

**Алгоритм трекинга при управлении соревновательной деятельностью команд
в онлайн на основе компьютерного зрения**

А.А.Полозов¹, Н.А. Мальцева¹, Г.С. Крамаренко¹, М.А. Липилин¹, А.Ф. Ахметзянов²

¹Уральский федеральный университет,

¹620014, г. Екатеринбург, ул. Мира 19, Россия,

²Сургутский государственный педагогический университет,

²628417, г. Сургут, ул. 50 лет ВЛКСМ, 10/2, Россия

Резюме. Цель. В статье представлены результаты исследования алгоритмов трекинга для анализа баскетбольной игры. Цель работы заключается в определении оптимального метода для применения отслеживания спортсменов в онлайн режиме. **Метод.** Исследование основано на методах и алгоритмах решения задач управления в организационных системах. **Результат.** Рассмотрены алгоритмы с реидентификацией объектов, учитывающие как динамику движения, так и внешний вид. В качестве кандидатов были выбраны наиболее популярные алгоритмы трекинга BYTE, взятого из алгоритма Bytetrack, и алгоритма Deepsort, показавшие высокий результат в задаче отслеживания пешеходов в толпе. Сравнение алгоритмов производилось по метрикам качества оценки трекинга МОТА и МОТР, а также по времени работы алгоритмов. Эксперименты проводились на датасете общей и спортивной направленности - MOT20 и SportMot. **Вывод.** Проведенное исследование показало, что наилучший результат при онлайн обработки кадров достигается алгоритмом ByteTrack. Он показал сопоставимые метрики качества при быстром времени выполнения. Авторы использовали открытые реализации алгоритмов и написали удобный интерфейс для проведения экспериментов над разными датасетами и источниками детекций.

Ключевые слова: алгоритм, баскетбол, трекинг, реидентификация, многообъектное отслеживание

Для цитирования: А.А. Полозов, Н.А. Мальцева, Г.С. Крамаренко, М.А. Липилин, А.Ф. Ахметзянов. Алгоритм трекинга при управлении соревновательной деятельностью команд в онлайн на основе компьютерного зрения. Вестник Дагестанского государственного технического университета. Технические науки. 2024; 51(2):120-127. DOI:10.21822/2073-6185-2024-51-2-120-127

**Tracking algorithm when managing competitive activities of top level teams online based
on computer vision**

A.A. Polozov¹, N.A. Maltseva¹, G.S. Kramarenko¹, M.A. Lipin¹, A.F. Akhmetzyanov²

¹Ural Federal University,

¹19 Mira St., Ekaterinburg 620014, Russia,

²Surgut State Pedagogical University,

²10/2 50 years of the Komsomol St., Surgut 628417, Russia

Abstract. Objective. The article presents the results of a study of tracking algorithms for analyzing a basketball game. The purpose of the work is to determine the optimal method for using athlete tracking when used online. **Method.** The research is based on methods and algorithms for solving management problems in organizational systems. **Result.** Algorithms with object re-identification are considered, taking into account both motion dynamics and appearance. The most popular tracking algorithms, BYTE, taken from the Bytetrack algorithm, and the

Deepsort algorithm, which showed high results in the task of tracking pedestrians in a crowd, were selected as candidates. The algorithms were compared using the MOTA and MOTP tracking assessment quality metrics, as well as the operating time of the algorithms. The experiments were carried out on a general and sports dataset - MOT20 и SportMot. **Conclusion.** The study showed that the best result in online frame processing is achieved by the ByteTrack algorithm. It showed comparable quality metrics with fast turnaround times. The authors used open implementations of the algorithms and wrote a convenient interface for conducting experiments on different datasets and detection sources.

Keywords: algorithm, basketball, tracking, re-identification, multi-object tracking

For citation: A.A. Polozov, N.A. Maltseva, G.S. Kramarenko, M.A. Lipin, A.F. Akhmetzyanov. Tracking algorithm when managing competitive activities of top level teams online based on computer vision. Herald of Daghestan State Technical University. Technical Sciences. 2024; 51(2):120-127. DOI:10.21822/2073-6185-2024-51-2-120-127

Введение. Чтобы подключить компьютеры к повседневной жизни иногда необходимо распознать объекты не только в рамках одного кадра, но и иметь возможность определить его местоположение во временном промежутке. Такую задачу выполняет алгоритмы трекинга, они отслеживают положения объекта относительно кадра. В начале алгоритм детекции или нейронная сеть находят интересующие объекты, определяя координаты и размеры прямоугольной рамки вокруг него, так называемые боксы [1, 2]. Затем трекинг определяет, встречался ли найденный объект ранее в кадрах, или является новым, ранее не найденным объектом. Каждому новому объекту присваивается уникальный номер – ID, который сохраняется всю последовательность кадров. Такие номера в последовательности кадров называются треклетами.

Сравнение объектов может выполняться по расстоянию между объектами – ближайший бокс ассоциируется с тем же объектом на предыдущем кадре. Такие методы относятся к классическим алгоритмам трекинга. В качестве расстояния может быть взят коэффициент перекрытия боксов – IOU [1]. Также существуют алгоритмы трекинга принимающие во внимание внешние признаки объекта – алгоритмы с реидентификацией объектов [2-11]. В качестве такого признака как правило выступает эмбединг бокса – числовой вектор, полученный с помощью преобразования изображения.

Основная задача трекинга состоит в сопоставлении положений целевого объекта на последовательности кадров, особенно если объект движется быстро относительно частоты кадров [6-14]. Еще более сложной задачей становится задача отслеживания объектов, которые пересекаются, меняют положение тела или долгое время не попадают в кадр. И самое сложное в задаче отслеживания считается, когда объекты перемещаются быстрее скорости пешехода. Такие ситуации встречается в анализе спортивных соревнований, в частности баскетбольных играх.

Постановка задачи. Цель исследования заключается в выборе устойчивого алгоритма трекинга в условиях спортивных соревнований. В соответствии с целью исследования поставлены задачи:

1. Учитываться должны все целевые объекты, хорошо и плохо различаемые в кадре, так как при движении спортсмена изображение может быть размытым.
2. Помимо определения местоположения по расстоянию между рамками объектов, должна быть дополнительная характеристика, привязанная к внешнему виду объекта
3. Вычислительная часть должна успевать обрабатывать большое количество информации с минимальной задержкой.

Методы исследования. Deepsort – специализированный алгоритм трекинга людей. Он был создан на основе метода SORT, который использует расстояния Махаланобиса и фильтр Калмана для оценки положения объектов по их местоположению

относительно кадров. DeepSort в свою очередь заменяет классические метрики сопоставления движения на метрику сопоставления объединяющую информацию о динамике движения и внешнем виде объекта. Для динамики движения используются параметры боксов, отфильтрованные и предсказанные с помощью фильтра Калмана – $(u, v, a, h, u', v', a', h')$, где u, v – это позиция предсказанного прямоугольника по X и Y, a – это соотношение сторон предсказанного прямоугольника (aspect ratio), h – высота прямоугольника, и производные по каждой величине. Для оценки внешнего вида используется нейронная сеть, вычисляющая эмбединг по изображению объекта. Она была разработана авторами алгоритма. Её архитектура состоит из 10 слоев (рис. 1), включая слои свертки Conv и остаточные слои Residual. Она была обучена более чем на 1 млн изображений 10000 разных людей для предсказания наиболее близких векторов по содержанию бокса.

Name	Patch Size/Stride	Output Size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max Pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10		128
Batch and ℓ_2 normalization		128

Рис. 1. Архитектура нейронной сети

Fig. 1. Neural network architecture

Последний слой сети вычисляет вектор размером 128×1 , который используется для подсчета расстояния между боксами. Оно включает в себя как предсказания с помощью фильтра Калмана, так и «косинусовое расстояние» (cosine distance) или коэффициент Отиаи. В итоге, расстояние от определенного объекта до предсказанного фильтром Калмана объекта (или объекта, который уже есть в числе тех, который наблюдался на предыдущих кадрах) равно:

$$D = \text{Lambda} * D_k + (1 - \text{Lambda}) * D_a$$

где D_a – это дистанция по внешней схожести, а D_k – расстояние Махаланобиса.

Дальше эта гибридная дистанция применяется в Венгерском алгоритме, чтобы правильно отсортировать определенные объекты с имеющимися ID.

ByteTrack. Метод BYTE не является полноценным алгоритмом трекинга, его можно назвать идеей, которая заключается в принципе сопоставления боксов. Благодаря ей, он может быть успешно интегрирован с 9 различными алгоритмами трекинга, включая трекинг на основе движения, трекинг с реидентификацией и другие. Сама идея метода заключается в том что BYTE учитывает все детекции в кадре, вне зависимости от порога уверенности (confidence) бокса.

Большинство методов трекинга получают идентификационные данные, связывая боксы с показателями, превышающими пороговое значение. Объекты с низкими показателями обнаружения попросту игнорируются, что приводит к значительной утрате истинных объектов и фрагментации траекторий. BYTE в свою очередь использует в процессе сопоставления все объекты – как с высокими, так и с низкими показателями. Он построен на предпосылке, что сходство с треклетами обеспечивает сильную подсказку для различения объектов и фона в детекциях с низкой уверенностью. Сначала BYTE сопоставляет детекции с высокой уверенностью с треклетами на основе сходства движения используя фильтр Калмана для предсказания местоположения объектов в новом кадре. Затем в зависимости от алгоритма трекинга, боксы сравниваются либо по IoU (коэффициент пересечения объектов) либо с использованием алгоритма реидентификации. Далее выполняется второе сопоставление между детекциями с низкой уверенностью и несопоставленными треклетами. В результате, загороженный человек с низкой оценкой обнаружения правиль-

но сопоставляется с предыдущим треклетом, а фон - удалится. Одним из примеров интеграции метода BYTE является алгоритм ByteTrack. ByteTrack использует сиамскую сетевую архитектуру, которая принимает два входных кадра и выводит оценку подобию, указывающую, одинаковые ли объекты в двух кадрах (рис.2).

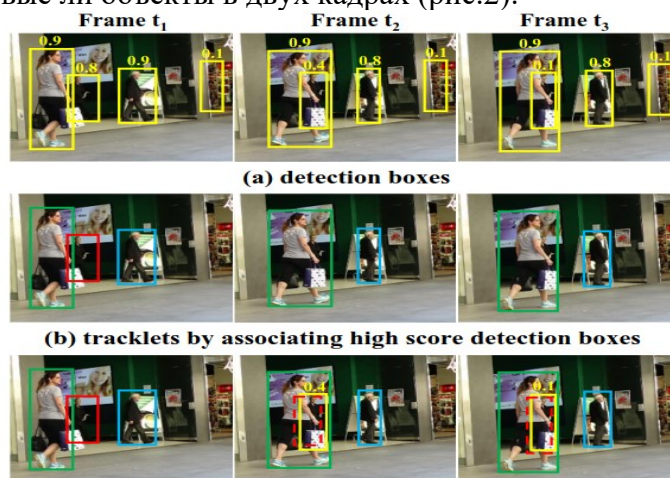


Рис. 2. Трекинг с учетом боксов с разным порогом уверенности
 Fig. 2. Tracking taking into account boxes with different confidence thresholds

Алгоритм использует простую, но эффективную технику увеличения данных для повышения производительности на сложных наборах данных. Было показано, что ByteTrack превосходит несколько современных алгоритмов отслеживания объектов, используя меньше параметров и обеспечивая более быстрое время вывода.

Обсуждение результатов. Для подсчета времени работы алгоритмов мы использовали видеокарту GeForce rtx3060. В качестве основного языка программирования был выбран Python 3.10. Эксперименты проводились с использованием двух наборов данных, MOT20 и SportMot. Координаты объектов с видео подавались на вход алгоритмам трекинга из трех разных источников: детектора YOLOv8 (с порогом уверенности 0.1), набора детекций сделанных авторами датасета (только для MOT20), и набором истинных меток (gt) из обоих датасетов (истинные метки объектов были отмечены людьми).

На рис. 3 приведен результат работы трекинга DeepSort на примере баскетбольной игры.

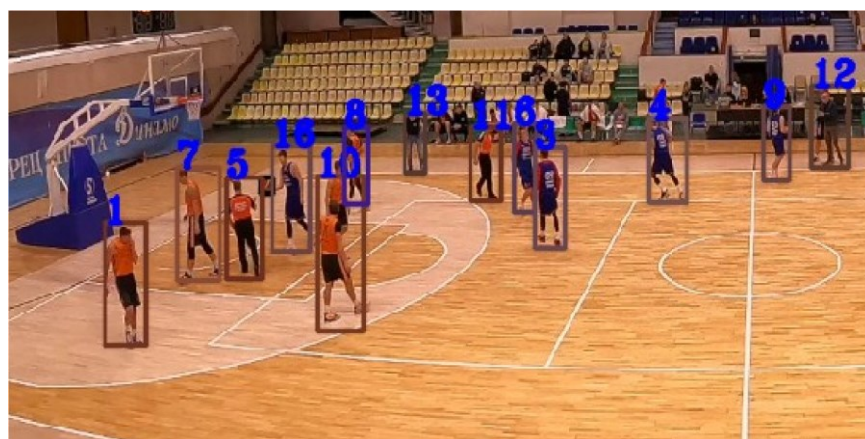


Рис. 3. Алгоритм DeepSort на примере баскетбольной игры
 Fig. 3. DeepSort algorithm using the example of a basketball game

Параметр уверенности сходимости боксов в ByteTrack установлен 0.7. Пороги track_thresh – 0.5, track_buffer – 50. Для каждого набора данных и каждого алгоритма измерялись метрики MOTA, MOTP, IDF1, FPS и среднее количество детекций на кадр (mDets). При вычислении fps учитывалось только время работы алгоритма трекинга, то есть не учитывалось время на загрузку кадров и получение детекций.

Метрики трекингов. Для оценки качества алгоритмов трекинга использовались классические метрики качества – MOTA, MOTP, IDF1 (табл.1-5).

MOTA (Multiple Object Tracking Accuracy) – метрика, которая измеряет общую точность трекинга, принимая во внимание все ошибки трекинга, включая ложные положительные, ложные отрицательные и переклечения идентификаторов объектов.

$$MOTA = 1 - \frac{|FN|+|FP|+|IDSW|}{|gtDet|}, \quad (1)$$

где: FN – объект пропущен

FP – отмечен объект, которого на самом деле нет,

IDSW – рядом проходящие объекты поменялись id,

gtDet – всего боксов на всех кадрах.

MOTP – метрика, которая измеряет точность локализации объекта. Она вычисляется как среднее значение IoU для всех верно сопоставленных детекций.

$$MOTP = \frac{1}{|TP|} \sum_{TP} S, \quad (2)$$

где: TP – детекции совпадающие с настоящим объектом.

S – IOU между найденным боксом и боксом указанным вручную.

IDF1 – гармоническое среднее между точностью и полнотой идентификации, где точность – это доля верно сопоставленных траекторий к числу предсказаний, а полнота – доля верно сопоставленных траекторий к числу реальных траекторий.

$$IDRecall = \frac{|IDTP|}{|IDTP|+0.5|IDFN|+0.5|IDFP|}, \quad (3)$$

где: IDTP – траектории, где IOU с GT траекторией больше порога,

IDFN – для GT траектории не нашлось предсказанной,

IDFP – предсказанная траектория, для которой не нашлось GT.

Таблица 1. Метрики, подсчитанные на детекциях размеченных вручную

Table 1. Metrics calculated on manually marked detections

Алгоритм/ Algorithm	Алгоритм детекции/ Detection algorithm	Датасет/ Dataset	Количе- ство кадров/ Number of frames	Mota	Motp	Idf1	FPS	Среднее количество детекций/ Average number of detections
deep-sort	gt	mot20 (train)	8931	0.989	0.076	0.644	0.85	149
byte	gt	mot20 (train)	8931	0.990	0.074	0.730	107	149

Таблица 2. Метрики, подсчитанные на детекциях с помощью детектора авторов датасета MOT20

Table 2. Metrics calculated on detections using the detector of the authors of the MOT20 dataset

Алгоритм/ Algorithm	Алгоритм детекции/ Detection algorithm	Датасет/ Dataset	Количе- ство кадров/ Number of frames	Mota	Motp	Idf1	FPS	Среднее количество детекций/ Av- erage number of de- tections
deep-sort	detections	mot20 (train)	8931	0.47	0.138	0.283	1.8	74
byte	detections	mot20 (train)	8931	0.476	0.138	0.356	190	74

Таблица 3. Метрики, подсчитанные на детекциях yolov8
Table 3. Metrics calculated on yolov8 detections

Алгоритм/ Algorithm	Алгоритм детекции/ Detection algorithm	Датасет/ Dataset	Количе- ство кадров/ Number of frames	Mota	Motp	Idf1	FPS	Среднее количество детекций/ Average number of detections
deep-sort	yolov8	mot20 (train)	8931	0.237	0.266	0.201	3	41
byte	yolov8	mot20 (train)	8931	0.062	0.235	0.092	718	41

Таблица 4. Метрики, подсчитанные на детекциях вручную
Table 4. Metrics calculated on manual detections

Алгоритм/ Algorithm	Алгоритм детекции/ Detection algorithm	Датасет/ Dataset	Количе- ство кадров/ Number of frames	Mota	Motp	Idf1	FPS	Среднее количество детекций/ Average number of detections
deep-sort	gt	sportMot (val)	26970	0.896	0.198	0.477	8	11
byte	gt	sportMot (val)	26970	0.902	0.199	0.055	1212	11

Таблица 5. Метрики подсчитанные на детекциях yolov8, датасет sportMot
Table 5. Metrics calculated on yolov8 detections, sportMot dataset

Алгоритм/ Algorithm	Detections from	Датасет/ Dataset	Frames	Mota	Motp	IDF1	FPS	mDets
deep-sort	yolov8	sportMot (val)	26970	0	0.238	0.035	5	21
byte	yolov8	sportMot (val)	26970	0.543	0.233	0.033	898	21

Вывод. Анализируя полученные результаты, можно сделать следующие выводы:

1. При тестировании на MOT20, используя в качестве детекций gt (верные метки датасета), и det (детекции, сделанные авторами датасета с помощью их детектора) алгоритмы выдают сопоставимые результаты, кроме idf1 и fps, где byte ощутимо выигрывает.
2. При использовании детекций от yolov8 с порогом 0.1, по точности deepsort сильно превзошел byte. Можно предположить, что это произошло из-за того, что yolov8 выдавал на видео людей с недостаточным уровнем уверенности, из-за чего byte не начинал их отслеживать.
3. На датасете sportMot результаты не так однозначны, при использовании и gt и yolov8 byte показывает более высокие результаты, однако idf1 лучше у deepsort. На motp результаты сопоставимы.
4. Во всех случаях, кроме mot20 yolov8, byte показывает лучшие результаты по метрике mota.

Алгоритм DeepSort [2] отлично справляется с трекингом пешеходов, в то время как алгоритм Byte хорошо работает вместе с детектором YOLOv8. Однако ни в одной из представленных работ не приводились примеры, связанные с трекингом в баскетболе.

Сравнение методов DeepSort и Byte, оба из которых используют методы реиндексации с помощью нейронных сетей, показало что методы сопоставимы по точности, но Byte превышает показатели времени вычислений (рис. 4).

Method	w/ Re-ID	MOT17			BDD100K			FPS
		MOTA↑	IDF1↑	IDs↓	mMOTA↑	mIDF1↑	IDs↓	
Sort		74.6	76.9	291	30.9	41.3	10067	30.1
DeepSORT	✓	75.4	77.2	239	24.5	38.2	10720	13.5
MOTDT	✓	75.8	77.6	273	26.7	39.8	14520	11.1
BYTE (ours)		76.6	79.3	159	39.4	48.9	27902	29.6
BYTE (ours)	✓	76.3	80.5	216	45.5	54.8	9140	11.8

Рис. 4. Сравнение с данными из статьи ByteTrack [1]

Fig. 4. Comparison with data from the ByteTrack article [1]

Отсюда можно сделать вывод что Byte является наиболее подходящим методом трекинга в баскетболе. В дальнейшем планируется продолжить работу над сравнениями алгоритмов трекинга. Для более точного определения результатов будет необходима тестовая выборка с заранее размеченными данными со спортивных соревнований.

Библиографический список:

1. Yifu Zhang¹, Peize Sun², Yi Jiang³, Dongdong Yu³, Fucheng Weng¹, Zehuan Yuan³, Ping Luo², Wenyu Liu¹, and Xinggang Wang. ByteTrack: Multi-Object Tracking by Associating Every Detection Box/ Huazhong University of Science and Technology ² The University of Hong Kong ³ ByteDance Inc.
2. N. Wojke, A. Bewley and D. Paulus, "Simple online and realtime tracking with a deep association metric," *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, 2017, pp. 3645-3649, doi: 10.1109/ICIP.2017.8296962.
3. Milan, Anton & Leal-Taixé, Laura & Reid, Ian & Roth, Stefan. (2016). MOT16: A Benchmark for Multi-Object Tracking.
4. Bernardin, K., Stiefel, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *J Image Video Proc* 2008, 246309 (2008). <https://doi.org/10.1155/2008/246309>
5. Punns NS, Sonbhadra SK, Agarwal S, Rai G. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. *arXiv*; 2021.
6. Wang J. et al. SportsTrack: An Innovative Method for Tracking Athletes in Sports Scenes //arXiv preprint arXiv:2211.07173. – 2022.
7. M.I.H. Azhar, F.H.K. Zaman, N.M. Tahir and H. Hashim, "People Tracking System Using DeepSORT," *2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, 2020, pp. 137-141, doi: 10.1109/ICCSCE50387.2020.9204956.
8. Y. Du et al., "StrongSORT: Make DeepSORT Great Again," in *IEEE Transactions on Multimedia*, doi: 10.1109/TMM.2023.3240881.
9. Yutao Cui, Chenkai Zeng, Xiaoyu Zhao, Yichun Yang, Gangshan Wu, and Limin Wang. Sportsmot: A large multi-object tracking dataset in multiple sports scenes. *arXiv preprint arXiv:2304.05170*, 2023.
10. Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian D. Reid, Stefan Roth, Konrad Schindler, Laura Leal-Taixé: MOT20: A benchmark for multi object tracking in crowded scenes. *CoRR abs/2003.09003* (2020)
11. Иванов, Ю.С. Анализ нейросетевых алгоритмов трекинга объектов / Ю.С. Иванов // Производственные технологии будущего: от создания к внедрению : Материалы V Международной научно-практической конференции, Комсомольск-на-Амуре, 06–11 декабря 2021 года / Редколлегия: С.И. Сухоруков (отв. ред.), А.С. Гудим, Н.Н. Любушкина. – Комсомольск-на-Амуре: Комсомольский-на-Амуре государственный университет, 2022. – С. 34-37. – EDN LXOWWD.
12. Д.А. Вражнов, А.В. Шаповалов, В.В. Николаев, "О качестве работы алгоритмов слежения за объектами на видео", *Компьютерные исследования и моделирование*, 4:2 (2012), 303–313
13. Chen, H. (2023). Appearance Aware Detector for MOT: An Enhanced ReID Branch for Tracking Memorize. *Academic Journal of Science and Technology*, 5(1), 46–48. <https://doi.org/10.54097/ajst.v5i1.5345> <https://habr.com/ru/articles/514450/>
14. <https://habr.com/ru/articles/166693/>

References:

1. Yifu Zhang¹, Peize Sun², Yi Jiang³, Dongdong Yu³, Fucheng Weng¹, Zehuan Yuan³, Ping Luo², Wenyu Liu¹, and Xinggang Wang. ByteTrack: Multi-Object Tracking by Associating Every Detection Box/ Huazhong University of Science and Technology ² The University of Hong Kong ³ ByteDance Inc.
2. N. Wojke, A. Bewley and D. Paulus, "Simple online and realtime tracking with a deep association metric," 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 2017; 3645-3649, doi: 10.1109/ICIP.2017.8296962.
3. Milan, Anton & Leal-Taixé, Laura & Reid, Ian & Roth, Stefan. (2016). MOT16: A Benchmark for Multi-Object Tracking.
4. Bernardin, K., Stiefelhagen, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. J Image Video Proc 2008, 246309 (2008). <https://doi.org/10.1155/2008/246309>
5. Punn NS, Sonbhadra SK, Agarwal S, Rai G. Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques. arXiv; 2021.
6. Wang J. et al. SportsTrack: An Innovative Method for Tracking Athletes in Sports Scenes //arXiv preprint arXiv:2211.07173. 2022.
7. M.I.H. Azhar, F.H.K. Zaman, N.M. Tahir and H. Hashim, "People Tracking System Using DeepSORT," 2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, 2020, pp. 137-141, doi: 10.1109/ICCSCE50387.2020.9204956.
8. Y. Du et al., "StrongSORT: Making DeepSORT Great Again," in IEEE Transactions on Multimedia, doi: 10.1109/TMM.2023.3240881.
9. Yutao Cui, Chenkai Zeng, Xiaoyu Zhao, Yichun Yang, Gangshan Wu, and Limin Wang. Sportsmot: A large multi-object tracking dataset in multiple sports scenes. arXiv preprint arXiv:2304.05170, 2023.
10. Patrick Dendorfer, Hamid RezaTofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian D. Reid, Stefan Roth, Konrad Schindler, Laura Leal-Taixé:MOT20: A benchmark for multi object tracking in crowded scenes. CoRR abs/2003.09003 (2020)
11. Ivanov, Yu. S. Analysis of neural network algorithms for object tracking / Yu. S. Ivanov // Manufacturing technologies of the future: from creation to implementation: Proceedings of the V International Scientific and Practical Conference, Komsomolsk-on-Amur, December 06–11, 2021 / Editorial Board: S.I. Sukhorukov (ed.), A.S. Gudim, N.N. Lyubushkina. – Komsomolsk-on-Amur: Komsomolsk-on-Amur State University, 2022;34-37. – EDN LXOWWD. (In Russ)
12. D.A. Vrazhnov, A.V. Shapovalov, V.V. Nikolaev, "On the quality of operation of algorithms for tracking objects on video," Computer Research and Modeling, 4:2 (2012), 303–313(In Russ)
13. Chen, H. (2023). Appearance Awarmed Detector for MOT: An Enhanced ReID Branch for Tracking Memorize. Academic Journal of Science and Technology, 5(1), 46–48. <https://doi.org/10.54097/ajst.v5i1.5345> Electronic resources <https://habr.com/ru/articles/514450/>
14. <https://habr.com/ru/articles/166693/>

Сведения об авторах:

Полозов Андрей Анатольевич, доктор педагогических наук, профессор, a.a.pozlov@mail.ru

Мальцева Наталья Анатольевна, инженер, Natalia.maltseva.susu@gmail.com

Крамаренко Георгий Сергеевич, студент; goshagks@ya.ru

Липилин Матвей Александрович, студент; Matvey.lipilin@gmail.com

Ахметзянов Артур Рахимзянович, преподаватель; Artur.rahimzyanovich@mail.ru

Information about authors:

Andrey A. Polozov, Dr. Sci. (Pedagogical); a.a.pozlov@mail.ru

Natalya A. Maltceva, Engineer; Natalia.maltseva.susu@gmail.com

Georgy S. Kramarenko, Student; goshagks@ya.ru

Matvey A. Lipilin, Student; Matvey.lipilin@gmail.com

Artur R. Akhmetzyanov, Teacher; Artur.rahimzyanovich@mail.ru

Конфликт интересов/Conflict of interest.

Авторы заявляют об отсутствии конфликта интересов/The authors declare no conflict of interest.

Поступила в редакцию/ Received 30.12.2023.

Одобрена после рецензирования / Revised 30.01.2024.

Принята в печать / Accepted for publication 30.01.2024.