI 4K,...

Hình ‑ Kit Raspberry Pi 4B+

**Thông số kĩ thuật :**

|  |  |
| --- | --- |
| SoC ( System on chip ) | Boardcom BCM2711 quad-core Cortex A72 @ 1.5GHz |
| GPU | VideoCore VI with OpenGL ES 1.1, 2.0, 3.0 |
| Video Decode | H.265 4Kp60, H.264 1080p60 |
| Video Encode | H.264 1080p30 |
| Memory | 1GB, 2GB or 4GB LPDDR4 |
| Storage | MicroSD card |
| Video & Audio Output | 2x mirco HDMI ports up to 4Kp60  3.5mm AV port ( composite + audio )  MIPI DSI connector |
| Camera | MIPI CSI connector |
| Ethernet | Native Gigabit Ethernet |
| WiFi | Dual Band 802.11 b/g/n/ac |
| Bluetooth | Bluetooth 5.0 + BLE |
| USB | 2x USB 3.0 + 2x USB 2.0 |
| Expansion | 40-Pin GPIO header |
| Power Supply | 5V via USB type-C to 3A  5V via GPIO header to 3A  Power over Ethernet via PoE HAT |
| Dimensions | 85x56 mm |

Bảng ‑ Thông số kĩ thuật Raspberry Pi 4

**Chức năng các linh kiện của Board mạch :**

-Cổng USB: 4 cổng USB này cho phép gắn 4 thiết bị USB khách nhau

như chuột, bàn phím….

-Cổng Ethernet: Cổng Ethernet để nhanh chóng thiết lập và truy cập

internet.

-Audio/Video: kết hợp giắc cắm âm thanh 3.5mm và video tổng hơp.

-Cổng CSI camera: cho phép module camera cắm vào trực tiếp.

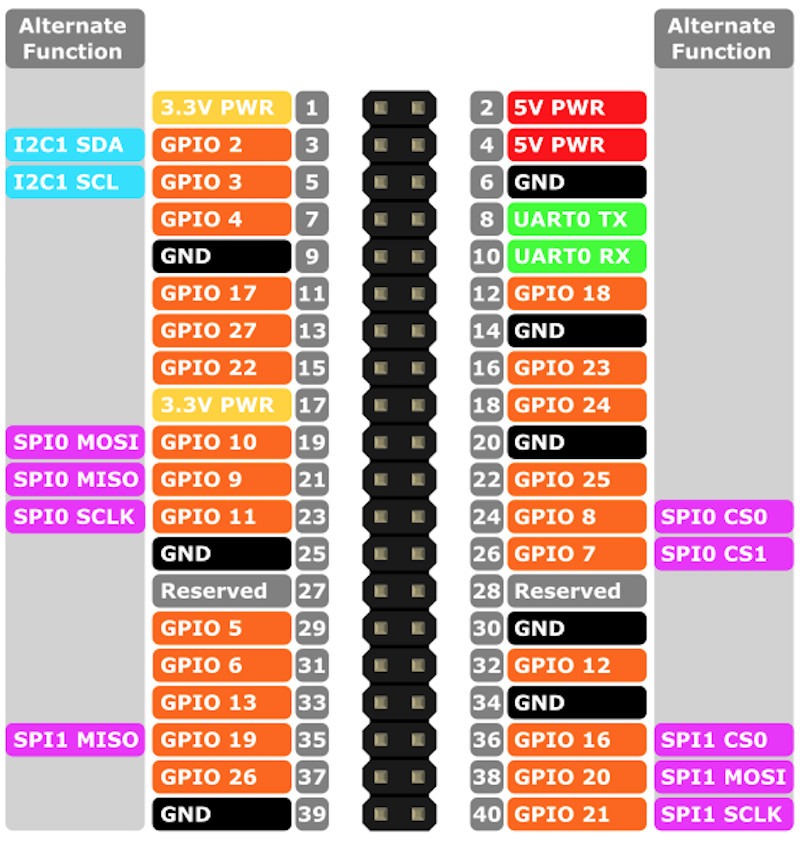
-Nguồn vào Micro USB: công cấp nguồn vào cho Raspberry.

-Cổng DSI: cho phép màn hình cắm vào trực tiếp.

-Khe cắm thẻ MicroSD: cắm thẻ chứa hê điều hành sẽ khởi động khi ta

cấp nguồn cho Rasperry.

-Bluetooth: hỗ trợ Bluetooth 4.1. Bluetooth năng lượng thấp (BLE).

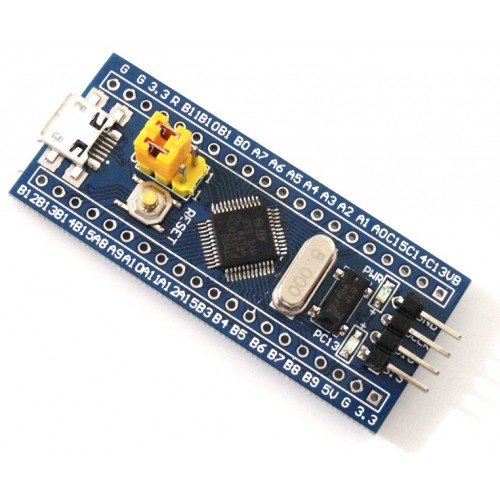
-Chân GPIO: hỗ trợ 40 chân vào ra.

Hình ‑ Sơ đồ GPIO Raspberry Pi

* + 1. STM32F103C8T6 :

-STM32 là một trong những dòng chip phổ biến của ST với nhiều họ thông dụng như F0,F1,F2,F3,F4….. Stm32f103 thuộc họ F1 với lõi là ARM COTEX M3. STM32F103 là vi điều khiển 32 bit, tốc độ tối đa là 72Mhz.

- Một số ứng dụng chính: dùng cho driver để điều khiển ứng dụng, điều khiển ứng dụng thông thường, thiết bị cầm tay và thuốc, máy tính và thiết bị ngoại vi chơi game, GPS cơ bản, các ứng dụng trong công nghiệp, thiết bị lập trình PLC, biến tần, máy in, máy quét, hệ thống cảnh báo, thiết bị liên lạc nội bộ…

-Kit phát triển STM32F103C8T6 Blue Pill có giá thành rẽ và cộng đồng sử dụng lớn thích hợp cho sinh viên và người yêu thích để khám phá thiết bị ARM Cortex-M3.

Hình ‑ Kit STM32F103C8T6 Blue Pill

**Sơ lược về phần cứng trên kit STM32F103C8T6 Blue Pill :**

-1 cổng Mini USB dùng để cấp nguồn, nạp cũng như debug.

-2 MCU bao gồm 1 MCU nạp và 1 MCU dùng để lập trình.

-Có chân Output riêng cho các chân mạch nạp trên MCU1.

-Có chân Output đầy đủ cho các chân MCU2.

-Chân cấp nguồn ngoài riêng cho MCU2 nếu không sử dụng nguồn từ USB.

-Thạch anh 32,768khz dùng cho RTC và Backup.

-Chân nạp dùng cho chế độ nạp boot loader.

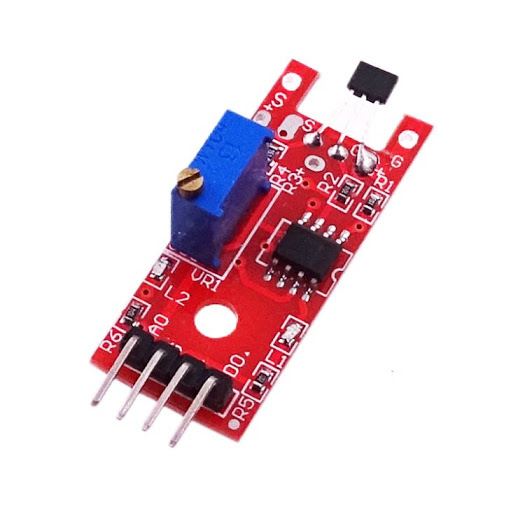
-Nút Reset ngoài và 1 led hiển thị trên chân PB9, 1 led báo nguồn cho MCU2.

**Cấu hình của STM32F103C8T6 :**

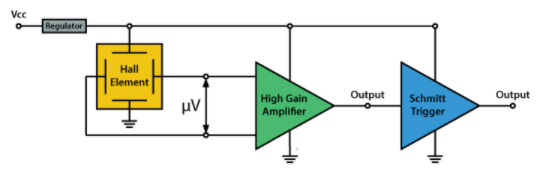
|  |  |
| --- | --- |
| Vi xử lí | ARM 32-bit Cortex M3 với clock max là 72Mhz |
| Bộ nhớ | 64 kbytes bộ nhớ Flash(bộ nhớ lập trình).  20kbytes SRAM. |
| Clock,rest và quản lí nguồn | Điện áp hoạt động 2.0V -> 3.6V.  Power on reset(POR), Power down reset(PDR) và programmable voltage detector (PVD).  Sử dụng thạch anh ngoài từ 4Mhz -> 20Mhz.  Thạch anh nội dùng dao động RC ở mode 8Mhz hoặc 40khz.  Sử dụng thạch anh ngoài 32.768khz được sử dụng cho RTC. |
| Trong trường hợp điện áp thấp | Có các mode :ngủ, ngừng hoạt động hoặc hoạt động ở chế độ chờ.  Cấp nguồn ở chân Vbat bằng pin để hoạt động bộ RTC và sử dụng lưu trữ data khi mất nguồn cấp chính. |
| GPIO | 20 I/O |
| 2 Bộ ADC 12 bit với 9 kênh cho mỗi bộ. | Khoảng giá trị chuyển đổi từ 0 – 3.6V.  Lấy mẫu nhiều kênh hoặc 1 kênh.  Có cảm biến nhiệt độ nội. |
| 7 kênh DMA | Bộ chuyển đổi này giúp tăng tốc độ xử lý do không có sự can thiệp quá sâu của CPU.  Hỗ trợ DMA cho ADC, I2C, SPI, UART. |
| 7 timer | 3 timer 16 bit hỗ trợ các mode IC/OC/PWM.  1 timer 16 bit hỗ trợ để điều khiển động cơ với các mode bảo vệ như ngắt input, dead-time..  2 watdog timer dùng để bảo vệ và kiểm tra lỗi.  1 sysTick timer 24 bit đếm xuống dùng cho các ứng dụng như hàm Delay…. |
| 9 kênh giao tiếp | 2 bộ I2C(SMBus/PMBus).  3 bộ USART(ISO 7816 interface, LIN, IrDA capability, modem control).  2 SPIs (18 Mbit/s).  1 bộ CAN interface (2.0B Active)  USB 2.0 full-speed interface |
| Kiểm tra lỗi CRC và 96-bit ID. |  |

Bảng ‑ Cấu hình STM32F103C8T6

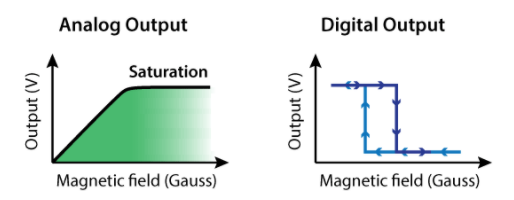
* + 1. Cảm biến từ trường Hall :

- Cảm biến Hall là loại cảm biến dùng để phát hiện từ tính của nam châm. Cảm biến Hall hoạt động dựa theo nguyên lí của hiệu ứng Hall.

Hình ‑ Module cảm biến từ trường hall

- Có hai loại cảm biến hall. Một loại có đầu ra analog và một loại có đầu ra digital. Cảm biến hall có đầu ra analog bao gồm bộ điều chỉnh điện áp (Regulator), thành phần hall (Hall Element), bộ khuếch đại (High Gain Amplifer). Đầu ra cảm biến là analog và tỉ lệ với đầu ra thanh phần hall hoặc cường độ từ trường. Đầu ra digital có 2 mức tích cực thấp và cao nhờ bổ xung thêm bộ Trigger Schmitt.

Hình ‑ Sơ đồ khối của cảm biến hall



Hình ‑ Dạng song đầu ra của chân Analog và Digital trên cảm biến Hall

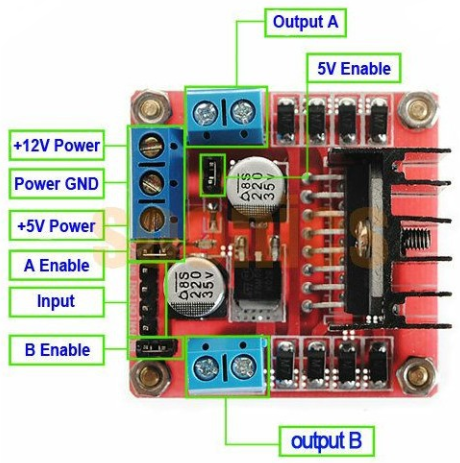
- Cảm biến hall được sử dụng như công tắc giới hạn, ví dụ trong máy in 3D và Máy CNC, cũng như để phát hiện và định vị trong các hệ thống tự động hóa công nghiệp, đo tốc độ quay…

* + 1. Module điều khiển động cơ L298:

-Mạch điều khiển động cơ DC L298 có khả năng điều khiển 2 động cơ DC, 1 động cơ bƣớc, có 4 lỗ nằm ở 4 góc thuận tiện cho ngƣời sử dụng cố định vị trí của module. Dòng tối đa 2A mỗi động cơ, mạch tích hợp diode bảo vệ và IC nguồn 7805 giúp cấp nguồn 5VDC cho các module khác.

+Có thể gắn tản nhiệt chống nóng cho IC, giúp IC có thể điều khiển với dòng đỉnh đạt 2A.

+IC L298N đƣợc gắn với các đi-ốt trên board giúp bảo vệ vi xử lý chống lại các dòng điện cảm ứng từ việc khởi động/ tắt động cơ.

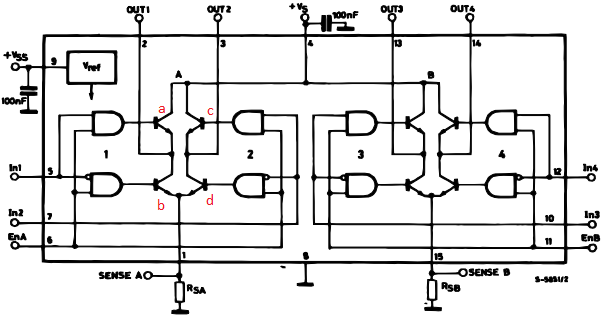
**Sơ đồ chân :**

Hình ‑ Module L298

-Input gồm IN1 , IN2 ,IN3, IN4. Trong đó : IN1, IN2 điều khiển OutputA ; IN3,IN4 điều khiển OutputB.

-Nguồn 12V cấp cho module hoạt động.

-GND nối chung với GND của vi điều khiển.

**Nguyên lý hoạt động của mạch cầu H :**

Hình ‑ Mạch cầu H

-Sử dụng nguyên lí của mạch cầu H để điều khiển động cơ.

-Ta xét mạch cầu H (A) và cố định 1 mức điện áp tương đương với mức High = 1 và Low = 0.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| IN1 | IN2 | Transistor a | Transistor b | Transistor c | Transistor d | Output |
| 0 | 0 | Không dẫn | Dẫn | Không dẫn | Dẫn | Không có dòng điẹn |
| 0 | 1 | Không dẫn | Dẫn | Dẫn | Không dẫn | Chạy theo chiều c-b |
| 1 | 0 | Dẫn | Không dẫn | Không dẫn | Dẫn | Chạy theo chiều a-d |
| 1 | 1 | Dẫn | Không dẫn | Dẫn | Không dẫn | Không có dòng điện |

Bảng ‑ Bảng giá trị hoạt động của mạch cầu H

**-**Tương tự cho mạch cầu H (B).

**Thông số kĩ thuật :**

|  |  |
| --- | --- |
| Driver | IC L298 tích hợp hai mạch cầu H. |
| Điện áp điều khiển | 5-12V. |
| Dòng tối đa cho mỗi cầu H | 2A |
| Điện áp của tín hiệu điều khiển | 5-7V. |
| Dòng của tín hiệu điều khiển | ~3,6mA |
| Công suất hao phí | 20W |
| Mức điện áp logic | Low : -0.3V-1.5V, High : 2.3V-Vss |
| Kích thước | 43x43x27 mm |

Bảng ‑ Thông số kĩ thuật module L298

* + 1. LCD 16x2:

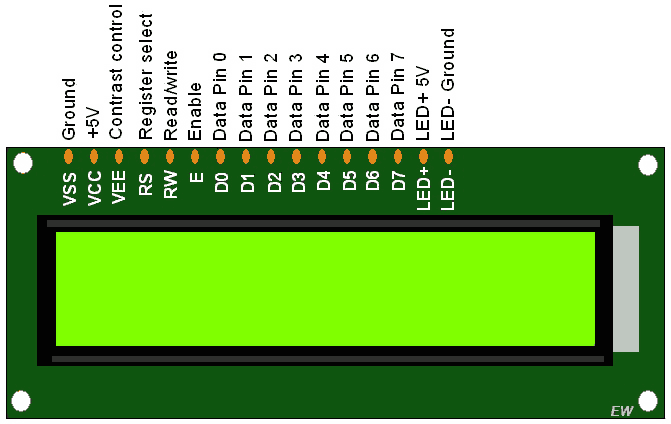
-Được sản xuất từ năm 1970, LCD ( Liquid Crystal Display ) là một loại vật chất phản xạ ánh sáng khi điện thế thay đổi. Nó hoạt động dựa trên nguyên tắc ánh sáng nền (Back Light). Nó bao gồm một lớp chất lỏng nằm giữa 2 lớp kính phân cực ánh sáng. Bình thường, khi không có điện áp, các tinh thể này được xếp thẳng hàng giữa hai lớp cho phép ánh sáng truyền qua theo hình xoắn ốc. Hai bộ lọc phân cực, 2 bộ lọc màu và 2 bộ cân chỉnh sẽ xác định cường độ ánh sáng đi qua và màu nào được tạo ra trên một pixel. Khi có điện áp cấp vào, lớp canh chỉnh sẽ tạo một vùng điện tích, canh chỉnh lại các tinh thể lỏng đó. Nó không cho phép ánh sáng đi qua để hiện thị lên hình ảnh tại vị trí điểm ảnh đó. Các điểm ảnh trong màn hình LCD là một transistor cực nhỏ ở một trong 2 chế độ: cho phép ánh sáng đi qua hoặc không. Điểm ảnh bao gồm 3 yếu tố màu: đỏ, xanh lá, xanh dương. Các màn hình LCD trước đây thường tiêu thụ điện năng nhiều, độ tương phản thấp cho đến khi các nhà khoa học người Anh tìm ra "Biphenyl" - vật liệu chính của tinh thể lỏng, thì LCD mới thực sự phổ biến. LCD xuất hiện đầu tiên trong các máy tính cầm tay, trò chơi điện tử cầm tay, đồng hồ điện tử.

-LCD 16x được sử dụng rất nhiều trong các ứng dụng của vi điều khiển. LCD có rất nhiều ưu điểm so với các dạng hiển thị khác. Nó có khả năng hiển thị kí tự đa dạng, trực quan (chữ, số và kí tự đồ họa), dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tốn rất ít tài nguyên hệ thống và giá thành rẻ…

**Thông số kĩ thuật :**

|  |  |
| --- | --- |
| Điện áp MAX | 7V |
| Điện áp MIN | -0.3V |
| Hoạt động ổn định | 2.7 – 5.5V |
| Điện áp ra mức cao | >2.4V |
| Điện áp ra mức thấp | <0.4V |
| Dòng điện cấp nguồn | 350uA - 600uA |
| Nhiệt độ hoạt động | - 30 - 75 độ C |

Bảng ‑ Thông số kĩ thuật LCD1602

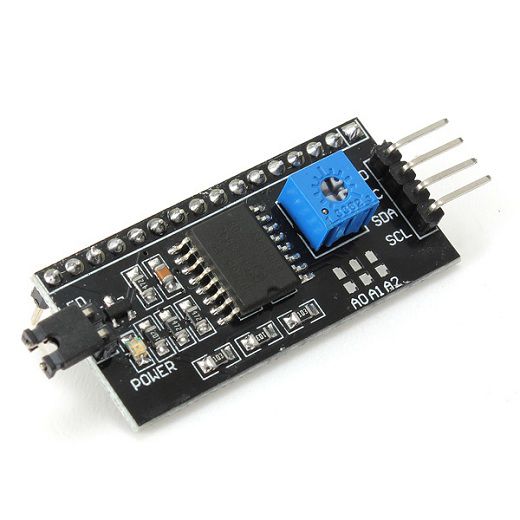
**Sơ đồ chân :**

**Chức năng các chân :**

|  |  |  |
| --- | --- | --- |
| Chân | Ký hiệu | Mô tả |
| 1 | VSS | Chân nối đât của LCD, nối chung đất với mạch điều khiển . |
| 2 | VDD | Chân cấp nguồn cho LCD, nối với nguồn 5V của mạch điều khiển. |
| 3 | VEE | Điều chỉnh độ tương phản của LCD. |
| 4 | RS | Chân chọn thanh ghi ( Register Select ).Có 2 chế độ :  +) Được nối với mức logic “0”: Bus D0-D7 sẽ nối với thanh ghi lệnh IR của LCD ở chế độ write (ghi) hoặc nối với bộ đệm địa chỉ của LCD ở chế độ read(đọc).  +) Được nối với mức logic “1”: Bus D0-D7 sẽ nối với thanh ghi dữ liệu DR của LCD. |
| 5 | R/W | Chận chọn chế độ đọc/ghi (read/write).  +) Được nối với mức logic “0”: LCD hoạt động ở chế độ ghi.  +) Được nối với mức logic “1”: LCD hoạt động ở chế độ đọc. |
| 6 | E | Chân cho phép ( Enable ). Sauk hi các tính hiệu được đặt lên Bus D0-D7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân E.  +Ở chế độ ghi : Dữ liệu ở bus sẽ được LCD chuyển vào thanh ghi bên trong nó khi phát hiện 1 xung của tín hiệu chân E.  +Ở chế độ đọc : Dữ liệu sẽ được LCD xuất ra D0-D7 khi phát hiện cạnh lên ở chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp. |
| 7-14 | D0-D7 | -8 đường của bus dữ liệu dùng để trao đổi thông tin. Có 2 chế độ sử dụng đường bus :  +Chế độ 4 bit : Dữ liệu được chuyển trên 4 đường từ D4-D7, bit MSB là D7.  +Chế độ 8 bit : Dữ liệu được chuyển trên cả 8 đường, bit MSB là D7. |
| 15 | Led+ | Nguồn dương của đèn nền. |
| 16 | Led- | Đất của đèn nền. |

Bảng ‑ Chức năng các chân trên LCD

* + 1. Module I2C-LCD:

-Thông thường, để điều khiển và hiển thị được kí tự từ vi điều khiển xuất ra màn hình 16x02 cần tới 7-8 dây nối đến chân vi điều khiển. Điều này gây ra rất nhiều phiền toái: đi sai dây, mạch rườm ra, khó viết code...Ta có thể khắc phục điều này bằng cách sử dụng giao tiếp I2C, việc điều khiển trực tiếp màn hình được chuyển sang cho IC xử lý nằm trên mạch. Chỉ việc gửi các mã lệnh cùng nội dung hiển thị, do vậy giúp vi điều khiển có nhiều thời gian để xử lý các tiến trình phức tạp khác.

Hình ‑ Module LCD-I2C

**Thông số kĩ thuật :**

|  |  |
| --- | --- |
| Điện áp hoạt động | 2.5-5V. |
| Hỗ trợ cho LCD | LCD1602, 1604, 2004( driver HD44780). |
| Giao tiếp | I2C. |
| Địa chỉ | 0x27 ( có thể điều chỉnh bằng cách ngắn mạch chân A0/A1/A2 ). |
| Kích thước | 41.5mm x 19mm x 15.3 mm ( LxWxH). |
| Trọng lượng | 5g. |
| Jump chốt | Cung cấp đèn nền cho LCD hoặc ngắt. |
| Biến trở | Xoay độ tương phản cho LCD. |

Bảng ‑ Thông số kỹ thuật module i2c-lcd

**Sơ đồ chân :**

|  |  |  |
| --- | --- | --- |
| Chân | Kí hiệu | Mô tả |
| 1 | Vcc | Kết nối vào nguồn từ 2.5-5V. |
| 2 | SDA | Kết nối với chân SDA của vi điều khiển. |
| 3 | SCL | Kết nối với chân SCL của vi điều khiển. |
| 4 | GND | Kêt nối chung đất với mạch điều khiển |
| Jump0-Jump15 |  | Nối tương ứng theo thứ tự 16 chân của LCD. |

Bảng ‑ Sơ đồ chân module i2c-lcd

****

Hình ‑ LCD được kết nối module lcd-i2c

* + 1. Động cơ DC giảm tốc GA25 Encoder :

Động cơ DC giảm tốc GA25 Encoder thường được sử dụng trong các ứng dụng cần xác định tốc độ, vị trí, chiều quay của động cơ DC: Robot mê cung, robot xe hai bánh tự cân bằng,..

Động cơ DC giảm tốc GA25 Encoder thực tế là động cơ DC GA25 thường có gắn thêm phần Encoder để có thể trả xung về vi điều khiển giúp xác định vị trí, vận tốc,… và thành Động cơ DC giảm tốc.

Về cách điều khiển thì động cơ DC giảm tốc GA25 Encoder sử dụng Driver như động cơ DC thường để điều khiển công suất động cơ, tốc độ và đảo chiều: L298, L293,…, chỉ có điểm khác biệt là có thêm phần encoder để có thể hồi tiếp (feedback) xung về Vi điều khiển, từ đó vi điều khiển tác động lại động cơ qua mạch công suất sử dụng các thuật toán điều khiển như PID,… để điều khiển tốc độ, vị trí,…

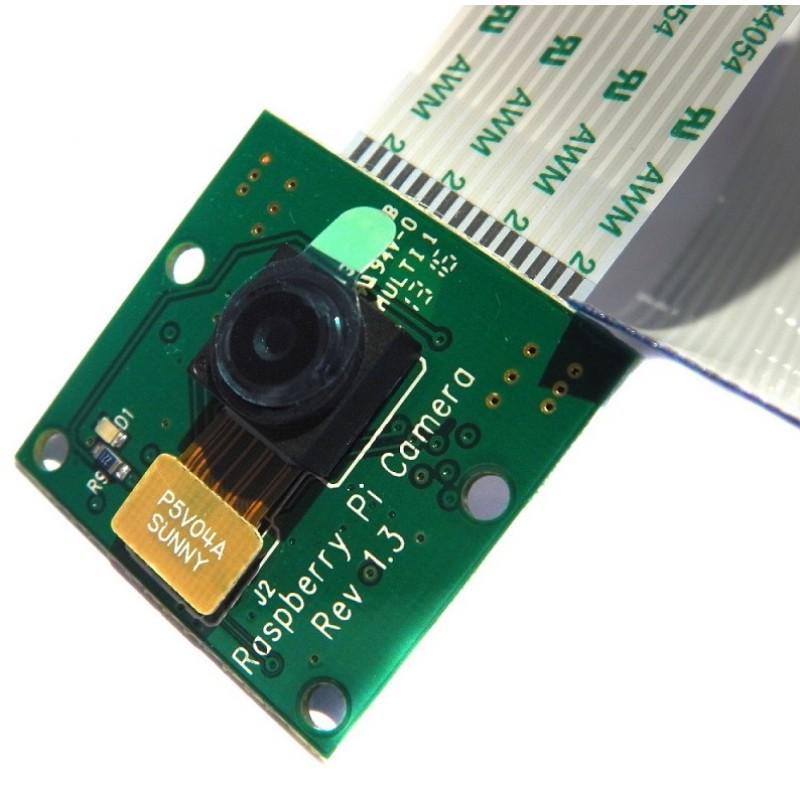
Hình ‑ Động cơ encoder G25

**Thông số kĩ thuật :**

|  |  |
| --- | --- |
| Tốc độ không tải | 1360rmp |
| Điện áp cấp cho động cơ hoạt động | 3-12V |
| Điện áp cấp cho encoder hoạt động | 3.3V |
| Đĩa encoder | 11 xung, 2 kênh A-B |
| Đường kính động cơ | 25mm |
| Đường kính trục | 4mm |
| Trọng lượng | 104g |

Bảng ‑ Thông số kĩ thuật động cơ encoder G25

* + 1. Camera pi v1.3 :

-Camera Raspberry Pi cắm trực tiếp vào đầu nối CSI trên Raspberry Pi. Nó có thể cung cấp hình ảnh độ phân giải tinh thể rõ nét 5MP hoặc quay video HD 1080p ở tốc độ 30 khung hình / giây! Tùy chỉnh được thiết kế và sản xuất bởi Tổ chức Raspberry Pi ở Anh, Camera Raspberry Pi có cảm biến Omnivision 5647 5MP trong một module lấy nét cố định. Module gắn với Raspberry Pi, bằng cách sử dụng cáp 15 Pin Ribbon, với Giao diện nối tiếp camera MIPI 15 chân chuyên dụng (CSI), được thiết kế đặc biệt để giao tiếp với máy ảnh. Bus CSI có khả năng tốc độ dữ liệu cực cao và nó mang dữ liệu điểm ảnh tới bộ xử lý BCM2837.

Hình ‑ Camera pi V1.3

**Thông số kĩ thuật**

|  |  |
| --- | --- |
| Độ phân giải | 2592x1944 ( 5 megapixel ) |
| Ống kính | F = 3,57mm, f/2.8 |
| Góc nhìn | 65o |
| Phạm vi lấy nét | 0,69m đến vô cực （ở mức 1,38m) |
| Hỗ trợ | 1080p @ 30 khung hình / giây với codec H.264 (AVC) , 720p @ 60fps và 640x480p @ 60/90 khung hình / giây |
| Giao diện | CSI |
| Kích thước | 25 mm x 25 mm x 10 mm (0,98 "x 0,98" x 0,39 ") |
| Trọng lượng | khoảng 2,8g |

Bảng ‑ Thông số kĩ thuật camera pi v1.3

## THIẾT KẾ:

### PHẦN CỨNG

**Yêu cầu đặc ra:**

-Dựa vào những yêu cầu của hệ thống đòi hỏi bộ xử lí trung tâm tính toán nhanh, dung lượng lớn, nhỏ gọn, có khả năng giao tiếp với nhiều đối tượng và thiết bị cảm biến khác nhau, sử dụng được nhiều giao thức giao tiếp, với những yêu cầu đó thì kit STM32F103T8C6 đã đáp ứng được những yêu cầu này.

-Sử dụng kit Raspberry Pi để thực hiện việc nhận dạng biển báo giao thông vì là một máy tính nhúng có hỗ trợ đầy đủ các phần mềm lập trình và dễ dàng giao tiếp với các vi điều khiển khác, hỗ trợ giao tiếp camera.

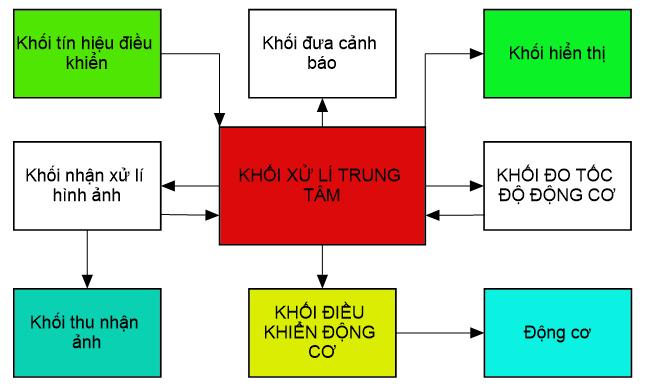
-Bên cạnh đó, thiết bị phải đảm bảo các chức năng :

+Hoạt động ổn định, thời gian trễ thấp.

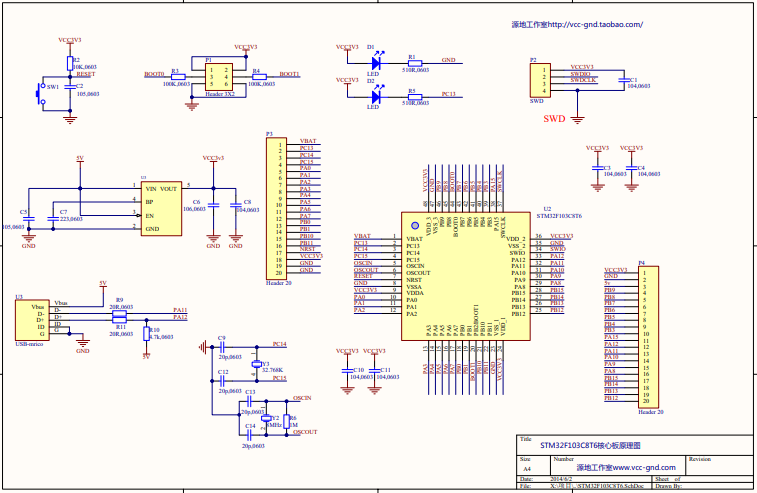
+Nhận dạng các biển báo giao thông, kết hợp với đo tốc độ và đưa ra cảnh báo nếu vượt quá tốc độ cho phép.

+Nhận dạng các biển báo và đo tốc độ phải có độ chính xác trên 90%.

+Thiết kế tiết kiệm năng lượng,gọn gàng,hoạt động ổn định, thời gian trễ thấp.

**Sơ đồ khối hệ thống :**

Hình ‑ Sơ đồ khối hệ thống

 **-**Khối xử lí trung tâm :

Hình ‑ STM32F103 Skematic

+Sử dụng STM32F103T8C6 có hiệu năng cao vì thế sẽ làm tốt nhiệm vụ quản lí, giám xác, giao tiếp tiếp với các khối các trong hệ thống.

+Nhận tín hiệu từ khối điều khiển

+Trao đổi dữ liệu với các khối xử lí hình ảnh và đo tốc độ động cơ thì từ nó đưa ra các quyết định của hệ thống.

+Từ đó xuất thông tin ra khối cảnh báo và hiển thị.

-Khối nhận xử lí hình ảnh :

+Sử dụng kit Raspberry Pi 4B, hỗ trợ các công cụ cho việc xử lí hình ảnh

+Nhận dữ liệu là hình từ khối thu hình ảnh. Sau đó lọc nhiễu,xử lí ảnh đầu vào và tiến hành thuật toán nhận dạng để phân loại và chuyển về tín hiệu logic.

+Truyền tín hiệu logic sử lí được về khối xử lí trung tâm.

-Khối thu nhận ảnh :

+Sử dụng Camera Pi v1.3 có kết nối phù hỗ hợp với Raspberry Pi, thu được hình ảnh chất lượng cao và ít nhiễu.

+Khả năng thu ảnh một cách real-time.

-Khối đo tốc độ động cơ :

+Sử dụng module cảm biến từ trường hall. Phát hiện từ trường từ một nam châm được gắn vào bánh xe, gửi tín hiệu về bộ xử lí trung tâm.

-Khối điều khiển tốc độ động cơ :

+Sử dụng module L298, nhận tín hiệu từ bộ xử lí trung tâm. Sau đó dựa vào nguyên lí điều khiển của mạch cầu H để điều khiển động cơ quay.

-Khối tín hiệu điều khiển :

+Một biến trở dùng để thay đổi mức điện áp trên chân ngõ ra từ đó tạo ra tín hiệu analog nhận vào bộ xử lí trung tâm. Tín hiệu để bộ điều khiển biết thay đổi tốc độ động cơ.

+Một nút nhấn là tín hiệu để bộ điều khiển quyết định chế độ hiển thị.Gồm 3 chế độ.

-Khối đưa ra cảnh báo :

+Buzzer phát ra âm thanh cho người dùng biết đang chạy quá tốc độ.

-Khối hiển thị :

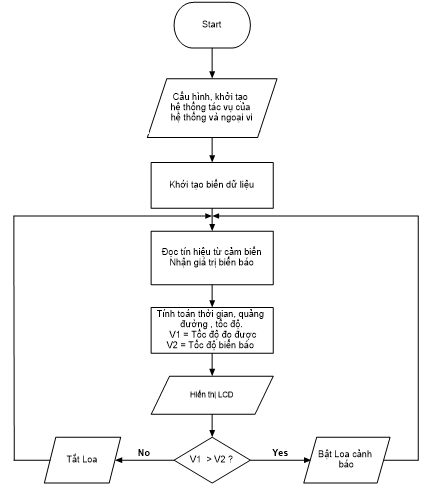
+LCD1602 sử dụng driver HD44780 kết nối với module i2c-lcd, có khả năng hiển thị 2 dòng với mỗi dòng 16 kí tự, màn hình có độ bền cao rất phổ biến trên thị trường, có ưu điểm là phẳng, cho hình ảnh sáng chân thật, tiết kiệm năng lượng.

* Nguyên lí hoạt động : Bộ xử lý trung tâm điều khiển tất cả các khối tác vụ khác. Hệ thống giao tiếp với người dùng thông qua khối ngoại vi. Khối xử lí ảnh và khối đo tốc độ động cơ sẽ gửi về bộ xử lý trung tâm để thực hiện đưa ra các cảnh báo nhắc nhở người dùng đã vượt quá tốc độ cho phép.

### PHẦN MỀM

**Sơ đồ giải thuật :**

-Chương trình chính :



Hình ‑ Chương trình chính

* Nguyên lý : +) Khi bắt đầu chương trình thì việc đầu tiên là sẽ cấu hình toàn cục cho cả hệ thống, việc này bao gồm : khởi tạo xung nhịp, kích hoạt các tác vụ như ngắt ngoài, timer, PWM, ADC, input/out…Tiếp theo là sẽ khởi tạo cho các ngoại vi hoạt động như LCD, cảm biến, mạch điều khiển…Cuối cùng là khởi tạo các biến lưu trữ dữ liệu để hoàn thành việc khởi tạo ban đầu.

+) Sau đó chương trình sẽ đi vào phần thực thi chương trình chính là một vòng lập vô hạn. Trong chương trình chính, bộ xử lí trung tâm sẽ liên tục 2 công việc : một đọc giá trị từ cảm biến sau đó thực hiện các phép đo và tính toán để tìm được các giá trị thời gian, vận tốc ,quãng đường; hai là nhận các tín hiệu biển báo giao thông được gửi về từ khối xử lí hình ảnh.

+)Sau khi đã có được dữ liệu thì sẽ tiếp giao tiếp với người dùng thông qua LCD và sẽ bật loa cảnh báo nếu tốc độ vượt quá giới hạn và tắt loa khi đúng tốc độ cho phép.

-Chương trình nhận dạng ảnh và gửi dữ liệu sang chương trình chính :

* Cấu hình của Convolution Neural Network dùng cho quá trình huấn luyện model :

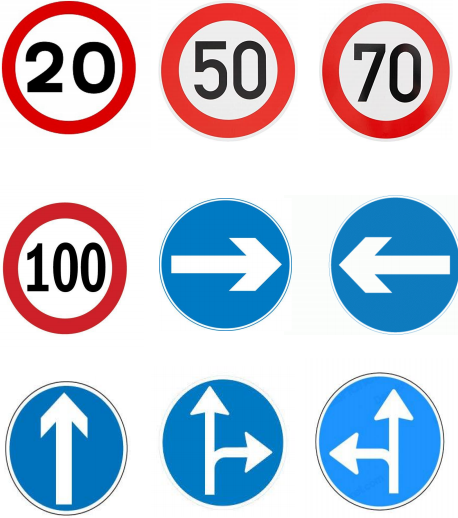
|  |  |
| --- | --- |
| Input shape | 32x32x3 |
| Convolution layer | Kernel 5x5x3 , stride = 1 ,no padding, number of filter = 60, Activation='relu'. |
| Convolution layer | Kernel 5x5x3 , stride = 1 ,no padding, number of filter = 60, Activation='relu'. |
| Maxpooling layer 1 | Pooling size 2x2,stride = 2,padding = “same”. |
| Convolution layer | Kernel 3x3x3 , stride = 1 ,no padding, number of filter = 30, Activation='relu'. |
| Convolution layer | Kernel 3x3x3 , stride = 1 ,no padding, number of filter = 30, Activation='relu'. |
| Maxpooling layer 2 | Pooling size 2x2,stride = 2,padding = “same”. |
| Drop out | P = 0.5 |
| Flatten output |  |
| Fully connected 1 | Activation='relu'. |
| Drop out | P = 0.5 |
| Fully connected 2 | Activation='softmax' |

Bảng ‑ Cấu hình Convolution Neural Network để huấn luyện

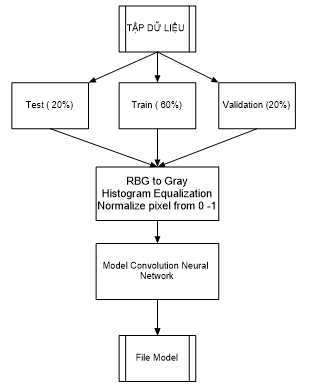
|  |  |
| --- | --- |
| ClassID | Biển báo |
| 0 | Speed Limit 20 km/h |
| 1 | Speed Limit 50 km/h |
| 2 | Speed Limit 70 km/h |
| 3 | Speed Limit 100 km/h |
| 4 | Turn Right Ahead |
| 5 | Turn lef Ahead |
| 6 | Only ahead |
| 7 | Straight or Right |
| 8 | Straight or left |

* Mục lục các biển báo :

Bảng ‑ Thứ tự biển báo



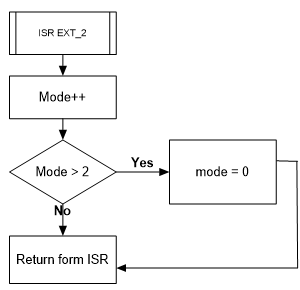
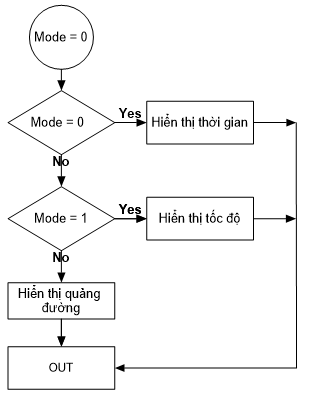
Hình ‑ Các mẫu biển báo giao thông sử dụng



Hình ‑ Lưu đồ nhận dạng ảnh

Hình ‑ Lưu đồ huấn luyện dữ liệu

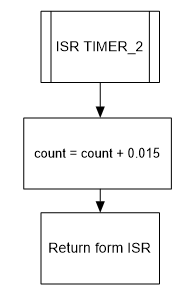
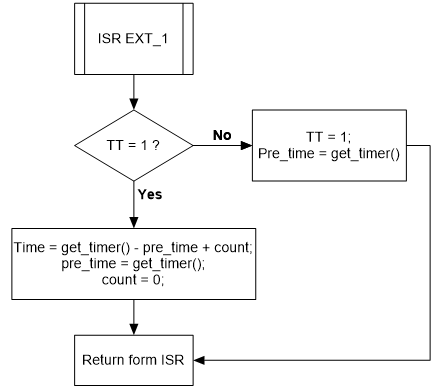


-Chương trình hiển thị :

Hình ‑ Lưu đồ chế độ hiển thị

Hình ‑ Lưu đồ ngắt ngoài khi nhắn nút

* Nguyên lý : Một nút nhấn được kết nối với chân ngắt ngoài 2 của vi điều khiển. Hệ thống bao gồm 3 chế chế hoạt động. Khởi tạo ban đầu có biến mode = 0 ( hiển thị thời gian ). Khi người dùng nhấn nút thì biến mode sẽ tăng lên 1 đơn vị và hiển thị chế độ tương ứng.

-Chương trình đọc cảm biến và đo tốc độ động cơ :

Hình ‑ Lưu đồ ngắt ngoài khi có từ trường đặt lên cảm biến

Hình ‑ Lưu đồ ngắt timer

* Nguyên lý : +) Đầu ra Digital ( chân D0 )của cảm biến Hall được nối với chân ngắt ngoài 1 của vi điều khiển. Ban đầu khi chưa có từ trường, chân D0 tích cực thấp ( mức 0 ), khi gặp từ trường là một nam châm gắn trên bánh xe chân D0 sẽ tích cực cao ( mức 1) từ đó kích hoạt ngắt ngoài.

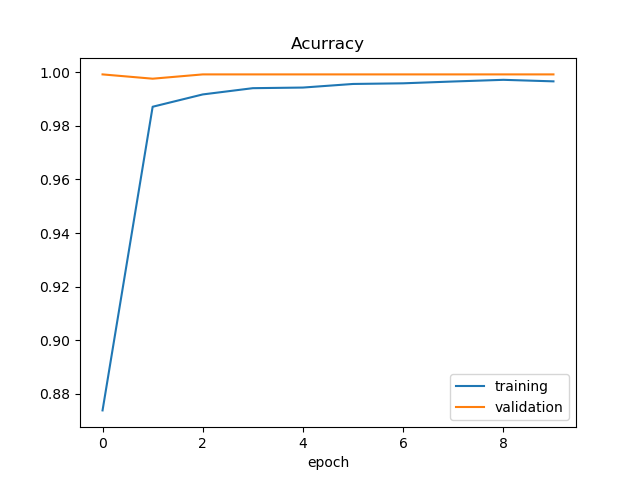
+) Trong quá trình đo thời gian giữa 2 lần xảy ra ngắt ngoài tương đương thời gian đi được 1 vòng của thiết bị. Ta sử dụng thêm ngắt timer để đo thời gian một cách chính xác hơn khi thanh ghi counter tràn.

# DỰ KIẾN KẾT QUẢ ĐẠT ĐƯỢC

## Kết quả sơ khởi đã đạt được :

Hệ thống dẫn dần hoàn thiện và đạt được một số kết quả :

* Huấn luyện thành công model CNN và nhận dạng được biển báo :

+)Kết quả mô hình khi hoàn thành huấn luyện :

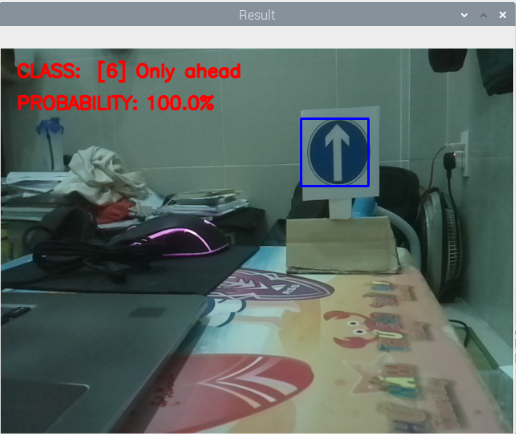
Hình ‑ Độ chính xác của model trong quá trình huấn luyện

Hình ‑ Sai số của model trong quá trình huấn luyện

+)Một số kết quả minh họa nhận dạng trên Raspberry :

Hình ‑ Kết quả minh họa biển báo 20km/h

Hình ‑ Kết quả minh họa biển báo 100km/h



Hình ‑ Kết quả minh họa biển báo “chỉ đi thẳng”

Hình ‑ Kết quả minh họa biển báo 20km/h

* Đo được thời gian quay của bánh xe từ đó tính được tốc độ.
* Thiết kế thành công bộ xử lí trung tâm điều khiển các khối chức năng.
* Hệ thống hoạt động ổn định.

## Kết quả dự kiến đạt được

Hệ thống sẽ phát triển thêm các chức năng thông minh sử dụng trí tuệ nhân tạo như cảnh báo nếu xe có chuyển động bất thường của các xe phía sau, phát hiện ngủ gật..Tích hợp thêm các cảm biến để hoàn thiện hệ thống hơn như cảnh báo nồn độ cồn, đo nhiệt độ xung quanh, gửi vị trí của xe về cho điện thoại thông qua module GPS và module Sim…

Đối với các hướng phát triển này và những kết quả đã thu được từ phần đề cương luận văn. Đến luận văn, dựa vào tính khả năng và tính khả thi của hệ thống sẽ hoàn thiện 100% hướng phát triển.

# KẾ HOẠCH THỰC HIỆN:

Với những yêu cầu đặt ra ban đầu. Tiến độ luận văn hiện tại đã hoàn thành được 50% trong đó bao gồm :

* Huấn luyện thành công model CNN, nhận dạng thành công biển báo trên kit Raspberry.
* Đo được thời gian quay của bánh xe, tính được tốc độ.
* Cảnh báo thành công khi vươt quá tốc độ cho phép.
* Đáp ứng đầy đủ các chế độ hiển thị
* Thiết kế thành công bộ xử lí trung tâm

Những nhiệm vụ còn lại của hệ thống bao gồm :

* Tìm hiểu giải thuật nhận dạng khuôn mặt, thuật toán cảnh báo ngủ báo gật.
* Tìm hiểu về các module SIM, GPS, nhiệt độ để tích hợp vào hệ thống.
* Tìm hiểu cách thức để cảnh báo khi phát hiện triển động bất ngờ của xe phía sau.
* Tích hợp lại với bộ xư lí trung tâm.
* Thiết kế bộ nguồn cho toàn hệ thống.

Thời gian thực hiện luận văn: 4 tháng

Kế hoạch:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nội dung | Tháng 2 | Tháng 3 | Tháng 4 | Tháng 5 |
| 1.Thuật toán nhận dạng khuôn mặt. | X |  |  |  |
| 2.Thuật toán cảnh báo ngủ gật | X | X |  |  |
| 3.Tìm hiểu về module Sim,GPS… |  | X |  |  |
| 4.Tìm hiểu chuẩn động bất ngờ của xe phía sau. |  |  | X |  |
| 5.Thiết kế bộ nguồn |  |  |  | X |
| 6.Hoàn thành báo cáo. |  |  |  | X |

Bảng ‑ Kế hoạch luận văn

# 

# TÀI LIỆU THAM KHẢO

1. <https://zingnews.vn/thi-truong-xe-may-viet-thay-doi-the-nao-sau-10-nam-post1032726.html>
2. <https://webvision.vn/giai-phap-cho-thuc-trang-giao-thong-viet-nam-hien-nay/>
3. <https://vamm.org.vn/thu-moi-tham-gia-nghien-cuu-khoa-hoc-nam-2017-voi-chu-de-vai-tro-cua-xe-may-trong-hien-tai-va-tuong-lai-tai-viet-nam/>
4. <https://irender.vn/y-nghia-va-vai-tro-cua-deep-learning/>
5. <https://viblo.asia/p/tensorflow-cho-nguoi-moi-bat-dau-eW65GwNx5DO>
6. Reference manual “STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced Arm®-based 32-bit MCUs”
7. <https://khuenguyencreator.com/tong-quan-ve-vi-xu-ly-arm-cortex-m3-m4-m7-m9/>
8. <http://kdientu.duytan.edu.vn/media/109456/gioi-thieu-giao-thuc-i2c.pdf>
9. <https://cmcdistribution.com.vn/kien-thuc-cnc/giao-tiep-uart-la-gi-ung-dung-cua-uart-trong-cuoc-song/>
10. http://dammedientu.vn/hieu-ung-hall-gioi-thieu-cam-bien-hall-va-ung-dung/
11. <https://d2l.aivivn.com/chapter_convolutional-neural-networks/lenet_vn.html>
12. <https://dlapplications.github.io/2018-07-06-CNN/>