|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | LENET(1998) | ALEXNET(2012) | VGG16(2014) | INCEPTION | RESNET(18) | DENCE | MOBILE | ENFICIENT |
| Ứng dụng | Nhận dạng ký tự viết tay | Nhận dạng hình ảnh phức tạp | Nhận dạng và phân loại hình ảnh và tác vụ thị giác khác trên dữ liệu quy mô lớn | Nhận dạng và phân loại hình ảnh và tác vụ thị giác khác trên dữ liệu quy mô lớn | // | // | // | // |
| Kích thước input | 32\*32\*1 | 224\*224\*3 | 224\*224\*3 | 224\*224\*3 | 224\*224\*3 | // | // | // |
| Số lớp tích chập | 2 | 5 | 13 | Inception v3 có 48 lớp, bao gồm các lớp tích chập, pooling, và các Inception blocks. | 17 |  |  |  |
| Số lớp FC | 2 | 3 | 3 |  | 1 |  |  |  |
| Kích thước mô hình | Nhỏ, đơn giản | Trung bình, Phức tạp | -Lớn, Rất sâu | Kiến trúc phức tạp hơn với khối Inception blocks, mỗi khối bao gồm nhiều phép tích chập và pooling khác nhau diễn ra song song. | -Lớn, Rất sâu | -Lớn, Rất sâu |  |  |
| Kỹ thuật chính | Sigmoid  AvgPool  GPU | ReLU  MaxPool  Dropout  GPU | Bộ lọc 3\*3, sâu hơn (lên tới 16 lớp 512 filter) | Sử dụng các bộ lọc với nhiều kích thước khác nhau trong cùng một lớp, điều này cho phép mô hình học được nhiều đặc điểm khác nhau của hình ảnh. | * Sử dụng kỹ thuật phần dư, Giúp duy trì thông tin trong mạng nơ-ron sâu và giảm vấn đề gradient vanishing. | * Mỗi khối dense bao gồm một chuỗi các lớp tích chập, và mỗi lớp trong khối kết nối trực tiếp với tất cả các lớp trước đó. Điều này tạo ra một mạng lưới kết nối dày đặc giữa các lớp. | * áp dụng một kỹ thuật gọi là tích chập tách biệt chiều sâu (Depthwise Separable Convolution) để giảm tải tính toán. * B1: Depthwise Convolution * B2: Tích chập điểm (Pointwise Convolution) | * Sử đụng kỹ thuật Compound Scaling và các kỹ thuật tối ưu MBConV, SE block, Swish Activation |
| Ưu điểm so với thế hệ trước |  | * Thay ReLu cho Sigmoid giảm phép tính toán do không có phép lũy thừa, đễ dàng sử dụng các phương pháp khởi tạo tham số khác nhau * Thay AvgP bằng MaxP giúp làm nổi bật hơn đặc trưng của ảnh * Dropout ngăn ngừa tình trạng OVF | * Sử dụng toàn bộ là bộ lọc 3\*3 và gộp 2\*2 thống nhất về mặt thiết kế * Cửa sổ tích chập hẹp và nhiều tầng cho kết quả cao hơn cửa sổ chập lớn nhưng ít tầng. | * mô hình học được nhiều đặc điểm khác nhau của hình ảnh. * Kích thước mô hình nhỏ hơn (~90 MB) so với VGG16. | * Hiệu suất rất tốt với các mạng nơ-ron sâu, và các khối residual giúp duy trì hiệu suất ổn định ngay cả khi số lượng lớp tăng. | * Các lớp chia sẻ và sử dụng thông tin từ tất cả các lớp trước đó, giúp tăng cường khả năng học và cải thiện việc truyền thông tin qua mạng. | * giảm số lượng phép tính và tham số đáng kể so với tích chập thông thường. | EfficientNet đã đạt được kết quả ấn tượng trên các tập dữ liệu lớn như ImageNet. So với các kiến trúc khác như ResNet và MobileNet, EfficientNet cho thấy:  **Độ chính xác cao hơn** với số lượng tham số và phép tính ít hơn.  **Hiệu quả tính toán tốt hơn**, phù hợp để triển khai trên các thiết bị có tài nguyên hạn chế. |
| Kết luận |  | LeNet là mạng nơ ron đầu tiên được xây dựng cho các tác vụ đơn giản như nhận số viết tay, trong khi ALEX là một bước tiến lớn hơn trong nhận dạng hình ảnh quy mô lớn, mở ra kỷ nguyên của DEEPL trên dữ liệu phức tạp hơn | * Được xây dựng bao gồm các khối tích chập * Sử dụng các các khối giúp mã nguồn ngắn gọn, thiết kế mạng phức tạp một cách hiệu quả hơn * Simonyan và Zisserman đã phát hiện rằng sử dụng mặt nạ nhỏ, sâu hiệu quả hơn sử dụng mặt nạ lớn nhưng ít tầng | * Thường có số lượng tham số và yêu cầu tính toán thấp hơn so với các mạng nơ-ron sâu khác nhờ vào việc sử dụng các khối Inception để thay thế các lớp tích chập lớn bằng nhiều lớp nhỏ hơn. |  | * Mỗi lớp kết nối trực tiếp với tất cả các lớp trước đó trong khối dense, tạo ra nhiều kết nối hơn so với ResNet. * Thường có ít tham số hơn so với ResNet với cùng độ sâu vì việc sử dụng lại đặc trưng từ các lớp trước đó giúp giảm số lượng bộ lọc cần thiết. |  | EfficientNet đã mở ra một kỷ nguyên mới trong việc phát triển các mạng nơ-ron hiệu quả, đặc biệt là cho các thiết bị có tài nguyên tính toán hạn chế. Nhờ vào **Compound Scaling** và các thành phần tối ưu như **MBConv**, **SE Block**, và **Swish Activation**, EfficientNet có thể đạt được độ chính xác cao với số lượng phép tính và tham số thấp hơn so với các mô hình truyền thống. |