TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**MÔN HỌC: XỬ LÝ ẢNH SỐ (IT701400)**

**BÁO CÁO CUỐI KỲ**

**ỨNG DỤNG HỖ TRỢ NGƯỜI KHIẾM THỊ NHẬN DIỆN CÁC ĐỐI TƯỢNG TRONG ĐỜI SỐNG HẰNG NGÀY**

*Người hướng dẫn*: **TS PHẠM VĂN HUY**

*Người thực hiện*: **NGUYỄN CAO KỲ - 52200056**

**TRẦN BÌNH KHANG - 52200115**

Lớp **: 24BD848010101**

Khoá **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2025**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**MÔN HỌC: XỬ LÝ ẢNH SỐ (IT701400)**

**BÁO CÁO CUỐI KỲ**

**ỨNG DỤNG HỖ TRỢ NGƯỜI KHIẾM THỊ NHẬN DIỆN CÁC ĐỐI TƯỢNG TRONG ĐỜI SỐNG HẰNG NGÀY**

*Người hướng dẫn*: **TS PHẠM VĂN HUY**

*Người thực hiện*: **NGUYỄN CAO KỲ - 52200056**

**TRẦN BÌNH KHANG - 52200115**

Lớp **: 24BD848010101**

Khoá **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2025**

LỜI CẢM ƠN

Đầu tiên, với lòng biết ơn chân thành và sâu sắc nhất, em muốn bày tỏ lòng biết ơn đến thầy TS. Phạm Văn Huy, người đã hỗ trợ em trong việc hoàn thành bài báo cáo cuối kỳ môn học ‘Xử lý ảnh số’. Thầy đã dành thời gian hướng dẫn và trả lời những thắc mắc của em với sự kiên nhẫn và tận tình. Nhờ đó, em đã tự tin hơn và hoàn thành bài báo cáo một cách hiệu quả.

Sự hướng dẫn của thầy đã giúp em hiểu rõ hơn về nội dung môn học và thậm chí còn cải thiện kỹ năng lập trình và tư duy logic. Em rất hạnh phúc và tự hào vì có cơ hội học hỏi từ thầy. Bài báo cáo đã mất khoảng 2 tuần để hoàn thành, và mặc dù có những hạn chế và bỡ ngỡ ban đầu, em mong nhận được ý kiến đóng góp quý báu từ thầy để nâng cao kiến thức của mình và khắc phục những thiếu sót.

Em cảm ơn thầy một lần nữa vì sự giúp đỡ và hướng dẫn tận tình. Hy vọng trong tương lai, em sẽ tiếp tục có cơ hội được học và hướng dẫn bởi thầy!

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của TS. Phạm Văn Huy. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 22 tháng 01 năm 2025*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Nguyễn Cao Kỳ*

*Trần Bình Khang*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Đề tài tập trung nghiên cứu và phát triển hệ thống nhận diện đối tượng dựa trên YOLO, tích hợp đo khoảng cách chính xác thông qua hiệu chỉnh tiêu cự. Hệ thống hỗ trợ người khiếm thị bằng cách xác định các vật thể trong môi trường xung quanh. Thông dịch tên đối tượng sang tiếng Việt và thông báo bằng âm thanh để cảnh báo vị trí và khoảng cách.

Công nghệ được sử dụng gồm YOLO V3 cho nhận diện nhanh và chính xác, thuật toán xử lý ảnh để đo khoảng cách, cùng cơ chế tổng hợp giọng nói tự nhiên nhằm tạo trải nghiệm tương tác trực quan. Hệ thống góp phần nâng cao khả năng nhận thức môi trường của người khiếm thị, giúp họ tránh vật cản và định hướng hiệu quả hơn.

Mặc dù đạt được nhiều kết quả khả quan, hệ thống còn gặp một số thách thức như tối ưu hóa độ chính xác trong điều kiện ánh sáng yếu, tốc độ xử lý thời gian thực và tích hợp thiết bị. Việc cải tiến thuật toán, mở rộng dữ liệu huấn luyện, và ứng dụng các công nghệ AI tiên tiến là hướng phát triển tương lai.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc15162)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc12041)

[TÓM TẮT iv](#_Toc7592)

[MỤC LỤC 1](#_Toc9813)

[CHƯƠNG 1 – MỞ ĐẦU 4](#_Toc13986)

[CHƯƠNG 2 – PHƯƠNG PHÁP GIẢI QUYẾT 5](#_Toc20206)

[2.1 Quy trình xử lý ảnh số 5](#_Toc4577)

[2.2 Nhận diện đối tượng 7](#_Toc12083)

[2.3 Đo khoảng cách với các vật thể 8](#_Toc18345)

[2.4 Xác Định Vị Trí Đối Tượng Trong Không Gian 8](#_Toc1076)

[2.5 Chuyển Đổi Thông Tin Thành Âm Thanh 9](#_Toc30526)

[CHƯƠNG 3 – CÔNG NGHỆ THỰC HIỆN 10](#_Toc13515)

[3.1 YOLOv3 (You Only Look Once version 3) 10](#_Toc17939)

[3.2 Mạng Nơ-ron Tích Chập (CNN - Convolutional Neural Networks) 11](#_Toc19100)

[3.3 Hàm Mất Mát (Loss Function) 11](#_Toc4433)

[3.4 Non-Maximum Suppression (NMS) 11](#_Toc2040)

[3.5 Text-to-Speech (TTS) 12](#_Toc11425)

[3.6 Cảm Biến Khoảng Cách (Distance Sensors) 12](#_Toc10744)

[3.7 Camera và Các Thiết Bị Đầu Vào 12](#_Toc8301)

[3.8 Xử Lý Hình Ảnh và Tiền Xử Lý 13](#_Toc8267)

[CHƯƠNG 4 – THỰC NGHIỆM 14](#_Toc10989)

[4.1 Xử lý ảnh và Nhận diện Đối tượng 14](#_Toc4230)

[4.2 Theo dõi và Tính Khoảng Cách 15](#_Toc8880)

[4.3 Ánh xạ Tên Đối Tượng Sang Tiếng Việt 16](#_Toc26880)

[4.4 Phát Âm Thanh Thông Báo 16](#_Toc3733)

[4.5 Hiển Thị Kết Quả và Quản Lý Tình Trạng Đối Tượng 16](#_Toc10108)

[4.6 Thực Thi và Kiểm Tra Kết Quả 17](#_Toc9793)

[CHƯƠNG 5 – KẾT LUẬN 18](#_Toc23088)

[TÀI LIỆU THAM KHẢO 19](#_Toc18331)

**DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ**

**DANH MỤC HÌNH**

[Hình 1 - BlobFromImage 5](#_Toc2030)

[Hình 2 - Bounding Box 7](#_Toc13654)

[Hình 3 - Text-to-Speech 9](#_Toc32631)

**DANH MỤC BẢNG BIỂU**

[Bảng 1 - Yêu cầu về mặt phần cứng 13](#_Toc31216)

CHƯƠNG 1 – MỞ ĐẦU

Trong cuộc sống hiện đại, công nghệ đã trở thành cầu nối giúp con người vượt qua những rào cản mà trước đây tưởng chừng không thể vượt qua. Đối với người khiếm thị, việc nhận biết và tương tác với thế giới xung quanh luôn là một thách thức lớn. Các công cụ hỗ trợ truyền thống, như gậy dò đường hay các thiết bị đọc chữ nổi, mặc dù đã đóng góp đáng kể, nhưng vẫn còn hạn chế trong việc cung cấp thông tin trực quan về môi trường xung quanh một cách toàn diện và tức thời.

Sự phát triển vượt bậc của lĩnh vực xử lý ảnh số, đặc biệt là các kỹ thuật nhận diện đối tượng, đã mở ra những hướng đi mới trong việc xây dựng các ứng dụng hỗ trợ người khiếm thị. Các mô hình trí tuệ nhân tạo như YOLO (You Only Look Once) không chỉ mang lại độ chính xác cao trong nhận diện đối tượng mà còn cho phép triển khai trên các thiết bị di động với hiệu suất ấn tượng. Kết hợp với khả năng đo khoảng cách dựa trên tiêu cự được hiệu chỉnh, công nghệ này có tiềm năng trở thành công cụ hữu ích, giúp người khiếm thị không chỉ nhận biết mà còn tương tác hiệu quả hơn với môi trường sống xung quanh.

Ngoài ra, việc tích hợp công nghệ dịch ngôn ngữ tự nhiên và phát thông báo bằng âm thanh tạo nên sự tương tác gần gũi và thân thiện hơn với người dùng, giúp họ cảm nhận thông tin không chỉ qua thị giác mà còn qua thính giác. Điều này đặc biệt quan trọng trong việc xây dựng các hệ thống hỗ trợ toàn diện, mang lại trải nghiệm dễ sử dụng và nâng cao chất lượng cuộc sống cho người khiếm thị.

Với mục tiêu góp phần giải quyết những khó khăn mà người khiếm thị đang gặp phải, đề tài này tập trung vào việc nghiên cứu và phát triển một hệ thống nhận diện đối tượng, đo khoảng cách và cung cấp thông báo âm thanh thời gian thực. Đây không chỉ là một hướng đi mang tính ứng dụng cao mà còn là một bước tiến trong việc ứng dụng công nghệ hiện đại để cải thiện cuộc sống của những người yếu thế trong xã hội.

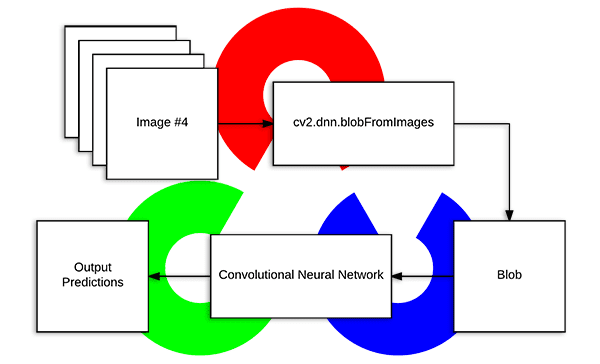
CHƯƠNG 2 – PHƯƠNG PHÁP GIẢI QUYẾT

2.1 Quy trình xử lý ảnh số

2.1.1 - Tiền xử lý ảnh

Ảnh đầu vào từ camera hoặc video được chuyển đổi sang dạng blob bằng hàm *cv2.dnn.blobFromImage*. Đây là quá trình chuẩn hóa giá trị điểm ảnh (pixel) về phạm vi [0, 1] và điều chỉnh kích thước ảnh về tiêu chuẩn (416x416) để phục vụ cho các bước xử lý tiếp theo.

Hàm *cv2.dnn.blobFromImage* trong OpenCV được sử dụng để chuyển đổi một bức ảnh thành một "blob" (dữ liệu đầu vào cho mạng nơ-ron), phục vụ cho các mô hình học sâu (deep learning), đặc biệt là khi làm việc với mô hình nhận diện đối tượng, phân loại ảnh, v.v.

Trong quá trình chuyển đổi, ảnh có thể được thay đổi kích thước, chuẩn hóa giá trị pixel, trừ đi giá trị trung bình và chuyển đổi giữa các không gian màu, giúp tạo ra một dữ liệu đầu vào chuẩn xác cho mô hình.

Hình 1 - BlobFromImage

2.1.2 - Xử lý vùng phát hiện

Các đối tượng được phát hiện sẽ được xác định vị trí (tọa độ và kích thước của bounding box). Các thông tin này bao gồm:

* Tọa độ góc trên trái: (x, y).
* Chiều rộng và chiều cao: (w, h).
* Tâm của mỗi đối tượng được tính toán từ tọa độ (center\_x, center\_y).
* Phân vùng không gian: Khung hình được chia thành các vùng không gian cụ thể như: bên trái, bên phải, chính giữa, phía trên, phía dưới, và các góc.
* Tâm của mỗi đối tượng được dùng để xác định đối tượng thuộc khu vực nào trong khung hình.

2.1.3 - Tính toán khoảng cách

Dựa trên chiều cao thực tế của đối tượng (từ một bảng giá trị đã được định nghĩa trước) và chiều cao của bounding box (tính từ kết quả phát hiện), chương trình tính toán khoảng cách từ đối tượng đến camera dựa trên công thức. *Khoảng cách này được chuyển đổi từ cm => m để dễ hiểu hơn.*

2.1.4 - Theo dõi đối tượng

* Sử dụng tọa độ tâm và khoảng cách, các đối tượng được theo dõi qua từng khung hình bằng cách so sánh khoảng cách giữa các đối tượng đã phát hiện và các đối tượng đang theo dõi.
* Nếu khoảng cách giữa tâm của một đối tượng mới và một đối tượng đã theo dõi nhỏ hơn ngưỡng định trước (DISTANCE\_THRESHOLD), đối tượng đó được coi là cùng một đối tượng.

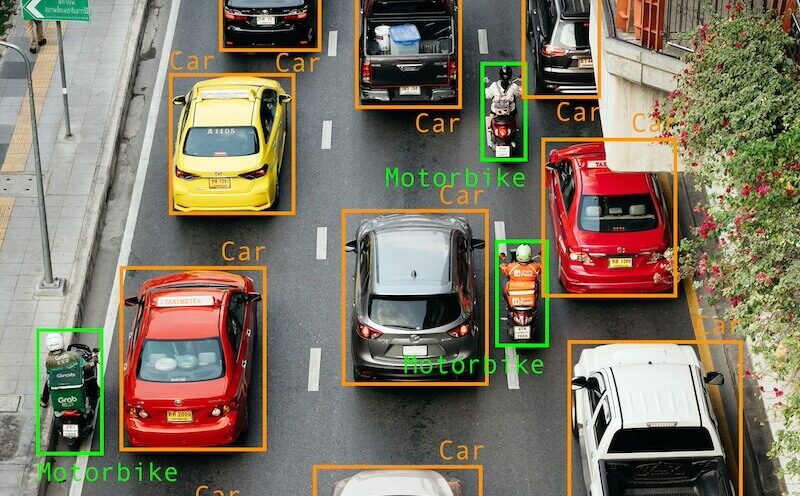
2.1.5 - Quản lý đối tượng mất dấu

Nếu một đối tượng không xuất hiện trong một số khung hình liên tiếp (dựa trên ngưỡng FRAMES\_TO\_FORGET), chương trình sẽ loại bỏ đối tượng đó khỏi danh sách theo dõi.

2.1.6 - Hiển thị và chú thích ảnh

Với mỗi đối tượng, chương trình vẽ một khung (bounding box) bao quanh đối tượng, cùng với thông tin chi tiết như:

* Tên đối tượng.
* Khoảng cách ước tính từ camera đến đối tượng.
* Vị trí đối tượng trong khung hình (ví dụ: "phía trên bên trái", "bên phải",...).

Các thông tin này được vẽ trực tiếp lên khung hình hiển thị bằng OpenCV (*cv2.rectangle* và *cv2.putText*).

Hình 2 - Bounding Box

2.2 Nhận diện đối tượng

Phần quan trọng nhất cũng như là cốt lỗi của vấn đề này đó là xác định được đối tượng vật thể việc xác định cụ thể vật thể không đơn giản như xác định số, xác định đường hay hình dạng cơ bản mà phải phân tích sâu về các hình dạng phức tạp khác cụ thể như con người, điện thoại,vv. Do đó phải xử dụng một mô hình học xâu để có thể thực hiện được vấn đề trên.

Các bước thực hiện:

* Thu thập dữ liệu hình ảnh từ camera tích hợp.
* Xử lý dữ liệu đầu vào qua mô hình được chọn để xác định loại đối tượng và tọa độ của chúng trong khung hình.
* Phân loại đối tượng thành các nhóm như: người, xe cộ, biển báo, động vật, v.v.
* Kết quả nhận diện sau đó sẽ được lưu trữ và xử lý nhằm phục vụ các bước tiếp theo, bao gồm đo khoảng cách và thông báo bằng âm thanh.

2.3 Đo khoảng cách với các vật thể

Hệ thống áp dụng phương pháp tính khoảng cách giữa người dùng và các đối tượng nhận diện được, dựa trên các yếu tố như chiều cao thực tế của đối tượng và thông số tiêu cự của camera. Công thức tính khoảng cách được tối ưu hóa để đảm bảo độ chính xác cao trong các môi trường thực tế.

Ví dụ, với một đối tượng như chiếc xe có chiều cao được biết trước, hệ thống sẽ tính toán khoảng cách dựa trên:

* Kích thước của đối tượng trong khung hình.
* Thông số hiệu chỉnh tiêu cự của camera (đã được xác định trước qua quá trình hiệu chuẩn).
* Kết quả này được sử dụng để đưa ra thông báo âm thanh, chẳng hạn: "Có một chiếc xe cách bạn 5 mét ở phía bên trái".

2.4 Xác Định Vị Trí Đối Tượng Trong Không Gian

Không gian xung quanh người dùng được chia thành các khu vực định hướng như trái, phải, trước, và sau. Khi một đối tượng được phát hiện, hệ thống sẽ phân tích vị trí của đối tượng trong khung hình và xác định khu vực tương ứng trong không gian thực.

Việc phân vùng không gian này giúp cung cấp thông tin định hướng trực quan hơn, ví dụ:

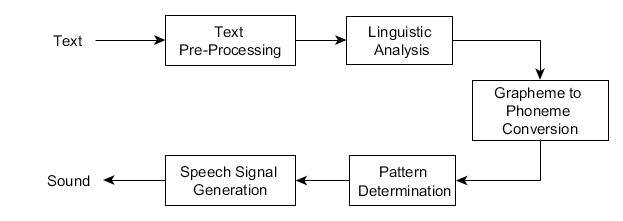
"Có một chiếc xe ở phía trước bên phải."

"Có một người ở phía sau bên trái."

*=> Thông tin định hướng này hỗ trợ người khiếm thị di chuyển an toàn và chủ động hơn, giảm thiểu nguy cơ va chạm.*

2.5 Chuyển Đổi Thông Tin Thành Âm Thanh

Thông tin nhận diện và phân tích được chuyển đổi thành âm thanh thông qua công nghệ Text-to-Speech (TTS). Trong hệ thống này, Đây là một bước cũng vô cùng quan trọng vì đây là phần tương tác trực tiếp đối với người dùng nên cần tốc độ đọc chuẩn, âm thanh chuẩn của từng quốc gia và khả năng dịch thuật đúng đắn.

Việc cung cấp thông tin liên tục giúp người khiếm thị luôn nhận thức được trạng thái môi trường xung quanh.

Hình 3 - Text-to-Speech

CHƯƠNG 3 – CÔNG NGHỆ THỰC HIỆN

Để xây dựng một hệ thống hỗ trợ người khiếm thị, việc tích hợp các công nghệ tiên tiến là rất quan trọng. Một trong những công nghệ nền tảng trong ứng dụng này là YOLOv3 (You Only Look Once version 3), nhưng ngoài YOLOv3, hệ thống còn sử dụng một loạt các công nghệ và thành phần khác để đảm bảo tính hiệu quả và chính xác của hệ thống. Mỗi công nghệ và thành phần đều đóng một vai trò quan trọng trong việc nhận diện đối tượng, đo khoảng cách, và cung cấp phản hồi chính xác, giúp người khiếm thị di chuyển một cách an toàn hơn trong môi trường thực tế. Dưới đây là một cái nhìn tổng quan về các công nghệ và thành phần cơ bản sử dụng trong ứng dụng này.

3.1 YOLOv3 (You Only Look Once version 3)

YOLOv3 là một phiên bản cải tiến của mô hình học sâu YOLO, nổi bật trong việc nhận diện đối tượng trong ảnh và video. Đây là một phương pháp mạnh mẽ, cho phép nhận diện và phân loại nhiều đối tượng trong một bức ảnh chỉ trong một lần xử lý duy nhất. So với các phiên bản trước, YOLOv3 mang lại độ chính xác cao hơn, đặc biệt với các đối tượng nhỏ hoặc ở những khung cảnh phức tạp.

YOLOv3 chia hình ảnh đầu vào thành các lưới và trong mỗi ô, mô hình dự đoán một số lượng bounding boxes nhất định, đi kèm với độ tin cậy (confidence score) và xác suất phân loại từng đối tượng. Điểm đặc biệt của YOLOv3 nằm ở khả năng:

Phát hiện đối tượng trên ba thang đo (scale), giúp tăng độ chính xác khi nhận diện các đối tượng có kích thước khác nhau.

Sử dụng kiến trúc Darknet-53 làm mạng nơ-ron nền tảng (backbone), cho phép trích xuất đặc trưng mạnh mẽ và giảm thiểu mất mát thông tin trong quá trình xử lý.

Áp dụng khái niệm residual connections, cải thiện khả năng học tập của mô hình và giúp việc huấn luyện nhanh hơn, ổn định hơn.

Nhờ vào kiến trúc mạng nơ-ron tích chập (CNN) tối ưu, YOLOv3 tự động trích xuất các đặc trưng quan trọng từ hình ảnh để dự đoán vị trí và loại đối tượng. Cách sử dụng một mô hình duy nhất cho toàn bộ quá trình nhận diện giúp YOLOv3 trở thành một công cụ nhanh và hiệu quả. Tốc độ xử lý của YOLOv3 cho phép hệ thống cung cấp phản hồi thời gian thực, giúp người khiếm thị nhanh chóng nhận biết các đối tượng trong môi trường xung quanh.

3.2 Mạng Nơ-ron Tích Chập (CNN - Convolutional Neural Networks)

Mạng nơ-ron tích chập (CNN) là thành phần cốt lõi trong mô hình YOLO. CNN có khả năng học và nhận diện các đặc trưng không gian trong ảnh, từ các hình dạng đơn giản như cạnh và góc đến các đặc trưng phức tạp hơn như khuôn mặt, động vật, hay các vật thể trong môi trường. Mạng này được xây dựng từ các lớp chập (convolutional layers) và lớp pooling, giúp giảm thiểu số lượng đặc trưng mà vẫn đảm bảo tính chính xác trong việc phân loại và nhận diện đối tượng.

Trong YOLO, CNN sẽ giúp trích xuất các đặc trưng từ ảnh, cho phép mô hình nhận diện chính xác các đối tượng ngay cả khi chúng xuất hiện trong những điều kiện phức tạp hoặc thay đổi.

3.3 Hàm Mất Mát (Loss Function)

Hàm mất mát là một phần quan trọng trong quá trình huấn luyện của YOLO, giúp mô hình học cách điều chỉnh các tham số để tối ưu hóa kết quả dự đoán. YOLO sử dụng một hàm mất mát kết hợp, bao gồm các thành phần chính như:

* Mất mát về tọa độ (bounding box loss): Đo lường sự khác biệt giữa các tọa độ của bounding box dự đoán và tọa độ thực tế.
* Mất mát về xác suất lớp (class probability loss): Đánh giá độ chính xác của việc phân loại đối tượng.
* Mất mát về độ tin cậy (confidence loss): Đo lường khả năng của mô hình trong việc nhận diện đối tượng và đánh giá sự tồn tại của chúng trong ảnh.

Quá trình huấn luyện sẽ giúp mô hình tối ưu hóa các tham số để giảm thiểu tổng các thành phần của hàm mất mát này, giúp cho kết quả nhận diện ngày càng chính xác hơn.

3.4 Non-Maximum Suppression (NMS)

Trong quá trình nhận diện đối tượng, YOLO có thể đưa ra nhiều bounding box cho một đối tượng do việc chia ảnh thành nhiều lưới. Tuy nhiên, điều này có thể dẫn đến việc xuất hiện nhiều dự đoán chồng lấn, gây ra kết quả dư thừa. Để khắc phục vấn đề này, kỹ thuật Non-Maximum Suppression (NMS) được sử dụng.

NMS hoạt động bằng cách loại bỏ các bounding box có độ tin cậy thấp và giữ lại bounding box có độ tin cậy cao nhất, từ đó giảm thiểu sự trùng lặp và cải thiện tính chính xác của các dự đoán. Quá trình này giúp đảm bảo rằng mỗi đối tượng chỉ được nhận diện một lần duy nhất, tránh gây nhiễu cho hệ thống.

3.5 Text-to-Speech (TTS)

Một trong những tính năng quan trọng trong hệ thống hỗ trợ người khiếm thị là khả năng chuyển đổi văn bản thành giọng nói thông qua công nghệ Text-to-Speech (TTS). Sau khi YOLO nhận diện được các đối tượng và đo khoảng cách, hệ thống TTS sẽ chuyển thông tin này thành lời nói, giúp người khiếm thị nhận diện môi trường xung quanh thông qua âm thanh.

Hệ thống TTS hiện đại sử dụng các mô hình học sâu như WaveNet để tạo ra giọng nói tự nhiên, dễ hiểu và gần gũi, phù hợp với ngữ cảnh của từng tình huống. Khi một đối tượng được phát hiện, hệ thống có thể phát ra thông báo như "Có một chiếc xe cách bạn 2 mét" để người khiếm thị có thể nhận diện và di chuyển một cách an toàn.

3.6 Cảm Biến Khoảng Cách (Distance Sensors)

Để tăng cường khả năng đo lường khoảng cách và hỗ trợ người khiếm thị, hệ thống còn sử dụng các cảm biến khoảng cách, như cảm biến siêu âm hoặc LiDAR. Các cảm biến này có thể đo khoảng cách từ người dùng đến các đối tượng xung quanh, giúp cung cấp thông tin về mức độ nguy hiểm của môi trường.

Các cảm biến này hoạt động bằng cách phát tín hiệu siêu âm hoặc laser và đo thời gian phản hồi từ các vật thể trong phạm vi. Kết hợp với thông tin từ YOLO, hệ thống có thể cung cấp những cảnh báo chính xác về các đối tượng gần người khiếm thị.

3.7 Camera và Các Thiết Bị Đầu Vào

Để mô hình YOLO có thể nhận diện đối tượng, hệ thống cần có một thiết bị đầu vào như camera, giúp thu thập hình ảnh hoặc video từ môi trường. Các camera này có thể là camera đơn giản, nhỏ gọn, hoặc các loại camera với độ phân giải cao để cung cấp thông tin chi tiết về môi trường xung quanh.

| Thiết bị / Thành phần | Thông số tối thiểu | Tỉ lệ nhận diện chính xác |
| --- | --- | --- |
| Camera | Camera độ phân giải 720p, 30fps, góc nhìn 60°, hỗ trợ kết nối qua USB | 70% - 80% |
| CPU | ARM Cortex-A7 hoặc tương đương (1GHz) | 75% - 85% |
| RAM | 1GB hoặc hơn | 75% - 85% |
| Bộ nhớ | 8GB bộ nhớ flash |  |
| Kết nối | WiFi 2.4GHz |  |
| Âm thanh | Có loa âm lượng đủ lớn hoặc có hỗ trợ jack 3.5mm |  |

Bảng 1 - Yêu cầu về mặt phần cứng

3.8 Xử Lý Hình Ảnh và Tiền Xử Lý

Trước khi hình ảnh được đưa vào YOLO để phân tích, chúng cần được tiền xử lý. Các bước tiền xử lý như chuẩn hóa ảnh, quay, cắt xén ảnh, hoặc tăng cường ảnh (augmentation) sẽ giúp cải thiện khả năng nhận diện của mô hình. Tiền xử lý giúp hệ thống YOLO hoạt động hiệu quả hơn trong các tình huống thực tế, nơi điều kiện ánh sáng hoặc góc nhìn có thể thay đổi.

CHƯƠNG 4 – THỰC NGHIỆM

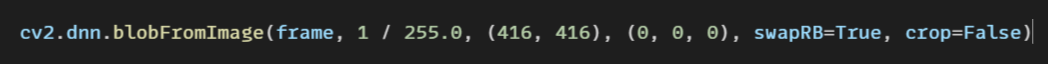
Trong phần thực nghiệm này, tiến hành sử dụng mô hình YOLO (You Only Look Once) để phát hiện đối tượng trong video hoặc hình ảnh được truyền vào. Quá trình nhận diện đối tượng sẽ được thực hiện từ việc xử lý hình ảnh, phát hiện và theo dõi các đối tượng cho đến việc phát âm thanh thông báo cho người dùng về các đối tượng đã được phát hiện.

Đầu tiên, sử dụng mô hình YOLO v3 với các tệp trọng số (weights), cấu hình (config), và danh sách các lớp (classes) đối tượng. Các tệp này được tải lên và sử dụng trong hàm cv2.dnn.readNet để tải mô hình YOLO vào OpenCV, cho phép ta phát hiện đối tượng trong các khung hình video.

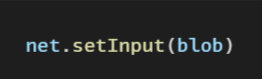
4.1 Xử lý ảnh và Nhận diện Đối tượng

Trong mỗi khung hình của video, hàm *cv2.dnn.blobFromImage* được sử dụng để chuyển đổi hình ảnh đầu vào thành một blob có thể được YOLO xử lý. Blob này có kích thước chuẩn là (416, 416) và được chuẩn hóa với tỷ lệ 1 / 255.0 để giảm dần giá trị pixel về khoảng [0,1]. Tiếp theo, phương thức *net.setInput(blob)* cung cấp input cho mô hình YOLO, và *net.forward(output\_layers)* được gọi để tính toán kết quả đầu ra từ mô hình.

Các hàm chính trong việc phát hiện đối tượng:



Tiền xử lý hình ảnh đầu vào thành một blob.



Cung cấp blob như là input cho mô hình.

{3518A52D-830C-46FE-B6C4-EB3F24F18FE7}

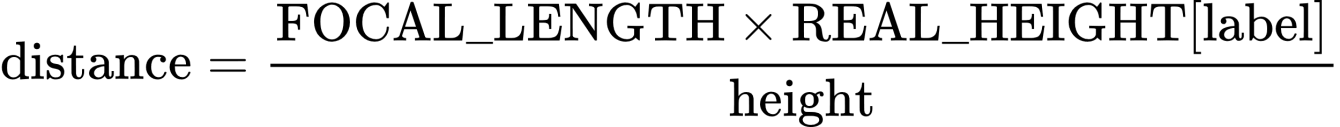
Trả về các đầu ra từ mô hình, bao gồm các thông tin về tọa độ và lớp của đối tượng.

Sau khi có được kết quả từ YOLO, chúng ta kiểm tra các giá trị tin cậy (confidence) để xác định xem đối tượng nào được phát hiện với độ tin cậy cao hơn ngưỡng (0.5 trong trường hợp này). Nếu một đối tượng đạt yêu cầu, các thông tin như tọa độ, chiều cao, chiều rộng, và lớp đối tượng sẽ được trích xuất để tiến hành vẽ các hộp bao quanh đối tượng.

4.2 Theo dõi và Tính Khoảng Cách

Để theo dõi các đối tượng đã phát hiện trong các khung hình liên tiếp, chúng ta cần lưu trữ thông tin của từng đối tượng. Một dictionary có tên tracked\_objects được sử dụng để theo dõi các đối tượng, với mỗi đối tượng sẽ có một ID duy nhất được sinh ra từ hàm uuid.uuid4(). Thông tin lưu trữ bao gồm:

* label: Tên đối tượng (dịch sang tiếng Việt).
* center: Tọa độ trung tâm của hộp bao quanh đối tượng.
* distance: Khoảng cách từ camera đến đối tượng tính theo đơn vị mét.
* frames\_since\_seen: Số khung hình từ lần cuối đối tượng được phát hiện.
* Khoảng cách được tính toán bằng công thức sau:

​

* Trong đó:

+) FOCAL\_LENGTH: tiêu cự của camera đã được hiệu chỉnh.

+) REAL\_HEIGHT[label]: chiều cao thực tế của đối tượng (tùy theo loại đối tượng) và height là chiều cao của hộp bao quanh đối tượng trong khung hình.

Mỗi lần đối tượng được phát hiện, chúng ta tính toán khoảng cách và theo dõi xem đối tượng đó có di chuyển ra khỏi khung hình hay không. Nếu đối tượng không xuất hiện trong một số khung hình liên tiếp (số khung hình này được định nghĩa qua biến FRAMES\_TO\_FORGET), đối tượng đó sẽ bị xóa khỏi danh sách theo dõi.

4.3 Ánh xạ Tên Đối Tượng Sang Tiếng Việt

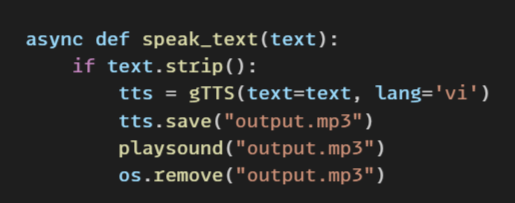
Để phục vụ cho mục đích hỗ trợ người khiếm thị, tên các đối tượng được ánh xạ từ tiếng Anh sang tiếng Việt. Một dictionary có tên *object\_name\_translation* được sử dụng để lưu trữ các tên đối tượng bằng tiếng Anh và tương ứng với tên gọi bằng tiếng Việt.

Ví dụ, đối tượng "person" sẽ được dịch thành "Con người", "car" thành "Xe hơi", và "dog" thành "Chó". Việc dịch này giúp người dùng có thể nhận diện dễ dàng hơn khi được thông báo bằng tiếng Việt.

4.4 Phát Âm Thanh Thông Báo

Khi phát hiện được đối tượng và tính được khoảng cách, chúng ta sẽ sử dụng thư viện gTTS (Google Text-to-Speech) để chuyển văn bản thành giọng nói. Hàm *speak\_text* thực hiện việc này một cách không đồng bộ. Văn bản được chuyển thành âm thanh thông qua hàm *gTTS(text=text, lang='vi'),* sau đó được lưu trữ tạm thời trong tệp MP3 và phát ra âm thanh bằng thư viện playsound.

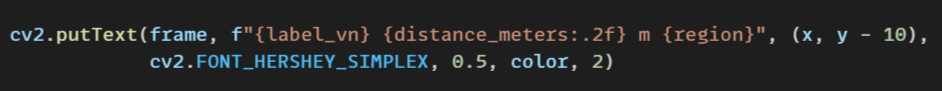
Dưới đây là mã nguồn thực hiện việc phát âm thanh:



Khi một đối tượng được phát hiện trong phạm vi xác định (bằng cách so sánh khoảng cách với ngưỡng nhất định), văn bản sẽ được tạo ra và gửi đến hàm speak\_text để phát âm thanh thông báo.

4.5 Hiển Thị Kết Quả và Quản Lý Tình Trạng Đối Tượng

Mỗi đối tượng sau khi được phát hiện và theo dõi sẽ được vẽ hộp bao quanh trên khung hình video bằng hàm *cv2.rectangle*. Thông tin về đối tượng, bao gồm tên (tiếng Việt), khoảng cách và khu vực trong khung hình, sẽ được hiển thị bằng hàm *cv2.putText.*



Ngoài ra, một phần quan trọng của hệ thống là quản lý các đối tượng bị mất tín hiệu trong vài khung hình liên tiếp. Để làm điều này, chúng ta sử dụng biến *frames\_since\_seen* để đếm số khung hình không có đối tượng xuất hiện, và nếu số này vượt quá ngưỡng FRAMES\_TO\_FORGET, đối tượng đó sẽ bị loại bỏ khỏi danh sách theo dõi.

4.6 Thực Thi và Kiểm Tra Kết Quả

Kết quả cuối cùng của hệ thống là việc phát hiện đối tượng trong video, theo dõi chúng trong thời gian thực, và cung cấp thông tin về chúng cho người dùng dưới dạng âm thanh. Hệ thống cũng có khả năng phân biệt các khu vực khác nhau trong khung hình (trái, phải, trên, dưới) và thông báo chính xác vị trí của đối tượng trong không gian.

Thông qua các hàm xử lý ảnh và âm thanh kết hợp với nhau, hệ thống có thể tự động nhận diện và thông báo về các đối tượng trong môi trường xung quanh, hỗ trợ đắc lực cho người khiếm thị trong việc nhận diện và phản ứng với các vật thể.

CHƯƠNG 5 – KẾT LUẬN

Trong khuôn khổ của đề tài, chúng tôi đã bước đầu hoàn thành nhiệm vụ đặt ra, bao gồm việc nghiên cứu các nguyên lý và phát triển một hệ thống nhận diện đối tượng kết hợp đo khoảng cách nhờ vào mô hình YOLO v3 và tiêu cự hiệu chỉnh. Hệ thống cũng đã tích hợp các tính năng bổ trợ như dịch tên đối tượng sang tiếng Việt và phát thông báo bằng âm thanh tự nhiên, đáp ứng các yêu cầu ban đầu trong việc hỗ trợ người khiếm thị.

Mặc dù đã đạt được những kết quả đáng khích lệ, hệ thống vẫn tồn tại một số hạn chế như sau:

* Thiếu vật tư chuyên dụng: Hệ thống chưa sử dụng depth camera, do đó độ chính xác trong đo khoảng cách có thể chưa đạt đến mức mong muốn. Việc trang bị thêm thiết bị chuyên dụng là điều cần thiết để tăng tính năng và độ đầy đủ.
* Hạn chế về nguồn cung cấp năng lượng: Hệ thống hiện tại phụ thuộc vào pin, nhưng chưa đảm bảo đủ năng lượng cho các điều kiện hoạt động dài hơn và cường độ cao. Việc nghiên cứu và đề xuất các giải pháp tối ưu năng lượng sẽ là một hướng tiếp cận trong tương lai.
* Thiếu các chức năng tương tác: Hiện tại, hệ thống chưa được trang bị các chức năng cho phép người dùng tương tác hai chiều với hệ thống như lựa chọn đối tượng để nhận diện hoặc cung cấp phản hồi. Việc phát triển thêm những tính năng này sẽ tăng tính ứng dụng thực tế.

Trên cơ sở những kết quả và hạn chế nêu trên, chúng tôi tin rằng đề tài đã đạt được những tiến bộ quan trọng và là nền tảng cho các nghiên cứu và phát triển tiếp theo. Trong tương lai, việc khắc phục các hạn chế nêu trên sẽ giúp hệ thống trở thành một công cụ đắc lực đối với người khiếm thị và nhiều lĩnh vực khác trong đời sống.

TÀI LIỆU THAM KHẢO

**Tiếng Việt**

1. https://phamdinhkhanh.github.io/2020/03/09/DarknetAlgorithm.html
2. https://123docz.net/trich-doan/3230358-mo-hinh-lstm.htm
3. https://huytranvan2010.github.io/YOLOv3-pretrained-model/
4. https://phamdinhkhanh.github.io/2020/03/09/Yolo-Object-Detection.html
5. https://phamdinhkhanh.github.io/2020/03/09/DarknetAlgorithm.html

**Tiếng Anh**

1. https://github.com/HimanchalChandra/Object-Detection-with-Voice-Feedback-YOLO-v3-and-gTTS
2. https://github.com/jasonyip184/yolo
3. https://developers.arcgis.com/python/latest/guide/yolov3-object-detector/
4. https://www.youtube.com/watch?v=WP4F6aK1Ft8