# Trích xuất đặc trưng là gì?

Trích xuất đặc trưng là một phần của quy trình giảm kích thước, trong đó, tập hợp dữ liệu thô ban đầu được chia và giảm thành các nhóm dễ quản lý hơn. Vì vậy, khi bạn muốn xử lý sẽ dễ dàng hơn. Đặc điểm quan trọng nhất của các tập dữ liệu lớn này là chúng có một số lượng lớn các biến. Các biến này yêu cầu nhiều tài nguyên máy tính để xử lý chúng. Vì vậy, việc trích xuất đặc trưng giúp có được tính năng tốt nhất từ ​​các tập dữ liệu lớn đó bằng cách chọn và kết hợp các biến thành các tính năng, do đó, giảm lượng dữ liệu một cách hiệu quả. Các tính năng này rất dễ xử lý, nhưng vẫn có thể mô tả tập dữ liệu thực tế với độ chính xác và độc đáo.

# Tại sao trích xuất đặc trưng hữu ích?

Kỹ thuật trích xuất các tính năng rất hữu ích khi bạn có một tập dữ liệu lớn và cần giảm số lượng tài nguyên mà không làm mất bất kỳ thông tin quan trọng hoặc liên quan nào. Tính năng trích xuất giúp giảm lượng dữ liệu dư thừa từ tập dữ liệu.

Cuối cùng, việc giảm dữ liệu sẽ giúp xây dựng mô hình với ít nỗ lực của máy hơn và cũng tăng tốc độ học và các bước tổng quát hóa trong quá trình máy học.

# Làm thế nào để máy tính lưu trữ hình ảnh?

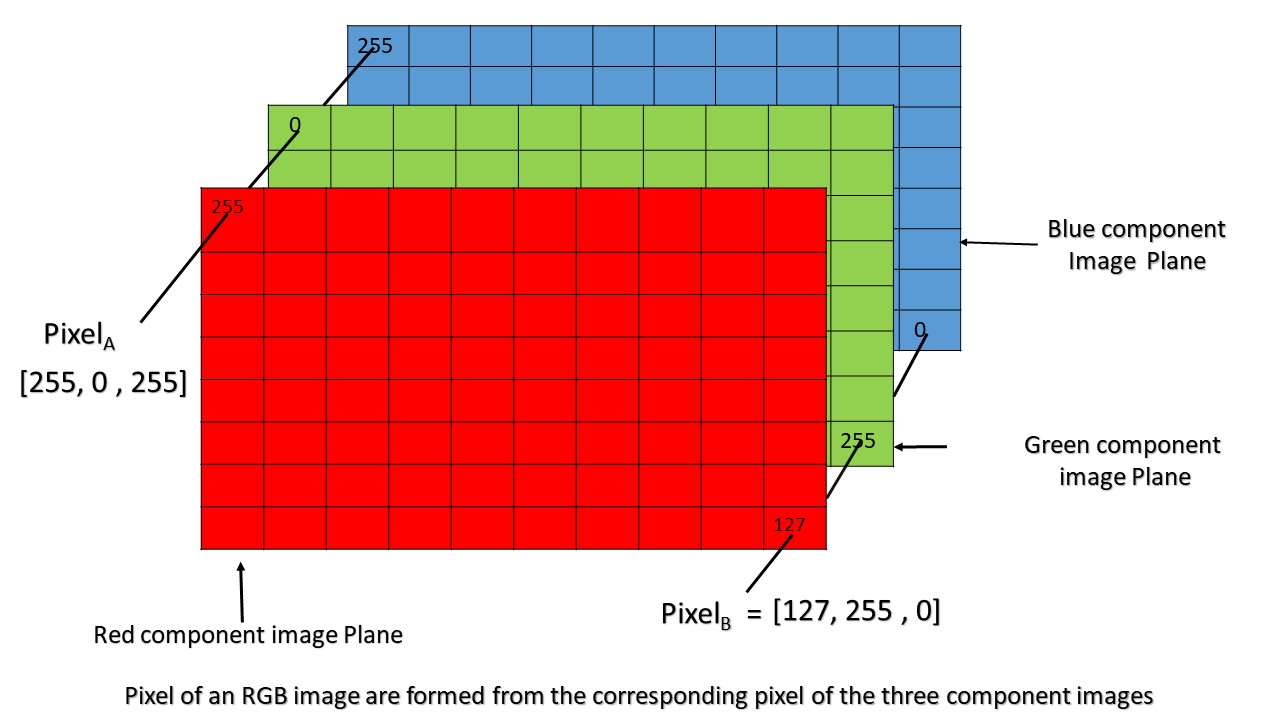
Điều đầu tiên, chúng ta cần hiểu cách một máy có thể đọc và lưu trữ hình ảnh. Việc tải ảnh, đọc chúng rồi xử lý qua máy rất khó vì máy không có mắt như chúng ta.

Máy móc nhìn thấy bất kỳ hình ảnh nào dưới dạng ma trận số. Kích thước của ma trận này thực sự phụ thuộc vào số lượng pixel của hình ảnh đầu vào.

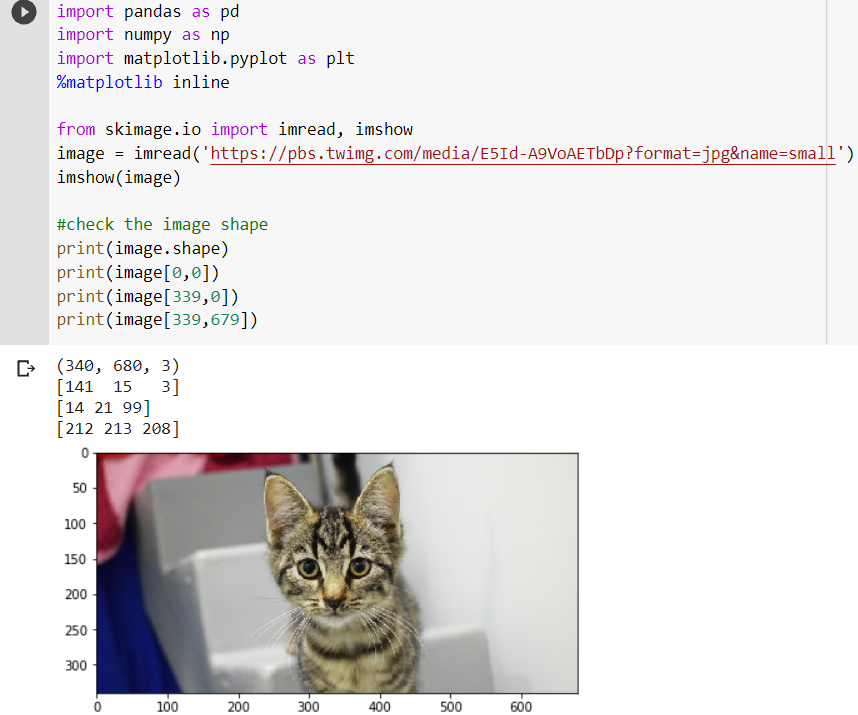
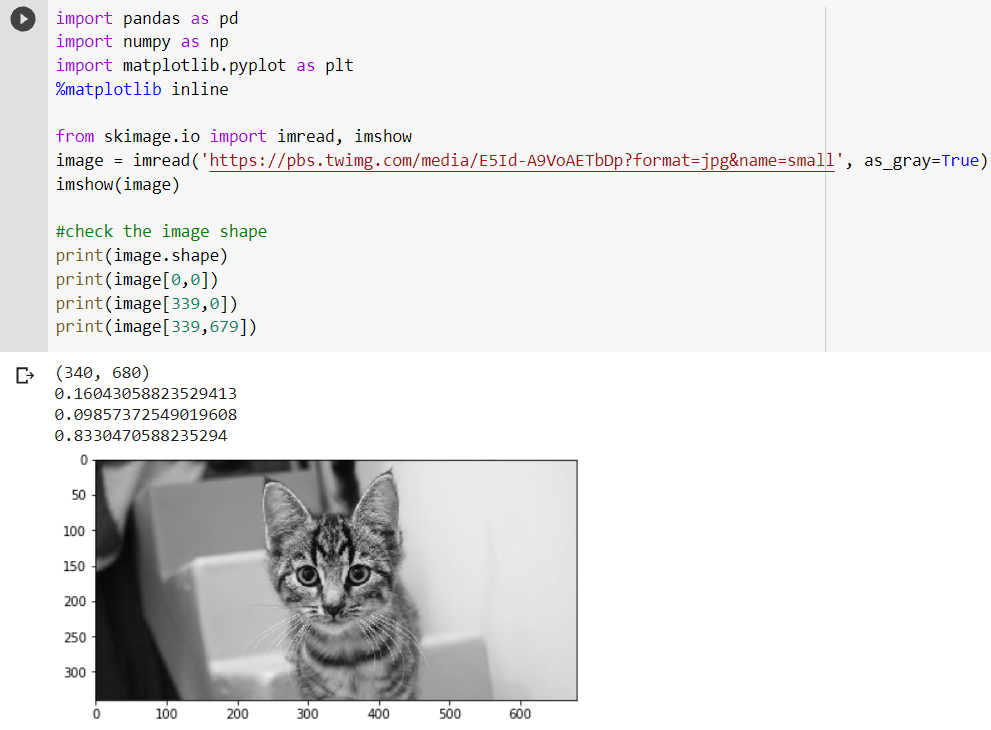
**Pixel là gì?**

Giá trị Pixel cho mỗi pixel là viết tắt hoặc mô tả độ sáng và màu sắc của pixel đó. Vì vậy, trong trường hợp đơn giản nhất của ảnh nhị phân, giá trị pixel là số 1 bit cho biết nền trước hoặc nền sau (ảnh trắng đen). Vậy pixel là số hoặc giá trị biểu thị cường độ hoặc độ sáng của pixel. Các số nhỏ gần 0 biểu thị màu đen và các số lớn gần 255 biểu thị màu trắng. Vậy, đây là khái niệm về pixel và cách máy nhìn thấy hình ảnh thông qua các con số mà không cần mắt.

Tuy nhiên, đối với trường hợp hình ảnh có màu, chúng ta có 3 ma trận hoặc 3 kênh: màu đỏ (R), màu xanh lá (G), xanh lam (B). Trong 3 ma trận này, mỗi ma trận có các giá trị từ 0-255 đại diện cho cường độ màu của pixel đó.



Có thể thấy chúng ta cũng có 3 ma trận đại diện cho kênh RGB. Ba kênh này được xếp chồng lên nhau và được sử dụng để tạo thành một hình ảnh có màu. Đây là cách máy tính có thể phân biệt giữa các hình ảnh.

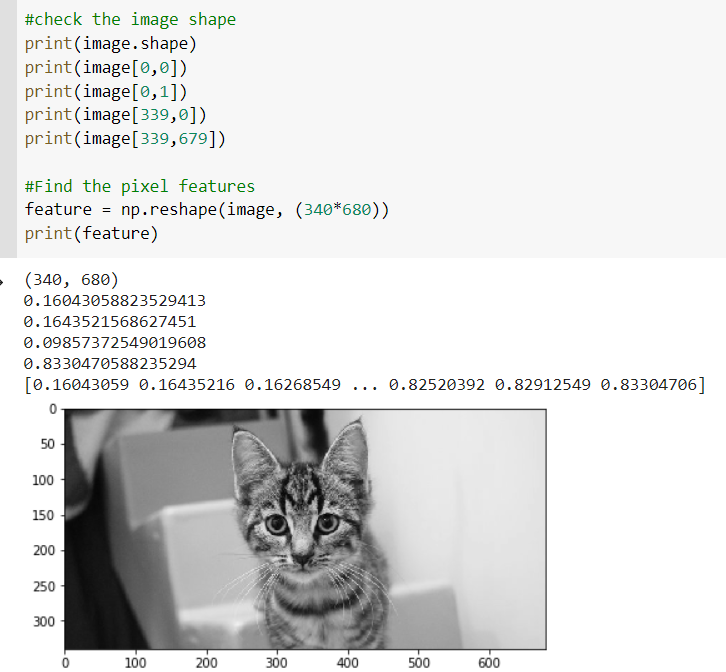


# Cách sử dụng kỹ thuật trích xuất đặc trưng cho dữ liệu hình ảnh:

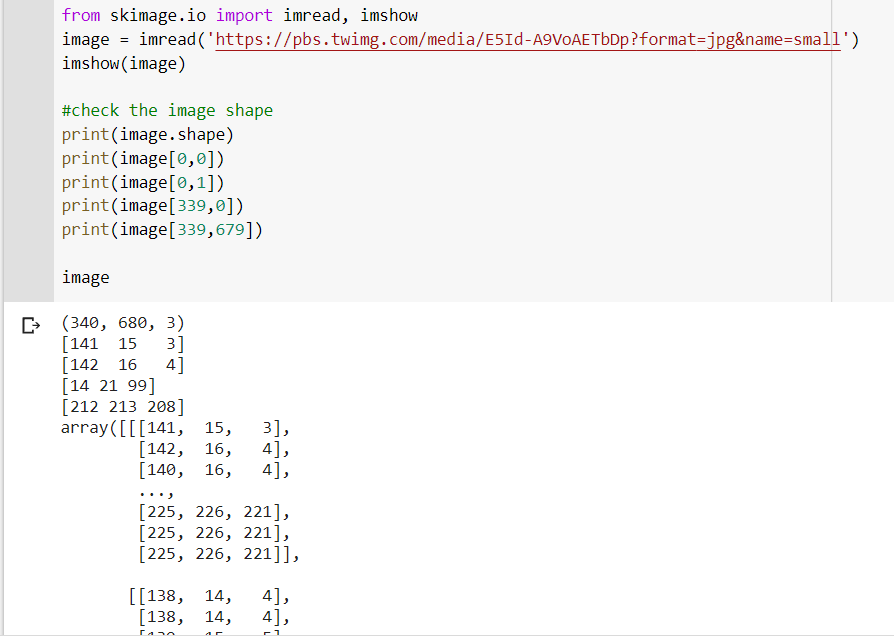
## Các đối tượng dưới dạng giá trị pixel thang độ xám

Nếu chúng ta sử dụng ví dụ tương tự như hình ảnh mà chúng ta sử dụng ở trên tthì kích thước của hình ảnh là 340 x 680. Số lượng các đặc trưng giống như số lượng pixel nên số lượng đặc trưng sẽ là 340 x 680 = 231200.

Vì vậy, làm cách nào để khai báo 231200 pixel là các đặc trưng của hình ảnh này? Giải pháp là, chúng ta chỉ cần nối từng giá trị pixel lần lượt để tạo ra một vectơ đặc trưng cho hình ảnh.



## Giá trị pixel trung bình trong kênh



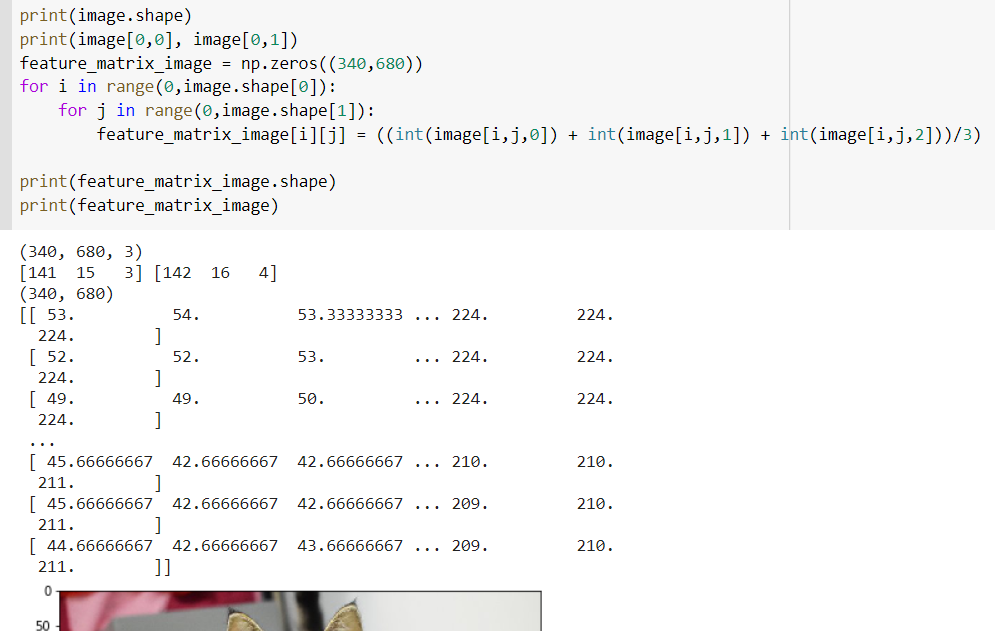
Đối với trường hợp này, hình ảnh có một kích thước (340, 680, 3). Ba giá trị này đại diện cho giá trị RGB cũng như số lượng kênh. Bây giờ chúng ta sẽ sử dụng phương pháp trước để tạo các đặc trưng.

Tổng số đặc trưng của trường hợp này là 340\*680\*3 = 693600

Từ quá khứ, tất cả chúng ta đều nhận thức được điều đó, số lượng các tính năng vẫn giữ nguyên. Trong trường hợp này, các giá trị pixel từ cả ba kênh của hình ảnh sẽ được nhân lên.



Trong hình ảnh màu này có một ma trận 3D có kích thước (340 \* 680 \* 3) trong đó 340 biểu thị chiều cao, 680 là chiều rộng và 3 là số kênh. Để có được giá trị pixel trung bình cho hình ảnh, chúng ta sẽ sử dụng vòng lặp for.



Bây giờ chúng ta đã tạo ra một ma trận mới có cùng chiều cao và chiều rộng nhưng chỉ có 1 kênh. Để chuyển đổi ma trận thành mảng 1 chiều, chúng ta sẽ sử dụng thư viện Numpy như phần trước.

