Гироскоп на основе монокулярной камеры

Кудеров П.В.

ИУ9-121

**Оглавление**

[Введение 3](#_Toc420044688)

[1 Аналитическая часть 4](#_Toc420044689)

[1.1 4](#_Toc420044690)

[2 Научно-исследовательская часть 5](#_Toc420044691)

[3 Проектно-конструкторская часть 6](#_Toc420044692)

[4 Технологическая часть 7](#_Toc420044693)

[4.1 Методология разработки и используемые средства 7](#_Toc420044694)

[4.2 Руководство пользователя 7](#_Toc420044695)

[4.2.1 Приложение Gyrocam 7](#_Toc420044696)

[5 Организационно-экономическая часть 10](#_Toc420044697)

[5.1 Введение 10](#_Toc420044698)

[5.2 Организация и планирование процесса разработки программы 10](#_Toc420044699)

[5.2.1 Техническое задание 10](#_Toc420044700)

[Заключение 11](#_Toc420044701)

[Приложение 1. Диаграмма Ганта выполненных работ 13](#_Toc420044702)

# Введение

Собственное движение автотехники полагается на системы позиционирования. Распространенные способы построения подобных систем:

гироскопы на основе инерциальных датчиков. Проблема – накопление ошибки со временем.

системы глобального (спутникового) наведения – низкая точность в местах густой застройки, под кронами деревьев, под мостами (так ли это?)

Набирает популярность использование методов компьютерного зрения. Популярные решения:

SLAM, SfM – они пытаются построить 3D модель окружающей среды.

Другой вариант – построение гироскопа на основе определения трехмерной ориентации по изображению.

В данной работе рассматривается метод определения трехмерной ориентации монокулярной камеры с использованием точек схождения перспективы (ТСП), обнаруженных на изображениях городских сцен и внутри помещений, то есть в условиях так называемого «Manhattan World» [ссылка1] (Coughlan & Yuille, 1999, 2003).

Нахождение ТСП на изображениях позволяет решать довольно широкий круг задач: калибровка камеры, восстановление 3D сцены, вычисление ориентации камеры в пространстве с последующей реализацией гироскопа, ...(добавить).

Существует целый ряд работ, посвященных вычислению ТСП на снимках в рамках «Manhattan World». ...(описать вкратце).

В данной работе рассматривается, а также сделана попытка реализовать метод, предложенный Вилле Хуттуненом и Робертом Пише (Huttunen & Piché, 2012).

# Аналитическая часть

## 

# Научно-исследовательская часть

Поставленную задачу разработки можно разделить.

# Проектно-конструкторская часть

# Технологическая часть

## Методология разработки и используемые средства

## Руководство пользователя

### Приложение Gyrocam

Результатом работы является Win32 консольное приложение, реализующее алгоритм нахождения точек схождения перспективы (ТСП) на изображении, а также векторов единичных направлений, соответствующих им.

Для использования данного приложения требуется установленная на компьютере пользователя библиотека компьютерного зрения opencv версии 3.0.0. Описание процесса установки и необходимые файлы можно найти по адресу [http://docs.opencv.org/3.0‑alpha/doc/tutorials/introduction/windows\_install/windows\_install.html](http://docs.opencv.org/3.0alpha/doc/tutorials/introduction/windows_install/windows_install.html).

Приложение запускается со следующими аргументами:

1. строка абсолютного или относительго пути к файлу изображения, которое следует обработать. Поддерживается такие форматы как bmp, jpg, jpeg,tiff, png.
2. строка абсолютного или относительного пути выходного файла изображения. Формат выходного изображения будет выбран на основе расширения в имени файла данного пути. Поддерживаемые форматы те же, что и во входном аргументе.

Для изображения, заданного входным аргументом, приложение рассчитывает единичные векторы ТСП и выводит окно. В данном окне отображается само исходное изображение с наложенными поверх него прямыми линиями разных цветов. Пример окна приложения показан на рисунке (**Рисунок 1**).

Сначала поверх изображения накладываются линии, соответствующие трем наилучшим (с вероятностью >= 95%) кластерам сегментов линий, сходящихся в какой-либо одной точке. Для них используются темные варианты трех цветов – темно-синий, темно-зеленый и темно-красный цвета в порядке выделения каждого кластера.

Далее поверх полученного изображения накладываются линии, соответствующие кластерам сегментов линий, сходящихся в вычисленных ТСП. Данные линии отображаются яркими вариантами трех цветов – синим, зеленым и красным (для сравнения, например, если темный вариант красного – bgr(0, 0, 127), то яркий – bgr(0, 0, 255)). Кластеры одного цвета, но разной степени яркости в общем случае могут как полностью совпадать, так и частично различаться. Связь между ними следующая – сегменты темного цвета как бы продуцируют некоторую ТСП, а яркого – индуцированы этой же ТСП. В случае полного совпадения кластеров одного цвета – поверх кластера темного цвета будет наложен кластер яркого цвета и он не будет виден.

Выведенное пользователю изображение также сохраняется на диск в выходной файл. Путь, название и расширение файла определяется соответствующим (вторым) входным аргументом командной строки.



Рисунок 1 – Окно с обработанным изображением, поверх которого наложены сегменты линий, соответствующие найденным ТСП. На нем виден случай, когда кластеры темных вариантов цветов полностью покрыты кластерами ярких цветов.

Помимо описанного выше окна с изображением после вычисления ТСП приложение выводит в консоль некоторую служебную информацию от модуля opencv и затем две матрицы 3х3 с числами с плавающей запятой (**Рисунок 2**). Каждая матрица имеет формат как в примере ниже:

[0.99999738, 0.0022501017, 0.00047868371;

0.93076748, 0.36539984, -0.012442469;

-0.10985497, -0.99394757, -0.00042265668]

Границы матрицы задаются квадратными скобками, строки матрицы разделяются точкой с запятой с последующим переводом строки, числа в рамках одной строки разделяются запятыми.

Первой матрице соответствуют единичные векторы найденных ТСП. Каждый вектор представляет собой строку матрицы. Порядок соответствия векторов ТСП наложенным на изображение линиям следующий – синий, зеленый, красный.

Вторая матрица – ближайшая к первой ортонормированная матрица в смысле нормы Фробениуса. Данная матрица является искомой матрицей поворота системы координат камеры.

Описанный выше вывод двух матриц также сохраняется на диск текстовый файл в том же формате. Полный путь данного файла получается прибавлением расширения «.txt» к строке пути второго входного аргумента (пути к выходному изображению). Например, если при запуске приложения вторым аргументом на вход передать относительный путь «../TestSamples/output.jpg», то матрицы будут записаны в файл по следующему пути: «../TestSamples/output.jpg.txt».



Рисунок 2 – Вывод в консоли матриц направлений ТСП (неортогональной и ортогональной, соответственно) на тестовом запуске.

Описанное консольное приложение было протестировано на наборе изображений YorkUrbanDb (The York Urban Line Segment Database). Набор состоит из 47 изображений внутри помещений и 55 изображений городских сцен. Для каждого изображения из базы приведены данные по точкам схождения перспективы, что позволяет оценить уровень точности работы приложения. Описание и сам набор изображений можно найти на странице базы <http://www.elderlab.yorku.ca/YorkUrbanDB/>.

Также может представлять интерес для тестирования набор изображений в составе так называемой Eurasian Cities Database. Описание и сам архив базы для скачивания доступен на <http://graphics.cs.msu.ru/en/research/projects/msr/geometry>.

# Организационно-экономическая часть

## Введение

Разрабатываемое в рамках дипломной работы программное обеспечение является.

Целью данного раздела является расчет трудоемкости, продолжительности разработки программного обеспечения и сметы затрат.

## Организация и планирование процесса разработки программы

### Техническое задание

.

# Заключение

В данной работе были

Библиография

1. Диаграмма Ганта выполненных работ