**BÁO CÁO**

**TỔNG QUAN CÁC KIẾN TRÚC MẠNG ĐƯỢC TÍCH HỢP SẴN TRONG KERAS**

**Các kiến trúc được tích hợp sẵn trong Keras**

* [Xception](https://keras.io/applications/" \l "xception)
* [VGG16](https://keras.io/applications/" \l "vgg16)
* [VGG19](https://keras.io/applications/" \l "vgg19)
* [ResNet50](https://keras.io/applications/" \l "resnet50)
* [InceptionV3](https://keras.io/applications/" \l "inceptionv3)
* [InceptionResNetV2](https://keras.io/applications/" \l "inceptionresnetv2)
* [MobileNet](https://keras.io/applications/" \l "mobilenet)
* [DenseNet](https://keras.io/applications/" \l "densenet)
* [NASNet](https://keras.io/applications/" \l "nasnet)
* [MobileNetV2](https://keras.io/applications/" \l "mobilenetv2)

Thống kê độ lớn và độ chính xác các mạng (đánh giá trên ImageNet dataset):

| Model | Size | Top-1 Accuracy | Top-5 Accuracy | Parameters | Depth |
| --- | --- | --- | --- | --- | --- |
| [Xception](https://keras.io/applications/" \l "xception) | 88 MB | 0.790 | 0.945 | 22,910,480 | 126 |
| [VGG16](https://keras.io/applications/" \l "vgg16) | 528 MB | 0.715 | 0.901 | 138,357,544 | 23 |
| [VGG19](https://keras.io/applications/" \l "vgg19) | 549 MB | 0.727 | 0.910 | 143,667,240 | 26 |
| [ResNet50](https://keras.io/applications/" \l "resnet50) | 99 MB | 0.759 | 0.929 | 25,636,712 | 168 |
| [InceptionV3](https://keras.io/applications/" \l "inceptionv3) | 92 MB | 0.788 | 0.944 | 23,851,784 | 159 |
| [InceptionResNetV2](https://keras.io/applications/" \l "inceptionresnetv2) | 215 MB | 0.804 | 0.953 | 55,873,736 | 572 |
| [MobileNet](https://keras.io/applications/" \l "mobilenet) | 17 MB | 0.665 | 0.871 | 4,253,864 | 88 |
| [DenseNet121](https://keras.io/applications/" \l "densenet) | 33 MB | 0.745 | 0.918 | 8,062,504 | 121 |
| [DenseNet169](https://keras.io/applications/" \l "densenet) | 57 MB | 0.759 | 0.928 | 14,307,880 | 169 |
| [DenseNet201](https://keras.io/applications/" \l "densenet) | 80 MB | 0.770 | 0.933 | 20,242,984 | 201 |

1. **Các kiến trúc được pre-trained trên dữ liệu ImageNet**

VGG16

VGG19

ResNet50

Inception V3

Xception

Dataset ImageNet:

Tập dữ liệu gồm 1000 nhãn, 1.2 triệu hình ảnh cho training, 50k ảnh cho validate và 100k ảnh cho test.

ImageNet challenge là điểm chuẩn thực tế để đánh giá các thuật toán phân loại.

Các mạng được tích hợp trong keras là những mạng nơ-ron hiệu quả cao nhất trong ImageNet challenge vài năm qua.

**1.1. VGG16 and VGG19**



Mạng VGG được đề xuất bởi Simonyan and Zisserman trong bài báo của họ năm 2014.

Cấu trúc mạng đơn giản, chỉ dùng các lớp 3x3 conv xếp chồng lên nhau, max pooling, cuối là 2 lớp fully connect theo sau là một trình phân loại softmax.

Hai nhược điểm của VGG:

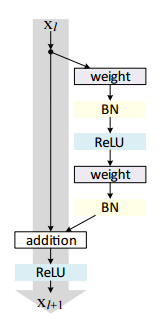
* Train chậm
* Bộ trọng số lớn, do độ sâu và lớp fully connect ( VGG16: 533MB, VGG19: 574MB)

VGG vẫn được sử dụng trong nhiều vấn đề phân loại, tuy nhiên các kiến trúc mạng nhỏ hơn thường hấp dẫn hơn ( như SquezeNet, GoogLeNet,v.v..)

#### 1.2. ResNet

ResNet là một dạng kiến trúc “mạng trong mạng” có nghĩa là mạng tổng thể được xây dựng từ các block mạng con.

Được giới thiệu bởi He et al. trong bài báo của họ năm 2015, Deep Residual Learning cho nhận dạng hình ảnh, kiến trúc ResNet đã trở thành một công trình tiêu biểu, chứng minh rằng các mạng cực kỳ sâu có thể được đào tạo bằng SGD chuẩn (và khởi tạo hợp lý) thông qua việc sử dụng các mô-đun residual:



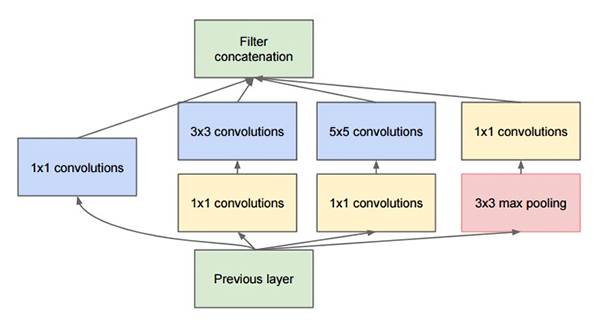
Với việc sử dụng “skip connection” sẽ giảm việc tính toán do đó làm tăng độ sâu của mạng mà theo lý thuyết mạng càng sâu khả năng nhận dạng tốt càng cao.

“skip connection” : Một cách đơn giản, khi trọng số ở nhánh phải (hình trên) của modun residual đã tối ưu thì quá trình cập nhật trọng số tiếp theo nhánh đó sẽ được bỏ qua mà đi thẳng qua nhánh trái.

Mặc dù ResNet (cụ thể là ResNet50 (50 lớp)) sâu hơn VGG16, VGG19 nhưng kích thước mô hình nhỏ hơn do sử dụng global average pooling thay vì fully connected. Kích thước mô hình chỉ 102MB cho ResNet50.

#### 1.3. Inception V3

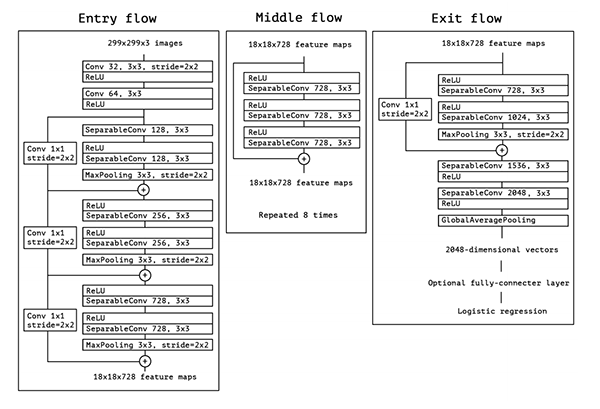
Lần đầu tiên được giới thiệu bởi Szegedy et al. trong bài báo của họ năm 2014



Modun Inception hoạt động như một “bộ tách tính năng đa cấp” bằng các CONV 1x1, 2x2, 3x3, 5x5, đầu ra sẽ được xếp chồng lên nhau rồi đưa vào lớp tiếp theo. khai thác được đặc trưng ở nhiều cấp cho mạng cái nhìn tổng quát hơn về ảnh.

Trọng số cho Inception V3 nhỏ hơn cả VGG và ResNet, đạt 96MB.

#### 1.4. Xception



Được đề xuất bởi François Chollet, người sáng tạo và giám đốc bảo trì Keras.

Xception là một phần mở rộng của Inception, nó thay thế các modun inception chuẩn bằng các CONV phân tách theo chiều sâu (Deepwise separable convolution).

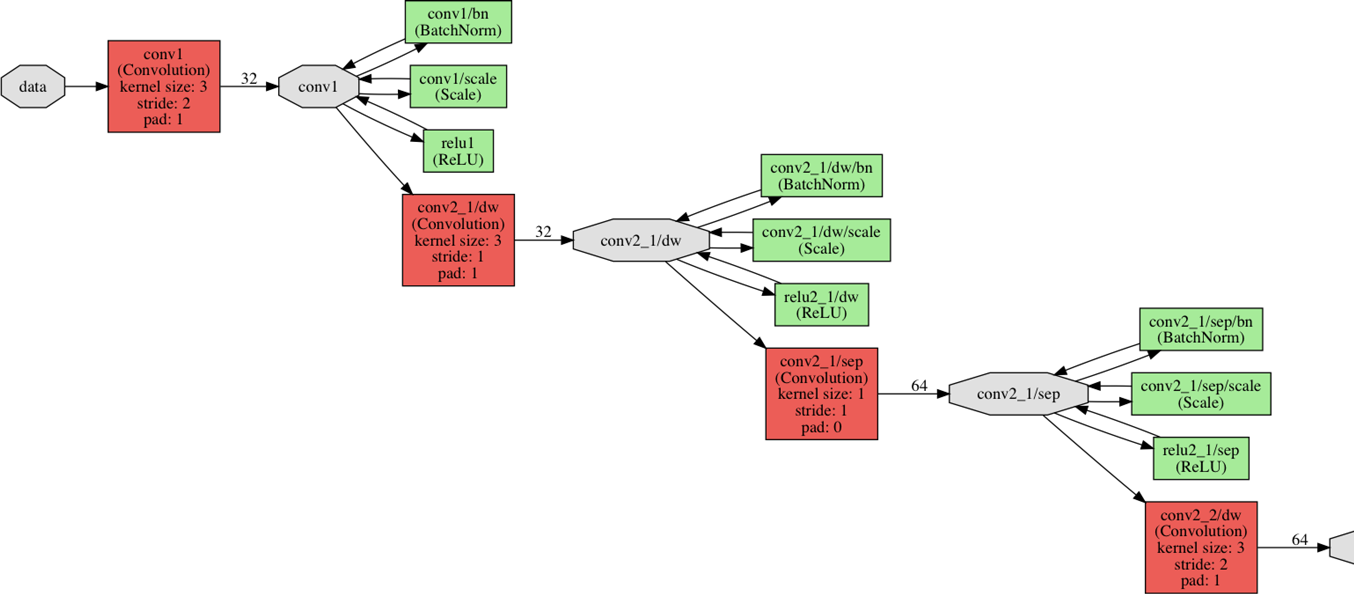
Xception có bộ trọng số chỉ 91MB.

**2. Các kiến trúc khác**

**2.1. MobileNet**

Được công bố bởi nhóm nghiên cứu của Google.

Tác giả khẳng định kiến trúc này hoạt động tốt trên các thiết bị di động và có độ chính xác gần như các mạng lớn hơn nhiều như VGG16.



Ý tưởng của MobileNet là sử dụng depthwise separable convolution để xây dựng một mạng nhẹ hơn thích hợp cho thiết bị di động.

MobileNetV2 có một vài cải tiến về kiến trúc dẫn đến độ chính xác cao hơn MobileNetV1

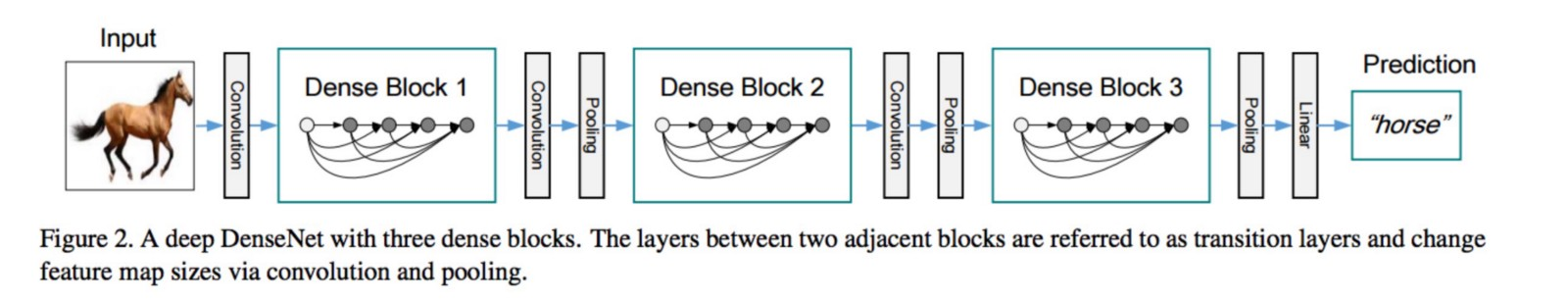
| Version | Top-1 Accuracy | Top-5 Accuracy |
| --- | --- | --- |
| MobileNet V1 | 70.9 | 89.9 |
| MobileNet V2 | 71.8 | 91.0 |

**2.2. DenseNet**

Theo các nghiên cứu gần đây chỉ ra rằng các mạng có thể sâu hơn, chính xác hơn và hiệu quả training cao hơn nếu đưa vào các kết nối ngắn giữa các lớp.

DenseNet là một phần mở rộng của ResNet

DenseNet đề xuất “concate” các đầu ra thay vì “add” như ResNet

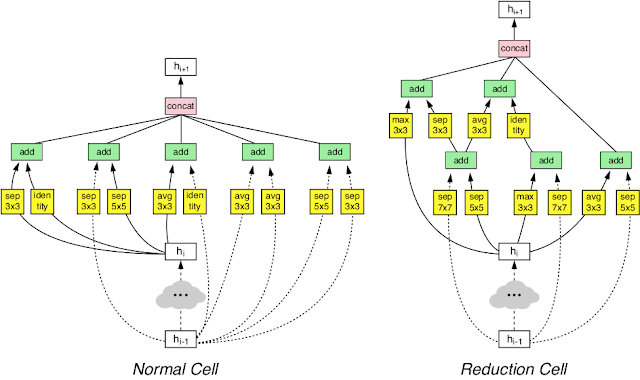
Với mỗi lớp , feature map đầu ra của mỗi lớp là đầu vào của tất cả các lớp tiếp theo.

Ưu điểm của kiến trúc này là: giảm mất mát gradient, sử dụng lại được các feature, giảm đáng kể lượng trọng số. Nhờ đó mô hình nhỏ gọn hơn và tăng hiệu quả trainning.

**2.3. NASNet (The Neural Architecture Search (B. Zoph and Q.V. Le, 2017))**

Mạng NASNet có kiến trúc học được từ bộ dữ liệu CIFAR-10 và được đào tạo với bộ dữ liệu ImageNet 2012. Mô hình này được sử dụng để generator feature map và xếp chồng lên Fast R-CNN. Sau đó toàn bộ mô hình được đào tạo tiếp bằng dataset COCO.

Mạng được sử dụng để nhận dạng đối tượng cho độ chính xác trung bình (mAP) lên tới 43.1%



**Tài liệu tham khảo:**

[https://keras.io/applications/#vgg16](https://keras.io/applications/" \l "vgg16)

<https://www.pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xception-keras/>

<http://machinethink.net/blog/googles-mobile-net-architecture-on-iphone/>

<https://vitalab.github.io/deep-learning/2017/03/21/xception.html>

<https://medium.com/comet-app/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852>

<https://towardsdatascience.com/densenet-2810936aeebb>

<https://ai.googleblog.com/2017/11/automl-for-large-scale-image.html>