МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

Балтийский федеральный университет им. И. Канта

Институт физико-математических наук и информационных технологий

**РЕФЕРАТ**

**Многопроцессорное программирование и оптимизация программ**

Обучающийся: Филатов Иван Дмитриевич, студент 2 курса

Направление подготовки: 02.04.03 «Математическое обеспечение и администрирование информационных систем»

Магистерская программа: Банковские информационные технологии

Калининград, 2021

**Оглавление**

[Постановка задачи 3](#_Toc89886528)

[Принцип работы 4](#_Toc89886529)

[Используемые технологии 5](#_Toc89886530)

[Схема классов 6](#_Toc89886531)

[Результаты 7](#_Toc89886532)

# Постановка задачи

Цель программы состоит в нахождении оптимального многоугольного контура, максимально точно соответствующего области на бинарной маске.

На вход программе подается набор масок с очертаниями крыш домов, найденных нейронной сетью (рис. 1, 2).

Программа преобразует каждую область произвольной формы в многоугольник с прямыми углами, который будет наиболее точно соответствовать форме здания (рис. 3).

Рис. 1. Снимок поверхности земли.

Рис. 2. Маска, выданная нейронной сетью.

Рис. 3. Результат работы программы.

# Принцип работы

Алгоритм применяется к каждой области произвольной формы.

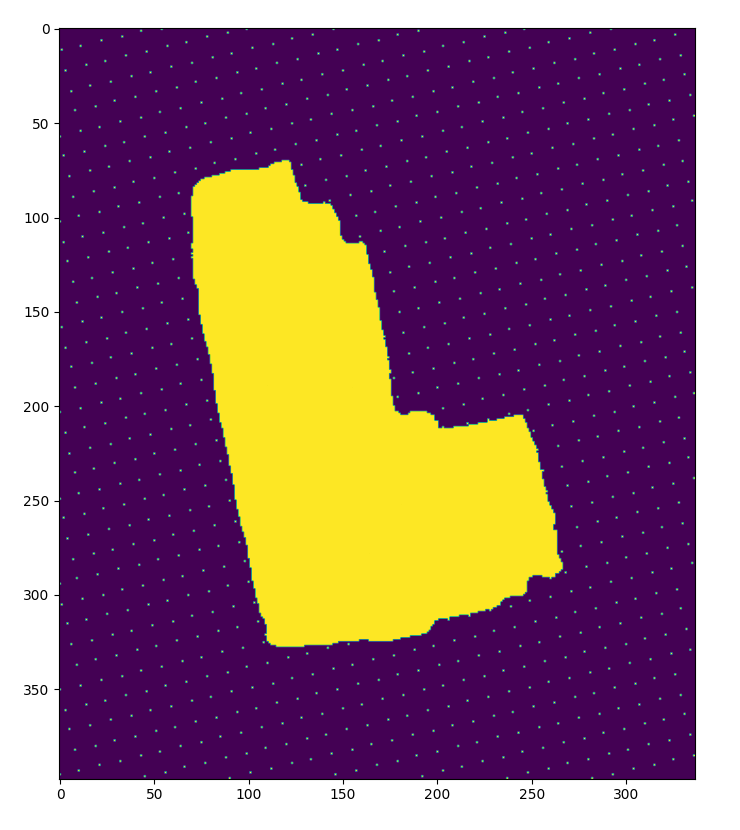
1. Область обрабатывается алгоритмом Рамера — Дугласа — Пекера, позволяющий уменьшить число точек контура области
2. На область накладываются варианты прямоугольной сетки с различными параметрами
3. Точки контура перемещаются в ближайшие узлы сетки
4. Для каждого варианта считается метрика IOU (Intersection over Union)
5. Выбирается вариант с наибольшей метрикой
6. Новая область отрисовывается на месте старой

Рис. 4. Визуализация промежуточного шага алгоритма – наложение варианта сетки на область.

# Используемые технологии

Варианты прямоугольной сетки применяются поочередно на каждую область, результат обработки одной области совершенно не зависит от результата обработки другой, что позволяет применить распараллеливание задач. Каждая область отдается на обработку в один поток.

Для распараллеливания применялась библиотека *joblib* и ее методы *Parallel, delayed*.

Синтаксис применения на языке программирования *python*:

*result = Parallel(n\_jobs)(*

*delayed(doTaskFunction)(arg) for arg in TasksForDoing*

*)*

где:

* *n\_jobs* – количество потоков для исполнения
* *doTaskFunction* – функция, выполняющая поставленную задачу над заданным набором аргументов для одной задачи
* *TasksForDoing* – список задач для выполнения

# Схема классов

В данной программе реализован ООП подход при реализации способа полигонизации контуров зданий.

В ходе написания программы был создан родительский абстрактный класс с методами, общими для любого способа полигонизации. К таким относятся алгоритмы прдобработки входного изображения.

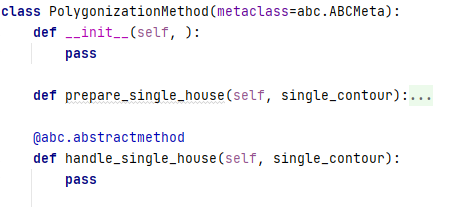
 Затем от родительского класса наследуются классы конкретного способа полигонизации, реализующие интерфейс метода *handle\_single\_house*

Рис. 5. Код родительского класса.

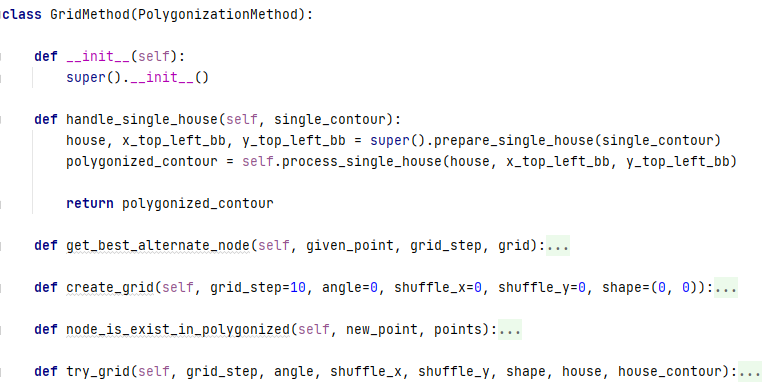


Рис. 6. Код класса-наследника, реализующий интерфейс метода *handle\_single\_house* в соответствии со способом полигонизации с помощью сеток.

# Результаты

Запуск программы осуществлялся на процессоре Intel i5-10400 (6-cores, 4.30 GHz)

Таблица ускорения:

|  |  |  |
| --- | --- | --- |
| Количество потоков | Время работы | Ускорение |
| 1 | 1964 сек. | 1 |
| 2 | 1047 сек. | 1,87 |
| 3 | 754 сек. | 2,6 |
| 4 | 581 сек. | 3,38 |
| 5 | 468 сек. | 4,0 |
| 6 | 447 сек. | 4,4 |