

Об алгоритмах компьютерного зрения для поиска порывов в сетчатых ограждающих конструкциях*

А. С. СМАГИН^{1,2}, К. Н. ДУБРОВИН^{1,†}

¹Вычислительный центр ДВО РАН, Хабаровск, Россия

²Институт горного дела ДВО РАН, Хабаровск, Россия

[†]Контактный e-mail: nobforward@gmail.com

Рассмотрены вопросы автоматизированного мониторинга состояния сетчатых ограждающих конструкций, которые используются на предприятиях, производящих морские биоресурсы. Предложен алгоритм выделения порывов сетчатых ограждений в подводных условиях с применением методов компьютерного зрения, который реализован в виде комплекса программ на языке Python. Приведены результаты работы программного комплекса. Показано, что методы компьютерного зрения эффективно справляются с определением целостности ячеек сети на слабо- и среднезашумленных изображениях. Для работы в более сложных оптических условиях в состав программного комплекса предложено включить нейросетевой модуль.

Ключевые слова: алгоритм, компьютерное зрение, выделение границ, машинное обучение, программный комплекс.

Библиографическая ссылка: Смагин А.С., Дубровин К.Н. Об алгоритмах компьютерного зрения для поиска порывов в сетчатых ограждающих конструкциях // Вычислительные технологии. 2019. Т. 24, № 6. С. 117–124.
DOI: 10.25743/ICT.2019.24.6.014.

Введение

Объектом исследования являются специализированные сетчатые ограждения, используемые для производства морских биоресурсов на предприятиях аквакультурного направления. Цель данной работы — разработка и исследование алгоритмов обработки изображений для системы компьютерного зрения автономного необитаемого подводного аппарата (АНПА), позволяющих автоматизировать процесс мониторинга состояния сетчатого ограждения и тем самым исключить нахождение человека в условиях агрессивной для него подводной среды.

Искусственные сетчатые ограждения на кадрах видеопотока, получаемого камерами АНПА, чаще всего слабо контрастируют с окружающим фоном. Кроме того, эти видеок кадры содержат различные шумы. Они возникают при недостаточной освещенности, оптических искажениях в водной среде и особенно при появлении в кадре разнообразных представителей подводной флоры, фауны, ила и подводного мусора.

*Title translation and abstract in English can be found on page 124.

© ИВТ СО РАН, 2019.

Одной из основных задач обработки и анализа изображений является сегментация, т. е. разделение изображения на области, для которых выполняются определенные критерии однородности (таким критерием может быть яркость). Основным классический метод сегментации — пороговая обработка изображения. Это одна из наиболее простых и важных процедур поэлементных преобразований, которая почти всегда предшествует процессу анализа и распознавания изображений [1]. Операция заключается в сопоставлении значения яркости каждого пикселя изображения с заданным значением порога. Выбор порогового значения дает возможность выделения на изображении областей определенного вида. Использование предварительной пороговой локальной фильтрации и бинаризации позволяет отсечь посторонние шумы и уже на “чистом” изображении быстро определить пространственные характеристики ячеек сетки с целью обнаружения порывов и деформаций сетчатого ограждения.

1. Фильтрация изображений

Точная цветопередача, ее улучшение, восстановление цветовой насыщенности на фото- и видеоизображениях играют важную роль в системах компьютерного зрения. Проблемы с цветопередачей возникают при съемке с недостаточно хорошим освещением либо при наличии искажений. Ряд артефактов, таких как недостаток или избыток света, затрудняют обработку изображений с целью выделения границ интересующих объектов [2].

В разработанном программном комплексе поиск границ сетчатого ограждения выполняется двухэтапным алгоритмом предобработки изображений. На первом этапе с применением алгоритма Retinex осуществляется компенсация артефактов, связанных с освещением [3, 4].

Алгоритм Retinex основан на принципах имитации устройства рецепторов сетчатки глаза человека. На первом шаге алгоритма осуществляется разделение исходного изображения I на RGB-каналы $C_i, i = 1, 2, 3$. Затем каждый RGB-канал исходного изображения разделяется на две компоненты: высокочастотную R -компоненту (reflectance image), содержащую детали изображения, и низкочастотную L -компоненту (illumination image), содержащую фон изображения. Значение каждого пикселя канала с координатами (x, y) представляется в виде произведения соответствующих пикселей L - и R -компонент [5]:

$$C_i(x, y) = L_i(x, y) \times R_i(x, y).$$

Для выделения L -компоненты используется низкочастотный фильтр Гаусса:

$$L_i(x, y) = C_i(x, y) \otimes G(x, y),$$

где $G(x, y) = \frac{1}{\sqrt{2\pi}c} e^{-(x^2+y^2)/2c^2}$ — фильтр Гаусса с размерностью ядра c ; (x, y) — координаты текущего пикселя; \otimes — оператор свертки.

На четвертом, последнем, шаге из натурального логарифма каждого цветового канала C_i вычитается натуральный логарифм выделенной L -компоненты:

$$\acute{C}_i(x, y) = \ln C_i(x, y) - \ln L_i(x, y),$$

где $\acute{C}_i(x, y)$ — i -я компонента выходного изображения.

После применения алгоритма Retinex основная часть значений пикселей каналов принадлежит диапазону $[-1, 1]$. Для визуализации изображения значения пикселей нормализуются по формуле

$$S_i(x, y) = \dot{C}_i(x, y) \times 255 + 127.5,$$

где S_i — i -я компонента итогового изображения.

Используемая в программном комплексе модификация MultiScale Retinex with Color Restoration имитирует биологические механизмы глаза, который приспособляется к сложным условиям, и хорошо справляется с задачей повышения яркости и контрастности изображения, проявляя скрытые из-за слабой освещенности и искажений детали изображения.

В MultiScale Retinex выходное значение цветового канала MSR_i является взвешенной суммой выходных значений Retinex с разными размерностями ядра фильтра Гаусса:

$$MSR_i(x, y) = \sum_{n=1}^N w_n s_{ni}(x, y),$$

где $s_{ni}(x, y)$ — значение i -й компоненты пикселя (x, y) после применения алгоритма Retinex, w_n — вес n -го значения Retinex. Color Restoration (восстановление цвета) подразумевает умножение значений всех пикселей каждого цветового канала, полученного в MultiScale Retinex, на величину $CR_i(x, y)$, определяемую следующим образом:

$$CR_i(x, y) = b \times \left[\ln(a MSR_i(x, y)) - \ln \left(\sum_{i=1}^3 MSR_i(x, y) \right) \right].$$

Здесь $a = 46$, $b = 25$.

На рис. 1 представлены исходное изображение, снятое под водой [6], и результат работы алгоритма на первом этапе предобработки. Отчетливо видно улучшение контрастности и яркости в темных областях изображения без потери качества в светлых областях.

На втором этапе изображение переводится в градации серого и бинаризуется с использованием порогового фильтра Отсу (Otsu) [7]. Рассмотрим, как работает этот фильтр. Пусть пиксели изображения имеют L уровней яркости, h_i — число элементов изображения, имеющих яркость i , H — общее число пикселей изображения. Тогда $p(i) = h_i/H$ — частота встречаемости пикселей, имеющих яркость i , где $i = 0, 1, \dots, L-1$. Значения яркостей пикселей изображения можно интерпретировать как случайные величины, а их гистограмму — как оценку плотности распределения вероятностей. Задаются начальное значение $t = 1$.

Пиксели изображения делятся на два класса: класс B , в который попадают пиксели, яркость которых меньше t , и класс O , куда попадают пиксели с яркостью не менее t . Далее вычисляются математические ожидания и дисперсии каждого класса:

$$\mu_B(t) = \sum_{i=0}^{t-1} ip(i), \quad \mu_O(t) = \sum_{i=t}^{L-1} ip(i),$$

$$\sigma_B^2(t) = \sum_{i=0}^{t-1} (i - \mu_B(t))^2 p(i), \quad \sigma_O^2(t) = \sum_{i=t}^{L-1} (i - \mu_O(t))^2 p(i).$$

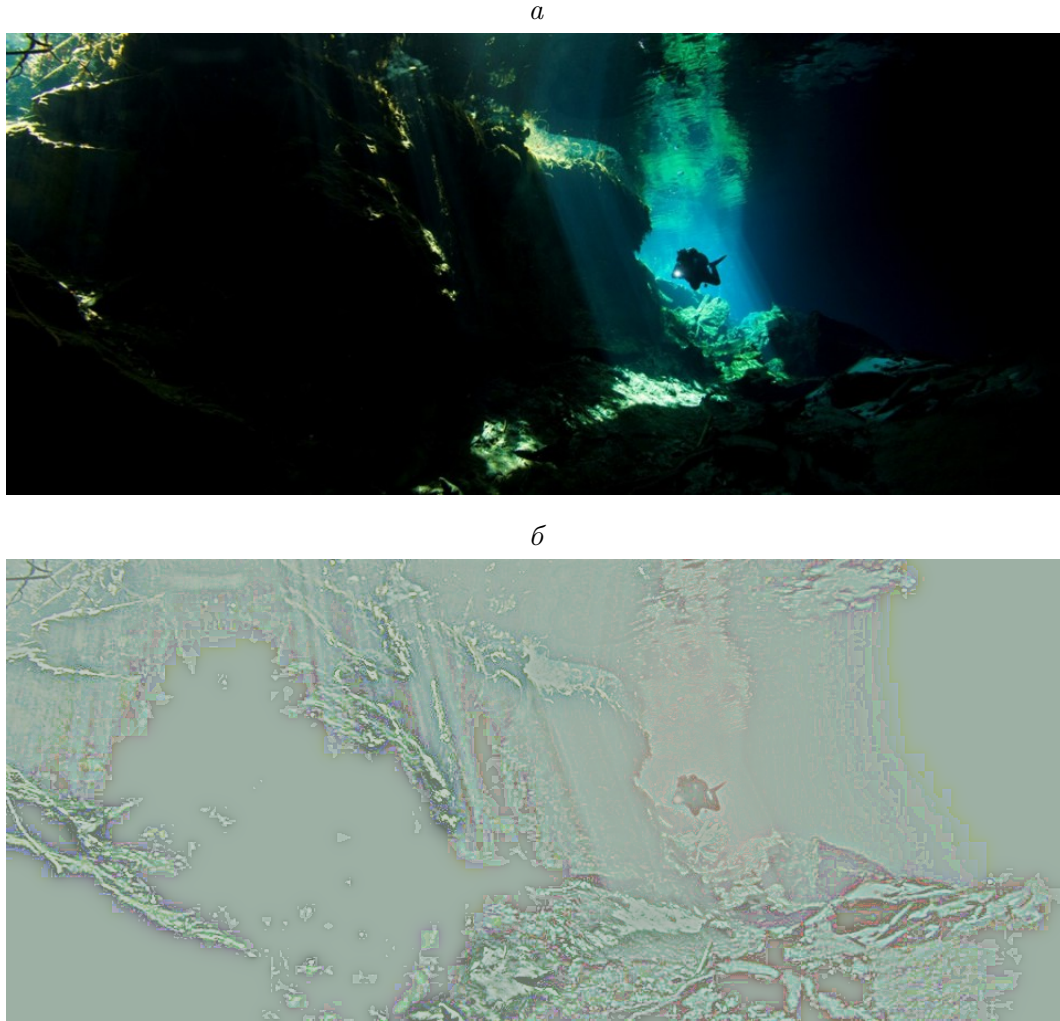


Рис. 1. Оригинал изображения (*a*) и результат работы алгоритма Multi-Scale Retinex with Color Restoration (*б*)

Внутриклассовая дисперсия определяется как взвешенная сумма дисперсий двух классов:

$$\sigma_w^2(t) = n_B(t)\sigma_B^2(t) + n_O(t)\sigma_O^2(t),$$

где $n_B(t) = \sum_{i=0}^{t-1} p(i)$ — доля пикселей, отнесенных к классу B ; $n_O(t) = \sum_{i=t}^{L-1} p(i)$ — доля пикселей, отнесенных к классу O .

Путем перебора всех возможных порогов $t = 1, 2, \dots, L$ можно определить оптимальный порог для разделения изображения на “полезную” и “фоновую” части, для которого σ_w^2 принимает наименьшее значение. Вычисление статистик второго порядка (дисперсий) для каждого значения t требует большого количества вычислений, поэтому Отсу предложил более простой способ, заключающийся в вычислении другой статистики — межклассовой дисперсии:

$$\sigma_M^2(t) = n_B(t)n_O(t)(\mu_O(t) - \mu_B(t))^2.$$

Вычисление $\sigma_M^2(t)$ требует вычисления только статистик первого порядка (математических ожиданий). Оптимальное значение t соответствует максимуму межклассовой дисперсии.

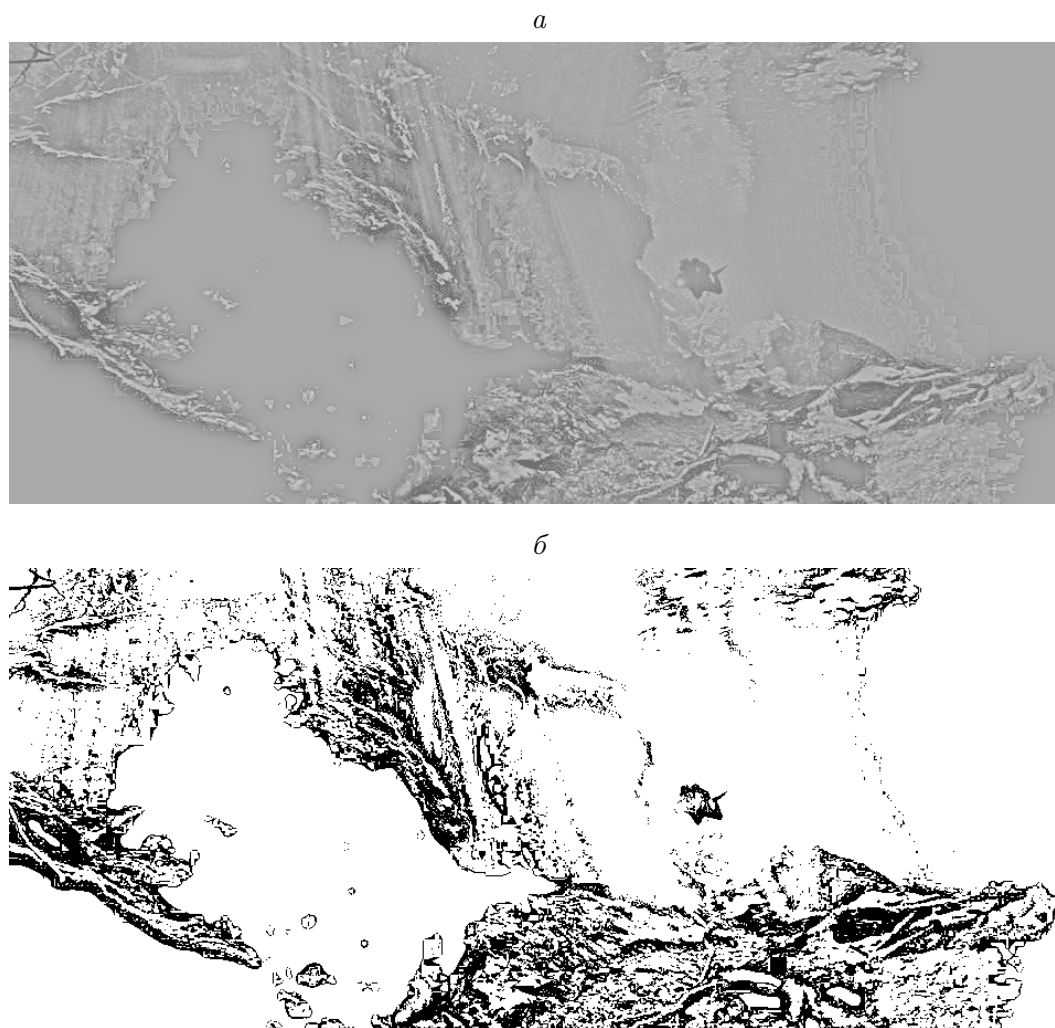


Рис. 2. Результаты перевода в градации серого (а) и применения порогового фильтра Отсу (б)

Поскольку дисперсия является мерой разброса уровней яркости вокруг среднего значения, большая ее величина свидетельствует о значительном отклонении от среднего. В связи с этим операция нахождения максимума для определения порога означает увеличение изолированности двух классов на бинаризованном изображении [8]. При оптимальном значении порога t^* пиксели, попавшие в класс B (лежащие ниже порога), принимают нулевое значение яркости (черный цвет) и обозначают содержательную часть изображения — результат фильтрации методом Отсу. Выбор именно этого фильтра обусловлен тем, что для него порог рассчитывается автоматически для каждого изображения и метод содержится в библиотеке компьютерного зрения OpenCV. На рис. 2 представлены изображение в градациях серого после первого этапа фильтрации и результат бинаризации по методу Отсу.

2. Определение порывов и границ сетчатого ограждения

Проверка целостности сетчатого ограждения осуществляется после завершения фильтрации и получения его незашумленного изображения. На этом этапе производится поиск всех контуров на изображении при помощи функции FindContours библиотеки

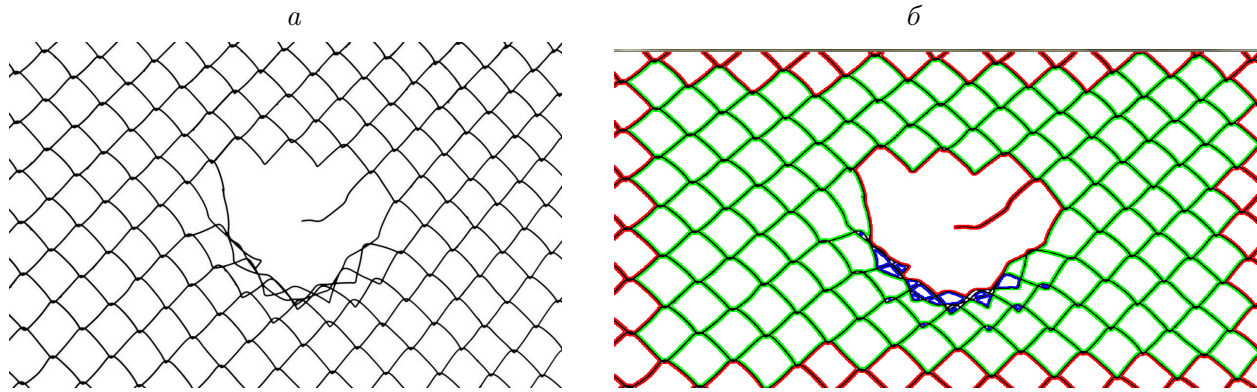


Рис. 3. Фильтрованное изображение (а) и результат определения целостности сети (б)

ки OpenCV, затем выполняется их классификация. Если контур является замкнутым многоугольником и его периметр находится в диапазоне от $0.5P$ до $2P$, где P — средний периметр, то этот контур классифицируется как целая ячейка сети и помечается зеленым цветом. Незамкнутый контур или замкнутый контур с периметром более $2P$ помечается красным цветом и классифицируется как порыв. Замкнутый контур с малым периметром (менее $0.5P$) воспринимается системой как сильно деформированный (для таких ячеек велика вероятность порыва в будущем) и помечается синим цветом. На рис. 3 представлен результат работы программы, определяющей целостность сети.

Заключение

Анализ результатов работы программного комплекса показал, что он успешно справляется с классификацией ячеек сети на не сильно зашумленных изображениях. Однако преобразований, осуществляемых на этапе предобработки, недостаточно для полного отсеечения посторонних объектов. Например, силуэт человека, который находится за сеткой (рис. 2, б), сливается с ней, из-за чего возможно обнаружение ложных порывов и деформаций ограждения. В связи с этим работа по усовершенствованию и расширению функционала программного комплекса будет продолжена. Конечным результатом исследования станет программный комплекс с нейросетевым модулем для улучшенной фильтрации внешнего фона и более эффективного обнаружения проблемных участков сетчатого ограждения.

Благодарности. Исследование выполнено при финансовой поддержке РФФИ (грант № 19-37-90046). При выполнении исследования были использованы ресурсы ЦКП “Центр данных ДВО РАН”.

Список литературы / References

- [1] Гонсалес Р., Вудс Р. Цифровая обработка изображений: Пер. с англ. М.: Техносфера, 2005. 1072 с.
Gonzalez, R.C., Woods, R.E. Digital image processing. Prentice Hall, 2007. 976 p.
- [2] Прэйт У. Цифровая обработка изображений: Пер. с англ. М.: Мир, 1982. 790 с.
Pratt, W.K. Digital image processing. CRC Press, 1978. 756 p.

- [3] **Zhang, S., Wang, T., Dong, J., Yu, H.** Underwater image enhancement via extended multi-scale Retinex // Neurocomputing. 2017. Vol. 245. P. 1–9.
- [4] **Yuen Peng Loh, Chee Seng Chan.** Getting to know low-light images with the Exclusively Dark dataset // Computer Vision and Image Understanding. 2019. Vol. 178. P. 30–42.
- [5] **Doo Hyun Choi, Ick Hoon Jang, Mi Hye Kim.** Color image enhancement using single-scale retinex based on an improved image formation model // 16th Europ. Signal Proc. Conf. (EUSIPCO 2008), Lausanne, Switzerland, August 25–29, 2008, copyright by EURASIP. Available at: <https://www.eurasip.org/Proceedings/Eusipco/Eusipco2008/papers/1569104826.pdf>
- [6] Coral Latvia. Available at: <http://crl.lv/ru/zemudens-skaistums/nggallery/page/3>
- [7] **Otsu, N.A.** Threshold selection method from gray-level histograms // IEEE Transactions on Systems, Man, and Cybernetics. 1979. Vol. 9, No. 1. P. 62–66.
- [8] **Radke, R.J., Andra, S., Al-Kofahi, O., Roysam, B.** Image change detection algorithms: a systematic survey // IEEE Trans. Imag. Process. 2005. Vol. 14(3). P. 294–307.
- [9] **Sorokin, A.A., Makogonov, S.V., Korolev, S.P.** The information infrastructure for collective scientific work in the Far East of Russia // Scientific and Technical Inform. Proc. 2017. Vol. 44, No. 4. P. 302–304.

Поступила в редакцию 30 сентября 2019 г.

On computer vision algorithms for searching breaks in meshed fencing constructions

SMAGIN, ALEXEY S.^{1,2}, DUBROVIN, KONSTANTIN N.^{1,*}

¹Computing Center of FEB RAS, Khabarovsk, 680000, Russia

²Mining Institute of FEB RAS, Khabarovsk, 680000, Russia

*Corresponding author: Dubrovin, Konstantin N., e-mail: nobforward@gmail.com

Purpose. The paper addresses image processing algorithms for the computer vision system of an autonomous uninhabited underwater vehicle, which automatically monitors the state of the mesh fence and thereby exclude the presence of a person in an aggressive underwater environment.

Methodology. The MultiScale Retinex with Color Restoration algorithm and the Otsu method were implemented using the Python programming language to pre-process and filter the image. Methods from OpenCV computer vision library were used to detect damage to the mesh fence.

Results. An algorithm is proposed for highlighting the impulses of mesh fencing in underwater conditions using computer vision methods implemented by the Python software. The software implementation results are provided. It is shown that computer vision methods effectively cope with determining the integrity of network cells in weakly and medium-noisy images. To work in more complex optical conditions, it is proposed to include a neural network module in the software package.

Findings. The analysis of the results of the software package showed that it successfully copes with the classification of network cells in clean images. However, the transformations carried out at the pre-processing stage are not enough to complete

noise elimination. In this regard, this study will continue to improve and expand the functionality of the software package. The result of the study will be a software package with a neural network module for full filtering of the external background and efficient detection of the mesh fence problem areas.

Keywords: algorithm, computer vision, edges detection, machine learning, software.

Cite: Smagin, A.S., Dubrovin, K.N. On computer vision algorithms for searching breaks in meshed fencing constructions // Computational Technologies. 2019. Vol. 24, No. 6. P. 117–124. (In Russ.) DOI: 10.25743/ICT.2019.24.6.014.

Acknowledgements. The reported study was funded by RFBR (grant No.19-37-90046). Resources of SAC “FEB RAS Data center” were used in this study.

Received September 30, 2019