R-CNN

Александр Захаров, M05-894a

1 Постановка задачи

Описываемая далее модель решает задачу выделения объектов. То есть необходимо выделить объекты на изображении и отнести их к определенному классу. Соответственно модель должна по входному изображению предсказать множество объектов, их границы(bounding box) и класс, к которому он относится.

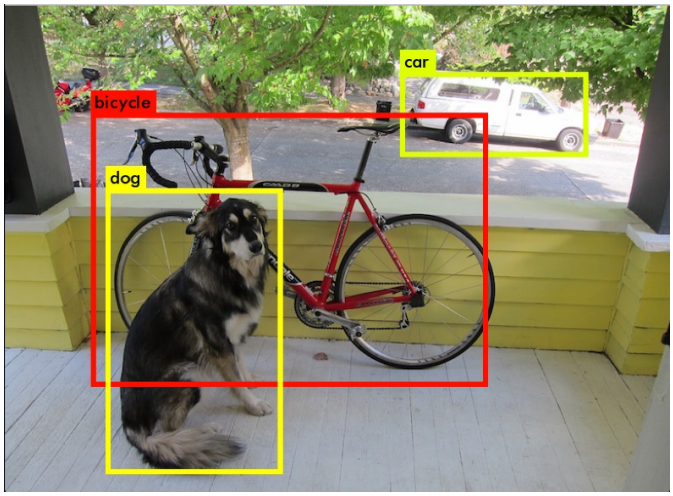

Рис. 1: Пример выделения

Метрикой, которая будет дальше чаще всего встречаться является Mean Average Precision, но сначала опишем, как считать успешность детекции. Так как выделение точного bounding box является слишком сложной и бессмысленной задачей, для определения удачности детектирования будет считаться следующая метрика, по предсказанному и базовому расположению объекта $IoU = \frac{AreaofOverlap}{AreaofUnion}$. Соответственно, если $IoU > p$, некоторой границы, будем считать что объекты сматчены. Далее проверяется совпадение классов. Далее для подсчета MAP сортируем предсказания в соответсвии с уверенностью модели. Далее считаем precision и recall на всех префиксах этого множества. Тогда $MAP = \frac{1}{|recalls|} \sum_{i=1}^{|recalls|} max(precision(recall) * I(recall >= recalls_i))$

В качестве примера реколы можно выбрать как $0; 0.1...1$

2 Описаниие архитектуры и применения

В модели есть несколько важных частей.

1. Генератор гипотез
2. Сверточная нейросеть
3. Классификатор SVM

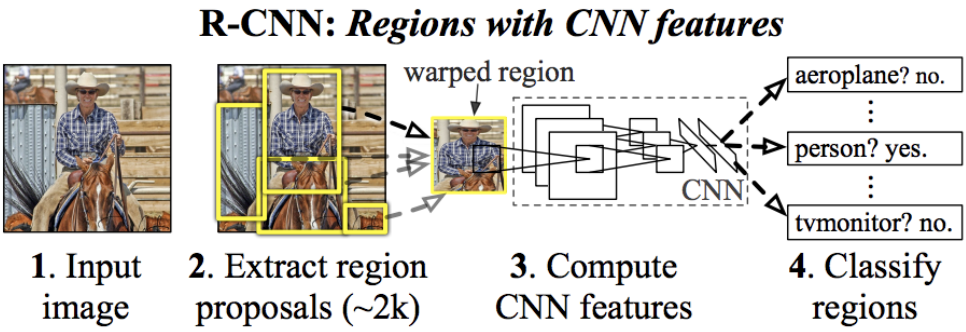

Рис. 2: Демонстрация работы модели по шагам

2.1 Генератор гипотез

В качестве генератора гипотез в модели R-CNN используется алгоритм selective search. Он не является обучаемым. В нем генерируется 2000 гипотез объектов на изображении.

2.2 Сверточная нейросеть

Сверточная нейросеть используется для извлечения фичей из гипотез выделенных selective search. Для этого каждая гипотеза приводится к размеру, который принимает на вход сеть, фичами же являются выходные знаечения последнего слоя сети(полносвязного). В оригинальной статье использовали архитектуру AlexNet.

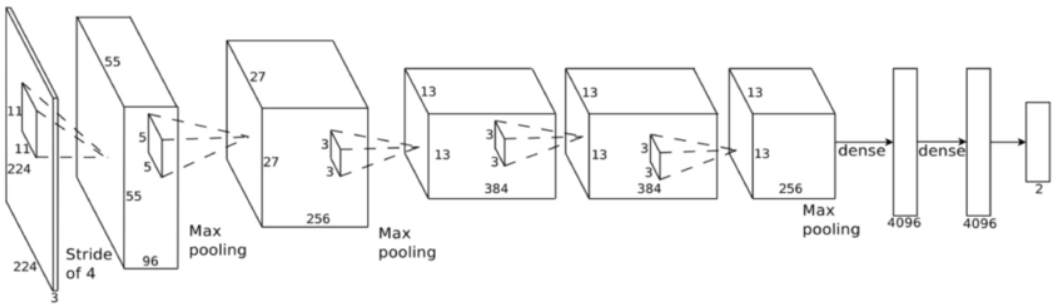

Рис. 3: Архитектура AlexNet

В дальнейшем в данной модели использовали и другие архитектуры, которые хорошо показали себя в классификации, например, VGG. Более новые архитектуры не использовались, в связи с тем, что были выпущены качественные улучшения данной модели. При этом сверточная нейросеть используется предобученная для задачи классификации.

2.3 SVM

Для определения класса, к которому относится объект используется линейный классификатор SVM. Также для более повышения качетсва детектирования, производится регрессия границ bounding box. Победивший класс определяется по наибольшей уверенности, среди всех. Для обучения необходимо обучить всего одну матрицу. Авторы утверждают, что SVM был добавлен ими, так как они не хотели файнтюнить сверточную сеть, чтобы вставить в нее softmax, однака даже с файнтюнингом, качество softmax добаленного прямо к нейросети было ниже.

3 Особенности обучения

В обучении этой архитектуры можно выделить следующие интересные пункты

1. Сверточная нейросеть сначала обучается для задачи классификации изображений ImageNet.
2. Затем нейросеть доучивают для задачи детекции, подавая только корректные bounding box из training set датасета для детекции с соответствующим классом.
3. Изначально SVM обучают с правильными примерами и случайными в качестве негативных. Затем производится дополнительная добыча негативных примеров из наиболее уверенных ошибок детекции.
4. Для увеличения размера обучающей выборки к правильным примерам добавляют также немного сдвинутые их версии.

4 Результаты работы

В качестве валидационного датасета использовался датасет PASCAL-VOC2007 и 2010. Бейзлайном выступала DPM модель на основе гистограм градиентов(HoG).

Модель	VOC-2007	VOC-2010
DPM	33.7	29.6
R-CNN(only cnn)	54.2	50.2
R-CNN(full)	58.5	53.7
R-CNN(VGG)	66.0	62.9

Таблица 1: Сравнение моделей

5 Достоинства и недостатки

5.1 Достоинства

На момент выхода эта модели показывала наилучшее качество в задаче обнаружения объектов на ихобраэении, так как она одна из первых максимально использовала весь потенциал сверточных нейросетей, которые уже держали лидирующие места в задачи классификации изображений.

5.2 Недостатки

1. Очень долго занимает применение сети для одного изображения 48 секунд.
2. Качество детекции сильно ограничено качеством selective search, так как он не обучаемый.
3. Много времени уходит на обучение 84 часа.

6 Краткий обзор улучшений

6.1 Fast R-CNN

Одной из главные проблем R-CNN является применение нейросети к каждой гипотезе, это занимает достаточно продолжительное время. В этой архитектуре сеть применяется только один раз ко всему изображению. На первом этапе используется полносверточная сеть, гипотезы же выделяются анологично обычной R-CNN, далее эти гипотезы при помощи RoI pooling слоя, выделяются фичи соответствующие этим гипотезам, и эти вектора подаются в полносвязные слои на выходе нейросети, их которых мы получаем предсказание класса гипотезы и границы, при помощи регрессии. Этой архитектуре требуется в 10 раз меньше времени на обучение и в 20 раз меньше времени на применении. Качество обнаружения при этом увеличивается примерно на 10%.

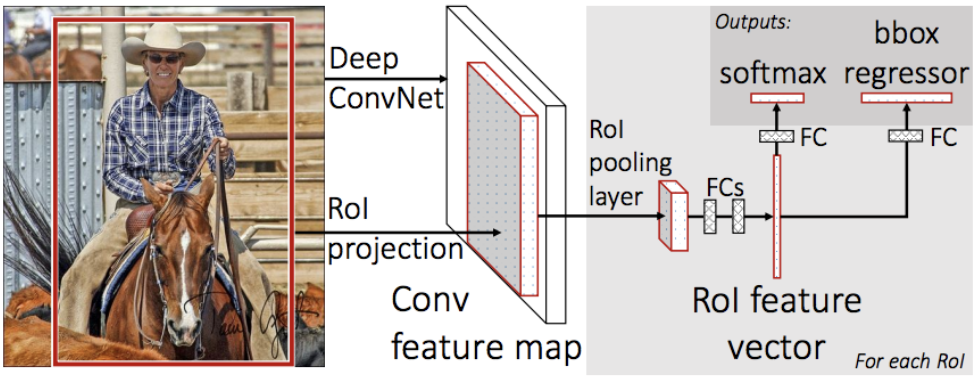

Рис. 4: Архитектура Fast R-CNN

В данной архитектуре скорость применения ограничивается уже не применением нейросети, а работой selective search.

6.2 Faster R-CNN

В данной модели отказались от selective search, как генератора гипотез. Весь процесс аналогичен Fact R-CNN, но предсказания гипотез делает отдельная нейросеть, которая обучается на фичах, полученных от полносверточной сети. Такое изменение позволило довести скорость работы почти до реального времени 5 кадров в секунду.

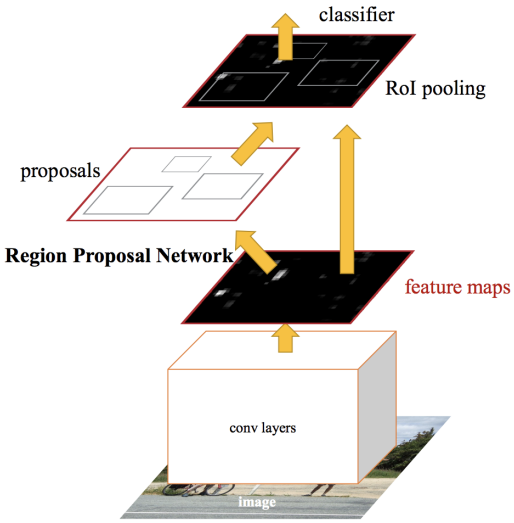

Рис. 5: Архитектура Faster R-CNN