Минобрнауки России

Федеральное государственное автономное образовательное учреждение высшего образования   
«Национальный исследовательский университет

«Московский институт электронной техники»

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

Направление подготовки — 01.03.04 «Прикладная математика»

Профиль — «Применение математических методов к решению инженерных и экономических задач»

|  |  |  |
| --- | --- | --- |
| Выполнил студент Димаков Владислав Сергеевич | | Группа: МП-40 |
| Оценка руководителя практики от кафедры ВМ-1 Козлитин Иван Алексеевич |  |  |
|  |  |  |

Москва

2016

Оглавление.

[Введение. 2](#_Toc470258317)

[Особые точки. 3](#_Toc470258318)

[Детекторы особых точек. 4](#_Toc470258319)

[Тест детекторов особых точек. 7](#_Toc470258320)

[Вычисление оптического потока методом Lucas-Kanade. 7](#_Toc470258321)

[Разработка стабилизатора потокового видеоизображения. 9](#_Toc470258322)

[Заключение. 11](#_Toc470258323)

[Список использованных источников. 11](#_Toc470258324)

# Введение.

Интерес к компьютерному зрению возник одним из первых в области искусственного интеллекта наряду с такими задачами, как автоматическое доказательство теорем и интеллектуальные игры.

Прогресс в области компьютерного зрения определялся двумя факторами: развитием теории и развитием аппаратного обеспечения. Долгое время теория и академические исследования опережали возможности практического использования систем компьютерного зрения, которое ограничивалось вычислительными ресурсами. Ведь чтобы выполнить даже простейшую обработку видеоизображения, нужно хотя бы один раз просмотреть все его пиксели, а для этого нужно выполнять тысячи операций в секунду.

Постепенно, благодаря росту производительности процессоров (а также развитию цифровых видеокамер), ситуация изменилась. Преодоление определенного порога производительности, необходимого для осуществления полезной обработки изображений за разумное время, открыло путь для большого числа приложений компьютерного зрения.

Однако действительно массовое применение методы компьютерного зрения получили лишь менее десяти лет назад, с достижением соответствующего уровня производительности процессоров у персональных и мобильных компьютеров.

Примерами применения систем компьютерного зрения могут быть:

1. Системы управления процессами (промышленные роботы, автономные транспортные средства).
2. Системы видеонаблюдения.
3. Системы организации информации (например, для индексации баз данных изображений).
4. Системы моделирования объектов или окружающей среды (анализ медицинских изображений, топографическое моделирование).
5. Системы взаимодействия (например, устройства ввода для системы человеко-машинного взаимодействия).
6. Системы дополненной реальности.

Именно широта применения систем компьютерного зрения стала для меня ключевым фактором при выборе темы практики. Ей стало: «Обнаружение и слежение за объектами в реальном времени на основе самообучающегося классификатора».

Так как данная задача требует обширных знаний в сфере компьютерного зрения, в текущем семестре было решено выполнить только ее часть, а именно разработку стабилизатора видеоизображения.

На основе этого сложился следующий список задач, которые решались в рамках выполнения индивидуального задания по практике:

1. Изучение научной и научно-технической литературы; получение общего представления о подходах к решению поставленной задачи.
2. Изучение программных средств, необходимых для решения поставленной задачи.
3. Решение учебной задачи (разработка стабилизатора видеоизображения).

# Особые точки.

Для выделения из изображения некоторой интерпретируемой информации необходимо привязаться к локальным особенностям изображения.

Особая точка или точечная особенность изображения – это точка изображения, окрестность которой можно отличить от окрестности любой другой точки изображения в некоторой другой окрестности особой точки . В качестве окрестности точки изображения для большинства алгоритмов берётся прямоугольное окно размера пикселей. Процесс определения особых точек достигается путём использования детектора и дескриптора.

Детектор – это метод извлечения особых точек из изображения. Детектор обеспечивает инвариантность нахождения одних и тех же особых точек относительно преобразований изображений.

Дескриптор – идентификатор особой точки, выделяющий её из остального множества особых точек. В свою очередь, дескрипторы должны обеспечивать инвариантность нахождения соответствия между особыми точками относительно преобразований изображений.

Особые точки обладают следующими свойствами:

* Отличимость – особая точка должна явно выделяться на фоне и быть уникальной в своей окрестности.
* Инвариантность – определение особой точки должно быть независимо к аффинным преобразованиям.
* Стабильность – определение особой точки должно быть устойчиво к шумам и ошибкам.
* Уникальность – кроме локальной отличимости, особая точка должна обладать глобальной уникальностью для улучшения различимости повторяющихся паттернов.
* Интерпретируемость – особые точки должны определяться так, чтобы их можно было использовать для анализа соответствий и выявления интерпретируемой информации из изображения.

# Детекторы особых точек.

Подходы к определению особых точек могут быть разделены на 3 категории:

* Основанные на интенсивности изображения: особые точки вычисляются напрямую из значений интенсивности пикселей изображения.
* Использующие контуры изображения: методы извлекают контуры и ищут места с максимальным значением кривизны или делают полигональную аппроксимацию контуров и определяют пересечения. Эти методы чувствительны к окрестностям пересечений, поскольку извлечение часто может быть неправильным в тех местах, где пересекаются 3 или более краев.
* На основе использования модели: используются модели с интенсивностью в качестве параметров, которые подстраиваются к изображениям-шаблонам до субпиксельной точности.

На практике для широкого применения наиболее распространены методы, основанные на интенсивности изображения.

Углы – это особые точки, которые формируются из двух или более граней. Грани обычно определяют границу между различными объектами или частями одного и того же объекта. Угол – это точка, у которой в окрестности интенсивность изменяется относительно центра . Углы определяются по координатам и изменениям яркости окрестных точек изображения. Главное свойство таких точек заключается в том, что в области вокруг угла у градиента изображения преобладают два доминирующих направления, что делает их различимыми.

Градиент – векторная величина, показывающая направление наискорейшего возрастания функции интенсивности изображения . Так как изображение дискретно, то вектор градиента определяется через частные производные по оси и через изменения интенсивностей соседних точек изображения.

**Moravec.**

Детектор Moravec – самый простой из существующих. Рассматривается изменение яркости квадратного окна (обычно размера , или пикселей) относительно интересующей точки при сдвиге окна на 1 пиксель в 8-ми направлениях (горизонтальных, вертикальных и диагональных).

Алгоритм:

1. Для каждого пикселя в изображении вычисляется изменение интенсивности:
2. Строится карта вероятностей нахождения углов в каждом пикселе изображения посредством вычисления оценочной функции:

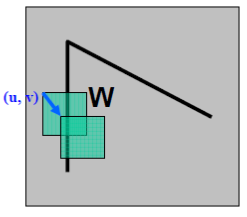
То есть определяется направление, которому соответствует наименьшее изменение интенсивности, т.к. угол должен иметь смежные ребра.

1. Производится отсечение пикселей, в которых значения ниже порогового значения .
2. Удаляются повторяющиеся углы с помощью применения процедуры поиска локальных максимумов функции отклика.

Все полученные ненулевые элементы карты соответствуют углам на изображении.

**Harris.**

Детектор Harris – улучшенный детектор Moravec. В нем рассматриваются производные яркости изображения по множеству направлений. Для изображения рассматривается окно (обычно его размер пикселей) в центре , а также его сдвиг на :



Тогда взвешенная сумма квадрата разностей между сдвинутым и исходным окном (т.е. изменение окрестности точки при сдвиге на ) равна:

где – весовая функция. Тогда в матричном виде можем записать:

где *M* – автокорреляционная матрица:

Угол характеризуется большими изменениями функции по всем возможным направлениям , что эквивалентно большим по модулю собственным значениям () матрицы *M*. Поскольку напрямую считать собственные значения является трудоёмкой задачей, Харрисом и Стефеном была предложена мера отклика:

где – эмпирическая константа, .

После вычисления *R* производится отсечение точек по найденному порогу. Далее находятся локальные максимумы функции отклика по окрестности заданного радиуса и выбираются в качестве уголковых особых точек.

**Shi-Tomasi.**

Угловой детектор Shi-Tomasi во многом совпадает с детектором Харриса, но отличается в вычислении меры отклика: алгоритм напрямую вычисляет значение , поскольку делается предположение, что поиск углов будет более стабильным.

**Förstner.**

Метод использует ту же самую меру «угловатости», что и детектор Harris, но в отличие от него собственные значения вычисляются явно. Функция отклика угла определяется следующим образом:

Также для правильности определения считается мера округлости угла, равная:

# Тест детекторов особых точек.

Мной был произведен тест различных детекторов особых точек: Harris, Shi-Tomasi, Förstner. Самыми «стойкими» к внешним факторам (освещенность, выраженность углов) оказались детекторы Shi-Tomasi и Förstner. Но так, как детектор Förstner на «зашумленных» изображениях находил большее число «неверных» особых точек, то окончательный выбор пал на детектор Shi-Tomasi.

На основе этого детектора была разработана функция, позволяющая производить поиск особых точек с различными параметрами (плотность особых точек на фрагмент изображения, порог отбора их «качества»).

# Вычисление оптического потока методом Lucas-Kanade.

В системах компьютерного зрения и обработки изображений часто возникает задача определения перемещений объектов в трехмерном пространстве с помощью оптического сенсора, то есть видеокамеры. Для этого достаточно найти смещения двухмерных проекций объектов в плоскости кадра.

Если мы хотим узнать на сколько тот или иной объект сместился по отношению к его же положению на предыдущем кадре за то время, которое прошло между фиксацией кадров, то нам необходимо воспользоваться одним из методов вычисления оптического потока.

**Метод Lucas-Kanade.**

Предположим, что значения пикселей переходят из одного кадра в следующий без изменений. Таким образом, мы делаем допущение, что пиксели, относящиеся к одному и тому же объекту, могут сместиться в какую либо сторону, но их значение останется неизменным. На математическом языке это допущение можно записать так:

где – функция яркости пикселей, и – смещение, – номер кадра в последовательности.

**Одномерный случай.**

|  |  |
| --- | --- |
| Пусть кадры имеют вид:  **https://habrastorage.org/storage2/201/6cc/59d/2016cc59d47b0dd7889a0933db6a6f1c.png**  На втором кадре изображение первого кадра смещено вправо. Именно это смещение необходимо найти. | *Представление кадров в виде функций:*  **https://habrastorage.org/storage2/2a4/52b/0a7/2a452b0a78e1525551e1b95ea29bdafb.png** |

Из графика видно, что:

Запишем разложение в ряд Тейлора:

Пусть достаточно хорошо аппроксимируется первой производной. Тогда:

Откуда:

**Двумерный случай.**

Запишем разложение в ряд Тейлора:

где вектор смещения: .

В соответствии со сделанным допущением:

Поскольку между двумя кадрами проходит единичный интервал времени, то можно сказать, что есть не что иное, как производная по времени, т.е.

Предположим, что соседние пиксели смещаются на одинаковое расстояние. Возьмем фрагмент изображения, пикселей, и условимся, что для каждого из пикселей и равны. Тогда вместо одного уравнения мы получим уравнений. Очевидно, что в общем случае система не имеет решения, поэтому будем искать такие и , которые минимизируют ошибку:

где – функция, определяющая весовые коэффициенты для пикселей.

Чтобы найти минимум воспользуемся методом наименьших квадратов.

Получим следующее уравнение в матричной форме:

где:

Если матрица обратима, то возможно вычислить и , которые минимизируют ошибку :

Таким образом, были найдены смещения пикселей на кадре.

Используя метод Lucas-Kanade, мной была реализована функция, позволяющая определять координаты заданных точек, изменяющих свое положение от кадра к кадру.

# Разработка стабилизатора потокового видеоизображения.

На основе полученных ранее функций, был разработан стабилизатор потокового видеоизображения со следующим принципом работы (приведен наиболее краткий и обобщенный вариант):

1. При инициализации детектор находит особые точки на первом кадре, полученном с камеры. Их координаты будут отслеживаться в дальнейшем.
2. Начиная со второго кадра, вычисляются отклонения отслеживаемых точек от их начальных положений.
3. Вычисляется общее смещение кадра относительно начального, как медианы выборки отклонений отслеживаемых точек.
4. Меняется область вывода изображения на экран, основываясь на вычисленном смещение кадра.

Также был реализован механизм ре-инициализации стабилизатора в случае, если числа точек, отслеживаемых на текущем кадре, недостаточно для верного определения смещения кадра.

На следующей странице демонстрируется пример работы стабилизатора.

|  |  |
| --- | --- |
|  | **Первый (инициализирующий) кадр.**  Белыми точками отмечены найденные на кадре особые точки, координаты которых будут отслеживаться в дальнейшем. |
|  | **Кадр после изменения положения камеры.**  Белыми точками отмечены новые координаты отслеживаемых точек. |
|  | **«Стабилизированный» кадр.** |

# Заключение.

Мне удалось реализовать простейший стабилизатор потокового видеоизображения, который демонстрирует хорошие результаты.

Навыки и наработки, полученные в процессе прохождения учебной практики, будут крайне полезны для написания моей квалификационной работы по теме: «Обнаружение и слежение за объектами в реальном времени на основе самообучающегося классификатора».

# Список использованных источников.

1. «Введение в компьютерное зрение» (https://www.lektorium.tv/course/22847)
2. Документация библиотеки OpenCV (http://opencv.org/documentation.html)
3. «Системы компьютерного зрения» (http://www.controlengrussia.com)
4. «Детекторы углов» (https://habrahabr.ru/post/244541/)
5. «Вычисление оптического потока» (https://habrahabr.ru/post/169055/)