

ИНФОРМАТИКА
INFORMATICSУДК 004
<https://doi.org/10.29235/1561-8323-2020-64-2-144-149>Поступило в редакцию 05.09.2019
Received 05.09.2019**Б. А. Залесский***Объединенный институт проблем информатики Национальной академии наук Беларуси,
Минск, Республика Беларусь***АЛГОРИТМ ОТСЛЕЖИВАНИЯ ОБЪЕКТОВ ДВИЖУЩЕЙСЯ ВИДЕОКАМЕРОЙ***(Представлено членом-корреспондентом А. В. Тузиковым)*

Аннотация. Представлен алгоритм АСТ (Adaptive Color Tracker) отслеживания объектов, наблюдаемых движущейся видеокамерой. Одной из особенностей работы алгоритма является адаптация набора признаков объекта к фону текущего кадра. При работе с текущим кадром из исходного набора признаков объекта, сформированного при его выделении на первом кадре, удаляются те, которые присущи не только объекту, но и в большой мере фону. Такие признаки не только не способствуют отделению объекта интереса от фона – они затрудняют корректное обнаружение объекта. Оставляются признаки объекта в большей мере характерные объекту и в то же время наименее характерные для фона текущего кадра. Признаки объекта и фона формируются на основе цветового представления кадров. Они вычисляются путем кластеризации 3D-векторов цвета пикселей кадров быстрой версией хорошо известного алгоритма k -средних или более простым и быстрым разбиением цветового пространства на 3D-параллелепипеды с последующей заменой цвета каждого пиксела на среднее значение векторов цвета, попавших в тот же параллелепипед, что и текущий цвет. Еще одна особенность алгоритма заключается в его вычислительной простоте, что делает возможным его использование на небольших мобильных вычислителях, например, на Jetson TX1 или TX2.

Алгоритм был протестирован на видеопоследовательностях, снятых различными видеокамерами, а также на общеизвестном наборе данных TV77, содержащем 77 различных размеченных видеопоследовательностей. Тесты показали работоспособность алгоритма. На тестовых изображениях его точность и быстродействие превосходили показатели трекеров, реализованных в библиотеке компьютерного зрения OpenCV 4.1.

Ключевые слова: видеопоследовательности, алгоритмы отслеживания объектов, адаптация признаков

Для цитирования: Залесский, Б. А. Алгоритм отслеживания объектов движущейся видеокамерой / Б. А. Залесский // Докл. Нац. акад. наук Беларуси. – 2020. – Т. 64, № 2. – С. 144–149. <https://doi.org/10.29235/1561-8323-2020-64-2-144-149>

Boris A. Zalesky*United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, Republic of Belarus***OBJECT TRACKING ALGORITHM BY MOVING VIDEO CAMERA***(Communicated by Corresponding Member Alexander V. Tuzikov)*

Abstract. The algorithm ACT (Adaptive Color Tracker) to track objects by a moving video camera is presented. One of the features of the algorithm is the adaptation of the feature set of the tracked object to the background of the current frame. At each step, the algorithm extracts from the object features those that are more specific to the object and at the same time are at least specific to the current frame background, since the rest of the object features not only do not contribute to the separation of the tracked object from the background, but also impede its correct detection. The features of the object and background are formed based on the color representations of scenes. They can be computed in two ways. The first way is 3D-color vectors of the clustered image of the object and the background by a fast version of the well-known k -means algorithm. The second way consists in simpler and faster partitioning of the RGB-color space into 3D-parallelepipeds and subsequent replacement of the color of each pixel with the average value of all colors belonging to the same parallelepiped as the pixel color. Another specificity of the algorithm is its simplicity, which allows it to be used on small mobile computers, such as the Jetson TX1 or TX2.

The algorithm was tested on video sequences captured by various camcorders, as well as by using the well-known TV77 data set, containing 77 different tagged video sequences. The tests have shown the efficiency of the algorithm. On the test images, its accuracy and speed overcome the characteristics of the trackers implemented in the computer vision library OpenCV 4.1.

Keywords: video sequences, object tracking algorithms, adaptation of feature set

For citation: Zalesky B. A. Object tracking algorithm by moving camera. *Doklady Natsional'noi akademii nauk Belarusi = Doklady of the National Academy of Sciences of Belarus*, 2020, vol. 64, no. 2, pp. 144–149 (in Russian). <https://doi.org/10.29235/1561-8323-2020-64-2-144-149>

Введение. Задача отслеживания объектов, наблюдаемых видеокамерой, в общем случае движущейся, хорошо известна в компьютерном зрении. Приведем ее традиционную формулировку. На одном из кадров видеопоследовательности, снимаемой камерой, выделена вручную или автоматически область, содержащая объект интереса. Задача заключается в автоматическом нахождении объекта интереса на последующих кадрах в режиме реального времени или близком к нему.

В настоящее время по данной тематике опубликовано значительное число работ, ссылки на которые приведены, например, в [1–7]. В работах описано множество трекеров, а также результаты экспериментального исследования их свойств. Основы для использования трекеров были заложены в 1980-х годах, хотя начало их активного использования приходится на 1990-е [1]. Первые трекеры были основаны на уже известных алгоритмах, применяемых для решения задач обработки изображений. К таковым можно отнести, например, алгоритм вычисления оптического потока Лукаса–Канаде, который был предназначен для совмещения изображений (image registration) [8] и корреляционные алгоритмы.

Классификация более современных алгоритмов отслеживания объектов приведена в [7]. Среди недавно разработанных трекеров упомянем CSR-DCF [8], относящийся к классу современных дискриминантно-корреляционных фильтров, показавший наилучшие результаты на наборе тестовых изображениях OTB100, а также современные нейросетевые трекеры [9].

Недостатками нейросетевых трекеров является необходимость их предварительного обучения, достаточно сложная программная реализация, высокая вычислительная сложность, а также чувствительность к шумам и искажениям изображения.

Материалы и методы исследования. Предлагается алгоритм АСТ (Adaptive Color Tracker) отслеживания объектов, наблюдаемых движущейся видеокамерой, особенностью которого является адаптация набора признаков объекта к фону текущего кадра. АСТ не требует предварительного обучения. При работе с текущим кадром из набора признаков объекта, вычисленного по его исходному изображению, алгоритм удаляет те, которые в большей мере характеризуют фон, а не сам объект. Признаки объекта и фона формируются на основе цветового представления изображений. Они вычисляются путем кластеризации 3D-векторов цвета пикселей кадров с помощью быстрой версии хорошо известного алгоритма k -средних или более простого и быстрого разбиения цветового пространства на 3D-параллелепипеды с последующей заменой вектора цвета каждого пикселя на среднее значение векторов цвета, попавших в тот же параллелепипед, что и текущий цвет.

Алгоритм достаточно просто реализуется программно. Он вычислительно не затратен, что делает возможным его использование на сравнительно небольших мобильных вычислителях, например, на Jetson TX1 или TX2.

Тестирование алгоритма проводилось на видеопоследовательностях, снятых различными видеокамерами, а также на общеизвестном наборе данных TV77, содержащем 77 различных размеченных видеопоследовательностей. Тесты показали возможность применения алгоритма для практического отслеживания объектов. На тестовых изображениях его точность и быстродействие превосходили показатели трекеров BOOSTING, MIL, KCF, TLD, MEDIANFLOW, GOTURN, MOSSE, CSRT, из библиотеки компьютерного зрения OpenCV 4.1.

АСТ основан на сравнении цветовых представлений объекта интереса и фона изображения, хотя предложенный подход может без изменений использоваться и для других числовых признаков. Для описания алгоритма, реализующего трекер, обозначим через $S = \{(x, y)\}$, $x = 0, \dots, m - 1$, $y = 0, \dots, n - 1$, множество пикселей цветного RGB-изображения $\mathbf{I} = \{I(\mathbf{p})\}$, $\mathbf{p} \in S$, с векторами цвета $I(\mathbf{p}) = (I_R(\mathbf{p}),$

$I_G(\mathbf{p}), I_B(\mathbf{p})$. Кадры видеопоследовательности, представляющие собой цветные изображения \mathbf{I}_t , будем индексировать дискретным временем t . Будем предполагать, что объект интереса, выделенный на исходном кадре \mathbf{I}_0 или другом изображении вручную либо автоматически каким-либо алгоритмом, представлен цветным изображением \mathbf{O} размера $k \times \ell$ ($k < m, \ell < n$) с множеством пикселей $S_{\mathbf{O}}$.

Как упоминалось выше, в алгоритме используются изображения, кластеризованные либо классическим алгоритмом k -средних, либо более простым и быстрым разбиением RGB-куба на 3D-параллелепипеды P_{μ} такие, что $P_{\mu} \cap P_{\nu} = \emptyset, \mu \neq \nu, \cup P_{\mu} = [0, \dots, 255]^3$, с последующей заменой каждого вектора цвета $I(\mathbf{p})$ на среднее значение

$$C_I(\mathbf{p}) = \frac{1}{|U(\mathbf{p})|} \sum_{I(\mathbf{q}) \in U(\mathbf{p})} I(\mathbf{q}),$$

где $U(\mathbf{p}) = \{I(\mathbf{q}) : I(\mathbf{q}) \in P_{\mu}, I(\mathbf{p}) \in P_{\mu}\}$.

Выполнение алгоритма начинается с получения кластерных представлений $\mathbf{C}_{\mathbf{O}} = \{C_{\mathbf{O}}(\mathbf{p})\}, \mathbf{p} \in S_{\mathbf{O}}$, изображения объекта \mathbf{O} и $\mathbf{C}_{\mathbf{B}} = \{C_{\mathbf{B}}(\mathbf{p})\}, \mathbf{p} \in S_{\mathbf{B}}, (S_{\mathbf{B}} = S \setminus S_{\mathbf{O}})$ фона $\mathbf{B} = \{I(\mathbf{p})\}, \mathbf{p} \in S_{\mathbf{B}}$. Одновременно с получением кластерного представления \mathbf{O} просчитывается сколько раз каждый цвет $C_{\mathbf{O}}(\mathbf{p})$ встречается в $\mathbf{C}_{\mathbf{O}}$.

Для наперед заданного целого числа K (которое является входным параметром для алгоритма) предварительное описание объекта формируется в виде упорядоченного $3K$ -мерного набора цветов кластеризованного объекта

$$F_{\mathbf{O}} = (C_{\mathbf{O}}^{(0)}, C_{\mathbf{O}}^{(1)}, \dots, C_{\mathbf{O}}^{(K-1)}),$$

где $C_{\mathbf{O}}^{(0)}$ соответствует цвету, наиболее часто встретившемуся в $\mathbf{C}_{\mathbf{O}}$. $C_{\mathbf{O}}^{(1)}$ – второму по встречаемости, и наконец $C_{\mathbf{O}}^{(K-1)}$ – K -му по встречаемости в $\mathbf{C}_{\mathbf{O}}$ цвету. Аналогично, на основе частоты встречаемости строится признаковое описание

$$F_{\mathbf{B}} = (C_{\mathbf{B}}^{(0)}, C_{\mathbf{B}}^{(1)}, \dots, C_{\mathbf{B}}^{(K-1)})$$

фона \mathbf{B} исходного кадра \mathbf{I}_0 .

Далее для каждого пиксела $\mathbf{p} \in S_{\mathbf{B}}$ фона \mathbf{B} вычисляются расстояния его цвета $I(\mathbf{p})$ до ближайшего $C_{\mathbf{O}}^{(j)}$ из $F_{\mathbf{O}}$ и $C_{\mathbf{B}}^{(j)}$ из $F_{\mathbf{B}}$

$$r_{\mathbf{O}}(\mathbf{p}) = \min_{0 \leq j \leq K} \|B(\mathbf{p}) - C_{\mathbf{O}}^{(j)}\| \text{ и } r_{\mathbf{B}}(\mathbf{p}) = \min_{0 \leq j \leq K} \|B(\mathbf{p}) - C_{\mathbf{B}}^{(j)}\|.$$

Если для $\mathbf{p} \in S_{\mathbf{B}}$ выполняется неравенство $r_{\mathbf{O}}(\mathbf{p}) \leq r_{\mathbf{B}}(\mathbf{p})$ (цвет пиксела $B(\mathbf{p})$ фона \mathbf{B} «притягивается» к кластеризованному изображению объекта, а не фона) в j -й элемент вектора $e(j)$, счетчика ошибок признака $C_{\mathbf{O}}^{(j)}$, добавляется единица. После просмотра всех пикселей фона в каждом элементе j вектора e будет храниться количество пикселей фона, которые по цвету ближе к одному из цветов кластерного представления изображения объекта $F_{\mathbf{O}}$, чем к любому цвету признака фона $F_{\mathbf{B}}$.

После этого выбирается целое число d (еще один входной параметр алгоритма), определяющее минимально допустимую размерность адаптированного вектора признаков объекта F_{adapt} , ($\dim(F_{\text{adapt}}) \geq d$), который формируется из $F_{\mathbf{O}}$ в соответствии со свойствами каждого текущего кадра \mathbf{I}_t и используется для обнаружения объекта на этом кадре. Фиксируется также целое L_{err} , ограничивающее максимальное число пикселей фона, которые могут быть ошибочно «притянуты» к каждому элементу $F_{\text{adapt}}^{(j)}$ адаптированного признака объекта.

Если число цветов исходного признака объекта $F_{\mathbf{O}}$, удовлетворяющих условию $e(j) < L_{\text{err}}$ больше d , т. е.

$$|\{j : e(j) < L_{\text{err}}\}| \geq d,$$

формируется адаптированный признак

$$F_{\text{adapt}} = (C_{\mathbf{O}}^{j_1}, C_{\mathbf{O}}^{j_2}, \dots, C_{\mathbf{O}}^{j_d}), d \leq k \leq K,$$

в котором оставляются только $C_{\mathbf{O}}^{(j)}$, удовлетворяющие условию $e(j) < L_{\text{err}}$.

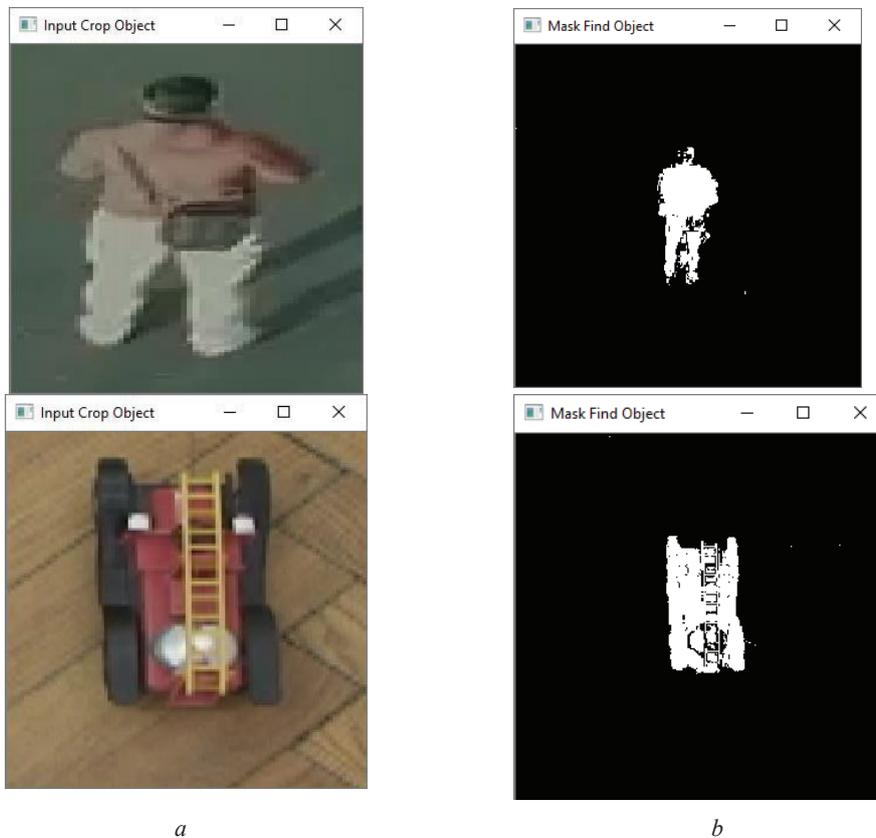
Если число $C_{\mathbf{O}}^{(j)}$, удовлетворяющие условию $e(j) < L_{\text{err}}$, меньше d , то в качестве $F_{\text{adapt}} = (C_{\mathbf{O}}^{j_1}, C_{\mathbf{O}}^{j_2}, \dots, C_{\mathbf{O}}^{j_d})$ выбираются ровно d цветов $C_{\mathbf{O}}^{(j)}$ с минимальными количеством ошибок $e(j)$.

Адаптированный признак F_{adapt} применяется начиная с исходного кадра (на котором уже выделено изображение \mathbf{O} , содержащее объект интереса) для того, чтобы удалить из $F_{\mathbf{O}}$ элементы $C_{\mathbf{O}}^{(j)}$, характеризующие в большей мере фон исходного кадра, а не сам объект. Для этого вычисляется с помощью описанных выше действий $F_{\mathbf{B}}$ по фону \mathbf{B} исходного кадра \mathbf{I}_0 , а также $F_{\text{adapt},0}$, а затем полагается и используется в дальнейшем $F_{\mathbf{O}} = F_{\text{adapt},0}$.

На рисунке приведены примеры выделения объектов интереса из содержащих их изображений \mathbf{O} с помощью $F_{\text{adapt},0}$, вычисленному по исходному кадру. На изображении \mathbf{O} отмечаются белым цветом пиксели, для которых

$$\min_{0 \leq j \leq k} \|O(\mathbf{p}) - C_{\text{adapt}}^{(j)}\| < \min_{0 \leq j \leq K} \|O(\mathbf{p}) - C_{\mathbf{B}}^{(j)}\|.$$

Для каждого текущего кадра процедура вычисления $F_{\text{adapt}} = F_{\text{adapt},t}$ на основе $F_{\mathbf{O}}$ и $F_{\mathbf{B}} = F_{\mathbf{B},t}$ повторяется, а затем в качестве области наиболее вероятного расположения объекта выбирается прямоугольник $S_t \subset S$, на котором больше всего цветов пикселей $I_t(\mathbf{p})$, $\mathbf{p} \in S_t$, «притягиваются» к $F_{\text{adapt},t}$ (для них $\min_{0 \leq j \leq k} \|I_t(\mathbf{p}) - C_{\text{adapt}}^{(j)}\| < \min_{0 \leq j \leq K} \|I_t(\mathbf{p}) - C_{\mathbf{B}}^{(j)}\|$).



Изображения \mathbf{O} , содержащие объект интереса (a) и маски объектов, найденные с помощью $F_{\text{adapt},0}$ (b)

Images \mathbf{O} containing tracked object (a) and object trackers found using $F_{\text{adapt},0}$ (b)

Результаты и их обсуждение. Для исследования характеристик построенного алгоритма АСТ было проведено его тестирование на видеопоследовательностях разных размеров, начиная от 320×240 до 1920×1080 (Full HD), снятых различными видеокамерами. Помимо этого, проводилось тестирование на общеизвестном наборе данных TV77, содержащем 77 различных размеченных видеопоследовательностей, а также сравнение АСТ с реализациями алгоритмов BOOSTING, MIL, KCF, TLD, MEDIANFLOW, GOTURN, MOSSE, CSRT из библиотеки компьютерного зрения OpenCV 4.1. 1920×1080 (Full HD). С нашей точки зрения, одним из основных критериев, которым должен удовлетворять трекер, является процент видеопоследовательностей, на которых не произошел срыв сопровождения объекта интереса. Еще один важный критерий – процент видеопоследовательностей, на которых после потери объекта трекер находит объект повторно и продолжает сопровождение.

Ниже приведены результаты тестирования разработанного алгоритма АСТ с помощью видеопоследовательностей из тестового набора данных TV77 и для сравнения – известного алгоритма CSRT, работа которого значительно лучше остальных алгоритмов, реализованных в OpenCV 4.1:

Трекер	Без срывов	Со срывом и возобновлением	Со срывом без возобновления	Среднее быстродействие
АСТ	76 %	24 %	0 %	68,1
CSRT	71 %	0 %	29 %	33,9

Заметим, что предложенный трекер АСТ не только показал лучшие результаты по сопровождению объектов без срыва, но он также не допустил ни одного случая безвозвратной потери сопровождаемого объекта.

Заключение. Проведенные эксперименты с видеопоследовательностями, снятыми различными видеокамерами, а также с видеопоследовательностями из известного набора данных TV77, применяемого для тестирования трекеров, показали работоспособность предложенного алгоритма АСТ. Положительными особенностями алгоритма являются простота его реализации, достаточное быстродействие (которое можно увеличить путем реализации алгоритма на CUDA), возможность повторного обнаружения объекта интереса в случаях его временного исчезновения из кадров или срыва сопровождения.

В будущем планируется исследовать возможность использования адаптированных признаков объекта не только на основе анализа первого, но и последующих кадров.

Список использованных источников

1. Yilmaz, A. Object tracking: A survey / A. Yilmaz, O. Javed, M. Shah // ACM Computing Surveys. – 2006. – Vol. 38, N 4. – P. 1–13. <https://doi.org/10.1145/1177352.1177355>
2. Visual tracking: An experimental survey / A. W. Smeulders [et al.] // IEEE TPAMI. – 2014. – Vol. 36, N 7. – P. 1442–1467. <https://doi.org/10.1109/tpami.2013.230>
3. Video processing techniques for traffic flow monitoring: A survey / B. Tian [et al.] // 14th International IEEE Conference on Intelligent Transportation Systems. – Washington, 2011. – P. 1103–1108. <https://doi.org/10.1109/itsc.2011.6083125>
4. Chen, Z. An experimental survey on correlation filter-based tracking [Electronic resource] / Z. Chen, Z. Hong, D. Tao. – 2015. – Mode of access: <https://arxiv.org/abs/1509.05520>. – Date of access: 05.09.19.
5. A survey of appearance models in visual object tracking / X. Li [et al.] // ACM Transactions on Intelligent Systems and Technology. – 2013. – Vol. 4, N 4. – P. 1–48. <https://doi.org/10.1145/2508037.2508039>
6. Liu, Q. Survey of single-target visual tracking methods based on online learning / Q. Liu, X. Zhao, Z. Hou // IET Computer Vision. – 2014. – Vol. 8, N 5. – P. 419–428. <https://doi.org/10.1049/iet-cvi.2013.0134>
7. Fiaz, M. Tracking Noisy Targets: A Review of Recent Object Tracking Approaches [Electronic resource] / M. Fiaz, A. Mahmood, S.-K. Jung. – 2018. – Mode of access: <https://arxiv.org/abs/1802.03098>. – Date of access: 05.09.19.
8. Lucas B. An iterative image registration technique with an application to stereo vision / B. Lucas, T. Kanade // Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI'81). – Vancouver, 1981. – Vol. 2. – P. 674–679.
9. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Electronic resource]. – Long Beach, 2019. – Mode of access: <http://openaccess.thecvf.com/CVPR2019.ru>. – Date of access: 05.09.19.

Reference

1. Yilmaz A., Javed O., Shah M. Object tracking: A survey. *Acm computing surveys*, 2006, vol. 38, no. 4, pp. 1–13. <https://doi.org/10.1145/1177352.1177355>
2. Smeulders A. W., Chu D. M., Cucchiara R., Calderara S., Dehghan A., Shah M. Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014, vol. 36, no. 7, pp. 1442–1468. <https://doi.org/10.1109/tpami.2013.230>
3. Tian B., Yao Q., Gu Y., Wang K., Li Y. Video processing techniques for traffic flow monitoring: A survey. *14th International IEEE Conference on Intelligent Transportation Systems*, Washington, 2011, pp. 1103–1108. <https://doi.org/10.1109/itsc.2011.6083125>
4. Chen Z., Hong Z., Tao D. *An experimental survey on correlation filter-based tracking*, 2005. Available at: <https://arxiv.org/abs/1509.05520> (accessed 5 September 2019).
5. Li X., Hu W., Shen C., Zhang Z., Dick A., Van Den Hengel A. A survey of appearance models in visual object tracking. *ACM Transactions on Intelligent Systems and Technology*, 2013, vol. 4, no. 4, pp. 1–48. <https://doi.org/10.1145/2508037.2508039>
6. Liu Q., Zhao X., Hou Z. Survey of single-target visual tracking methods based on online learning. *IET Computer Vision*, 2014, vol. 8, no. 5, pp. 419–428. <https://doi.org/10.1049/iet-cvi.2013.0134>
7. Fiaz M., Mahmood A., Jung S.-K. *Tracking Noisy Targets: A Review of Recent Object Tracking Approaches*, 2018. Available at: <https://arxiv.org/abs/1802.03098> (accessed 5 September 2019).
8. Lucas B., Kanade T. An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*, Vancouver, 1981, vol. 2, pp. 674–679.
9. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, 2019. Available at: <http://openaccess.thecvf.com/CVPR2019.py> (accessed 5 September 2019).

Информация об авторе

Залеский Борис Андреевич – д-р физ.-мат. наук, заведующий лабораторией. Объединенный институт проблем информатики НАН Беларуси (ул. Сурганова, 6, 220012, Минск, Республика Беларусь). E-mail: zalesky@newman.bas-net.by.

Information about the author

Zalesky Boris A. – D. Sc. (Physics and Mathematics), Head of the Laboratory. United Institute of Informatics Problems of the National Academy of Sciences of Belarus (6, Surganov Str., 220012, Minsk, Republic of Belarus). E-mail: zalesky@newman.bas-net.by.