

# АНАЛИТИЧЕСКИЙ ОБЗОР НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ ОБНАРУЖЕНИЯ ВОЗГОРАНИЙ ПРИ ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ

**В.А. Зорин, Р.В. Мещеряков**

*Институт проблем управления им. В.А. Трапезникова Российской академии наук, г. Москва, Российская Федерация  
E-mail: mrv@ipu.ru*

**Аннотация.** Благодаря развитию нейронных сетей робототехнические системы научились обнаруживать и распознавать объекты в режиме реального времени, что поспособствовало применению РТС в автономном режиме в самых различных сценариях, в том числе для обнаружения возгораний при чрезвычайных ситуациях. В статье рассматривается ряд существующих алгоритмов обнаружения на основе нейронных сетей, включая сверточные нейронные сети, региональные сверточные нейронные сети и их варианты, глубокие нейронные сети со сверточной долговременной кратковременной памятью (ConvLSTM), методы, интегрирующие глубокое обучение с корреляционной фильтрацией посредством самостоятельного обучения, сиамские нейронные сети для отслеживания целей и семейство алгоритмов YOLO (You Only Look Once). Описаны основные характеристики и различия между нейросетевыми алгоритмами, а также приведено сравнение эффективности, по критериям средней точности (mAP – mean Average Precision) и скорости обработки – частоте кадров в секунду (FPS – Frame Per Second).

Выводы статьи дают представление о компромиссах между точностью, скоростью и требованиями конкретных задач в задачах обнаружения, что позволяет сделать обоснованный выбор по применению того или иного алгоритма.

*Ключевые слова:* нейросетевые алгоритмы; БПЛА; обнаружение; сверточные нейронные сети; YOLO.

## Введение

Надежное обнаружение возгораний является критически важным компонентом локализации чрезвычайных ситуаций. Методы, основанные на нейронных сетях, стали уравновешивающим компромиссом между вычислительной эффективностью и точностью обнаружения.

Методы глубокого обучения значительно улучшили компьютерное зрение, поэтому всё чаще используются в современных системах обнаружения и, например, предупреждения о столкновениях [1].

Технологии искусственного интеллекта (ИИ) и искусственные нейронные сети открыли возможность обрабатывать сложные данные для обнаружения, распознавания и отслеживания. Например, системы автоматического распознавания целей, базирующиеся на сверточных нейронных сетях (CNN), доказали свою эффективность в определении целей на инфракрасных изображениях при обучении на синтетических данных [2], где мерой эффективности выступил критерий высокой надежности обнаружения.

Системы обхода препятствий на базе технологий искусственного интеллекта позволяют БПЛА безопасно маневрировать и перемещаться в сложных условиях [3]. Методы машинного обучения позволяют адаптироваться к среде, распознавать препятствия и избегать их во время полета. Используемое для таких задач обучение с подкреплением показало эффективность в решении проблемы автономного избегания препятствий и отслеживания целей.

Архитектуры нейронных сетей, такие как YOLO, были разработаны для улучшения обнаружения небольших объектов за счет расширения возможностей извлечения признаков и локализации, например, для подсчета яблок в сложных садах для сбора урожая с помощью роботизированной руки [4].

В статье рассматриваются известные алгоритмы на основе нейросетей и как показатели производительности с точки зрения точности (mAP) и возможностей работы в режиме реального времени (FPS) помогают сделать наиболее предпочтительный выбор для применения.

## 1. Методы обнаружения, основанные на нейронных сетях

### 1.1. Сверточные нейронные сети (CNN)

CNN – класс методов глубокого обучения для обработки данных, которые имеют схему сетки, похожей на организацию зрительной коры животных [5], и предназначена для автоматического и адаптивного изучения пространственных иерархий признаков – от низкоуровневых до высокоуровневых шаблонов [6].

Некоторые методы обнаружения полагаются на CNN в качестве своей основной модельной архитектуры. Юнг и др. [7], Ким и др. [8] предлагают CNN для прогнозирования предупреждающих ситуаций на примерах с участием пешеходов и велосипедистов, если в первом случае авторы используют текущий кадр изображения для прогнозирования, то во втором обнаруживают и отслеживают объекты в последовательности кадров.

CNN составляют основу многих современных систем обнаружения. Они характеризуются иерархией сверточных слоев, которые извлекают пространственные объекты из изображений, за которыми следуют объединенные слои для уменьшения размерности и полностью связанные слои для классификации.

В основе CNN лежит операция свертки, которая является действием над парой матриц  $A(n_x \times n_y)$  и  $B(m_x \times m_y)$ , результатом которого является матрица  $C = A \times B$ , размером  $(n_x - m_y + 1) \times (m_x - m_y + 1)$ .

Каждый элемент результата вычисляется как скалярное произведение матрицы  $B$  и некоторой подматрицы  $A$  такого же размера (подматрица определяется положением элемента в результате). Логический смысл свертки заключается в том, что чем больше величина элемента свертки, тем больше эта часть матрицы  $A$  была похожа на матрицу  $B$  (в смысле скалярного произведения). Поэтому матрицу  $A$  называют изображением, а матрицу  $B$  – фильтром или образцом. Для двумерного изображения  $A$  на входе и ядра свертки (фильтра)  $B$  операция свертки может быть записана в виде (1), где  $A(i, j)$  – значение пикселя в позиции  $A(i, j)$  изображения на входе,  $K(m, n)$  – весовой коэффициент ядра свертки (фильтра) в позиции  $(m, n)$ ,  $C(i, j)$  – матрица на выходе в позиции  $(i, j)$ .

$$C(i, j) = (A \cdot B)(i, j) = \sum_{m=0}^{M-1} \cdot \sum_{n=0}^{N-1} (i+m, j+n) \cdot K(m, n). \quad (1)$$

### Основные характеристики CNN

*1. Локальные поля восприятия и общие весовые коэффициенты уменьшают количество параметров*

Локальные поля восприятия и общие весовые коэффициенты – два фундаментальных принципа сверточных нейронных сетей. Вместо того чтобы соединять каждый нейрон в одном слое со всеми нейронами в предыдущем слое, нейрон в CNN связан только с небольшой локализованной областью входного изображения. Такое локальное подключение позволяет сети фиксировать пространственно-локальные узоры, такие как края или текстуры, при этом значительно сокращая общее количество параметров. В CNN один и тот же набор весовых коэффициентов (фильтров) применяется к разным областям входного изображения. Такое распределение весовых коэффициентов означает, что одни и те же объекты распознаются независимо от их положения на изображении. Повторно используя фильтры для всего входного сигнала, CNN всё больше ограничивают количество параметров по сравнению с полностью подключенными уровнями.

В совокупности эти конструктивные решения повышают эффективность вычислений в CNN и помогают избежать переобучения, поскольку сети требуется изучать меньше параметров, сохраняя при этом основные характеристики данных.

2. Алгоритм эффективен при захвате пространственных иерархий, что имеет решающее значение для распознавания объектов

Данная способность обусловлена уникальными архитектурными особенностями, которые позволяют сети фокусироваться на пространственно-локальных узорах, таких как края или текстуры, и являясь основными строительными блоками изображений, поскольку нейроны в CNN подключены только к небольшой локализованной области входного изображения.

При наложении нескольких сверточных слоев, CNN учатся обнаруживать простые закономерности в начальных слоях, которые затем объединяются в последующих слоях для формирования более сложных объектов. Это иерархическое обучение отражает то, как люди воспринимают визуальную информацию, позволяя сети распознавать объекты на различных уровнях абстракции.

## 1.2. Региональные сверточные нейронные сети (R-CNN) и их вариации

**R-CNN.** В основе работы региональных сверточных нейронных сетей лежит получение набора регионов, которые предположительно содержат объекты для классификации, и затем их дальнейшая обработка в сверточной нейронной сети. R-CNN [9] сначала генерирует предложения по регионам, используя такой алгоритм, как создание рамок [10]. Области предложений вырезаются из изображения и изменяются в размерах. Затем CNN классифицирует обрезанные и измененные в размерах области. Наконец, ограничивающие рамки предложения по региону уточняются с помощью метода опорных векторов (SVM), который обучается с использованием функций CNN. Алгоритм отличается более высокой точностью обнаружения, но низкой скоростью обработки информации.

**Fast R-CNN.** Как и в детекторе R-CNN, в детекторе Fast R-CNN [11] также используется алгоритм, подобный краевым блокам, для генерации предложений по регионам. В отличие от детектора R-CNN, который обрезает и изменяет размер предложений по регионам, детектор Fast R-CNN обрабатывает все изображение целиком. В то время как детектор R-CNN должен классифицировать каждый регион, Fast R-CNN объединяет функции CNN, соответствующие каждому предложению по региону. Fast R-CNN более эффективен, чем R-CNN, потому что в быстром R-CNN детекторе вычисления для перекрывающихся регионов выполняются совместно.

Скорость обработки информации повышается примерно на 5 кадров в секунду при одновременном увеличении карты примерно на 66 % для наборов данных VOC (Visual Object Classes).

**Faster R-CNN.** Модель Faster R-CNN [12] основана на модели Fast R-CNN. Добавляет сеть региональных предложений (RPN) для генерации региональных предложений непосредственно в сети вместо использования внешнего алгоритма, такого как пограничные блоки. RPN использует привязочные блоки для обнаружения объектов. Создание предложений по регионам в сети происходит быстрее и лучше соответствует данным. Достигает около 70 % mAP (VOC) примерно при 7 кадрах в секунду, эффективно балансируя скорость и точность.

**Mask R-CNN.** Модель Mask R-CNN [13] расширяет Faster R-CNN, добавляя ветку для прогнозирования маски объекта параллельно с существующей ветвью для распознавания ограничительной рамки. Модель Mask R-CNN проста в обучении и лишь немного замедляет Faster R-CNN, работая со скоростью 5 кадров в секунду. Модель полезна для задач сегментации с аналогичной точностью обнаружения (около 70 % карты).

## 1.3. Сети ConvLSTM

Глубокая нейронная сеть со слоями сверточной долговременной памяти – тип рекуррентной нейронной сети, объединяющая сверточные нейронные сети CNN и долгосрочную память рекуррентных LSTM-нейронных сетей для пространно-временного прогнозирования, которая имеет сверточные структуры как при переходе входного сигнала в состояние, так и при переходе от состояния к состоянию [14]. ConvLSTM определяет будущее состояние определенной ячейки в сетке по входным данным и прошлым состояниям ее локальных соседей. Это может быть легко достигнуто с помощью оператора свертки в переходах от состояния к состоянию и от входа к состоянию.

Для описания математической модели предположим, что наблюдаем динамическую систему в пространственной области, представленной сеткой  $M \times N$ , состоящей из  $M$  строк и  $N$  столбцов.

Внутри каждой ячейки сетки есть  $P$  измерений, которые меняются со временем. Таким образом, наблюдение в любой момент времени может быть представлено тензором  $X \in RP \times M \times N$ , где  $R$  обозначает область наблюдаемых признаков. Если периодически записывать наблюдения, то получим последовательность тензоров  $X_t$ . Чтобы избавиться от недостатка LSTM в части использования полных соединений при переходе от входа к состоянию и от состояния к состоянию, в которых пространственная информация не кодируется, входы  $X_t$ , выходы ячейки  $C_t$ , скрытые состояния  $H_t$  и элементы управления  $i_t, f_t, o_t$  в ConvLSTM представлены трехмерными тензорами. Можем представить их в виде векторов, расположенных на пространственной сетке. ConvLSTM определяет будущее состояние определенной ячейки в сетке по входным данным и прошлым состояниям её локальных соседей.

Этого можно легко достичь, используя оператор свертки при переходах от состояния к состоянию и от входа к состоянию. Ключевые уравнения ConvLSTM приведены в (2) где  $*$  – оператор свертки,  $[o]$  – произведение Адамара.

$$\begin{aligned}i_t &= \sigma(W_{xt} * X_t + W_{ht} * H_{t-1} + W_{ct}[o]C_{t-1} + b_i), \\f_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf}[o]C_{t-1} + b_f), \\C_t &= f_t[o]C_{t-1} + i_t[o]\tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c), \\o_t &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co}[o]C_{t-1} + b_o), \\H_t &= o_t[o]\tanh(C_t).\end{aligned}\tag{2}$$

В задачах обнаружения столкновения Стрикленд и др. [15] предложили использовать ConvLSTM для обработки видео из нескольких источников. Авторы выполнили несколько проходов через нейронную сеть с отсевом, случайно сбрасывая соединения в единицах ConvLSTM, что дало оценку неопределенности и предсказанной вероятности [16]. В алгоритме повышена производительность обнаружения за счет сохранения временной согласованности между кадрами. Исследование [17] показало улучшение mAP примерно на 10 % для задач последовательного обнаружения по сравнению с традиционными покадровыми методами, которое объясняется способностью ConvLSTM эффективно фиксировать как пространственные, так и временные зависимости в последовательных данных.

ConvLSTM был применен к задачам семантической сегментации, что привело к заметному повышению производительности по показателю среднего числа пересечений (mIoU). Исследование [18] показало способность ConvLSTM моделировать временные последовательности, что способствует более точному прогнозированию сегментации.

### 1.4. Глубокое обучение с корреляционной фильтрацией и самообучением

Глубокое обучение с корреляционной фильтрацией и самоконтролем (Deep Learning with Correlation Filtering and Self-Supervised Learning) – достаточно новый подход, который сочетает в себе преимущества глубоких нейронных сетей с традиционными методами корреляционной фильтрации, усиленными возможностями обучения на основе немаркированных данных.

Корреляционные фильтры используются в визуальном отслеживании, поскольку они позволяют быстро сопоставлять шаблоны путем вычисления взаимных корреляций между изученными представлениями объектов и целевыми шаблонами. В платформы глубокого обучения модуль корреляционной фильтрации может быть интегрирован в качестве дифференцируемого уровня, который уточняет карты объектов, позволяя сети быстро локализовать объекты или отслеживать цели в последовательных кадрах. Такая интеграция не только ускоряет обработку, но и повышает надежность в динамических средах.

Автономное обучение решает проблему ограниченного количества помеченных данных путем постановки предварительных, которые позволяют нейросети извлекать полезные представления функций из немаркированных данных. Интегрированные в систему глубокого обучения с корреляционной фильтрацией методы самоконтроля помогают улучшить качество объектов и приводят к повышению эффективности обнаружения и отслеживания без чрезмерной зависимости от аннотированных наборов данных.

Объединив корреляционную фильтрацию с самоконтролируемым глубоким обучением, исследователи [19] разработали методы, которые используют как пространственное сопоставление с образцом, так и всестороннее изучение признаков без контроля. Такая синергия обеспечивает более точное и эффективное отслеживание и обнаружение.

### 1.5. Сети сиамских близнецов (Сиамские нейронные сети)

Сиамские сети (Siamese Twin Networks), названные так в честь сиамских близнецов, – две идентичные подсети с общим весом (общим весовым коэффициентом) для сравнения и сопоставления характеристик между целевым шаблоном и регионами-кандидатами в последующих кадрах. Такая архитектура позволяет модели эффективно вычислять показатели сходства, обеспечивая надежное отслеживание в режиме реального времени. Обе ветви Сиамской сети идентичны, что означает, что они извлекают похожие элементы из целевого и потенциального патчей. Такая согласованность имеет решающее значение для точного сравнения сходства. После выделения признаков используется показатель сходства на корреляционном уровне, чтобы определить, насколько область-кандидат соответствует целевому шаблону.

Благодаря оптимизированной архитектуре и распределению нагрузки Сиамские сети могут работать на высоких скоростях.

**SiamFC** (Fully-Convolutional Siamese Networks – полностью сверточные Сиамские сети). Пример [20], продемонстрировавший, как Сиамские сети могут быть использованы для визуального отслеживания, обеспечивая высокую производительность в режиме реального времени и устойчивость к изменениям внешнего вида объекта.

**SiamRPN** (Siamese Region Proposal Network – Сиамская сеть региональных предложений) [21] расширяет рамки Сиамской сети за счет включения механизма региональных предложений, который улучшает процесс отслеживания и повышает точность. SiamRPN работает со скоростью ~160 кадров в секунду.

Используя глубокое представление объектов и согласованное распределение веса, эти сети могут эффективно справляться с различиями во внешнем виде объектов, их масштабе и скрытии. Архитектура, присущая Сиамским сетям, позволяет быстро вычислять сходство, что делает их популярным выбором в решении задач с минимальной задержкой.

### 1.6. Алгоритмы семейства YOLO

**YOLO** (“You Only Look Once” – «Вы Смотрите Только Один раз») [22] представляет семейство одноступенчатых детекторов, которые предсказывают ограничительные рамки и классифицируют вероятности непосредственно по полным изображениям за одну оценку.

Сетевая структура выглядит как обычная CNN со сверточными и максимальными объединенными слоями. Математическое описание функции свертки мы рассмотрели в (1), отдельный интерес представляет описание отображения результата работы алгоритма через функцию потерь.

Предсказание ограничительной рамки имеет 5 компонентов:  $(x, y, w, h, \text{доверие})$ . Координаты  $(x, y)$  представляют центр рамки относительно местоположения ячейки сетки. Эти координаты нормализованы между 0 и 1. Размеры рамки  $(w, h)$  также нормализованы до  $[0, 1]$  относительно размера изображения.

1. Функция потерь, связанная с прогнозируемым положением ограничительной рамки  $(x, y)$ , где  $x'$  и  $y'$  – фактическое положение на основе обучения. Считаем сумму по каждому предсказателю ограничивающей рамки  $(j = 0 \dots B)$  каждой ячейки сетки  $(i = 0 \dots S^2)$ .  $1_{ij}^{obj}$  – принимает значение 1, если объект присутствует в ячейке и 0 в остальных случаях.

$$\lambda_{x,y} \sum_{i=0}^{S^2} \cdot \sum_{j=0}^B \left( 1_{ij}^{obj} \left( (x - x')^2 + (y - y')^2 \right) \right). \quad (3)$$

2. Функция потерь, связанная с оценкой достоверности для каждого предсказателя ограничивающей рамки.  $C$  – оценка достоверности,  $C'$  – это пересечение при объединении предсказанного ограничивающего прямоугольника с основным значением истинности.  $1_{ij}^{obj} = 1$ , когда в ячейке

есть объект, и 0 – в противном случае,  $1_{ij}^{no}$  – полная противоположность. Параметры  $\lambda$ , как и в первой части, используются для разного веса частей функций потерь. Это необходимо для повышения стабильности модели. Самый высокий штраф – для координатных предсказаний ( $\lambda = 5$ ) и самый низкий – для доверительных предсказаний, когда объект отсутствует ( $\lambda_{no} = 0,5$ ).

$$\sum_{i=0}^{S^2} \cdot \sum_{j=0}^B 1_{ij}^{obj} (C_i - C'_i)^2 + \lambda_{no} \sum_{i=0}^{S^2} \cdot \sum_{j=0}^B 1_{ij}^{no} (C_i - C'_i)^2 . \quad (4)$$

Детальную эволюция модели от 1-й до 11-й версии приводить не будем, лишь отметив основные изменения некоторых из них.

**YOLO v3** [23] балансирует скорость и точность, обеспечивая около 33 % отображения на СОСО при скорости примерно 30 кадров в секунду.

**YOLO v4** дополнительно оптимизирует сетевую архитектуру, достигая примерно 43 % отображения на СОСО при аналогичной производительности в реальном времени, со скоростью в реальном времени ~ 65 кадров в секунду на Tesla V100 [24].

**YOLO v8** предлагает передовую производительность в плане точности и скорости [25]. Основываясь на достижениях предыдущих версий, YOLOv8 представила новые функции и оптимизации, которые позволяют решать различные задачи обнаружения объектов в широком спектре приложений. Модель была разработана путем интеграции модуля блока остаточной-расширенной повторной параметризации с расширением, обобщенной сети пирамид признаков и функции потерь. Усовершенствованная модель была обучена и оценена на комплексном наборе данных, достигнув точности – 81,43 %, полноты – 68,48 % и значений mAP50 – 81,68 %.

**YOLO v11** – новейшая модель YOLO [26], обеспечивающая самую высокую производительность (SOTA) в различных задачах, включая обнаружение объектов, сегментацию, оценку позы, отслеживание и классификацию, использует возможности различных приложений и областей ИИ.

Сравнивая последние версии модели, следует отметить, что YOLOv9 демонстрирует существенную точность, но испытывает трудности с обнаружением мелких объектов и эффективностью, тогда как YOLOv10 демонстрирует относительно более низкую точность из-за архитектурных решений, которые влияют на его производительность при обнаружении перекрывающихся объектов, но превосходит их по скорости и эффективности. YOLOv11 неизменно демонстрирует высокую производительность с точки зрения точности, скорости, вычислительной эффективности и размера модели. Модель YOLOv11m достигла замечательного баланса точности и эффективности, набрав баллы mAP50-95 0,795, 0,81 и 0,325 для наборов данных Traffic Signs, African Wildlife и Ships соответственно, при этом сохраняя среднее время вывода 2,4 мс, размер модели 38,8 Мб и около 67,6 GFLOP в среднем [27].

В исследовании [28] удалось достичь высоких скоростей обработки изображения. YOLOv10 менее точен, чем YOLO11 [27], но превзошел по скорости и эффективности обработки. Несмотря на то, что алгоритмы демонстрируют многообещающую производительность, все еще есть возможности для совершенствования.

Высокая скорость и высокая точность делают семейство YOLO перспективным в решении задач БПЛА.

## 2. Сравнительная оценка

В таблице приведен сравнительный обзор нескольких алгоритмов на основе нейросетей, включая детекторы объектов и алгоритмы отслеживания, с приблизительными показателями производительности и примечательными особенностями. Значения mAP (средняя точность) и FPS (кадров в секунду) зависят от наборов данных (VOC или СОСО для детекторов) и аппаратных настроек, используемых в соответствующих исследованиях. Для алгоритмов отслеживания, таких как SiamFC, SiamRPN, ConvLSTM, и подходов, использующих корреляционную фильтрацию с самообучением, показатель mAP напрямую неприменим. Для алгоритма SiamFC используется другая метрика – отслеживание.

Сравнение некоторых нейросетевых алгоритмов

пп	Алгоритм	mAP	FPS, к/с	Особенность и применимость к поиску возгораний
1	<b>R-CNN</b>	~58 % (VOC)	< 1	Использует выборочный поиск для генерации предложений по регионам и применяет CNN к каждому варианту из них независимо. Точный, но затратный с точки зрения вычислений. Применяется при ограниченных площадях наблюдения с короткими временами задержек.
2	<b>Fast R-CNN</b>	~66 % (VOC)	~5	Вводит RoI (области интереса) для совместного использования функций свертки в предложениях, сокращая избыточные вычисления и повышая скорость по сравнению с R-CNN. Применяется при средних площадях мониторинга с выделением областей возгорания.
3	<b>Faster R-CNN</b>	~70 % (VOC)	~7	Включает сеть региональных предложений (RPN), сочетая высокую точность с повышенной скоростью. Применяется при средних площадях мониторинга с выделением областей возгорания.
4	<b>Mask R-CNN</b>	~70 % (VOC) (обнаруж.)	~5	Расширяет R-CNN, добавляя ветвь сегментации, которая предоставляет маски объектов на уровне пикселей наряду с ограничивающими рамками. Применяется при средних площадях мониторинга с выделением областей с последующей сегментацией и классификацией.
5	<b>YOLO v8</b>	~64–74 %	~80–200+	Популярная модель семейства YOLO, оптимизированная как для повышения точности обнаружения, так и для быстрого вывода в реальном времени на современном оборудовании. Поставляется в нескольких размерах, которые обеспечивают компромисс между скоростью и точностью. Позволяет определять местоположение и тип возгорания с детализацией.
6	<b>YOLO v11</b>	~76 % (COCO, вариат.)	~100–290*	Последняя в серии YOLO, основанная на разработках своих предшественников, особенно YOLOv8. Предлагает пять масштабированных моделей от нано до сверхбольших. Включает в себя многочисленные приложения, такие как обнаружение объектов, сегментация экземпляров, классификация изображений, оценка позы и ориентированное обнаружение объектов. Позволяет определить местоположение и тип возгорания с детализацией.
7	<b>SiamFC</b>	N/A *	~80–90	Разработан для визуального отслеживания путем изучения метрики сходства между целевым шаблоном и регионами-кандидатами; отличается скоростью и простой работы в реальном времени. Позволяет определять фронт горения.
8	<b>SiamRPN</b>	N/A*	~150–200	Улучшает SiamFC за счет интеграции механизма предложения региона для более точной локализации цели; широко используется для приложений высокоскоростного отслеживания. Позволяет определять фронт горения.
9	<b>ConvLSTM (for Sequential Detection)</b>	+~10 %	~25–30 (прибл.)	Включает сверточные слои LSTM для захвата как пространственных, так и временных зависимостей в последовательных данных, что приводит к повышению точности обнаружения в видеопотоках по сравнению с независимой обработкой кадров. Позволяет проводить некоторую семантическую классификацию возгораний
10	<b>DL with Correlation Filtering and Self-Supervised Learning</b>	+~5–10 %	~60	Объединяет традиционные методы корреляционной фильтрации с глубоким изучением функций; самообучение сводит к минимуму зависимость от помеченных данных, улучшая пространственное сопоставление и повышая устойчивость к изменениям внешнего вида. Сочетание интерпретации последовательности выявления и классификации возгораний.

### 3. Обсуждение

Выбор алгоритмов часто предполагает компромисс между скоростью и точностью. Двухступенчатые детекторы, такие как Faster R-CNN и Mask R-CNN, обеспечивают высокую точность обнаружения, но обычно работают медленнее, что может ограничить их использование в задачах реального времени.

Одноступенчатые детекторы, такие как YOLO, обеспечивают высокую скорость вывода, что делает их идеальными для БПЛА, требующих быстрого принятия решений, а вариативность размеров моделей внутри семейства делает их практически идеальным выбором.

Алгоритмы ConvLSTM и Сиамские нейросети вводят дополнительные измерения – временную согласованность и надежное отслеживание, которые имеют решающее значение для изменяющейся обстановки, в которой работают РТС.

Интеграция корреляционной фильтрации с самообучением ещё больше расширяет возможности систем обнаружения, особенно в сценариях с ограниченным количеством помеченных данных.

Представленные в таблице особенности позволяют выбирать нейросетевые алгоритмы под имеющиеся задачи мониторинга возгораний и имеющиеся вычислительные ресурсы.

### Выводы

В статье рассмотрены нейросетевые алгоритмы, возможные для применения на борту РТС для обнаружения возгораний при чрезвычайных ситуациях, приводится сравнение методов обнаружения и излагаются рекомендации по выбору соответствующего подхода с учетом операционных ограничений и поставленных задач.

Выбор подходящего алгоритма зависит от поставленных задач и имеющихся вычислительных ресурсов, например, в задачах, когда точность важнее скорости, следует применять двухступенчатые детекторы на основе региональных сверточных нейросетей, таких как Faster R-CNN. Для работы в реальном времени предпочтительнее одноступенчатые детекторы, такие как YOLO, которые обладают значительными преимуществами в скорости, что делает их подходящими для сценариев с быстро движущимися РТС. В задачах отслеживания подходы ConvLSTM и Сиамские сети повышают производительность видеопоследовательностей за счет использования временной памяти и надежного сопоставления по сходству. В задачах по адаптивности в условиях нехватки данных подходят методы, сочетающие глубокое обучение с корреляционной фильтрацией и обучением под контролем, которые предоставляют надежные альтернативы, когда количество помеченных данных ограничено. Гибридный подход, использующий сильные стороны нескольких алгоритмов, может предложить наилучшее решение для сложных задач, которые требуется решить на борту РТС.

Перспективным направлением развития рассмотренных алгоритмов является создание специализированных полигонов для апробации и тестирования обнаружения возгораний.

### Литература

1. Chitraranjan, C. Vision-Based Collision Warning Systems with Deep Learning: A Systematic Review / C. Chitraranjan, V. Vipulanathan, T. Sritharan // *J Imaging*. – 2025. – Vol. 11, Iss. 2. – P. 64.
2. CNN-Based Target Recognition and Identification for Infrared Imaging in Defense Systems / A. d’Acremont, R. Fablet, A. Baussard, G. Quin // *Sensors*. – 2019. – Vol. 19, Iss. 9. – P. 2040.
3. Autonomous Obstacle Avoidance and Target Tracking of UAV: Transformer for Observation Sequence in Reinforcement Learning / W. Jiang, T. Cai, G. Xu, Y. Wang // *Knowledge-Based Systems*. – 2024. – Vol. 290. – 111604.
4. Enhanced Deep Learning Model for Apple Detection, Localization, and Counting in Complex Orchards for Robotic Arm-Based Harvesting / T. Jin, X. Han, P. Wang *et al.* // *Smart Agricultural Technology*. – 2025. – Vol. 10. – 100784.
5. Fukushima, K. Neocognitron: a Self Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position / K. Fukushima // *Biological Cybernetics*. – 1980. – Vol. 36. – P. 193–202.
6. Convolutional Neural Networks: an Overview and Application in Radiology / R. Yamashita, M. Nishio, R.K.G. Do, K. Togashi // *Insights into Imaging*. – 2018. – Vol. 9. – P. 611–629.



7. End-to-End Pedestrian Collision Warning System based on a Convolutional Neural Network with Semantic Segmentation / H. Jung, M.K. Choi, K. Soon, W.Y. Jung // Proc. 2018 IEEE International Conference on Consumer Electronics (ICCE); Las Vegas, NV, USA, 12–14 January 2018, – P. 1–3.
8. Predicting Vehicle Collisions using Data Collected from Video Games / H. Kim, K. Lee, G. Hwang, C. Suh // Mach. Vis. Appl. – 2021. – Vol. 32. – Article no. 93.
9. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation / R. Girshick, J. Donahue, T. Darrell, J. Malik // CVPR '14 Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition. – 2014. – P. 580–587.
10. Zitnick, C.L. Edge Boxes: Locating Object Proposals from Edges / C.L. Zitnick, P. Dollar // Computer Vision-ECCV. – 2014. – P. 391–405.
11. Girshick, R. Fast R-CNN / R. Girshick // 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015. – P. 1440–1448.
12. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks / S. Ren, K. He, R. Girshick, J. Sun // 2015 – <https://arxiv.org/abs/1506.01497> (дата обращения 12.03.2025)
13. Mask R-CNN / K. He, G. Gkioxari, P. Dollár, R. Girshick // 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017. – 2017. – P. 2980–2988.
14. Convolutional LSTM Network: a machine learning approach for precipitation nowcasting / X. Shi, Z. Chen, H. Wang *et al.* // Proc. 29th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'15), Vol. 1. MIT Press, Cambridge, MA, USA. – 2015 – P. 802–810
15. Strickland, M. Deep Predictive Models for Collision Risk Assessment in Autonomous Driving / M. Strickland, G. Fainekos, H.B. Amor // Proc. 2018 IEEE International Conference on Robotics and Automation (ICRA); Brisbane, Australia. 21–25 May 2018. – 2018. – P. 4685–4692.
16. Gal, Y. Uncertainty in Deep Learning / Y. Gal // 2016. – <http://106.54.215.74/2019/20190729-liuzy.pdf> (дата обращения 12.03.2025)
17. [https://github.com/hunwenpinghao/SSD\\_ConvLSTM\\_final](https://github.com/hunwenpinghao/SSD_ConvLSTM_final) (дата обращения 12.03.2025)
18. Nabavi, S.S. Future Semantic Segmentation with Convolutional LSTM / S.S. Nabavi, M. Rochan, Y. Wang // British Machine Vision Conference. – 2018. – <https://arxiv.labs.arxiv.org/html/1807.07946> (дата обращения 12.03.2025)
19. Self-Supervised Context-Aware Correlation Filter for Robust Landmark Tracking in Liver Ultrasound Sequences / L. Ma, J. Wang, S. Gong *et al.* // Biomedical Engineering / Biomedizinische Technik. – 2024. – Vol. 69, Iss. 4. – P. 383–394.
20. Fully-Convolutional Siamese Networks for Object Tracking / L. Bertinetto, J. Valmadre, J.F. Henriques *et al.* // Computer Vision – ECCV 2016 Workshops. ECCV 2016. Lecture Notes in Computer Science. – Springer, Cham, 2016. – Vol. 9914.
21. High Performance Visual Tracking with Siamese Region Proposal Network / B. Li, J. Yan, W. Wu *et al.* // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA. – 2018. – P. 8971–8980.
22. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon, S. Divvala, R. Girshick, A. Farhadi // 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA. – 2016. – P. 779–788.
23. Redmon, J. YOLOv3: An Incremental Improvement / J. Redmon, A. Farhadi // 2018. – <https://pjreddie.com/media/files/papers/YOLOv3.pdf> (дата обращения 10.03.2025)
24. Bochkovskiy, A. YOLOv4: Optimal Speed and Accuracy of Object Detection / A. Bochkovskiy, C.-Y. Wang, H.-Y. Liao // arXiv: Computer Vision and Pattern Recognition. – 2020 – <https://arxiv.org/abs/2004.10934> (дата обращения 11.03.2025).
25. Jocher, G. Ultralytics YOLOv8, 8.0.0 / G. Jocher, A. Chaurasia, J. Qiu // 2023 – <https://github.com/ultralytics/ultralytics> (дата обращения 11.03.2025)
26. <https://docs.ultralytics.com/models/yolo11/> (дата обращения 13.03.2025)
27. Evaluating the Evolution of YOLO (You Only Look Once) Models: A Comprehensive Benchmark Study of YOLO11 and Its Predecessors / N. Jegham, C.Y. Koh, M. Abdelatti, A. Hendawi // 2024. – <https://arxiv.org/html/2411.00201v1>.
28. Al Rabbani, A.M. YOLOv11 for Vehicle Detection: Advancements, Performance, and Applications in Intelligent Transportation Systems. / A.M. Al Rabbani // 2024. – <https://arxiv.org/html/2410.22898v1>

*Поступила в редакцию 18 марта 2025 г.*

### Сведения об авторах

Зорин Василий Александрович – соискатель, Институт проблем управления им. В.А. Трапезникова Российской академии наук, г. Москва, Российская Федерация, ORCID iD: <https://orcid.org/0000-0002-7308-8387>, e-mail: [v.a.zorin@mail.ru](mailto:v.a.zorin@mail.ru).

Мещеряков Роман Валерьевич – доктор технических наук, профессор РАН, директор Центра интеллектуальных робототехнических систем, Институт проблем управления им. В.А. Трапезникова Российской академии наук, г. Москва, Российская Федерация, ORCID iD: <https://orcid.org/0000-0002-1129-8434>, e-mail: [mrv@ipu.ru](mailto:mrv@ipu.ru).

---

*Bulletin of the South Ural State University  
Series "Mathematics. Mechanics. Physics"  
2025, vol. 17, no. 2, pp. 23–34*

---

DOI: 10.14529/mmph250203

## ANALYTICAL REVIEW OF NEURAL NETWORK ALGORITHMS FOR FIRE DETECTION IN EMERGENCY SITUATIONS

**V.A. Zorin, R.V. Meshcheryakov**

*V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences, Moscow,  
Russian Federation  
E-mail: [mrv@ipu.ru](mailto:mrv@ipu.ru)*

**Abstract.** Advances in neural networks have enabled unmanned aerial vehicles (UAVs) to detect and recognize objects in real time, which has facilitated the use of UAVs autonomously in a variety of scenarios, including fire detection in emergency situations. The paper reviews a number of existing neural network-based detection algorithms, including convolutional neural networks, regional convolutional neural networks and their variants, deep neural networks with convolutional long short-term memory (ConvLSTM), methods integrating deep learning with correlation filtering through self-training, Siamese neural networks for target tracking, and the YOLO (You Only Look Once) family of algorithms. The main characteristics and differences between neural network algorithms are described, and a comparison of their performance in terms of mean average precision (mAP) and frame rate per second (FPS) is given. The conclusions of the article provide insight into the trade-offs between accuracy, speed and task-specific requirements in detection tasks, which allows one to make an informed choice on the use of one or another algorithm.

*Keywords:* Neural network algorithms; UAVs; detection; convolutional neural networks; YOLO.

### References

1. Chitraranjan C., Vipulanathan V., Sritharan T. Vision-Based Collision Warning Systems with Deep Learning: A Systematic Review. *J Imaging*, 2025, Vol. 11, Iss. 2, p. 64. DOI: 10.3390/jimaging11020064
2. d'Acremont A., Fablet R., Baussard A., Quin G. CNN-Based Target Recognition and Identification for Infrared Imaging in Defense Systems. *Sensors*, 2019, Vol. 19, Iss. 9, p. 2040. DOI: 10.3390/s19092040
3. Jiang W., Cai T., Xu G., Wang Y. Autonomous Obstacle Avoidance and Target Tracking of UAV: Transformer for Observation Sequence in Reinforcement Learning. *Knowledge-Based Systems*, 2024, Vol. 290, 111604. DOI: 10.1016/j.knosys.2024.111604
4. Jin T., Han X., Wang P., Zhang Zh., Guo J., Ding F. Enhanced Deep Learning Model for Apple Detection, Localization, and Counting in Complex Orchards for Robotic Arm-Based Harvesting. *Smart Agricultural Technology*, 2025, Vol. 10, 100784. DOI: 10.1016/j.atech.2025.100784.
5. Fukushima K. Neocognitron: a Self Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. *Biological Cybernetics*, 1980, Vol. 36, pp. 193–202. DOI: 10.1007/BF00344251

6. Yamashita R., Nishio M., Do R.K.G., Togashi K. Convolutional Neural Networks: an Overview and Application in Radiology. *Insights into Imaging*, 2018, Vol. 9, pp. 611–629. DOI: 10.1007/s13244-018-0639-9
7. Jung H., Choi M.K., Soon K., Jung W.Y. End-to-End Pedestrian Collision Warning System based on a Convolutional Neural Network with Semantic Segmentation. *Proc. 2018 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 12–14 January 2018, pp. 1–3.
8. Kim H., Lee K., Hwang G., Suh C. Predicting Vehicle Collisions using Data Collected from Video Games. *Mach. Vis. Appl.*, 2021, Vol. 32, Article no. 93. DOI: 10.1007/s00138-021-01217-2
9. Girshick R., Donahue J., Darrell T., Malik J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *CVPR '14 Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.
10. Zitnick C.L., Dollar P. Edge Boxes: Locating Object Proposals from Edges. *Computer Vision-ECCV*, 2014, pp. 391–405.
11. Girshick R. Fast R-CNN. *Proc. 2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, pp. 1440–1448, DOI: 10.1109/ICCV.2015.169
12. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. 2015. <https://arxiv.org/abs/1506.01497> (12.03.2025)
13. He K., Gkioxari G., Dollár P., Girshick R. Mask R-CNN. *Proc. 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 2980–2988, DOI: 10.1109/ICCV.2017.322
14. Shi X., Chen Z., Wang H., Yeung D.-Y., Wong W.-K., Woo W.-C. Convolutional LSTM Network: a Machine Learning Approach for Precipitation Nowcasting. *Proc. 29th International Conference on Neural Information Processing Systems – Vol. 1 (NIPS'15)*, Vol. 1, MIT Press, Cambridge, MA, USA, 2015, pp. 802–810.
15. Strickland M., Fainekos G., Amor H.B. Deep Predictive Models for Collision Risk Assessment in Autonomous Driving. *Proc. 2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia, May 21–25, 2018, pp. 4685–4692. DOI: 10.1109/icra.2018.8461160
16. Gal Y. *Uncertainty in Deep Learning*. 2016. <http://106.54.215.74/2019/20190729-liuzy.pdf> (12.03.2025)
17. [https://github.com/hunwenpinghao/SSD\\_ConvLSTM\\_final](https://github.com/hunwenpinghao/SSD_ConvLSTM_final) (12.03.2025)
18. Nabavi S.S., Rochan M., Wang Y. Future Semantic Segmentation with Convolutional LSTM. *Proc. British Machine Vision Conference*, 2018. <https://arxiv.labs.arxiv.org/html/1807.07946> (12.03.2025)
19. Ma L., Wang J., Gong S., Lan L., Geng L., Wang S., Feng X. Self-Supervised Context-Aware Correlation Filter for Robust Landmark Tracking in Liver Ultrasound Sequences. *Biomedical Engineering / Biomedizinische Technik*, 2024, Vol. 69, Iss. 4, pp. 383–394. DOI: 10.1515/bmt-2022-0489
20. Bertinetto L., Valmadre J., Henriques J.F., Vedaldi A., Torr P.H.S. Fully-Convolutional Siamese Networks for Object Tracking. In: Hua, G., Jégou, H. (eds) *Computer Vision – ECCV 2016 Workshops. ECCV 2016. Lecture Notes in Computer Science*, 2016, Vol. 9914. Springer, Cham. DOI: 10.1007/978-3-319-48881-3\_56
21. Li B., Yan J., Wu W., Zhu Z., Hu X. High Performance Visual Tracking with Siamese Region Proposal Network. *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 8971–8980. DOI: 10.1109/CVPR.2018.00935
22. Redmon J., Divvala S., Girshick R., Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91
23. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement. 2018, <https://pjreddie.com/media/files/papers/YOLOv3.pdf> (10.03.2025)
24. Bochkovskiy A., Wang C.-Y., Liao H.-Y. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv: Computer Vision and Pattern Recognition*, 2020, <https://arxiv.org/abs/2004.10934> (дата обращения 11.03.2025)
25. Jocher G., Chaurasia A., Qiu J. Ultralytics YOLOv8, 8.0.0, 2023, <https://github.com/ultralytics/ultralytics> (11.03.2025)
26. <https://docs.ultralytics.com/models/yolo11/> (13.03.2025)

27. Jegham N., Koh C.Y., Abdelatti M., Hendawi A. (2024). Evaluating the Evolution of YOLO (You Only Look Once) Models: A Comprehensive Benchmark Study of YOLO11 and Its Predecessors. <https://arxiv.org/html/2411.00201v1>

28. Al Rabbani A.M. YOLOv11 for Vehicle Detection: Advancements, Performance, and Applications in Intelligent Transportation Systems. 2024, <https://arxiv.org/html/2410.22898v1>

*Received March 18, 2025*

### **Information about the authors**

Zorin Vasilij Aleksandrovich is Researcher, V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences, Moscow, Russian Federation, ORCID iD: <https://orcid.org/0000-0002-7308-8387>, e-mail: [v.a.zorin@mail.ru](mailto:v.a.zorin@mail.ru).

Meshcheryakov Roman Valer'evich is Dr. Sc. (Engineering), Professor of the Russian Academy of Sciences, Director of the Center for Intelligent Robotic Systems, V.A. Trapeznikov Institute of Control Sciences of the Russian Academy of Sciences, Moscow, Russian Federation, ORCID iD: <https://orcid.org/0000-0002-1129-8434>, e-mail: [mrv@ipu.ru](mailto:mrv@ipu.ru).