Национальный исследовательский
Нижегородский государственный университет им. Н.И. Лобачевского
Институт информационных технологий, математики и механики

Образовательный курс
«Современные методы и технологии глубокого обучения в компьютерном зрении»

Глубокие модели для сопровождения объектов на видео

При поддержке компании Intel

Васильев Евгений

Содержание

- Цель лекции
- Постановка задачи сопровождения объектов
- Открытые наборы данных
- Показатели качества сопровождения объектов
- Глубокие модели для сопровождения объектов на видео
- Заключение

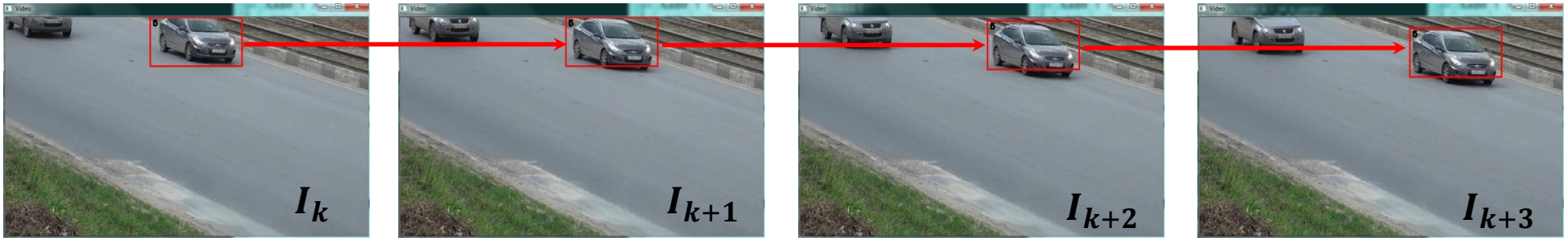
Цель лекции

- ***Цель*** – изучить глубокие нейросетевые модели для решения задачи сопровождения объектов

ПОСТАНОВКА ЗАДАЧИ СОПРОВОЖДЕНИЯ ОБЪЕКТОВ

Постановка задачи (1)

- Сопровождение объектов на видео предполагает начальный захват положения интересующего объекта (обычно в момент первого появления в кадре видео) и восстановление положений этого объекта на последующих кадрах видеопотока



Постановка задачи (2)

- $I_0, I_1, \dots, I_{N-1}$ – последовательность кадров, где N – количество кадров
- Для сопровождения объектов необходимо построить отображение ψ множества положений B_k на кадре I_k на множество положений B_{k+1} на кадре I_{k+1}:

$$\psi: B_k \to B_{k+1} \cup \{b\}, \qquad b = \big((-1,-1),(-1,-1)[,s,c]\big)$$

где b – фиктивное положение, используемое для обозначения ситуации потери объекта алгоритмом сопровождения

- Если I_k – первый кадр, на котором обнаружен объект, $r_0(k)$ – индекс положения, q – количество кадров, тогда ***траектория*** – последовательность положений

$$T^k_{r_0(k)} = \left(b^k_{r_0}, b^{k+1}_{r_1}, \dots, b^{k+q-1}_{r_{q-1}}\right), \qquad b^{k+i}_{r_i} = \psi\left(b^{k+i-1}_{r_{i-1}}\right), i = \overline{1, q-1}$$

ОТКРЫТЫЕ НАБОРЫ ДАННЫХ

Наборы данных (1)

Набор данных	Количество видео	Количество изображений	Количество траекторий
Сопровождение одного объекта (каждое видео содержит один объект)			
Long-Term Visual Object Tracking Benchmark [https://amoudgl.github.io/tlp]	50	676 000	50
TrackingNet [https://tracking-net.org]	30 643	14 220 000	30 643
VOT Challenge [http://www.votchallenge.net]	60	21 455	60

Наборы данных (2)

Набор данных	Количество видео	Количество изображений	Количество траекторий	Количество объектов в разметке
Сопровождение нескольких объектов				
Multiple Object Tracking Benchmark [https://motchallenge.net]	21	11 286	1 221	101 345
CityFlow [https://www.aicitychallenge.org]	40	~117 000	666	229 680

Наборы данных (3)

- Набор данных CityFlow содержит более 25 часов видео с 40 дорожных камер США, находящихся в одном городе, причем видео сняты в одно время и синхронизированы по времени
- Источником данных для TrackingNet являются видео с YouTube, поэтому данный набор содержит большое количество видео разного качества, в том числе, значительное число видео низкого разрешения
- VOT Challenge помимо RGB-изображений включает в себя изображения с камеры глубины и инфракрасной камеры
- Некоторые видео из набора данных VOT Challenge Benchmark содержат разметку окаймляющими прямоугольниками, стороны которых не параллельны осям координат

Multiple Object Tracking Benchmark (1)

- Multiple Object Tracking Challenge (MOT) – конкурс по сопровождению объектов на видеопоследовательности, проводится с 2015 года
- Официальная страница [https://motchallenge.net]
- Последняя версия в открытом доступе – MOT17 за 2017 год
  - 21 видео перемещения автомобилей и людей в городе
  - Суммарно 15 948 изображений и 1 638 траекторий в тренировочном наборе, 17 757 изображений и 2 355 траекторий в тестовом наборе
- Набор содержит видео как со статичных камер, так и видео от первого лица с перемещаемых камер

* Leal-Taixé L., Milan A., Reid I., Roth S., Schindler K. MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. – 2015. – [https://arxiv.org/pdf/1504.01942.pdf].

Multiple Object Tracking Benchmark (2)

Примеры изображений с траекториями и окаймляющими прямоугольниками

* Multiple Object Tracking Benchmark. MOT17 [https://motchallenge.net/data/MOT17].

** Leal-Taixé L., Milan A., Reid I., Roth S., Schindler K. MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. – 2015. – [https://arxiv.org/pdf/1504.01942.pdf].

Long-Term Visual Object Tracking Benchmark

- Long-Term Visual Object Tracking Benchmark – бенчмарк, содержащий набор длительных видео с одиночными объектами
- Официальная страница [https://amoudgl.github.io/tlp]
- Набор данных состоит из 50 видео с разным контекстом и с одиночными объектами для сопровождения
- Всего более 400 минут, 676 тыс. кадров

Примеры первых кадров видеопоследовательностей

* Moudgil A., Gandhi V. Long-Term Visual Object Tracking Benchmark. – 2019. – [https://arxiv.org/abs/1712.01358].

TrackingNet

- TrackingNet – большой набор данных, состоящий из видео с YouTube разного качества и разрешения
- Официальная страница [https://tracking-net.org]
- Более 30 тыс. видео, суммарно 140 часов
- Более 14 миллионов детектированных объектов

Примеры последовательностей кадров с сопровождаемым объектом

* Müller M., Bibi A., Giancola S., Al-Subaihi S., Ghanem B. TrackingNet: A Large-Scale Dataset and Benchmark for Object Tracking in the Wild. – 2018. – [https://arxiv.org/abs/1803.10794].

ПОКАЗАТЕЛИ КАЧЕСТВА СОПРОВОЖДЕНИЯ ОБЪЕКТОВ

Показатели качества

- Критерии выбора показателей качества сопровождения:
  - Отражать качество восстановления положения на каждом очередном кадре видео, содержащем сопровождаемый объект
  - Отражать качество сопровождения на протяжении всей последовательности кадров, содержащих объект
  - Для каждого сопровождаемого объекта построенная траектория должна быть единственной
  - Обеспечивать сопоставимость показателей для разных видов алгоритмов сопровождения (2D, 3D трекеров, трекеров центроидов, трекеров областей и др.)

Рассматриваемые показатели качества

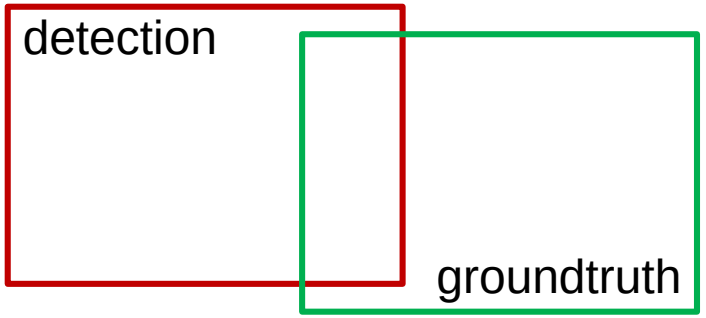
- Показатели качества сопровождения одиночных объектов:
  - Точность (Accuracy)
  - Надежность (Robustness)

- Показатели качества сопровождения нескольких объектов:
  - Достоверность сопровождения нескольких объектов (Multiple Object Tracking Accuracy, MOTA)
  - Точность сопровождения нескольких объектов (Multiple Object Tracking Precision, MOTP)

Точность

- ***Точность**** (Accuracy) – средняя доля перекрытия обнаруженных (detection) и размеченных (groundtruth) окаймляющих прямоугольников траектории по всем кадрам видео

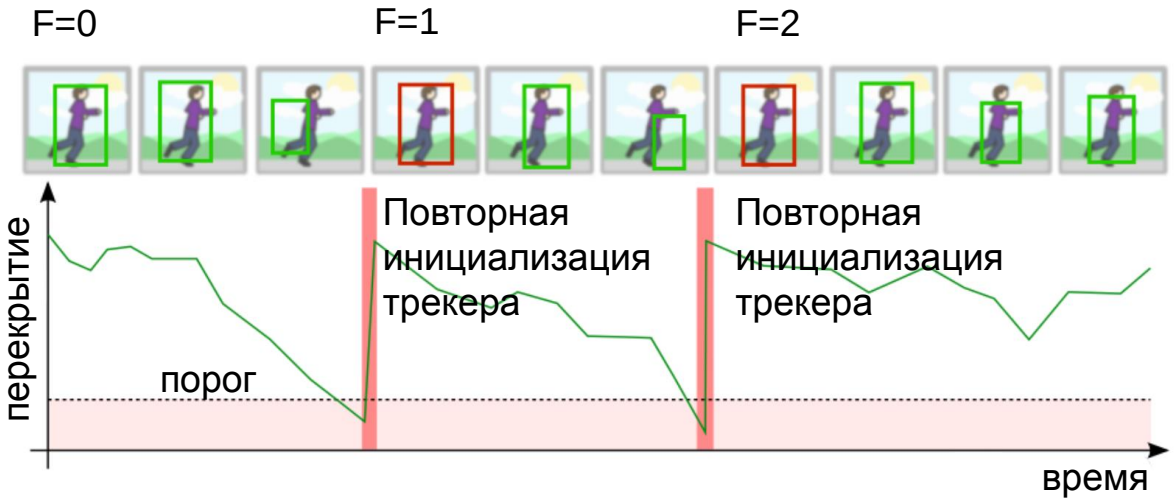
$$A = \frac{1}{N}\sum_{t=1}^{N}\frac{S_{d\cap g}}{S_{d\cup g}}$$



* Kristan M., et al. The Visual Object Tracking VOT2014:Challenge and results. – 2014. – [https://votchallenge.net/vot2014/download/vot_2014_presentation.pdf].

Надежность

- ***Надежность**** (Robustness) показывает, сколько раз трекер теряет сопровождаемый объект и должен быть повторно инициализирован
- Сопровождаемый объект считается потерянным, когда доля перекрытия обнаруженного и размеченного объектов траектории меньше порогового значения



* Kristan M., et al. The Visual Object Tracking VOT2014:Challenge and results. – 2014. – [https://votchallenge.net/vot2014/download/vot_2014_presentation.pdf].

Достоверность сопровождения нескольких объектов (1)

- ***Достоверность сопровождения нескольких объектов**** (Multiple Object Tracking Accuracy, MOTA)
- Один из наиболее известных показателей, который используется для сравнения алгоритмов сопровождения в разных конкурсах и бенчмарках
- Обозначения:
  - t – номер текущего кадра видео
  - $\{o_1, o_2, \dots, o_n\}$ – множество наблюдаемых (размеченных) объектов на кадре t
  - $\{h_1, h_2, \dots, h_m\}$ – множество положений, построенных в результате сопровождения, на кадре t

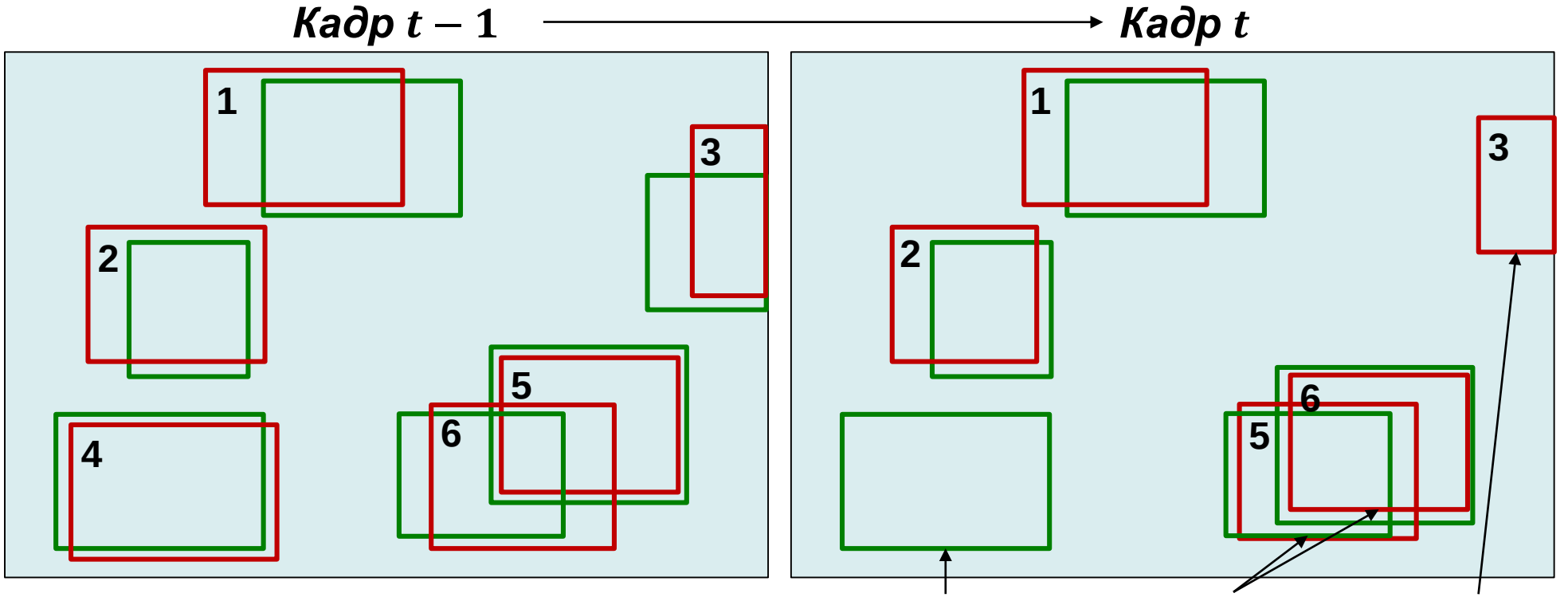
* Bernardin K., Stiefelhagen R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics // Image and Video Processing. – 2008.

Достоверность сопровождения нескольких объектов (2)

- ***Достоверность сопровождения нескольких объектов*** (Multiple Object Tracking Accuracy, MOTA)
- Общая схема вычислений для каждого кадра t:
  1. Вычисление ошибки построения положений объектов и поиск наилучшего соответствия построенных $\{h_1, h_2, \dots, h_m\}$ и размеченных $\{o_1, o_2, \dots, o_n\}$ положений
  2. Накопление ошибок:
     1. Подсчет размеченных объектов, для которых не нашлось соответствующих построенных положений, – ***пропусков (misses)***
     2. Подсчет построенных положений, для которых не нашлось реальных объектов, – ***ложные срабатывания (false positives)***
     3. Подсчет случаев, когда у объекта изменился идентификатор траектории, – ***ошибки несоответствия (mismatch errors)***

* Bernardin K., Stiefelhagen R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics // Image and Video Processing. – 2008.

Достоверность сопровождения нескольких объектов (3)



– размеченный объект

– сопровождаемый объект

m – **miss** (объект потерян)

mme – **mismatch error** (сменился идентификатор траектории)

fp – **false positive** (реальный объект вышел из кадра)

Достоверность сопровождения нескольких объектов (4)

- ***Достоверность сопровождения нескольких объектов**** (Multiple Object Tracking Accuracy, MOTA) определяется следующим образом:

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t},$$

где m_t – количество пропусков на кадре t,
fp_t – количество ложных срабатываний на кадре t,
mme_t – количество несоответствий на кадре t,
g_t – количество размеченных объектов на кадре t

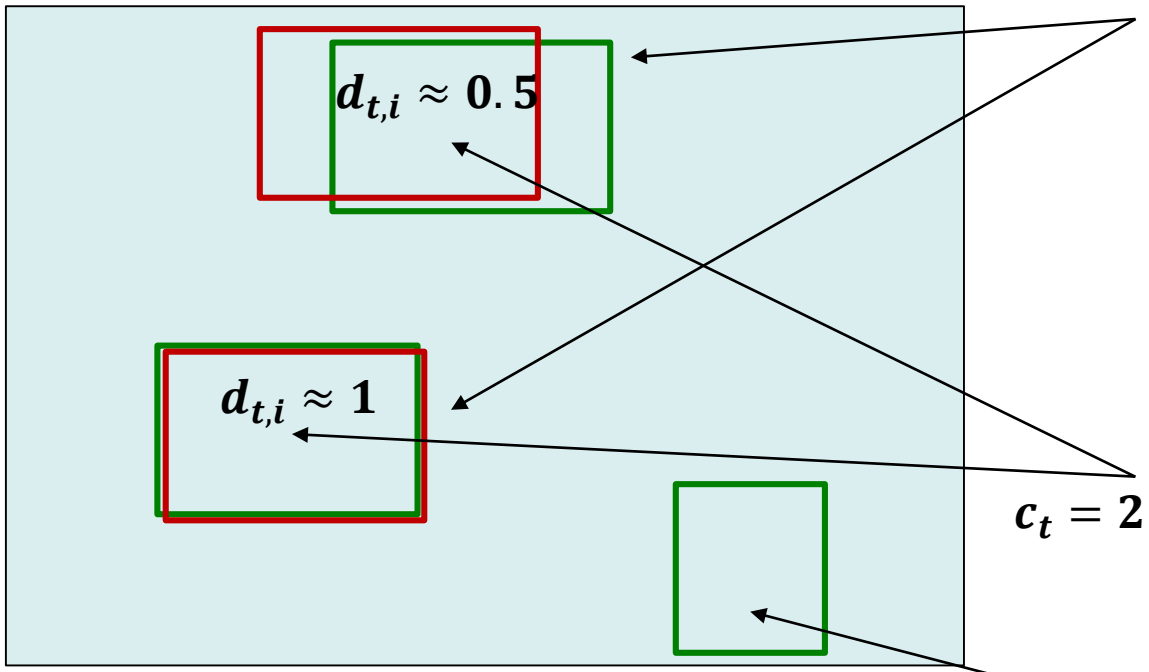
- ***Примечание:*** для поиска соответствий могут использоваться разные подходы (например, Венгерский алгоритм для матрицы соответствия построенных и размеченных положений)

* Bernardin K., Stiefelhagen R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics // Image and Video Processing. – 2008.

Точность сопровождения нескольких объектов (1)

- ***Точность сопровождения нескольких объектов*** (Multiple Object Tracking Precision, MOTP)
- Метрика отражает качество восстановления положений объектов в задаче сопровождения
- Обозначения:
  - $\{o_1, o_2, \dots, o_n\}$ – множество размеченных объектов на кадре t
  - $\{h_1, h_2, \dots, h_m\}$ – множество положений, построенных в результате сопровождения, на кадре t
  - $\{P_0^t, P_1^t, \dots, P_k^t\}$ – множество пар соответствий между размеченными и построенными положениями объектов в траекториях. Соответствия строятся с использованием Венгерского алгоритма

Точность сопровождения нескольких объектов (2)



$d_{t,i}$ – доля перекрытия окаймляющих прямоугольников из пары P_i^t

c_t – количество пар соответствий размеченных и построенных положений объектов

$c_t = 2$

Объект без пары, не участвующий в вычислении метрики

– размеченное положение

– построенное положение

Точность сопровождения нескольких объектов (3)

- ***Точность сопровождения нескольких объектов*** (Multiple Object Tracking Precision, MOTP) вычисляется следующим образом:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} = \frac{\sum_{t,i} \frac{o_i \cap h_i}{o_i \cup h_i}}{\sum_t c_t}$$

  - c_t – количество соответствий на кадре t
  - $d_{t,i}$ – доля перекрытия окаймляющих прямоугольников из пары P_i^t
  - $o_i \cap h_i$ – пересечение окаймляющих прямоугольников из пары P_i^t
  - $o_i \cup h_i$ – объединение окаймляющих прямоугольников из пары P_i^t

Точность сопровождения нескольких объектов (4)

- В некоторых бенчмарках в качестве $d_{t,i}$ в предыдущей формуле может использоваться не перекрытие окаймляющих прямоугольников, а Евклидово расстояние между центрами окаймляющих прямоугольников:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} = \frac{|Center(o_i) - Center(h_i)|}{\sum_t c_t}$$

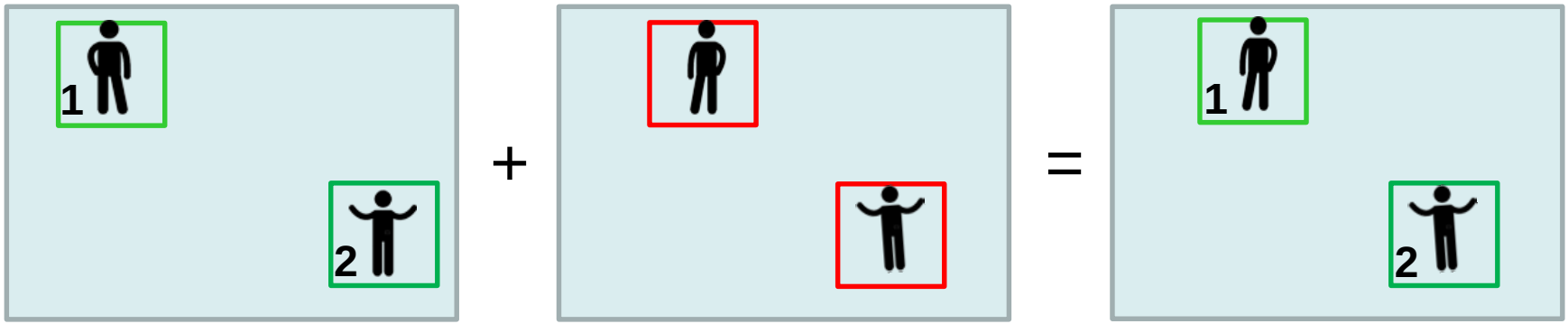
- ***Примечание:*** метрика не учитывает потери объекта в процессе сопровождения

* Bernardin K., Elbs A., Stiefelhagen R. Multiple Object Tracking Performance Metrics and Evaluation in a Smart Room Environment. – 2006. – [https://cvhci.anthropomatik.kit.edu/~stiefel/papers/ECCV2006WorkshopCameraReady.pdf].

ГЛУБОКИЕ МОДЕЛИ ДЛЯ СОПРОВОЖДЕНИЯ ОБЪЕКТОВ

Схема сопровождения нескольких объектов через сопоставление объектов (1)

- Построены траектории объектов на кадре t
- Обнаружение объектов на новом кадре (детектирование выполняется независимо от предыдущих кадров)
- Сопоставление траекторий на кадре t и объектов на кадре $t+1$, чтобы получить лучшее предположение об объектах на кадре $t+1$



Положения объектов в траекториях на кадре t

Обнаруженные объекты на кадре $t+1$

Положения объектов в траекториях на кадре $t+1$

Схема сопровождения нескольких объектов через сопоставление объектов (2)

- Для сопоставления траекторий и обнаруженных объектов строится матрица сходства
- Матрица сходства A для N траекторий и M объектов – это матрица размера $N \times M$, где каждый элемент a_{ij} представляет собой коэффициент сходства траектории T_i с объектом R_j
  - Коэффициент сходства a_{ij} вычисляется, например, на основе положения, размера и внешнего вида объекта в траектории и обнаруженного объекта
- Нахождение наилучших соответствий по матрице соответствий является задачей оптимизации, которая решается с помощью Венгерского алгоритма

Классификация методов и глубоких моделей сопровождения

- ***Классификация методов по доступу к данным:***
  - Офлайн-методы
  - Онлайн-методы

- ***Классификация глубоких моделей по способу их использования в схеме сопровождения через сопоставление объектов:***
  - Дополнение существующего алгоритма сопровождения
  - Встраивание глубоких моделей в алгоритм сопровождения (deep network embedding)
  - Полная замена алгоритма сопровождения на глубокие модели (end-to-end deep networks)

Классификация методов по доступу к данным

- ***Офлайн-методы сопровождения объектов*** имеют доступ ко всем кадрам видеопоследовательности одновременно, и строят траектории объектов для всего видео
- ***Онлайн-методы сопровождения объектов*** имеют последовательный доступ к кадрам, и строят предсказание траекторий для текущего кадра только на основании предыдущих кадров

- Интерес индустрии и исследователей обращен к онлайн-методам сопровождения, поскольку они позволяют получать необходимую информацию в реальном времени

Классификация глубоких моделей по способу их использования в схеме сопровождения

- ***Дополнение алгоритма сопровождения***
  - Построение глубокого описания объектов для последующего сопровождения (deep descriptions for objects)
- ***Встраивание глубоких моделей в алгоритм сопровождения*** (deep network embedding)
  - Глубокие модели заменяют некоторые этапы работы алгоритмов сопровождения (например, этап детектирования)
- ***Полная замена алгоритма сопровождения на глубокие модели*** (end-to-end deep networks)
  - Глубокие модели заменяют все этапы сопровождения объектов (детектирование объектов, поиск соответствий, добавление и удаление траекторий)

Рассматриваемые модели (1)

Дополнение алгоритма сопровождения

- ***DeepSORT (2017)***
  - Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric // International Conference on Image Processing. – 2017. – P. 3645–3649. – [https://arxiv.org/pdf/1703.07402.pdf], [https://ieeexplore.ieee.org/document/8296962] (опубликованная версия).

Рассматриваемые модели (2)

Встраивание глубоких моделей в алгоритм сопровождения (deep network embedding)

- ***SINT (2016)***
  - Tao R., Gavves E., Smeulders A. Siamese instance search for tracking. – 2016. – [https://arxiv.org/pdf/1605.05863.pdf], [https://ieeexplore.ieee.org/document/7780527] (опубликованная версия).
- ***SiameseNET (2017)***
  - Leal-Taixé L., Canton-Ferrer C., Schindler K. Learning by tracking: siamese CNN for robust target association. – 2016. – [https://arxiv.org/pdf/1604.07866.pdf], [https://ieeexplore.ieee.org/document/7789549] (опубликованная версия).
- ***GOTURN (2016)***
  - Held D., Thrun S., Savarese S. Learning to track at 100 FPS with deep regression networks. – 2016. – [https://arxiv.org/pdf/1604.01802.pdf].

Рассматриваемые модели (3)

Полная замена алгоритма сопровождения объектов на глубокую модель (end-to-end deep networks)

- ***RNN-LSTM (2017)***
  - Milan A., Rezatofighi S.H., Dick A.R., et al. Online multi-target tracking using recurrent neural networks. – 2017. – [https://arxiv.org/pdf/1604.03635.pdf], [https://dl.acm.org/doi/10.5555/3298023.3298181] (опубликованная версия).

DeepSORT (1)

- При сопоставлении траекторий и обнаруженных объектов в алгоритме сопровождения через нахождение соответствий необходимо вычислить степень сходства между описаниями траекторий и обнаруженных объектов
- Сравнение можно выполнять по следующим критериям:
  - По положению
  - По размеру
  - По внешнему виду

DeepSORT (2)

- Сравнение объектов по внешнему виду значительно повышает вероятность правильного сопровождения объектов в случае повторной идентификации объектов, которая требуется при пересечении и перекрытии объектов
- Изначально признаки для построения внешнего вида объекта конструировались вручную, но сейчас признаки извлекаются с использованием нейронных сетей
- Для построения описания объектов часто используются модели, основанные на широко известных классификационных нейронных сетях

DeepSORT (3)

- Идея метода DeepSORT состоит в том, чтобы с помощью глубокой модели получить дескриптор, который может описать все особенности изображения объекта
- Строится и обучается классификатор изображений на тренировочном наборе данных, а затем отбрасывается последний слой, отвечающий за классификацию
- Вместо построения и обучения модели с нуля могут использоваться существующие классификационные модели, от которых отбрасывается последний слой. Таким образом, полученная модель дает высокоуровневые признаки объекта

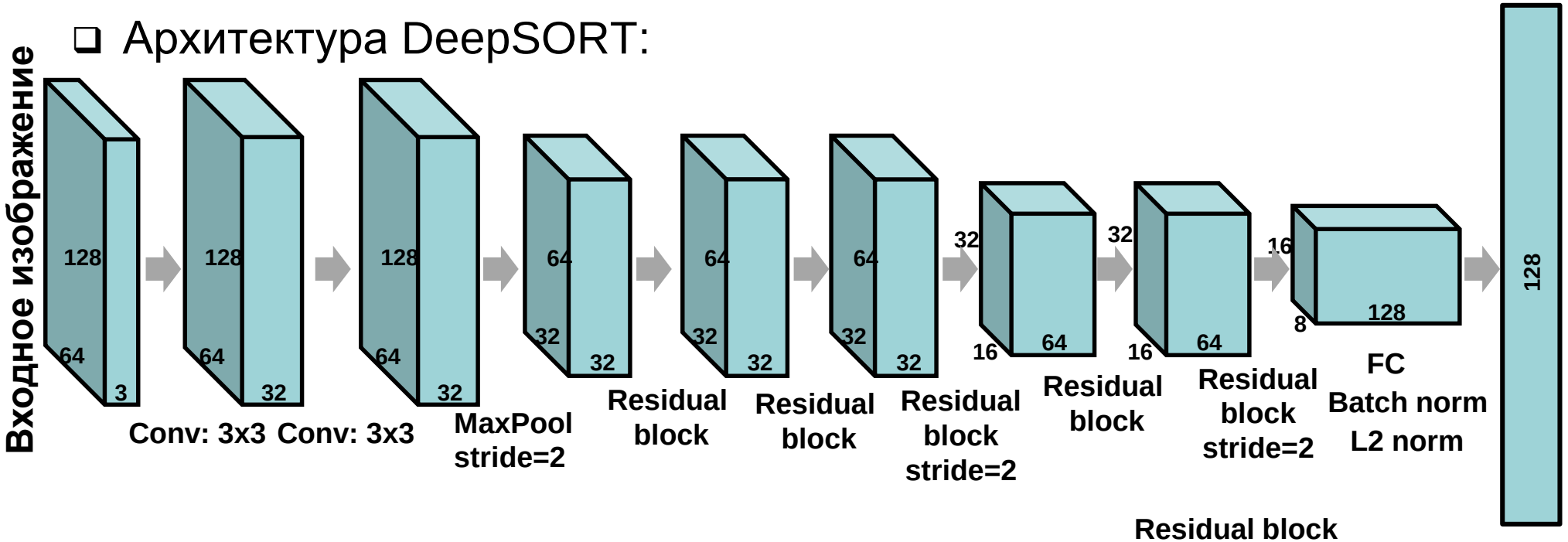
DeepSORT (4)

- DeepSORT (Deep Simple Online and Realtime Tracking) – модель, обеспечивающая построение ***дескриптора*** (описания детектируемого объекта) в виде вектора из 128 чисел
- Вход модели:
  - Изображение объекта разрешения 128x64 в формате RGB
- Выход модели:
  - Дескриптор объекта – тензор размера 1x128

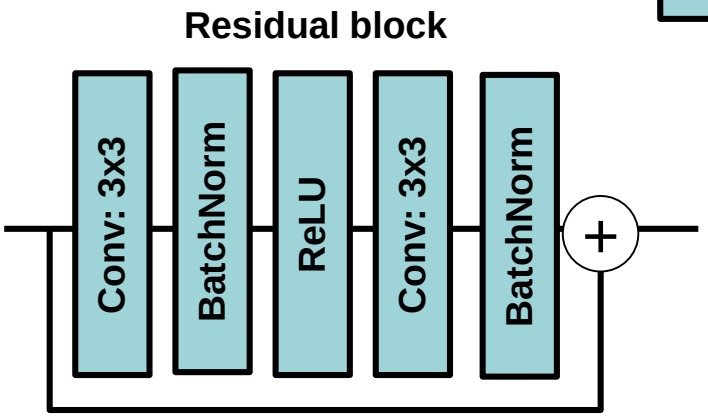
* Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric. – 2017 – [https://arxiv.org/pdf/1703.07402.pdf].

DeepSORT (5)

- Архитектура DeepSORT:



- Архитектура остаточных блоков:
  - В блоках с параметром ‘stride=2’ свертки выполняются с шагом 2



DeepSORT (6)

- Дескриптор объекта, полученный из исходного изображения с помощью модели DeepSORT, участвует в построении коэффициента сходства между траекторией и новым срабатыванием детектора, что позволяет повысить качество сопровождения объектов
- Коэффициент сходства внешнего вида D_a:
  - С помощью глубокой модели вычисляется дескриптор для изображения объекта из траектории a_i и обнаруженного объекта a_j
  - Коэффициент сходства вычисляется как косинусное расстояние между дескрипторами объектов a_i и a_j

$$D_a = cosine\_distance(a_i, a_j) = \frac{(a_i, a_j)}{\|a_i\| \cdot \|a_j\|}$$

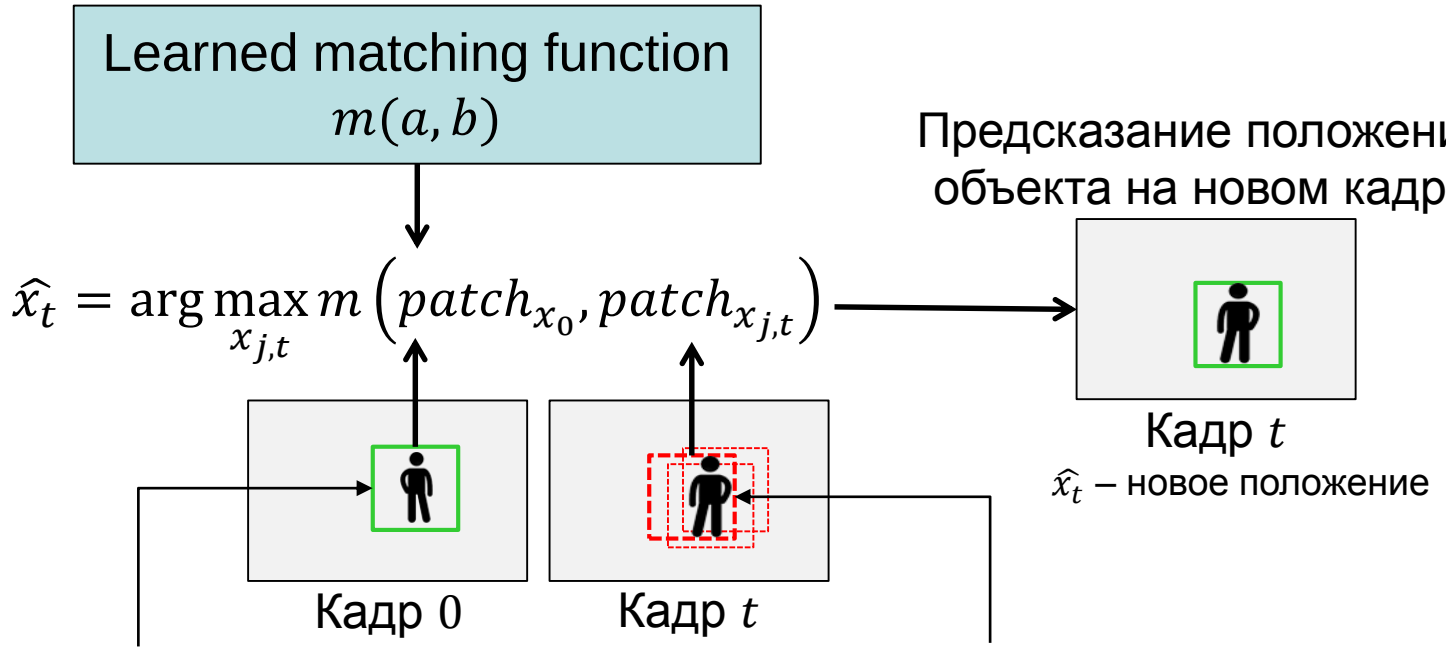
SINT (1)

- SINT (Siamese INstance search for Tracking) – метод, который получает на вход информацию о локализации объекта на первом кадре и гипотезы о положении объекта на текущем кадре и на основании этой информации возвращает предсказание положения объекта на текущем кадре
- Метод SINT не специализируется на каком-то классе объектов (пешеходы, автомобили и т.д.), метод рассчитан на произвольные категории

* Tao R., Gavves E., Smeulders A. Siamese instance search for tracking. – 2016. – [https://arxiv.org/pdf/1605.05863.pdf], [https://ieeexplore.ieee.org/document/7780527].

SINT (2)

- Цель SINT состоит в том, чтобы построить обучаемую функцию, обеспечивающую наилучшее сопоставление между первым кадром и гипотезами о положении объекта на текущем кадре



x_0 – окаймляющий прямоугольник для объекта
$patch_{x_0}$ – кадр + прямоугольник x_0

$x_{j,t}$ – гипотеза j для кадра t
$patch_{x_{j,t}}$ – кадр + прямоугольник $x_{j,t}$

* Tao R., Gavves E., Smeulders A. Siamese instance search for tracking. – 2016. – [https://arxiv.org/pdf/1605.05863.pdf], [https://ieeexplore.ieee.org/document/7780527].

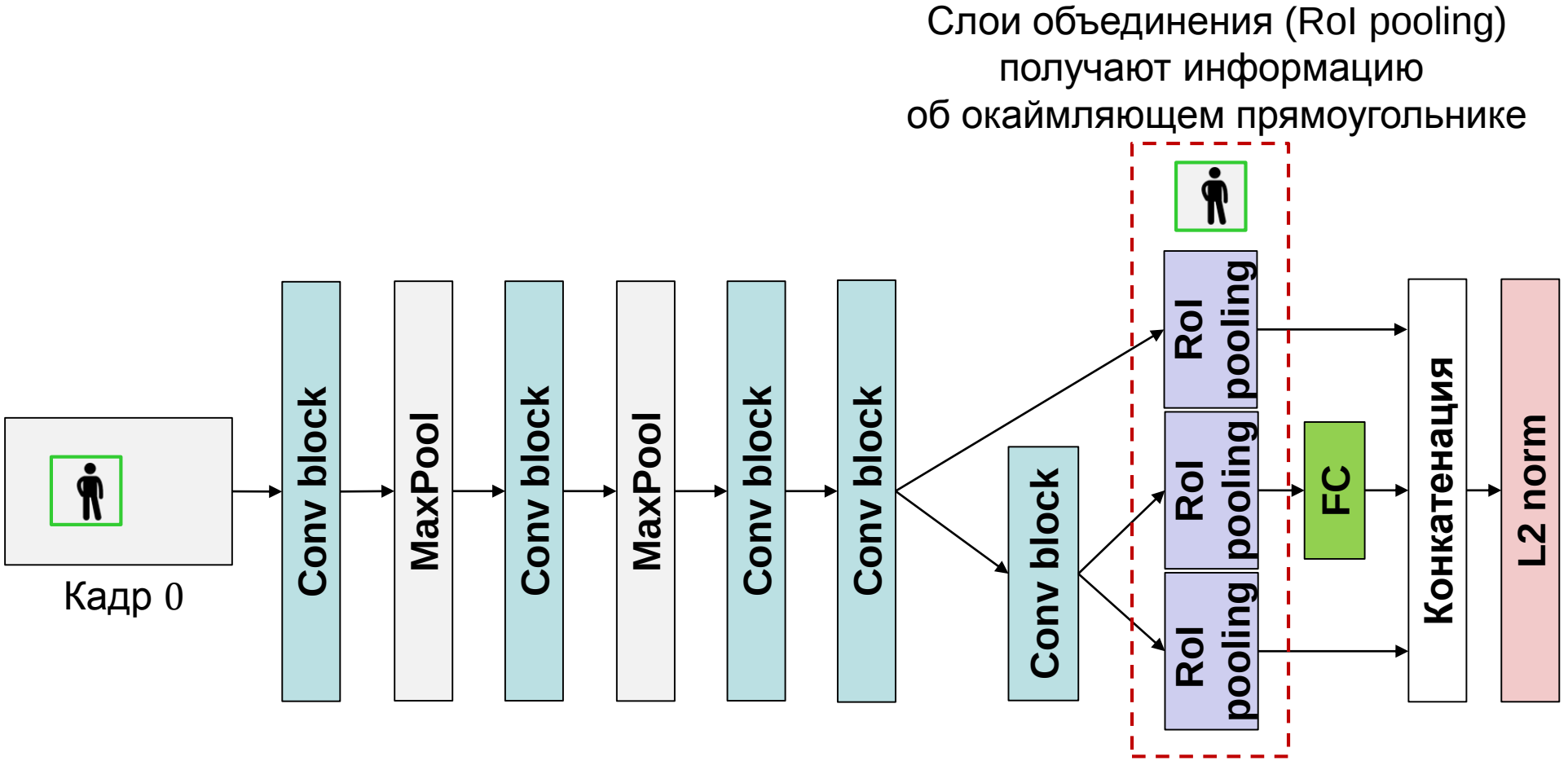
SINT (3)

- Сиамская модель состоит из двух идентичных ветвей
- Одна ветвь модели получает на вход первый кадр и окаймляющий прямоугольник из траектории, другая ветвь – текущий кадр и гипотезы (окаймляющие прямоугольники) о расположении объекта на текущем кадре
- Вход одной ветви модели:
  - Изображение – тензор размера 1x3x512x512
  - Набор окаймляющих прямоугольников – тензор размера Nx4, где N – количество гипотез о расположении объекта на новом кадре
- Алгоритм генерации гипотез подробно описан в исходной статье*

* Tao R., Smeulders A., Chang S.-F. Attributes and categories for generic instance search from one example. – 2015. – [openaccess.thecvf.com/content_cvpr_2015/papers/Tao_Attributes_and_Categories_2015_CVPR_paper.pdf].

SINT (4)

- Архитектура одной ветви:



SINT (5)

- Состав сверточных блоков (Conv block):
  - Conv: 3x3
  - Conv: 3x3
  - ReLU
- Параметры слоев объединения (RoI pooling):
  - Размер сетки по ширине (pooled width): 7
  - Размер сетки по высоте(pooled height): 7

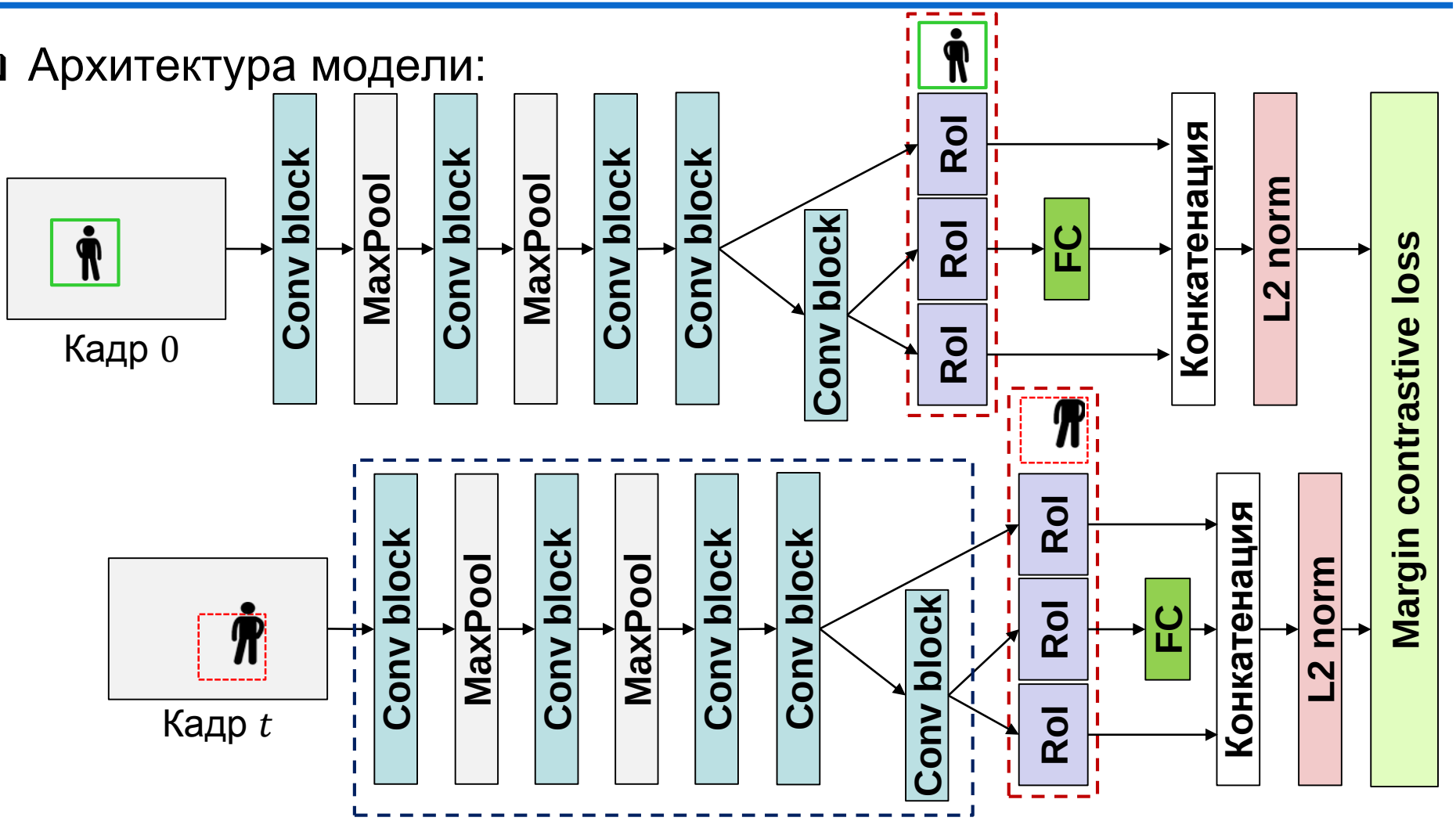
- Модель содержит небольшое количество слоев, чтобы больше опираться на низкоуровневые признаки объекта – ребра, углы

SINT (6)

- На выходе ветви формируется описание объекта
- Выход модели:
  - Дескриптор объекта в окаймляющем прямоугольнике – вектор размерности 54 272 (7*7*512 + 4 096 + 7*7*512)
- Дескриптор объекта строится посредством объединения выходов с трех слоев:
  - Выход первого слоя объединения с пространственным размером 7x7 и 512 каналами, вытянутый в одномерный вектор (flatten)
  - Выход полносвязного слоя – вектор размерности 4 096
  - Выход третьего слоя объединения с пространственным размером 7x7 и 512 каналами, вытянутый в одномерный вектор (flatten)

SINT (7)

- Архитектура модели:



Сверточная часть модели вычисляется один раз для всех гипотез

SINT (8)

- При обучении модели используется ***сравнительная функция потерь с отступом*** (margin contrastive loss)

$$L(x_j, x_k, y_{jk}) = \frac{1}{2} y_{jk} D^2 + \frac{1}{2}(1 - y_{jk}) \max\{0, \epsilon - D^2\},$$

где $D^2 = \|f(x_j) - f(x_k)\|$ – Евклидово расстояние между дескрипторами x_j и x_k,

$$y_{jk} = \begin{cases} 1, \text{если } x_j \text{ и } x_k - \text{это один и тот же объект} \\ 0, \text{если } x_j \text{ и } x_k - \text{это разные объекты} \end{cases}$$

ϵ (margin) – значение, при превышении которого объекты будут считаться разными

SINT (9)

- После вычисления дескрипторов для всех пар ‘объект-гипотеза’, выбирается та пара, дескриптор которой наиболее близок дескриптору объекта на первом кадре

$$\widehat{x_t} = \arg\max_{x_{j,t}} m\left(patch_{x_0}, patch_{x_{j,t}}\right)$$

$$m(x, y) = f(x)^T f(y)$$

где $f(x)$ – дескриптор, вычисленный для кадра с окаймляющим прямоугольником,

t – номер кадра, $x_{j,t}$ – гипотеза j для кадра t,

$patch_{x_0}$ – кадр 0 целиком и координаты окаймляющего прямоугольника объекта (разметка)

$patch_{x_{j,t}}$ – кадр t целиком и координаты окаймляющего прямоугольника, соответствующего гипотезе j

SiameseNET (1)

- Принцип работы SiameseNET*:
  - Детектируются объекты на кадрах M и N, и для каждой пары изображений объектов с помощью глубокой модели вычисляется оценка вероятности того, что это изображения одного объекта
  - Также вычисляется ***контекст*** для каждой такой пары
  - На основе дескриптора, полученного с последнего полносвязного слоя сети и контекста пары строится и обучается классификатор стохастического градиентного бустинга**, в дальнейшем он вычисляет правильные сопоставления объектов между кадрами M и N

* Leal-Taixé L., Canton-Ferrer C., Schindler. K. Learning by tracking: siamese CNN for robust target association. – 2016. – [https://arxiv.org/pdf/1604.07866.pdf].

** Friedman. J. Stochastic gradient boosting // Computational Statistics & Data Analysis. – 2002. P. 367–378.

SiameseNET (2)

- Пусть положение объекта в момент времени t_1 определяется парой $p_1 = (x, y)$, а его размеры $S_1 = (w, h)$, для второго момента времени вводятся аналогичные обозначения
- Контекст для пары объектов состоит из следующих значений:
  - Относительное изменение размера (relative size change)

$$\frac{S_1 - S_2}{S_1 + S_2} = \left(\frac{w_1 - w_2}{w_1 + w_2}, \frac{h_1 - h_2}{h_1 + h_2}\right)$$

  - Изменение положения (position change)

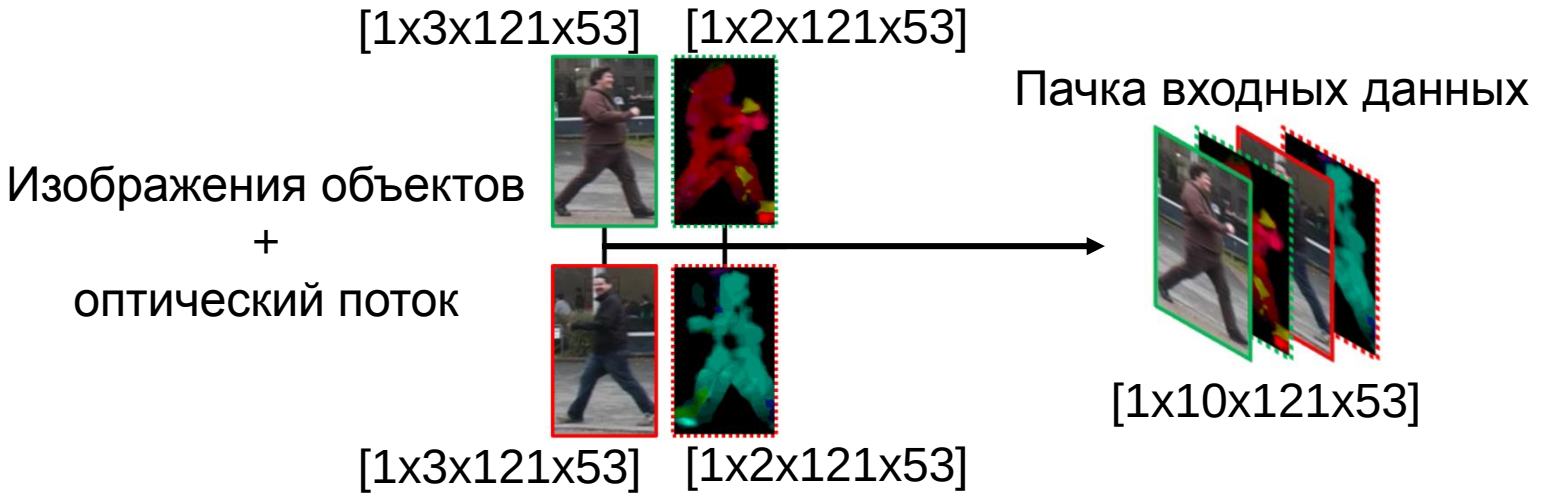
$$p_1 - p_2$$

  - Относительная скорость перемещения (relative velocity)

$$\frac{p_1 - p_2}{t_2 - t_1}$$

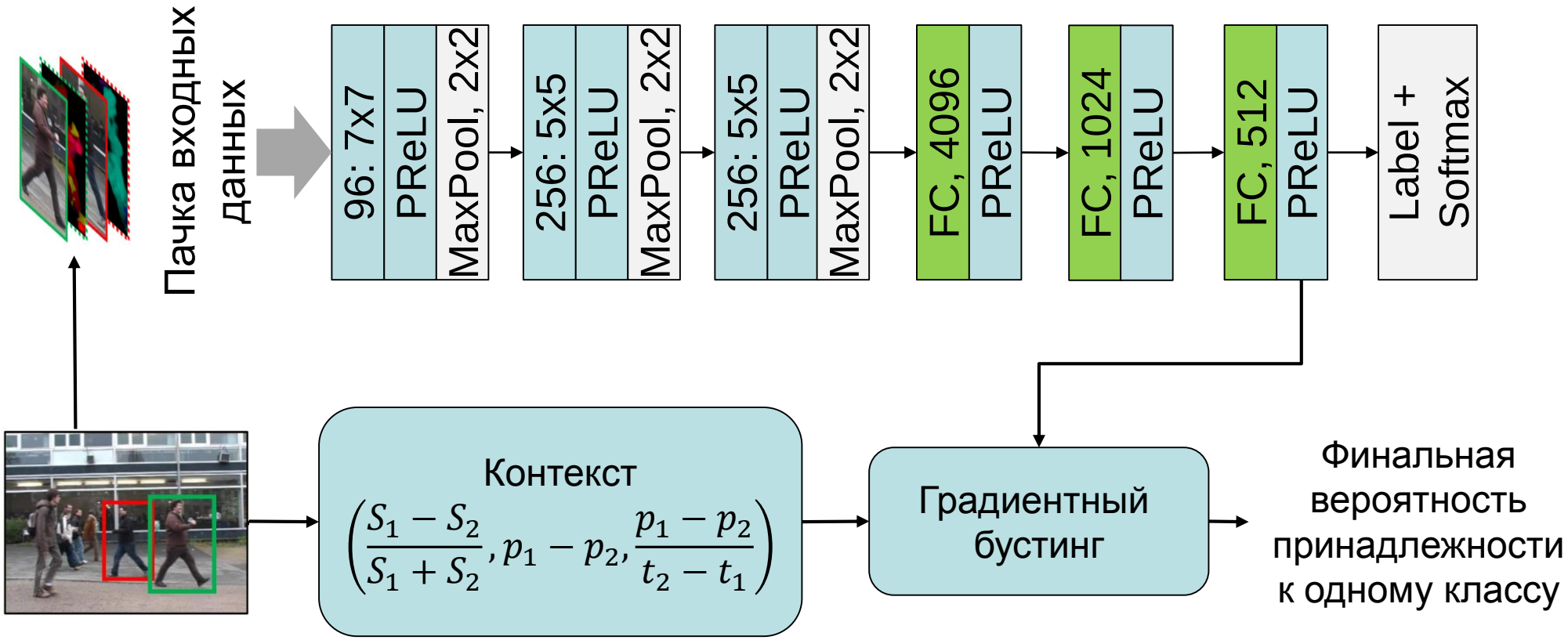
SiameseNET (3)

- SiameseNETs, как правило, содержат две ветви с двумя входами, на которые подаются изображения
- Разработчики SiameseNET экспериментировали с сиамскими сетями, и выяснили, что лучше соединить обе ветви сиамской сети в одну посредством объединения входных изображений в пачку входных данных



* Leal-Taixé L., Canton-Ferrer C., Schindler. K. Learning by tracking: siamese CNN for robust target association. – 2016. – [https://arxiv.org/pdf/1604.07866.pdf].

SiameseNET (4)



* Leal-Taixé L., Canton-Ferrer C., Schindler. K. Learning by tracking: siamese CNN for robust target association. – 2016. – [https://arxiv.org/pdf/1604.07866.pdf].

SiameseNET (5)

- Изображения на вход глубокой модели подаются в формате LUV, а не RGB
- Вход модели – тензор размера 10x121x53, 10 каналов тензора формируется из следующих данных:
  - Изображение объекта 1 в формате LUV (3 канала)
  - Оптический поток объекта 1 (2 канала)
  - Изображение объекта 2 LUV (3 канала)
  - Оптический поток объекта 2 (2 канала)
- Оптический поток вычисляется методом Фернебака*

* Farnebäck G. Two-Frame Motion Estimation Based on Polynomial Expansion. – 2003. – [http://www.diva-portal.org/smash/get/diva2:273847/FULLTEXT01.pdf].

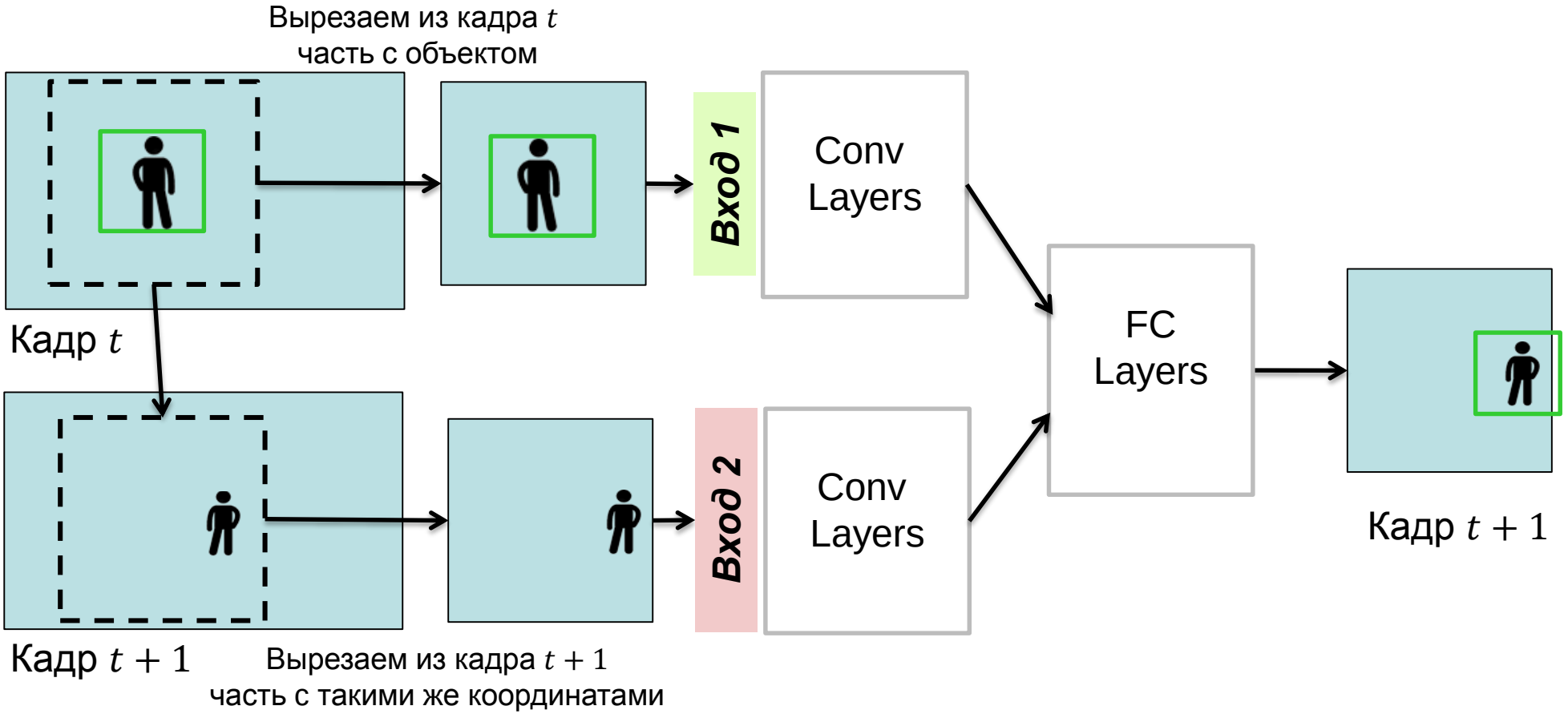
GOTURN (1)

- GOTURN (Generic Object Tracking Using Regression Network) обучается путем сравнения пар обрезанных кадров:
  - На первом кадре расположение объекта известно, и кадры обрезаются до размера, в два раза превышающего окаймляющий прямоугольник вокруг объекта, с центром изображения в центре объекта
  - Затем алгоритм пытается предсказать положение того же объекта на втором кадре
  - Сверточная нейронная сеть обучается предсказывать положение окаймляющего прямоугольника на втором кадре в рамках вырезанной области

* Held D., Thrun S., Savarese S. Learning to track at 100 FPS with deep regression networks. – 2016. – [https://arxiv.org/pdf/1604.01802.pdf].

GOTURN (2)

- Схема работы сети:



* Held D., Thrun S., Savarese S. Learning to track at 100 FPS with deep regression networks. – 2016. – [https://arxiv.org/pdf/1604.01802.pdf].

GOTURN (3)

- На вход 1 подается изображение с сопровождаемым объектом в центре. Изображение обрезано так, чтобы размер объекта составлял половину от размера изображения
- На вход 2 подается изображение нового кадра, но оно обрезано так же, как и первое
- Выходом модели является предсказание окаймляющего прямоугольника на втором изображении

* Held D., Thrun S., Savarese S. Learning to track at 100 FPS with deep regression networks. – 2016. – [https://arxiv.org/pdf/1604.01802.pdf].

GOTURN (4)

- Архитектура GOTURN повторяет архитектуру CaffeNet (AlexNet) с небольшими изменениями:
  - Модель GOTURN содержит две ветви сверточных слоев, аналогичные сверточным слоям CaffeNet, и перед полносвязными слоями выходы сверточных слоев двух ветвей конкатенируются, полносвязные слои идентичны CaffeNet
  - Размер выхода последнего слоя изменен с 1x1000 на 1x4 (выход модели – координаты окаймляющего прямоугольника)
- GOTURN является компактной и быстрой сверточной сетью

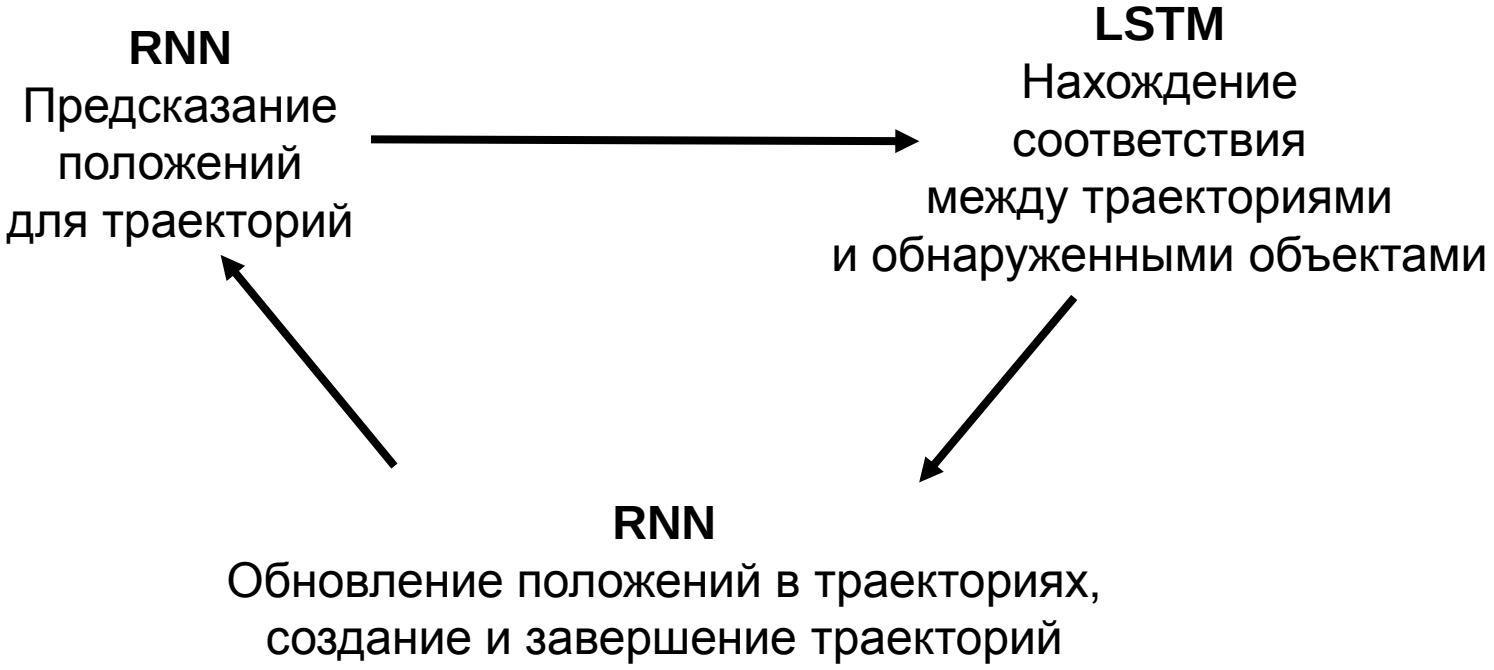
RNN-LSTM (1)

- Идея данной работы – заменить все шаги алгоритма сопровождения через нахождение соответствий на глубокие нейронные сети
- Группа моделей с LSTM-архитектурой вычисляют матрицу соответствий между обнаруженными объектами на кадре $t + 1$ и траекториями на кадре t (data association)
- Модель с RNN-архитектурой используется для предсказания положения объектов, обновления положения объектов и оценки вероятности существования объектов, а также для предсказания создания новых траекторий или удаления старых

* Milan A., Rezatofighi S.H., Dick A.R., et al. Online multi-target tracking using recurrent neural networks. – 2017. – [https://arxiv.org/pdf/1604.03635.pdf], [https://dl.acm.org/doi/10.5555/3298023.3298181].

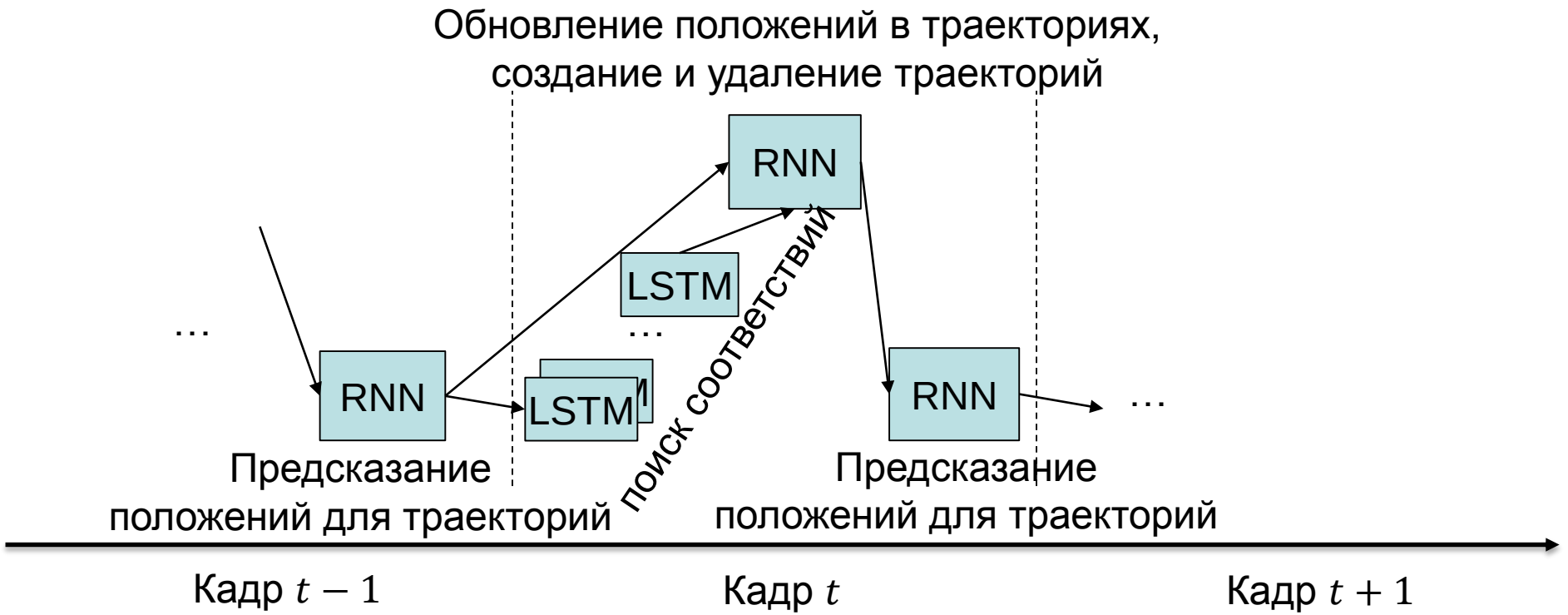
RNN-LSTM (2)

- Результаты работы RNN используются как вход LSTM
- Выход LSTM используется как вход RNN, в результате возникает цикличность работы



RNN-LSTM (3)

- Схема алгоритма:



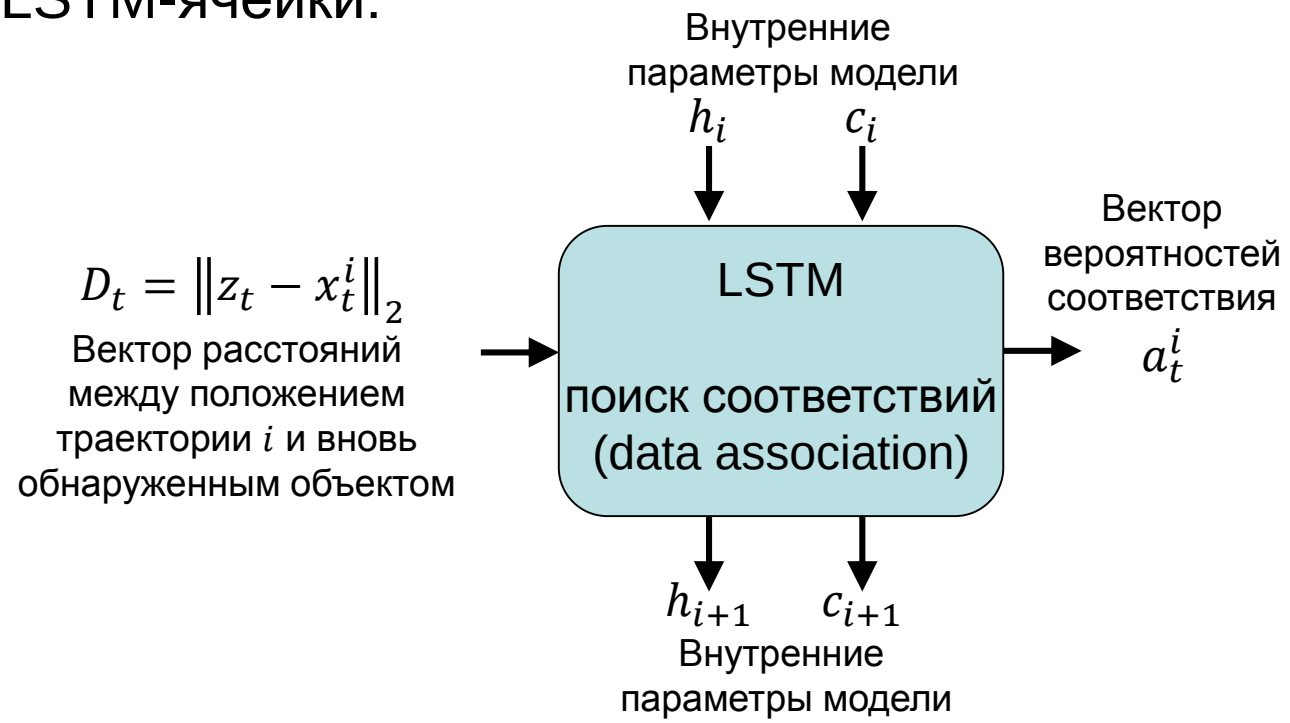
* Milan A., Rezatofighi S.H., Dick A.R., et al. Online multi-target tracking using recurrent neural networks. – 2017. – [https://arxiv.org/pdf/1604.03635.pdf], [https://dl.acm.org/doi/10.5555/3298023.3298181].

RNN-LSTM (4)

- RNN-модель обеспечивает следующий функционал:
  - Предсказание положений объектов
  - Обновление положений объектов с учетом информации о вновь обнаруженных объектах
  - Создание или завершение траекторий объектов
- LSTM-модель решает задачу нахождения соответствия между предсказанием положений объектов траекторий и обнаруженными объектами на основании матрицы расстояний между траекториями и объектами

RNN-LSTM (5)

- LSTM заменяет жадные алгоритмы поиска соответствий между траекториями и обнаруженными объектами, в частности, Венгерский алгоритм
- Одна LSTM-ячейка для одной траектории
- Схема LSTM-ячейки:



RNN-LSTM (6)

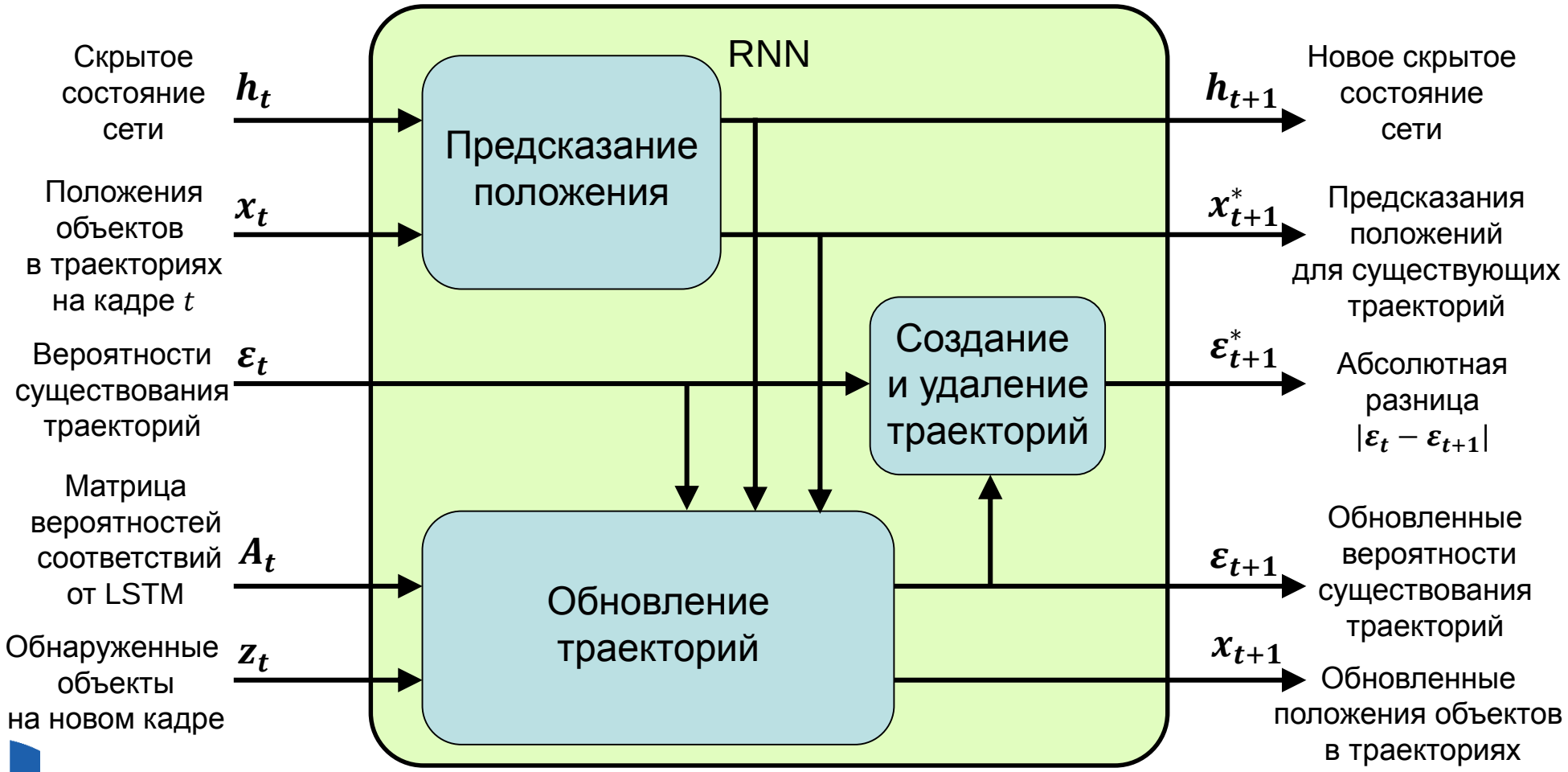
- Входы LSTM-ячейки:
  - Состояние сети c_i
  - Скрытое состояние сети h_i
  - Вектор расстояний между траекторией и обнаруженными объектами на новом кадре $D_t = \left\| z_t - x_t^i \right\|_2$
- Выходы LSTM-ячейки:
  - Состояние сети c_{i+1}
  - Скрытое состояние сети h_{i+1}
  - Вектор вероятностей соответствия между траекторией и объектами на новом кадре a_t^i

RNN-LSTM (7)

- Из векторов вероятностей a_t^i LSTM-моделей строится матрица вероятностей соответствия A_t, $A_t \in [0,1]^{N \times (M+1)}$
  - Каждый элемент $a_t^{i,j}$ вектора a_t^i содержит вероятность того, что объект j является частью траектории i
  - Сумма всех значений вектора вероятностей a_t^i равна 1
  - Количество строк матрицы равно N – количеству траекторий
  - Количество столбцов матрицы равно $M + 1$ (количество обнаруженных объектов на кадре + последний столбец для ситуаций, когда вероятности не были вычислены)
- Матрица A_t идет на вход RNN-модели

RNN-LSTM (8)

- Схема RNN-модели:



RNN-LSTM (9)

- Схема работы RNN-модели:
  1. В блоке предсказания положения на основе положений x_t в текущих траекториях строится предсказание положений x^*_{t+1} в траекториях на новом кадре
  2. В блоке обновления траекторий на основе обнаруженных объектов z_t, матрицы соответствия A_t, вероятности существования объектов ε_t, предсказания положений объектов x^*_{t+1} происходит вычисление новых положений x_{t+1} для траекторий и обновленных вероятностей существования траекторий ε_{t+1}. Матрица A_t нужна, чтобы лучше обрабатывать ситуации пересечения траекторий и другие неоднозначные ситуации

RNN-LSTM (10)

- Схема работы RNN-модели:
  3. В блоке создания и удаления траекторий на основе старых вероятностей существования траекторий ε_t и обновленных вероятностей существования траекторий ε_{t+1} вычисляется абсолютная разница $\varepsilon^*_{t+1} = |\varepsilon_t - \varepsilon_{t+1}|$. Данная разница необходима при обучении модели, чтобы глубокая модель научилась обрабатывать ситуации, когда детектирование объекта отсутствует, но заканчивать траекторию не надо, поскольку объект появится на следующем кадре

RNN-LSTM (11)

- Входы RNN:
  - Скрытое состояние h_t сети
  - Текущие положения объектов в траекториях x_t – вектор окаймляющих прямоугольников (x, y, w, h) размерности N, где x, y – координаты центра окаймляющего прямоугольника, а w, h – его размеры
  - Вероятности существования для траекторий ε_t
  - Матрица вероятностей соответствия между траекториями и объектами на новом кадре $A_t \in [0,1]^{N \times (M+1)}$
  - Обнаруженные объекты на новом кадре z_t – вектор окаймляющих прямоугольников размера M. Вектор строится с помощью некоторого алгоритма детектирования объектов (например, на базе глубоких моделей)

RNN-LSTM (12)

- Выходы RNN:
  - Новое скрытое состояние h_{t+1} сети
  - Предположение о новом положении объектов в траекториях x^*_{t+1}
  - Абсолютная разница $\varepsilon^*_{t+1} = |\varepsilon_{t+1} - \varepsilon_t|$ для обучения
  - Обновленные вероятности существования траекторий $\varepsilon_{t+1} \in (0,1)^N$. Во время работы модели, если данное значение становится меньше 0.6, то траектория завершается (termination)
  - Обновленные положения объектов x_{t+1} в траекториях – вектор окаймляющих прямоугольников размерности N

Заключение

- В настоящее время происходит развитие алгоритмов сопровождения объектов на видео, появляются все более крупные наборы данных и разрабатываются новые глубокие модели
- На текущий момент нельзя считать задачу сопровождения объектов решенной в целом
- Однако, системы, способные решить задачу сопровождения для конкретных классов объектов (автомобили, пешеходы) с высокой точностью уже существуют и применяются в промышленности, производстве, торговле

Основная литература (1)

- Milan A., et al. MOT16: A benchmark for multi-object tracking. – 2016. – [https://arxiv.org/pdf/1603.00831.pdf].
- Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric // International Conference on Image Processing. – 2017. – P. 3645–3649. – [https://arxiv.org/pdf/1703.07402.pdf], [https://ieeexplore.ieee.org/document/8296962].
- Tao R., Gavves E., Smeulders A. Siamese instance search for tracking. – 2016. – [https://arxiv.org/pdf/1605.05863.pdf], [https://ieeexplore.ieee.org/document/7780527].
- Leal-Taixé L., Canton-Ferrer C., Schindler K. Learning by tracking: siamese CNN for robust target association. – 2016. – [https://arxiv.org/pdf/1604.07866.pdf], [https://ieeexplore.ieee.org/document/7789549].

Основная литература (2)

- Held D., Thrun S., Savarese S. Learning to track at 100 FPS with deep regression networks. – 2016. – [https://arxiv.org/pdf/1604.01802.pdf].
- Milan A., Rezatofighi S.H., Dick A.R., et al. Online multi-target tracking using recurrent neural networks. – 2017. – [https://arxiv.org/pdf/1604.03635.pdf], [https://dl.acm.org/doi/10.5555/3298023.3298181].
- Ciaparrone G., et al. Deep learning in video multi-object tracking: a survey. – 2019. – [https://arxiv.org/pdf/1907.12740.pdf].

Авторский коллектив

- **Турлапов Вадим Евгеньевич**
  д.т.н., профессор кафедры МОСТ ИИТММ ННГУ
  vadim.turlapov@itmm.unn.ru
- **Васильев Евгений Павлович**
  преподаватель кафедры МОСТ ИИТММ ННГУ
  evgeny.vasiliev@itmm.unn.ru
- **Гетманская Александра Александровна**
  преподаватель кафедры МОСТ ИИТММ ННГУ
  alexandra.getmanskaya@itmm.unn.ru
- **Кустикова Валентина Дмитриевна**
  к.т.н., доцент каф. МОСТ ИИТММ ННГУ
  valentina.kustikova@itmm.unn.ru