

РЕАЛИЗАЦИЯ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ НА АРХИТЕКТУРЕ YOLOv4 С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ PYTORCH

*Н.Г. Марков, д.т.н., профессор ОИТ ИШИТР,
С.А. Ткачёв, аспирант гр. А9-39
Томский политехнический университет
E-mail: sat12@tpu.ru*

Введение

В настоящее время создание систем компьютерного зрения (СКЗ) продолжает оставаться актуальным научным направлением. Использование сверточных нейронных сетей (СНС) является наиболее популярным средством реализации алгоритмического обеспечения СКЗ.

Для мониторинга окружающей среды необходимо разработать СКЗ на основе СНС, которые позволяют детектировать объекты на изображениях земной поверхности в реальном времени.

Целью данной статьи является реализация СНС на архитектуре YOLOv4 с использованием библиотеки PyTorch для решения задачи детектирования объектов на изображениях земной поверхности в реальном времени.

Архитектура YOLOv4

Существует два типа архитектур СНС для детектирования объектов: одноэтапные и двухэтапные. Одноэтапная архитектура СНС способна детектировать объекты без необходимости предварительного этапа. Двухэтапная архитектура СНС, напротив, использует предварительный этап, на котором обнаруживаются важные области, а затем классифицируются, чтобы увидеть, был ли детектирован объект в этих областях. Преимущество одноэтапной архитектуры СНС заключается в том, что она способна быстро делать прогнозы, что позволяет использовать ее в режиме реального времени.

На рисунке 1 представлена схема, которая показывает различие двух типов архитектур.

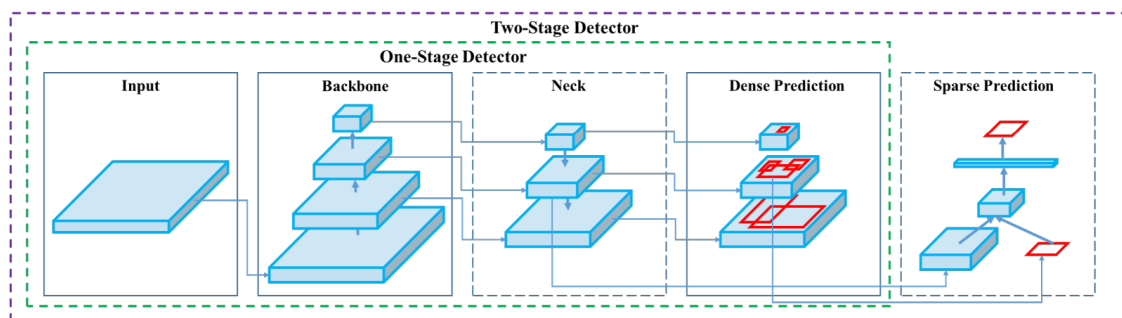


Рис. 1. Различие одноэтапных архитектур от двухэтапных.

На схеме выше указаны 4 блока, которые следуют после входного изображения:

- Backbone (DenseNet, CSP, CSPDarknet53)
- Neck (FPN, SPP)
- Dense Prediction (Head) – используется в одноэтапных архитектурах (YOLO, SSD и др.)
- Sparse Prediction – используется в двухэтапных архитектурах (Faster-R-CNN и др.)

Использование блока Backbone – это один из способов, с помощью которого можно повысить точность, спроектировать более глубокую СНС и увеличить сложность модели. В архитектуре YOLOv4 используется CSPDarknet53 [1] в качестве блока Backbone.

Основная цель блока Neck – добавить дополнительные слои между блоком Backbone и блоком Head, чтобы получить «более богатую» пространственную и семантическую информацию. В архитектуре YOLOv4 используются концепция FPN [2], которая постепенно заменяется модифицированными SAM, PAN и SPP [3] в качестве блока Neck.

В блоке Head происходит тот же процесс, что и в архитектуре YOLOv3 – СНС определяет координаты bounding-boxes (x , y , w , h) вместе с оценкой достоверности для класса. Цель состоит в том, чтобы разделить изображение на сетку, состоящую из нескольких ячеек, а затем для каждой ячейки предсказать вероятность наличия объекта с помощью anchor-boxes. На выходе получается вектор с координатами bounding-boxes и классами вероятностей.

Последний блок Sparse Prediction не используется в архитектуре YOLOv4. Наглядно архитектура YOLOv4 [4] представлена на рисунке 2.

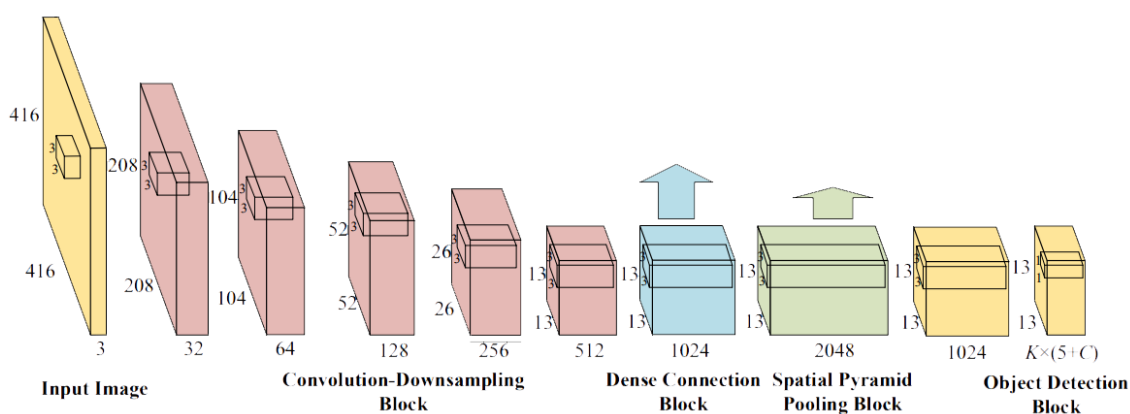


Рис. 2. Архитектура YOLOv4.

Реализация и тестирование СНС

Для реализации СНС использовалась библиотека PyTorch – это набирающий популярность мощный инструмент для глубокого машинного обучения.

После обучения СНС на собственном наборе данных, состоящим из 485 цифровых изображений дронов размера 416x416, проводилось ее тестирование на новых наборах данных.

Результаты тестирования СНС представлены в таблице 1.

Таблица 1. Результаты тестирования СНС

Набор данных	Размер, шт.	Время детектирования, мин.	Точность модели, %
Цифровые изображения дронов	1800	147	82
Цифровые изображения дронов, крупный план	200	16	75
Цифровые изображения дронов, дальний план	200	15	61
Термальные изображения дронов, ч/б диапазон	2000	162	24
Термальные изображения дронов, и/к диапазон	400	35	11

Точность партии определялась как сумма правильных предсказаний, деленная на размер партии. Общая точность модели определялась как среднее значение полученных точностей на каждой итерации. Важно уточнить, что на каждом изображении присутствовал только один дрон.

Заключение

В результате проведения тестирования можно сделать вывод о том, что СНС является работоспособной и детектирует дроны на цифровых изображениях с точностью 82%. Ухудшение результатов СНС показывает в следующих случаях:

- когда меняется размер дрона на изображении;
- когда детектирование проводится на термальных изображениях.

Список использованных источников

1. Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection / arXiv: journal. – 2020.
2. Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie. Feature Pyramid Networks for Object Detection / arXiv: journal. – 2017.
3. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition / arXiv: journal. – 2015.
4. Zhanchao Huang, Jianlin Wang. DC-SPP-YOLO: Dense Connection and Spatial Pyramid Pooling Based YOLO for Object Detection / arXiv: journal. – 2019.