

УДК 004.932

UDC 004.932

**АВТОМАТИЧЕСКАЯ РЕТУШЬ  
ПОРТРЕТОВ****AUTOMATIC PORTRAIT RETOUCH***В. В. Крят, А. О. Пешков,**V. V. Kryat, A. O. Peshkov,**О. А. Гнедина**O. A. Gnedina*

Донской государственной технической  
университет, Ростов-на-Дону, Российская  
Федерация

Don State Technical University, Rostov-on-Don,  
Russian Federation

[kryat.vladislav@yandex.ru](mailto:kryat.vladislav@yandex.ru)[kryat.vladislav@yandex.ru](mailto:kryat.vladislav@yandex.ru)[alex.peshkov.j@gmail.com](mailto:alex.peshkov.j@gmail.com)[alex.peshkov.j@gmail.com](mailto:alex.peshkov.j@gmail.com)[gnedina16@mail.ru](mailto:gnedina16@mail.ru)[gnedina16@mail.ru](mailto:gnedina16@mail.ru)

Большинство портретных снимков подлежат ретушированию с совершением одинаковой последовательности действий. Авторами рассмотрен общий алгоритм, позволяющий автоматизировать процесс ретуши портретов с использованием автоматического поиска лица на изображении, детектирования областей кожного покрова, частотного разложения этих областей на текстуру и тон, обработки тонового слоя с использованием билатерального размытия и комбинирования обработанных участков с оригинальным изображением с помощью преобразования Пуассона.

Most portraits should be edited with the same sequence of actions. This article describes a general algorithm for automated portrait retouching with automatic face detection, detection of skin regions, frequency decomposition of these areas into texture and tone, processing of the tonal layer using bilateral filter and combining the processed areas with the original photo using the Poisson blending.

**Ключевые слова:** портрет, ретуширование, Гаусс, билатеральное размытие, Пуассон.

**Keywords:** portrait, retouching, gaussian blur, bilateral filter, Poisson.

**Введение.** Качество портретной фотографии зависит от множества переменных: освещения, ракурса, технических характеристик и настроек программного обеспечения камеры. Кроме этого, важной составляющей качественной фотографии является внешний вид изображенного человека. Профессиональная ретушь портретных фотографий производится в специальном графическом редакторе, при этом совершается одинаковый набор действий: устранение недостатков текстуры кожи, выравнивание тона кожи, цветокоррекция.

**Цель исследования.** Целью данного исследования является разработка алгоритма автоматической ретуши портретных фотографий. На основе полученного алгоритма возможно спроектировать и реализовать программное средство для мобильных платформ.

**Детектирование лицевого контура на изображении.** Первым шагом является выделение контуров лица на фотографии. Для этого используется множество методов, которые можно классифицировать на:

- 1) эмпирические методы;
- 2) методы характерных инвариантных признаков;
- 3) распознавание с помощью шаблонов, заданных разработчиком;
- 4) методы обнаружения по внешним признакам, обучающиеся системы.

В алгоритме будет использоваться метод Виолы-Джонса [1], относящийся ко второму классу. Данный метод является одним из лучших по соотношению «точность распознавания/быстродействие». Основные алгоритмические принципы:

1) интегральное представление изображений [2]. Изображение представляется в виде матрицы, в каждой ячейке которой содержится сумма интенсивностей всех пикселей слева и выше данного. Расчет ведется согласно формуле:

$$L(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j). \quad (1)$$

Такое представление предполагает возможность быстро вычислять суммарную яркость произвольного прямоугольника на изображении;

2) использование признаков Хаара. Данные признаки дают точечное значение перепада яркости по оси абсцисс и ординат;

3) использование классификатора. Для каждой области изображения возвращается результат булевой функции *true* или *false* (лицо/не лицо).

Общий алгоритм сканирования изображения состоит из следующих этапов:

- 1) выбор окна сканирования;
- 2) выбор признаков;
- 3) масштабирование окна согласно выбранному признаку, причем в одном окне может вычисляться более  $10^5$  вариантов расположения признаков (рис. 1);

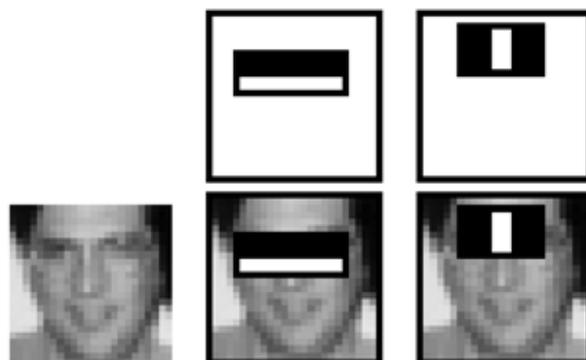


Рис. 1. Пример подстановки примитивов Хаара

- 4) отправка классификатору обнаруженных признаков;
- 5) сдвиг окна на одну ячейку.

В итоге вычисляется прямоугольная область, с высокой долей вероятности содержащая лицо.

**Обнаружение черт лица.** Следующим этапом является выделение на лице таких деталей, как губы, нос, брови и глаза. Для этих целей удобно использовать специальную аннотацию характерных точек лица. Каждый набор точек описывает каркас какой-либо черты лица:

- 1) 0–17 — челюсть;
- 2) 17–22 — правая бровь;
- 3) 22–27 — левая бровь;
- 4) 27–35 — нос;
- 5) 36–42 — правый глаз;
- 6) 42–48 — левый глаз;
- 7) 48–68 — рот.

В результате получается следующая схема лица (рис. 2).

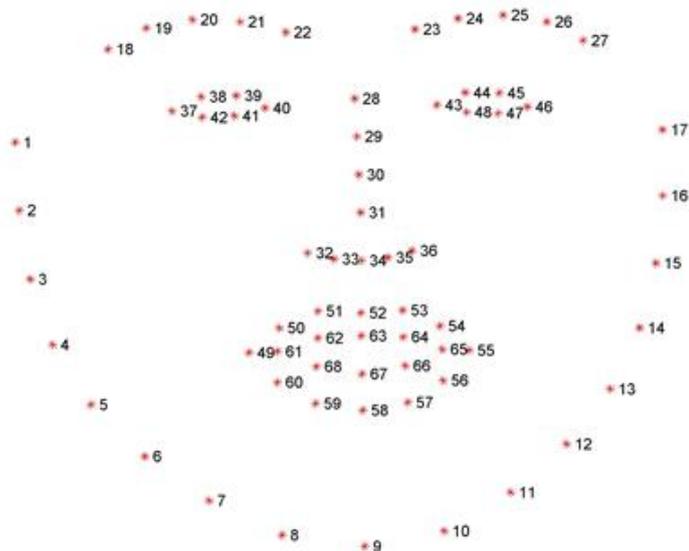


Рис. 2. 68 точек лицевых черт

Для решения вышеупомянутой задачи будем использовать открытую библиотеку компьютерного зрения *OpenCV* [3]. Метод с большой точностью возвращает словарь, содержащий номер точки  $N$  и ее координаты на фотографии  $P(x,y)$  — *Dictionary* $\langle$ Number, point( $X,Y$ ) $\rangle$ .

Полученные данные классифицируются по группам, согласно номерам точек, и объединяются в списки.

**Генерирование маски кожи лица.** Для последующей обработки лица необходимо выделить участки с кожей, а также создать соответствующую маску.

Одной из серьезных проблем при обнаружении участков с кожей является тот факт, что фотографии делаются при различном освещении, следовательно, тон кожи приобретает определенную цветовую температуру. Поиск по заранее заданным цветовым паттернам не даст хороших результатов. Предлагаемый метод не зависит от цветовой температуры и призван детектировать участки кожи почти при любых условиях.

На предыдущем шаге нами была получена прямоугольная область, в которой содержится лицо и координаты глаз, рта, бровей и носа. Можно утверждать, что в рамках этой области точно содержатся участки с кожей, причем тон кожи существенно не влияет на качество их обнаружения. Примем как маску для поиска кожи прямоугольник с найденным лицом и отнимем от него области глаз, бровей и рта, так как эти области резко отличаются по цвету от цвета кожи и могут внести погрешности в дальнейшие вычисления. Пример маски изображен на рис. 3.

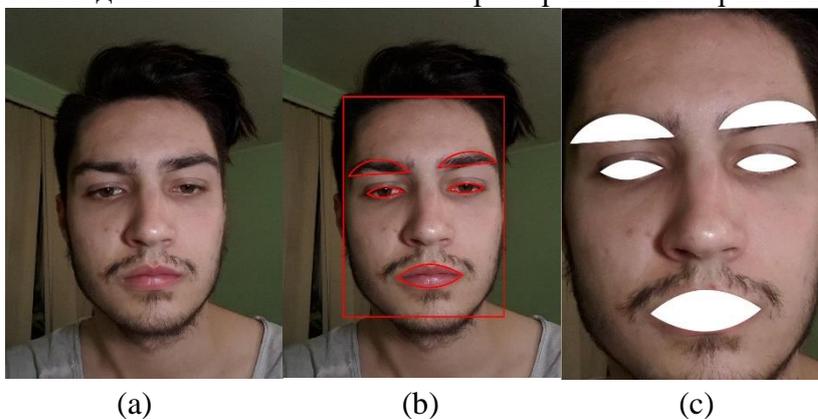


Рис. 3. Пример маски: исходная фотография (a); данные от детектора лица и черт лица (b); область интереса после наложения маски (c)

Далее участки кожи, попадающие под маску, конвертируются в цветовое пространство  $YCbCr$ , вычисляется среднее значение, как образец для нормального тона кожи, и стандартное отклонение для каждой компоненты пространства  $YCbCr$ . Во избежание погрешности у гистограмм соответствующих компонент с левого и правого края отбрасывается 10%.

Таким образом, маска кожи может быть получена следующим образом:

$$M(I_p) = \begin{cases} 1, & \mu_Y - 2\sigma_Y \leq Y(I_p) \leq \mu_Y + 2\sigma_Y, \\ \mu_{Cb} - 2\sigma_{Cb} \leq Cb(I_p) \leq \mu_{Cb} + 2\sigma_{Cb}, \\ \mu_{Cr} - 2\sigma_{Cr} \leq Cr(I_p) \leq \mu_{Cr} + 2\sigma_{Cr}, \\ 0, & \text{в ином случае,} \end{cases} \quad (2)$$

где  $M(I_p)$  — полученная маска;  $Y(I_p)$ ,  $Cb(I_p)$ ,  $Cr(I_p)$  — значение пикселя  $p$  изображения  $I$  в схеме  $YCbCr$ ;  $Y$ ,  $Cb$ ,  $Cr$  — среднее значение соответствующих компонент;  $\sigma_Y$ ,  $\sigma_{Cb}$ ,  $\sigma_{Cr}$  — стандартное отклонение соответствующих компонент. Полученная маска кожи представлена на рис. 4.



Рис. 4. Маска кожи, наложенная на исходное изображение

**Разделение низких и высоких частот.** На данном этапе возникает задача убрать с лица на фотографии видимые недостатки: шрамы, угревую сыпь, какие-либо пятна. Авторы предлагают простой и эффективный способ обработки.

Разделим изображение с кожей на две составляющие: текстуру и тон. Для получения текстуры применим фильтр размытия по Гауссу с небольшим радиусом к копии исходного изображения и отнимем почленно яркостную составляющую каждого пикселя оригинального изображения от размытого изображения [4]:

$$y(m, n) = \frac{1}{2\pi r^2} \sum_{u,v} e^{-\frac{(u^2+v^2)}{2r^2}} x(m+u, n+v). \quad (3)$$

Таким образом, полученная разница будет являться высокочастотной составляющей изображения кожи, иными словами — ее текстурой. Размытое по Гауссу изображение будем считать слоем, содержащим информацию только о тоне кожи. На этом слое останется информация с недостатками кожи, а в текстуре будет лишь сама структура кожи. Это позволит избавиться от большей части недостатков кожи, работая только с тоновой составляющей. Разделенные текстура и тон показаны на рис. 5.

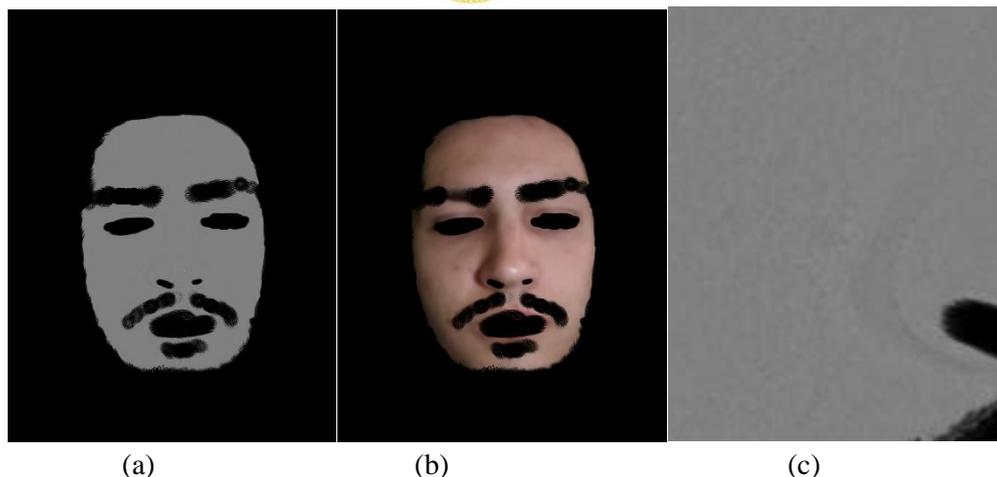


Рис. 5. Разделенные текстура и тон: слой с тоном (а); слой с текстурой (b); текстура в приближении (с)

**Выравнивание тона с помощью билатерального размытия.** Так как информация о тоне кожи доступна отдельно от текстуры, возможно сгладить тон, размыв его каким-либо фильтром. В качестве фильтра размытия можно использовать простое размытие по Гауссу, однако, данный фильтр вносит в изображение слишком усредняющий эффект, из-за чего свето-теневой рисунок лица сильно искажается. Поэтому было принято решение для сглаживания тона использовать билатеральный фильтр [5], сохраняющий границы. Прямой билатеральный фильтр является крайне ресурсозатратным, поэтому будем использовать фильтр по методу билатеральной решетки, так как он является более эффективным по соотношению быстродействия и качества.

В методе билатеральной решетки результат зависит только от расстояния между пикселями. Изображение представляется в трехмерном пространстве, где к стандартным координатам плоскости изображения добавляется координата интенсивности пикселя, т. е. вычисляется однородный вектор по формуле:

$$\begin{pmatrix} I_p * W_p \\ W_p \end{pmatrix} = \sum_{q \in S} G_{\sigma}(|p - q|) G_{\sigma}(|I_p - I_q|) \begin{pmatrix} I_q \\ 1 \end{pmatrix}. \quad (4)$$

Полученное пространство сжимается с заданной частотой дискретизации, при этом информация о каждом пикселе сохраняется в соответствующей накопительной ячейке. Структура таких ячеек называется билатеральной решеткой. Далее применяется билатеральный фильтр с учетом расстояния между пикселями — свертка с трехмерной функцией Гаусса с параметрами, после чего результат восстанавливается по исходным позициям пикселей. Данный алгоритм позволяет производить обработку с ядрами любого размера за короткий промежуток времени.

**Детектирование и устранение дефектов кожи на тоновом слое.** Поставим задачу найти такие кожные дефекты, как покраснения, неровности, пятна. Разобьем тоновый слой на сетку с  $N$  квадратными ячейками:

$$N = \frac{W * H}{a^2}, \quad (5)$$

где  $W$  — ширина маски;  $H$  — высота маски;  $a$  — длина стороны квадрата.

Размер ячейки выбирается согласно формуле:

$$a = \frac{\max(W, H)}{15}. \quad (6)$$

В итоге каждая ячейка будет содержать небольшой участок кожи (рис. 6).

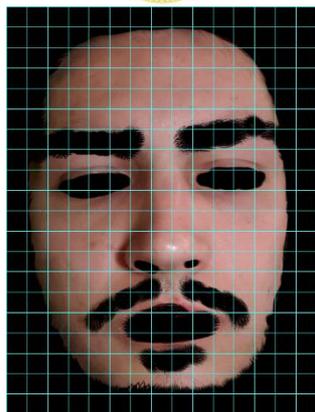


Рис. 6. Квантованная маска кожи

Далее необходимо составить гистограмму относительно показателя яркости для каждой ячейки и всего слоя в целом. Пример гистограммы продемонстрирован на рис. 7.



Рис. 7. Гистограмма тонового слоя от темного к светлому

На основе полученной гистограммы мы можем вычислить среднее значение яркости каждого пикселя:

$$P_{Y(avg)} = \sqrt{\frac{N}{N-1} * \sigma^2} = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (P_i - \bar{P}_i)^2}, \quad (7)$$

где  $P_Y^i$  — значение яркости каждого пикселя;  $N$  — общее количество пикселей.

Рассчитаем среднее значение для каждой из областей, используя аналогичную методику:

$$P_{Y(arg)}^k = \sqrt{\frac{N}{N-1} * \sigma^2} = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (P_i^k - \bar{P}_i^k)^2}. \quad (8)$$

В результате получим нормальное значение тона кожи  $P_{Y(arg)}$ , с которым будет сравниваться значение  $k$ -ой области  $P_{Y(arg)}^k$ . Если значение внутри области будет отличаться более чем на  $2\sigma$  от  $P_{Y(arg)}$ , то в данной области находится дефект кожи. В итоге получим средний показатель отклонения, значение которого будет использовано как коэффициент, на который мы будем опираться при билатеральном размытии (рис. 8):

$$\sigma_{avg} = \frac{1}{N} (\sigma_1 + \sigma_2 + \dots + \sigma_N). \quad (8)$$

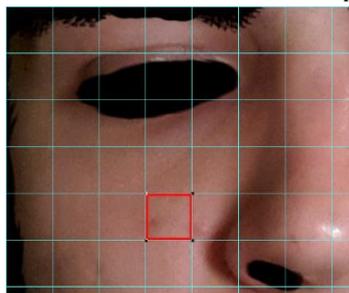


Рис. 8. Отклонение внутри области от нормы

В результате увеличения радиуса при билатеральном размытии средний показатель отклонения будет уменьшаться. Как только значение среднего внутри каждой области не будет превышать  $2\sigma$ , увеличение степени размытия будет остановлено, цвет кожи визуальнo станет более ровным, дефекты разглядятся. Результат размытия можно увидеть на рис. 9.



Рис. 9. Результат применения билатерального фильтра

**Комбинирование в финальное изображение.** Для бесшовного наложения отретушированных слоев с кожей на оригинальное изображение будет использоваться преобразование Пуассона [6]. Это преобразование позволяет незаметно для глаза интегрировать часть одного изображения в другое. Финальный результат в сравнении с оригиналом представлен на рис. 10.



Рис. 10. Сравнение оригинального изображения с обработанным

**Заключение.** Был разработан общий алгоритм, позволяющий автоматически ретушировать портреты, качественно улучшая изображение лица. Сложность алгоритма линейно зависит от размера изображения, ключевой точкой является обнаружение лица в кадре.

Преимуществом алгоритма является работа при различных уровнях освещенности и отсутствие потерь в качестве фактуры кожи за счет отделения тона и текстуры кожного покрова.

В дальнейшем планируется качественная модификация алгоритма — использование более сложных и более точных статистических подходов к обнаружению кожного покрова на фотографии, редактирование слоя с текстурой наравне с тоновым слоем. Для обнаружения более сложных дефектов кожи на текстуре предполагается использование алгоритма Габора, а для замены дефектных участков — алгоритм поиска паттернов кожи и генерации заплаток на их основе.

#### **Библиографический список.**

1. Местецкий, Л. М. Математические методы распознавания образов / Л. М. Местецкий. — Москва : МГУ, 2004. — 85 с.
2. Гонсалес, Р. Цифровая обработка изображений ISBN 5-94836-028-8 / Р. Гонсалес. — Москва : Техносфера, 2005. — 1072 с.
3. Pracial Python and OpenCV [Электронный ресурс] / Image search engine. — Режим доступа : <http://www.pyimagesearch.com/2017/04/10/detect-eyes-nose-lips-jaw-dlib-opencv-python/> (дата обращения : 09.05.17).
4. Быстрое размытие по Гауссу [Электронный ресурс] / Хабрахабр. — Режим доступа : <https://habrahabr.ru/post/151157/> (дата обращения : 09.05.17).
5. Анализ эффективности алгоритмов билатеральной фильтрации [Электронный ресурс] / Научная электронная библиотека eLIBRARY.RU. — Режим доступа : <https://elibrary.ru/item.asp?id=17651005> (дата обращения : 09.05.17).
6. Применение преобразования Пуассона для бесшовного наложения изображений [Электронный ресурс] / Хабрахабр. — Режим доступа : <https://habrahabr.ru/post/213403/> (дата обращения : 09.05.17).