

XV Международная научно-практическая конференция студентов аспирантов и молодых учёных  
«Молодёжь и современные информационные технологии»

## РАЗРАБОТКА АЛГОРИТМА СЛЕЖЕНИЯ ЗА РАСПОЗНАННЫМИ ПО ВИДЕОПОТОКУ ОБЪЕКТАМИ

А.А. Волков, А.А. Войцеховский

А.Ю. Зарницын, Б.И. Пякилля

Томский политехнический университет  
aav38@tpu.ru

### Аннотация

Одной из задач при анализе видеопотока является задача слежения за объектом. В случае слежения за объектом выбирается нужный объект путём его выделения. Далее задача алгоритма состоит в верном выделении объекта на следующих кадрах объекта.

Существующие алгоритмы располагают только апостериорной информацией, полученной из выделенного оператором объекта, и имеют ряд недостатков, обусловленных отсутствием знаний об объекте слежения. Это в первую очередь связано с постановкой задачи.

Суть данной работы заключается в доработке одного из существующих алгоритмов слежения, решающего задачу слежения наилучшим образом, путём добавления к нему модуля знаний об объекте слежения. Модуль знаний будет представлять собой искусственную нейронную сеть, которая будет обучена классифицировать объекты на кадрах видеопотока и корректировать работу алгоритма слежения.

Для отработки данной идеи была использована библиотека компьютерного зрения OpenCV, в которой уже реализовано большинство необходимых функций.

Данную программу возможно доработать для решения задачи управления мобильным роботом путём установки взаимосвязи между относительным положением объекта и дальнейшей выработки алгоритма слежения.

### Способы слежения, их преимущества и недостатки.

Отыскивать объект на видео возможно двумя способами: распознаванием и слежением. В случае распознавания, программе известно, как выглядит объект слежения, и она последовательно проверяет части изображения с целью найти похожие объекты. Недостатком такого подхода является невозможность следить за объектом в случае его частичного или полного перекрытия, а также если внешний вид объекта слежения сильно меняется.

Для слежения необходимо в начальный момент времени выделить объект и дальше он будет программа будет отслеживать объект оценивая оптический поток, который характеризует относительное изменение положение объектов на следующем кадре видео. Так как известно где находился объект в предыдущих кадрах, то известна скорость и направление движения объекта. Используя эти данные возможно с

большой точностью предсказать следующее положение объекта. В случае кратковременной потери объекта слежения из поля зрения объектива программа уже не сможет работать.

Анализ вышеназванных недостатков позволил выработать решение, заключающиеся в использовании комбинированного метода.

В начальный момент времени выделяется объект слежения и затем классификатор обучается по данному изображению объекта и по следующим, которые были получены алгоритмом слежения по относительному смещению объекта слежения. В итоге классификатор обучается на конкретном объекте слежения, что позволяет добиться лучшего распознавания в данных условиях. Данный способ требует постепенного изменения внешнего вида и не способен обрабатывать резкое изменение освещения.

### Существующие алгоритмы слежения

Большая часть алгоритмов слежения реализована в библиотеке компьютерного зрения OpenCV, которая имеет большой набор функций для работы с изображениями и видеопотоком. Для данного исследования были выбраны те алгоритмы, которые имеют наибольшую точность и скорость работы. Ниже приведена краткая характеристика каждого из алгоритмов, а также их достоинства и недостатки.

#### 1) MIL (Multiple Instance Learning).

Данный алгоритм использует текущее положение объекта как положительный пример для обучения классификатора, а также несколько частей изображения, которые по размеру равны объекту и находятся в небольшой окрестности рядом с текущим положением объекта, как потенциально положительные примеры. Таким образом, если текущее положение объекта слежения неточно, то существует шанс того, что положительный пример с верным текущим положением объекта находится в наборе потенциально положительных [1]. Устойчиво работает при частичном перекрытии объекта, но при кратковременном полном перекрытии алгоритм теряет объект слежения.

#### 2) TLD (Tracking, Learning, Detection).

Исходя из названия можно определить, что данный алгоритм декомпозирует долговременное слежение в три компоненты: кратковременное слежение, обучение и распознавание. Модуль слежения покадрово сопровождает объект. Модуль распознавания корректирует модуль слежения,

если это необходимо. Модуль обучения аккумулирует все изображения объекта и стремится снизить ошибку.

Среди достоинств стоит отметить работу в реальном масштабе времени, стабильность к долговременным перекрытиям объекта слежения и устойчивое определение изменения масштаба объекта слежения.

### 3) GOTURN.

Данный алгоритм основан на применении сверточной нейронной сети, которая уже обучена на большой выборке различных объектов. На каждом кадре нейронная сеть стремится найти выбранный в начальный момент времени объект. Причем объект слежения должен принадлежать к классам объектов, которые нейронная сеть способна распознать.

К достоинствам данного алгоритма стоит отнести верность определения потерю объекта. К недостаткам нужно отнести низкую устойчивость слежения в случаях заслонения объекта.

### **Описание и принцип работы улучшенного алгоритма**

Оценив существующие алгоритмы слежения, было решено выбрать в качестве базового алгоритм TLD.

Для классификации объекта, а также определения его положения на кадре было решено использовать сверточную нейронную сеть. Данный вид нейросетей позволяет получить наилучшие результаты при распознании изображений [4]. Кроме того, сверточная нейронная сеть имеет меньше настраиваемых параметров, чем аналогичные по точности архитектуры. Это позволяет снизить риски переобучения, а также ускорить процесс обучения.

Нейронная сеть имеет следующую архитектуру:

1. Входной слой размерности 100x100x3;
2. Слой свертки с 32 фильтрами, ядром 3x3 и функцией активации ReLU;
3. Слой объединения (pooling) с ядром 3x3;
4. Слой свертки с 64 фильтрами, ядром 3x3 и функцией активации ReLU;
5. Слой объединения с ядром 3x3;
6. Слой свертки с 96 фильтрами, ядром 3x3 и функцией активации ReLU;
7. Слой объединения с ядром 3x3;

Далее идут пять ветвей с полносвязными сетями:

- Ветвь классификации – один полносвязный слой из 256 нейронов с функцией активации ReLU и слой из трех нейронов с функцией softmax;
- Четыре ветви координат – в каждой два полносвязных слоя из 256 нейронов с функцией активации ReLU и выходной слой из одного нейрона с линейной функцией активации.

На вход сети подается изображение с тремя

цветовыми каналами (RGB), которое предварительно подвергается процедуре изменения размера до 100x100 пикселей. На выходе у сети четыре параметра: класс объекта, координаты верхнего левого угла, ширина и высота захватывающей рамки (bounding box).

При обучении нейронной сети для уменьшения переобучения был выбран метод dropout [5]. Его суть заключается в случайном обнулении заданного процента от всех весов слоя в процессе обучения. Эта часть нейронов не обучается в момент своей неактивности, следовательно, сеть не обучается на случайных шумах в данных.

Архитектура выбрана эмпирическим путем, основываясь на существующих архитектурах сверточных нейронных сетей для решения подобных задач.

Принцип работы алгоритма следующий. Начальный кадр видео обрабатывается нейронной сетью, которая классифицирует объекты на кадре и определяет положение объекта слежения. Далее эта информация поступает на алгоритм TLD, который осуществляет слежение за объектом. Через небольшие интервалы времени сравниваются положения объекта, полученные при помощи алгоритма TLD и нейронной сети. Если различие превышает порог, то алгоритму TLD указывается новое положение объекта слежения, которое было определено нейронной сетью.

### **Вывод**

В ходе данной работы был разработан алгоритм слежения за объектами двух классов: теннисные мячи и бумажные стаканчики. Его точность и устойчивость работы оказались выше, чем у имеющихся в библиотеке OpenCV.

### **Список использованных источников**

1. Boris Babenko, Ming-Hsuan Yang, Serge Belongie, "Robust Object Tracking with Online Multiple Instance Learning", *IEEE TPAMI*, 2011
2. Zdenek Kalal, Krystian Mikolajczyk, Jiri Matas, "Tracking Learning Detection", *IEEE Transactions on pattern analysis and machine intelligence*, vol. 6, no. 1, 2010
3. David Held, Sebastian Thrun, Silvio Savarese, "Learning to Track at 100 FPS with Deep Regression Networks", *European Conference on Computer Vision (ECCV)*, 2016
4. Nitish Srivastava "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", *Journal of Machine Learning Research*, no. 15, 2014
5. Yann LeCun, Yoshua Bengio "Convolution Networks for Images, Speech, and Time-Series", *The handbook of brain theory and neural networks*, pp 255-258.