

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Математико-механический факультет

Кафедра Системного Программирования

Иванов Александр Аркадьевич

Рандомизированные методы в задачах  
выделения  
целей на аэрофотоснимках

Курсовая работа

Научный руководитель:

д.ф.-м.н., проф. Граничин О. Н.

Санкт-Петербург

2013

## Оглавление

1. Введение .....	3
2. Постановка задачи .....	5
3. Обзор.....	6
3.1. Алгоритм SURF. ....	8
4. Реализация .....	10
5. Другая реализация.....	13
6. Заключение .....	17

# 1. Введение

На сегодняшний момент технологии распознавания образов находят широкое практическое применение во многих областях человеческой деятельности. Это распознавание текстов, обработка аудиопотока и видеопотоков с целью выявления случаев нарушения авторских прав. Во многих медицинских учреждениях уже стоит оборудование, способное автоматически распознать симптомы заболевания, или локализовать проблемную область у пациента при ультразвуковом сканировании, магнитно-резонансной томографии. Сегодня невозможно представить повседневную жизнь без приборов считывания штрих кодов с товаров при обслуживании на кассе. Также в нашей стране уже несколько лет внедряется во многих регионах система автоматического оповещения автомобилистов о совершенном нарушении правил дорожного движения. Всё это стало возможно благодаря современным достижениям в области компьютерного зрения и естественному увеличению производительных мощностей аппаратных платформ.

Новые технологии позволяют осуществлять интеллектуальную обработку непрерывных видеопотоков, на лету делать анализ фотоснимков наблюдаемых территорий. В связи с этим все чаще находят применение беспилотные летательные аппараты в сфере мониторинга местности и сбора данных. Газовые и нефтяные корпорации могут существенно сократить свои расходы на повседневное обеспечение безопасности своих газо и нефтепроводов, благодаря возможности частичного замещения человеческого труда роботами со специальными датчиками утечек, которые в состоянии преодолевать большие расстояния и собирать информацию о возможных повреждениях на линиях. Данные могут как записываться на носитель, так и непосредственно передаваться в центр контроля полетов по

беспроводным каналам связи.

На кафедре системного программирования математико-механического факультета СПбГУ в последние годы стремительно развивается проект под руководством профессора О.Н. Граничина малогабаритных беспилотных летательных аппаратов. Исследования проходят по многим направлениям, в числе которых задача распознавания объектов на полученных в ходе испытаний аэрофотоснимков. Текущая конфигурация легких планеров позволяет установку на корпус периферийных модулей, таких как:

- автопилот
- модуль GSM
- фотокамера, по своим характеристикам сравнимой с теми, что реализуются в современных мобильных устройствах
- и др.

Типичная высота, набираемая планером варьируется в пределах до 200 метров. Однако реальные характеристики позволяют данному аппарату безопасно летать на высоте до километра.

На сегодня имеются несколько коммерческих применений данного БПЛА, среди которых важное место занимает задача обнаружения на фотоснимках территории, облетаемой планером, сравнительно небольших объектов. Это может быть как животная особь в пределах заповедной зоны, так и неодушевленный предмет: машина, строение и т.п.

## 2. Постановка задачи

При облете территории производится непрерывная фотосъемка низлежащей местности. Данные накапливаются на внутреннем носителе БПЛА, и при необходимости могут быть отправлены на центр управления полетами (в дальнейшем “базу”). В июле этого года были проведены плановые испытания БПЛА в малонаселенной местности, одной из целей которых был сбор необходимых материалов к задаче распознавания образов. На имеющихся снимках в качестве объекта поиска была выбрана машина (см. рис. 1). Требуется разработать эффективный алгоритм поиска по имеющимся данным.



Рис. 1

### 3. Обзор

Снимки, получаемые БПЛА имеют интервал в 1.5-2 секунды. За этот промежуток времени БПЛА может достичь угловой точки траектории и произвести поворот. Так же в силу небольшого веса планера, он подвержен влиянию погодных условий, что так же сказывается на угле, под которым был получен очередной кадр. В таком случае два последовательных изображения претерпевают различного рода искажения относительно друг друга. Это аффинные преобразование объектов в кадре, дополнительные шумы и не исключены вариации в масштабе.

Существующие подходы в области распознавания образов, которые активно применяются к анализу изображений, можно разделить на следующие три категории:

- Метод скользящего просмотра

Идея метода основана на поиске максимума корреляции имеющегося шаблона на заданной области поиска. При этом осуществляется последовательное сопоставление имеющегося образца с изображением в каждой точке. Из основных недостатков данного метода можно выделить следующие:

- ограниченный набор объектов поиска, которые необходимо заранее подготавливать, что сильно сужает область его применения
- даже при небольших изменениях геометрических характеристик изображения, точность поиска сильно падает. Отсюда вытекает как следствие следующий пункт
- вычислительная сложность перебирающих алгоритмов настолько велика, что для достижения приемлемой точности расчетов, обычные современные рабочие станции не в состоянии оперативно осуществить поиск.

- Машинное обучение

В данную категорию входят методы, опирающиеся при поиске на широкую базу заранее подготовленных примеров. Это различные методы классификации шаблона по имеющемуся обучающему множеству, подготовка которого занимает порядочное время. Важным критерием является построение таких наборов положительных и отрицательных примеров, что в контексте конкретной задачи получаемые результаты будут давать удовлетворительные графики precision-recall. К сожалению, добиться такой точности суть весьма нетривиальная задача, которая имеет стабильные результаты на распознавании относительно сложных объектов. Наибольшее применение данные методики получили в области распознавания лиц, пешеходов и ряда иных объектов сложной структуры. В задаче же нахождения объектов на аэрофотоснимках в первую очередь приходится искать сравнительно простые геометрические фигуры небольших размеров.

- Методы выделения характерных признаков

Работа таких методов не требует наличия обученного классификатора и они применимы к изображениям практически произвольного класса. Основной идеей является выделение ключевых особенностей в кадре. За этим следует соотнесение полученных данных с характерными признаками шаблона. Высокая скорость расчетов делает целесообразным их применение к задаче обнаружения объектов на аэрофотоснимках.

### 3.1. Алгоритм SURF.

SURF – Speeded Up Robust Features. Данный метод является развитием алгоритма SIFT – Scale-Invariant Feature Transform. Основная задача алгоритма SIFT заключается в обнаружении локальных характерных особенностей изображения. В контексте нашей задачи основными достоинствами данного метода являются устойчивость к масштабированию, вращению и относительная устойчивость по отношению к аффинным преобразованиям. Алгоритм начинает свою работу с нахождения в изображении локальных экстремумов, на основе чего выделяются особые точки. При этом обеспечивается инвариантность относительно поворота сцены.

- Особые точки определяются с помощью матрицы Гессе. Гессиан является симметрической квадратичной формой, описывающей поведение функции во втором порядке. Матрица этой функции образована её вторыми частными производными.

– Градиент в точке вычисляется с помощью фильтров Хаара



где черные области имеют значение -1, а белые +1. Это даст значение перепада градиента по осям.

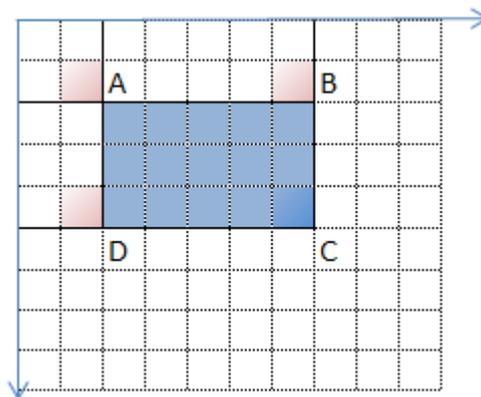
– Каждой особой точке сопоставляется свой дескриптор, который является набором из 64 чисел. Так как в точках достигается максимум гессиана, то точки обособлены друг от друга на определенном расстоянии, зависящем от значения вычисленного в них градиента. Инвариантность же достигается за счет обобщения градиента, который усредняется вокруг точки в целом с помощью фильтра Хаара.

– Фильтры Гессе и Хаара считаются эффективно за счет использования интегрального представления изображений

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

Идея заключается в удобной для подсчетов реализации хранения результатов сумм. Так,

если имеется прямоугольник ABCD



то сумма внутри прямоугольника выражается формулой:

$$Summ = I(A) + I(C) - I(B) - I(D)$$

– Полученное значение гессиана позволяет найти локальный экстремум яркости изображения. Масштаб матрицы Гессе обуславливает размер области, на которой считается дескриптор. Тем самым достигается инвариантность масштаба.

## 4. Реализация

Сегодня реализацию алгоритма SURF можно найти в библиотеке OpenCV. Поскольку библиотека является кроссплатформенной, а также имеется реализация для OS Android, то её можно использовать для непосредственных расчетов на самом БПЛА. Была выбрана реализация на языке C++. Библиотека также предоставляет полный набор интерфейсов для языка Python.

Основной класс для осуществления поиска особых точек изображения **SurfFeatureDetector**. Строится детектор с эмпирически выбранным для нашей задачи порогом гессiana равным 100:

```
SurfFeatureDetector detector( minHessian );
```

Полученные на вход шаблон и изображение предварительно свертываются по Гауссу с ядром равным 5 для устранения резких перепадов градиентов:

```
GaussianBlur( in, out, Size( ker, ker ), 0, 0 );
```

Определяются ключевые точки

```
detector.detect( img, keypoints );
```

и строится дескриптор

```
extractor.compute( img, keypoints, descriptors );
```

Для сравнения векторов дескрипторов был выбран **BruteForceMatcher**:

```
std::vector< DMatch > matches;  
matcher.match( descriptors_1, descriptors_2,  
                  matches );
```

С его помощью можно найти углы целевого объекта. Соединяя углы

линией получаем следующий результат для серии последовательных снимков (см. рис. 2, 3, 4):



Рис. 2



Рис. 3

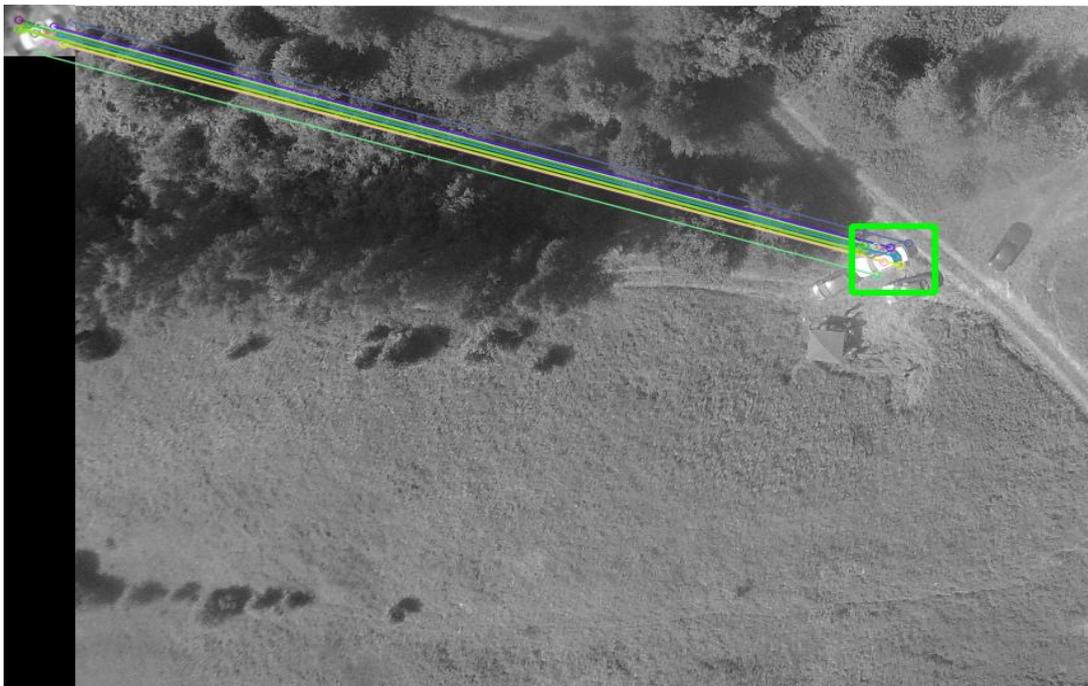


Рис. 4

Данный метод показал свою эффективность в распознавании небольших объектов на аэрофотоснимках. Хотя данный алгоритм и не имеет значительного процента ложных срабатываний, всё же при определенных положениях камеры, результат может быть неудовлетворительным.

По этой причине следует также уделить внимание другому подходу.

## 5. Другая реализация

Как не странно, но речь пойдет о методе скользящего просмотра. В мае 2013 года был сделан научный доклад “Хорошая технология делает трудную задачу просто” А.Н. Тереховым об успешном распараллеливании базового алгоритма полного перебора с помощью применения языка HaSCoL. В предложенном решении удалось уменьшить вычислительную сложность алгоритма поиска шаблона 128x128 в изображении 512x512 приблизительно в 1200 раз.

Предлагаемое решение:

◆ Считаем сумму и сумму квадратов пикселей

$$zv = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} a_{ij} \quad s2 = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} a_{ij}^2$$

◆ Считаем знаменатель нормировки

$$az = \sqrt{s2 - m * n * \left(\frac{zv}{m * n}\right)^2} = \sqrt{s2 - \frac{zv^2}{m * n}}$$

◆ Считаем нормированные значения

$$ap_{ij} = \frac{a_{ij} - \frac{zv}{m * n}}{az} = \frac{m * n * a_{ij} - zv}{m * n * az}$$

◆ Итоговое значение расчетной функции

$$F_{kl} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} zp_{k+i,l+j} * ap_{ij}$$

Не смотря на то, что подобную процедуру на сегодняшний момент невозможно провести непосредственно на борту БПЛА, сам факт того, что данный алгоритм может обрабатывать изображения на базе достаточно быстро, говорит о перспективности исследования в этом направлении.

Однако у предложенного алгоритма присутствуют все вышеупомянутые

проблемы, характерные для методов данного класса. Потому, чтобы алгоритм был более устойчив к вращениям и масштабированию, необходимо увеличить нагрузку за счет поиска шаблона в разных ориентациях и размерах. Но важным фактором, который способен значительно сократить возможное количество комбинаций, являются дополнительные гироскопические данные самого БПЛА, которые могут быть соотнесены в тот или иной момент с получаемыми снимками. Таким образом, требуется дополнительная проверка на эффективность применения методов стохастической оптимизации к данной реализации алгоритма.

Для всякой точки изображения шаблон сравнивается лишь частично с заданной вероятностью выборки значимых пикселей. Для каждого порога вероятности, который берется в моей работе с шагом в 10%, программа запускается несколько раз и берется усреднение по времени работы и отклонению.

Для иллюстрации эффективности подхода рандомизации приведем полученные результаты работы программы, которая эмулирует данные с GPS (считается, что машина могла переместиться не более чем на расстояние  $r$  от её местоположения на предыдущем снимке) (см. рис. 5, 6, 7)



Рис. 5



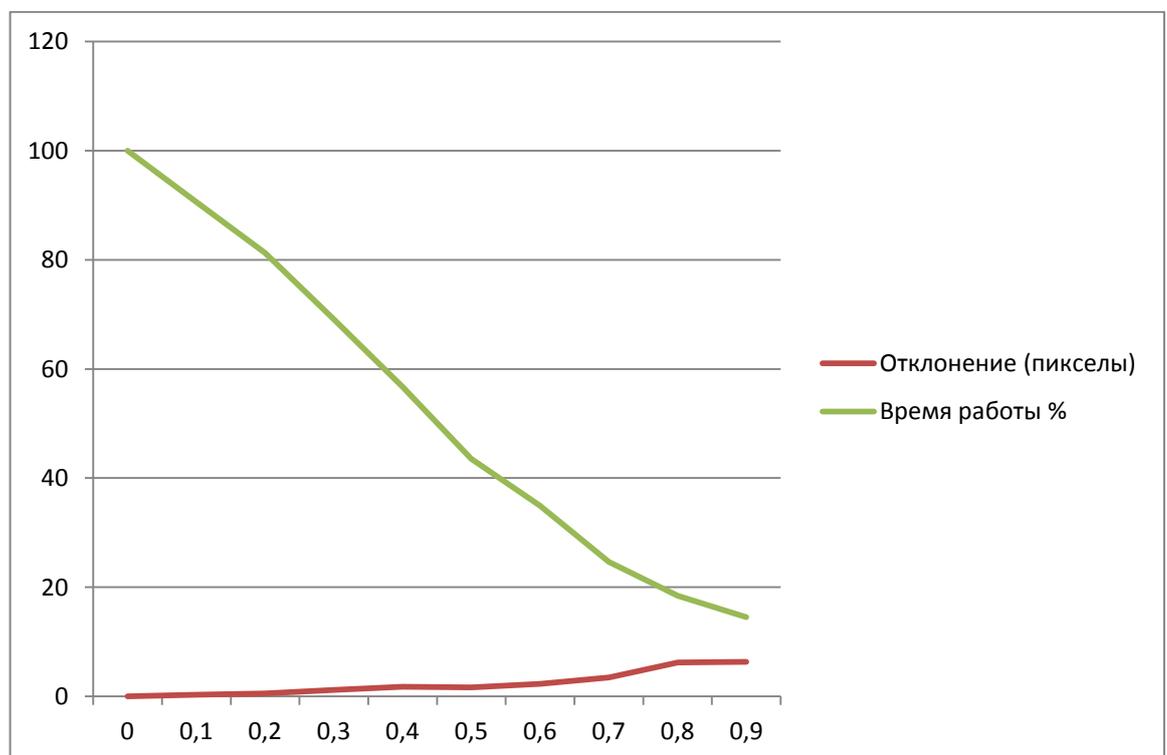
Рис. 6



Рис. 7

где соответствующими цветами обозначены регионы пересечения с искомой областью для выборок 100% (красный), 90% (желтый), 80% (зеленый) и 70% (синий). Ниже приведены данные по среднему отклонению всех выборок начиная со 100% и заканчивая лишь 10% значимых пикселей шаблона.

Вероятность	Отклонение	Время	Время (%)
0	0	342,4	100
0,1	0,3	310	91
0,2	0,58	278,2	81
0,3	1,22	236,6	69
0,4	1,8	194,2	57
0,5	1,64	149	44
0,6	2,3	119,7	35
0,7	3,5	84,3	25
0,8	6,2	63,1	18
0,9	6,35	49,7	15



## **6. Заключение**

Алгоритмы распознавания образов охватывают широкую сферу применения. В то же время каждый из них имеет наиболее эффективное прикладное значение лишь в тех условиях, где особенности контекста позволяют ему справляться с задачей преимущественно лучше других. Алгоритмы перебора при учете как можно более полного набора возможных сценариев изменений сцены неизменно дают максимальную корреляцию. При этом, как видно из данной работы, увеличение производительности на порядок путём стохастической оптимизации не сказывается сильно на результате поиска. В тех случаях, когда необходимо проводить вычисления на самом летательном аппарате в режиме реального времени, приведенная реализация алгоритма компьютерного зрения способна решать эту задачу фактически без необходимости учета гироскопических данных.

## Список литературы

1. Speeded-Up Robust Features (SURF) by Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool //   
<http://www.vision.ee.ethz.ch/~surf/eccv06.pdf>
2. Notes on the OpenSURF Library by Christopher Evans, January 18, 2009 //   
<http://www.cs.bris.ac.uk/Publications/Papers/2000970.pdf>
3. R. Szeliski Computer Vision: Algorithms and Applications
4. OpenCV official documentation //   
<http://docs.opencv.org/>
5. «Хорошая технология делает трудную задачу простой» А.Н. Терехов, май 2013