

**REAL-TIME ОБРАБОТКА ВИДЕО: СРАВНИТЕЛЬНЫЙ АНАЛИЗ
АРХИТЕКТУР YOLO и SSD**

**REAL-TIME VIDEO PROCESSING: COMPARATIVE ANALYSIS
OF YOLO AND SSD ARCHITECTURES**

Михайлов А.М., студент

Евдокимова С.А., к.т.н., доцент

ФГБОУ ВО «Воронежский государственный лесотехнический университет
имени Г.Ф. Морозова»

г. Воронеж, Россия

tg3381234@gmail.com

Mikhailov A. M., student

Evdokimova S.A., CSc (Engineering), Associate Professor

FSBEI HE «Voronezh State University of Forestry and Technologies named
after G.F. Morozov»

Voronezh, Russian Federation

Аннотация: С развитием нейронных сетей появилась потребность в real-time детекции. В данных задачах важны точность и скорость. В статье приводится сравнительный анализ YOLO и SSD - двух ключевых архитектур, которые активно применяются в настоящее время.

Summary: With the development of neural networks, there is a need for real-time detection. In these tasks, accuracy and speed are important. This article provides a comparative analysis of YOLO and SSD - two key architectures that are actively used today.

Ключевые слова: real-time, YOLO, SSD, однопроводные детекторы, машинное обучение, компьютерное зрение.

Keywords: real time, YOLO, SSD, single pass detectors, machine learning, computer vision.

С развитием нейросетей появилась потребность в распознавании объектов в реальном времени, так как уже не для всех задач достаточно определить детекцию на статическом изображении. Детекция в потоке уже решает многие

проблемы в видеонаблюдении, в автономных автомобилях, анализе трафика, AR/VR и в спортивной аналитике[1, 2]. Зачастую в этих задачах используются две наиболее популярные архитектуры – YOLO (You Only Look Once) и SSD. Проведем сравнительный анализ этих двух архитектур, рассмотрим, как они работают, и в чем заключаются их ключевые различия.

До появления таких современных решений как YOLO, SSD существовали классические (двухэтапные) методы вроде Faster R-CNN, но они были слишком медленные и поэтому на смену к ним пришли одноэтапные детекторы[3, 4]. Поэтому в этой статье мы не будем затрагивать классические методы.

YOLO – является одной из самых популярных детекций в реальном времени на сегодняшний день. За 10 лет архитектура прошла путь от простой CNN модели до anchor-free и transformer-гибридных моделей [5-9]. Это семейство однопроходных (one-stage) детекторов объектов, которые предсказывают классы объектов и bounding box'ы за один проход. Так как YOLO не использует отдельные модули для генерации регионов (region proposals), то он значительно быстрее двухэтапных детекторов.

Архитектура нейронной сети алгоритма YOLO состоит из 24 сверточных слоев и двух полносвязных слоя в конце [5-7]. На вход подается изображение 448×448, которое разбивается на одинаковые квадратные ячейки 64×64, образуя таблицу 7×7. В дальнейшем все операции происходят с этими ячейками (рисунок 1).

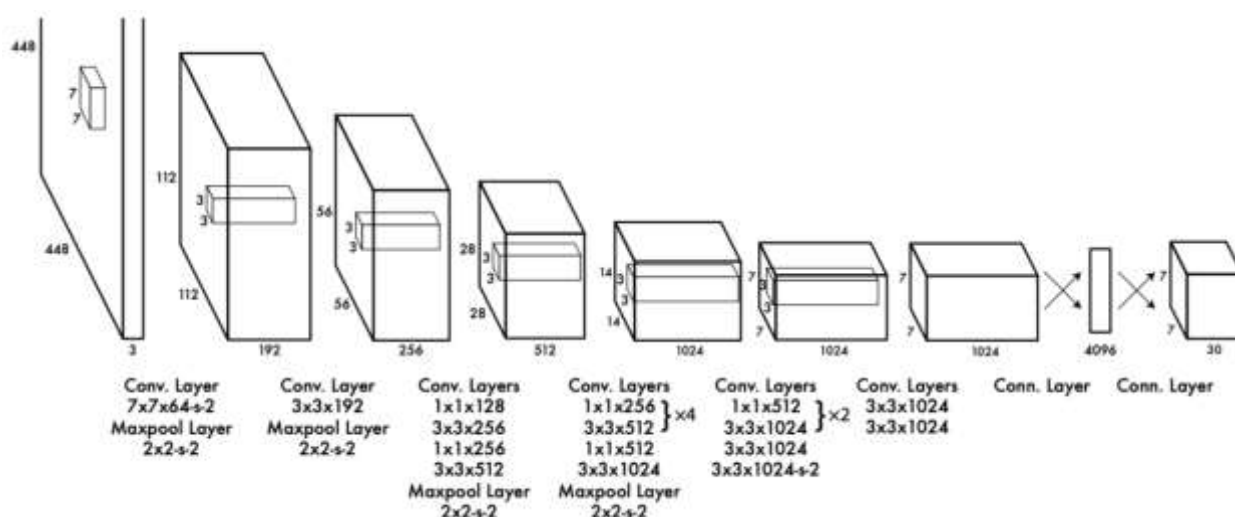


Рисунок 1 – Архитектура YOLO

Координаты ограничивающей рамки (bounding box) изображения определяются по формулам [6]:

$$b_x = \sigma(t_x) + c_x ;$$

$$b_y = \sigma(t_y) + c_y;$$

где b_x, b_y – координаты центра предсказанного bounding box,

t_x, t_y – предсказанные нейросетью смещения,

σ – сигмоида (чтобы ограничить значение в пределах ячейки),

c_x, c_y – координаты левого верхнего угла grid-ячейки.

$$b_w = p_w + e^{t_w};$$

$$b_h = p_h + e^{t_h}$$

где b_w, b_h – ширина и высота bbox (относительно анкера),

p_w, p_h – ширина и высота анкера (anchor box),

t_w, t_h – предсказанные нейросетью коэффициенты масштабирования.

Из плюсов YOLO можно выделить:

- очень высокая скорость (от 45 FPS на GPU);
- простота развёртывания (ONNX, TensorRT).

Минусы:

- хуже работает с мелкими объектами.

SSD – Single Shot MultiBox Detector – также является однопроходным (one stage) детектором, был представлен в 2016 году, не требует генерации RoI[10]. SSD использует предобученные модели, такие как VGG-16 или ResNet и добавляет дополнительные сверточные сети на выходе. На рисунке 2 можно заметить, что чем дальше мы продвигаемся по слоям, тем более крупные объекты мы можем редактировать.

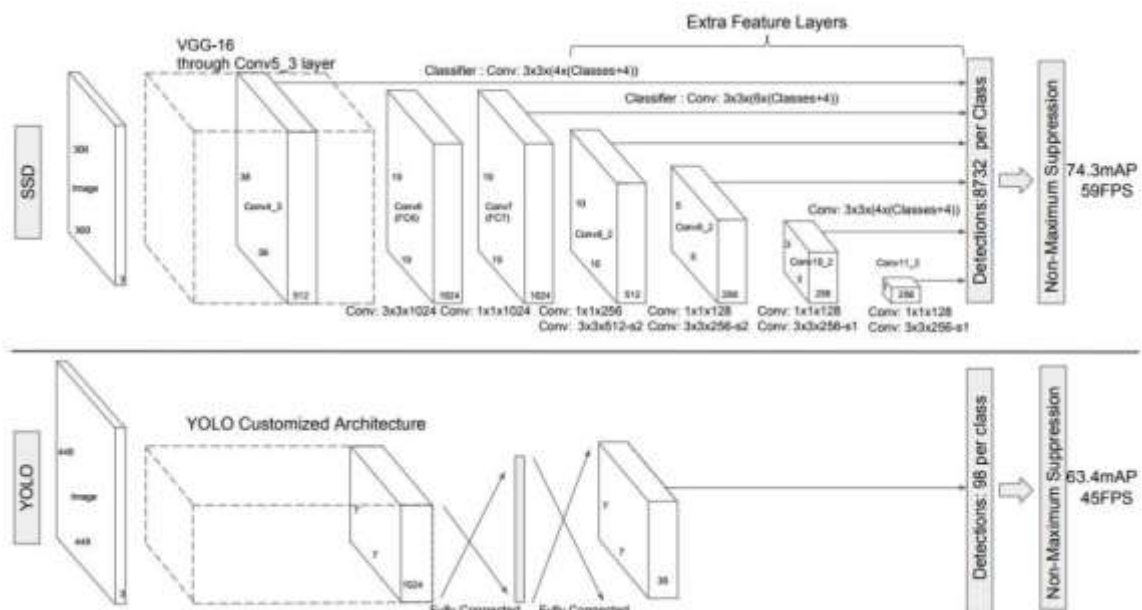


Рисунок 2 – Архитектура SSD

У SSD proposal фиксированы для всех возможных вариантов, что значительно ускоряет модель. На рисунке 3 видим, что feature map разбивается на 8×8 и в каждой ячейке мы ставим 6 анкоров. В отличие от YOLO мы предсказываем поправки на сколько нам нужно передвинуть центр объекта и изменить его ширину и длину.

Таким образом, смещение координат проходят по формуле [10]:

$$\hat{g}_x = (g_x - d_x) / d_w;$$

$$\hat{g}_y = (g_y - d_y) / d_h;$$

где g_x, g_y – координаты центра истинного bbox (ground truth);

d_x, d_y – координаты центра anchor box;

d_w, d_h – ширина и высота anchor box.

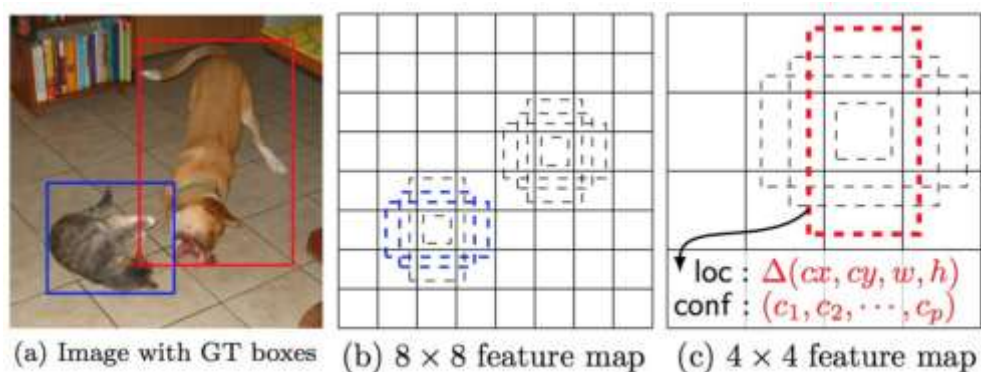


Рисунок 3 – Принципы работы SSD

Для начала обозначим проблемы, которые возникают при real-time детекции, так как именно решения этих проблем являются особенностью каждой архитектуры.

Главные требования к алгоритмам обработки видео:

- высокая FPS-скорость (≥ 30 кадров/сек для плавного потока);
- баланс между точностью и быстродействием;
- поддержка edge-устройств (телефоны, камеры, дроны).

Именно по этим показателям будем оценивать наши модели. После исследования скорости и точности двух моделей на минутном видео получили следующие результаты (таблица 1).

Также были проведены исследования с разными моделями YOLO по ключевым характеристикам – скорость и точность (рисунок 4).

Полученные данные показывают, что YOLO является более быстрой моделью, хоть и не такой точной как SSD. Однако, учитывая, что в real-time си-

стемах, скорость важнее точности, эту архитектуру больше используют и внедряют в боевые проекты.

Таблица 1 – Результаты сравнения SSDи YOLO

	SSD	YOLO (v8)
Тип детектора	Однопроходный (one-stage)	Однопроходный (one-stage)
Backbone	VGG-16 / ResNet	CSPDarknet / EfficientNet
Anchor Boxes	Фиксированные (default boxes)	Anchor-free (в YOLOv8)
Multi-scale	Feature maps разных уровней	FPN / PANet
Скорость (FPS)	20-60 (на GPU)	50-200 (на GPU)
Точность (mAP@0.5)	70-80 (COCO)	60-75 (COCO)
Оптимизация	Устаревшая (2016)	Поддержка TensorRT, OpenVINO

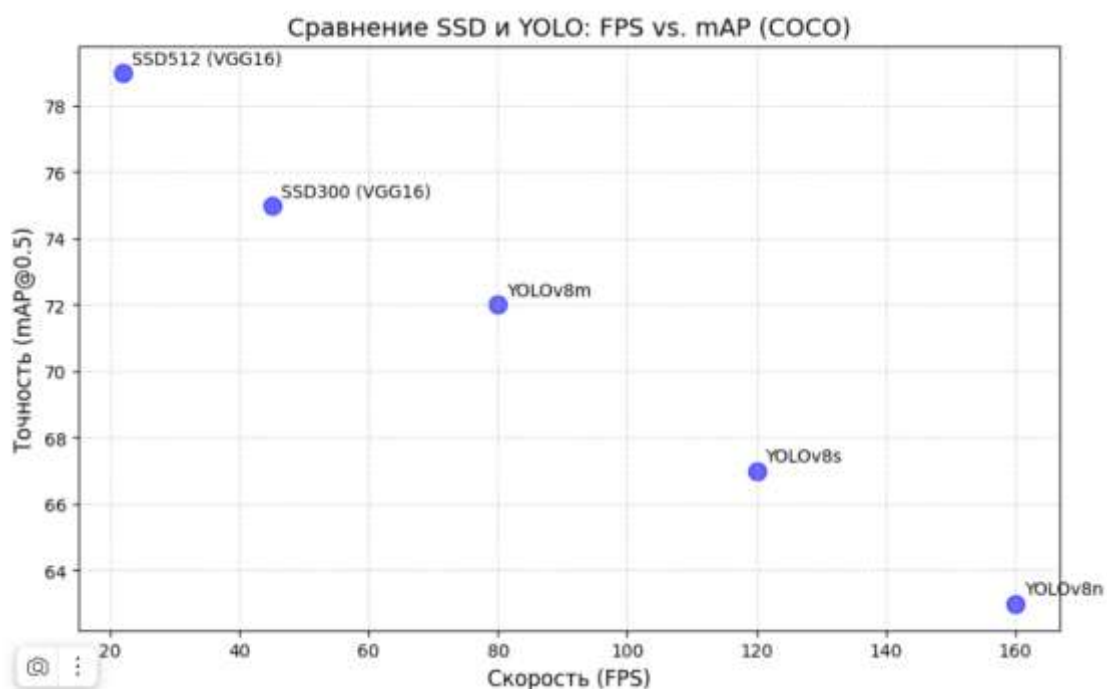


Рисунок 4 – Сравнение скорости и точности YOLO и SSD

В заключение можно сказать, что архитектура YOLO является хорошим выбором для новых проектов. Рекомендуется применять эту модель в видеонаблюдение, дронах, автономных автомобилях. Также существует много других моделей, которые требуют исследований и внимания. Из самых популярных можно выделить: EfficientDet, DETR, Vision Transformers. Также проводятся и внедряются квантовые версии YOLO для микроконтроллеров. Как видно, об-

ласть активно исследуется, и каждая из моделей имеет свои специфики и подходит для разных задач.

Список литературы

1. Макаров, М.А. Искусственный интеллект в промышленной информатике : учебное пособие / М.А. Макаров, И.Ю. Зайцев. – М. : МИРЭА – РТУ, 2024. – 118 с.
2. Vision-based approach for human motion detection and smart appliance control / S. Swami [etal.] // IAES International Journal of Robotics and Automation. – 2024. – Vol. 13, № 4. – Pp. 445-451.
3. Howard, A. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications / A. Howard, M. Zhu, B. Chen [et al.] – arXiv:1704.04861, 2017. – 9 с.
4. Tan, M. EfficientDet: Scalable and Efficient Object Detection / M. Tan, R. Pang, Q. V. Le // CVPR. – 2020. – P. 10781-10790.
5. Redmon, J. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon, S. Divvala, R. Girshick, A. Farhadi. – arXiv:1506.02640, 2015. – 10 p.
6. Redmon, J. YOLO9000: Better, Faster, Stronger / J. Redmon, A. Farhadi. – URL: <https://arxiv.org/pdf/1612.08242> (дата обращения: 16.03.2025).
7. История YOLO – самой известной архитектуры компьютерного зрения. – URL: <https://datasecrets.ru/articles/20> (дата обращения: 16.03.2025).
8. Чжу, И. Исследование нейронных сетей YOLO / И. Чжу, Т.П. Новикова // Моделирование информационных систем и технологий : сборник материалов международной научно-практической конференции. – Воронеж, 2022. – С. 338-342.
9. Bochkovskiy, A. YOLOv4: Optimal Speed and Accuracy of Object Detection / A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao // arXiv preprint. – 2020. – arXiv:2004.10934.
10. Liu, W. SSD: Single Shot MultiBox Detector / W. Liu, D. Anguelov, D. Erhan [et al.] – ECCV 2016. – Springer, 2016. – С. 21-37.

References

1. Makarov, M.A. Artificial intelligence in industrial informatics : a textbook / M.A. Makarov, I.Yu. Zaitsev. – M. : MIREA – RTU, 2024. – 118 p.

2. Vision-based approach for human motion detection and smart appliance control / S. Swami [et al.] // IAES International Journal of Robotics and Automation. – 2024. – Vol. 13, № 4. – Pp. 445-451.
3. Howard, A. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications / A. Howard, M. Zhu, B. Chen [et al.] – arXiv:1704.04861, 2017. – 9 c.
4. Tan, M. EfficientDet: Scalable and Efficient Object Detection / M. Tan, R. Pang, Q. V. Le // CVPR. – 2020. – P. 10781-10790.
5. Redmon, J. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon, S. Divvala, R. Girshick, A. Farhadi. – arXiv:1506.02640, 2015. – 10 p.
6. Redmon, J. YOLO9000: Better, Faster, Stronger / J. Redmon, A. Farhadi. – URL: <https://arxiv.org/pdf/1612.08242>(accessed: 03/16/2025).
7. The history of YOLO, the most famous computer vision architecture. – URL: <https://datasecrets.ru/articles/20> (date of request: 03/16/2025).
8. Zhu, I. Research of YOLO neural networks / I. Zhu, T.P. Novikova // Modeling of information systems and technologies : proceedings of the international scientific and practical conference. –Voronezh, 2022. –pp. 338-342.
9. Bochkovsky, A. YOLOv4: Optimal Speed and Accuracy of Object Detection / A. Bochkovsky, C.-Y. Wang, H.-Y. M. Liao // arXiv preprint. – 2020. – arXiv:2004.10934.
10. Liu, W. SSD: Single Shot MultiBox Detector / W. Liu, D. Anguelov, D. Erhan [et al.] – ECCV 2016. – Springer, 2016. – C. 21-37.