**YOLO nhận dạng và phân lớp đối tượng**

1. **Mô hình CNN**

Sử dụng package slim của tensorflow để khai báo mô hình vgg-16.

Với ảnh đầu vào là 224x224, chia ảnh thành lưới các ô vuông với gridsize là 7x7, và với mỗi ô vuông sử dụng linear regression dự đoán 2 boundary box và có 3 class nên output\_shape cần có là 7x7x13.Mô hình CNN có 5 tầng max pooling với size 2x2 sẽ giảm 32 lần kích thước ảnh xuống còn 7x7 ở output đầu ra. Đồng thời thay vì sử dụng tầng full connected ở các tầng cuối cùng, em thay thế bằng tầng 1x1 conv với 13 feature maps để output shape cho ra 7x7x13.

Kiến trúc CNN gồm 5 lớp Conv, 5 lớp MaxPooling, sau đó là lớp Conv với kernel size = (1,1) thay cho tầng fully conntected. Hàm IOU để đánh giá độ chính xác của các boundary box. Loss\_function được sử dụng để tính độ lỗi cho mô hình dựa vào 3 độ lỗi con: Classification loss, Localization loss, Confidence loss.

Cấu trúc layer của mô hình

* Lớp Conv2D (16, kernel (3,3), input\_shape = (224,224,3))
* Lớp MaxPooling2D (pool\_size = (2,2))
* Lớp Conv2D (32, kernel (3,3))
* Lớp MaxPooling2D (pool\_size = (2,2))
* Lớp Conv2D (64, kernel (3,3))
* Lớp MaxPooling2D (pool\_size = (2,2))
* Lớp Conv2D (128, kernel (3,3))
* Lớp MaxPooling2D (pool\_size = (2,2))
* Lớp Conv2D (256, kernel (3,3))
* Lớp MaxPooling2D (pool\_size = (2,2))

2 lớp Conv2D tiếp theo để thay thế cho fully connected

* Lớp Conv2D (512, kernel (1,1))
* Lớp Conv2D (13, kernel (1,1), activation=None)
* Hàm IOU, Loss
* Optimizer cho hàm Loss: Adam
* Hàm filter giữ lại các boundary box có xác suất của lớp là lớn nhất và lớn hơn 0.2

1. **Bộ dữ liệu (Dataset)**

Do độ chính xác còn thấp và khả năng xử lý còn yếu nên em sử dụng tập dữ liệu gồm các hình học đơn giản để train và test trước, chưa áp dụng cho các mẫu xe thực tế.