МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования   
**«Национальный исследовательский   
Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

**Кафедра: ИАНИ**

Магистерская программа: «Прикладная информатика»

Профиль подготовки: «Прикладная информатика в области принятия решения»

**ОТЧЕТ**

по преддипломной практике

на тему:

**«»**

**Выполнил:** студент группы 381707-м

Губарев Сергей Юрьевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Научный руководитель:**

кандидат физико-математических наук, доцент

Чернышова Наталия Николаевна

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород  
2019

# СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 3](#_Toc2693996)

[ПОСТАНОВКА ЗАДАЧИ 4](#_Toc2693997)

[НАХОЖДЕНИЕ ВЛОЖЕННЫХ ПОЛИТОПОВ **Ошибка! Закладка не определена.**](#_Toc2693998)

[ШИРОКИЕ ПОЛИТОПЫ **Ошибка! Закладка не определена.**](#_Toc2693999)

[АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ ЦЕЛОЧИСЛЕННОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ 9](#_Toc2694000)

[ЗАКЛЮЧЕНИЕ 12](#_Toc2694001)

# ПОСТАНОВКА ЗАДАЧИ

Задачи, которые необходимо решить для достижения результатов:

1. изучить проблему предметной области
2. сформулировать задачу и определить ее место в проблемном домене
3. определить ожидаемые результаты (какой положительный эффект должен в идеале получиться).
4. формализовать задачу
5. провести анализ публичных источников, для определения уже достигнутых результатов в данной области
6. разработать и реализовать программное решение задачи
7. провести анализ полученного решения

# Формализация задачи

Суть математического решения задачи восстановления пространственного объекта по двум или более изображениям заключается в описании изменения положения камеры устройства при выполнении различных кадров. В общем случае положение определяется подъемом на высоту ∆. Предполагается, что устройства камеры, зафиксированные на жестком креплении, так же имеется шкала для съемки поверхности с разной высот. Расстояние от поверхности до объектива замеряется с высокой точностью. Главным считается то изображение, фото которого получено первым.

## Входные и выходные данные

В качестве входных параметров мы имеем[1]:

1. параметры оптической системы, при помощи которой были получены изображения поверхности микроскопического объекта (фокусное расстояние, наблюдаемая ширина в фокусе, коэффициент для вычисления абсолютной высоты фокуса). Формат файла: файл формата .camera, имя файла – [“имя проекта”.camera], файл содержит следующие значения:

* фокусное расстояние (f=”значение параметра”)
* наблюдаемая ширина в фокусе (w=”значение параметра”)
* коэффициент для вычисления абсолютной высоты фокуса (k=”значение параметра”), по умолчанию k=1.

1. набор изображений одинакового размера, полученный микросъемкой одного и того же объекта с разной высоты. Изображения должны содержаться в виде файлов типа PNG. Имя файлов должны быть в следующем формате: [“имя проекта”\_”относительная высота на которой было сделано изображение”.png]. Предельно допустимые размеры входных изображений от 4\*4 пикселей до 4K (4096\*3072 пикселя).

Входные данные должны содержаться в заданной системе директории при запуске.

ПО «Get3DModel» должно формировать следующие выходные данные[1]:

1. Файл формата OBJ, который будет содержать трехмерные координаты точек, записанные в стандартном формате. Имя файла: [3DModel.obj]
2. Файл формата PNG которое содержит восстановленное изображение объекта с высокой глубиной резкости. Имя файла: [sharpImage.png]

Выходные данные должны сохраняться в заданную системе директорию при запуске.

## Математическая модель

Есть серия изображений поверхности объекта с малой глубиной резкости. Изображения получены микросъемкой одного и того же объекта на разной высоте. Также имеется информация об оптической системе (фокусное расстояние, наблюдаемая ширина в фокусе, коэффициент для вычисления абсолютной высоты фокуса).

Допущение: Рассматриваются только непрозрачные объекты, а их изображения имеют одинаковый размер.

Задача: Необходимо определить координаты точек принадлежащих поверхности восстанавливаемых объектов, равномерно распределенных по исследуемой области, а также восстановить изображение объекта с высокой глубиной резкости.

Исходные параметры:

- количество изображений поверхности объекта

–размер полученных изображений: (m\*sпикселей)

–относительная высота оптической системы, на которой получено i-ое изображение,

F – фокусное расстояние

W– наблюдаемая ширина в фокусе

сoef - коэффициент для вычисления абсолютной высоты фокуса

- матрицакоординат эталонной модели.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | ... | m |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| ... |  |  |  |  |
| s |  |  |  |  |

- высота точки (z-координата) эталонной модели, соответствующая пикселям, ;

R – параметр равномерности, выраженный в процентах

Структура решения:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 2 | ... | m |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| ... |  |  |  |  |
| s |  |  |  |  |

Решение представляет собой матрицу Z размером m\*s.Элемент матрицы:

Ограничения на решения:

Точки восстанавливаемых объектов, высоты которых найдены алгоритмом, должны быть равномерно распределены по исследуемой области. Для этого необходимо вычислить вектор:

, где

– количество уровней равномерного распределения.

Координата вектора:

,, где

–количество областейi-ого уровня, содержащих хотя бы однуточку с найденной высотой.

–количество областейi-ого уровня, на которые делим изображение.

Поэтому исходя из параметра равномерности необходимо выполнение следующего условия:

Критерий:

Рассмотрим вектора:

Пусть , тогда сточностью до обозначения:

Пусть -множество номеров компонент вектора , при которых алгоритм нашел высоты точек 3D изображения (). Оценка решения происходит на основе сравнения найденных высот 3D изображения с высотами эталонной модели.

Необходимо минимизировать модуль среднего отклонения решения от эталонной модели (при расчете не учитываются точки, чьи высоты не найдены):

# АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ реконструкции 3D поверхности по серии изображений

Алгоритм приложения «Get3DModel»

Разработанный алгоритм является итерационным, где количество итераций равно количеству входных снимков, по которым нужно построить карту глубин.

- оптимальное значение градиента для каждого пикселя на конкретной итерации;

- значение пикселя в изображении, переведенного в монохром;

- матрица высот. Изначально все элементы в ней равны -1;

**Алгоритм оценки градиента**

Шаг 1: Принимаем входные данные: снимок, сделанный на определенной высоте.

Шаг 2: Преобразуем снимок в монохромное изображение по формуле:

Шаг 3: Вычисляем «градиент» каждого пикселя, используя определенное ядро свертки G (список ядер в приложении1):

Если матрица «окружения» не может быть полностью определена для какого-либо пикселя в силу того, что это граничный пиксель изображения (пример: , тогда заполняем нулями те позиции в матрице, которые не могут быть определены.

Пример граничного пикселя и его матрицы «окружения»:

Шаг 4: Нормируем «градиент» каждого пикселя, то есть значение, полученное на шаге 3, делим на t, где ,где

G– ядро свертки размерности n.

**Алгоритм валидации точек**

В каждом ядре есть граничная ∆, 0≤ Δ ≤1 – процент порогового значения (пока равна 0), которая получена в результате работы дополнительной программы, где был найден такой процент порогового значения ∆, что сохраняя ограничение на равномерность средняя ошибка стремится к минимуму.

Порог th вычисляется так:

th = min(grad) + ( max(grad) – min(grad) )\*∆

Все градиенты, полученные алгоритмом поиска градиента на зафиксированном ядре, которые меньше полученного порога th, не рассматриваются при расчете высоты.

**Алгоритм динамического подбора ядра**

Выбор стоит между ядрами c размерностями: 3x3; 5x5; и 7x7.

Для каждого изображения в точке (x, y) вычисляются три значения градиента с помощью трех ядер разной размерности.

Обработав все изображения, для точки (x, y) мы будем иметь три списка градиентов. Каждый список соответствует одному из ядер: 3x3, 5x5, 7x7.

Чтобы определить оптимальное ядро для точки (x, y) необходимо:

Отсортировать каждый список по возрастанию.

Отбросить в каждом списке первые 2/3 значений.

Вычислить для каждого списка дисперсию.

Ядром для точки (x, y) будет являться ядро, соответствующее списку с минимальной дисперсией.

Дисперсия вычисляется по формуле:

.

**Алгоритм вычисления высоты**

Последовательно находим матрицу градиентов для каждого изображения.

Для подсчета градиента в определенном пикселе используется ядро, найденное алгоритмом динамического подбора ядра.

После расчета градиентов текущего изображения проводим операцию сравнения найденных значений матрицы градиентов с оптимальными значениями матрицы градиентов на данном этапе:

Если , то и

**Алгоритм работы программы.**

С помощью алгоритма валидации, получаем список достоверных точек.

Динамически подбираем ядра для полученных точек.

Используя найденные ядра алгоритмом динамического подбора вычисляем высоту.

# ЗАКЛЮЧЕНИЕ

В результате исследования в данной области «Реконструкция 3D модели поверхности микроскопического объекта по серии изображений», был достигнуты следующие результаты:

1. Исследована предметная область
2. Сформулирована задача
3. Построена математическая модель
4. Исследован и разработан алгоритм решения задачи
5. Разработана архитектура приложения
6. Подготовлен тестовый базис
7. Разработана тестовая инфраструктура
8. Проведен анализ полученных результатов: найдена оптимальная комбинация параметров

В результате проделанной работы было разработано ПО, с помощью которого могут быть получены координаты точек в трехмерном пространстве, записанных в файле форма OBJ и восстановленное изображение объекта с высокой глубиной резкости – файл формата PNG.