**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KHOA HỌC MÁY TÍNH**

Logo, company name

Description automatically generated

**BÁO CÁO ĐỒ ÁN**

**NHẬP MÔN THỊ GIÁC MÁY TÍNH**

**CS331.M21.KHCL**

***ĐỀ TÀI*: PHÂN LOẠI THỜI TIẾT**

| **GIẢNG VIÊN HƯỚNG DẪN:** | **TS. MAI TIẾN DŨNG** |
| --- | --- |
| **SINH VIÊN THỰC HIỆN:** | **LÊ NGUYỄN TIẾN ĐẠT– MSSV: 20521167** |

**TP. HỒ CHÍ MINH, 2/2022**

**MỤC LỤC**

I. Giới thiệu chung về đề tài 3

1.Lý do chọn đề tài 3

2.Input và output 3

3.Bộ Data 4

II. Giới thiệu về Data Augmentation 4

1.Khái niệm 4

2.Ứng dụng vào bài toán 5

3.Một số ví dụ đặc trưng7

III. Giới thiệu về model CNN 8

1.Khái niệm 8

2.Cách hoạt động của CNN9

IV. Giới thiệu về Transfer Learning và VGG16 10

1. Khái niệm Transfer Learning 11

2. VGG16 12

V. Kết quả 14

VI. So sánh 20

Tài liệu tham khảo 21

**I.Giới thiệu chung về đề tài**

**1.Lý do chọn đề tài**

Việc nhận diện được các hiện tượng thời tiết thường có ảnh hưởng rất lớn tới nhiều khía cạnh khác nhau trong đời sống của chúng ta, ví dụ như dự báo thời tiết, đánh giá tình trạng của những con đường, giao thông, nông nghiệp, quản lý rừng bền vững và việc nhận diện các môi trường tự nhiên. Tuy nhiên, hiện nay lại có rất ít đề tài hay bài báo hướng đến việc phân loại những hiện tượng tự nhiên, chủ yếu quan san bằng mắt thường của con người.

Vì vậy, đề tài mà em hướng tới trong đồ án môn học này là vô cùng cần thiết tới cuộc sống thường ngày. Nhóm chúng em có đưa ra hai phương pháp để giải quyết vấn đề trên, cụ thể đó là một model CNN được điều chỉnh một số chức năng cho phù hợp với bài toán, hai là sử dụng model VGG16. Chúng em chọn 2 phương pháp này chủ yếu vì lý do như sau:

* CNN là mô hình để phân lớp rất hiệu quả trên 1 tập dữ liệu lớn,nên nó có khả năng cho 1 kết quả vô cùng chính xác.
* VGG16 là một model từng đạt giải thưởng cao trong cuộc thi phân loại ảnh vào năm 2015, nó mang lại độ chính xác cao và thời gian chạy vô cùng tốt nên được áp dụng nhiều vào các bài toán nhận diện và phân loại ảnh.

**2.Input và output**

* Input: tập data gồm 5 loại thời tiết gồm rainy (mưa), cloudy (nhiều mây), foggy (có sương mù), shine (chói sáng), sunrise (mặt trời mọc) từ tập data Multi-class weather dataset (MWD). Các bức ảnh chỉ gồm một hiện tượng thời tiết duy nhất.
* Output: Là nhãn của các hiện tượng thời tiết như sau: Rainy, Cloudy, Foggy, Shine, Sunrise.

**3.Bộ Data**

Bộ data Multi-class Weather Dataset (MWD) là bộ data lớn được sử dụng trong bài báo nghiên cứu mang tên “Multi-class weather recognition from still image using heterogeneous ensemble method”. Bộ dataset cung cấp một tập các bức ảnh về các hiện tượng thời tiết được trích xuất bởi nhiều phương thức khác nhau để có thể dự đoán được nhiều điều kiện thời tiết khác nhau.



II.Giới thiệu về kỹ thuật Data Augmentation

Với mục đích tăng cường dữ liệu cho tập train, bọn em

III.Giới thiệu về Model CNN

**1.Khái niệm**

CNN hay convolutional neural network (CNN) là 1 phương thức của mạng học sâu (Deep Neural Network). CNN là 1 thuật toán Machine Learning mà đầu vào là một bức ảnh cùng với một số dữ liệu khác như weights hay biases hay nhiều vật thể có trong một bức ảnh và đầu ra của nó có thể phân biệt vật thể này với những vật thể còn lại.

CNN hoạt động bằng cách trích xuất features từ những bức ảnh. Bất cứ CNN nào cũng bao gồm các bước sau đây:

1. Lớp Input là một bức ảnh xám hoặc ảnh màu.

2. Lớp Output là 2 hay nhiều nhãn chưa class

3. Lớp Hidden bao gồm nhiều lớp convolution, lớp ReLU, lớp pooling và một fully-connected Neural Network.

Ở đồ án này, chúng em sẽ thiết kế một mạng Neural Network dùng để nhận diện các hiện tượng thời tiết sử dụng một dataset để train bao gồm 3348 bức ảnh với 5 loại hiện tượng thời tiết ( rainy, cloudy, sunrise, shine, foggy).

**2. Áp dụng thuật toán CNN vào bài toán**

Model CNN bao gồm có 2 bước chính: Feature extraction và classification.

Feature extraction là giai đoạn mà các filters và lớp sẽ được dùng để trích xuất thông tin từ những bức ảnh và một khi hoàn thành hết các bước trên sẽ tự chuyển tới giai đoạn tiếp theo là classification nơi mà những bức ảnh sẽ được phân loại dựa vào các nhãn input của bài toán.

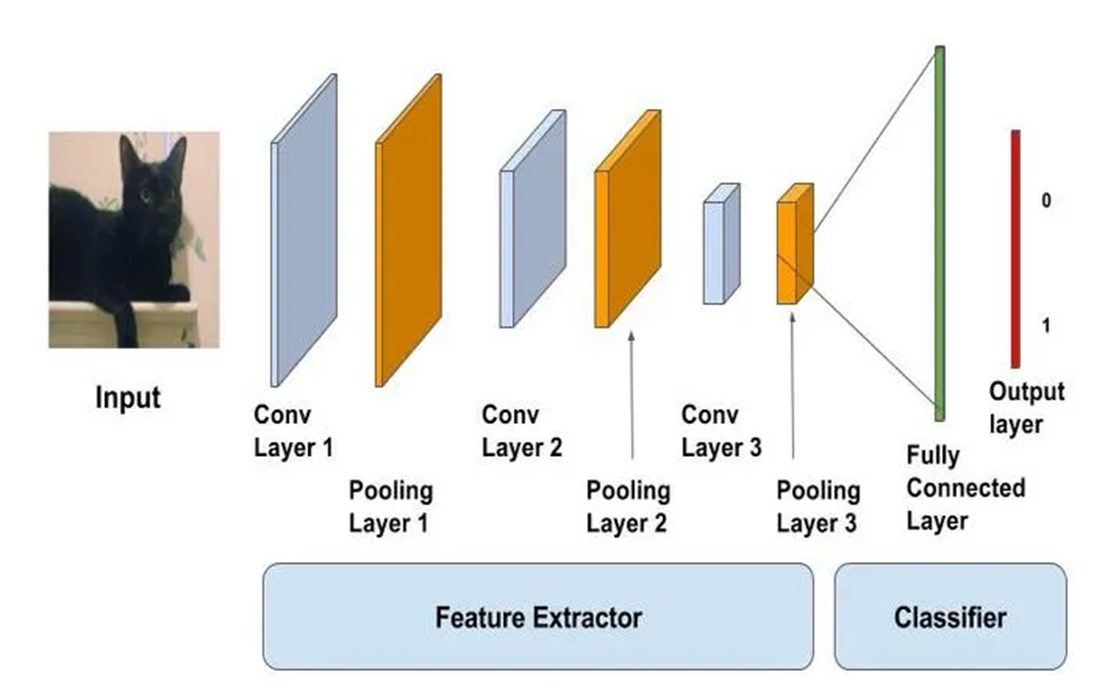
Một mạng CNN thường sẽ như sau:

1. Input layer

2. Convolution layer + Activation function

3. Pooling layer

4. Fully – Connected Layer



Ở đồ án này chúng em sẽ áp dụng một model CNN đơn giản với 2 CNN blocks gọi là Model 2.

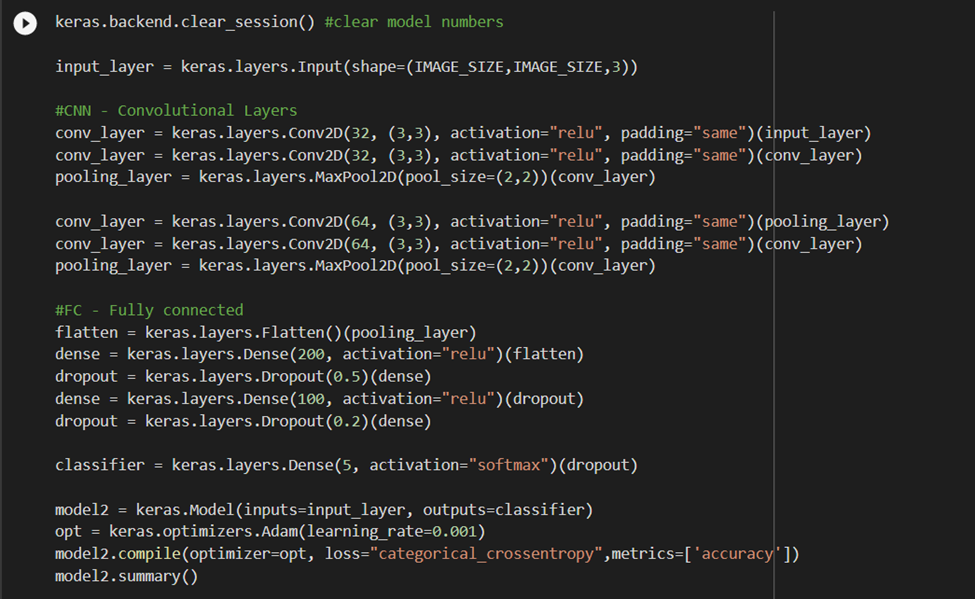
**Kiến trúc của Model 2**

Chúng ta sẽ sử dụng 2 khổi convolutional như sau:

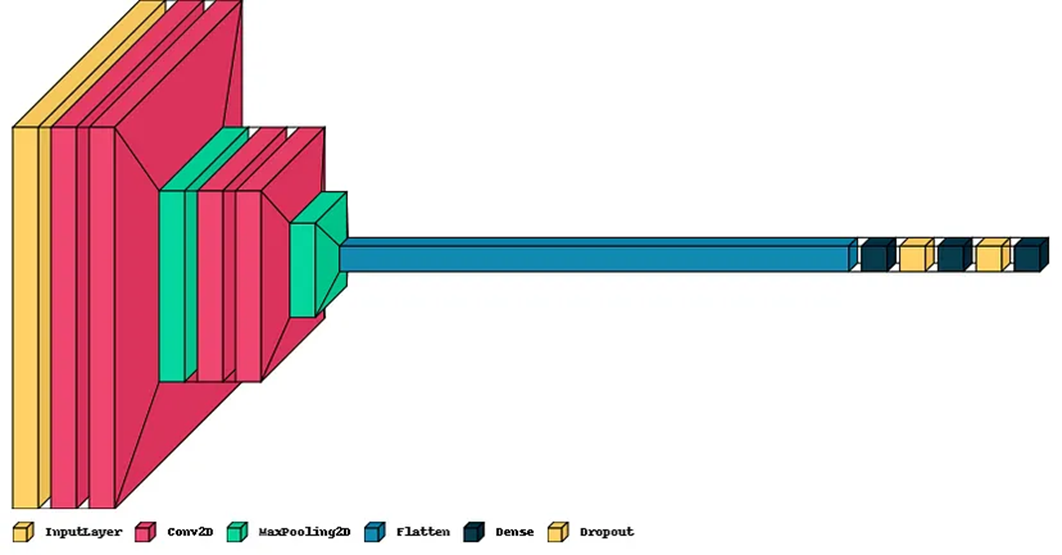
1. Khối 1: hai lớp 32 filters, theo sau là 1 lớp Max-pooling.

2. Khối 2: hai lớp 64 filters, theo sau là 1 lớp Max-pooling.

3. Fully – connected: 2 lớp Dense và 2 lớp dropout, cuối cùng là 1 lớp classify sử dụng hàm kích hoạt “softmax”.

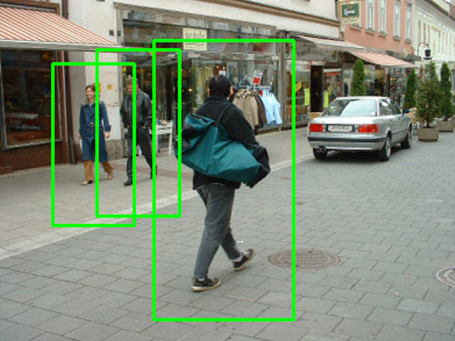


Ta trực quan hóa kiến trúc của Model 2 như sau:



**2.Các ứng dụng của HOG**

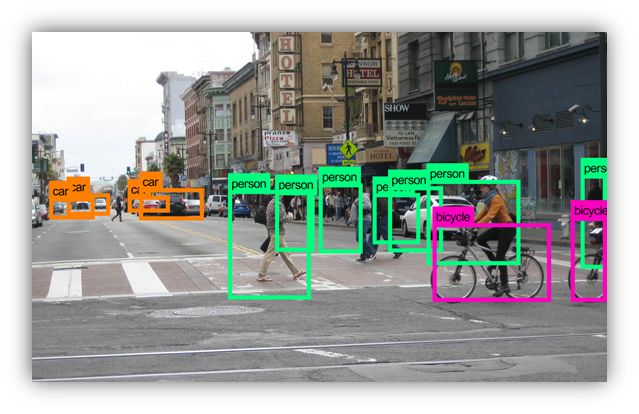
* **Nhận diện người(human detection)**: Lần đầu tiên ứng dụng này được giới thiệu trong bài báo [Histograms of Oriented Gradients for Human Detection](https://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf) của Dalal và Trigg. HOG có thể phát hiện được một hoặc nhiều người đi bộ trên cùng một hình ảnh.



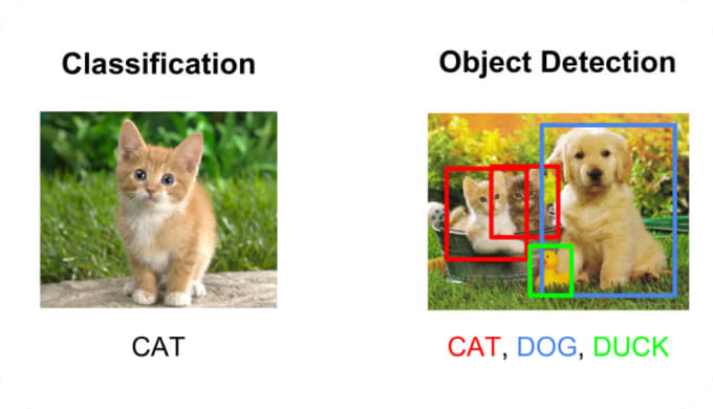
* **Nhận diện khuôn mặt(face detection)**: HOG cũng là một thuật toán rất hiệu quả được áp dụng trong bài toán này. Bởi nó có khả năng biểu diễn các đường nét chính của khuôn mặt dựa trên phương và độ lớn gradient thông qua các véc tơ trên mỗi cell như hình mô tả bên dưới.



* **Nhận diện các vật thể khác**: Ngoài ra còn rất nhiều các trường hợp nhận diện vật thể trên ảnh tĩnh như phương tiện, tín hiệu giao thông, động vật hoặc thậm chỉ là ảnh động từ video.



* **Tạo feature cho các bài toán phân loại ảnh**: Nhiều bài toán phân loại ảnh được xây dựng trên một bộ dữ liệu kích thước nhỏ thì sử dụng các mạng học sâu chưa chắc đã mang lại hiệu quả và dễ dẫn tới overfiting. Nguyên nhân vì dữ liệu ít thường không đủ để huấn luyện cho máy tính nhận tốt các đặc trưng của vật thể. Khi đó sử dụng HOG để tạo đặc trưng sẽ mang lại kết quả tốt hơn.



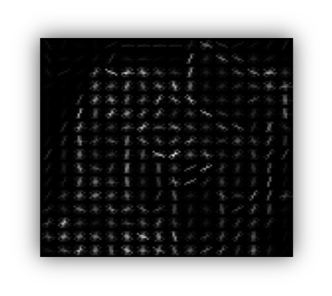
**3.Một số ví dụ về đặc trưng HOG**

**VD1:**

Sau đây nhóm em lấy ngẫu nhiên 2 hình ảnh từ tập data của mình:



Ở hình đầu tiên này ta có thể thấy các đường nét và hình ảnh của mẹt bún đậu được miêu tả rất chi tiết và rõ qua từng các gradient ngang và dọc.



Ở hình thứ 2 này hình ảnh miêu tả đặc trưng HOG của chả giò thậm chí còn rất rõ ràng và chi tiết.

=> chính vì vậy, bọn em sử dụng đặc trưng HOG để xử lý ảnh vì nó có khả năng xử lý ảnh vô cùng tốt và chi tiết.

III.Giới thiệu về Model Support Vector Machine

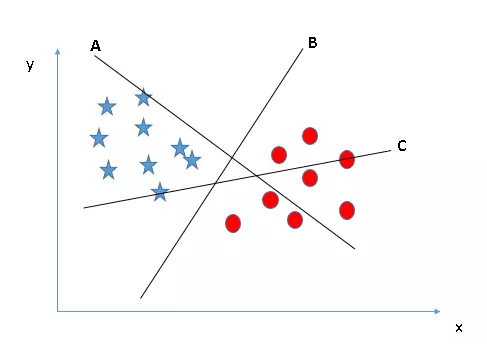
**1.Khái niệm**

SVM là một thuật toán giám sát, nó có thể sử dụng cho cả việc phân loại hoặc đệ quy. Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại. Trong thuật toán này, chúng ta vẽ đồi thị dữ liệu là các điểm trong n chiều ( ở đây n là số lượng các tính năng bạn có) với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó chúng ta thực hiện tìm "đường bay" (*hyper-plane*) phân chia các lớp. Hyper-plane nó chỉ hiểu đơn giản là 1 đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt.



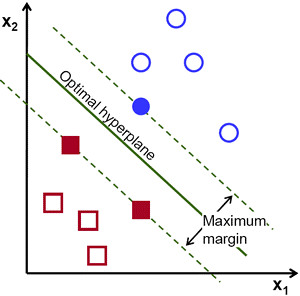
**2.Cách hoạt động của SVM**

Quy tắc số một để chọn 1 hyper-lane, chọn một hyper-plane để phân chia hai lớp tốt nhất. Trong ví dụ này chính là đường B.



Quy tắc thứ hai chính là xác định khoảng cách lớn nhất từ điểu gần nhất của một lớp nào đó đến đường hyper-plane. Khoảng cách này được gọi là "Margin", Hãy nhìn hình bên dưới, trong đấy có thể nhìn thấy khoảng cách margin lớn nhất đấy là đường C. Cần nhớ nếu chọn lầm hyper-lane có margin thấp hơn thì sau này khi dữ liệu tăng lên thì sẽ sinh ra nguy cơ cao về việc xác định nhầm lớp cho dữ liệu.





Margin là khoảng cách giữa một hyperbol trong không gian 2D đến 2 điểm dữ liệu gần nhất tương ứng với các phân lớp.

phương pháp SVM luôn cố gắng cực đại hóa margin này, từ đó thu được một siêu phẳng tạo khoảng cách xa nhất so với 2 lớp. Nhờ vậy, SVM có thể giảm thiểu việc phân lớp sai (misclassification) đối với điểm dữ liệu mới đưa vào.

**IV. Model CNN**

**1.Khái niệm**

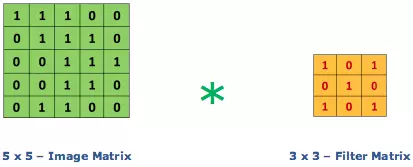
Trong mạng neural, mô hình mạng neural tích chập (CNN) là 1 trong những mô hình để nhận dạng và phân loại hình ảnh. Trong đó, xác định đối tượng và nhận dạng khuôn mặt là 1 trong số những lĩnh vực mà CNN được sử dụng rộng rãi.

Về kỹ thuật, mô hình CNN để training và kiểm tra, mỗi hình ảnh đầu vào sẽ chuyển nó qua 1 loạt các lớp tích chập với các bộ lọc (Kernals), tổng hợp lại các lớp được kết nối đầy đủ (Full Connected) và áp dụng hàm Softmax để phân loại đối tượng có giá trị xác suất giữa 0 và 1. Hình dưới đây là toàn bộ luồng CNN để xử lý hình ảnh đầu vào và phân loại các đối tượng dựa trên giá trị.

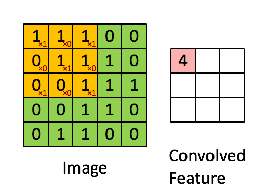
**2.Các steps để áp dụng model CNN**

**Step1: Lớp tích chập - Convolution Layer**

* Tích chập là lớp đầu tiên để trích xuất các tính năng từ hình ảnh đầu vào. Tích chập duy trì mối quan hệ giữa các pixel bằng cách tìm hiểu các tính năng hình ảnh bằng cách sử dụng các ô vương nhỏ của dữ liệu đầu vào. Nó là 1 phép toán có 2 đầu vào như ma trận hình ảnh và 1 bộ lọc hoặc hạt nhân.
* Xem xét 1 ma trận 5 x 5 có giá trị pixel là 0 và 1. Ma trận bộ lọc 3 x 3 như hình bên dưới.

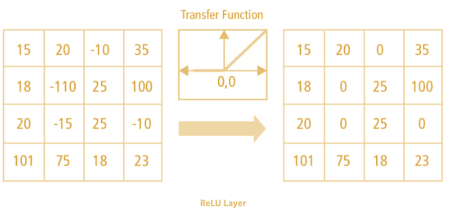
****

* **Sau đó, lớp tích chập của ma trận hình ảnh 5 x 5 nhân với ma trận bộ lọc 3 x 3 gọi là 'Feature Map' như hình bên dưới.**

****

**Step2: Hàm phi tuyến - ReLU**

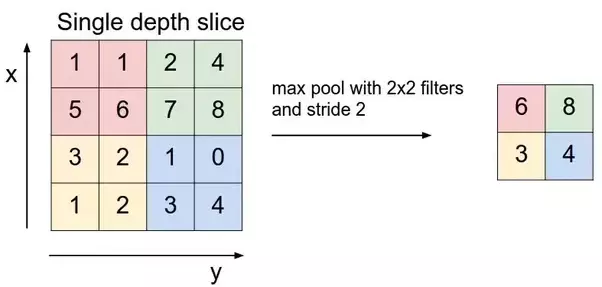
* ReLU viết tắt của Rectified Linear Unit, là 1 hàm phi tuyến. Với đầu ra là: ƒ (x) = max (0, x).
* Tại sao ReLU lại quan trọng: ReLU giới thiệu tính phi tuyến trong ConvNet. Vì dữ liệu trong thế giới mà chúng ta tìm hiểu là các giá trị tuyến tính không âm.



* Có 1 số hàm phi tuyến khác như tanh, sigmoid cũng có thể được sử dụng thay cho ReLU. Hầu hết người ta thường dùng ReLU vì nó có hiệu suất tốt.

**Step3: Pooling layer**

* Lớp pooling sẽ giảm bớt số lượng tham số khi hình ảnh quá lớn. Không gian pooling còn được gọi là lấy mẫu con hoặc lấy mẫu xuống làm giảm kích thước của mỗi map nhưng vẫn giữ lại thông tin quan trọng. Các pooling có thể có nhiều loại khác nhau:
* Max Pooling
* Average Pooling
* Sum Pooling
* Max pooling lấy phần tử lớn nhất từ ma trận đối tượng, hoặc lấy tổng trung bình. Tổng tất cả các phần tử trong map gọi là sum pooling

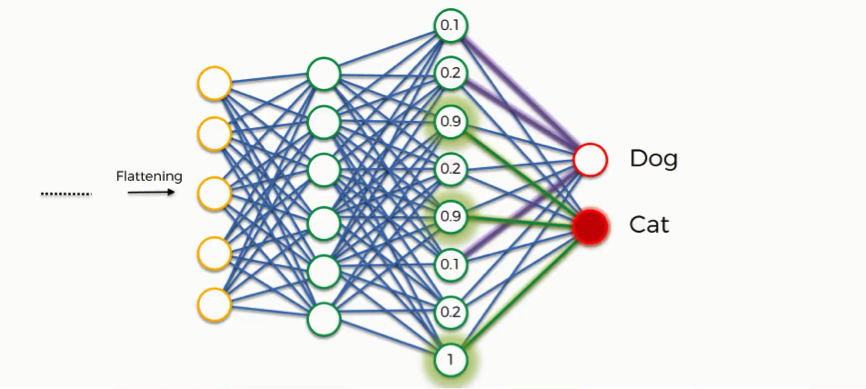


**Step4: Fully Connection**

Fully Connected Layer thường sử dụng để đưa ra các kết quả.

**Ví dụ:** Sau khi các lớp Convolutional Layer và Pooling Layer nhận được các ảnh đã truyền qua chúng, bạn sẽ thu được kết quả là Model đã đọc được khá nhiều thông tin về ảnh. Do đó, để có thể liên kết các đặc điểm này lại và cho ra Output, bạn cần dùng đến Fully Connected Layer.

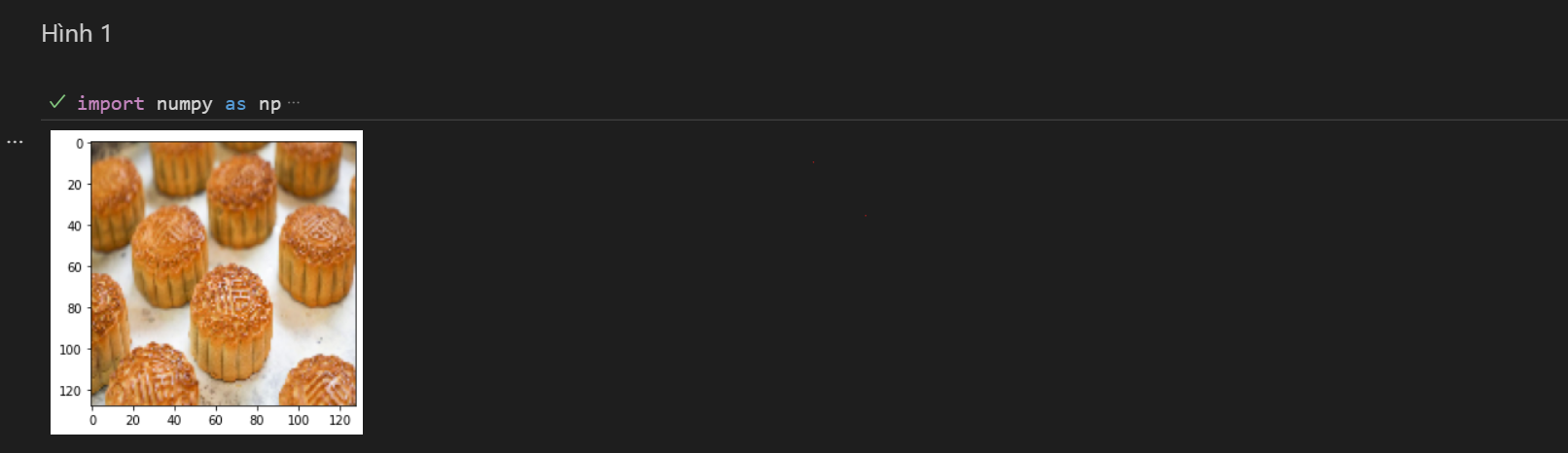
Bên cạnh đó, khi có được các dữ liệu về hình ảnh, Fully Connected Layer sẽ chuyển đổi chúng thành những mục có phân chia chất lượng. Tương tự như kiểu chia chúng thành các phiếu bầu và đánh giá để chọn ra hình ảnh đạt chất lượng tốt nhất. Dù vậy, quá trình này không được coi là quá trình dân chủ nên rất ít sử dụng.

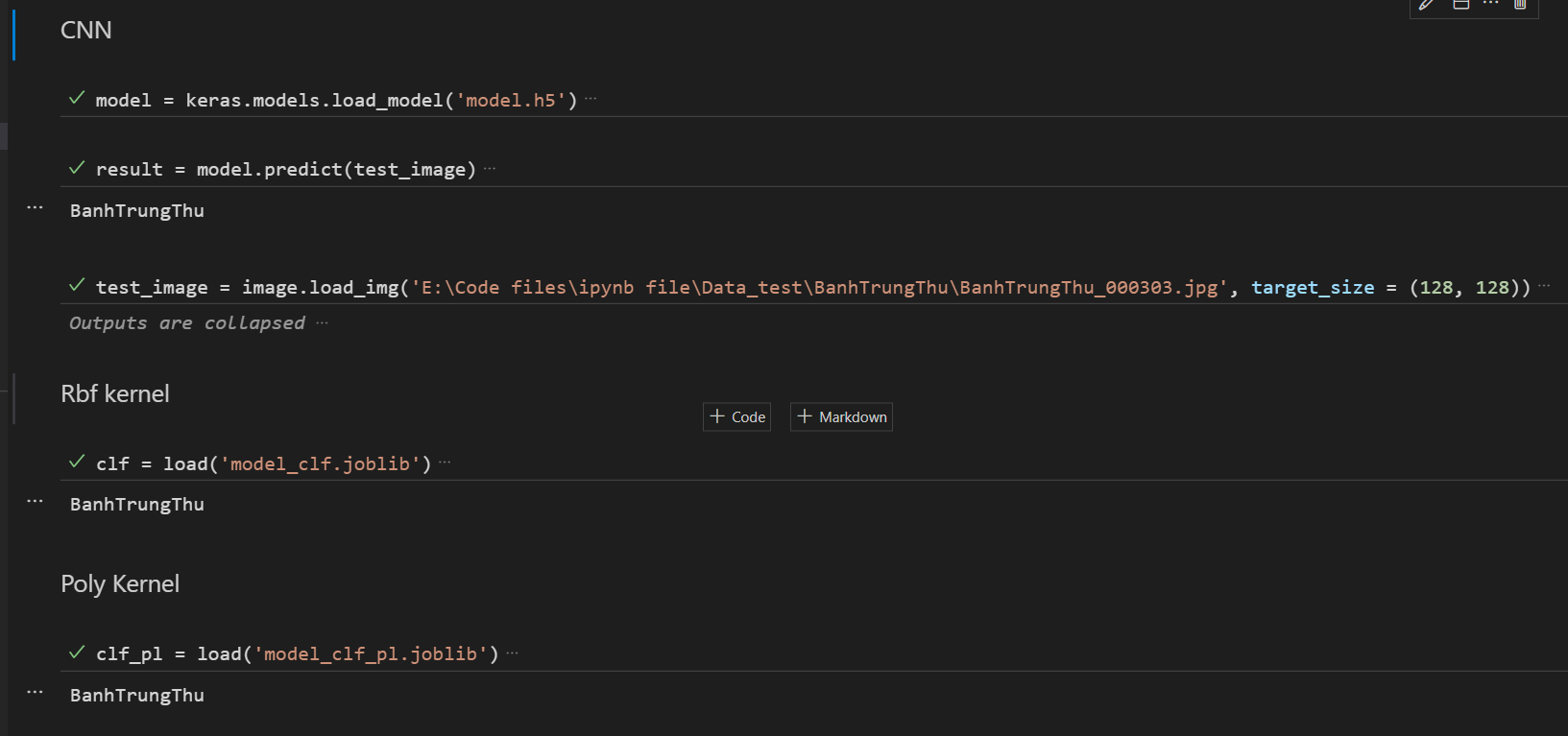


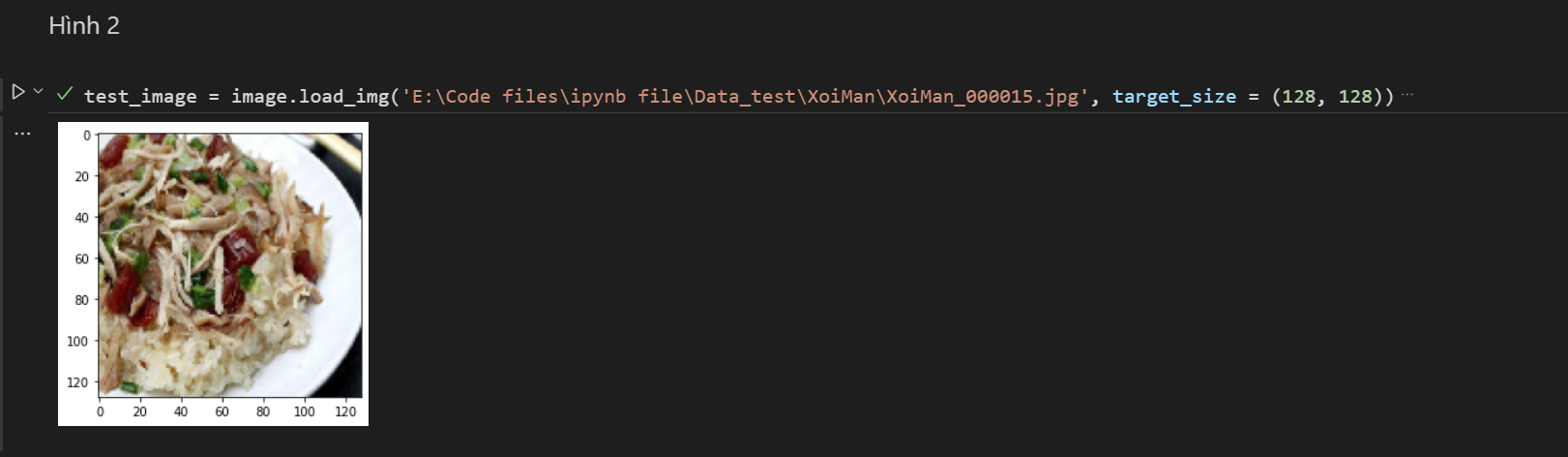
**V.Kết Quả**

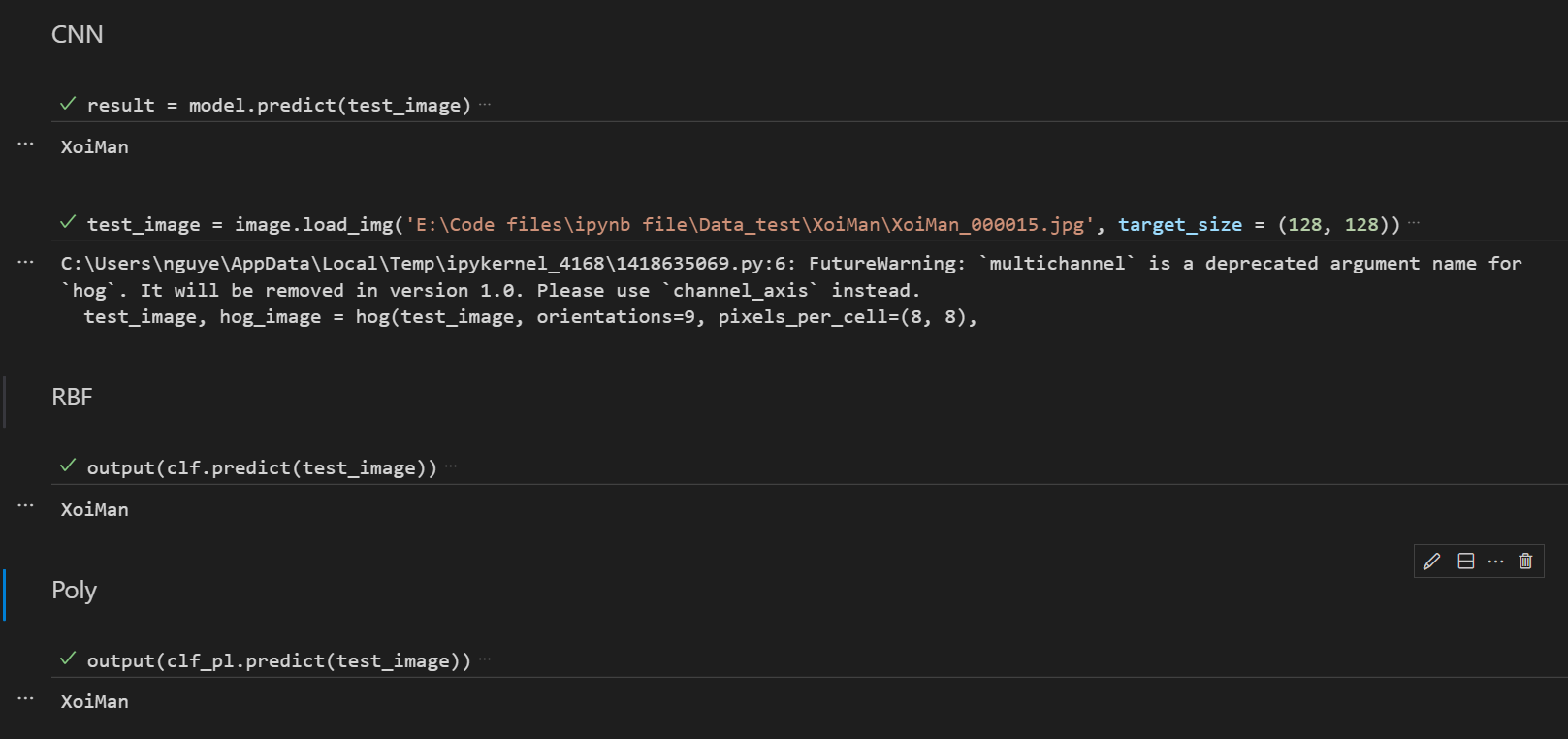
Nhóm chúng em đã sử dụng 6 trường hợp khác nhau để đưa ra kết quả chính xác nhất.

**Trường hợp 1 và 2**: Cả 2 mô hình đều train đúng



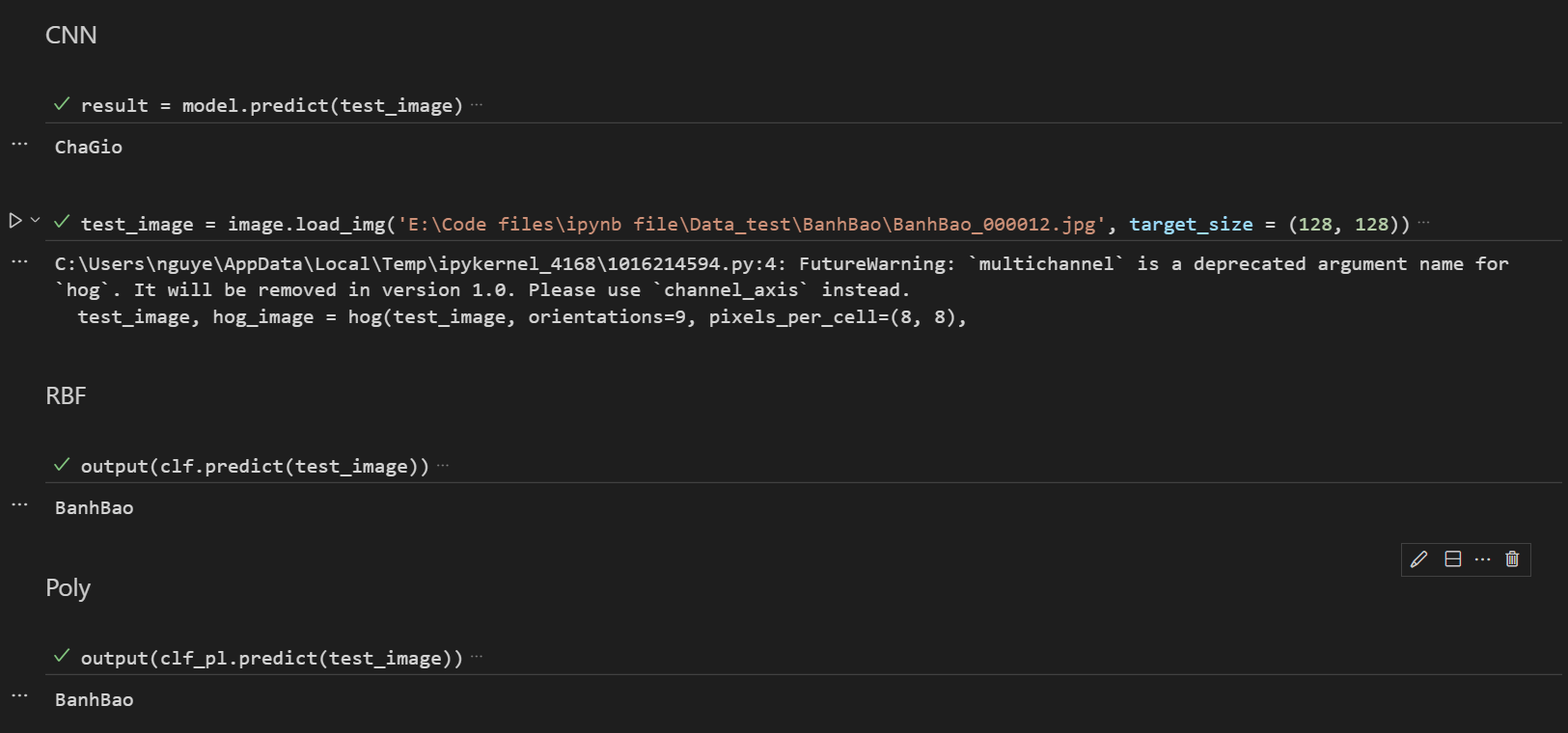




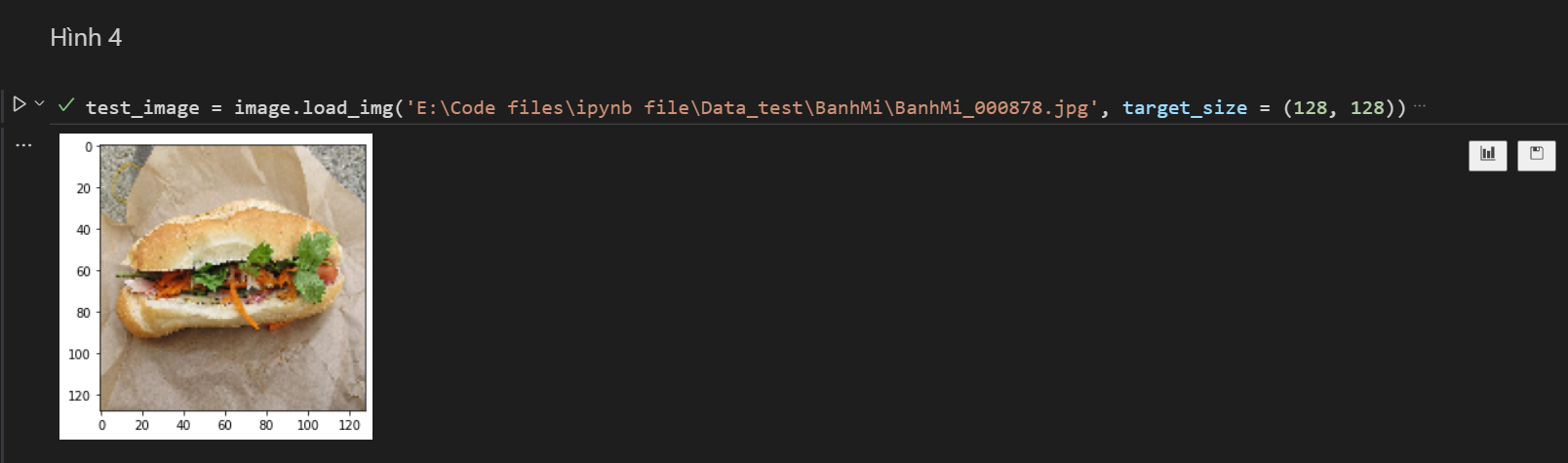


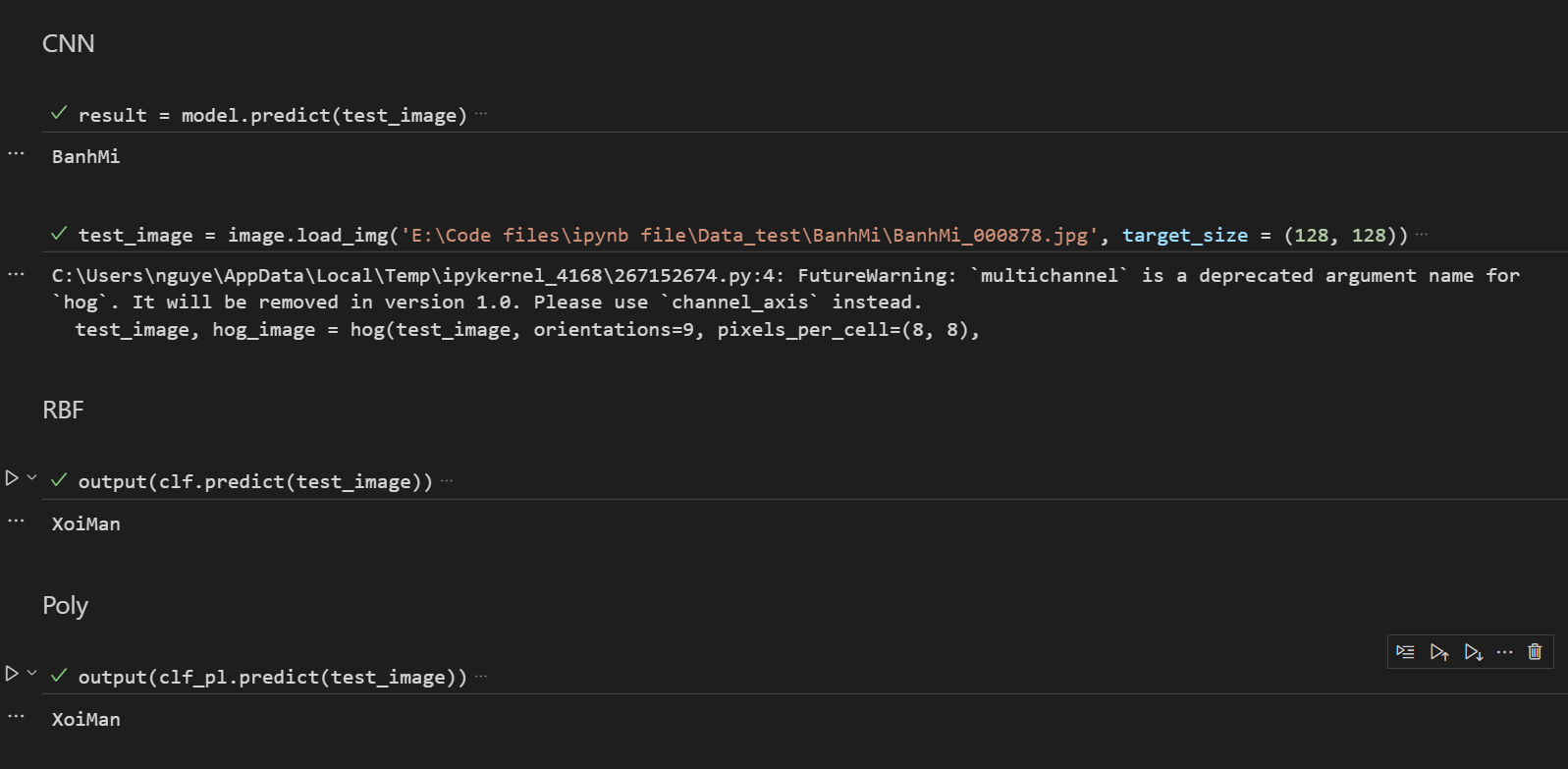
**Trường hợp 3**: Model CNN nhân diện sai, SVM nhận diện đúng



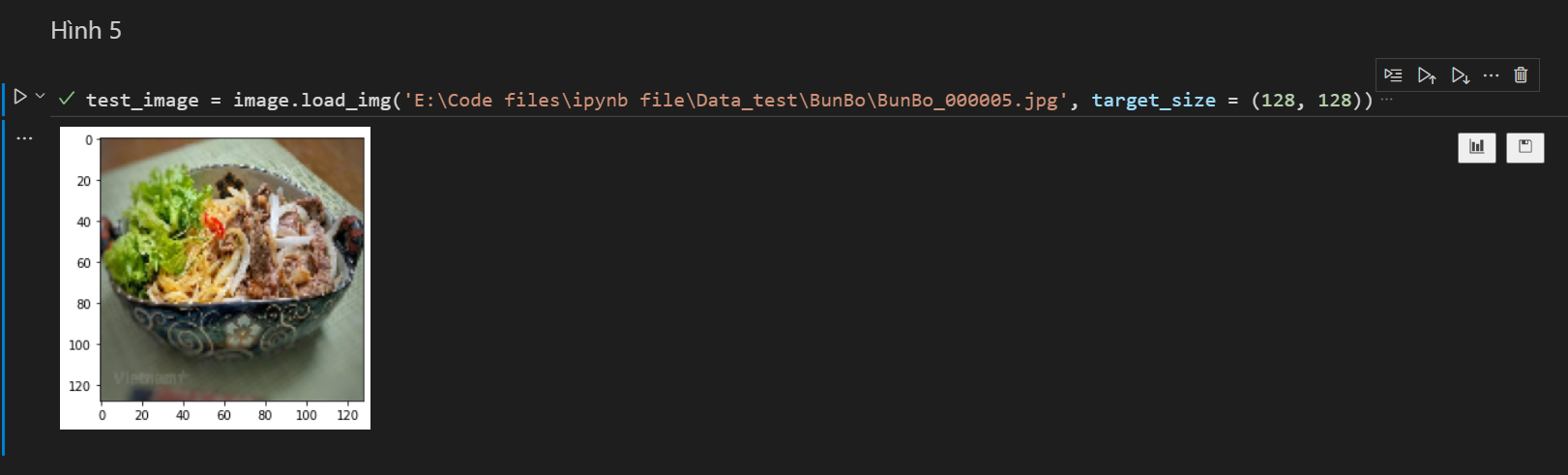


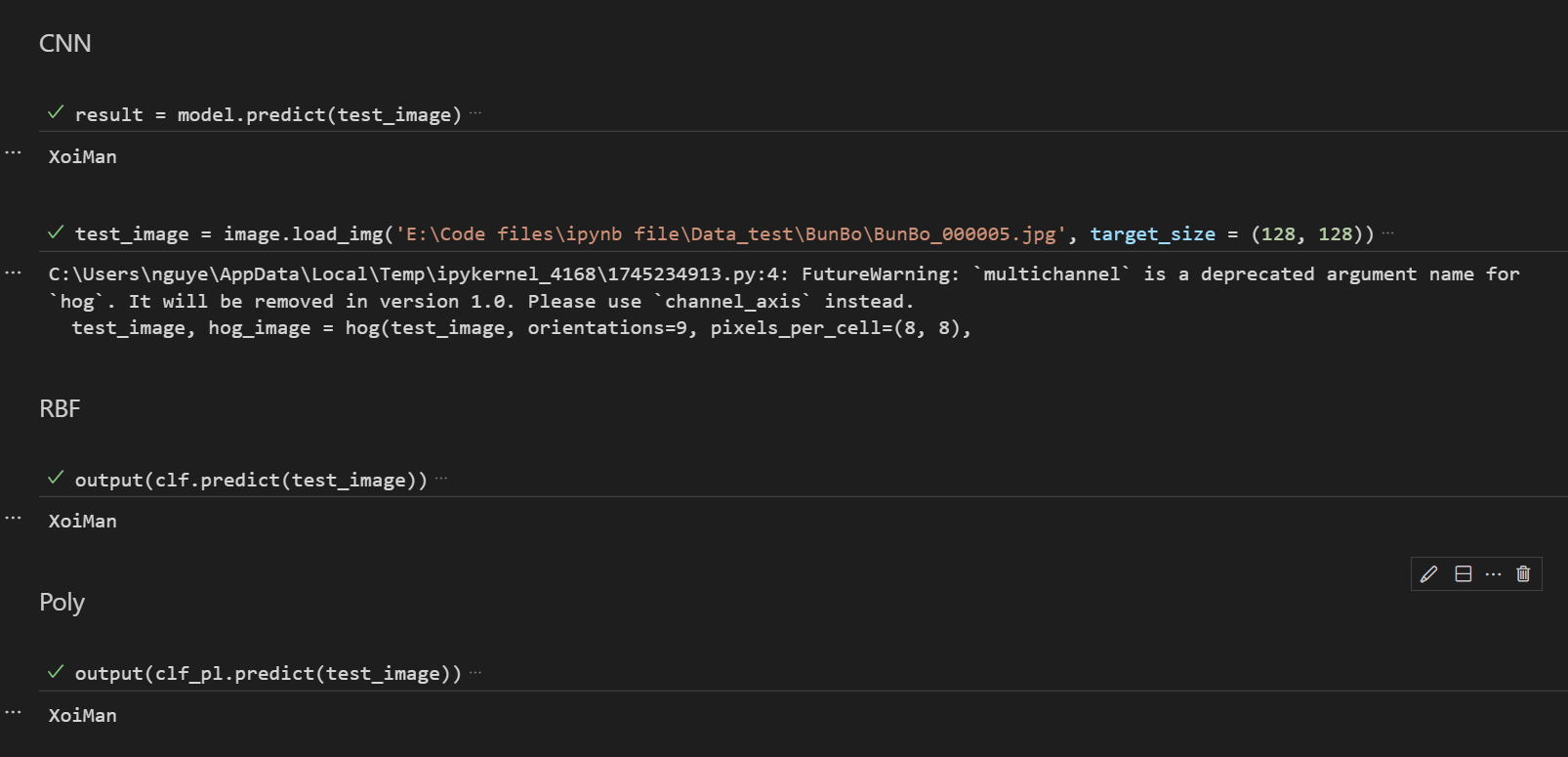
**Trường hợp 4**:Model CNN nhận diện đúng, model SVM nhận diện sai.





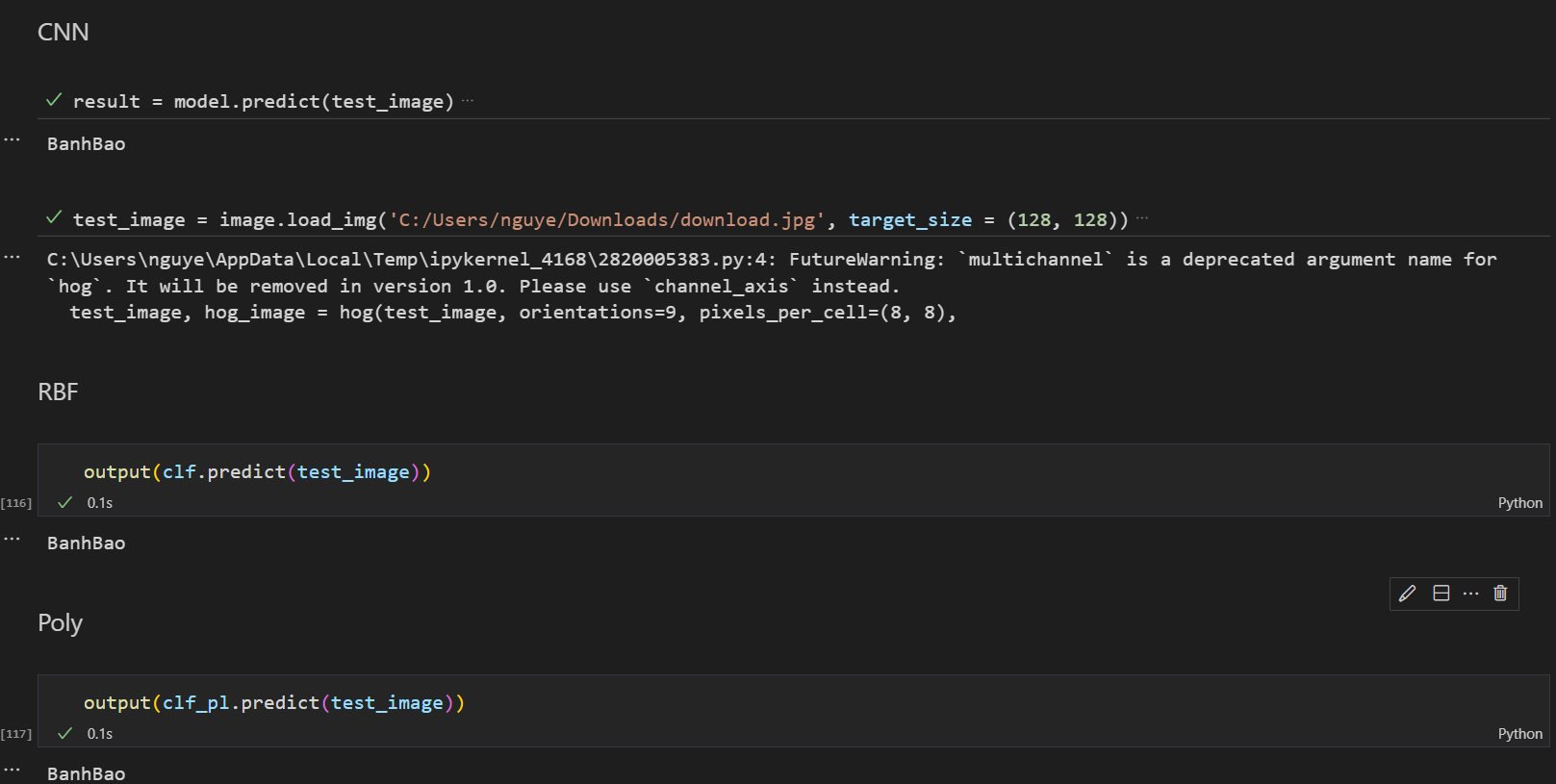
**Trường hợp 5:** Cả 2 model đều nhận dạng sai





**Trường hợp 6:** Nhận dạng 1 ảnh từ bên ngoài không có trong tập data





**VI.So sánh**

Sau khi thực nghiệm, chúng em có 1 bảng so sánh về 2 phương pháp như sau:

|  | HOG + SVM | CNN |
| --- | --- | --- |
| Độ chính xác | SVM(kernel = poly):0,45  SVM(kernel = rbf): 0,42 | 0,83 |

=> Trong bài toán nhận dạng này thì model CNN tỏ ra hiệu quả hơn khá nhiều so với model SVM, tuy vậy cũng có những trường hợp model SVM lại đúng.

**Tài liệu tham khảo:**

<https://wiki.tino.org/convolutional-neural-network-la-gi/>

<https://www.superdatascience.com/blogs/deep-learning-a-z-convolutional-neural-networks-cnn-step-3-flattening/>

<https://viblo.asia/p/gioi-thieu-ve-support-vector-machine-svm-6J3ZgPVElmB>

<https://viblo.asia/p/deep-learning-tim-hieu-ve-mang-tich-chap-cnn-maGK73bOKj2>

<https://phamdinhkhanh.github.io/2019/11/22/HOG.html#1-gi%E1%BB%9Bi-thi%E1%BB%87u-v%E1%BB%81-thu%E1%BA%ADt-to%C3%A1n-hog>

<https://viblo.asia/p/tim-hieu-ve-phuong-phap-mo-ta-dac-trung-hog-histogram-of-oriented-gradients-V3m5WAwxZO7>