

УДК 004.932

**ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ БАНКОВСКИХ КАРТ  
ДЛЯ РАСПОЗНАВАНИЯ ИХ РЕКВИЗИТОВ  
С ИСПОЛЬЗОВАНИЕМ МОБИЛЬНЫХ УСТРОЙСТВ ПОД УПРАВЛЕНИЕМ iOS**

**А.Ю. КУРИЛОВИЧ**

*(Представлено: канд. техн. наук, доц. Р.П. БОГУШ)*

*Предложен алгоритм предварительной обработки изображений банковских карт для мобильных устройств под управлением iOS, который включает такие этапы, как: детектирование области карты; сегментация изображения банковской карты; преобразование цветных сегментов к оттенкам серого; улучшение контрастности; подчеркивание границ символов с использованием операций математической морфологии; уточнение границ сгруппированных блоков символов. Выходными данными алгоритма являются также координаты уточненной области символов на изображении и используемый на карте язык.*

В настоящее время большинство приложений мобильного банкинга требуют ручного ввода реквизитов банковской карты (данных всех информационных полей, нанесенных на карту) в систему для их наглядного представления на экране смартфона и совершения, в последующем, расчетных операций. Процесс заполнения каждого из этих полей в мобильных приложениях требует от человека определенного терпения, усидчивости и особой внимательности, кроме этого занимает время. Поэтому, разработка алгоритмического и программного обеспечения распознавания реквизитов банковских карт для мобильных устройств является актуальной.

Предлагаемый алгоритм предварительной обработки изображений банковских карт для мобильных устройств под управлением iOS требует выполнения следующих шагов: детектирование области карты; сегментация изображения банковской карты; преобразование цветных сегментов к оттенкам серого; улучшение контрастности; подчеркивание границ символов с использованием операций математической морфологии; уточнение границ сгруппированных блоков символов.

В качестве входных данных алгоритма выступает полученный с камеры устройства видеопоток, разделенный во времени на кадры, представляющие собой изображения в цветовом пространстве RGB, без альфа-канала. Учитывая характеристику устройства iPhone 6 (2014 г.в.), камера обладает 8-мегапиксельной ПЗС-матрицей, вспышкой из двух светодиодов и автофокусировкой. Размеры кадра при этом могут варьироваться в зависимости от конкретной модели устройства, и достигать размеров 3840×2160 пикселей. Все устройства младше вышеуказанного обладают более совершенными характеристиками. Операционная система (выше iOS 2) способна выполнять автоматическое удержание фокуса на объекте съемки, автоматическую регулировку яркости и баланса белого на фото. Вышеперечисленное говорит о том, что изображение, поступающее напрямую от системы с фотокамеры, будет иметь оптимальные входные параметры: глубину резкости, яркость, достаточный размер кадра для передачи деталей банковской карты.

Отделение изображения банковской карты от фона осуществляется путем обработки его алгоритмами iOS, которые базируются на методе скоростного обнаружения объектов Виолы—Джонса [1] и глубоком обучении по методу OverFeat [2].

Критерием выбора контура карты из множества детектированных прямоугольников  $R$ , где количество найденных прямоугольных областей  $N$ , выступает соотношение размеров ее сторон. Поскольку размеры сторон карты ( $m_o \times n_o$ ) определены ISO/IEC 7811-1:2017, соотношение ее сторон является постоянной величиной  $\frac{n_o}{m_o}$ . Представим объект найденного прямоугольника  $r_i$  как  $R_i(m_i, n_i)$ , где  $i = 0, 1, \dots, N-1$ . Тогда относятся к регионам интереса будут прямоугольники  $r$  для которых выполняется условие (1). Точным ответствием считается прямоугольник с наибольшей длинной стороны  $n_{max}$  из всех отображенных.

$$\frac{n_i}{m_i} = \frac{n_o}{m_o} \quad (1)$$

Размеры прямоугольной области  $r_o(m_{max}, n_{max})$  и ее расположение относительно целого изображения  $p_o(x_o, y_o)$  используются для извлечения области карты из оригинального кадра.

Из изображения карты  $I$  с размерами  $m_{max} \times n_{max}$  (далее  $m_i \times n_i$ ) выделяются области интереса алгоритма, содержащие информацию о номере банковской карты, дате истечения срока ее действия, владельце. Размер и расположение этих областей определено ISO/IEC 7811-1:2017 и может быть записано в следующем представлении:

- область  $C(x_C, y_C, n_C, m_C)$  – номер карты;
- область  $D(x_D, y_D, n_D, m_D)$  – дата срока истечения;
- область  $E(x_E, y_E, n_E, m_E)$  – имя держателя карты.

Поскольку размеры  $m_I \times n_I$  полученного изображения карты  $I$  могут варьироваться, необходимо преобразовать  $C$ ,  $D$  и  $E$  к виду, который будет применим для их корректного отделения от изображения  $I$ . Для этого необходимо определить два коэффициента масштабирования: по ширине (2) и по высоте (3).

$$C_{scale.width} = \frac{n_I}{n_C} \tag{2}$$

$$C_{scale.height} = \frac{m_I}{m_C}, \tag{3}$$

Умножив каждый из параметров обозначенных выше областей на соответствующий ему коэффициент масштабирования получим размеры этих областей относительно размера полученного изображения карты  $I$ .

$$\begin{aligned} C_I(x_C \cdot C_{scale.width}, y_C \cdot C_{scale.height}, n_C \cdot C_{scale.width}, m_C \cdot C_{scale.height}) \\ D_I(x_D \cdot C_{scale.width}, y_D \cdot C_{scale.height}, n_D \cdot C_{scale.width}, m_D \cdot C_{scale.height}) \\ E_I(x_E \cdot C_{scale.width}, y_E \cdot C_{scale.height}, n_E \cdot C_{scale.width}, m_E \cdot C_{scale.height}) \end{aligned}$$

Отделив фрагменты областей  $C_I$ ,  $D_I$ ,  $E_I$  от изображения карты  $I$ , получим изображения:  $I_C$ ,  $I_D$ ,  $I_E$  соответственно номера карты, время истечения ее срока действия и имени владельца.

Согласно ISO/IEC 7810 - 2006 номер банковской карты содержит 16 цифр распределенных на 4 равные группы по 4 цифры в каждой. Исходя из этого выделенное на предыдущем этапе изображение  $I_C$  с областью номера карты  $C_I$ , разделяется на 4 равные области  $C_{I1}$ ,  $C_{I2}$ ,  $C_{I3}$ ,  $C_{I4}$  (4).

$$C_{Ij}(x_{C_I} + (\frac{n_{C_I}}{4} \times (j - 1)), y_{C_I}, \frac{n_{C_I}}{4}, m_{C_I}) \tag{4}$$

$j=1,2,3,4$

Выделив рассчитанные области  $C_{I1}$ ,  $C_{I2}$ ,  $C_{I3}$ ,  $C_{I4}$  на изображении  $I_C$  получим соответственно изображения  $I_{C1}$ ,  $I_{C2}$ ,  $I_{C3}$ ,  $I_{C4}$  - групп номера карты.

Следующим шагом является преобразование изображения в полутоновое. Далее для повышения яркостной разницы между контурами символов и фоном изображений применяется процедура повышения контрастности. Наиболее эффективным для данной задачи является метод нормализации гистограммы, так как выполняет растяжку не всего диапазона интенсивностей, а только его наиболее информативного участка. Это позволяет не рассматривая истинные экстремальные значения яркостей назначить условия их определения с заданной погрешностью, что усиливает эффект контрастности за счет потери шумовых областей с редко встречающимися интенсивностями [3].

Для определения цвета фона изображения и цвета символов мы вычисляем среднюю яркость изображения  $Y_{aver}$ . Если  $Y_{aver} > 127.5$  то принимаем цвет фона полученного изображения как светлый, а цвет символов как темный, в противном случае – наоборот. С использованием операций математической морфологии, осуществляется подчеркивание границ символов. Если цвет символов светлый, используется функция морфологического преобразования White Top Hat [4]. Для темных символов – функция Black Top Hat ( $T_b(I)$ ). Структурирующий элемент для ядер обоих фильтров имеет прямоугольную форму и размер  $n_b \times m_b$ , рассчитывающиеся по формуле (5), где  $n_I$ , ширина поступившего для морфологического преобразования изображения.

$$\begin{aligned} n_b &= n_I \cdot 0.06 \\ m_b &= \frac{n_b}{3} \end{aligned} \tag{5}$$

На следующем этапе выполняется бинаризация изображений с адаптивным порогом на основе анализа локального региона [5, 6]. Метод преобразует изображение в градациях серого  $I$  к монохромному изображению согласно (6).

$$bin_{x,y} = \begin{cases} g_{max}, & \text{if } Y_{xy} > T(Y_{xy}, block\_size, c) \\ 0, & \text{else} \end{cases} \tag{6}$$

где  $Y_{x,y}$  значение яркости пикселя с координатами  $x, y$  изображения  $I$ ;  $g_{max}=255$  - значение максимальной яркости;  $T(Y_{x,y}, block\_size, c)$  - адаптивный порог бинаризации, рассчитываемый индивидуально для каждого пикселя; где функция  $T$  - представляет собой весовую сумму [5] (взаимную корреляцию с окном Гаусса, Гауссианом (7) [6]) блока пикселей размером  $block\_size \times block\_size$  соседних с обрабатываемым пикселем (с координатами  $x, y$ ) минус коэффициент  $c$  [5].

$$G_i = \alpha \cdot e^{-\frac{(i - \frac{block\_size-1}{2})^2}{2 \cdot \sigma^2}} \quad (7)$$

где  $i=0..block\_size-1$ ,  $\alpha$  выбрана таким образом, что  $\sum_i G_i = 1$

Для уменьшения количества шумов, удаления из бинаризованного изображения лишних неинформативных деталей, и грубого приближения символов к изолированным закрашенным блокам, отличающимся от фона изображения - к нему применяются последовательно морфологические операции закрытия и эрозии.

Структурный элемент  $b$  ядер обоих фильтров имеет эллипсную форму, что лучше подходит при обработке символов шрифт которых, OCR-B, согласно ISO/IEC 7811-1:2017, имеет сглаженные края.

Размер ядер  $n_b \times n_b$ , где  $n_b$  - не должен превышать размера равного трети толщины символа  $w_{syml}$ , иначе появляется вероятность утери контура символа при обработке.

Толщина символа  $w_{syml}$  не зависит от типа обрабатываемого фрагмента изображения карты, и является строго определенной для OCR-B. Проекция значения ширины шрифта (мм.) на размер в пикселях относительно размера области полной банковской карты, рассчитывается на основании ее полной ширины (8), и является константой в рамках обработки текущей области карты.

$$w_{syml} = n_0 \cdot 0.004884 \quad (8)$$

Для уточнения краев области символов на обработанном изображении, используется метод скользящего вертикально окна. Высота окна варьируется от размера шрифта символов. Для банковской карты это значения определяемые ISO/IEC 7811-1:2017,  $H_{cn}$  - высота шрифта номер карты (4,0мм),  $H_{exd}$  - высота шрифта поля даты срока истечения карты (2,85мм),  $H_{mn}$  - высота шрифта поля держателя карты (2,65мм). Тогда их проекции на высоту в пикселях относительно размеров области полной карты  $h_{cn}$ ,  $h_{exd}$ ,  $h_{mn}$ . Общий вид расчета представлен формулой (9),

$$h = \frac{H \cdot m_0}{54.0} \quad (9)$$

где 54.0 мм - высота карты по ISO 7810 ID-1, а  $m_0$  - высота детектированной области карты (пкс.).

Выходными данными алгоритма являются обработанные и улучшенные изображения полей реквизитов банковской карты с параметрами: уточненная область символов, полученная на предыдущем этапе; язык.

В данной статье решена задача разработки алгоритма предварительной обработки изображений банковских карт для распознавания их реквизитов. Предложенный алгоритм представляет собой описание обработки полученного с камеры устройства кадра видеопоследовательности и включает в себя: определение контуров карты по алгоритму скоростного обнаружения объектов Виолы—Джонса и методу OverFeat; выделение областей интереса; улучшение качества символов в этих областях с помощью метода нормализации гистограммы и морфологических преобразований Top Hat; уточнение границ символов через применение бинаризации с адаптивным порогом, выполнение морфологических операций над результатами, а после поиска максимально подходящей области с помощью вертикально скользящего окна.

#### ЛИТЕРАТУРА

1. Viola, P. and Jones, M.J. Robust Real-time Object Detection Using a Boosted Cascade of Simple Features. In Proceedings of the Computer Vision and Pattern Recognition Conference, 2001.
2. Sermanet, Pierre, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, and Yann LeCun. OverFeat: Integrated Recognition, Localization and Detection Using Convolutional Networks. arXiv:1312.6229 [Cs], December, 2013.

3. Кустикова, В.Д. Разработка мультимедийных приложений с использованием библиотек OpenCV и IPP : учеб. пособие / В.Д. Кустикова. – Н. Новгород : Нижегород. гос. ун-т им. Н.И. Лобачевского, 2012.
4. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1072 с.
5. OpenCV documentation [Электронный ресурс] / Miscellaneous Image Transformations. AdaptiveThreshold. – Режим доступа: [https://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous\\_transformations.html](https://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html), свободный. – Дата доступа: 30.11.2017.
6. OpenCV documentation [Электронный ресурс] / Image Filtering. getGaussianKernel. – Режим доступа: <https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html>, свободный. – Дата доступа: 30.11.2017.