

ОТСЛЕЖИВАНИЕ ОБЪЕКТОВ НА ВИДЕОЗАПИСИ ПРИ ПОМОЩИ МЕТОДОВ ГЛУБОКОГО ОБУЧЕНИЯ

Сидоренко К.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Калугина М.А. – канд. физ.-мат. наук, доцент

В докладе рассматриваются два алгоритма отслеживания объектов на видеозаписи, основанные на методах глубокого обучения. Описываются принципы реализации алгоритмов, производится их сравнение, выделяются достоинства и недостатки каждого алгоритма.

Задача отслеживания объектов на видео изучается и развивается уже не первый десяток лет. Необходимость в отслеживании объектов возникает, например, для решения следующих задач: отслеживание людей в толпе для обеспечения безопасности при проведении массовых мероприятий, учет количества посетителей в различных заведениях для подсчета важных статистик для бизнеса, отслеживание перемещения и поведения скота в фермерских хозяйствах, трекинг автомобилей в системах автоматического определения правонарушений и идентификации нарушителей. Несмотря на то, что в большинстве своем современные алгоритмы трекинга объектов обучены отслеживать преимущественно людей, транспортные средства и животных, можно обучить модель отслеживанию практически любых других объектов [1].

Необходимо понимать различия между задачами обнаружения (детектирования) и отслеживания объектов. В то время как алгоритмы обнаружения объектов лишь определяют цель на фото или видео, алгоритмы отслеживания связывают информацию, полученную с предыдущих кадров таким образом, чтобы не терять объект, а также при необходимости предсказывать траекторию его движения. Так, при использовании методов детектирования объектов для задачи трекинга, модель будет всякий раз терять цель, когда объект на некоторое время пропадает из видимости (например, автомобиль проезжает через мост). Однако, алгоритмы отслеживания включают в себя задачу определения объектов в каждом кадре, поэтому этап детектирования играет очень важную роль.

В работе рассматриваются два популярных алгоритма трекинга – SORT [2] и DeepSORT [3]. DeepSORT по своей сути является модификацией алгоритма SORT. SORT включает в себя следующие ключевые концепты:

1. *Обнаружение объекта.* На данном этапе происходит определение объектов в текущем кадре. В результате получаем положение и размер прямоугольника (ограничивающая рамка), который содержит в себе обнаруженный объект. Надо учитывать, что в задаче отслеживания объектов на видеозаписи критическую роль играет скорость работы алгоритмов, поэтому на каждой стадии необходимо выбирать наиболее быстродействующие алгоритмы. В качестве алгоритма детектирования будем использовать YOLO [4]. Данный алгоритм отличается высокой скоростью работы (последние версии алгоритма обрабатывают изображения со скоростью 70 кадров в секунду), что позволяет применять его и для видеозаписей.

2. *Оценка модели.* Задача данного этапа заключается в создании представления и модели движения объекта, используемых для передачи цели в следующий кадр. Здесь мы аппроксимируем межкадровые перемещения каждого объекта с помощью линейной модели постоянной скорости.

3. *Связывание данных.* На этом этапе выполняется связывание обнаруженных объектов в текущем кадре с уже захваченными целями. Для ограничивающей рамки каждой захваченной цели выбирается наиболее близкая ограничивающая рамка обнаруженного объекта в текущем кадре.

4. *Создание и удаление уникального идентификатора объекта отслеживания.* Когда в кадре появляется новый, ранее не обнаруженный объект, ему присваивается уникальный идентификатор, и объект начинает отслеживаться. Когда отслеживаемый объект пропадает из кадра, его отслеживание заканчивается. Однако, если тот самый объект вновь попадает в кадр, ему присвоится новый идентификатор. Данный факт может привести к некоторым неудобствам, например к ошибочному расчету статистики.

Обозначенную проблему решает алгоритм DeepSORT. Алгоритм реализует все описанные концепции алгоритма SORT, но дополнительно использует нейронную сеть для реидентификации объектов [5]. Такая сеть создает так называемый хеш-код для каждого отслеживаемого объекта, по которому его можно заново распознать. Этот хеш-код запоминается и в случае, если объект покидает зону видимости, а через некоторое время возвращается в кадр, он рассматривается как уже ранее отслеживаемый объект со старым идентификатором. Чтобы реидентификационная сеть работала хорошо, ее обучают по уникальным признакам объектов. Например, для людей это может быть одежда или лицо, если оно отчетливо видно на видео, для машин – марка автомобиля. В то же время, нейронная сеть реидентификации требует тонкой настройки и качественного обучения, что влечет за собой дополнительные расходы времени и вычислительных мощностей. Для сравнения,

для отслеживания определенных объектов в алгоритме SORT достаточно обучить нейронную сеть обнаружения объектов.

В качестве демонстрации и сравнения работы двух алгоритмов проверим их на одной и той же тестовой видеозаписи. На короткой (30 секунд) видеозаписи пешеход движется по улице и на одно мгновение скрывается за кадр. Проследим, как алгоритмы справятся с поставленной задачей. Сперва запустим алгоритм SORT.



Рисунок 1 – Пример работы алгоритма SORT

На рисунке 1 видно, что после того, как пешеход на некоторое время пропадает из обзора камеры, алгоритм SORT определяет этого пешехода как два разных, что можно понять по отличию цвета рамок. Теперь запустим алгоритм DeepSORT на той же видеозаписи.



Рисунок 2 – Пример работы алгоритма DeepSORT

Как видно из рисунка 2, DeepSORT хорошо справился с проблемой кратковременного исчезновения пешехода из кадра. Однако стоит отметить, что по времени работы DeepSORT несколько уступает своему «предку»: алгоритм SORT обработал видео на 5 секунд быстрее.

В итоге проделанной работы были изучены два алгоритма трекинга объектов – SORT и DeepSORT. Оба алгоритма обладают своим набором преимуществ, поэтому выбор конкретного алгоритма зависит от поставленной задачи.

Список использованных источников:

1. How to train to detect your custom objects [Электронный ресурс]. – Режим доступа : <https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects>.
2. Simple Online and Realtime Tracking / A. Bewley [et al.] // arXiv:1602.00763v2 [cs.CV], 2017.
3. Simple Online and Realtime Tracking with a Deep Association Metric / N. Wojke [et al.] // arXiv: 1703.07402v1 [cs.CV], 2017.
4. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon [et al.] // arXiv:1506.02640v5 [cs.CV], 2016.
5. Pedestrian Attribute Recognition: A Survey / X. Wang [et al.] // arXiv: 1901.07474v1 [cs.CV], 2019.