**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**имени М.В.ЛОМОНОСОВА**

**Факультет космических исследований**

**Робототехника и Интеллектуальные технологии**

**Выпускная Квалификационная Работа**

**«Применение точечных детекторов в алгоритмах обработки изображений ландшафтных сцен»**

Выполнил:

Шелепанов Даниил Андреевич

М-208 РИ

Научный руководитель:

д.ф.-м.н. Богуславский Андрей Александрович

Москва – 2021

ВВЕДЕНИЕ

В последние 20 лет применение систем технического зрения (СТЗ) в анализе изображений ландшафтов для решения разнообразных задач достигло уровня успешного применения. Среди различных работ в данной области можно выделить:

* Работы, связанные с картографированием местности с помощью летательных аппаратов с бортовыми камерами. [https://arxiv.org/pdf/1901.00211.pdf]
* Работы, связанные с определением характеристик движения летательных аппаратов. Опрееление компонент движения посадочного модуля марсианских автоматических станций Spirit и Opportunity описано в [10, 11].

В работах по сопоставлению изображений друг с другом и с картами существенным допущением является то, что на изображениях ландшафтов как правило отсутствуют объекты с заранее известной моделью, то есть в основу работы таких систем необходимо поставить точечные признаки. В качестве первых этапов обработки в рассматриваемых системах выполняется их обнаружение и прослеживание.

Цель данной работы — выбрать детектор особенностей, который удовлетворял бы требованиям скорости и надёжности: обрабатывать не менее 25 кадров в секунду и стабильно отслеживать признаки на нексольких последовательных кадрах.

ОБЗОР УСКОРИТЕЛЕЙ И СТАНДАРТОВ

Для алгоритмов обработки изображений характерно большое количество одинаковых операций, например, операция свёртки, выполненная для каждой подобласти изображения заданного размера. Среди архитектур современных вычислителей, подходящих для параллельного выполнения таких операций и относительно доступных, можно выделить SIMD – архитектуры [ссылка на таксономию Флинна], а именно:

* Видеокарты – устройства, преобразующие графическаий образ, хранящийся в памяти компьютера, в изображение, пригодное для вывода на монитор. Как правило, видеокарты реализуют арифметические операции с векторами чисел, что позволяет выполнить одну инструкцию одновременно для нескольких наборов аргументов. В последнее время видеокарты реализуют тезнику GPGPU – технику использования видеокарты для неграфических вычислений.
* CPU с поддержкой SSE и AVX – расширений набора команд векторными операциями

В данной работе использованы CPU Intel Core i5 и GPU Nvidia GTX 1050.

Среди стандартов, реализующих технику GPGPU, наиболее популярны следующие:

* CUDA – это программно-аппаратная платформа, реализованная на GPU Nvidia и позволяющая использовать ресурсы GPU для произвольных вычислений.
* OpenCL – это стандарт, описывающий интерфейс к прораммированию приложений, использующих параллельные вычисления на различных CPU и GPU. Главное отличие от CUDA – разработка ведется несколькими компаниями, следовательно, стандарт открыт и реализован многими производителями для своих вычислителей.

Таким образом, в данной работе используются следующие пары вычислитель – стандарт параллельности:

* Intel Core i5 + SSE/AVX
* Nvidia GeForce GTX 1050 + CUDA
* Nvidia GeForce GTX 1050 + OpenCL

ОБЗОР ДЕТЕКТОРОВ

Детектор углов Харриса

Детектор Харриса является детектором не признаков, а особых точек, то есть не включает в себя метод построения дескриптора. Предназначен для обнаружения углов. Для этого в нем используются собственные значения и матрицы:

где - первая производная локальной свертки окрестности изображения в точке с ядром гауссовой функции со стандартным отклонением . Cобственные значения и матрицы представляют изменения интенсивностей изображения в ортогональных направлениях. Рассмотрим меру угловатости (cornerness measure):

при небольшом значении параметра , например, . В силу общих свойств собственных значений имеем:

.

Детектор Харриса широко используется в задачах получения визуальной одометрии, например, в [5, 7].

Детектор SIFT

SIFT (Scale Invariant Feature Transform) – детектор, инвариантный к масштабированию и поворотам, то есть с достаточной степенью точности с его помощью можно найти одни и те же признаки на масштабированных и повернутых друг относительно друга изображениях. Непосредственно данный детектор в работе не используется, но его стоит упомянуть, так как следующий детектор SURF использует его принципы.

В качестве метода поиска особых точек используется поиск локальных максимумов в пространстве масштабов разности гауссиан (DoG). Пространство масштабов состоит из сверток гауссова ядра с исходным изображением в разных масштабах (коэффициент масштабирования). Набор сверток изображения одного масштаба называется октава. В каждой октаве, состоящей из изображений, путем последовательных вычситаний изображений друг из друга получается набор разностей гауссиан.

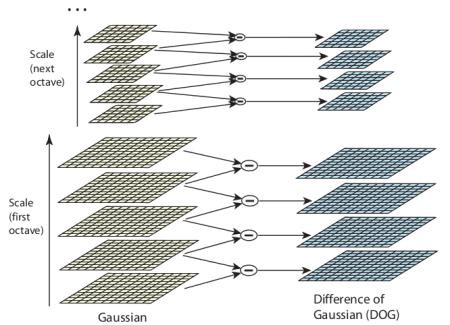


Рис. 1: пространство масштабов DoG

Каждая точка сравнивается со своей окрестностью, которая состоит из 8 соседних точек на текущем масштабе, 9 точек на следующем и 9 на предыдущем масштабах:

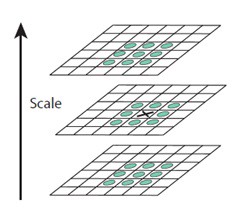


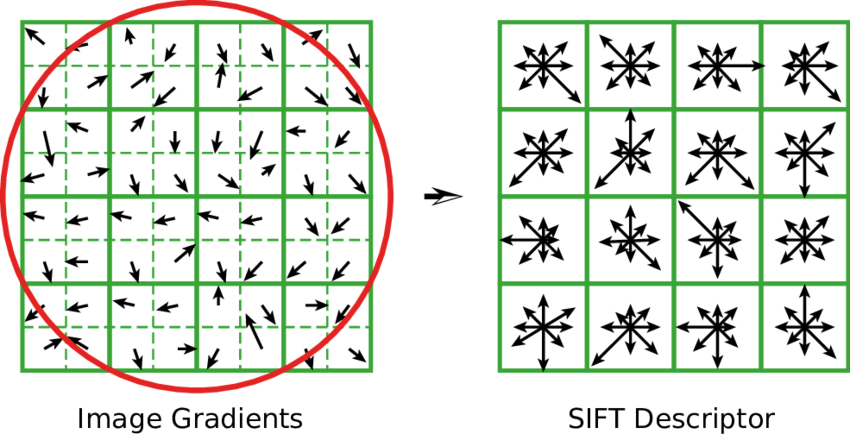
Рис. 2: поиск локального максимума

Если точка является локальным максимумом, она становится точкой-кандидатом. Так как точек-кандидатов получается слишком много, точки, лежащие на границах и в слабоконтрастных областях, отбрасываются.

В качестве дескриптора точки SIFT использует гистограмму направлений. В окрестности радиуса , где - номер слоя в пространстве масштабов, для каждой точки вычисляются ее модуль и направление:

Направление отображается в 1 из 36 счетчиков, каждый из которых представляет интервал 10 градусов, начальные значения счетчиков равны нулю. При попадании направления в какой-либо интервал к его значению прибавляется соответствующий модуль. Получается гистограмма градиентов. Локальные максимумы счетчиков величиной не менее 80% глобального максимума определяют доминантные направления, которые обеспечивают инвариант относительно вращения.

Для обеспечения инварианта относительно масштабирования окрестность точки разбивается на 16 квадратов, в каждом из которых строится гистограмма с интервалом 45 градусов из значений градиентов, попавших в данный квадрат. В результате получаются 16 векторов длиной 8, из которых можно составить один вектор длиной 128, это и есть SIFT-дескриптор.

Рис. 3: слева – схема градиентов на окрестности точки, разбитая на 16 квадратов, справа – SIFT-дескриптор.

Детектор SURF

SURF (Speeded-Up Robust Features) основан на тех же идеях, что и SIFT, при его разработке ставилась цель повысить эффективность работы. Так, в SIFT лаплассиан гауссианы аппроксимировался разностью гауссиан, в SURF лаплассиан гауссианы аппроксимируется бокс-фильтром:

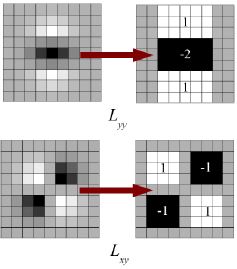


Рис. 4: пара слева – ядро свертки производной по y и ее упрощенная аппроксимация в SURF, пара справа – производная по диагонали и ее упрощенная аппроксимация в SURF. Элементы в серых ячейках принимают значение 0.

Так как свертку с бокс-фильтром можно производить с интегральными изображениями, достигается ощутимое повышение производительности по сравнению с SIFT.

В качестве метода построения дескриптора используется гистограмма градиентов, как и в SIFT, но в нем используются взвешенные суммы по прямоугольным подокнам (хаароподобные признаки), окружающим особую точку, чтобы упростить и ускорить аппроксимацию значений градиентов.

Детектор SURF широко используется в составе систем автоматической посадки [8, 9].

Детектор ORB

ORB (Oriented FAST and Rotated BRIEF) состоит из модифицированных детектора особых точек FAST и дескриптора BRIEF.

Детектор особых точек FAST сравнивает значение яркости точки p со значением яркости окружающих его 16 пикселей: если более 8 пикселей ярче или темнее точки p, то она выбирается как особая точка:

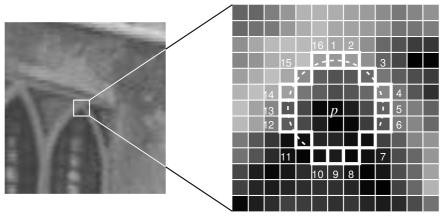


Рис. 5: детектор особых точек FAST

По умолчанию FAST не является инвариантным относительно масштаба и поворота, поэтому в детекторе ORB FAST применяется к пирамиде масштабов исходного изображения, затем применяется “мера угловатости” детектора Харриса, чтобы выделить N наиболее характерных точек на разных масштабах. Так обеспечивается инвариантность относительно масштаба.

Окружность радиуса , используемая в детекторе особых точек, интерпретируется как окрестность точки p, для которой определются моменты и главное направление:

BRIEF В качестве дескриптора использует битовый вектор, определяемый локальным бинарным паттерном (Local Binary Pattern). Пусть

Тогда выражение:

определяет LBP-код длиной n бит. В BRIEF, мдоифицированном для ORB, предлагается использовать ранее вычисленное направление для вычисления дескриптора, инвариантного к вращению, на основе локального бинарного паттерна.

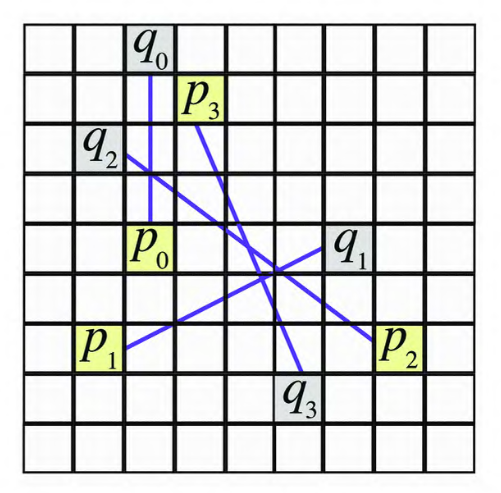


Рис. 6: локальный бинарный паттерн со случайно выбранными парами

ТЕСТОВЫЕ ВИДЕОПОСЛЕДОВАТЕЛЬНОСТИ

МЕТОДИКА ВЫПОЛНЕНИЯ РАБОТЫ

Для оценки различных детекторов и архитектур необходимо провести пробные запуски детекторов для каждой видеопоследовательности, указанной ранее.. Пусть D – набор детекторов, V – набор видео, A – набор архитектур. Тогда оценка будет производиться следующим образом:

Для каждого d из D:

Для каждого a из A:

Для каждого v из V:

если d не поддерживает a:

прервать итерацию, начать следующую

вычислить fps(d, a, v)

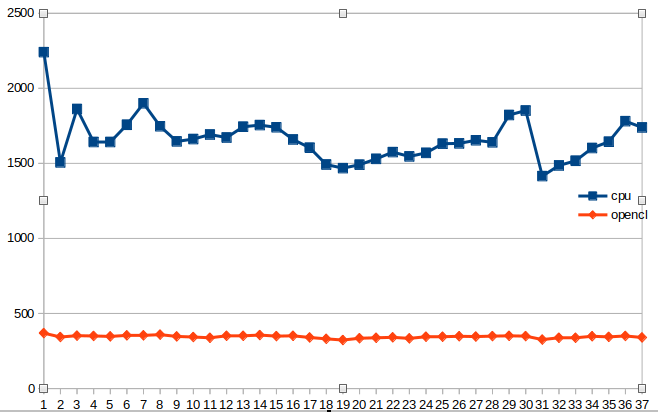
вычислить avgTrace(d, a, v)

вычислить maxTrace(d, a, v)

Где:

* fps(d, a, v)- среднее количество кадров в секунду, обрабатываемых детектором d на архитектуре v для видео d
* avgTrace(d, a, v)- среднее количество кадров, на которых удалось отследить большинство точечных особенностей (не менее 50%)
* maxTrace(d, a, v)- максимальное количество кадров, на которых удалось отследить хотя бы одну точечную особенность

ИНТЕРПРЕТАЦИЯ РЕЗУЛЬТАТОВ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Рис. 7: графики зависимости времени работы детектора SURF от номера кадра первой видеопоследовательности

ВЫВОД

СПИСОК ЛИТЕРАТУРЫ