

УДК 004.93'1

НЕЙРОСЕТЕВАЯ СИСТЕМА ОТСЛЕЖИВАНИЯ И РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ

Ляшов М.В., Береза А.Н., Бабаев А.М., Коцюбинская С.А.

*Институт сферы обслуживания и предпринимательства (филиал) ДГТУ,
Шахты, e-mail: maxl85@mail.ru*

Данная статья посвящена разработанной нейросетевой системе отслеживания объектов в видеопотоке. Предлагаемый алгоритм позволяет в режиме реального времени определять местоположение целевого объекта в текущем кадре видеопотока. Для отслеживания системе достаточно предоставить текущий и предыдущий кадры видео, а также положение целевого объекта на предыдущем кадре. Система предварительно не обучена для отслеживания определенного объекта – целевой объект определяется при запуске системы. Для повышения точности и надежности системы были использованы как методы отслеживания объектов, так и методы распознавания образов. Оценка смещения объекта от кадра к кадру была реализована на основе метода Лукаса – Канаде и медианного фильтра. Распознавание объекта осуществляется с помощью сверточной нейронной сети, в частности, архитектуры MobileNet. В работе приведены результаты экспериментальных исследований на основе четырех видеофайлов. Итоги экспериментов говорят о том, что предлагаемая комбинация технологий распознавания и отслеживания объектов позволяет решить проблему потери целевого объекта при его исчезновении или частичном перекрытии другим объектом. Кроме того, разработанный алгоритм инвариантен по отношению к форме объекта и изменению освещенности среды.

Ключевые слова: распознавание образов, отслеживание объектов, компьютерное зрение, оптический поток, сверточные нейронные сети, OpenCV, Tensorflow

OBJECT TRACKING AND RECOGNITION SYSTEM BASED ON NEURAL NETWORK IN A VIDEO STREAM

Lyashov M.V., Bereza A.N., Babaev A.M., Kotsyubinskaya S.A.

*Institute of Service and Business (branch) Don State Technical University,
Shakhty, e-mail: maxl85@mail.ru*

This article is devoted to the development of a neural network tracking system of objects in a video stream. The proposed algorithm allows to define a location of a target object in the current shot of a video stream in real time mode. It is enough for the tracking to provide system with current and previous shots of the video and also a location of the target object on the previous shot. The system is not previously trained for tracking a certain object – a target object is determined at launch of the system. To increase the correctness and the reliability of the system both the methods of object's tracking and methods of pattern recognition were used. Object shift assessment was realized on the Lucas–Kanade method and the median filter. Object recognition is performed using a convolutional neural network, in particular, by the MobileNet architecture. The paper presents the results of experimental studies based on four video files. The results of the experiments show that the proposed combination of the technologies of object recognition and tracking allows solving the problem of loss of the target at object disappearance or partial overlapping by another object. Moreover, the developed algorithm is invariant in relation to the form of an object and the change of the intensity of environmental illumination.

Keywords: object recognition, object tracking, computer vision, optical flow, convolutional neural network, OpenCV, Tensorflow

В эпоху больших данных бурный рост переживает область компьютерного зрения. Компьютерное зрение позволяет решать различные задачи, одной из которых является отслеживание движущихся объектов в видеопотоке.

Главными требованиями к системам отслеживания объектов в видеопотоке являются: высокая точность обнаружения перемещений объекта и возможность обработки последовательности кадров в режиме реального времени (более 10 кадров в секунду).

В данной работе представлена интеллектуальная система отслеживания произвольных объектов в видеопотоке. Разработанная система не является предварительно обученной для отслеживания каких-либо конкретных объектов, а позволяет определять целевой объект при запуске.

Цель исследования заключалась в разработке алгоритма отслеживания объектов в видеопотоке, инвариантного по отношению к свойствам объекта (цвет, форма и т.д.) и среды (например, освещение). Также в работе алгоритма учтены возможные потери отслеживаемого объекта вследствие его выхода из поля зрения камеры или перекрытия другим объектом. В этом случае отслеживаемый объект может быть повторно обнаружен автоматически.

Современные методы отслеживания объектов в видеопотоке. Наиболее популярными методами отслеживания объектов в видеопотоке являются алгоритмы:

1. Online Boosting Tracker [1] – один из первых алгоритмов отслеживания объектов, имеющий высокую точность отслеживания. Основан на онлайн-версии алгоритма

AdaBoost, использующего для поиска объекта каскад Хаара.

2. MIL Tracker (Multiple Instance Learning Tracker) [2] является развитием идеи обучения с учителем и алгоритма Online Boosting Tracker. Основное отличие заключается в том, что в данном случае в качестве положительного примера рассматривается не только текущее местоположение объекта, но и другие области, содержащие его.

3. KCF Tracker (Kernelized Correlation Filters Tracker) [3] – KCF основывается на идеях двух предыдущих алгоритмов, однако за счет использования математических свойств перекрывающихся областей положительных примеров происходит значительное увеличение быстродействия при более простой реализации.

Недостатком описанных выше методов является неспособность повторного обнаружения объекта в случае его потери.

Предлагаемое решение. Разработанный алгоритм обеспечивает возможность отслеживания передвижений объекта в видеопотоке при его потере из-за перекрытия другим объектом или исчезновения из поля зрения камеры. Расположение объекта на изображении задается ограничительным окном – это прямоугольная область, содержащая фрагмент изображения с отслеживаемым объектом.

В качестве входной информации алгоритму необходимо предоставить текущий кадр видеопотока и ограничительное окно на предыдущем кадре. На выходе алгоритма будут получены координаты ограничительного окна, соответствующего положению объекта на текущем кадре видеопотока.

Блок-схема интеллектуальной системы отслеживания объектов в видеопотоке представлена на рис. 1.

Система включает четыре компонента:

1. Алгоритм отслеживания (трекер) позволяет оценивать смещение ограни-

тельного окна от предыдущего кадра к текущему кадру. Он адаптивен к небольшим изменениям освещения и внешнего вида объекта. При этом предполагается, что данный компонент может ошибаться в случае частичного или полного перекрытия отслеживаемого объекта. Исходя из этого требуется дополнительный компонент, корректирующий работу алгоритма отслеживания.

2. Алгоритм распознавания (детектор) является корректирующим компонентом системы. Его работа не зависит от перекрытия или исчезновения объекта. Детектор обнаруживает объект при его появлении в кадре и при необходимости корректирует или повторно инициализирует трекер, работая с ним параллельно. Для повышения точности детектора после каждого кадра необходимо обновлять его знания об отслеживаемом объекте и об объектах, расположенных на фоне.

3. Объединение результатов предоставляет обобщенный вариант ограничительного окна, полученный на основе результатов детектора и трекера. Если ни трекер, ни детектор не выдают ограничительных окон, то считается, что объект потерян. В противном случае для каждого из ответов на основе НСС вычисляется параметр, который определяет величину подобия ограничительного окна и положительных примеров обучающей выборки (описано ниже). Система возвращает то ограничительное окно, которое обладает наибольшим значением этого параметра.

4. Алгоритм обучения – это компонент, который выполняет обучение детектора для достижения более высокой точности. На каждой итерации алгоритма обучение анализирует выход трекера и детектора, оценивает ошибки, совершаемые детектором, и обновляет детектор, чтобы избежать этих ошибок в будущем.

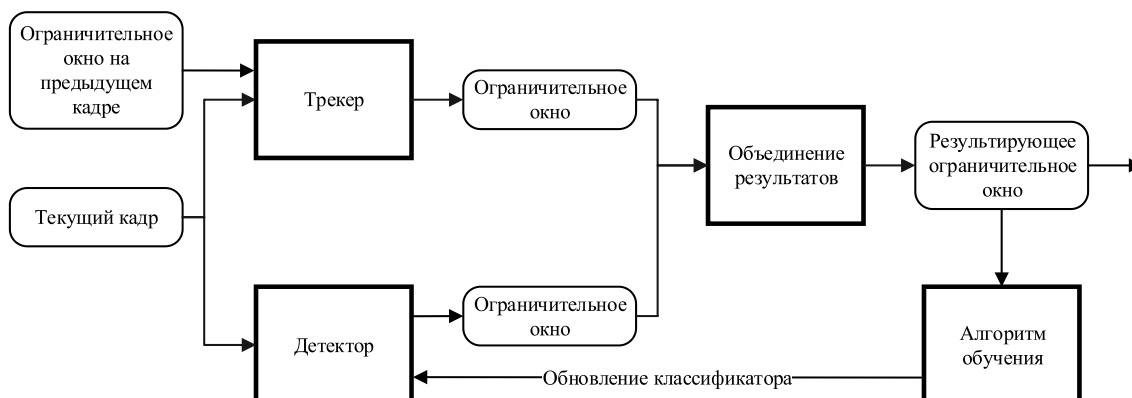


Рис. 1. Блок-схема системы

Для оценки смещения объекта между кадрами были протестированы следующие алгоритмы: фильтр Калмана, алгоритм сдвига среднего (Mean-Shift) [4], алгоритм непрерывно адаптирующегося сдвига (CamShift) [5], алгоритм Лукаса – Канаде (Lucas-Kanade) [6]. При анализе особое внимание уделялось вычислительной сложности алгоритмов, а также тому, насколько эффективно они работают в условиях меняющегося размера, больших сдвигов и частичного перекрытия объекта. По результатам тестирования выбор был сделан в пользу метода Лукаса – Канаде в его пирамидальном варианте, который является компромиссом в отношении быстродействия и надежности.

При использовании метода Лукаса – Канаде в большинстве случаев имеются ошибочно найденные точки. Их количество зависит от величины смещения и изменения условий среды между кадрами. Для отсеи-

вания ошибочных точек может быть применена двухуровневая фильтрация, использованная в алгоритме медианного потока (Median-Flow tracker) [7].

На первом уровне выполняется фильтрация на основе нормализованной корреляции (Normalized-Cross Correlation, далее – NCC), которая является одним из компонентов технологии сопоставления шаблона. Сопоставление шаблона позволяет обнаружить местоположение некоторого изображения-шаблона T с размерностью $w \times h$ на исходном изображении I с размерностью $W \times H$. Для этого T путем вычисления NCC последовательно сравнивается с каждым фрагментом I , имеющим размерность $w \times h$. На выходе будет получено изображение в градациях серого R с размерностью $(W - w + 1) \times (H - h + 1)$, где яркость каждого пикселя $R(x, y)$ указывает на сходство его окрестности с шаблоном. Значение яркости $R(x, y)$ определяется следующей метрикой:

$$R(x, y) = \frac{\sum_{x', y'} [T(x', y') - \bar{T}_{x, y}] * [I(x + x', y + y') - \bar{I}_{x, y}]}{\sqrt{\sum_{x', y'} [T(x', y') - \bar{T}_{x, y}]^2 * \sum_{x', y'} [I(x + x', y + y') - \bar{I}_{x, y}]^2}},$$

где $\bar{T}_{x, y}$ – среднее значение яркости для шаблона;

$\bar{I}_{x, y}$ – среднее значение яркости для текущего фрагмента исходного изображения.

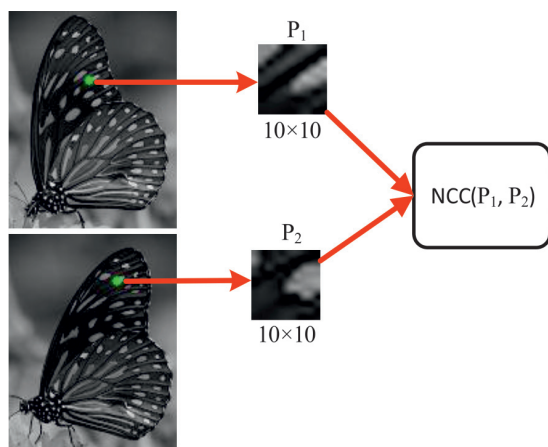


Рис. 2. Сравнение окрестностей ключевых точек путем вычисления NCC

Для фильтрации точек отслеживания используется частный случай NCC, когда размерности T и I совпадают. Для всех точек на предыдущем и следующем кадрах формируются изображения размером 10×10 пик-

селей с центром в ключевой точке. Далее полученные изображения для соответствующих точек попарно сравниваются посредством вычисления NCC. Описанный процесс проиллюстрирован на рис. 2. После вычисления NCC для всех пар точек смежных кадров выполняется следующая фильтрация точек:

1. Находится медианное значение последовательности значений NCC для всех пар точек.

2. Точки, для которых значение нормализованной корреляции оказалось меньше медианного значения, будут отброшены.

Итоговая схема одной итерации алгоритма отслеживания движения объектов приведена на рис. 3.

Распознавание объектов с помощью свёрточной нейронной сети. Для выполнения распознавания объектов в видеопотоке выбор был сделан в пользу современных архитектур свёрточных нейронных сетей. СНС позволяют распознавать изображения с точностью, значительно превосходящей точность других методов классификации изображений [8, 9].

В ходе выполнения работы были проанализированы наиболее популярные современные архитектуры свёрточных нейронных сетей, применяемые в задачах классификации изображений [10–12] (табл. 1).

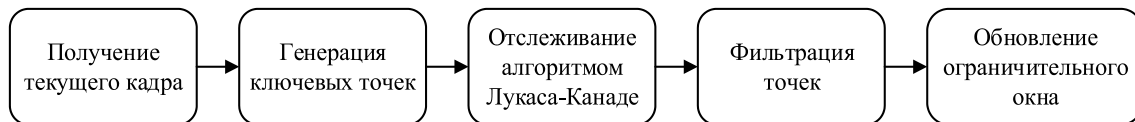


Рис. 3. Одна итерация алгоритма отслеживания движения объектов

Таблица 1

Популярные архитектуры свёрточных нейронных сетей

	AlexNet	VGG Net	Google GoogLeNet	ResNet	MobileNet
Год	2012	2014	2014	2015	2017
Кол-во слоев	8	16	22	152	102
Кол-во параметров, млн	60	138	5	60	5
Точность, %	84,7	92,7	93,33	96,4	87,3
FPS	5	7	19	11	до 450

Таблица 2

Описание тестовых видеофайлов

Название	Количество кадров	Движение камеры	Перекрытие	Полное исчезновение	Изменение размера объекта
Jumping	313	Да	Нет	Нет	Нет
Box	1161	Нет	Да	Нет	Нет
Motocross	2665	Да	Да	Да	Да
Car Chase	9928	Да	Да	Да	Да

Одним из последних решений в сфере распознавания образов является семейство сетей MobileNets [12], которые были предложены компанией Google в 2017 г. Они предназначены для решения задач компьютерного зрения на мобильных устройствах, то есть устройствах с малым объемом памяти и вычислительных ресурсов. MobileNet проигрывают представителям семейства сетей ResNet и Inception в плане точности классификации. Однако MobileNet имеет малые задержки (частота обработки кадров видеопотока может достигать 450 кадров в секунду) при обработке одного изображения и небольшие требования к ресурсам устройства, вследствие чего данная архитектура была выбрана в качестве основы построения детектора в разрабатываемой системе отслеживания объектов.

Экспериментальные исследования. Для реализации системы был выбран язык программирования Python а также библиотеки OpenCV и TensorFlow. В качестве аппаратной платформы для построения и тестирования разрабатываемой системы был выбран персональный компьютер

с графической картой GeForce GTX 1080 Ti и установленным программным обеспечением CUDA и cuDNN. Это позволило достичь значительного ускорения работы нейронной сети MobileNet и получить систему, работающую в режиме реального времени.

Для проведения испытаний использовались четыре видеофайла: Jumping, Box, Motocross и Car Chase (табл. 2) [7].

Для оценки качества работы предлагаемого алгоритма было проведено его сравнение с алгоритмами Online Boosting Track, MIL Tracker и KCF Tracker. Использовались следующие критерии:

1. Скорость работы алгоритма, измеряемая как кадровая частота в количестве кадров, обработанных за секунду (Frames per Second, далее FPS).

2. Точность, вычисляемая на основе количества успешно распознанных кадров, когда ограничительное окно покрывает более 50% объекта. Кадры, на которых отслеживаемый объект не присутствовал, не учитывались.

Результаты тестирования приведены на рис. 4.

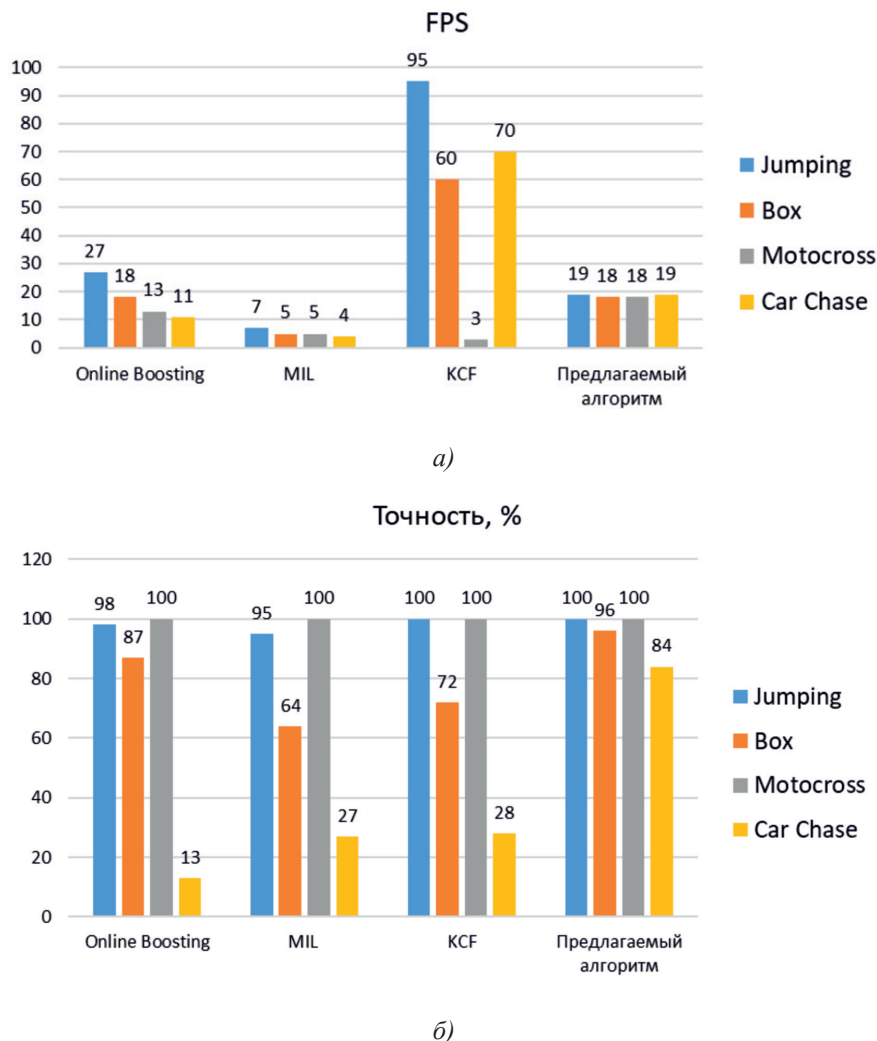


Рис. 4. Результаты сравнения предлагаемого алгоритма отслеживания объектов в видеопотоке и современных методов: а) количество кадров в секунду (FPS); б) точность отслеживания объекта

По результатам исследований можно сделать вывод, что предложенный в данной работе алгоритм не уступает аналогам в плане точности отслеживания объекта, а в некоторых случаях и превосходит их. Это во многом объясняется тем, что разработанная технология отслеживания объектов включает в себя детектор, который позволяет повторно инициализировать трекер в случае потери объекта. Кроме того, алгоритм показал стабильный показатель скорости обработки видеопотока, который в незначительной мере зависит от размера отслеживаемого объекта.

Выводы

В данной работе была описана система реального времени для отслеживания объекта в видеопотоке. Система показала высокие

результаты распознавания одного объекта на кадрах видеофайлов при использовании графических карт. Дальнейшая работа будет направлена на повышение точности и надежности работы системы, а также на увеличение скорости обработки видеопотока и обеспечения возможности одновременного отслеживания нескольких объектов.

Работа выполняется при поддержке РФФИ (проекты № 17-07-01323, № 18-07-01054).

Список литературы

1. Grabner H., Grabner M., Bischof H. Real-time Tracking via On-line Boosting. In Proceedings British Machine Vision Conference (BMVC). 2006. vol. 1. P. 47–56.
2. Babenko B. Visual Tracking with Online Multiple Instance Learning. Computer Vision and Pattern Recognition, CVPR 2009. IEEE Conference. 2009. P. 101–108.

3. Henriques, J. F. High-Speed Tracking with Kernelized Correlation Filters. *IEEE transactions on pattern analysis and machine intelligence*. 2015. P. 301–314.
4. Dorin Comaniciu and Peter Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. *IEEE Trans. Intell.* 2002. P. 603–619.
5. Shujun Yao, Xiaodong Chen, Sen Wang, Zhihai Jiao, Yi Wang. Camshift algorithm -based moving target recognition and tracking system. *IEEE International Conference on Virtual Environments Human-Computer Interfaces and Measurement Systems (VECIMS) Proceedings*. 2012. P. 513–529.
6. Fleet D.J., Weiss Y. Optical flow estimation. *Handbook of Mathematical Models in Computer Vision*. 2005. P. 237–257.
7. Kalal Z., Mikolajczyk K., Matas J. Forward-Backward Error: Automatic Detection of Tracking Failures. *International Conference on Pattern Recognition In International Conference on Pattern Recognition*. 2010. P. 23–26.
8. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. The MIT Press, 2016. 800 p.
9. Simonyan K. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR 2014*. P. 587–601.
10. Szegedy C. Going deeper with convolutions. *Computer Vision and Pattern Recognition (CVPR), IEEE Conference*. 2015. P. 488–502.
11. He K. Deep Residual Learning for Image Recognition. *Computer Vision and Pattern Recognition (CVPR), IEEE Conference*. 2016. P. 770–778.
12. Howard A.G. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv*. 2017. URL: <https://arxiv.org/abs/1704.04861> (дата обращения: 17.10.2018).