

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  
THÀNH PHỐ HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



TIỂU LUẬN CHUYÊN NGÀNH

# TÌM HIỂU HỌC SÂU CHO BÀI TOÁN PHÂN LOẠI ẢNH ỨNG THỰC VẬT

GVHD: TS. Nguyễn Thiên Bảo

SVTH:

*Phạm Quốc Việt* 17110254

*Nguyễn Hùng Duy* 17110112

*Đặng Huỳnh Hoàng Long* 17110173

Tp. Hồ Chí Minh, tháng 12 năm 2020

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  
THÀNH PHỐ HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



TIỂU LUẬN CHUYÊN NGÀNH

# TÌM HIỂU HỌC SÂU CHO BÀI TOÁN PHÂN LOẠI ẢNH UNG THƯ VÚ

GVHD: TS. Nguyễn Thiên Bảo

SVTH:

*Phạm Quốc Việt* 17110254

*Nguyễn Hùng Duy* 17110112

*Đặng Huỳnh Hoàng Long* 17110173

Tp. Hồ Chí Minh, tháng 12 năm 2020

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020

### **NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH**

**Họ và tên sinh viên:** Phạm Quốc Việt

MSSV: 17110254

Chuyên ngành: Công nghệ phần mềm

Lớp: 17110CLST2

**Họ và tên sinh viên:** Nguyễn Hùng Duy

MSSV: 17110112

Chuyên ngành: Công nghệ phần mềm

Lớp: 17110CLST2

**Họ và tên sinh viên:** Đặng Huỳnh Hoàng Long

MSSV: 17110173

Chuyên ngành: Công nghệ phần mềm

Lớp: 17110CLST1

**Giảng viên hướng dẫn:** T.S Nguyễn Thiên Bảo

Ngày nhận đề tài: 14/09/2020

Ngày nộp đề tài: 23/12/2020

1. Tên đề tài: **Tìm hiểu học sâu cho bài toán phân loại ảnh ung thư vú**
2. Các số liệu, tài liệu ban đầu: Data training từ GVHD và Google Image, tài liệu nghiên cứu Deep Learning, CNN từ GVHD và tự tìm kiếm.
3. Nội dung thực hiện đề tài:
  - Tìm hiểu bài toán ung thư vú, tìm hiểu học sâu.
  - Áp dụng học sâu vào giải quyết bài toán.
  - Xây dựng các mô hình phỏng đoán bệnh ung thư vú dựa trên hình ảnh, sử dụng CNN.
  - Xây dựng website và ứng dụng winform phân loại bệnh ung thư vú dựa trên ba mô hình có tỉ lệ cao nhất.
4. Sản phẩm: Một website và một ứng dụng Winform phân loại bệnh ung thư vú dựa trên 03 mô hình có tỉ lệ cao nhất.

TRƯỞNG NGÀNH

GIẢNG VIÊN HƯỚNG DẪN





## Độc lập – Tự do – Hạnh phúc

\*\*\*

**Họ và tên sinh viên:** Phạm Quốc Việt    MSSV: 17110254  
**Họ và tên sinh viên:** Nguyễn Hùng Duy                                        MSSV: 17110112  
**Họ và tên sinh viên:** Đặng Huỳnh Hoàng Long                              MSSV: 17110173  
**Chuyên ngành:** Công nghệ phần mềm  
**Tên đề tài:** Tìm hiểu học sâu cho bài toán phân loại ảnh ung thư vú  
**Họ và tên Giáo viên phản biện:** .....

## 6. Về nội dung đề tài & khối lượng thực hiện:

.....

.....

.....

## 7. Ưu điểm:

.....

## 8. Khuyết điểm:

.....

.....

9. Đề nghị cho bảo vệ hay không?

.....

10. Đánh giá loại:

10. Đánh giá loại:

11. Điểm: .....(Bằng chữ: .....

11. Điểm: .....(Bằng chữ: .....)

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020

Giáo viên phản biện

(Ký & ghi rõ họ tên)

## **LỜI CAM ĐOAN**

Nhóm xin cam đoan toàn bộ nội dung liên quan tới đề tài được trình bày trong tiểu luận là nhóm tự tìm hiểu và nghiên cứu, dưới sự hướng dẫn của GVHD T.S Nguyễn Thiên Bảo. Các tài liệu, kết quả trình bày trong bài báo cáo này hoàn toàn trung thực, chưa từng được công bố trong bất kỳ bài báo cáo nào khác.

Trong tiểu luận có dùng một số tài liệu tham khảo như đã nêu trong phần tài liệu tham khảo.

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020

Nhóm sinh viên thực hiện

Phạm Quốc Việt

Nguyễn Hùng Duy

Đặng Huỳnh Hoàng Long

## LỜI CẢM ƠN

Lời đầu tiên, nhóm xin gửi lời cảm ơn chân thành đến GVHD T.S Nguyễn Thiên Bảo. Trong quá trình học tập và tìm hiểu môn Tiểu luận chuyên ngành, nhóm đã nhận được sự quan tâm giúp đỡ, hướng dẫn rất tận tình và tâm huyết của thầy. Thầy đã giúp nhóm tích lũy thêm nhiều kiến thức để có cái nhìn sâu sắc và hoàn thiện hơn trong đồ án môn học. Từ những kiến thức thầy đã truyền đạt, nhóm đã dần thấm nhuần được những lý thuyết và kỹ năng môn Tiểu luận chuyên ngành.

Có lẽ kiến thức là vô hạn mà sự tiếp nhận kiến thức của bản thân mỗi người luôn tồn tại những hạn chế nhất định. Do đó, trong quá trình hoàn thành bài tiểu luận này, chắc chắn không tránh khỏi những thiếu sót. Bản thân nhóm rất mong nhận được những góp ý đến từ thầy hướng dẫn và giáo viên phản biện để bài tiểu luận của nhóm dần được hoàn thiện hơn.

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020  
Nhóm sinh viên thực hiện

Phạm Quốc Việt  
Nguyễn Hùng Duy  
Đặng Huỳnh Hoàng Long

# MỤC LỤC

<b>NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH .....</b>	<b>1</b>
<b>PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN .....</b>	<b>2</b>
<b>PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN .....</b>	<b>3</b>
<b>LỜI CAM ĐOAN.....</b>	<b>4</b>
<b>LỜI CẢM ƠN.....</b>	<b>5</b>
<b>MỤC LỤC.....</b>	<b>6</b>
<b>DANH MỤC HÌNH ẢNH.....</b>	<b>9</b>
<b>DANH MỤC BẢNG BIỂU .....</b>	<b>11</b>
<b>DANH MỤC VIẾT TẮT .....</b>	<b>12</b>
<b>LỜI MỞ ĐẦU.....</b>	<b>13</b>
<b>CHƯƠNG I: TỔNG QUAN ĐỀ TÀI.....</b>	<b>14</b>
<b>1 Bài toán ung thư vú.....</b>	<b>14</b>
<b>1.1 Hiện trạng ung thư vú .....</b>	<b>14</b>
<b>1.2 Các loại ung thư vú .....</b>	<b>16</b>
<b>2. Dùng công nghệ tiên tiến, xử lý ảnh để giải quyết bài toán ung thư vú.....</b>	<b>19</b>
<b>2.1 Học máy .....</b>	<b>19</b>
<b>2.2 Học sâu.....</b>	<b>19</b>
<b>2.3 Khả năng dùng học sâu để giải quyết bài toán ung thư vú .....</b>	<b>19</b>
<b>2.3.1 Giới thiệu xử lý ảnh .....</b>	<b>19</b>
<b>2.3.2 Bài toán phân loại.....</b>	<b>20</b>
<b>3.Mục tiêu đề tài.....</b>	<b>21</b>
<b>3.1 Phạm vi nghiên cứu .....</b>	<b>21</b>
<b>3.2 Ý nghĩa khoa học và ý nghĩa thực tiễn.....</b>	<b>21</b>
<b>3.2.1 Ý nghĩa khoa học.....</b>	<b>21</b>
<b>3.2.2 Ý nghĩa thực tiễn.....</b>	<b>21</b>
<b>3.3 Phương pháp nghiên cứu .....</b>	<b>21</b>
<b>3.3.1 Phương pháp nghiên cứu lý thuyết .....</b>	<b>21</b>
<b>3.3.2 Phương pháp nghiên cứu thực nghiệm.....</b>	<b>22</b>
<b>4.Các công trình liên quan .....</b>	<b>22</b>
<b>CHƯƠNG II: TỔNG QUAN VỀ HỌC SÂU.....</b>	<b>23</b>



<b>1</b>	<b>Phương pháp học máy truyền thống</b>	<b>23</b>
1.1	Học có giám sát	23
1.2	Học không có giám sát	23
1.3	Hồi quy	24
1.4	Học tăng cường	24
<b>2</b>	<b>Phương pháp học sâu</b>	<b>24</b>
<b>3</b>	<b>Mạng neural tích chập</b>	<b>25</b>
3.1	Kiến trúc mạng neural tích chập	25
3.1.1	Lớp tích chập	25
3.1.2	Lớp giảm kích thước	26
3.1.3	Lớp hiệu chỉnh	27
3.1.4	Lớp chuẩn hóa	27
3.1.5	Lớp kết nối đầy đủ	27
3.2	Một số kiến trúc mạng neural tích chập	28
3.2.1	DenseNet	28
3.2.1.1	DenseNet121	29
3.2.1.2	DenseNet201	30
3.2.2	Xception	30
	<b>CHƯƠNG III: ÁP DỤNG HỌC SÂU ĐỂ PHÂN LOẠI</b>	<b>32</b>
	<b>ẢNH UNG THƯ VÚ</b>	<b>32</b>
1.	Phương pháp gia tăng dữ liệu	32
2.	Quy trình thực hiện huấn luyện mô hình	32
3.	Quá trình sau khi huấn luyện mô hình	33
	<b>CHƯƠNG IV: THỰC NGHIỆM</b>	<b>34</b>
1.	Môi trường thực hiện hệ thống	34
1.1	Kaggle	34
1.2	Jupyter	35
1.3	Tài nguyên sử dụng	35
1.3.1	CPU	35
1.3.2	GPU	35
1.3.3	TPU	36
2.	Framework	36

2.1	Django .....	36
2.2	ReactJS.....	37
2.3	Winform Application.....	37
3.	Tập dữ liệu.....	38
3.1	Thu thập dữ liệu.....	38
3.1.1	Tập dữ liệu BACH của ICIAR 2018 .....	38
3.1.2	Tập dữ liệu của Yan Rui, Ren Fei, Wang Zihao (15) .....	39
3.1.3	Tổng hợp .....	39
3.2	Xử lý dữ liệu và đóng gói dữ liệu.....	40
3.3	Chia dữ liệu .....	41
4.	Kết quả.....	42
4.1	Kết quả mô hình DensetNet201 .....	42
4.2	Kết quả mô hình DenseNet121.....	46
4.3	Kết quả mô hình Xception .....	51
4.4	Đóng gói mô hình .....	54
5.	Giao diện.....	54
5.1.	Winform Application .....	54
5.1.1	Giao diện trước khi phỏng đoán .....	55
5.1.2	Giao diện sau khi phỏng đoán .....	55
5.2.	Web Application .....	56
5.2.1	Giao diện web trước khi dự đoán .....	56
5.2.2	Giao diện web sau khi dự đoán.....	56
<b>CHƯƠNG V: TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN.....</b>		<b>58</b>
1	Kết quả đạt được.....	58
2	Hướng phát triển.....	58
<b>TÀI LIỆU THAM KHẢO .....</b>		<b>60</b>

## DANH MỤC HÌNH ẢNH

Hình 1 Tỷ lệ mắc ung thư vú trên toàn thế giới (1).....	14
Hình 2 Phân bố tỷ lệ mắc ung thư vú giữa các nhóm tuổi tại Hà Nội.....	15
Hình 3 Tình trạng bình thường, không có dấu hiệu của bệnh ung thư vú.....	16
Hình 4 Tình trạng ung thư vú lành tính của bệnh ung thư vú .....	17
Hình 5 Tình trạng ung thư biểu mô ống tuyến vú tại chỗ của bệnh ung thư vú.....	18
Hình 6 Tình trạng ung thư vú xâm lấn của bệnh ung thư vú.....	18
Hình 7 Tổng quan về phân loại hình ảnh ung thư vú bằng CNN (13) .....	20
Hình 8 Chi tiết CNN của mạng phân loại (13).....	21
Hình 9 Mối liên hệ giữa trí tuệ nhân tạo, học máy và học sâu .....	24
Hình 10 Kiến trúc mạng tích chập.....	25
Hình 11 Phương thức Average Pooling và Max Pooling (18) .....	26
Hình 12 Minh họa hoạt động của lớp pooling (18) .....	27
Hình 13 DenseNet (20).....	28
Hình 14 Các layer chi tiết của mô hình DenseNet (20).....	28
Hình 15 Cấu trúc của mô hình DenseNet121 .....	29
Hình 16 Thông số chi tiết của mô hình DenseNet121 dựa vào hình 14.....	29
Hình 17 Cấu trúc của mô hình DenseNet201 .....	30
Hình 18 Thông số chi tiết của mô hình DenseNet201 dựa vào hình 14.....	30
Hình 19 Luồng đi của mô hình Xception (23) .....	31
Hình 20 Các bước thực hiện train dữ liệu .....	32
Hình 21 Quá trình test .....	33
Hình 22 Hình ảnh về Kaggle (26) .....	34
Hình 23 Hình ảnh về Jupyter (28) .....	35
Hình 24 Hình ảnh về Framework Django (33).....	36
Hình 25 Hình ảnh về ReactJS (35) .....	37
Hình 26 Hình ảnh về Winform (37) .....	37
Hình 27 Một vài hình ảnh về tập dữ liệu Benign .....	38
Hình 28 Code minh họa thay đổi kích cỡ ảnh .....	40
Hình 29 Code minh họa co giãn ảnh .....	40
Hình 30 Chuyển đổi ảnh 1D sang 3D.....	41
Hình 31 Đóng gói dữ liệu dưới dạng HDF5.....	41
Hình 32 Code minh họa chia dữ liệu.....	41
Hình 33 Công thức Precision (38).....	42
Hình 34 Công thức Recall (38).....	42
Hình 35 Mô phỏng các lớp của mô hình DenseNet201 .....	43
Hình 36 Phần trăm độ chính xác và sai lệch của mô hình DenseNet201.....	43
Hình 37 Thống kê độ chính xác qua 4 nhãn của mô hình DenseNet201 .....	44
Hình 38 Confusion Matrix qua 4 nhãn của mô hình DenseNet201 .....	45

Hình 39 Đồ thị thể hiện độ chính xác và sai lệch của dữ liệu đang học mô hình và đang kiểm thử mô hình DenseNet201 .....	46
Hình 40 Mô phỏng các layer của mô hình DenseNet 121 .....	47
Hình 41 Phần trăm độ chính xác và sai lệch của mô hình DenseNet121 .....	47
Hình 42 Đồ thị biểu diễn độ chính xác và sai lệch của dữ liệu đang học và kiểm thử của mô hình DenseNet121 .....	48
Hình 43 Thống kê độ chính xác qua 4 nhãn của mô hình DenseNet121 .....	49
Hình 44 Confusion Matrix qua 4 nhãn của mô hình DenseNet121 .....	50
Hình 45 Mô phỏng các lớp của mô hình Xception.....	51
Hình 46 Phần trăm độ chính xác của và sai lệch của mô hình Xception .....	51
Hình 47 Thống kê độ chính xác qua 4 nhãn của mô hình Xception .....	52
Hình 48 Confusion Matrix của bốn nhãn của mô hình Xception.....	53
Hình 49 Đồ thị biểu diễn độ chính xác và độ sai lệch của mô hình Xception .....	54
Hình 50 Code minh họa đóng gói.....	54
Hình 51 Giao diện ứng dụng trước khi dự đoán.....	55
Hình 52 Giao diện ứng dụng sau khi dự đoán ra kết quả .....	55
Hình 53 Giao diện web sau khi dự đoán ra kết quả.....	56
Hình 54 Giao diện web sau khi dự đoán ra kết quả.....	57

## **DANH MỤC BẢNG BIỂU**

Bảng 1 Các thông số kết quả của ba mô hình (21).....	29
Bảng 2 Biểu đồ cột thể hiện số lượng ảnh trong mỗi nhãn .....	39
Bảng 3 Bảng thống kê số lượng ảnh trong mỗi nhãn .....	40

## DANH MỤC VIẾT TẮT

Các từ viết tắt	Tiếng Anh	Tiếng Việt
DNN	Deep Neural Network	Mạng neural sâu
CNN	Convolutional Neural Network	Mạng neural tích chập
DCNN	Deep Convolutional Neural Network	Mạng neural tích chập sâu
DCIS	Ductal Carcinoma in Situ	Ung thư biểu mô tuyến sữa tại chỗ
IDC/ILC	Invasive ductal/lobular carcinoma	Ung thư biểu mô ống xâm lấn/ung thư biểu mô tiểu thùy xâm lấn
TP	True Positive	Đoán là có thì nó sẽ xảy ra
FP	False Positive	Đoán là có nhưng nó lại là không
FN	False Negative	Đoán là không nhưng nó lại là có
GCP	Google Cloud Platform	Nền tảng đám mây của google
GPU	Graphics Processing Unit	Bộ xử lý đồ họa
CPU	Central Processing Unit	Bộ xử lý trung tâm
TPU	Tensor Processing Uni	Bộ xử lý tensor

## LỜI MỞ ĐẦU

Deep Learning là một thuật toán dựa trên một số ý tưởng từ não bộ tới việc tiếp thu nhiều tầng biểu đạt, cả cụ thể lẫn trừu tượng, qua đó làm rõ nghĩa của các loại dữ liệu. Deep Learning được ứng dụng trong nhận diện hình ảnh, nhận diện giọng nói, xử lý ngôn ngữ tự nhiên, phân loại hình ảnh, phân loại phong cách thiết kế dựa trên bản vẽ.

Convolutional Neural Network (CNNs-Mạng neural tích chập) là một trong những mô hình Deep Learning tiên tiến giúp chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. Trong tiểu luận chuyên ngành này, nhóm đi vào nghiên cứu về mạng neural cũng như mạng Convolutional cũng như các ý tưởng của mô hình CNNs trong phân lớp ảnh và áp dụng trong việc phân loại ảnh của bệnh ung thư vú.

Nội dung báo cáo của nhóm gồm năm chương:

Chương 1: Tổng quan đề tài

Chương 2: Tổng quan về học sâu

Chương 3: Áp dụng học sâu giải để giải quyết bài toán ung thư vú

Chương 4: Thực nghiệm

Chương 5: Tổng kết và hướng phát triển

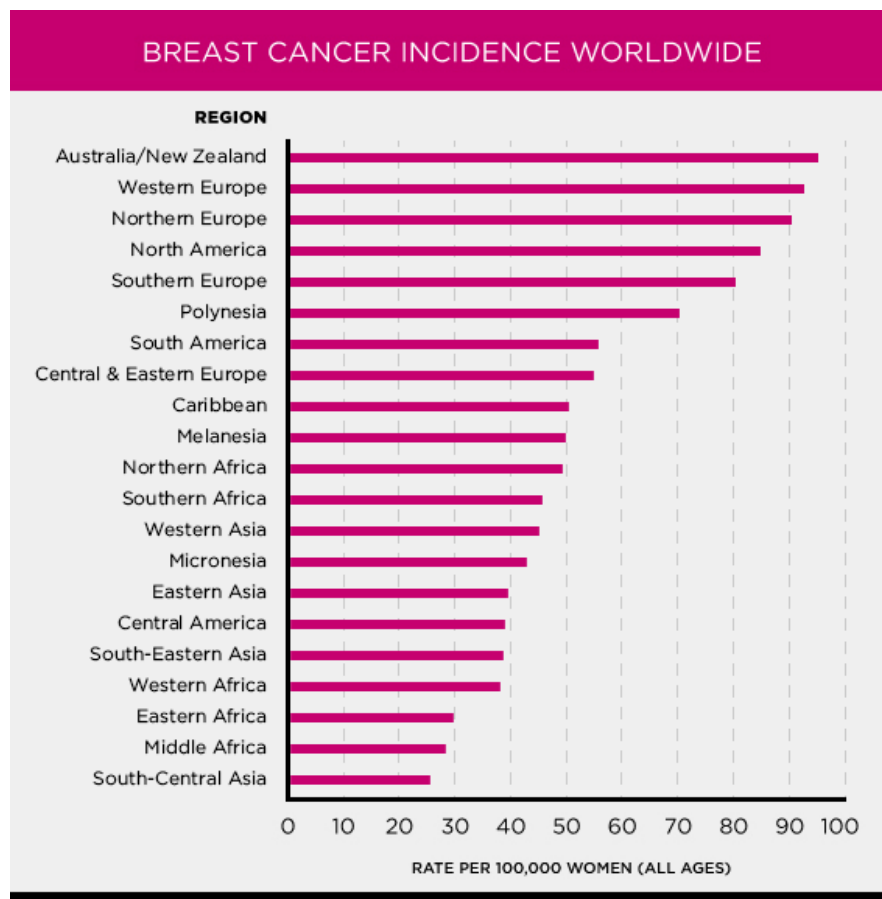
# CHƯƠNG I: TỔNG QUAN ĐỀ TÀI

## 1 Bài toán ung thư vú

Vú là một bộ phận nhạy cảm trong cơ thể con người nói chung và phụ nữ nói riêng. Thế nhưng, bệnh ung thư vú có nguy cơ cao xảy ra trên cơ thể người phụ nữ và căn bệnh này rất khó chữa trị. Ngày nay, căn bệnh ung thư vú này ngày càng phổ biến hơn nữa. (1)

### 1.1 Hiện trạng ung thư vú

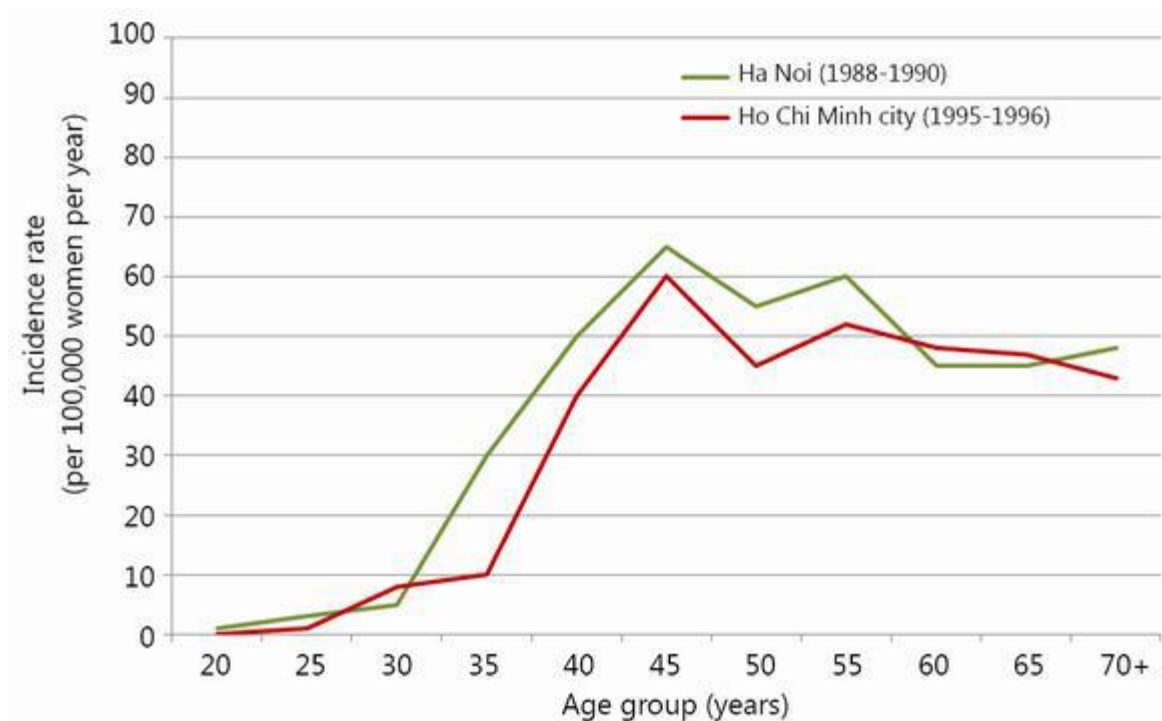
*Tình hình thế giới:* Ung thư vú là bệnh ung thư phổ biến nhất ở phụ nữ trên toàn thế giới. Nhìn chung, các nước phát triển (như Mỹ, Anh và Úc) có tỷ lệ ung thư vú cao hơn các nước đang phát triển (như Campuchia, Nepal và Rwanda). Ước tính có hơn 600.000 ca tử vong do ung thư vú ở phụ nữ và nam giới trên toàn thế giới xảy ra vào năm 2018. (1)



Hình 1 Tỷ lệ mắc ung thư vú trên toàn thế giới (1)



*Tình hình Việt Nam :* Năm 2012, có khoảng 11.060 trường hợp nữ được chẩn đoán ung thư vú, với 64,7% trường hợp dưới 50 tuổi. Số liệu này cho thấy ung thư vú là loại ung thư hàng đầu ở phụ nữ Việt Nam hoặc đứng thứ 5 trong tổng số các trường hợp ung thư ở nữ. Ngoài ra, sự gia tăng nhanh chóng số lượng phụ nữ béo phì ở Việt Nam từ 5% năm 1980 lên 12% năm 2013 có thể đã góp phần vào sự gia tăng tỷ lệ mắc ung thư vú gần đây (2). Nhóm tuổi phụ nữ mắc bệnh ung thư vú phổ biến nhất ở Việt Nam là từ 45 đến 55 tuổi. Tỷ lệ bệnh nhân ung thư vú ở Việt Nam ở giai đoạn 0-I và II lần lượt là 14,7% và 61,2% (3), trong khi đó ở các giai đoạn xa (III, IV) là 27,6% (3). Những con số này trái ngược với ở Hoa Kỳ, nơi 58,6% trường hợp ung thư được chẩn đoán ở giai đoạn (4) và 72% trường hợp ở Úc là do ung thư không xâm lấn. (5)



Hình 2 Phân bố tỷ lệ mắc ung thư vú giữa các nhóm tuổi tại Hà Nội và Thành phố Hồ Chí Minh (2)

### *Khả năng chữa trị:*

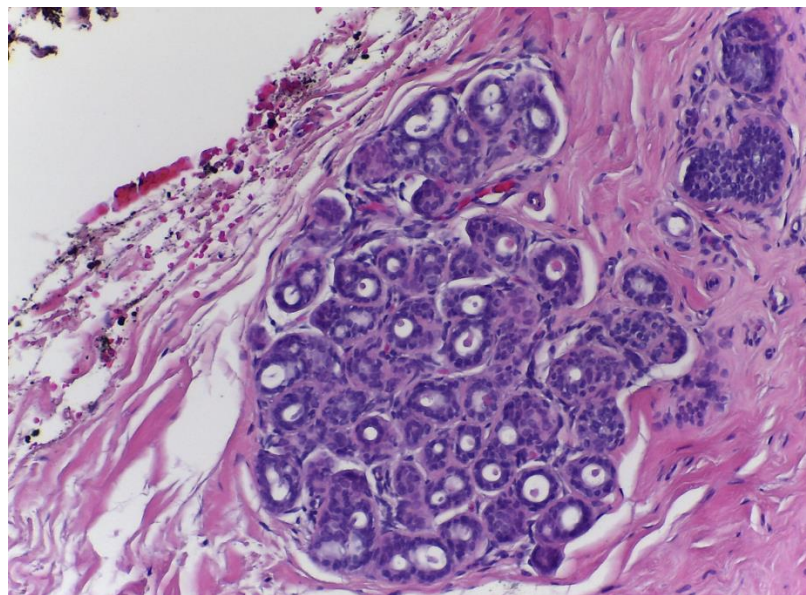
Hoàn toàn có thể chữa khỏi nếu được phát hiện sớm. Thực tế cho thấy 80% bệnh nhân ung thư vú có cơ hội chữa khỏi bệnh khi phát hiện ở giai đoạn đầu. Tỷ lệ chữa khỏi giảm xuống còn 60% khi ung thư vú chuyển sang giai đoạn thứ 02. Và khi đến các giai đoạn về sau thì cơ hội lại càng giảm xuống và việc điều trị chỉ mang đến tác dụng duy trì sự sống, chứ hoàn toàn không có khả năng chữa khỏi. (6)

Phát hiện giai đoạn đầu vẫn có thể sống khỏe mạnh đến 20 năm và lâu hơn thế nữa nếu tuân thủ đúng phác đồ điều trị bác sĩ đưa ra. Bởi vậy việc tầm soát, khám định kỳ sức khỏe là điều hết sức quan trọng để phát hiện sức khỏe bản thân. (6)

Nếu ung thư vú được phát hiện sớm ở giai đoạn 0, tỷ lệ sống của người bệnh sau 05 năm lên đến 96%. Và tỷ lệ sống sau 05 năm ở các giai đoạn sau của bệnh giảm dần xuống, cụ thể ở giai đoạn 01 là 92%, 85% ở giai đoạn 02, 67% ở giai đoạn 03. Và nếu người bệnh phát hiện bệnh ở giai đoạn 04 thì tỷ lệ sống sau 05 năm chỉ còn hơn 20%. (6)

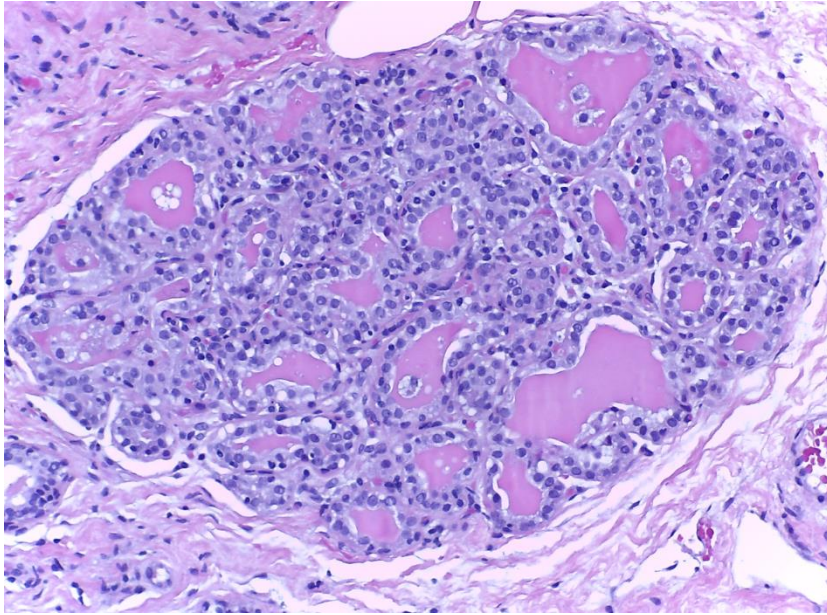
## **1.2 Các loại ung thư vú**

*Normal:* Tình trạng âm tính, không có dấu hiệu của bệnh ung thư vú.



Hình 3 Tình trạng bình thường, không có dấu hiệu của bệnh ung thư vú

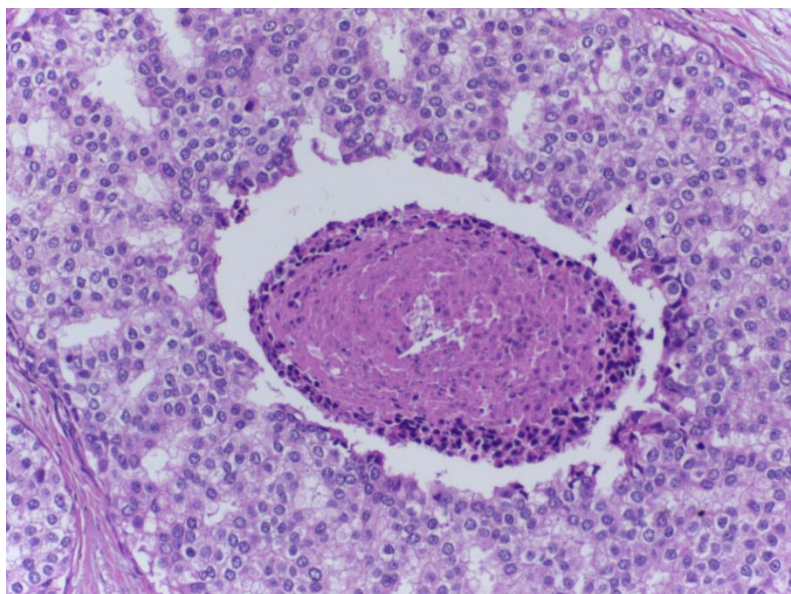
*Benign (Non-cancerous)*: Tình trạng u lành tính (không ung thư) là sự phát triển bất thường hoặc những thay đổi khác trong mô vú và nó không phải là ung thư. (7)



Hình 4 Tình trạng ung thư vú lành tính của bệnh ung thư vú

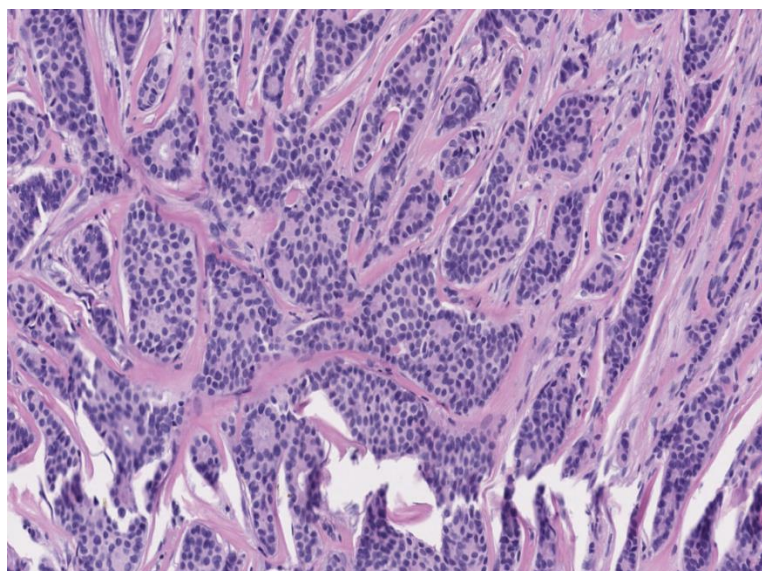
*In Situ (DCIS)*: Ung thư biểu mô ống dẫn sữa xảy ra khi các tế bào trong một số ống dẫn sữa hình thành tế bào ung thư. Những tế bào này đều nằm trong các ống dẫn sữa và không lan sang các mô xung quanh. (8)





Hình 5 Tình trạng ung thư biểu mô ống tuyến vú tại chỗ của bệnh ung thư vú

*Invasive (IDC/ILC):* Xảy ra khi các tế bào ung thư từ bên trong ống dẫn sữa hoặc tiểu thùy bùng phát vào mô vú gần đó. (9)



Hình 6 Tình trạng ung thư vú xâm lấn của bệnh ung thư vú

## 2. Dùng công nghệ tiên tiến, xử lý ảnh để giải quyết bài toán ung thư vú

Ngày nay, với việc phát triển công nghệ càng ngày vượt bậc thì việc giao tiếp giữa con người và máy móc trở nên dễ dàng hơn. Với nhiều kiến trúc học máy khác nhau được áp dụng như mạng neural nhân tạo (ANN) hoặc mạng neural học sâu (DNN) đã từng bước được áp dụng vào các lĩnh vực của đời sống và kết quả áp dụng này đã vượt trội xa hơn các phương pháp học truyền thống khác.

Đặc biệt với sự phát triển mạnh mẽ của phần cứng cho phép tính toán hàng tỷ phép tính trong thời gian rất ngắn. Và đó là cơ sở tạo tiền đề cho mạng CNN trở nên phổ biến hơn. Một trong ứng dụng của mạng CNN đó chính là cho phép máy móc có khả năng nhìn và phân tích, phân loại hình ảnh.

Bởi vì do sự cấp thiết và nguy hiểm của căn bệnh ung thư vú đối với con người, nên nhóm quyết định chọn đề tài **tìm hiểu học sâu cho bài toán phân loại ảnh ung thư vú** để làm đề tài cho tiểu luận chuyên ngành của nhóm.

### 2.1 Học máy

Học máy là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các thuật toán cho phép các hệ thống học tự động từ dữ liệu để giải quyết những vấn đề cụ thể. (10)

### 2.2 Học sâu

Học sâu là một chi của ngành học máy dựa trên một tập hợp các thuật toán để huấn luyện mô hình dữ liệu trừu tượng hóa ở mức cao bằng cách sử dụng nhiều lớp xử lý với cấu trúc phức tạp, hoặc bằng cách khác bao gồm nhiều biến đổi phi tuyến tính. (11)

### 2.3 Khả năng dùng học sâu để giải quyết bài toán ung thư vú

#### 2.3.1 Giới thiệu xử lý ảnh

Xử lý ảnh là đối tượng nghiên cứu trong lĩnh vực thị giác máy tính, là quá trình biến đổi từ một ảnh ban đầu sang một ảnh mới với các đặc tính và tuân theo ý muốn của người sử dụng. (12)

Xử lý ảnh bao gồm quá trình phân tích, phân lớp các đối tượng, làm tăng chất lượng kích cỡ ảnh, phân đoạn và tách cạnh. (12)

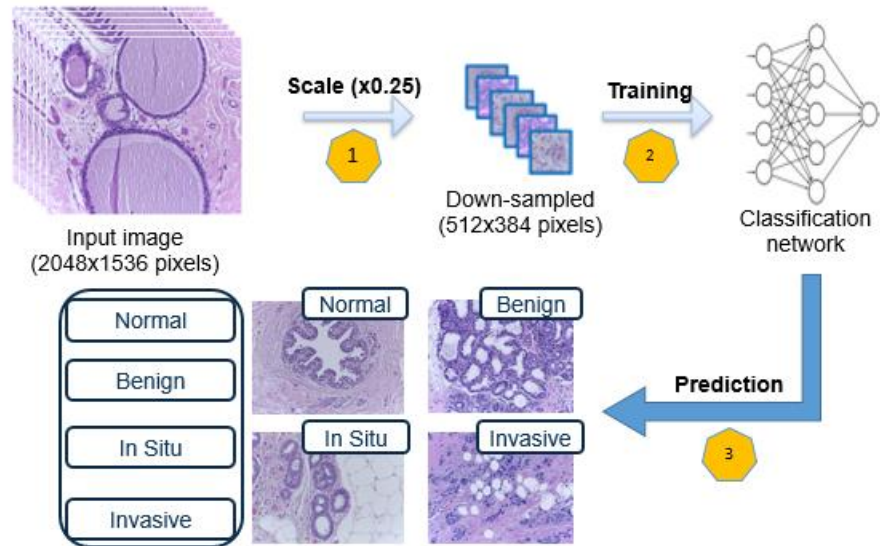
Mục đích xử lý ảnh là để biến đổi ảnh làm tăng chất lượng của ảnh. Tự động nhận dạng ảnh, đoán ảnh, đánh giá các nội dung của ảnh. (12)

Nhận biết và đánh giá nội dung của ảnh là sự phân tích một hình ảnh thành nhiều phần để phân biệt với nhau qua các đặc trưng, dựa vào đó ta có thể mô tả cấu trúc của hình ảnh ban đầu. (12)

Có các phương pháp nhận dạng ảnh như nhận dạng ảnh của các đối tượng trên ảnh, tách cạnh, phân đoạn. (12)

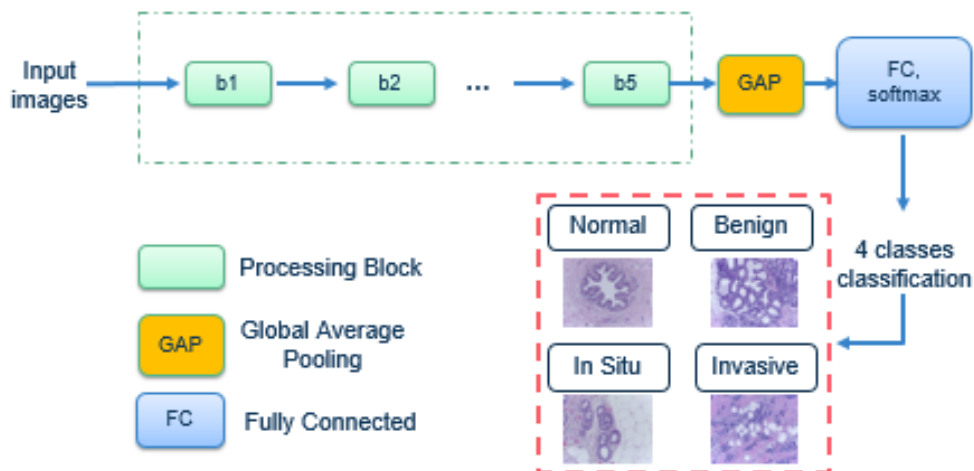
### 2.3.2 Bài toán phân loại

Một mạng lưới phân loại bao gồm năm khối xử lý, lớp tổng hợp trung bình toàn cầu (GAP) và lớp softmax kết nối đầy đủ (FC). (13)



Hình 7 Tổng quan về phân loại hình ảnh ung thư vú bằng CNN (13)

Có ba thành phần trong khối xử lý, đơn vị trên và dưới là chập tiêu chuẩn để giảm kích thước, đơn vị ở giữa theo nguyên tắc thiết kế của DenseNet, trong đó một chuỗi các tích chập đang được thực thi, mỗi lần thực thi đều lấy tất cả các đầu vào dưới dạng tái sử dụng tính năng. (13)



Hình 8 Chi tiết CNN của mạng phân loại (13)

Tính năng tái sử dụng để giảm số lượng tham số, mỗi thành phần riêng biệt tìm tính năng có cấu trúc bằng cách tiến hành tích chập hai lần với tốc độ giãn nở tiêu chuẩn. (13)

### 3. Mục tiêu đề tài

#### 3.1 Phạm vi nghiên cứu

Phạm vi nghiên cứu của đề tài là dựa vào đặc trưng của hình ảnh bệnh ung thư vú từ tập dữ liệu lưu trữ BACH của ICIAR 2018 (14) và Yan Rui (15)

#### 3.2 Ý nghĩa khoa học và ý nghĩa thực tiễn

##### 3.2.1 Ý nghĩa khoa học

Đưa ra các mô hình cho bài toán phân loại bệnh ung thư vú.

Đánh giá thực nghiệm trên các tập dữ liệu thực.

##### 3.2.2 Ý nghĩa thực tiễn

Kết quả nghiên cứu của đề tài có thể làm tài liệu tham khảo cho bài toán phân loại ung thư vú trên các ứng dụng của hệ điều hành Windows và trên Website.

#### 3.3 Phương pháp nghiên cứu

##### 3.3.1 Phương pháp nghiên cứu lý thuyết

Bài toán phân loại ảnh là một trong những bài toán phổ biến trong học sâu. Vì vậy, nhóm sẽ nghiên cứu học sâu để áp dụng giải quyết bài toán ung thư vú.

### *3.3.2 Phương pháp nghiên cứu thực nghiệm*

Lấy dữ liệu đã thu thập được sau đó huấn luyện các mô hình của học sâu và đánh giá độ chính xác.

Tiến hành thực nghiệm và đánh giá độ chính xác dựa trên các mô hình và dữ liệu, thể hiện cụ thể qua ứng dụng winform và một Website.

## **4. Các công trình liên quan**

Trong bài báo của Canh Phong Nguyen, Anh Hoang Vo và Bao Thien Nguyen, báo cáo phương pháp, kết quả đánh giá độ chính xác cho phân loại bệnh ung thư vú. Trong bài báo cáo này, họ có tập dữ liệu hình ảnh ban đầu và họ lật ngang, lật dọc và quay 90 độ cho ra bộ dữ liệu mới để huấn luyện được mô hình tốt nhất có thể. Bên cạnh đó, họ còn áp dụng kỹ thuật gia tăng thời gian thử nghiệm được áp dụng để đưa ra kết quả tốt hơn. (13)

Trong nghiên cứu này, họ áp dụng CNN để tìm ra các mô hình có tỷ lệ thành công cao nhất để phân loại bệnh ung thư vú từ tập dữ liệu lưu trữ BACH (14). CNN sẽ hữu ích cho việc phân loại bệnh ung thư vú. Trong bài báo trên, thì tỷ lệ thành công của bài toán phân loại ung thư vú đạt tới 78% về độ chính xác. (13)



## CHƯƠNG II: TỔNG QUAN VỀ HỌC SÂU

### 1 Phương pháp học máy truyền thống

Học máy là khái niệm mà một chương trình máy tính có thể học và thích ứng với dữ liệu mới mà không cần sự can thiệp của con người. Học máy là một lĩnh vực của trí tuệ nhân tạo giúp cho các thuật toán tích hợp sẵn của máy tính luôn cập nhật bất kể những thay đổi của nền kinh tế toàn cầu. (15)

Lượng dữ liệu khổng lồ gọi là BigData, đang dần trở nên sẵn sàng và sử dụng do sự tiến bộ vượt bậc của nền công nghệ hiện nay, đặc biệt là khả năng tính toán và khả năng lưu trữ trên đám mây. Các công ty và chính phủ nhận ra những thông tin chi tiết có thể thu được từ việc khai thác dữ liệu lớn nhưng thiếu nguồn lực và thời gian cần thiết để tìm hiểu thông tin phong phú của nó. Do đó, các biện pháp trí tuệ nhân tạo (AI) đang được các ngành công nghiệp khác nhau sử dụng để thu thập, xử lý, giao tiếp và chia sẻ thông tin hữu ích từ các tập dữ liệu. Một phương pháp AI ngày càng được sử dụng nhiều hơn để xử lý dữ liệu lớn là học máy. (12)

#### 1.1 Học có giám sát

Học có giám sát (supervised learning) là cách học với tập dữ liệu huấn luyện ban đầu được gán nhãn hoàn toàn từ trước. (12)

Học có giám sát là thuật toán dự đoán nhãn của một dữ liệu mới dựa trên tập dữ liệu huấn luyện mà trong đó mỗi mẫu dữ liệu đều đã được gán nhãn. (17)

Khi đó, thông qua một quá trình huấn luyện, một mô hình sẽ được xây dựng để cho ra các dự đoán và khi các dự đoán bị sai thì mô hình này sẽ được tinh chỉnh lại. Việc huấn luyện sẽ tiếp tục cho đến khi mô hình đạt được mức độ chính xác mong muốn trên dữ liệu huấn luyện. Điều này cũng giống như khi chúng ta đi học trên lớp, ta biết câu trả lời chính xác từ giáo viên (tập dữ liệu có nhãn) và từ đó ta sẽ sửa chữa nếu làm sai. (17)

Học có giám sát là nhóm phổ biến nhất trong các thuật toán học máy. (17)

#### 1.2 Học không có giám sát

Học không có giám sát (unsupervised learning) là cách học với tập dữ liệu huấn luyện ban đầu hoàn toàn chưa được gán nhãn. (12)

Là thuật toán dự đoán nhãn của một dữ liệu mới dựa trên tập dữ liệu huấn luyện mà trong đó tất cả các mẫu dữ liệu đều chưa được gán nhãn. (17)

### 1.3 Hồi quy

Một bài toán được xem là hồi quy nếu nhãn không được chia thành các nhóm mà là một giá trị thực cụ thể (miền giá trị là liên tục). Hầu hết các bài toán dự báo (giá cổ phiếu, giá nhà, ...) thường được xếp vào bài toán hồi quy.

(17) Nếu như phát hiện khuôn mặt là bài toán phân lớp thì dự đoán tuổi là bài toán hồi quy. Tuy nhiên dự đoán tuổi cũng có thể coi là phân lớp nếu ta cho tuổi là một số nguyên dương  $N$  và khi đó ta sẽ có  $N$  lớp khác nhau tính từ 01.

### 1.4 Học tăng cường

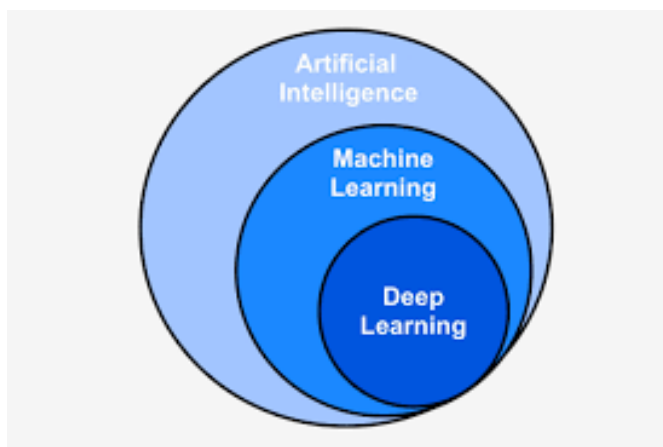
Học tăng cường (reinforcement learning) là cách học để giải quyết các bài toán giúp cho một hệ thống tự động xác định hành vi dựa trên hoàn cảnh để đạt được lợi ích cao nhất. (12)

## 2 Phương pháp học sâu

Học sâu là một tập con của phương pháp học máy tuy nhiên nó rộng hơn dựa trên đại diện học của dữ liệu. Một quan sát về một hình ảnh có thể được biểu diễn bằng nhiều cách như một vector của các giá trị cường độ cho mỗi điểm ảnh, hoặc một cách trừu tượng hơn như là một tập hợp các cạnh, các khu vực hình dạng cụ thể. (12)

Về cơ bản, học sâu là một hệ thống máy tính chứa rất nhiều dữ liệu, để chúng có thể sử dụng và đưa ra quyết định về những dữ liệu khác. Dữ liệu này được nạp thông qua các mạng thần kinh tương tự như học máy, những mạng logic phức tạp cần thiết để xử lý các bộ dữ liệu lớn. (12)

Học sâu có hai loại là mạng neural tích chập và mạng neural nhân tạo. Tuy nhiên, bài toán phân loại bệnh ung thư vú sử dụng phân loại hình ảnh là thể mạnh của CNN nên nhóm quyết định sử dụng CNN. (12)



Hình 9 Mối liên hệ giữa trí tuệ nhân tạo, học máy và học sâu

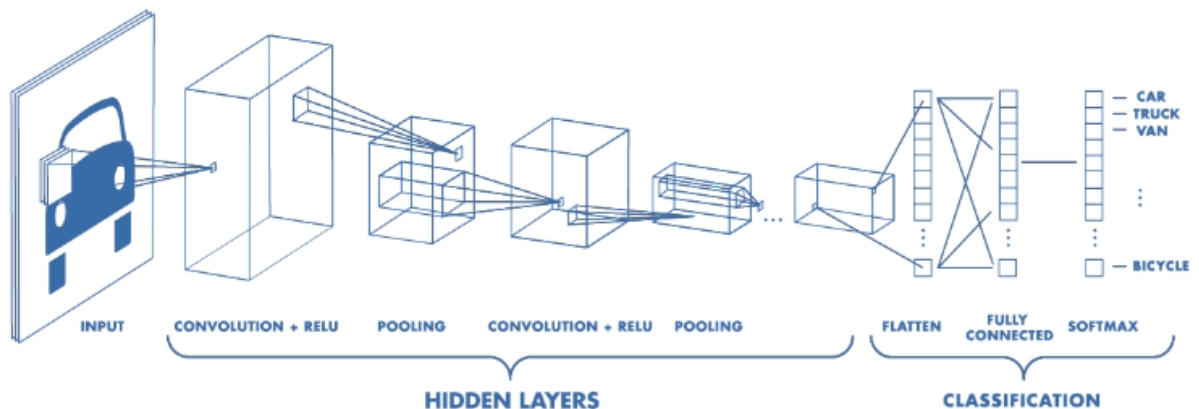
### 3 Mạng neural tích chập

Convolutional Neural Network (CNNs – Mạng neural tích chập) là một trong những mô hình học sâu tiên tiến. Nó giúp cho chúng ta xây dựng được hệ thống thông minh với độ chính xác cao như hiện nay. (17)

CNN được ứng dụng trong nhiều bài toán như nhận dạng ảnh, phân tích video, ảnh MRI, hoặc cho các bài toán của lĩnh vực xử lý ngôn ngữ tự nhiên. (17)

#### 3.1 Kiến trúc mạng neural tích chập

Kiến trúc của mạng neural tích chập gồm một lớp đầu vào (input layer), một lớp đầu ra (output layer) và một số lớp giữa gọi là lớp ẩn (hidden layer). Các lớp ẩn gồm: lớp tích chập (convolutional layer), lớp giảm kích thước (pooling layer), lớp sửa dữ liệu (ReLU layer), lớp kết nối đầy đủ (FC- fully connected layer) và lớp chuẩn hóa (normalization layer). (18)



Hình 10 Kiến trúc mạng tích chập

##### 3.1.1 Lớp tích chập

Tích chập được sử dụng đầu tiên trong xử lý tín hiệu số. Nhờ vào nguyên lý biến đổi thông tin có thể áp dụng kỹ thuật này vào xử lý ảnh và video số. (18)

Trong lớp tích chập sử dụng một bộ các bộ lọc có kích thước nhỏ hơn với ma trận đầu vào và áp lên một vùng của ma trận đầu vào và tiến hành tính tích chập giữa bộ filter, giá trị của ma

trận trong vùng cục bộ đó. (18)

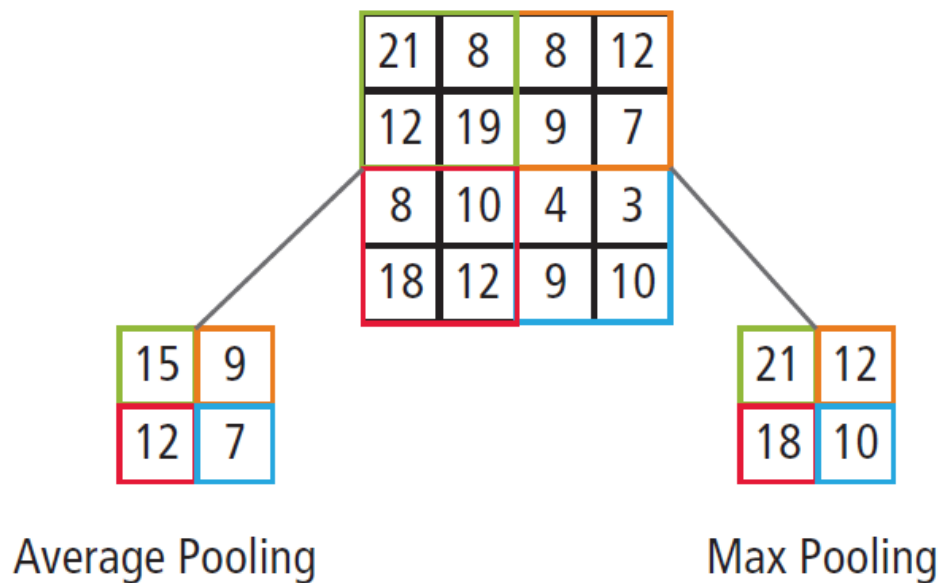
Các filter sẽ dịch chuyển một bước trượt (stride) chạy dọc theo ma trận đầu vào và quét toàn bộ ma trận. Trọng số của filter ban đầu sẽ được khởi tạo ngẫu nhiên và sẽ được học dần trong quá trình huấn luyện mô hình. (18)

### 3.1.2 Lớp giảm kích thước

Lớp Pooling sử dụng một cửa sổ trượt quét qua toàn bộ ma trận dữ liệu theo một bước trượt cho trước để tiến hành lấy mẫu.

Các phương thức lấy phổ biến trong lớp Pooling là MaxPooling (lấy giá trị lớn nhất), MinPooling (lấy giá trị nhỏ nhất) và AveragePooling (lấy giá trị trung bình).

Công dụng của lớp Pooling dùng để giảm kích thước dữ liệu, các tầng trong CNN chồng lên nhau có lớp Pooling ở cuối mỗi tầng giúp cho kích thước dữ liệu được co lại nhưng vẫn giữ được các đặc trưng để lấy mẫu. (18)



Hình 11 Phương thức Average Pooling và Max Pooling (18)

### 3.1.3 Lớp hiệu chỉnh

Lớp hiệu chỉnh được xây dựng với ý nghĩa đảm bảo tính phi tuyến của mô hình huấn luyện. Khi đã thực hiện một loạt các phép tính toán tuyến tính qua các lớp tích chập. Lớp Kích hoạt phi tuyến nói chung sử dụng các hàm kích hoạt phi tuyến như ReLU hoặc sigmoid, tanh... để giới hạn phạm vi biên độ cho phép của giá trị đầu ra.

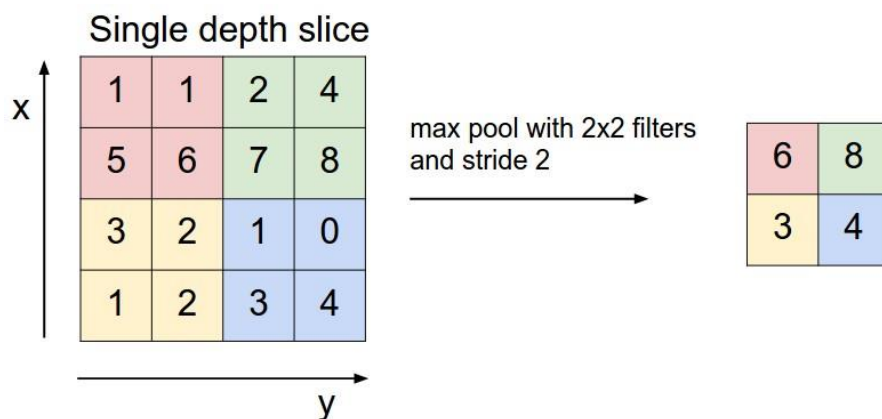
Trong số các hàm kích hoạt này, hàm ReLU được chọn do cài đặt đơn giản, tốc độ xử lý nhanh mà vẫn đảm bảo được tính toán hiệu quả. Cụ thể, phép tính toán của hàm ReLU chỉ đơn giản là chuyển tất cả các giá trị âm thành giá trị 0. Thông thường, lớp ReLU được áp dụng ngay phía sau lớp tích chập, với đầu ra là một ảnh mới có kích thước giống với ảnh đầu vào, các giá trị điểm ảnh cũng hoàn toàn tương tự trừ các giá trị âm đã bị loại bỏ. (18)

### 3.1.4 Lớp chuẩn hóa

Có nhiều kiểu khác nhau của lớp chuẩn hoá được đề xuất sử dụng trong cấu trúc CNN. Lớp chuẩn hoá thường được sử dụng giữa lớp tích chập và lớp ReLU nhằm mục đích tăng huấn luyện và làm giảm sự nhạy cảm trong khởi tạo tham số của mạng. Tuy nhiên những lớp chuẩn hoá này đôi khi không được ưu tiên sử dụng vì trong thực tế một số bài toán đóng góp của nó thực sự không gây ảnh hưởng nhiều vào kết quả chung. (18)

### 3.1.5 Lớp kết nối đầy đủ

Lớp kết nối đầy đủ (fully connected layer) là một lớp giống như mạng neural truyền thẳng các giá trị được tính toán từ các lớp trước sẽ được liên kết đầy đủ vào trong neural của lớp tiếp theo. Tại lớp này sẽ tiến hành phân lớp dữ liệu bằng cách kích hoạt hàm softmax để tính xác suất ở lớp đầu ra. (18)



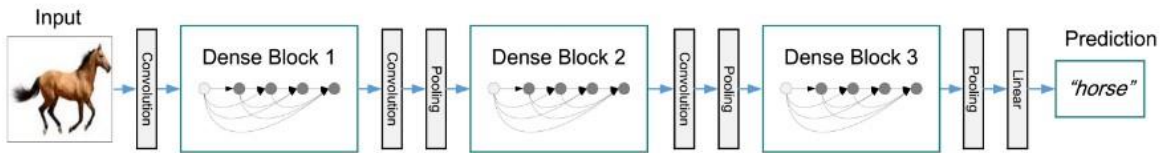
Hình 12 Minh họa hoạt động của lớp pooling (18)

### 3.2 Một số kiến trúc mạng neural tích chập

#### 3.2.1 DenseNet

Trước tiên chúng ta sẽ đi tìm hiểu về object detection. Object detection có nghĩa là đề cập đến khả năng của hệ thống máy tính và phần mềm để định vị các đối tượng trong một hình ảnh và xác định từng đối tượng. Object Detection đã được sử dụng rộng rãi để phát hiện khuôn mặt, phát hiện xe, đếm số người đi bộ, hệ thống bảo mật và xe không người lái. (19)

DenseNet (Dense connected convolutional network) là một trong những network mới nhất cho visual object recognition. Nó cũng gần giống ResNet nhưng có một vài điểm khác biệt. DenseNet có cấu trúc gồm các dense block và các transition layers. Với CNN truyền thống nếu chúng ta có L layer thì sẽ có L connection, còn trong densenet sẽ có  $L(L+1)/2$  connection. (20)



Hình 13 DenseNet (20)

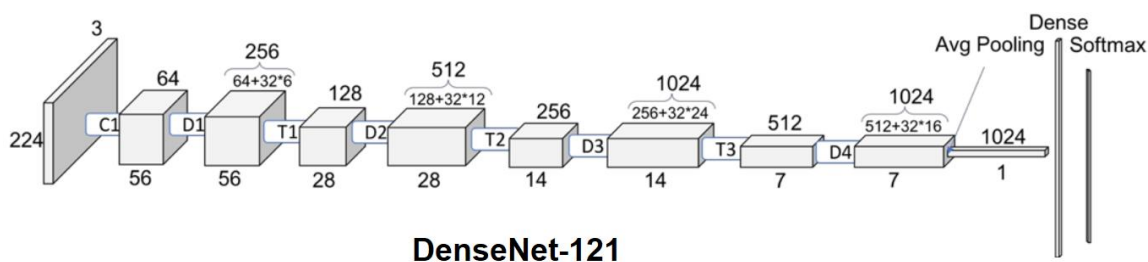
Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	$112 \times 112$	$7 \times 7$ conv, stride 2			
Pooling	$56 \times 56$	$3 \times 3$ max pool, stride 2			
Dense Block (1)	$56 \times 56$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	$56 \times 56$	$1 \times 1$ conv			
	$28 \times 28$	$2 \times 2$ average pool, stride 2			
Dense Block (2)	$28 \times 28$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	$28 \times 28$	$1 \times 1$ conv			
	$14 \times 14$	$2 \times 2$ average pool, stride 2			
Dense Block (3)	$14 \times 14$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 64$
Transition Layer (3)	$14 \times 14$	$1 \times 1$ conv			
	$7 \times 7$	$2 \times 2$ average pool, stride 2			
Dense Block (4)	$7 \times 7$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$
Classification Layer	$1 \times 1$	$7 \times 7$ global average pool			
		1000D fully-connected, softmax			

Hình 14 Các layer chi tiết của mô hình DenseNet (20)

Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	depth
Xception	88M	0.790	0.945	22,910,480	126
DenseNet201	80M	0.773	0.936	20,242,984	201
DenseNet121	33M	0.750	0.923	8,062,054	121

Bảng 1 Các thông số kết quả của ba mô hình (21)

### 3.2.1.1 DenseNet121



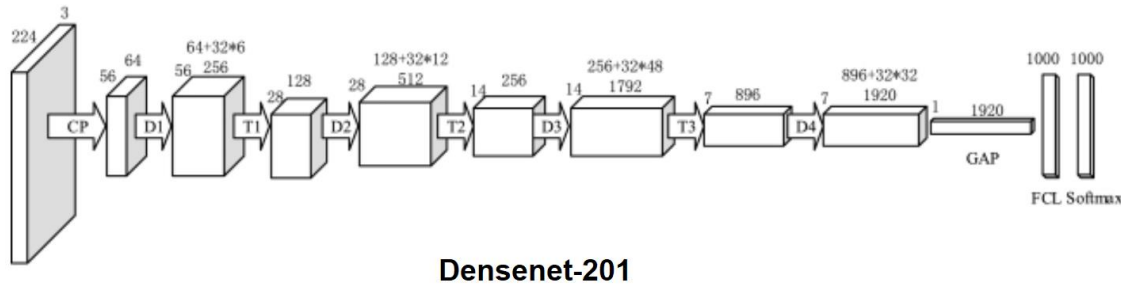
Hình 15 Cấu trúc của mô hình DenseNet121

Tổng số lớp của mô hình DenseNet121 được tính sau đây:

Công thức tính số lớp của mô hình DenseNet121: $5 + (6 + 12 + 24 + 16) * 2 = 121$		
Trong đó:		
5 - Convolution and Pooling Layer		
3 - Transition Layer (6,12,24)		
1 - Classification Layer (16)		
2 - DenseBlock (1x1 and 3x3 conv)		

Hình 16 Thông số chi tiết của mô hình DenseNet121 dựa vào hình 14

### 3.2.1.2 DenseNet201



Hình 17 Cấu trúc của mô hình DenseNet201

Công thức tính số lớp của mô hình DenseNet201: $5 + (6 + 12 + 48 + 32) * 2 = 201$		
Trong đó:		
5 - Convolution and Pooling Layer		
3 - Transition Layer (6,12,48)		
1 - Classification Layer (32)		
2 - DenseBlock (1x1 and 3x3 conv)		

Hình 18 Thông số chi tiết của mô hình DenseNet201 dựa vào hình 14

### 3.2.2 Xception

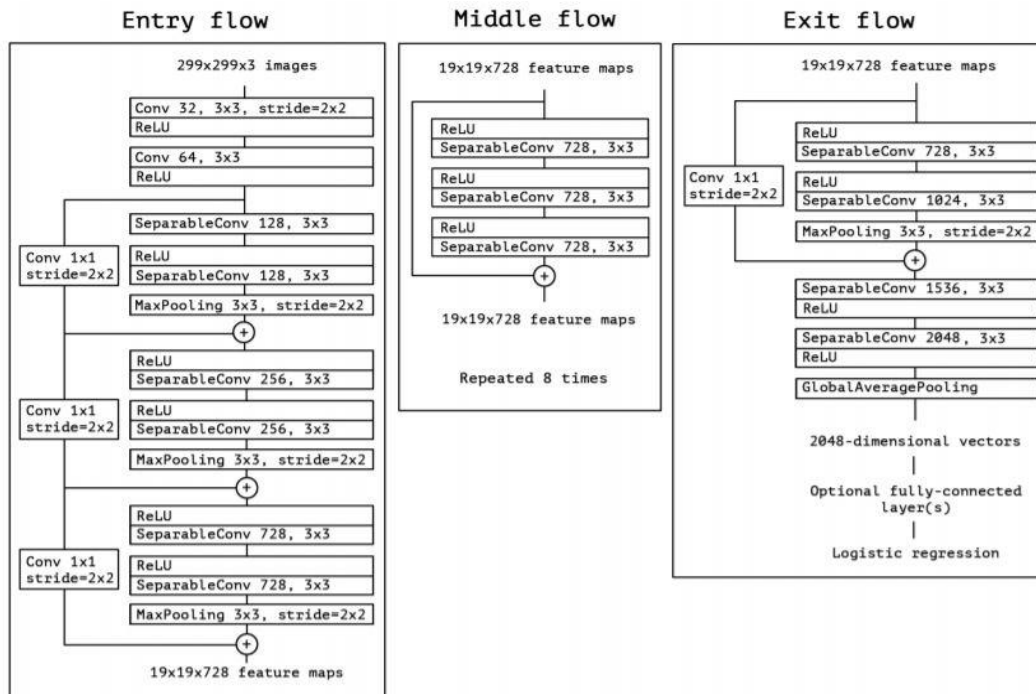
Xception là một kiến trúc mạng neural tích hợp sâu liên quan đến các chuyển đổi có thể tách rời theo chiều sâu.

Nó được phát triển bởi các nhà nghiên cứu của Google. Google đã trình bày giải thích về các mô đun khởi động trong mạng neural tích chập như là một bước trung gian giữa phép chập thông thường và phép toán tích chập phân tách theo chiều sâu.

Xception là một kiến trúc mạng neural tích chập chỉ dựa vào các lớp tích chập có thể phân tách theo chiều sâu. Nó là một chồng tuyến tính của các lớp tích chập được phân tách theo chiều sâu. Nó là một sự chuyển thể từ Inception. Nó có độ sâu 71 lớp, nó có thể phân loại hình ảnh



thành 1000 loại đối tượng như bàn phím, chuột, bút chì và nhiều động vật, và có kích thước đầu vào hình ảnh là 299 x 299. (22)



Hình 19 Luồng đi của mô hình Xception (23)

Mô hình của Xception được chia thành 3 luồng. Đầu tiên, Xception sẽ đi qua luồng nhập, sau đó qua luồng giữa và lặp lại 8 lần và cuối cùng là thông qua luồng thoát. (23)

Hiệu suất: Vượt trội hơn Inception V3 trên tập dữ liệu ImageNet. Xception có cùng số lượng tham số (parameters) với Inception V3. Hiệu suất của Xception hiệu quả hơn Inception V3 do sử dụng hiệu quả hơn các tham số của mô hình. (23)

Kiến trúc: Kiến trúc của Xception có 36 lớp tích chập tạo thành cơ sở trích xuất tính năng của mạng. 36 lớp tích chập này được cấu trúc thành 14 modules và được kết nối tuyến tính tới những modules xung quanh nó ngoại trừ modules đầu và cuối cùng. Số parameters: 22,910,480. (23)

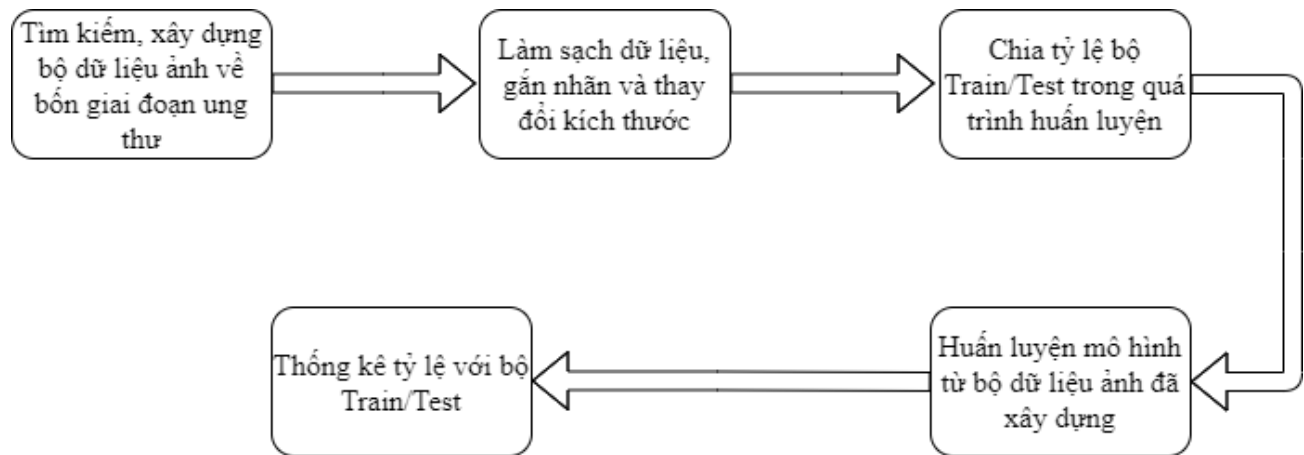
### CHƯƠNG III: ỨNG DỤNG HỌC SÂU ĐỂ PHÂN LOẠI ẢNH ỨNG THU VÚ

#### 1. Phương pháp gia tăng dữ liệu

Gia tăng dữ liệu thường được dùng trong thị giác máy tính. Thị giác máy tính là một lĩnh vực bao gồm các phương pháp thu nhận, xử lý ảnh kỹ thuật số, phân tích và nhận dạng các hình ảnh, nói chung là dữ liệu đa chiều từ thế giới thực để cho ra các thông tin số hoặc biểu tượng. Ở thị giác máy tính, chúng ta có thể gần như chắc chắn lật, quay hoặc phản chiếu hình ảnh mà không gặp rủi ro trong việc thay đổi nhãn. (13)

Trong lúc thử nghiệm đã có ý tưởng là đánh giá thường xuyên sự thay đổi của hình ảnh thử nghiệm ban đầu bằng phương pháp thử nghiệm truyền thống để cải thiện việc phân loại một cách chính xác hơn. (13)

#### 2. Quy trình thực hiện huấn luyện mô hình

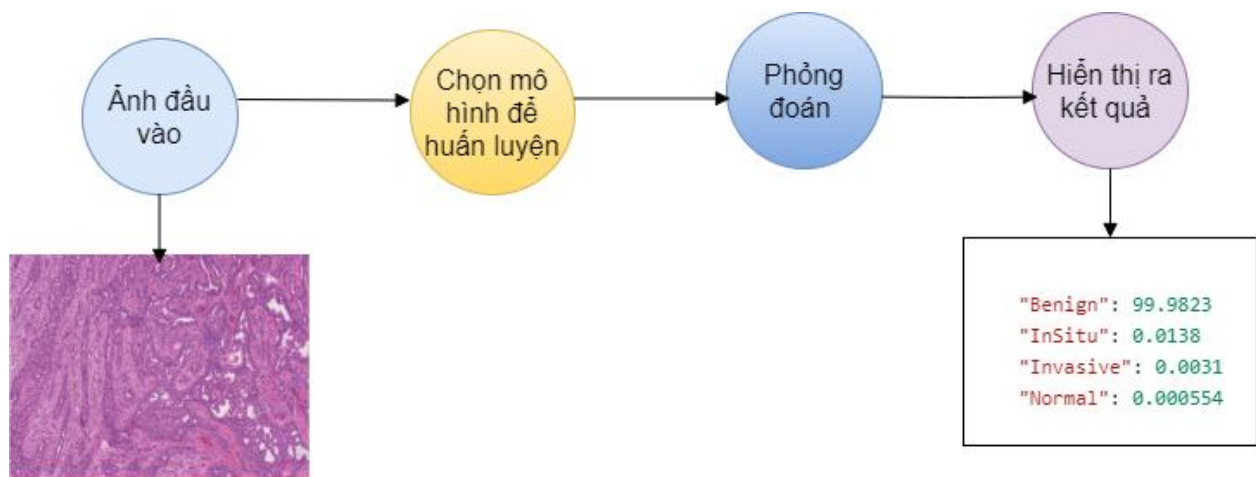


Hình 20 Các bước thực hiện train dữ liệu

Tìm kiếm và xây dựng dữ liệu qua ICIAR2018 BACH Challenge (The International Conference on Image Analysis and Recognition) (14) và Yan Rui, Ren Fei, Wang Zihao (24). Sau đó, làm sạch dữ liệu, loại bỏ những dữ liệu xấu đi chỉ lấy những dữ liệu tốt, và sau đó phân loại và gom nhóm các dữ liệu thành bốn nhãn Benign, In Situ, Invasive và Normal. Tiếp theo nhóm chia tỷ lệ cho bộ dữ liệu đó thành hai tập dữ liệu train và test để huấn luyện. Sau đó nhóm

cho bộ dữ liệu train học qua các mô hình. Sau khi học qua các mô hình xong thì nhóm đưa bộ dữ liệu test tiếp tục kiểm tra độ chính xác mô hình. Cuối cùng thì chỉ có ba mô hình có tỷ lệ chính xác cao nhất đó là DenseNet201, DenseNet121, Xception. Sau đó, nhóm sẽ thống kê tỷ lệ chính xác của hai bộ dữ liệu train và test ứng với ba mô hình nhóm đã sử dụng.

### 3. Quá trình sau khi huấn luyện mô hình



Hình 21 Quá trình test

Tập ảnh đầu vào là ảnh đã được gán nhãn. Trích xuất đặc trưng của tập dữ liệu rồi đưa vào mạng CNN để huấn luyện. Nhận dạng các giai đoạn của ung thư vú dựa trên các trích xuất đặc trưng của những hình ảnh thu được từ tập dữ liệu đã huấn luyện. Từ đó cho ra kết quả đó thuộc giai đoạn nào. Sau khi huấn luyện xong đóng gói thành file HDF5 để tiện sử dụng. Tạo một WebAPI tải lại file HDF5 để đoán ra kết quả. Tạo UI (WebUI, Windows Form) để người dùng dễ dàng sử dụng.

## CHƯƠNG IV: THỰC NGHIỆM

Sau khi tìm được tập dữ liệu tốt, nhóm sử dụng tập dữ liệu này để huấn luyện mô hình trên Kaggle Notebook và Jupyter. Sau khi huấn luyện xong, có được những mô hình có độ chính xác cao nhóm sử dụng IDE Pycharm và sử dụng Framework Django bằng ngôn ngữ Python để viết API. Tiếp theo nhóm muốn hiển thị lên web nên nhóm quyết định sử dụng IDE Visual Studio Code và Framework Reactjs bằng ngôn ngữ JavaScript để xây dựng giao diện. Bên cạnh đó, nhóm còn tạo ra ứng dụng Winform bằng ngôn ngữ C-Sharp.

### 1. Môi trường thực hiện hệ thống

#### 1.1 Kaggle

Kaggle là một chi nhánh của Google, hoạt động như một cộng đồng các nhà khoa học dữ liệu và các nhà phát triển. Đây là doanh nghiệp lớn chuyên hoạt động về những dịch vụ và các sản phẩm có liên quan đến mạng Internet như là dịch vụ công nghệ quảng cáo trực tuyến, các công cụ tìm kiếm thông tin, phần cứng, phần mềm. (25)

Kaggle triển khai về dịch vụ nền tảng bộ dữ liệu công cộng và tại đây mọi người ở bất kỳ quốc gia nào đều có thể tự do chia sẻ những bộ dữ liệu cho nhau. (25)

Kaggle Kernels là một nền tảng không tốn phí để chạy môi trường Jupyter notebooks trình duyệt. Điều này có nghĩa là bạn có thể lưu lại những cấu hình trong notebooks của bạn trên trình duyệt miễn là máy tính của bạn có kết nối Internet. (25)



Hình 22 Hình ảnh về Kaggle (26)

## 1.2 Jupyter

Jupyter là một công cụ mã nguồn mở miễn phí với mục đích nhắm đến khoa học dữ liệu và giáo dục, giúp mọi người cùng học Python dễ dàng hơn. Jupyter có tính tương tác nên có thể sử dụng làm môi trường chạy thử và giảng dạy. (27)

Jupyter notebook là công cụ cho phép bạn đưa cả code Python và các thành phần văn bản phức tạp như hình ảnh, công thức, video, biểu thức... vào trong cùng một file giúp cho việc trình bày trở nên dễ hiểu hơn, giống như một file trình chiếu nhưng lại có thể thực hiện chạy code tương tác trên đó. Các file notebook này có thể được chia sẻ với mọi người và có thể thực hiện lại các công đoạn một cách nhanh chóng và chính xác như những gì tác giả của file này tạo ra (27).



Hình 23 Hình ảnh về Jupyter (28)

## 1.3 Tài nguyên sử dụng

### 1.3.1 CPU

CPU (Central Processing Unit) là bộ xử lý trung tâm của máy tính. CPU xử lý tất cả các lệnh mà nó nhận được từ phần cứng và phần mềm chạy trên máy tính. (29)

### 1.3.2 GPU

GPU là từ viết tắt của Graphics Processing Unit, hiểu một cách đơn giản thì nó là đơn vị xử lý đồ họa chuyên dụng, nhiệm vụ chính là tăng tốc, xử lý đồ họa cho bộ xử lý của CPU. Còn GPU sở hữu những tính năng vượt trội, vượt xa so với các trình điều khiển đồ họa cơ bản. (30)

### 1.3.3 TPU

Bộ xử lý tensor (TPU) là một loại bộ xử lý độc quyền được google thiết kế vào năm 2016 để sử dụng với mạng neural và trong các dự án học máy. (31)

## 2. Framework

### 2.1 Django

Django là một web framework miễn phí mã nguồn mở được viết bằng ngôn ngữ Python. Django sử dụng mô hình Model-View-Control (MVC). Django được phát triển bởi Django Software Foundation (DSF) – một tổ chức phi lợi nhuận độc lập. (32)

Mục tiêu chính của Django là đơn giản hóa việc tạo các website phức tạp có sử dụng cơ sở dữ liệu. Django tập trung vào tính năng “có thể tái sử dụng” và “có thể tự chạy” của các component, tính năng phát triển nhanh, không làm lại những gì đã làm. Một số website phổ biến được xây dựng từ Django là Pinterest, Instagram, Mozilla và Bitbucket. (32)

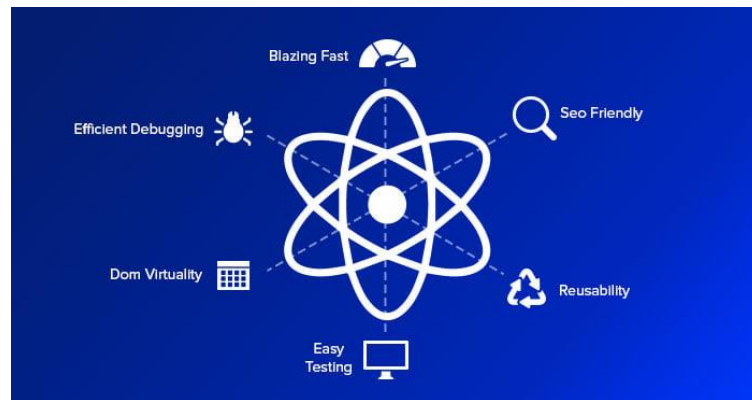


Hình 24 Hình ảnh về Framework Django (33)

## 2.2 ReactJS

ReactJS là thư viện JavaScript phổ biến nhất để xây dựng giao diện người dùng (UI). Nó cho tốc độ phản hồi tuyệt vời khi người dùng nhập liệu bằng cách sử dụng phương pháp mới để render trang web. (34)

Hàm render chính là mô tả cụ thể của giao diện người dùng tại bất cứ thời điểm nào. Vì thế nếu dữ liệu thay đổi, ReactJS sẽ thực hiện việc cập nhật giao diện người dùng với dữ liệu tương ứng. (34)



Hình 25 Hình ảnh về ReactJS (35)

## 2.3 Winform Application

Winform là cách cơ bản để cung cấp các thành phần giao diện trong môi trường .NET Framework, được xây dựng trên Windows API, cơ bản bao gồm là một form là khung dùng hiển thị thông tin của người dùng, các tính năng khác thì được đặt trong form và được lập trình để đáp ứng sự kiện. (36)

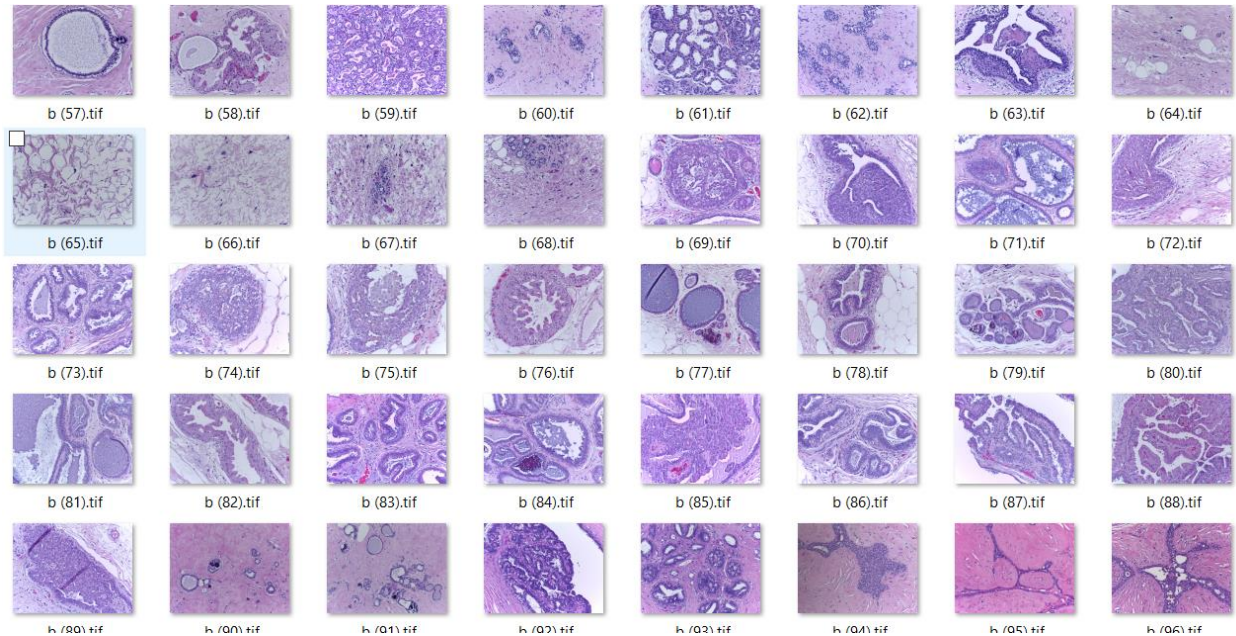


Hình 26 Hình ảnh về Winform (37)



### 3. Tập dữ liệu

Những hình ảnh mô bệnh học có vai trò rất quan trọng trong việc chẩn đoán cũng như là tiêu chuẩn để phân biệt xem khối u đó là lành tính hay ác tính, đã di căn hay chưa. Các mẫu khối u được lấy thường nhuộm bằng Hematoxylin và Eosin (H&E) và kiểm tra dưới kính hiển vi quang học. Những hình ảnh này sau mỗi lần được học sẽ cung cấp kết quả về chẩn đoán giai đoạn của bệnh chính xác hơn.



Hình 27 Một vài hình ảnh về tập dữ liệu Benign

#### 3.1 Thu thập dữ liệu

Sau quá trình tìm kiếm dữ liệu trên Internet, nhóm đã tìm được tập dữ liệu từ BACH của ICIAR 2018 (14) và Yan Rui (15) :

##### 3.1.1 Tập dữ liệu BACH của ICIAR 2018

Tổng cộng gồm 400 ảnh có kích thước 2048x1536 pixels dưới dạng \*.tif bao gồm bốn nhãn: Normal (100 ảnh), Benign (100 ảnh), In Situ (100 ảnh) và Invasive (100 ảnh).



### 3.1.2 Tập dữ liệu của Yan Rui, Ren Fei, Wang Zihao (15)

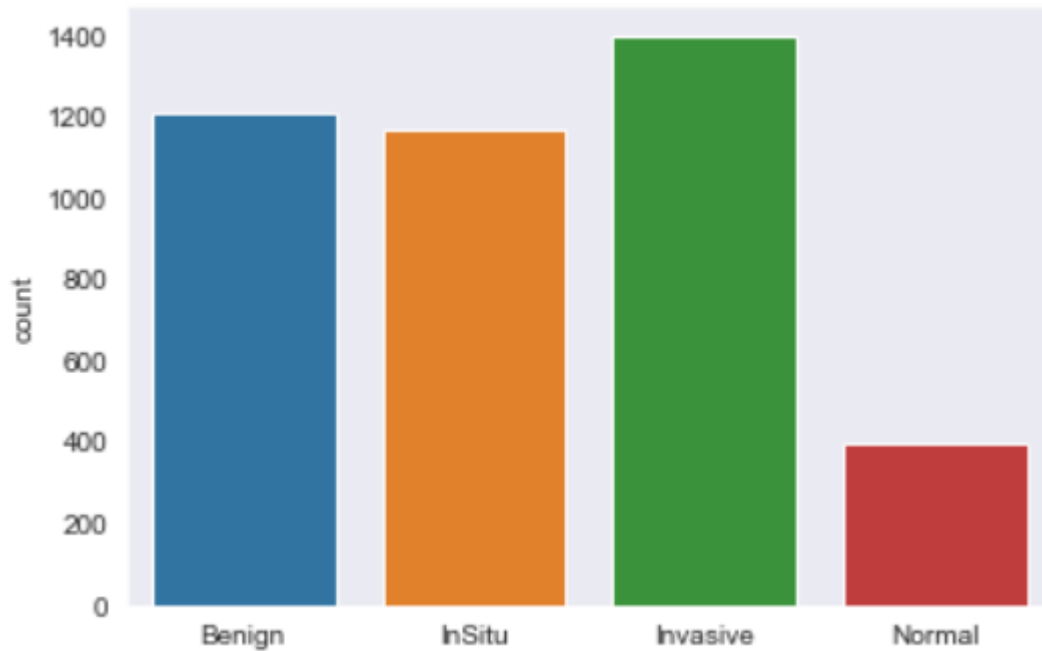
Tổng cộng gồm 3771 ảnh có kích thước 2048x1536 pixels bao gồm bốn nhãn: Normal (299 ảnh), Benign (1106 ảnh), In Situ (1066 ảnh) và Invasive (1300 ảnh).

### 3.1.3 Tổng hợp

Bộ dữ liệu hình ảnh của nhóm sau khi trộn hai tập dữ liệu thu thập được ở trên có 4171 ảnh với kích thước 2048x1536 pixels.

Đường dẫn dữ liệu sử dụng của nhóm tại đây (38)

Bộ dữ liệu phân bố theo từng nhãn được thể hiện dưới bảng sau :



Bảng 2 Biểu đồ cột thể hiện số lượng ảnh trong mỗi nhãn

Labels	Count
Benign	1206
InSitu	1166
InVasive	1400
Normal	399

Bảng 3 Bảng thống kê số lượng ảnh trong mỗi nhãn

### 3.2 Xử lý dữ liệu và đóng gói dữ liệu

Đầu tiên, nhóm thay đổi kích cỡ tất cả các ảnh về 224x224 pixels.

```
img_arr = cv2.imread(os.path.join(path, img), cv2.IMREAD_COLOR)
img_size = 224
resized_arr = cv2.resize(img_arr, (img_size, img_size))
```

Hình 28 Code minh họa thay đổi kích cỡ ảnh

Sau đó, co giãn (scale) hệ số màu của ảnh [0...225] thành [0...1] để mô hình dễ dàng hội tụ hơn.

```
# Normalize the data
x = np.array(x) / 255
```

Hình 29 Code minh họa co giãn ảnh

Kế tiếp chuyển đổi ảnh từ một chiều sang ba chiều để phù hợp với đầu vào của mô hình.

```
x = x.reshape(-1, img_size, img_size, 3)
y = np.array(y)
```

Hình 30 Chuyển đổi ảnh 1D sang 3D

Cuối cùng tập dữ liệu được đóng gói dưới dạng \*.h5 (HDF5) cho tốc độ đọc ghi nhanh và dễ dàng tái sử dụng để huấn luyện những mô hình khác có kích cỡ ảnh đầu vào là 224x224 pixels.

```
import numpy as np
import h5py
hf = h5py.File('BreastCancerdatanew4k.h5', 'a')
hf.close()
```

Hình 31 Đóng gói dữ liệu dưới dạng HDF5

### 3.3 Chia dữ liệu

Dữ liệu đã được đóng gói, nhóm tải lên môi trường Kaggle để thực hiện việc chia tập dữ liệu và huấn luyện.

```
x_train,x_test,y_train,y_test = train_test_split(
    x , y , test_size = 0.2 , stratify = y , random_state = 0)
```

Hình 32 Code minh họa chia dữ liệu

Train Set: 3336 ảnh, kích thước 224x224.

Test Set: 835 ảnh, kích thước 224x224.

#### 4. Kết quả

Để dễ dàng hiểu được các chỉ số trong huấn luyện mô hình thì nhóm xin giới thiệu sơ lược về phép đo Precision-Recall.

Precision thể hiện khả năng mô hình dự đoán đúng nhãn X, thành phần trong công thức khiến cho Precision tăng hay giảm không phải là TP mà là FP. Chính vì vậy, khi Precision cao đồng nghĩa là việc FP nhỏ hay số nhãn dự đoán nhầm sang nhãn X là thấp. (38)

$$Precision = \frac{TP}{TP + FP}$$

Hình 33 Công thức Precision (38)

Recall thể hiện khả năng mô hình dự đoán không bị sót bởi nhãn X, cũng như Precision thì Recall phụ thuộc vào FN hay nói cách khác là nó phụ thuộc vào khả năng mô hình dự đoán sai nhãn. (38)

$$Recall = \frac{TP}{TP + FN}$$

Hình 34 Công thức Recall (38)

Cách tính sử dụng độ chính xác chỉ cho chúng ta biết được bao nhiêu phần trăm chất lượng dữ liệu được phân loại mà không chỉ ra được cụ thể mỗi loại được phân loại như thế nào (lớp nào được phân loại đúng nhiều nhất và dữ liệu thuộc lớp nào thường bị phân loại nhầm vào lớp khác). Để có thể đánh giá các giá trị này, chúng ta sử dụng một ma trận được gọi là confusion matrix. (39)

Với giá trị Batch\_size = 64 và Epochs = 100 sau khi huấn luyện ba mô hình và cho ra kết quả độ chính xác tốt.

##### 4.1 Kết quả mô hình DensetNet201

Mô hình DenseNet201 được tạo ra theo thứ tự các lớp sau:

```
model.compile(optimizer = "adam", loss = 'categorical_crossentropy', metrics = ['accuracy'])
model.summary()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/densenet/densenet201_weights_tf_dim_ordering_tf_kern
els_notop.h5
74842112/74836368 [=====] - 1s 0us/step
Model: "sequential_1"
-----
Layer (type)                Output Shape                Param #
-----
densenet201 (Model)          (None, 7, 7, 1920)         18321984
-----
max_pooling2d_1 (MaxPooling2 (None, 3, 3, 1920)         0
-----
flatten_1 (Flatten)          (None, 17280)               0
-----
dense_1 (Dense)              (None, 4)                   69124
-----
Total params: 18,391,108
Trainable params: 18,094,340
Non-trainable params: 296,768
-----
```

Hình 35 Mô phỏng các lớp của mô hình DenseNet201

Sau khi tạo ra các lớp của mô hình DenseNet201, sau quá trình huấn luyện mô hình thì nhóm in ra giá trị chính xác và sai lệch của mô hình DenseNet201. Phần trăm độ chính xác của mô hình DenseNet201 là 88,14% và độ sai lệch của mô hình này là 0,44.

```
> print("Loss of the model is - ", model.evaluate(x_test,y_test)[0])
print("Accuracy of the model is - ", model.evaluate(x_test,y_test)[1]*100, "%")

27/27 [=====] - 1s 53ms/step - accuracy: 0.8814 - loss: 0.4415
Loss of the model is - 0.44150739908218384
27/27 [=====] - 2s 59ms/step - accuracy: 0.8814 - loss: 0.4415
Accuracy of the model is - 88.14371228218079 %
```

Hình 36 Phần trăm độ chính xác và sai lệch của mô hình DenseNet201

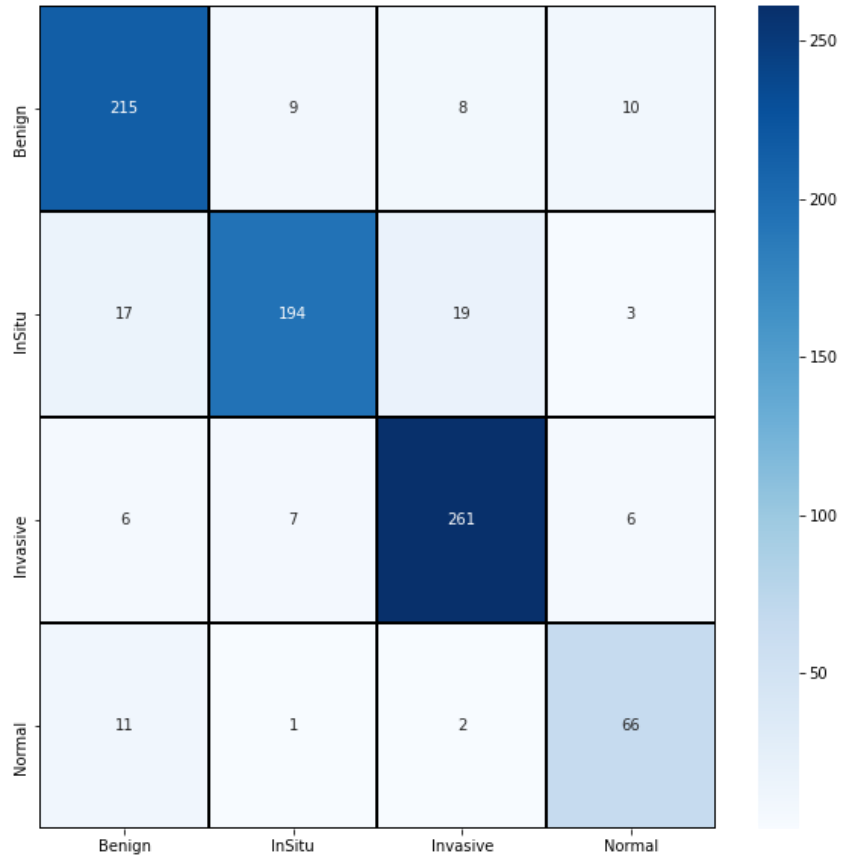
Thống kê các giá trị của precision, recall, f1-score, support của bốn nhãn của mô hình DenseNet201 và sau đó tính macro avg và weighted avg của bốn giá trị trên.

```
print(classification_report(y_test_inv, predictions, target_names = labels))
```

	precision	recall	f1-score	support
Benign	0.86	0.89	0.88	242
InSitu	0.92	0.83	0.87	233
Invasive	0.90	0.93	0.92	280
Normal	0.78	0.82	0.80	80
accuracy			0.88	835
macro avg	0.86	0.87	0.87	835
weighted avg	0.88	0.88	0.88	835

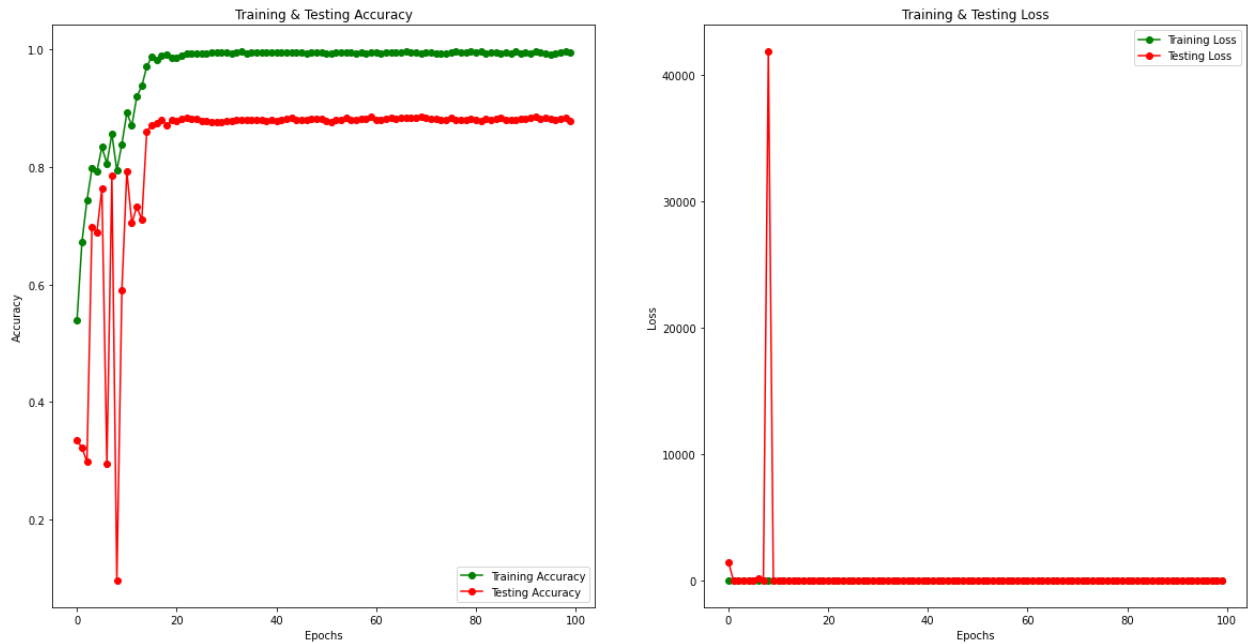
Hình 37 Thống kê độ chính xác qua 4 nhãn của mô hình DenseNet201

Sau đó, nhóm xuất ra confusion matrix để đánh giá độ chính xác của mô hình DenseNet201.



Hình 38 Confusion Matrix qua 4 nhãn của mô hình DenseNet201

Tiếp theo, vẽ ra hai biểu đồ biểu hiện độ chính xác và sai lệch của dữ liệu trong quá trình huấn luyện và sai lệch của mô hình DenseNet201.



Hình 39 Đồ thị thể hiện độ chính xác và sai lệch của dữ liệu đang học mô hình và đang kiểm thử mô hình DenseNet201

## 4.2 Kết quả mô hình DenseNet121

Mô hình DenseNet121 được tạo ra theo thứ tự các lớp sau:



```
[6]: model = tf.keras.models.load_model('../input/t1cn-model/BreastCancer224densenet121.h5')
model.summary()
```

```
Model: "sequential"
-----
Layer (type)                 Output Shape              Param #
-----
densenet121 (Model)          (None, 7, 7, 1024)       7037504
-----
max_pooling2d (MaxPooling2D) (None, 3, 3, 1024)       0
-----
flatten (Flatten)            (None, 9216)              0
-----
dense (Dense)                 (None, 4)                 36868
-----
Total params: 7,074,372
Trainable params: 6,923,012
Non-trainable params: 151,360
-----
```

Hình 40 Mô phỏng các layer của mô hình DenseNet 121

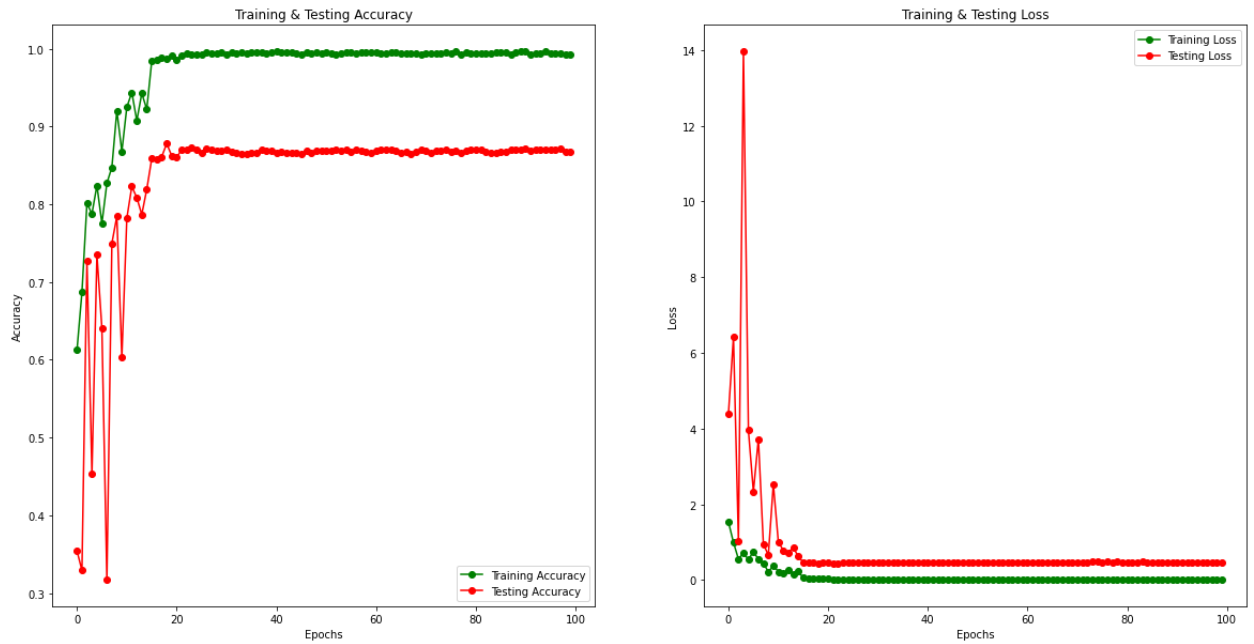
Sau khi tạo ra các lớp của mô hình DenseNet121, nhóm in ra giá trị chính xác và sai lệch của mô hình DenseNet121. Phần trăm độ chính xác của mô hình DenseNet121 là 88,98% và độ sai lệch của mô hình này là 0,41.

```
print("Loss of the model is - ", model.evaluate(x_test,y_test)[0])
print("Accuracy of the model is - ", model.evaluate(x_test,y_test)[1]*100, "%")
```

```
27/27 [=====] - 7s 247ms/step - loss: 0.4163 - accuracy: 0.8898
Loss of the model is - 0.41634616255760193
27/27 [=====] - 7s 260ms/step - loss: 0.4163 - accuracy: 0.8898
Accuracy of the model is - 88.98203372955322 %
```

Hình 41 Phần trăm độ chính xác và sai lệch của mô hình DenseNet121

Tiếp theo, vẽ ra hai biểu đồ biểu hiện độ chính xác và sai lệch của dữ liệu trong quá trình huấn luyện và sai lệch của mô hình DenseNet121.



Hình 42 Đồ thị biểu diễn độ chính xác và sai lệch của dữ liệu đang học và kiểm thử của mô hình DenseNet121

Thống kê các giá trị của precision, recall, f1-score, support của bốn nhãn của mô hình DenseNet121 và sau đó tính macro avg và weighted avg của bốn giá trị trên.

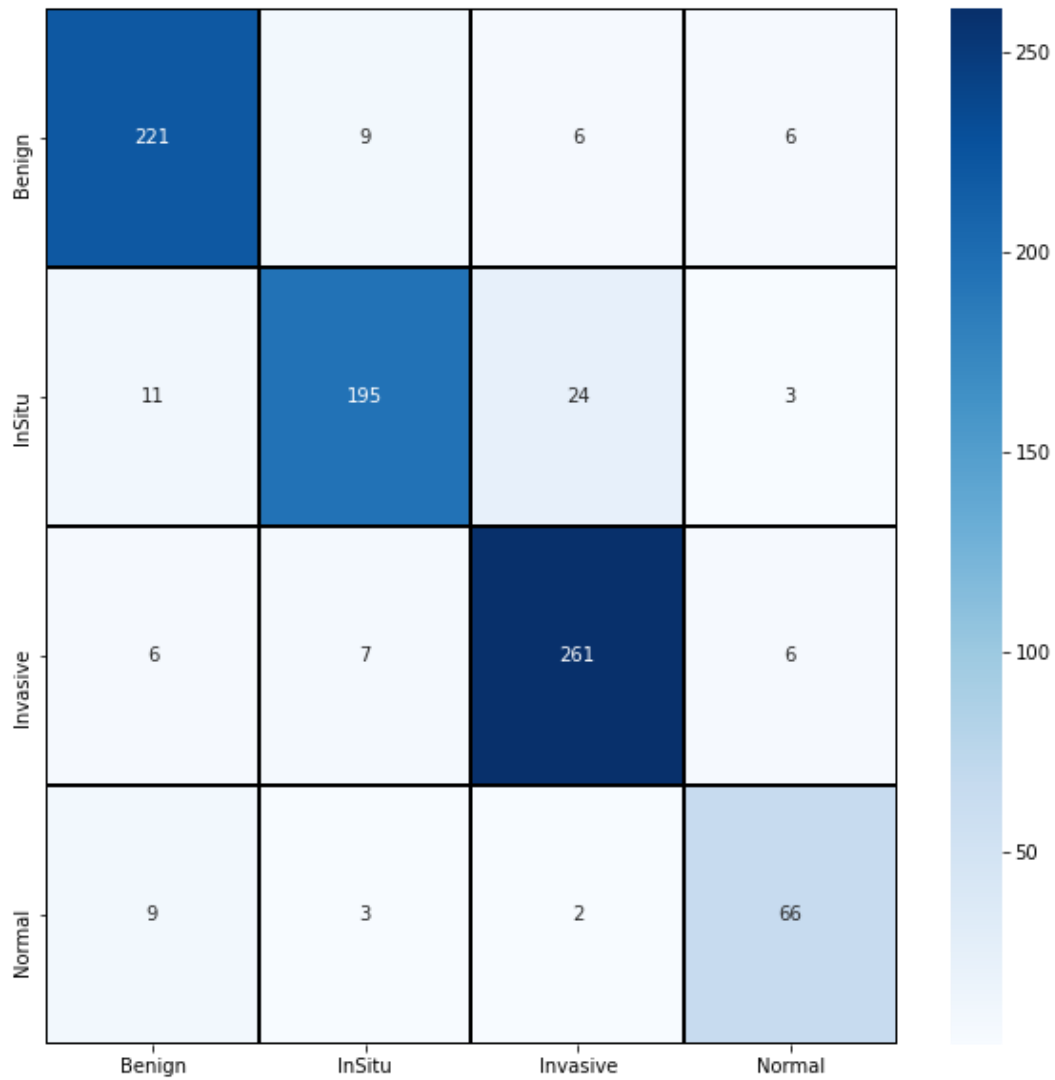
```
print(classification_report(y_test_inv, predictions, target_names = labels))
```

	precision	recall	f1-score	support
Benign	0.89	0.91	0.90	242
InSitu	0.91	0.84	0.87	233
Invasive	0.89	0.93	0.91	280
Normal	0.81	0.82	0.82	80
accuracy			0.89	835
macro avg	0.88	0.88	0.88	835
weighted avg	0.89	0.89	0.89	835

+ Code + Markdown

Hình 43 Thống kê độ chính xác qua 4 nhãn của mô hình DenseNet121

Sau đó, nhóm vẽ ra confusion matrix để đánh giá độ chính xác dự đoán của mô hình DenseNet121.



Hình 44 Confusion Matrix qua 4 nhãn của mô hình DenseNet121

### 4.3 Kết quả mô hình Xception

Mô hình Xception được tạo ra theo thứ tự các lớp sau :

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/xception/xception_weights_tf_dim_ordering_tf_kernels
_notop.h5
83689472/83683744 [=====] - 1s 0us/step
Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
xception (Model)            (None, 7, 7, 2048)       20861480
-----
max_pooling2d (MaxPooling2D) (None, 3, 3, 2048)       0
-----
flatten (Flatten)           (None, 18432)            0
-----
dense (Dense)               (None, 4)                73732
-----
Total params: 20,935,212
Trainable params: 20,791,500
Non-trainable params: 143,712
-----
```

Hình 45 Mô phỏng các lớp của mô hình Xception

Sau khi huấn luyện và đạt được kết quả ở mô hình Xception. In ra độ chính xác và sai lệch của mô hình Xception. Phần trăm độ chính xác của mô hình Xception là 88,4% còn độ sai lệch của mô hình Xception là 0.54.

```
> print("Loss of the model is - ", model.evaluate(x_test,y_test)[0])
print("Accuracy of the model is - ", model.evaluate(x_test,y_test)[1]*100 , "%")

27/27 [=====] - 1s 21ms/step - accuracy: 0.8838 - loss: 0.5428
Loss of the model is - 0.5428113341331482
27/27 [=====] - 1s 21ms/step - accuracy: 0.8838 - loss: 0.5428
Accuracy of the model is - 88.3832335472107 %
```

Hình 46 Phần trăm độ chính xác của và sai lệch của mô hình Xception

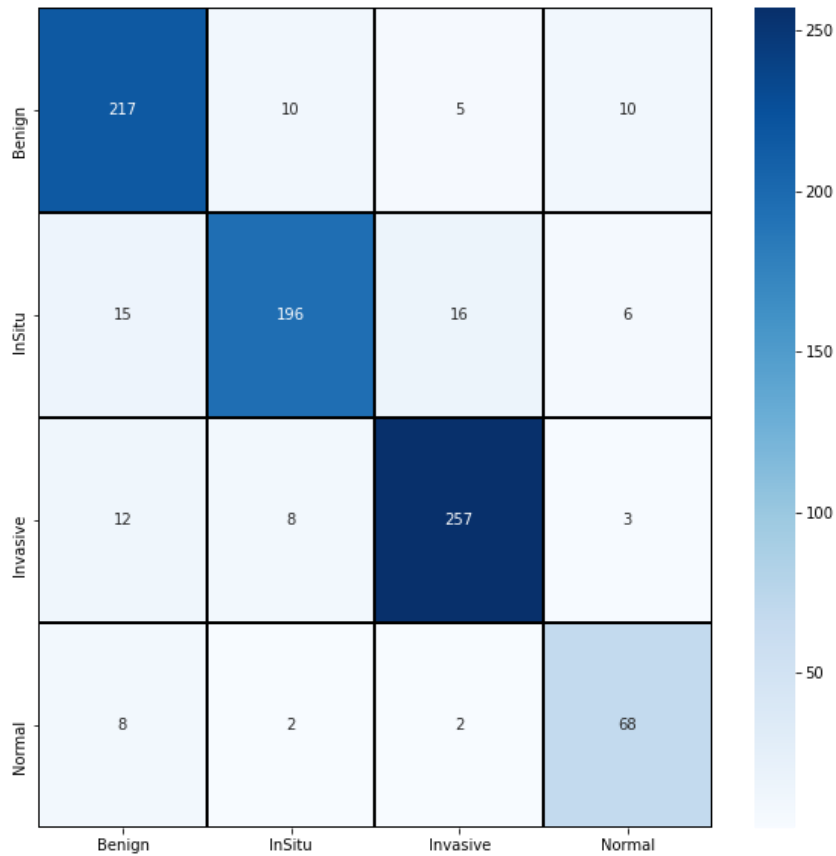
Thống kê các giá trị của precision, recall, f1-score, support của bốn nhãn của mô hình Xception và sau đó tính macro avg và weighted avg của bốn giá trị trên.

```
print(classification_report(y_test_inv, predictions, target_names = labels))
```

	precision	recall	f1-score	support
Benign	0.86	0.90	0.88	242
InSitu	0.91	0.84	0.87	233
Invasive	0.92	0.92	0.92	280
Normal	0.78	0.85	0.81	80
accuracy			0.88	835
macro avg	0.87	0.88	0.87	835
weighted avg	0.89	0.88	0.88	835

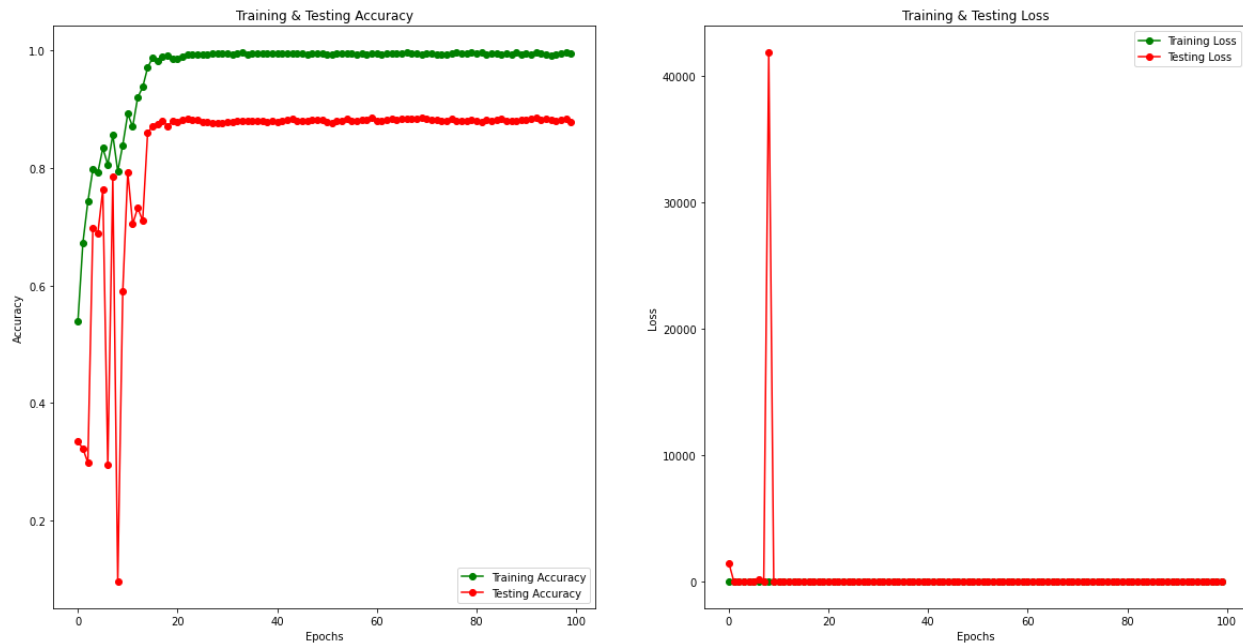
Hình 47 Thống kê độ chính xác qua 4 nhãn của mô hình Xception

Sau đó, nhóm vẽ ra confusion matrix để đánh giá độ chính xác dự đoán của mô hình Xception.



Hình 48 Confusion Matrix của bốn nhãn của mô hình Xception

Tiếp theo, vẽ ra hai biểu đồ biểu hiện độ chính xác và sai lệch của dữ liệu trong quá trình huấn luyện và sai lệch của mô hình Xception.



Hình 49 Đồ thị biểu diễn độ chính xác và độ sai lệch của mô hình Xception

#### 4.4 Đóng gói mô hình

Sau khi huấn luyện và đóng gói mô hình vào file .HDF5 để sử dụng vào nhiều mục đích khác nhau.

```
model.save('BreastCancer224densenet121.h5')
```

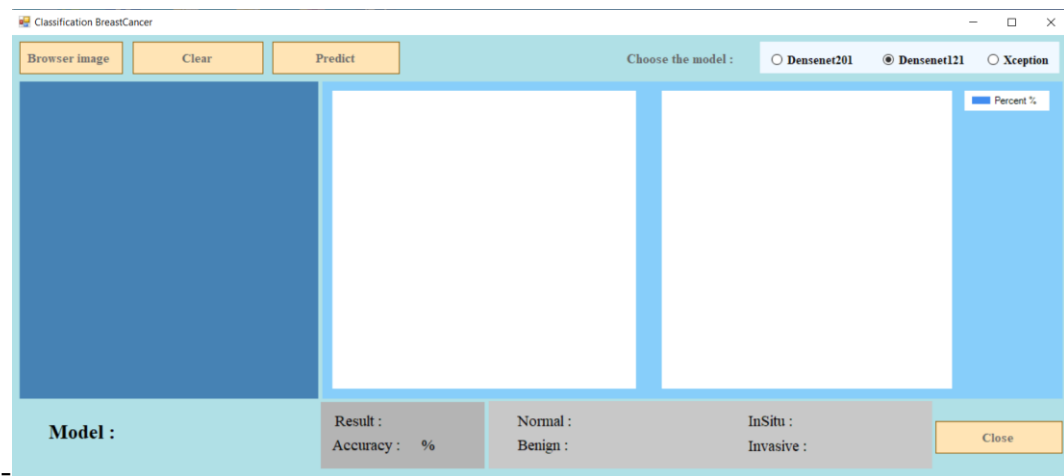
Hình 50 Code minh họa đóng gói

## 5. Giao diện

### 5.1. Winform Application

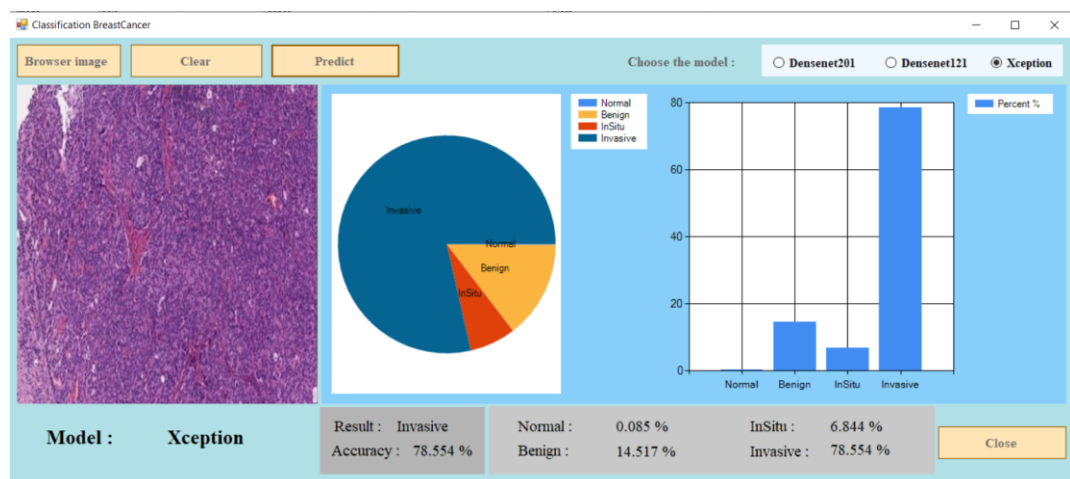


### 5.1.1 Giao diện trước khi phỏng đoán



Hình 51 Giao diện ứng dụng trước khi dự đoán

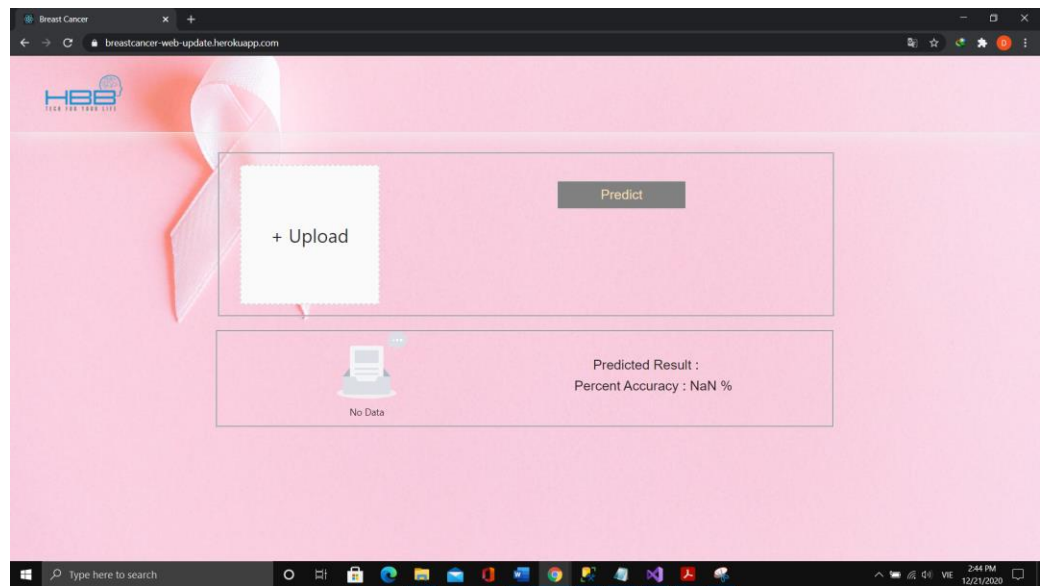
### 5.1.2 Giao diện sau khi phỏng đoán



Hình 52 Giao diện ứng dụng sau khi dự đoán ra kết quả

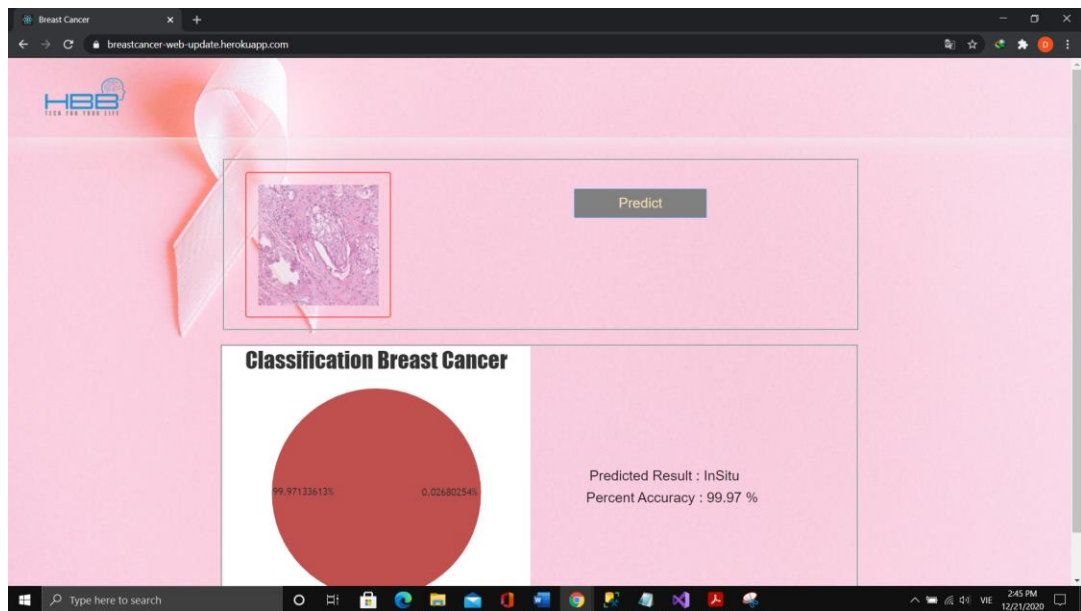
## 5.2. Web Application

### 5.2.1 Giao diện web trước khi dự đoán



Hình 53 Giao diện web sau khi dự đoán ra kết quả

### 5.2.2 Giao diện web sau khi dự đoán



Hình 54 Giao diện web sau khi dự đoán ra kết quả

## CHƯƠNG V: TỔNG KẾT VÀ HƯỚNG PHÁT TRIỂN

### 1 Kết quả đạt được

Kết quả nghiên cứu của tiểu luận chuyên ngành sử dụng kỹ thuật học sâu trong việc giải quyết bài toán phân loại ảnh ung thư vú đã đạt được kết quả như sau:

Hiểu rõ hơn về căn bệnh ung thư vú như triệu chứng, đặc điểm, các giai đoạn và trường hợp bệnh.

Tìm hiểu về học máy, học sâu, một số vấn đề cơ bản trong bài toán phân loại ung thư vú.

Có được kiến thức và áp dụng được học máy và học sâu.

Ứng dụng học máy và học sâu để phân tích dữ liệu, huấn luyện mô hình để xử lý bài toán phân loại ảnh ung thư vú.

Thu thập được tập dữ liệu và huấn luyện mô hình bằng Kaggle.

Sử dụng Pycharm để làm môi trường cho việc viết ngôn ngữ Python.

Sử dụng được Kaggle Notebook để huấn luyện trên các mô hình và trong đó có ba mô hình có tỷ lệ chính xác cao nhất là DenseNet201, DenseNet121, Xception.

Đã ứng dụng được những mô hình trên và tạo được hai giao diện gồm một Website và một ứng dụng Winform.

Về ưu điểm:

Tìm được dữ liệu ban đầu tốt (ít dữ liệu thô).

Huấn luyện ra được mô hình chính xác với tỷ lệ tương đối cao 89% cho cả 03 mô hình trên.

Về nhược điểm:

Đối với 04 nhãn gồm Benign, In Situ, Invasive, Normal thì dữ liệu để huấn luyện nhãn Normal còn khá ít nên tỷ lệ chính xác ở nhãn Normal không được đánh giá cao.

Chưa xây dựng được ứng dụng trên thiết bị di động.

### 2 Hướng phát triển

Trong tương lai, nhóm dự định sẽ hiệu chỉnh lại các mô hình cũ, sau đó cải tiến mô hình và đề xuất ra một mô hình mới chuyên dùng trong bài toán phân loại ảnh ung thư vú.

Tiếp theo nhóm sẽ phát triển phần mềm trên các hệ điều hành di động để người dùng có thể dễ dàng sử dụng.

Ứng dụng GAN để tăng số lượng dữ liệu, qua đó tăng độ hiệu quả cho mô hình.

## TÀI LIỆU THAM KHẢO

1. <https://www.komen.org/breast-cancer/facts-statistics/what-is-breast-cancer/>.  
<https://www.komen.org>. [Online]
2. <http://www.cancerbiomed.org/index.php/cocr/article/view/875/946#B37>.  
<http://www.cancerbiomed.org>. [Online]
3. D., Nguyen Ba. *Breast cancer situation in women in some provinces/cities from 2001 to 2007*. Vietnamese J Oncol . Vietnam : s.n., 2009.
4. Patterns, Sariego J. *breast cancer presentation in the United States*. US : Am Surg, 2009. pp. 9-50.
5. 2010, Australian Institute of Health and Welfare & Australasian Association of Cancer Registries. *Cancer in Australia: an overview*. Australia : AIHW, 2010.
6. <https://www.vinmec.com/vi/tin-tuc/thong-tin-suc-khoe/tien-luong-kha-nang-song-cua-benh-nhan-ung-thu-vu-theo-tung-giai-doan/>. <https://www.vinmec.com>. [Online]
7.  
[https://www.breastcancer.org/symptoms/benign?fbclid=IwAR1VVPARLD\\_fSylhCMEEmviT22hGOtwo8E4NccGODrBNvpTGwTPiIg7vLqq8#:~:text=Benign%20\(non%2Dcancerous\)%20breast,those%20caused%20by%20breast%20cancer](https://www.breastcancer.org/symptoms/benign?fbclid=IwAR1VVPARLD_fSylhCMEEmviT22hGOtwo8E4NccGODrBNvpTGwTPiIg7vLqq8#:~:text=Benign%20(non%2Dcancerous)%20breast,those%20caused%20by%20breast%20cancer). <https://www.breastcancer.org>. [Online]
8. <https://yhoccongdong.com/thongtin/ung-thu-bieu-mo-ong-tuyen-vu-tai-cho/#:~:text=Khi%20c%3%A1c%20t%E1%BA%BF%20b%C3%A0o%20ung,c%C3%B3%20ti%C3%AAn%20l%C6%B0%E1%BB%A3ng%20r%E1%BA%A5t%20t%E1%BB%91t>.  
<https://yhoccongdong.com>. [Online]
9. <https://www.komen.org/breast-cancer/facts-statistics/invasive-breast-cancer/?fbclid=IwAR1gwzgcYBUTCpVIDGrA4TSpsVMa3uC7b3uSg91XOqjqSQwzYf9YvvWnzeU>. <https://www.komen.org>. [Online]
10.  
[https://vi.wikipedia.org/wiki/H%E1%BB%8Dc\\_m%C3%A1y#:~:text=H%E1%BB%8Dc%20m%C3%A1y%20\(ti%E1%BA%BFng%20Anh%3A%20machine,nh%E1%BB%AFng%20v%E1%BA%A5n%20%C4%91%E1%BB%81%20c%E1%BB%A5%20th%E1%BB%83](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y#:~:text=H%E1%BB%8Dc%20m%C3%A1y%20(ti%E1%BA%BFng%20Anh%3A%20machine,nh%E1%BB%AFng%20v%E1%BA%A5n%20%C4%91%E1%BB%81%20c%E1%BB%A5%20th%E1%BB%83).  
<https://vi.wikipedia.org>. [Online]
11.  
[https://vi.wikipedia.org/wiki/H%E1%BB%8Dc\\_s%C3%A2u#:~:text=H%E1%BB%8Dc%20s%C3%A2u%20hay%20Th%C3%A2m%20h%E1%BB%8Dc,nhi%E1%BB%81u%20bi%E1%BA](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_s%C3%A2u#:~:text=H%E1%BB%8Dc%20s%C3%A2u%20hay%20Th%C3%A2m%20h%E1%BB%8Dc,nhi%E1%BB%81u%20bi%E1%BA)

- BFn%20%C4%91%E1%BB%95i%20phi%20tuy%E1%BA%BFn. <https://vi.wikipedia.org>. [Online]
12. HỒNG, TRẦN THỊ. *SỬ DỤNG KỸ THUẬT HỌC SÂU*. PGS. TS. TRẦN VĂN LĂNG. Đồng Nai : s.n., 2019. pp. 4-5, LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN.
13. Canh Phong Nguyen, Anh Hoang Vo. *Breast Cancer Histology Image Classification*. Bao Thien Nguyen. HCM : s.n., 2018. p. 2.
14. <https://iciar2018-challenge.grand-challenge.org/evaluation/challenge/leaderboard/>. <https://iciar2018-challenge.grand-challenge.org>. [Online] 2018.
15. [https://vi.wikipedia.org/wiki/H%E1%BB%8Dc\\_m%C3%A1y](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y). <https://vi.wikipedia.org>. [Online]
16. BRIAN BEERS, Peter Westfall. <https://www.investopedia.com/terms/r/regression.asp>. <https://www.investopedia.com>. [Online] 02 19, 2020.
17. <https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/>. <https://topdev.vn>. [Online]
18. Pham, Quoc. <https://pbcquoc.github.io/cnn/>. <https://pbcquoc.github.io>. [Online] 04 03, 2019.
19. Hiếu, Trung. <https://viblo.asia/p/computer-vision-object-detection-nhan-dien-vat-the-chi-voi-10-dong-code-su-dung-imageai-naQZRbdjZvx>. <https://viblo.asia>. [Online] 07 15, 2018.
20. <https://dlapplications.github.io/2018-07-06-CNN/>. <https://dlapplications.github.io>. [Online] 2018.
21. <https://keras.io/api/applications/>. <https://keras.io>. [Online]
22. <https://www.youtube.com/watch?v=nMBCSroJ7bY&t=31s>. <https://www.youtube.com>. [Online]
23. <https://www.youtube.com/watch?v=gOVMsY25D20>. <https://www.youtube.com>. [Online] 03 20, 2020.
24. Yan Rui, Ren Fei, Wang Zihao. [http://ear.ict.ac.cn/?page\\_id=1616](http://ear.ict.ac.cn/?page_id=1616). <http://ear.ict.ac.cn>. [Online] 2019.
25. Nguyễn, Hồng. <https://timviec365.vn/blog/kaggle-la-gi-new9205.html>. <https://timviec365.vn>. [Online] 10 10, 2020.
26. Kirkland, Eddie. <https://towardsdatascience.com/increasing-kaggle-revenue-analyzing-user-data-to-recommend-the-best-new-product-f93fddbb4e0f>. <https://towardsdatascience.com>. [Online] 11 27, 2019.

27. FirebirD. <https://allaravel.com/blog/jupyter-notebook-cong-cu-khong-thieu-khi-hoc-python>. <https://allaravel.com>. [Online] 07 21, 2019.
28. Commons, Wikimedia. [https://commons.wikimedia.org/wiki/File:Jupyter\\_logo.svg](https://commons.wikimedia.org/wiki/File:Jupyter_logo.svg). <https://commons.wikimedia.org>. [Online] Project Jupyter Contributors, 2017.
29. Nhat, Phuong. <https://quantrimang.com/cpu-la-gi-142876>. <https://quantrimang.com>. [Online]
30. <https://hoanghaphc.vn/gpu-la-gi>. <https://hoanghaphc.vn>. [Online] 12 07, 2018.
31. Analytics, Internet of Things. <https://filegi.com/tech-term/tensor-processing-unit-tpu-10124/>. <https://filegi.com>. [Online]
32. <https://vinasupport.com/django-la-gi-tong-quan-ve-django-framework/>. <https://vinasupport.com>. [Online] 10 25, 2020.
33. ToXiNgau. <https://blog.haposoft.com/series-django-p1-cai-dat-django/>. <https://blog.haposoft.com>. [Online] 07 26, 2018.
34. G, Hải. <https://www.hostinger.vn/huong-dan/react-la-gi-va-no-hoat-dong-nhu-the-nao/>. <https://www.hostinger.vn>. [Online] 02 23, 2020.
35. Teknotrait. <https://teknotrait.com/what-is-reactjs/>. <https://teknotrait.com>. [Online]
36. <https://sites.google.com/site/1074netshare/home/software-technology/programming/c/desktop-user-interfaces/windows-forms/chuong-1---gioi-thieu-windows-forms>. <https://sites.google.com>. [Online]
37. <http://www.montegodata.co.uk/Manual/CartEditor.htm>. <http://www.montegodata.co.uk>. [Online]
38. Mai, Chi. <https://viblo.asia/p/ml-06-chung-ta-nen-dung-so-lieu-nao-de-do-luong-mo-hinh-model-va-cac-van-de-phan-loai-gap-phai-nhu-the-nao-phan-2-aWj53Og156m>. <https://viblo.asia>. [Online] 07 16, 2018.
39. <https://machinelearningcoban.com/2017/08/31/evaluation/>. <https://machinelearningcoban.com>. [Online]