



Исправление искажений в портретных снимках на широкоугольную камеру

Курсовая работа

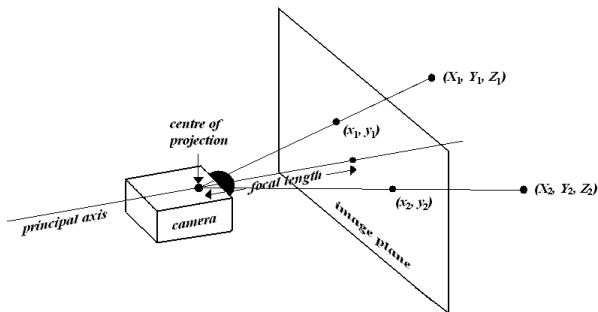
Автор: Александра Михайловна Елисеева, 17.Б10мм
Научный руководитель: ст. преп. А. А. Пименов

Санкт-Петербургский государственный университет
Кафедра системного программирования

26 мая 2020г.

Введение

- Форма объектов на изображении — результат прохода лучей от них через оптическую систему камеры и активации светочувствительной матрицы
- В большинстве оптических систем используется перспективная проекция



- Мобильная фотография очень популярна
- Люди любят фотографировать своих близких (и себя)
- Камеры с широкоугольными объективами позволяют вместить большее количество людей на фото
- Характеристики камер нескольких смартфонов для примера:

Phone Model	Selfie Camera FOV	Rear Camera FOV
Samsung Galaxy Note 10	80	123
Google Pixel 4	90	77
iPhone 11 Pro	80	120

Обзор существующих решений

Глобальные проективные преобразования: исправляют одни искажения, неизбежно создают другие



(a) Исходное изображение



(b) Стереографическая проекция



(c) Проекция Панини

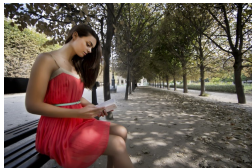


(d) Проекция Меркатора

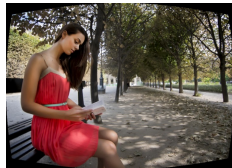
Обзор существующих решений

Коммерческие продукты: некоторые требуют дополнительной информации, некоторые функционируют достаточно хорошо

- DxO ViewPoint



(a) До



(b) После

- Adobe Photoshop (Perspective Warp)



(a) До



(b) После

Обзор существующих решений

Локальная коррекция: сочетание перспективной и стереографической проекций в методе ¹

- Исправляет форму лиц
- Сохраняет форму остальных объектов



(a) До



(b) После

¹ Distortion-Free Wide-Angle Portraits on Camera Phones. YICHANG SHIN, WEI-SHENG LAI, and CHIA-KAI LIANG, Google. 2019.

Постановка задачи

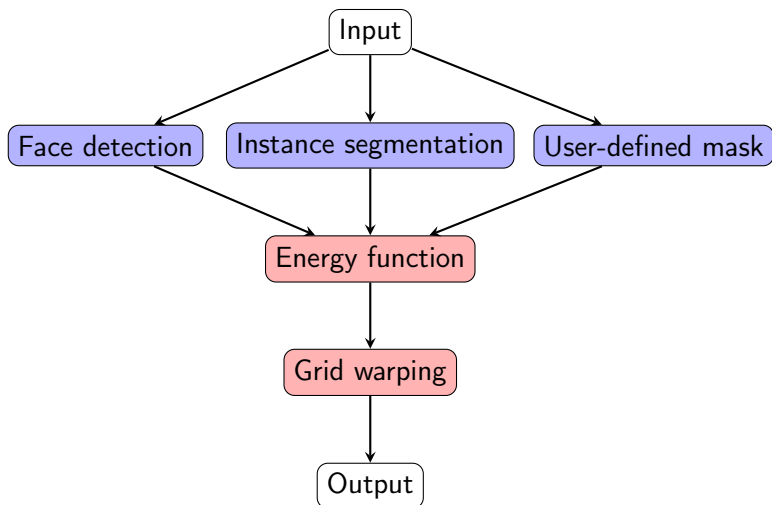
Целью данной курсовой работы является реализация функциональности метода ¹

Задачи:

- Изучить предметную область, ознакомиться с имеющимися результатами, выбрать стек технологий
- Разработать общую архитектуру приложения, соединяющего в себе этапы метода ¹
- Добавить функциональность обнаружения лиц и людей на изображении с использованием внешних библиотек и с использованием ввода от пользователя
- Реализовать и оптимизировать ценовую функцию на сетке
- Реализовать преобразование изображения по сетке
- Протестировать полученные результаты и написать отчет

¹ Distortion-Free Wide-Angle Portraits on Camera Phones. YICHANG SHIN, WEI-SHENG LAI, and CHIA-KAI LIANG, Google. 2019.

Архитектура приложения





Определение лиц и людей



(a) Haar Cascade Classifier

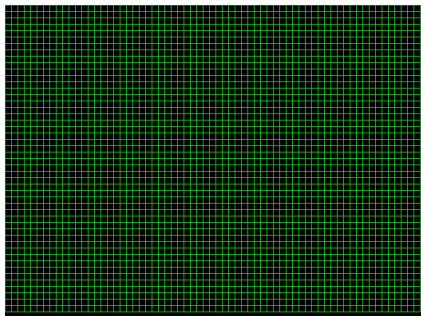


(b) Mask R-CNN

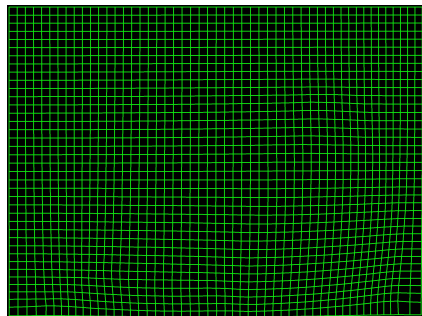


(c) Заданная пользователем маска

Преобразование изображения по сетке



(a) Исходная сетка



(b) Оптимальная сетка

- Преобразование лиц

$$E_f = \sum_{k \in \text{Faces}} \sum_{i \in F_k} w_k \cdot m_i \cdot \|v_i - (S_k \cdot u_i + t_k)\|_2^2 + \lambda(S_k) \quad (1)$$

$$S_k = \begin{pmatrix} a_k & b_k \\ -b_k & a_k \end{pmatrix}$$

$$\lambda(S_k) = w_s \|a_k - s_t\|_2^2$$

- Сохранение прямых

$$E_b = \sum_i \sum_{j \in N(i)} \left\| (v_i - v_j) \times \frac{p_i - p_j}{\|p_i - p_j\|_2} \right\|_2^2 \quad (2)$$

- Регуляризация

$$E_r = \sum_i \sum_{j \in N(i)} \|v_i - v_j\|_2^2 \quad (3)$$

- Сохранение исходного размера изображения

$$E_a = E_l + E_r + E_t + E_b \quad (4)$$

$$\begin{cases} E_l = \mathbb{I}(v_{i,x} > 0) \cdot \|v_{i,x}\|_2^2 & \forall v_{i,x} \in \delta_{left} \\ E_r = \mathbb{I}(v_{i,x} < W) \cdot \|v_{i,x} - W\|_2^2 & \forall v_{i,x} \in \delta_{right} \\ E_t = \mathbb{I}(v_{i,y} > 0) \cdot \|v_{i,y}\|_2^2 & \forall v_{i,y} \in \delta_{top} \\ E_b = \mathbb{I}(v_{i,y} < H) \cdot \|v_{i,y} - H\|_2^2 & \forall v_{i,y} \in \delta_{bottom} \end{cases}$$

Тестирование

- FOV 120 - заметные изменения



(a) До



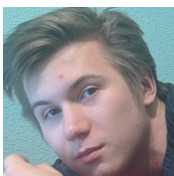
(b) Моя программа



(c) Стереогра-
фическая
проекция



(a)



(b)



(c)

Тестирование

Фон изображения не искажается



(a) До



(b) После

Тестирование

- FOV 70 - изменения не очень заметны



(a) До



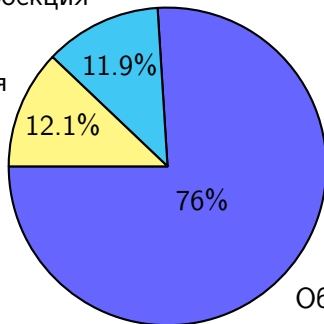
(b) Моя программа

Тестирование

Проведен небольшой опрос: «Какая фотография кажется вам наиболее естественной и близкой к реальности?»

Стереографическая проекция

Исходная фотография



Обработка приложения

Результаты

- Разработана архитектура приложения
- В приложение встроена функциональность instance segmentation с использованием фреймворка Mask R-CNN, функциональность определения лиц с использованием готового Haar Cascade Classifier и функциональность получения маски, задаваемой пользователем
- С использованием OpenCV реализовано поточечное аффинное преобразование изображения по сетке
- С использованием NumPy и SciPy реализована оптимизационная задача: ценовая функция, необходимые ограничения
- Полученное приложение протестировано на фотографиях с разным FOV и положением людей

Репозиторий на GitHub