Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Ижевский государственный технический университет имени

М. Т. Калашникова»

Кафедра «Программное обеспечение»

Отчет

по лабораторной работе

на тему

«Система аэронавигации»

по дисциплине

«Конструирование программного обеспечения»

Выполнили

ст. гр. Б08-191-1,2 Максимов А.Н.

Торхов Д.И.

Принял Власов В.Г.

Ижевск

2016

Содержание

Перечень рисунков

Перечень таблиц

1.ВВЕДЕНИЕ

1.1 Назначение системы

1.2 Область применения системы

1.3 Определения, акронимы, аббревиатуры

1.4 Ссылки

1.5 Обзор системы

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

2.3 Основные функциональные возможности системы

2.4 Основные условия системы

2.5 Основные ограничения системы

2.6 Характеристики пользователя

2.7 Допущения и зависимости

2.8 Оперативные сценарии

3. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

3.1 Физические

3.1.1 Конструкция

3.1.2 Износостойкость

3.1.3 Адаптируемость

3.1.4 Условия окружающей среды

3.2 Рабочие характеристики системы

3.3 Безопасность системы

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Ремонтопригодность системы

3.5.3 Надежность системы

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

4. ИНТЕРФЕЙСЫ СИСТЕМЫ

5. КЛАССЫ КОДИРОВАНИЯ

5.1 Описание классов

5.2 Соответствие классов проектирования и классов кодирования

6. ИСХОДНЫЙ КОД

**Перечень рисунков**

IDataStorage

<Algorithm

Params>

Хранение параметров алгоритмов

IDataStorage

<Bitmap>

Хранение результатов обработки

IDataStorage

<T>

Хранилище данных

IPoint

Descriptor

Обработчик особых точек

IContour

Detector

Детектор контуров

IImage

PreProcessor

Предварительная обработка

IImage

Processor

Обработчик изображений

Просмотр результатов обработки

Выбор параметров алгоритмов

Выбор изображений для обработки

Интерфейс пользователя

AirNavSystem

Рис. 1 Классы проектирования

1.ВВЕДЕНИЕ

1.1 Назначение системы

Система предназначена для тестирования и демонстрации алгоритмов аэронавигации БПЛА.

1.2 Область применения системы

Система применяется на персональных компьютерах под управлением операционной системой семейства Windows.

1.3 Определения, акронимы, аббревиатуры

БПЛА - беспилотный летательный аппарат.

1.4 Ссылки

1.5 Обзор системы

Система представляет собой приложение, работающее на компьютерах под управлением операционной системой семейства Windows, которое демонстрирует работу некоторых алгоритмов аэронавигации по снимкам, полученных с камеры БПЛА. Для демонстрации пользователь выбирает параметры обработки снимков, снимки, подтверждает операцию, после чего получает результат.

2. ОБЩЕЕ ОПИСАНИЕ СИСТЕМЫ

2.1 Системный контекст

2.2 Режимы и состояния системы

- выбор файлов обработки

- выбор параметров обработки

- обработка изображений

- демонстрация результата

2.3 Основные функциональные возможности системы

2.4 Основные условия системы

2.5 Основные ограничения системы

Система работает с только с изображениями формата bmp.

2.6 Характеристики пользователя

Пользователь - конечный пользователь системы. Использует систему для тестирования или демонстрации алгоритмов аэронавигации.

2.7 Допущения и зависимости

В некоторых случаях система может давать неверный результат из-за различных факторов таких как поворот БПЛА или схожесть объектов на снимке.

2.8 Оперативные сценарии

Система предполагает следующий сценарий: пользователь выбирает изображения для обработки, параметры обработки, после чего система обрабатывает выбранные изображения и демонстрирует результат пользователю.

3. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ, УСЛОВИЯ И ОГРАНИЧЕНИЯ СИСТЕМЫ

Предварительная обработка

Обработчик изображений

Просмотр результатов обработки

Выбор параметров алгоритмов

Выбор изображений для обработки

Интерфейс пользователя

AirNavSystem

Построение траектории движения

Сопоставление особых точек на соседних кадрах

Сопоставление выделенных контуров с эталонными

Описание контуров

Выделение особых точек

Выделение контуров

Хранение параметров алгоритмов

Хранение результатов обработки

Хранилище данных

Обработчик особых точек

Детектор контуров

3.1 Физические

3.1.1 Конструкция

Система находится на персональном компьютере пользователя.

3.1.2 Износостойкость

3.1.3 Адаптируемость

При какой либо ошибке обработки изображения система информирует пользователя, прерывает текущую обработку и продолжает работу в нормальном режиме.

3.1.4 Условия окружающей среды

3.2 Рабочие характеристики системы

3.3 Безопасность системы

3.4 Информационный менеджмент

3.5 Работа системы

3.5.1 Эргономика системы

3.5.2 Ремонтопригодность системы

3.5.3 Надежность системы

Система не является полностью надежной, т.к. возможны неверные результаты.

3.6 Стратегия и регулирование

3.7 Устойчивость жизненного цикла системы

4. ИНТЕРФЕЙСЫ СИСТЕМЫ

Детектор контуров:

interface IContourDetector

{

Bitmap Gradient(Bitmap source, out Bitmap gradientMagnitude, out Bitmap gradientAngle, out Bitmap gradientDirection);

Bitmap GetContours(Bitmap image, ContoursAlgorithmParams param);

ChainCode[] DescribeContours(Bitmap Contours);

TopologicalGraph BuildGraphFromChainCode(ChainCode chainCode);

}

Сохранение данных на диск:

interface IDataStorage<T>

{

T[] Load();

void Store(params T[] objects);

}

Базовая обработка изображений:

interface IImagePreProcessor

{

int[,] GetColorHistogram(Bitmap source);

Bitmap Contrast(Bitmap source, double threshold);

Bitmap Filter(Bitmap source, int dimension);

Bitmap EqualiseHistogram(Bitmap source);

Bitmap GaussFilter(Bitmap source, int windowSize);

Bitmap LinearFilter(Bitmap source, int windowSize);

Bitmap MedianFilter(Bitmap source, int windowSize);

Bitmap StretchHistogram(Bitmap source, double lowThreshold, double highThreshold);

Bitmap ToBlackAndWhite(Bitmap source);

}

Нахождение траетории и местоположения БПЛА:

public interface IImageProcessor

{

Point GetCurrentLocation(ContoursAlgorithmParams param, Bitmap image, Bitmap areaMap);

Point[] GetTrajectory(PointsAlgorithmParams param, params Bitmap[] images);

}

Объект на изображении:

interface IMapObject

{

List<Point> Points { get; }

Point Middlepoint { get; }

MapObjectType Type { get; }

double GetDistanceTo(MapObject mapObject);

double GetDistanceTo(Point point);

double GetAngleTo(MapObject mapObject);

double GetAngleTo(Point point);

}

Описание точек:

interface IPointDescriptor<T>

{

double [] Values { get; }

double GetDistinctWith(T other);

}

Детектор точек:

interface IPointsDetector

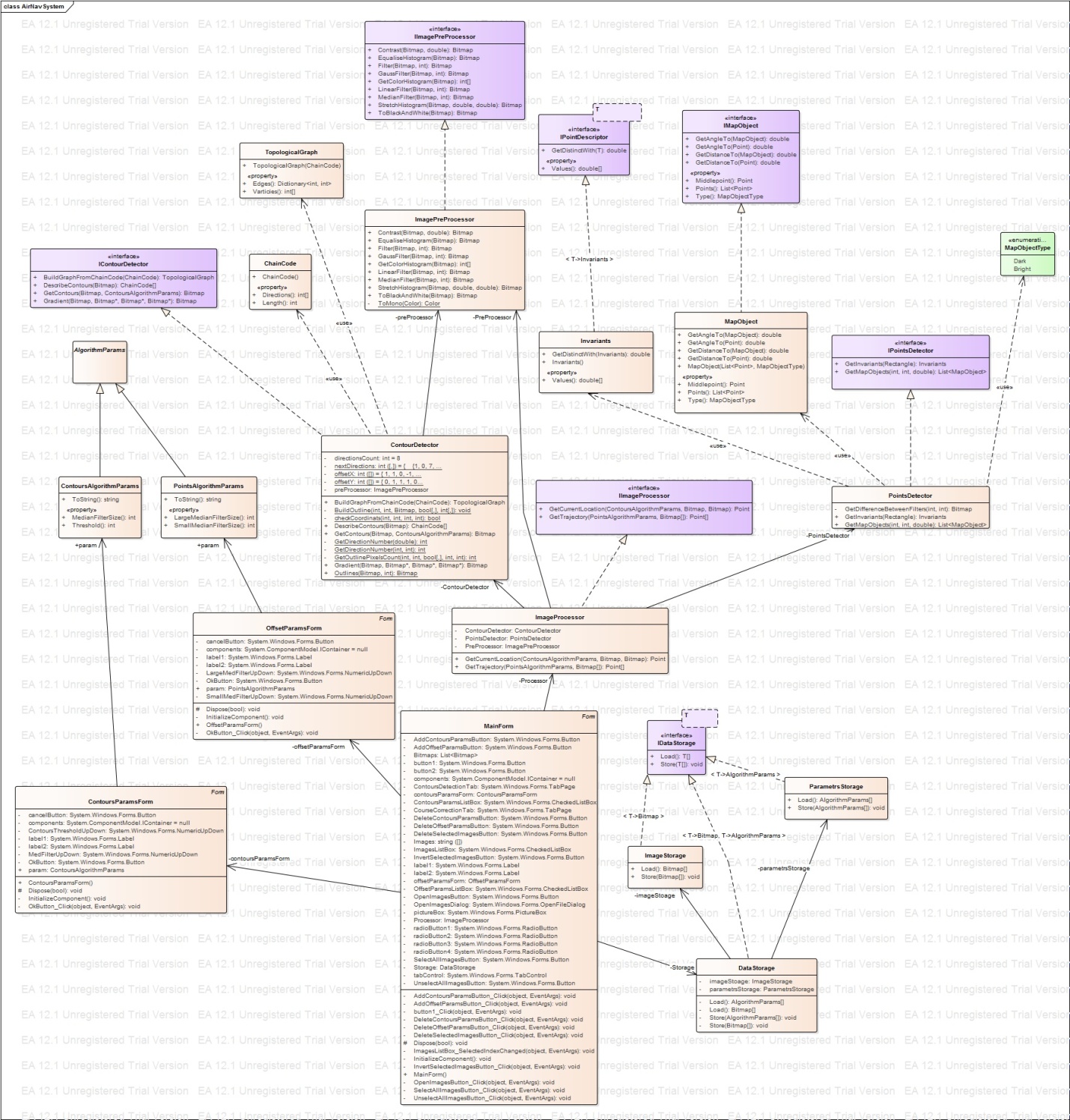
{

Invariants GetInvariants(Rectangle where);

List<MapObject> GetMapObjects(int minuendDimension, int subtrahendDimension, double threshold);

}

5. КЛАССЫ КОДИРОВАНИЯ



5.1 Описание классов:

1. AlgorithmParams - абстрактный класс, параметры алгоритма.
2. ChainCode - класс, цепной код контура изображения.
3. ContoursAlgorithmParams - класс, параметры алгоритма поиска контуров изображения.
4. Invariants - класс, дескрипторы объектов изображения (инварианты Ху).
5. MapObject - класс, объект на изображении.
6. MapObjectType - перечисление, тип объекта на изображении.
7. PointsAlgorithmParams - класс, параметры алгоритма нахождения смещения изображений.
8. TopologicalGraph - класс, топологический граф.
9. ContourDetector - класс, детектор контуров.
10. DataStorage - класс, сохранение изображения и параметры алгоритмов.
11. ImagePreProcessor - класс, базовая обработка изображений.
12. ImageProcessor - класс, нахождение текущего положения и траектории БПЛА.
13. ImageStorage - класс, сохранение изображений.
14. ParametrsStorage - класс, сохранение параметров алгоритмов.
15. PointsDetector - класс, детектор объектов.
16. ContoursParamsForm - класс, форма ввода параметров алгоритма поиска контуров.
17. MainForm - класс, главная форма программы.
18. OffsetParamsForm - класс, форма ввода параметров алгоритма нахождения смещения изображений.

5.2 Соответствие классов проектирования и классов кодирования

Интерфейс пользователя - ContoursParamsForm, MainForm, OffsetParamsForm.

Обработчик изображений - ImageProcessor.

Предварительная обработка - ImagePreProcessor.

Детектор контуров - ContourDetector.

Обработчик особых точек - PointsDetector.

Хранилище данных - DataStorage.

Хранение результатов обработки - ImageStorage.

Хранение параметров обработки - ParametrsStorage.

6. ИСХОДНЫЙ КОД

Исходный код системы располагается в репозитории по адресу https://github.com/zxvad/laboratory\_2015\_16/tree/master/Система аэронавигации /AirNavSystem