# Применение трансформерных моделей для интеллектуального мониторинга фотоэлектрических систем

*М.Д. Трегубенко*

*Южный федеральный университет, Ростов-на-Дону*

**Аннотация:** В данной работе проводится всесторонний сравнительный анализ производительности современных архитектур глубокого обучения для задачи обнаружения объектов. Исследование сосредоточено на двух основных семействах моделей: трансформерных архитектурах, включая «Трансформер обнаружения объектов» (Detection Transformer — DETR) и его усовершенствованные варианты, такие как «Трансформер обнаружения объектов в реальном времени» (Real-Time Detection Transformer — RT-DETR), «Трансформер обнаружения объектов с мелкозернистым уточнением распределения» (Detection Transformer with Fine-grained Distribution Refinement — D-FINE) и «Трансформер обнаружения объектов с улучшенным сопоставлением» (Detection Transformer with Improved Matching — DEIM), а также на популярных одностадийных детекторах семейства «Посмотри на изображение один раз» (You Only Look Once — YOLO), в частности, на версиях YOLOv11 и YOLOv12. Оценка моделей выполняется на специализированном наборе данных изображений, который содержит в себе различные дефекты солнечных панелей и состоит из пяти классов, что позволяет выявить сильные и слабые стороны каждой архитектуры в контексте специфических прикладных задач.
**Ключевые слова:** компьютерное зрение, энергетика, энергетическая инфраструктура, глубокое обучение, солнечные панели, нейронные сети, детекция, трансформеры, одностадийные детекторы, распознавание образов.

## Введение

Задача детекции объектов является одной из ключевых областей компьютерного зрения, находящей применение в различных сферах, включая возобновляемые источники энергии (ВИЭ). Анализ изображений энергетического оборудования и солнечных панелей требует разработки моделей, способных эффективно обрабатывать вариативность освещения, ракурсов и специфические дефекты. Данное направление автоматизации является актуальным и перспективным, как в ТЭК [1][2], так и в других областях [3][4].

Эволюция методов детекции прошла путь от классических подходов к современным системам глубокого обучения. На сегодняшний день выделяются два основных направления: одностадийные детекторы семейства

«Посмотри на изображение один раз» (You Only Look Once — YOLO) [5], известные высокой скоростью работы, и трансформерные архитектуры класса «Трансформер обнаружения объектов» (Detection Transformer — DETR) [6], предлагающие новый подход к детекции через прямое предсказание множества объектов без использования якорных рамок и подавления немаксимумов.

Выбор оптимальной модели для специализированных задач остается сложным, так как производительность на общих бенчмарках не всегда переносится на узкоспециализированные данные.

Цель работы — сравнение современных трансформерных моделей: DETR, «Трансформер обнаружения объектов в реальном времени» (Real-Time Detection Transformer — RT-DETR) [7], «Трансформер обнаружения объектов с мелкозернистым уточнением распределения» (Detection Transformer with Fine-grained Distribution Refinement — D-FINE) [8], «Трансформер обнаружения объектов с улучшенным сопоставлением» (Detection Transformer with Improved Matching — DEIM) [9] с моделями YOLOv11 [10] и YOLOv12 [11] на данных дефектов солнечных панелей. Задачи исследования включают анализ архитектур моделей, сбор и обработку специализированного набора данных, определение метрик качества и проведение экспериментального сравнения.

Повышение точности автоматического выявления дефектов фотоэлектрических модулей напрямую влияет на надежность и эффективность солнечных электростанций, особенно в условиях удаленного мониторинга в рамках цифровизации ТЭК.

## Теоретические основы

**DETR.** DETR революционизировал подход к детекции объектов, представив ее как задачу прямого предсказания множества, что позволило устранить генерацию якорных рамок и подавление немаксимумов.

Архитектура DETR состоит из трех компонентов: предобученая светрочная нейронная сеть, например «Остаточная сеть» (Residual Network — ResNet) [12], для извлечения признаков, трансформер-кодер преобразует признаки в последовательность токенов, трансформер-декодер обрабатывает N запросов объектов параллельно и «прогнозирующая прямая сеть» (Feed-Forward Network — FFN) предсказывает координаты рамки и метку класса для каждого из N выходов. Обучение основывается на глобальной функции потерь с оптимальным сопоставлением предсказаний и истинных объектов через венгерский алгоритм. Преимущества DETR: полностью сквозная архитектура без подавления немаксимумов и якорей, хорошая производительность на крупных объектах. Недостатки: медленная сходимость и низкая эффективность для мелких объектов.

**RT-DETR.** Алгоритм RT-DETR адаптирован для работы в реальном времени. Ключевые нововведения: эффективный гибридный кодер, обрабатывающий многомасштабные признаки из последних трех стадий базовой сети; выбор запросов, учитывающий предсказанный коэффициент пересечения по объединению (intersection over union — IoU), для повышения качества начальных предсказаний; масштабируемый декодер, позволяющий гибко настраивать скорость вывода путем изменения количества слоев без переобучения.

**D-FINE.** D-FINE переопределяет задачу регрессии рамок как итеративное уточнение вероятностных распределений. Основные компоненты: Мелкозернистое уточнение распределения трансформирует процесс от предсказания координат к моделированию распределений для каждой стороны рамки, обеспечивая детализированное представление локализации; Глобальная оптимальная локализация с самообучением передает информацию о локализации с глубоких слоев на ранние, ускоряя сходимость и упрощая задачу для глубоких слоев.

**DEIM.** DEIM решает проблему медленной сходимости DETR через увеличение количества и качества сопоставлений. Плотное однозначное взаимодействие интегрирует различные техники аугментации данных, создавая композитные изображения с большим количеством объектов, что увеличивает число обучающих пар при сохранении структуры сопоставления «один-к-одному» и смягчает проблему «разреженного надзора». Функция потерь, учитывающая сопоставимость (Matchability-Aware Loss — MAL) — новая функция потерь, разработанная для оптимизации обучения на сопоставлениях различного качества. Она масштабирует величину потерь в зависимости от IoU между предсказанной и истинной рамками, уделяя больше внимания и придавая больший вес низкокачественным, но все же положительным, сопоставлениям по сравнению с некоторыми другими функциями потерь, как например, вариофокальная функция потерь. Концептуально, MAL можно представить как:

$$L_{MAL} = f(IoU, p) \times L_{base}, \quad (1)$$

где $L_{base}$ - базовая функция потерь (например, фокальная или кросс-энтропийная для классификации и $L_1$ – норма или обобщенное пересечение по объединению для регрессии), а $f(IoU, p)$ - модулирующая функция, которая увеличивает вклад от пар с низким IoU, но все еще считающихся положительными.

**YOLO.** Семейство моделей YOLO представляет собой популярный класс одностадийных детекторов, обеспечивающих баланс между скоростью и точностью для приложений реального времени.

Общая архитектура YOLO обрабатывает изображение за один проход, предсказывая ограничивающие рамки и вероятности классов для всех объектов. Входное изображение делится на сетку S×S, где каждая ячейка

отвечает за объекты, центры которых в ней расположены, предсказывая B рамок и C вероятностей классов.

Типичная архитектура YOLO состоит из трех основных частей. Базовая сеть извлекает признаки из входного изображения. Часто используются модифицированные версии известных архитектур. Например, перекрестная сеть (Cross-Stage-Partial Network — CSPNet) [13] или более новая сеть с остаточным эффективным объединением слоев (Residual Efficient Layer Aggregation Networks — R-ELAN) [14] в YOLOv12. «Шея» агрегирует и комбинирует признаки, извлеченные на разных уровнях базовой сети, для формирования многомасштабных представлений. Примеры включают сеть пирамиды признаков и сеть агрегации путей. «Голова» использует агрегированные признаки для предсказания координат ограничивающих рамок, оценок уверенности и вероятностей классов.

**YOLOv11.** YOLOv11 предлагает усовершенствования архитектуры для повышения производительности.

Ключевые компоненты: Слой «C3k2» улучшает извлечение признаков с помощью сверточных ядер 2×2, снижая вычислительную сложность. Входной тензор x разделяется на части $x_1$ и $x_2$, где $x_2$ проходит через сверточные блоки перед конкатенацией с $x_1$.

Быстрый пространственный пирамидальный пулинг (Spatial Pyramid Pooling Fast — SPPF) эффективно захватывает признаки на нескольких масштабах через последовательное применение максимального пулинга с различными окнами, снижая вычислительную нагрузку.

Сверточный блок с параллельным пространственным вниманием (Convolutional block with Parallel Spatial Attention — C2PSA) интегрирует механизм параллельного пространственного внимания, позволяя сети фокусироваться на значимых областях карты признаков и подавлять шум, что особенно полезно для сложных сцен.

**YOLOv12.** YOLOv12 представляет собой эволюционный шаг в развитии семейства YOLO, интегрирующий механизмы внимания в традиционную CNN-архитектуру для повышения точности при сохранении высокой скорости работы.

Основные инновации YOLOv12 включают:

R-ELAN: модифицированная базовая сеть с добавлением остаточных связей (с коэффициентом масштабирования $\alpha$=0.01), улучшающих поток градиентов и предотвращающих их затухание. Математически это выражается как $y = x + \alpha F(x)$, где $F(x)$ представляет основные вычисления блока.

Разделяемые свертки 7×7: использование разделяемых сверток с большим ядром (7x7) для эффективного пространственного кодирования, что позволяет механизмам внимания обрабатывать позиционную информацию без явных позиционных кодировок и значительно сокращает вычислительную сложность.

Область-ориентированное внимание с использованием «мгновенного внимания» (FlashAttention): ключевой механизм в «шее» модели, разделяющий карту признаков на несколько зон и применяющий внимание к каждой зоне независимо. Это снижает вычислительную сложность с $O(N^2)$ до линейной зависимости. Основная формула внимания:

$$Attention(Q_i, K_i, V_i) = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i, \quad (2)$$

где Q, K, V - матрицы запросов, ключей и значений. Вычисления ускоряются с помощью FlashAttention через оптимизацию функции «софтмакс» (softmax) и эффективное использование видеопамяти.

## Набор данных

Набор данных содержит 7375 изображений, 11085 аннотаций классов и 5 классов солнечных панелей в видимом и инфракрасном спектрах:

- «Целая панель» — панель чистая и не разбитая (2656 аннотаций).
- «Грязная панель» — панель загрязнена (2524 аннотации).
- «Заснеженная панель» — панель занесена снегом (2083 аннотации).
- «Выгоревшая панель» — видны выгоревшие области в ИК спектре (2042 аннотации).
- «Сломанная панель» — панель разбита (1780 аннотаций).

Набор данных был расширен до 13337 изображений с помощью следующих методов аугментации:

- Поворот на 90 градусов влево и вправо.
- Приближение от 0 до 15 %.
- Вращение от -15 до 15 градусов.
- Сдвиг на 10 градусов по горизонтали и вертикали.

Датасет был разделены на обучающую, валидационную и тестовую выборки в соотношении 90:5:5 соответственно.

Пример изображений из набора данных представлен на рис. 1.

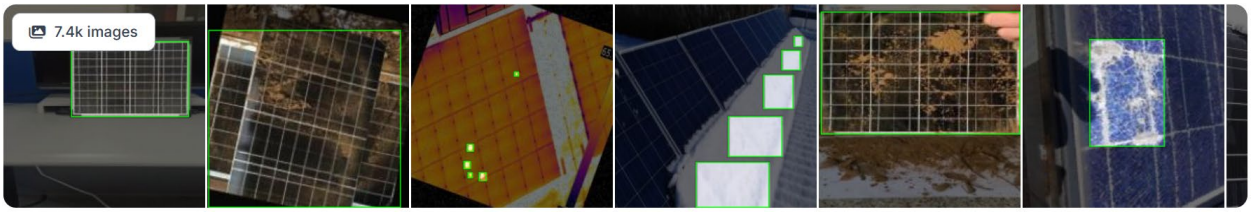


Рис. 1 - изображения набора данных

## Параметры эксперимента и метрики качества

**Используемая аппаратная часть и ПО.** Для обеспечения сопоставимости результатов измерения производительности используется метрика частоты кадров в секунду (Frames per second — FPS). В качестве основной платформы для тестирования используется графический процессор

NVIDIA 4090 laptop. Тесты также включают процессор Intel i9-14900HX 2.2 ГГц и 32 ГБ оперативной памяти для корректной работы моделей.

Эксперименты проводятся с использованием актуальной версии библиотеки глубокого обучения «Pytorch», сопутствующих ей зависимостей, а также соответствующих версий специальных библиотек компании NVIDIA для оптимизации вычислений на видеокарте. Операционная система – Ubuntu Linux.

Измерение FPS выполняется на тестовой выборке набора данных. Для получения стабильных и репрезентативных результатов измерения проводятся многократно, и вычисляются средние значения. Размер батча при измерении скорости вывода устанавливается равным 1, чтобы имитировать сценарий обработки одиночных изображений в реальном времени. Используется точность вычислений FP16.

**Метрики качества.** Intersection over Union (IoU), также известный как индекс Жаккара, является фундаментальной метрикой для оценки точности локализации предсказанной ограничивающей рамки по отношению к истинной рамке. IoU измеряет степень их перекрытия:

$$IoU = \frac{AreaofOverlap}{AreaofUnion}, \tag{3}$$

где «Area of Overlap» – это площадь пересечения предсказанной и истинной рамок, а «Area of Union» – это площадь их объединения.

Значение IoU варьируется от 0 (нет перекрытия) до 1 (полное совпадение). В задачах обнаружения объектов IoU используется для классификации предсказаний:

- *Истинно-положительные (True Positive – TP):* Предсказанная рамка корректно классифицирует объект и ее IoU с соответствующей истинной рамкой превышает заданный порог (например, IoU>0.5).

- *Ложноположительные (False Positive – FP):* Предсказанная рамка либо неверно классифицирует объект, либо ее IoU с истинной рамкой ниже порога, либо объект предсказан там, где его нет.
- *Ложноотрицательные (False Negative – FN):* Истинный объект не был обнаружен моделью.

**Точность (Precision) и Полнота (Recall).** *Точность (Precision — P):* эта метрика показывает, какая доля объектов, идентифицированных моделью как положительные, действительно являются таковыми. Высокая точность означает малое количество ложных срабатываний:

$$P = \frac{TP}{TP + FP} \tag{4}$$

*Полнота (Recall — R):* эта метрика показывает, какую долю всех истинно положительных объектов модель смогла правильно обнаружить. Высокая полнота означает малое количество пропущенных объектов:

$$R = \frac{TP}{TP + FN} \tag{5}$$

*Средняя точность (Average Precision — AP):* AP является одной из наиболее распространенных метрик для оценки качества моделей обнаружения объектов. Она агрегирует информацию всей PR-кривой в одно числовое значение. AP рассчитывается как площадь под PR-кривой.

*Усредненная средняя точность (Mean Average Precision — mAP):* mAP – это усредненное значение AP по всем рассматриваемым классам объектов. Эта метрика дает общую оценку производительности модели по всему набору классов:

$$mAP = \frac{1}{N_{classes}} \sum_{i=1}^{N_{classes}} AP_i, \tag{6}$$

где $APi$ – это Average Precision для i-го класса, а $N_{classes}$ – общее количество классов.

## Результаты эксперимента

Результаты моделей представлены в Таблице № 1. Метрики подсчитаны на тестовой выборке.

Таблица № 1

Результаты экспериментов

| № | Модель | Params (M) | Precision | Recall | mAP@ 0.5 | mAP@ [.5:.95] | FPS |
|---|---|---|---|---|---|---|---|
| 1 | Yolov11x | 57 | 0.9183 | 0.9246 | 0.9490 | 0.7961 | 121 |
| 2 | Yolov11l | 25 | 0.8967 | 0.9029 | 0.9332 | 0.7770 | 217 |
| 3 | Yolov12x | 59 | 0.9242 | 0.9305 | 0.9551 | 0.8034 | 109 |
| 4 | Yolov12l | 27 | 0.9016 | 0.9088 | 0.9389 | 0.7807 | 198 |
| 5 | D-FINE-L | 31 | 0.9067 | 0.9127 | 0.9369 | 0.7860 | 174 |
| 6 | D-FINE-X | 62 | 0.9312 | 0.9376 | 0.9621 | 0.8121 | 110 |
| 7 | DEIM-D-FINE-L | 31 | 0.9183 | 0.9246 | 0.9490 | 0.7961 | 174 |
| 8 | DEIM-D-FINE-X | 62 | 0.9395 | 0.9460 | 0.9713 | 0.8232 | 110 |
| 9 | RT-DETRv2l | 42 | 0.8899 | 0.8961 | 0.9194 | 0.7714 | 160 |
| 10 | RT-DETRv2x | 76 | 0.9195 | 0.9258 | 0.9507 | 0.7975 | 104 |

## Заключение

На основе проведенного исследования модель DEIM-D-FINE-X продемонстрировала наивысшую точность (mAP@[.5:.95]=0.8232) на датасете дефектов солнечных панелей. Современные трансформерные архитектуры, включая D-FINE, DEIM-D-FINE и RT-DETRv2, значительно оптимизированы по скорости и теперь способны работать в реальном времени (160-180 FPS на RTX 4090), сохраняя высокую точность.

Ключевые преимущества трансформеров заключаются в их способности обрабатывать глобальные зависимости через механизмы внимания, что особенно важно для сложных сцен и детекции мелких объектов. Хотя архитектуры YOLO сохраняют лидерство по скорости в некоторых конфигурациях, трансформеры активно сокращают разрыв в производительности, одновременно устанавливая новые стандарты точности.

Перспективные направления исследований включают дальнейшую оптимизацию скорости DETR-подобных моделей, разработку гибридных архитектур, объединяющих преимущества трансформеров и сверточных сетей, а также создание универсальных методов адаптации моделей к специфическим задачам с минимальными затратами.

## Литература

1. Гольдзон И.А., Завьялов А.П., Лопатин А.С. О перспективах использования систем автоматизированного контроля технического состояния оборудования объектов ТЭК с использованием беспилотных технологий // Автоматизация, телемеханизация и связь в нефтяной промышленности. – 2019. – № 6(551). – С. 25–30. – DOI: 10.33285/0132-2222-2019-6(551) -25-30

2. Трегубенко М.Д. Анализ методов компьютерного зрения для распознавания дефектов солнечных панелей // Известия ЮФУ. Технические науки. – 2024. – № 4. – С. 144–151. – DOI: 10.18522/2311-3103-2024-4-144-151

3. Бобков А.В., Ду Кэхао, Дай Ифань, Ван Чжун, Чэнь Хао. Облегченная модифицированная сеть YOLO для детектирования объектов дорожной сцены // Инженерный вестник Дона, 2025, №8. URL: ivdon.ru/magazine/archive/n8y2025/10269.

4. Зубова Д.П., Семенист С.А., Широбокова С.Н. Проектирование компонента классификации объектов и интерпретации их действий с использованием методов компьютерного зрения и машинного обучения // Инженерный вестник Дона, 2025, №6. URL: ivdon.ru/magazine/archive/n6y2025/10148.

5. Hasan R.H., Hassoo R.M., Aboud I.S. Yolo Versions Architecture // International Journal of Advances in Scientific Research and Engineering. – 2023. – Vol. 9, No. 11 – P. 73.

6. Carion N. et al. End-to-end object detection with transformers // European Conference on Computer Vision. – Cham: Springer, 2020. – pp. 213–229.

7. Zhao Y. et al. Detrs beat yolos on real-time object detection // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2024. – pp. 16965–16974.

8. Peng Y. et al. D-FINE: redefine regression task in DETRs as fine-grained distribution refinement // arXiv preprint arXiv: 2410.13842. – 2024.

9. Huang S. et al. DEIM: DETR with improved matching for fast convergence // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2025. – pp. 15162–15171.

10. Khanam R., Hussain M. Yolov11: An overview of the key architectural enhancements // arXiv preprint arXiv: 2410.17725. – 2024.

11. Tian Y., Ye Q., Doermann D. Yolov12: Attention-centric real-time object detectors // arXiv preprint arXiv: 2502.12524. – 2025.

12. He K. et al. Deep residual learning for image recognition // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2016. – pp. 770–778.

13. Wang C.-Y. et al. CSPNet: A new backbone that can enhance learning capability of CNN // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2020. – pp. 1551–1560.

14. Zhang X. Efficient Long-Range Attention Network for Image Superresolution // European Conference on Computer Vision. – Cham: Springer, 2022. – Vol. 13667. – pp. 649–667.

## References

1. Gol'dzon I.A., Zav'yalov A.P., Lopatin A.S. Avtomatizaciya, telemehanizaciya i svyaz' v neftyanoj promyshlennosti. 2019. No 6(551). pp. 25-30. DOI: 10.33285/0132-2222-2019-6(551) -25-30

2. Tregubenko M.D. Izvestiya YuFU. Tekhnicheskie nauki. 2024. No 4. pp. 144-151. DOI: 10.18522/2311-3103-2024-4-144-151

3. Bobkov A.V., Du Kekhao, Dai Ifan', Van Chzhung, Chen' Khao. Inzhenernyj vestnik Dona. 2025. No 8. URL: ivdon.ru/magazine/archive/n8y2025/10269

4. Zubova D.P., Semenist S.A., Shirbobokova S.N. Inzhenernyj vestnik Dona. 2025. No 6. URL: ivdon.ru/magazine/archive/n6y2025/10148

5. Hasan R.H., Hassoo R.M., Aboud I.S. International Journal of Advances in Scientific Research and Engineering. 2023. Vol. 9, No. 11. P. 73

6. Carion N. et al. European Conference on Computer Vision. Cham: Springer, 2020. pp. 213-229.

7. Zhao Y. et al. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024. pp. 16965-16974.

8. Peng Y. et al. arXiv preprint arXiv: 2410.13842. 2024.

9. Huang S. et al. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2025. pp. 15162-15171.

10. Khanam R., Hussain M. arXiv preprint arXiv: 2410.17725. 2024.

11. Tian Y., Ye Q., Doermann D. arXiv preprint arXiv: 2502.12524. 2025.

12. He K. et al. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016. pp. 770-778.

13. Wang C.-Y. et al. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020. pp. 1551-1560.

14. Zhang X. European Conference on Computer Vision. Cham: Springer, 2022. Vol. 13667. pp. 649-667.

**Дата поступления: 14.08.2025**

**Дата публикации: 25.09.2025**