

УДК 004.932.2

## МЕТОД ОБНАРУЖЕНИЯ ОСТАВЛЕННЫХ ПРЕДМЕТОВ ПО ПОСЛЕДОВАТЕЛЬНОСТИ ДАЛЬНОСТНЫХ ИЗОБРАЖЕНИЙ\*

В. А. Куликов, Е. В. Иванова

*Институт автоматки и электрометрии СО РАН,  
630090, г. Новосибирск, просп. Академика Коптюга, 1  
E-mail: kulikov.victor@gmail.com*

Предложен новый метод обнаружения оставленных предметов, основанный на анализе последовательности дальностных изображений, и созданы алгоритмы определения квазистационарного фона и оставленных объектов, работающие в реальном времени. Проведено сравнение эффективности предложенных алгоритмов с алгоритмом на основе анализа яркостных изображений.

*Ключевые слова:* детектор оставленных предметов, дальностные изображения, оценка фона.

**Введение.** Поиск оставленных предметов является одной из ключевых задач систем безопасности антитеррористического назначения. Существующие методы, основанные на анализе последовательностей яркостных изображений [1–6], имеют ряд недостатков: потеря объекта при длительных перекрытиях, дублирование объекта при его перемещении внутри сцены, проблемы обнаружения объектов, слабоконтрастных с фоном, и чувствительность к изменению освещения. Дублирование объектов обычно возникает из-за невозможности отличить оставленный объект от исчезнувшего. Эта проблема частично решена в работе [5], где исследуется связь между пикселями объекта и окружающими пикселями фона. Проблема слабых теней в задачах обнаружения объекта решена в случаях [5, 7], когда в качестве модели фона используется мультимодальная смесь гауссоид. Всё перечисленное делает задачу обнаружения оставленных предметов громоздкой и требует значительных вычислительных мощностей.

В настоящее время широкую популярность приобретают сенсоры, позволяющие получать дальностные изображения, элементы которых содержат значения расстояния от объекта на сцене до сенсора.

Применение дальностных изображений даёт возможность создать алгоритм обнаружения оставленных предметов, не зависящий от воздействия теней, слабой контрастности, а также незначительных изменений освещения. Существующие программные решения для обработки дальностных изображений [8, 9] предназначены в первую очередь для трёхмерной реконструкции сцены и интерфейса пользователь—компьютер, поэтому исследование использования этих сенсоров в научных целях и системах видеонаблюдения остаётся актуальным.

Процесс обработки дальностных изображений осложняется шумом, зависящим от значения пикселя, неопределёнными значениями точек из-за окклюзий и отражений, сильными вариациями дальности на границах объектов. Характеристики данных недостатков зависят от метода определения дальности сенсором.

---

\*Работа выполнена при поддержке Сибирского отделения РАН (междисциплинарный проект «Исследование генетических и нейрофизиологических механизмов действия нейротрофического фактора мозга и его синтетических агонистов» 2012–2014 гг. (грант № 54)).

В предложенной работе для получения дальностных изображений применялся сенсор "Microsoft Kinect" и моделировалась трёхмерная сцена программой Blender [10].

Целью данной работы является создание алгоритма с использованием анализа последовательности дальностных изображений для обнаружения оставленных предметов в реальном времени.

**Постановка задачи.** Необходимо по регистрируемой последовательности дальностных изображений  $I_t(x, y)$ , где  $(x, y)$  — координаты пикселей изображения ( $x = \overline{1, W}$ ,  $y = \overline{1, H}$ ,  $W$  и  $H$  — ширина и высота кадра), индекс  $t$  — номер текущего кадра анализируемой сцены, установить наличие оставленных предметов. Если дальность в элементе изображения не определена, то его значение будет равно нулю. Предполагаем, что сцена состоит из стационарного фона, движущихся объектов и оставленных предметов. Под оставленным предметом будем понимать связанный компонент изображения, не являющийся фоном, площадь которого лежит в заданном интервале  $(S_{\min}, S_{\max})$  пикселей, и его координаты не изменяются в  $k$  последовательных кадрах. Для удобства сравнения предлагаемого алгоритма с алгоритмом из [1] была введена следующая структура маркировки точек изображения:

$$K_t(x, y) = \begin{cases} 1 & \text{(принадлежит оставленному предмету),} \\ 0 & \text{(принадлежит фону),} \\ -1 & \text{(принадлежит движущемуся объекту).} \end{cases}$$

Алгоритм выделения оставленных объектов состоит из двух этапов: вычисления фона и анализа объектов в сцене.

**Алгоритм построения фона на дальностных изображениях.** Основная идея построения модели фона заключается в накоплении пикселей изображения с наибольшим значением дальности с учётом шума и неопределённого значения дальности в некоторых элементах изображения. Обозначим  $Bg(x, y)$  фоновое изображение, накапливаемое в течение исполнения программы или по ограниченному количеству кадров. Результат выделения фона сохраняется в виде бинарной маски

$$O_t(x, y) = \begin{cases} 1 & \text{(передний план),} \\ 0 & \text{(фон).} \end{cases}$$

Шаг 0. Если фоновое изображение отсутствует, то в качестве первого приближения фона берётся первое изображение  $Bg(x, y) = I_1(x, y)$ , на выход поступает маска, у которой все точки помечены как фон.

Шаг 1. Для последующих кадров организуется цикл по всем точкам фона. Если  $Bg(x, y) = 0$  и  $I_t(x, y) \neq 0$ , то запускается процедура обновления фона.

Шаг 2. Для исключения заполнения фона неизвестными точками, принадлежащими объекту, используется величина близости значений точки текущего изображения и ненулевых точек в окрестности неизвестной точки фона:

$$|I_t(x, y) - Bg(x + a, y + b)| < \theta,$$

где  $a, b \in [1, -1]$ , а  $\theta$  — заранее заданный порог. Если это условие выполняется хоть для одной точки из окрестности, то такая точка изображения встраивается в фон  $Bg(x, y) = I_t(x, y)$  и  $O_t(x, y) = 0$ , иначе она считается передним планом и  $O_t(x, y) = 1$ .

Шаг 3. Если значение фона в точке известно:  $Bg(x, y) \neq 0$ , то проверяется неравенство

$$Bg(x, y) - I_t(x, y) > \theta.$$

Если неравенство выполняется и  $I_t(x, y) \neq 0$ , то данная точка характеризуется как перед-

ний план ( $O_t(x, y) = 1$ ), иначе запускается процедура проверки принадлежности полученной точки фону (шаг 2).

**Алгоритм определения оставленных объектов.** Необходимо принять решение: является ли объект, найденный на изображении, оставленным предметом с учётом того, что оставленные объекты могут быть перекрыты другими объектами.

Возможны две стратегии: быстрая стратегия с низкими затратами памяти, предусмотренная для сцен с небольшим количеством движения, и точная стратегия, позволяющая отличать оставленные объекты друг от друга и надёжно обнаруживать оставленный объект в потоке людей. Эти стратегии различаются только методом подсчёта кадров с объектом.

*Быстрая стратегия определения оставленных объектов.* Введём для каждого элемента изображения счётчик кадров

$$N_t(x, y) = \begin{cases} N_t(x, y) + 1, & \text{если } O_t(x, y) \neq 0, \\ 0 & \text{иначе,} \end{cases}$$

т. е. если значение элемента принадлежит переднему плану, то счётчик кадров увеличивается, иначе он обнуляется.

*Точная стратегия определения оставленных объектов.* Для этой процедуры требуется хранение истории из  $l$  кадров ( $l \gg k$ ). Надо определить количество кадров из истории, на которых были отмечены пиксели оставленного объекта из текущего кадра:

$$N_t(x, y) = \sum_{i=t-l}^{t-1} \begin{cases} 1, & \text{если } |I_t(x, y) - I_i(x, y)| < \theta \text{ и } O_t(x, y)O_i(x, y) \neq 0, \\ 0 & \text{иначе.} \end{cases}$$

Находим разность значений двух пикселей, принадлежащих переднему плану. Если она меньше заданного порога  $\theta$ , то считаем, что пиксель принадлежит одному и тому же объекту в моменты времени  $t$  и  $i$ . Очевидно, что программную реализацию данной стратегии можно значительно упростить за счёт рекуррентного подсчёта разницы между соседними кадрами.

После того как счётчик кадров для элемента превышает заранее заданный порог  $k$ , считаем элемент изображения принадлежащим оставленному объекту:

$$K_t(x, y) = \begin{cases} 1, & N_t(x, y) > k, \\ 0 & \text{иначе.} \end{cases}$$

Далее оценивается площадь связанных компонентов на изображении  $K_t(x, y)$  и принимается решение: является ли данный объект оставленным предметом или движущимся объектом.

**Результаты вычислительного эксперимента.** Для сравнения различных методов оценки оставленных предметов в программе Blender [10] созданы три последовательности кадров одной сцены (рис. 1): яркостная, дальностная и эталонная, содержащая только оставленный объект. Вся сцена была ограничена пространством в 6 м по глубине.

Применяя различные уровни шума к дальностным изображениям, авторы сравнивали результат работы предложенного алгоритма с истинным изображением оставленного предмета. Для всей последовательности измерялись следующие показатели: количество пикселей, ложно классифицированных как оставленный объект, и количество пикселей оставленного объекта, не замеченных программой. Также измерялась скорость работы программы.

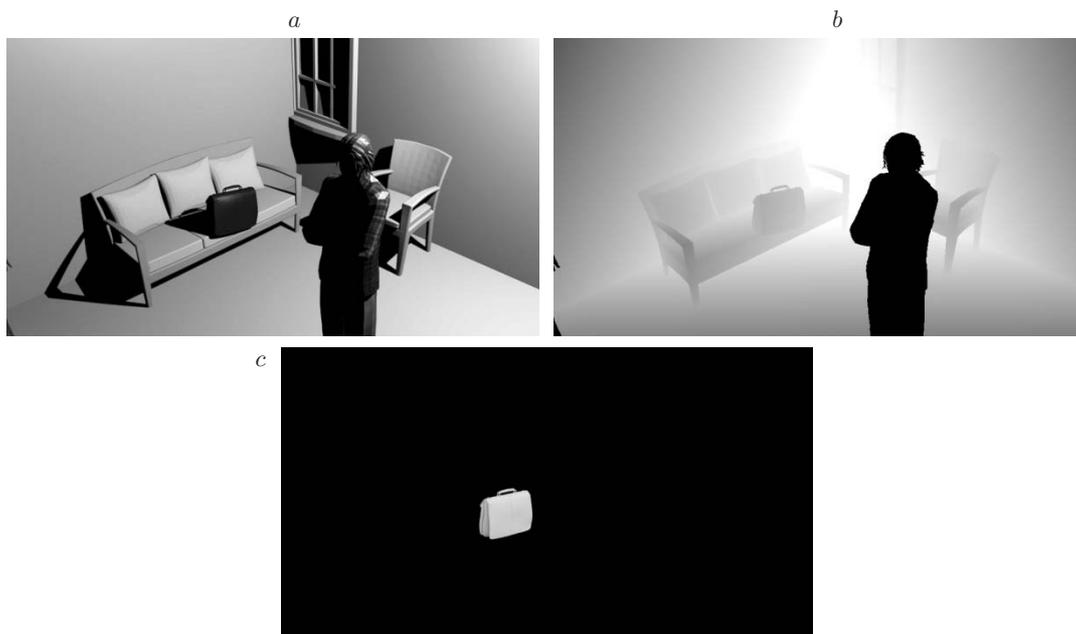


Рис. 1. Последний кадр сгенерированной тестовой последовательности: *a* — яркостное изображение, *b* — дальностное изображение, *c* — изображение, содержащее только оставленный предмет

Моделирование шума на дальностных изображениях состояло из двух этапов: моделирования шума приёмника и моделирования точек, в которых глубина не определяется.

При моделировании шума приёмника использовалось нормальное распределение с дисперсиями  $\sigma = 0,01, 0,02$  и  $0,09$  м. Так как точность измерения падает в зависимости от расстояния до объекта, применялась модель  $\sigma I_t(x, y)$ . Другими словами, дисперсия в точке зависит от значения дальности в этой точке: чем больше расстояние, тем больше дисперсия. При данной модели шума оптимально выбирать порог бинаризации  $\theta > 6\sigma$  или вычислять его в зависимости от яркости точки изображения.

Для моделирования точек, в которых глубина не определена из-за эффекта окклюзии, брались точки, лежащие на краях объектов. Для вычисления краёв применялся алгоритм Кенни [11]. Далее все точки из изображения краёв и их окрестности были обнулены на исходном глубинном изображении. Следует отметить, что последовательное использование нормального шума и моделирование окклюзий создают эффект, наблюдаемый в сенсоре "Microsoft Kinect", где количество неопределённых точек зависит от глубины (рис. 2).



Рис. 2. Зашумлённое дальностное изображение (чёрным отмечены пиксели с неопределённой глубиной)

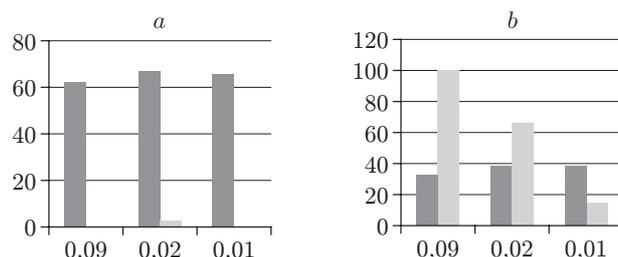


Рис. 3. Оценка работы алгоритма: с порогами 0,09 м (а) и 0,02 м (b) (тёмные столбцы — пропуски, светлые — ложные срабатывания)

**Оценка точности предложенного алгоритма в зависимости от уровня шума.** Тестирование устойчивости алгоритма с разными порогами  $\theta$  к шуму с различными дисперсиями (0,09, 0,02, 0,01 м) дало возможность получить в процентных отношениях оценки пропуска точек объекта и ложных тревог. За 100 % взята площадь оставленного объекта из эталонных изображений. В ходе тестирования установлены пределы использования предложенной методики, а также ряд закономерностей, позволяющих подбирать порог в зависимости от задачи (рис. 3). При больших уровнях шума порог на отсечение переднего плана (0,02 м) обнаруживает большое число ложных объектов. Пропуски части пикселей оставленных объектов играют менее критичную роль, так как обычно находится только часть оставленного объекта. На рис. 3, b столбец при значении 0,09 был выровнен для визуализации на 100 %, на самом деле ошибка составляла значительно больший процент площади.

**Сравнение предложенного алгоритма с основанным на яркостных изображениях [1]** производилось на реальных данных. На квазистационарный фон был помещён оставленный предмет; после работы алгоритмов контур оставленного предмета подчёркивается белой линией (рис. 4). Изображение на рис. 4, b является глубинным изображением с наложенной цветной текстурой. Чёрным отмечены области с неопределённой глубиной. Предложенный метод имеет преимущества в ситуациях, когда происходит дублирование объекта после перемещения внутри сцены, при длительных окклюзиях и при низком контрасте яркости между объектом и фоном. Основным недостатком метода при использовании сенсора "Microsoft Kinect" является невозможность применения его вне помещений. Также этот сенсор плохо работает на отражающих поверхностях. Производительность измеря-

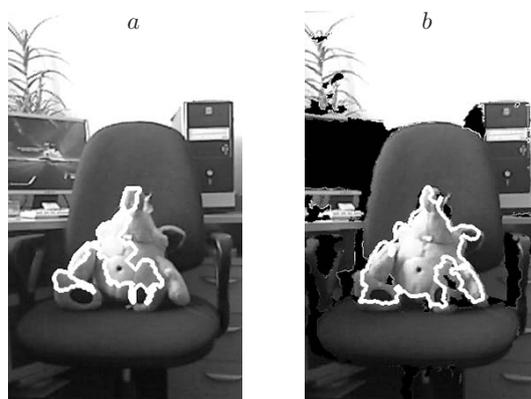


Рис. 4. Результат работы алгоритма выделения оставленного предмета: а — по последовательности цветных изображений алгоритмом из [1], b — предложенным методом по дальностным изображениям

лась на компьютере с процессором AMD Athlon 64 X2 Dual Core 5400+ (частота 2,81 ГГц и оперативная память 1,87 Гбайт). Программа была собрана в режиме отладки на компиляторе GCC. Время обработки одного кадра размером 640 × 480 пикселей составляло 16 мс.

**Заключение.** В данной работе предложено использование сенсора глубины сцены для обнаружения оставленных предметов и создан итеративный алгоритм оценки фона, обнаруживающий оставленные предметы по последовательности дальностных изображений, устойчивый к длительным окклюзиям и избавляющий от тиражирования объекта при его перемещении на изображении. Показана зависимость характеристик обнаружения от шума на серии смоделированных изображений. Предложенный алгоритм прост в реализации и может обрабатывать поток дальностных изображений в режиме реального времени.

### СПИСОК ЛИТЕРАТУРЫ

1. **Иванов В. А., Киричук В. С., Орлов С. И.** Пространственно-временной анализ видео-последовательностей для обнаружения оставленных предметов // *Автометрия*. 2011. **47**, № 1. С. 30–36.
2. **Porikli F., Ivanov Y., Hoga T.** Robust abandoned object detection using dual foregrounds // *EURASIP Journ. Adv. Sign. Proc.* 2008. N 1. 197875.
3. **Beynon M., van Hook D., Seibert M. et al.** Detecting abandoned packages in a multi-camera video surveillance system // *Proc. of the IEEE Conf. Advanced Video and Signal Based Surveillance (AVSS'03)*. 21–22 July, 2003. Miami, USA. P. 221–228.
4. **Spagnolo P., Caroppo A., Leo M. et al.** An abandoned/removed object detection algorithm and its evaluation on PETS datasets // *IEEE Intern. Conf. Advanced Video and Signal Based Surveillance (AVSS'06)*. 22–24 November, 2006. Sydney, Australia. P. 17.
5. **Tian Y., Feris R., Hampapur A.** Real-time detection of abandoned and removed objects in complex environments // *IEEE Intern. Workshop on Visual Surveillance (in conjunction with ECCV'08)*. Marseille, France, 2008.
6. **Bird N., Atev S., Caramelli N. et al.** Real time, online detection of abandoned objects in public areas // *Proc. of the IEEE Intern. Conf. Robotics and Automation*. May, 2006. Orlando, USA. P. 3775–3780.
7. **Chandran V., Denman S., Sridharan S.** Abandoned object detection using multi-layer motion detection // *Proc. of the Intern. Conf. on Signal Processing and Communication Systems*. Queensland, Australia, 2007. P. 439–448.
8. **Rusu R. B., Cousins S.** 3D is here: Point cloud library (PCL) // *Proc. of the IEEE Intern. Conf. on Robotics and Automation*. May 9–13, 2011. Shanghai, China. P. 1–4.
9. **OpenNI** — certifying organization. URL: <http://www.openni.org/> (дата обращения: 9.08.2011).
10. **Powell A. W.** Blender 2.5. Lighting and Rendering. Birmingham, UK: Packt Publishing, 2010. 252 p.
11. **Canny J.** A computational approach to edge detection // *IEEE Trans. Pattern Anal. and Machine Intell.* 1986. **8**, N 6. P. 679–698.

*Поступила в редакцию 9 августа 2011 г.*