

**ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**BÁO CÁO BÀI TẬP LỚN XỬ LÝ ẢNH INT3404E 20  
NĂM HỌC 2023 – 2024**

**Sino-nom Character Recognition**

Nhóm 15: Ngô Tùng Lâm - K67-CA-CLC2  
Lương Thị Linh - K67-CA-CLC2  
Nguyễn Văn Sơn - K67-CA-CLC2  
Nguyễn Thị Thu Trang - K67-CA-CLC2

Giảng viên hướng dẫn: GS. Lê Thanh Hà  
ThS. Nguyễn Công Thương

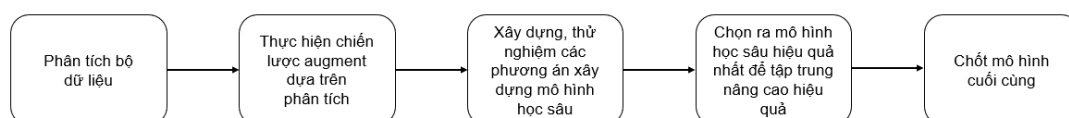
**HÀ NỘI - 2024**

# 1 Tổng quan bài toán

Bài toán cần giải quyết là phân loại kí tự Nôm, thuộc nhóm bài toán Image Classification (Phân loại ảnh). Mục tiêu chính của bài toán này đó chính là phân loại một hình ảnh kí tự Nôm đầu vào (input) thành một nhãn (label) đầu ra (output).

## 2 Giải quyết bài toán

Để giải quyết bài toán trên, nhóm đã xây dựng kế hoạch triển khai tổng thể gồm các bước như hình dưới đây:



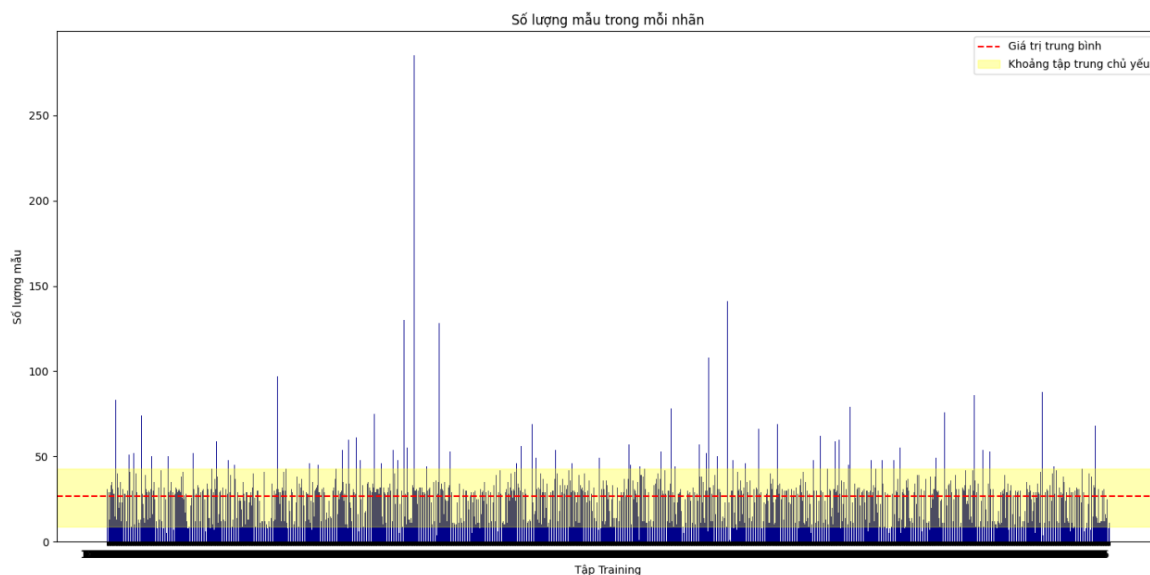
Hình 1: Sơ đồ pipeline cơ bản

### 2.1 Phân tích dữ liệu

#### 2.1.1 Phân bố dữ liệu giữa các nhãn

Phân tích chung :

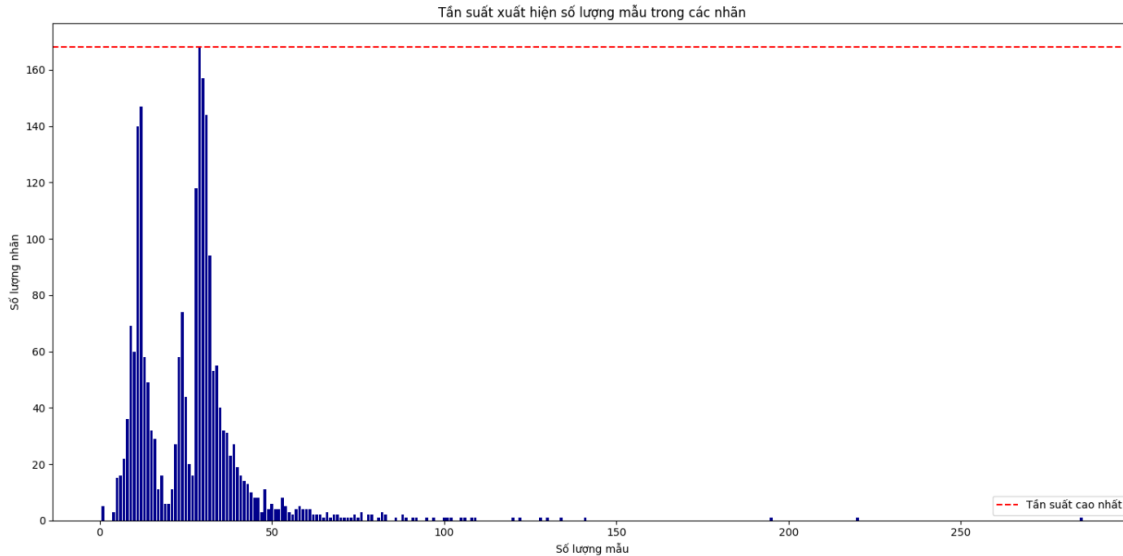
##### - Thống kê chi tiết



Hình 2: Biểu đồ phân bố số lượng mẫu giữa các nhãn

- Nhãn có nhiều mẫu nhất: 1677, số lượng mẫu: 285
- Nhãn có ít mẫu nhất: 954, số lượng mẫu: 1

- Giá trị trung bình: 26.6728
- Độ lệch chuẩn: 17.1145
- Dữ liệu tập trung chủ yếu trong khoảng từ 9 đến 43 ảnh.



Hình 3: Biểu đồ tần suất xuất hiện số lượng mẫu trong các nhãn

- Số lượng mẫu xuất hiện nhiều nhất trong các nhãn: 29
- Tần suất xuất hiện của số lượng mẫu này: 168

### Nhận xét :

- **Phân bố không đồng đều:** Phân bố dữ liệu giữa các nhãn hiện tại không đồng đều, có sự chênh lệch đáng kể giữa các nhãn. Điều này có thể dẫn đến mô hình bị bias và không tổng quát hóa tốt. Các nhãn có ít mẫu sẽ khó khăn hơn trong việc học các đặc trưng cụ thể của chúng.
- **Phân bố tập trung:** Có 1 số lượng lớn nhãn (168 nhãn) có đúng 29 mẫu, điều này cho thấy sự phân bố không đồng đều nhưng có một tập trung nhất định quanh giá trị này. Đây là 1 dấu hiệu tốt trong việc xác định số lượng mẫu có thể tăng cường để làm giảm độ chênh lệch.

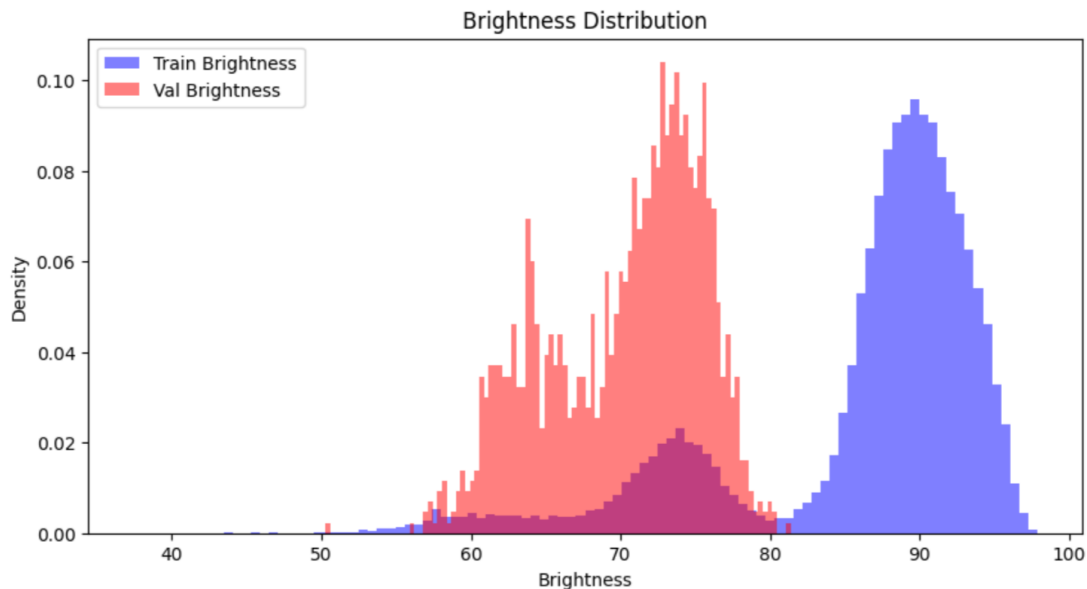
### Hướng giải quyết :

- **Data Augmentation:** Sử dụng kỹ thuật tăng cường dữ liệu để tăng số lượng mẫu cho các nhãn có ít mẫu đến gần với số lượng mẫu trung bình của tập dữ liệu. Mục tiêu là làm giảm sự chênh lệch về số lượng mẫu giữa các nhãn và tạo điều kiện cho mô hình học các đặc trưng một cách cân bằng hơn.
- **Số lượng mẫu tăng cường:** Có thể tăng cường dữ liệu cho các nhãn có ít hơn 30 mẫu bằng cách áp dụng các kỹ thuật tăng cường để đạt đến số lượng này. Chi tiết về các kỹ thuật sẽ được nhắc tới ở phần sau.

### 2.1.2 Phân bố độ sáng ảnh

#### Phân tích chung :

Để tính toán phân bố độ sáng của hình ảnh, nhóm đã duyệt qua tất cả các hình ảnh trong tập dữ liệu, chuyển đổi nó sang không gian màu LAB (vì nó phân tích thành phần sáng (L) ra khỏi các thành phần màu (A, B) giúp đo độ sáng không bị ảnh hưởng bởi màu sắc) và tính toán mức độ sáng trung bình. Sau khi tổng hợp tất cả giá trị độ sáng để tạo ra một phân phối thì nhóm đã minh họa dưới dạng biểu đồ KDE như hình:



Hình 4: Biểu đồ phân bố độ sáng

#### Nhận xét :

Việc phân bố độ sáng của tập huấn luyện rộng hơn so với tập validation là một ưu điểm. Điều này có nghĩa là mô hình đã được huấn luyện trên một phạm vi độ sáng đa dạng hơn, giúp nó có khả năng tổng quát hóa tốt hơn trên các điều kiện ánh sáng khác nhau.

#### Hướng giải quyết :

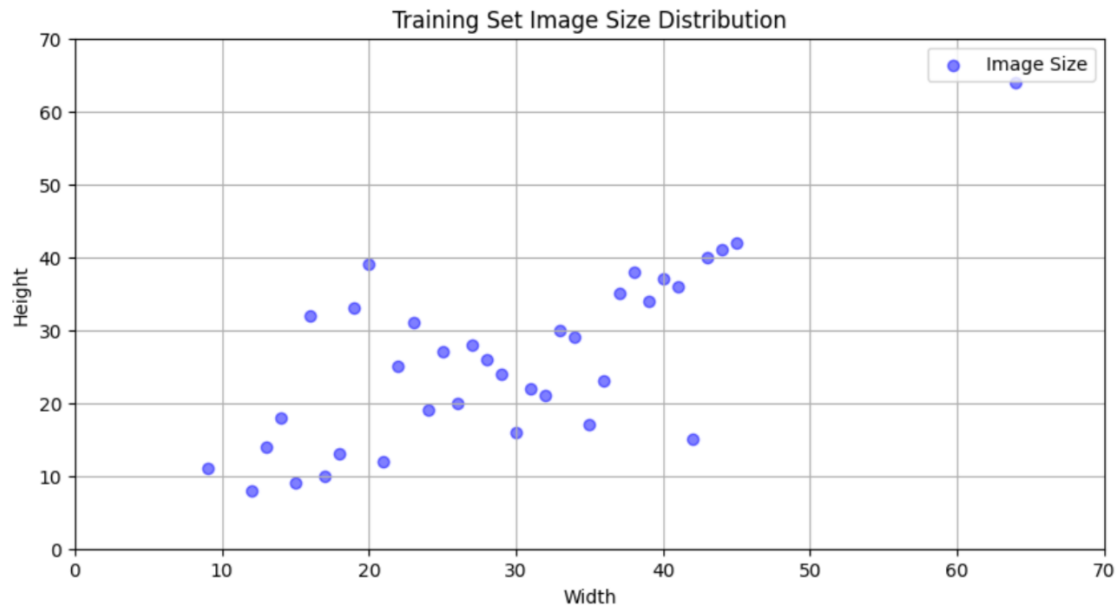
Không cần phải thay đổi nhiều trong quá trình huấn luyện của mô hình.

### 2.1.3 Phân bố chiều rộng và chiều cao ảnh

#### Phân tích chung :

Tập training:

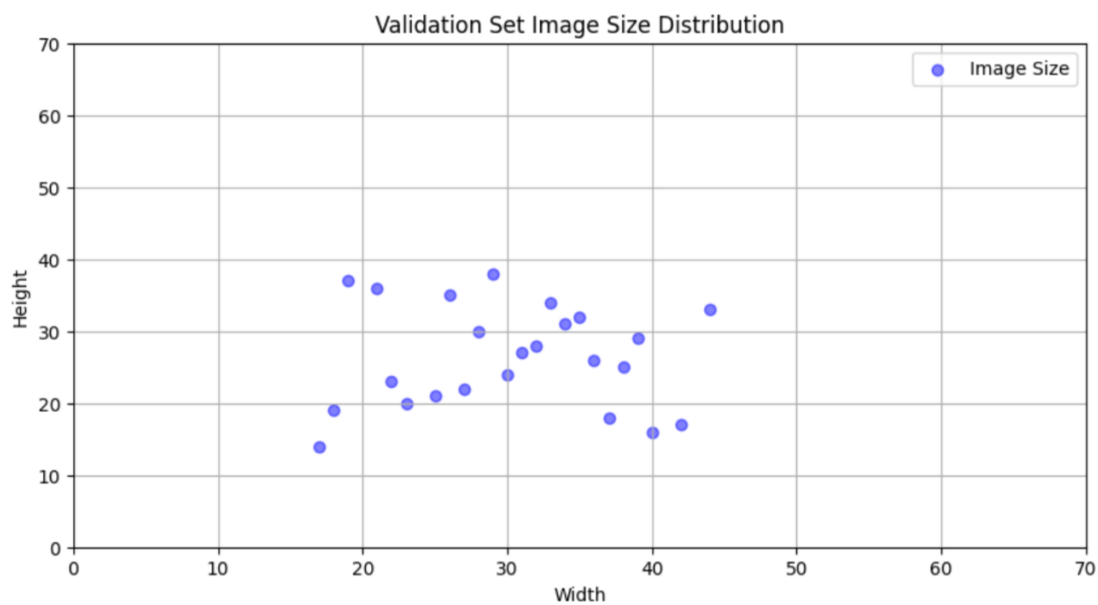
- Chiều rộng và chiều cao phổ biến nhất là 64 với tần suất là 45,532 ảnh.
- Chiều cao phân phối đa dạng với các giá trị từ 9 đến 45 pixels.
- Chiều rộng phân phối đa dạng từ 4 đến 42 pixels.



Hình 5: Biểu đồ phân bố kích thước của ảnh trong tập training

Tập validation:

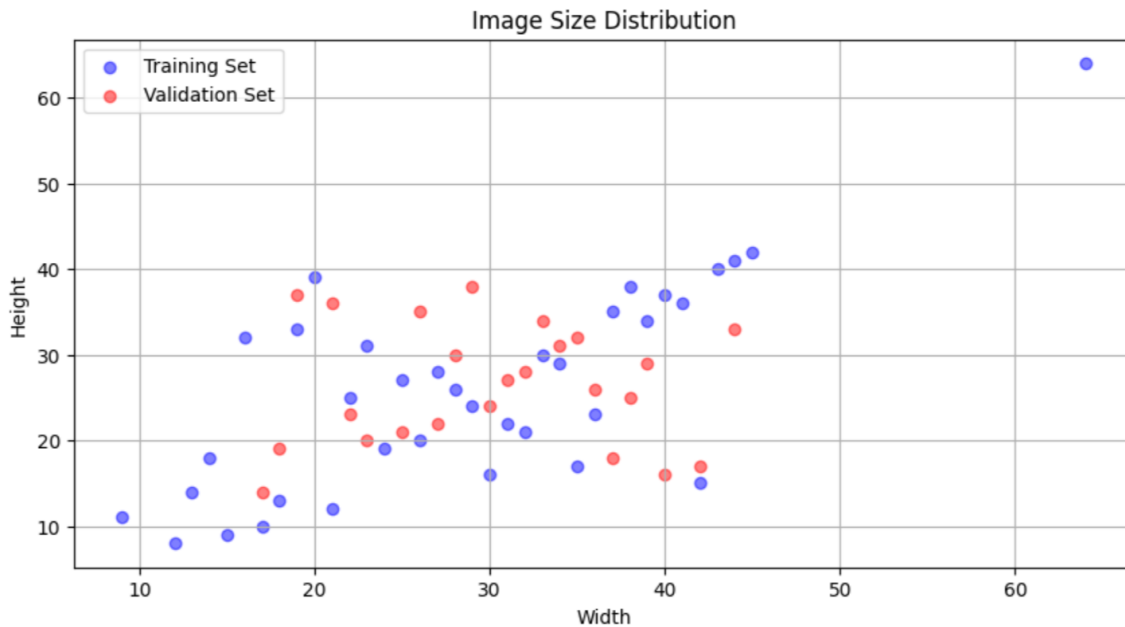
- Chiều rộng phổ biến nhất là 34 với tần suất là 154 ảnh.
- Chiều cao phổ biến nhất là 26 với tần suất là 202 ảnh.
- Cả chiều rộng và chiều cao phân phối đa dạng từ 4 đến 44 pixels và từ 14 đến 38 pixels.



Hình 6: Biểu đồ phân bố kích thước của ảnh trong tập validation

Kích thước ảnh:

- Tập training chứa 45,532 ảnh có kích thước 64x64, các ảnh còn lại có kích thước nhỏ hơn, nhỏ nhất là 13x9.
- Tập validation chứa 1,392 ảnh có kích thước nhỏ hơn 40x40, nhỏ nhất là 14x16.



Hình 7: Biểu đồ phân bố kích thước của ảnh trong tập training và tập validation

### Nhận xét :

Có sự khác biệt đáng kể trong phân phối chiều rộng và chiều cao giữa hai tập dữ liệu. Tập validation bao gồm các kết hợp chiều cao và chiều rộng mà tập training thiếu.

### Hướng giải quyết :

Nhóm thực hiện quá trình resize tất cả các ảnh trong cả hai tập về cùng một kích thước chuẩn là 224x224 pixels.

## 2.2 Tiền xử lý dữ liệu

### 2.2.1 Tăng cường dữ liệu

Để giảm sự chênh lệch đáng kể trong số lượng ảnh giữa các nhãn của tập train, việc tăng cường dữ liệu là một phương pháp quan trọng. Tăng cường dữ liệu giúp tạo ra một tập dữ liệu cân bằng hơn, từ đó cải thiện khả năng tổng quát hóa của mô hình và giảm thiểu hiện tượng overfitting. Phương pháp này tạo ra các bản sao biến đổi của dữ liệu hiện có bằng cách sử dụng các kỹ thuật như color jitter, random crop, random grayscale, Gaussian blur, random rotation, horizontal and vertical flip. Trải qua quá trình tìm hiểu và thử nghiệm, nhóm lựa chọn sử dụng 3 kỹ thuật phù hợp nhất với bài toán được của mình là color jitter, random grayscale và random rotation.

### 2.2.1.1 Các kỹ thuật augmentation cơ bản

Nhóm sử dụng 3 kỹ thuật tăng cường dữ liệu khác nhau bằng cách sử dụng các transform của thư viện torchvision. Cụ thể như sau:

- **RandomRotation**: Áp dụng xoay ngẫu nhiên cho mỗi ảnh trong khoảng  $(-15^\circ, 15^\circ)$ . Quá trình này giúp tăng cường dữ liệu bằng cách tạo ra các ảnh có góc nhìn khác nhau, từ đó làm tăng tính đa dạng của tập dữ liệu.



(a) Ảnh gốc



(b) Ảnh sau khi áp dụng RandomRotation

Hình 8: Kết quả ảnh trước và sau khi áp dụng RandomRotation

- **ColorJitter** : Thay đổi màu sắc của ảnh bằng cách điều chỉnh độ sáng (brightness), độ tương phản (contrast), độ bão hòa (saturation), và tông màu (hue) với giá trị ngẫu nhiên trong khoảng  $\pm 0.2$ . Quá trình này giúp tạo ra các biến thể về màu sắc của ảnh, từ đó làm tăng tính đa dạng của tập dữ liệu. Cụ thể:
  - Độ sáng (brightness): Điều chỉnh độ sáng của ảnh bằng cách tăng hoặc giảm độ sáng so với ảnh gốc. Với giá trị 0.2, độ sáng có thể được tăng lên tối đa 20
  - Độ tương phản (contrast): Điều chỉnh độ tương phản của ảnh bằng cách tăng hoặc giảm sự tương phản so với ảnh gốc. Giá trị 0.2 có nghĩa là độ tương phản có thể được tăng hoặc giảm tối đa 20
  - Độ bão hòa (saturation): Điều chỉnh độ bão hòa màu của ảnh bằng cách tăng hoặc giảm sự bão hòa so với ảnh gốc. Giá trị 0.2 có nghĩa là độ bão hòa màu có thể được tăng hoặc giảm tối đa 20
  - Tông màu (hue): Điều chỉnh tông màu của ảnh bằng cách thay đổi màu sắc của ảnh. Với giá trị 0.2, màu sắc có thể thay đổi tối đa  $0.2 * 180$  độ. Điều này thường dịch các màu trong không gian màu HSL.



(a) Ảnh gốc



(b) Ảnh sau khi áp dụng ColorJitter

Hình 9: Kết quả ảnh trước và sau khi áp dụng ColorJitter

- **RandomGrayscale**: Chuyển đổi một phần ảnh thành ảnh xám với xác suất 0.2. Quá trình này giúp tạo ra các biến thể về độ sáng và màu sắc của ảnh, từ đó cũng đóng góp vào tính đa dạng của tập dữ liệu.



(a) Ảnh gốc



(b) Ảnh sau khi áp dụng RandomGrayscale

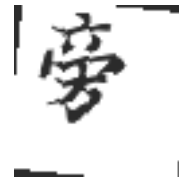
Hình 10: Kết quả ảnh trước và sau khi áp dụng RandomGrayscale

### 2.2.1.2 Kết hợp các kỹ thuật augmentation cơ bản

Nhóm kết hợp nhiều kỹ thuật cơ bản đã đề cập ở trên là RandomRotation, ColorJitter và RandomGrayscale trong cùng 1 bức ảnh để tạo ra các biến thể phong phú hơn cho dữ liệu. Cách tiếp cận này giúp tạo ra một tập dữ liệu đa dạng và phong phú hơn, giúp mô hình học được nhiều biểu diễn khác nhau của các chữ Nôm và cải thiện hiệu suất của mô hình trong quá trình nhận dạng.



(a) Ảnh gốc



(b) Ảnh sau khi áp dụng kết hợp các kỹ thuật augmentation cơ bản

Hình 11: Kết quả ảnh trước và sau khi áp dụng kết hợp các kỹ thuật augmentation cơ bản

### 2.2.1.3 Kết quả sau khi thực hiện tăng cường dữ liệu

Sau khi hoàn thành quá trình tăng cường dữ liệu, tập train của chúng tôi đã được mở rộng từ 56.813 ảnh lên 72.082 ảnh, tức là đã sinh thêm 15.269 ảnh mới. Tập val vẫn giữ nguyên số lượng 1.392 ảnh do việc tăng cường dữ liệu chỉ được thực hiện trên tập train. Việc áp dụng các kỹ thuật tăng cường dữ liệu đã giúp giảm đáng kể sự chênh lệch về số lượng ảnh giữa các nhãn trong tập train. Điều này góp phần tạo ra một tập dữ liệu cân bằng hơn, giúp mô hình học được nhiều biểu diễn khác nhau của các chữ Nôm và từ đó cải thiện khả năng tổng quát hóa cũng như hiệu suất của mô hình trong quá trình nhận dạng.

## 2.2.2 Chuẩn hóa hệ màu RGB, chuẩn hóa kích thước ảnh

### 2.2.2.1 Các hướng giải quyết

Sau khi phân tích tập dữ liệu được cung cấp, nhóm đã xác định hai hướng giải quyết để chuẩn hóa dữ liệu nhằm cải thiện độ chính xác của mô hình nhận dạng chữ Nôm.

#### Hướng giải quyết 1: Chuẩn hóa dữ liệu theo chuẩn ImageNet

Chuẩn hóa hệ màu RGB:

- Các giá trị trung bình: (0.485, 0.456, 0.406)
- Các giá trị độ lệch chuẩn: (0.229, 0.224, 0.225)

Chuẩn hóa kích thước ảnh: Tất cả các ảnh được resize về kích thước 224x224, phù hợp với kích thước ảnh đầu vào của nhiều mô hình học sâu tiêu chuẩn được huấn luyện trên ImageNet.



## Hướng giải quyết 2: Chuẩn hóa dữ liệu RGB dựa trên trung bình và độ lệch chuẩn của các kênh của tập training

Chuẩn hóa hệ màu RGB:

- Các giá trị trung bình: (5.3396e-07, 5.31996e-07, 5.2775e-07)
- Các giá trị độ lệch chuẩn: (1.5747e-07, 1.5842e-07, 1.6069e-07)

Chuẩn hóa kích thước ảnh: Tương tự với hướng giải quyết 1, các ảnh cũng được resize về kích thước 224x224.

### 2.2.2.2 Thử nghiệm và đánh giá

Kết quả của quá trình thử nghiệm và đánh giá cho thấy việc chuẩn hóa dữ liệu RGB và kích thước ảnh theo chuẩn ImageNet đã cung cấp dự đoán có độ chính xác cao hơn trên tập validation. Như vậy, trong bài toán này, việc sử dụng chuẩn hóa dữ liệu theo chuẩn ImageNet đã cải thiện hiệu suất của mô hình so với việc sử dụng chuẩn hóa dựa trên trung bình và độ lệch chuẩn của tập train.

### 2.2.2.3 Kết quả sau khi thực hiện chuẩn hóa hệ màu RGB và chuẩn hóa kích thước

Sau quá trình thử nghiệm và đánh giá, nhóm quyết định chuẩn hóa hệ màu RGB và chuẩn hóa kích thước trên tập dữ liệu được cung cấp theo chuẩn ImageNet.



Hình 12: Kết quả ảnh trước và sau khi chuẩn hóa

## 2.3 Xây dựng mô hình

### 2.3.1 Xây dựng mô hình CNN từ đầu

#### Tổng quan về mô hình CNN

Mạng neural tích chập gồm 1 lớp input, 1 lớp output, kèm theo các lớp ẩn bao gồm lớp convolutional, pooling, fully connected và normalization. CNN xử lý hình ảnh gốc trực

tiếp mà không cần tiền xử lý phức tạp. Các lớp của CNN có các neuron sắp xếp theo 3 chiều: chiều rộng, chiều cao và chiều sâu, giúp giảm hình ảnh thành một vector điểm số lớp. Trong nhận dạng đối tượng hình ảnh, CNN thường đạt hiệu suất tốt và bảo toàn mối quan hệ hàng xóm và tính cục bộ không gian, phát triển tốt với hình ảnh có kích thước thực tế và không phụ thuộc vào số chiều đầu vào.

### Hướng tiếp cận của nhóm

Nhóm đã tìm hiểu những bài viết, bài nghiên cứu về chủ đề nhận diện, phân loại các ký tự Nôm, ký tự Trung Quốc (do có sự tương đồng giữa ký tự Nôm và ký tự Trung Quốc) và các bài toán liên quan để xây dựng nên một mô hình CNN hoàn chỉnh. Dưới đây là kiến trúc chung của mô hình:

Conv-32
Pool
Conv-64
Pool
Conv-128
Pool
Conv-256
Pool
Conv-256
Pool
Conv-512
Pool
FC1024
Dropout 0.25
FC1024
Dropout 0.25
FC2139
Softmax

Bảng 1: Kiến trúc chung của mô hình CNN

### Kết luận

Sau quá trình thử nghiệm, kết quả cho thấy mô hình học chậm, tỉ lệ chính xác thấp, không phù hợp với bài toán và tập dữ liệu được giao. Nhóm đã quyết định thử cách tiếp cận khác: sử dụng các mô hình có độ phức tạp lớn hơn (các pretrained model) và thực hiện tinh chỉnh trên các mô hình đó.

#### 2.3.2 Transfer learning sử dụng các pretrained model

Nhóm đã nghiên cứu, tiến hành thử nghiệm transfer learning trên 5 pretrained model là YOLOv8, VGG16, ResNet50, ResNet18, EfficientNet-B0. Và dưới đây là một số nghiên cứu của nhóm về các mô hình trên và kết quả thử nghiệm:

##### 2.3.2.1 YOLOv8

YOLO là mô hình phát hiện đối tượng nổi trội về tốc độ và độ chính xác, được giới

thiệu vào năm 2016 bởi Joseph Redmon và cộng sự và YOLOv8 hiện là phiên bản mới nhất.

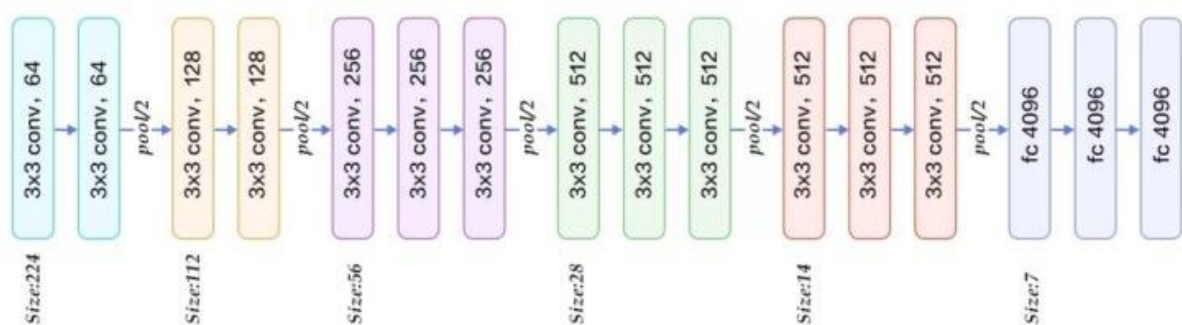
Mô hình phân loại hình ảnh YOLOv8 được thiết kế để nhận diện 1000 lớp đã được định nghĩa trước trong hình ảnh theo thời gian thực.

Tuy nhiên, thuật toán YOLO có một số hạn chế như khó khăn trong việc nhận diện các đối tượng nhỏ và không thể thực hiện phân loại chi tiết.

### 2.3.2.2 VGG16

Được phát triển bởi Karen Simonyan và Andrew Zisserman thuộc Đại học Oxford vào năm 2014, với kiến trúc đơn giản nhưng hiệu quả cao trong các nhiệm vụ nhận dạng hình ảnh, đặc biệt là phân loại hình ảnh.

- **Kiến trúc chủ đạo:** Thay vì sử dụng một kernel size lớn, VGG chỉ sử dụng các kernel size 3x3 cho tất cả lớp convolution, với stride 1 và padding, tối đa hóa sự đa dạng các đặc trưng học được hơn vì sử dụng 3 lần biến đổi phi tuyến.
- **Hiệu suất:** Model sau khi train bởi mạng VGG16 đạt độ chính xác 92.7% top-5 test trong dữ liệu ImageNet gồm 14 triệu hình ảnh thuộc 1000 lớp khác nhau.
- **Kích thước mô hình:** Mô hình VGG nhìn chung là một mô hình có số lượng tham số cực lớn với hơn 128 triệu tham số, không gian lưu trữ của mô hình ước tính khoảng trên 600Mb.



Hình 13: Cấu trúc mô hình VGG16

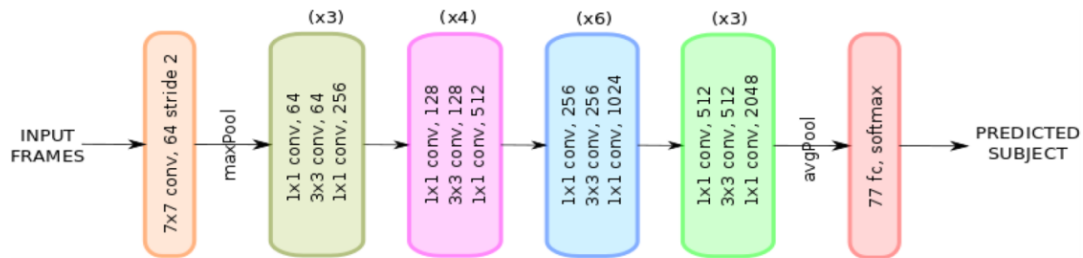
### 2.3.2.3 ResNet50

Được giới thiệu bởi Kaiming He và các đồng nghiệp vào năm 2015, ResNet50 là một mô hình mạng nơ-ron tích chập sâu (CNN) với 50 lớp, nổi bật với các khối residual giúp giải quyết vấn đề vanishing gradient, cho phép huấn luyện các mạng rất sâu hiệu quả hơn.

- **Kiến trúc chủ đạo:** ResNet50 được xây dựng từ 50 lớp, trong đó chủ yếu là các khối bottleneck. Các khối này giúp tăng độ sâu của mạng mà không làm mất đi hiệu suất, nhờ vào việc sử dụng các kết nối tắt (skip connections). Các lớp convolution, batch normalization và hàm kích hoạt ReLU được kết hợp lại để trích xuất và học các đặc trưng từ dữ liệu hình ảnh, giúp mạng nơ-ron có khả năng phân loại và nhận dạng hiệu quả.

- **Hiệu suất:** Trên tập dữ liệu ImageNet, nó đạt được độ chính xác top-1 là khoảng 76.15% và độ chính xác top-5 là khoảng 92.87%. Các con số này thể hiện ResNet50 là một trong những mô hình hiệu quả nhất trong bài toán phân loại ảnh, đặc biệt là trên bộ dữ liệu lớn và phức tạp như ImageNet.
- **Kích thước mô hình:** ResNet50 có khoảng 25.6 triệu tham số, làm cho nó lớn hơn nhiều so với các mô hình CNN cổ điển nhưng vẫn dễ quản lý hơn so với các phiên bản ResNet sâu hơn như ResNet101 hay ResNet152. Việc triển khai và sử dụng ResNet50 đòi hỏi phần cứng với khả năng xử lý tốt, như GPU, để tối ưu hóa thời gian đào tạo và suy luận.

Dưới đây là hình ảnh minh họa về cấu trúc của ResNet50:



Hình 14: Cấu trúc mô hình Resnet50

#### 2.3.2.4 ResNet18

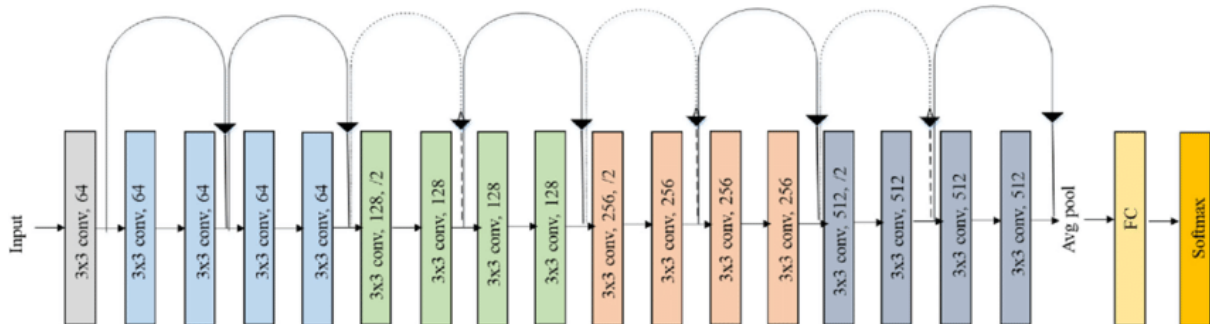
ResNet18 là một kiến trúc mạng nổi tiếng trong loạt mô hình ResNet (Residual Network), được giới thiệu bởi Microsoft Research vào năm 2015. Nó là một trong những mô hình đầu tiên sử dụng khối residual để giải quyết vấn đề của việc huấn luyện mạng sâu.

Một số điểm nổi bật về ResNet18 bao gồm:

- **Kiến trúc chủ đạo :** ResNet18 sử dụng một kiến trúc mạng sâu với 18 lớp, bao gồm các lớp convolution, batch normalization và hàm kích hoạt ReLU (Rectified Linear Unit). Kiến trúc của nó bao gồm các khối cơ bản và các khối bottleneck, giúp giảm độ sâu của mạng mà vẫn giữ được hiệu suất.
- **Hiệu suất:** ResNet18 đạt được hiệu suất tốt trên nhiều tác vụ khác nhau trong lĩnh vực thị giác máy tính. Trên tập dữ liệu ImageNet, nó đạt được độ chính xác top-1 là khoảng 69.76% và độ chính xác top-5 là khoảng 89.08%.

- **Kích thước mô hình:** ResNet18 có kích thước khá nhỏ so với các mô hình mạng sâu khác, chỉ có khoảng 11.7 triệu tham số. Điều này khiến nó trở thành một lựa chọn phổ biến cho các ứng dụng có hạn chế về tài nguyên tính toán.

Dưới đây là hình ảnh minh họa về cấu trúc của ResNet18:



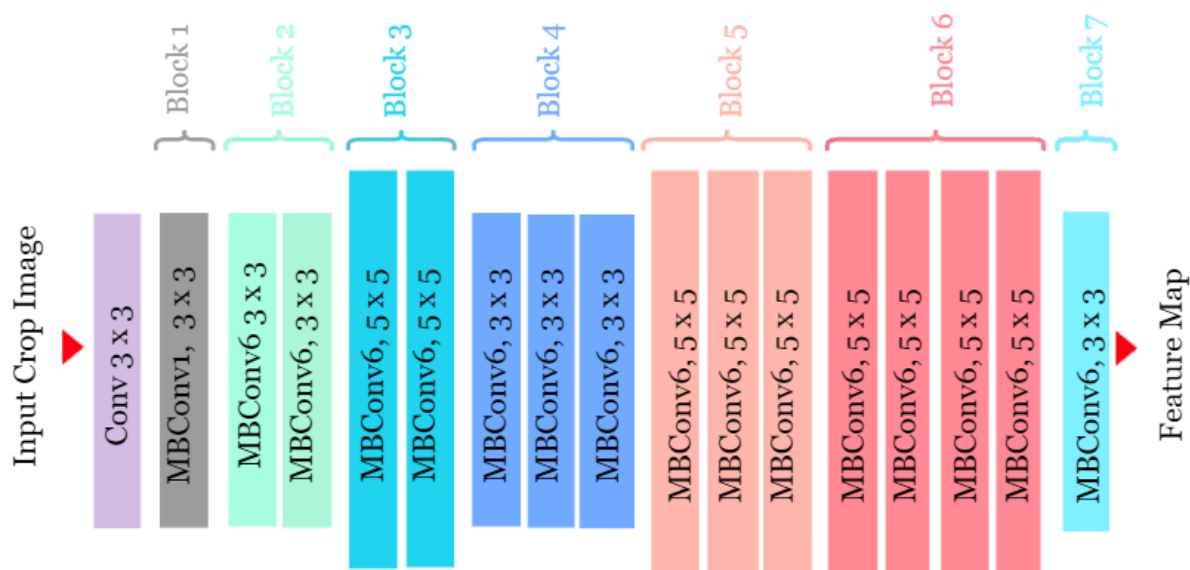
Hình 15: Cấu trúc mô hình Resnet18

### 2.3.2.5 EfficientNet-B0

Là một trong những mô hình nổi tiếng thuộc dòng EfficientNet, được giới thiệu bởi Google AI vào năm 2019, hiệu suất vượt trội so với các mô hình trước đó, đồng thời sử dụng ít tài nguyên hơn. Một số điểm nổi bật về mô hình này bao gồm:

- **Kiến trúc chủ đạo:** sử dụng 'Compound Scaling', trong đó 3 yếu tố chính của mạng (chiều sâu, chiều rộng, và độ phân giải ảnh) được tỷ lệ hóa một cách cân đối.
- **Hiệu suất:** đạt hiệu suất cao trên nhiều tác vụ khác nhau, đặc biệt là trong nhận dạng ảnh, nó đạt 77.1% top-1 accuracy và 93.3% top-5 accuracy trên ImageNet với ít tham số và phép tính hơn so với các mô hình trước đây như ResNet hay Inception.
- **Kích thước mô hình:** khoảng 5.3 triệu tham số và yêu cầu khoảng 0.39 tỷ FLOPs, khiến nó trở thành 1 trong những mô hình nhẹ và hiệu quả nhất

Hình ảnh mô hình của cấu trúc:



Hình 16: Cấu trúc mô hình EfficientNet-B0

### 2.3.3 Kết quả thử nghiệm

Sau quá trình thử nghiệm huấn luyện dựa trên bộ dữ liệu gốc, ta thu được kết quả như sau:

Model	Kết quả
YoloV8	Không phù hợp, accuracy rất thấp
VGG16	Accuracy đạt 40,1% trên tập validation sau khi train 100 epochs
ResNet50	Accuracy đạt 89.95% trên tập validation sau khi train 100 epochs
ResNet18	Accuracy đạt 90,37% trên tập validation sau khi train 100 epochs
EfficientNetB0	Accuracy đạt 93,03% trên tập validation sau khi train 100 epochs

Bảng 2: Tổng quan kết quả của các pretrained models thử nghiệm

#### Nhận xét trong quá trình thử nghiệm:

Trong quá trình thử nghiệm, nhóm nhận thấy rằng bộ dữ liệu của mình không tương đồng với bộ dữ liệu ImageNet. Do đó, nếu chỉ huấn luyện layer cuối cùng và cố định các layer trước của mô hình pretrained, độ chính xác của mô hình sẽ bị giảm. Vì vậy, nhóm quyết định để toàn bộ các layer có thể thay đổi trong quá trình huấn luyện, mặc dù phải đánh đổi thời gian huấn luyện lâu hơn để mô hình có thể học hiệu quả hơn.

#### Kết luận:

Theo quan sát của nhóm, EfficientNet-B0 là pretrained model hiệu quả nhất với accuracy trên cả 2 tập dữ liệu training và validation cao vượt trội so với các mô hình còn lại, thời gian chạy ngắn. Vì vậy, nhóm chọn mô hình trên để tập trung huấn luyện nhằm cải thiện hiệu quả học của mô hình

## 2.4 Tập trung huấn luyện mô hình, cải thiện hiệu quả dự đoán

Để tăng độ chính xác của mô hình đã chọn, nhóm đã sử dụng một vài phương pháp sau:

- Sử dụng bộ dữ liệu đã được tăng cường
- Thử nghiệm thay đổi các hyperparameters
- Thử nghiệm optimizer (Adam và SGD) theo từng giai đoạn huấn luyện

Cuối cùng, khả năng dự đoán của mô hình đã tăng, accuracy trên tập validation đạt 95,47%

### 3 Kết quả và thảo luận

Sau khi thử nghiệm trên tập dữ liệu kiểm tra của giáo viên, mô hình của nhóm đã đạt độ chính xác 95,8% trên 3300 ảnh ký tự. EfficientNet-B0 là một mô hình học sâu hiệu quả, với số lượng tham số ít và yêu cầu ít FLOPs, khiến nó trở thành một mô hình nhẹ và hiệu suất cao so với các mô hình khác. Ngoài ra, mô hình này còn đạt hiệu quả tốt trong việc xử lý các ảnh có kích thước nhỏ.