



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Geoinformatics Group <http://geolab.gis.land/>

Исследование и разработка алгоритмов приблизительных расчётов для растровых СУБД

Выполнил студент группы БПИ-183

Герасименко Е.Р.

Научный руководитель:

Доцент департамента программной инженерии, к.т.н.

Рамон Антонио Родригес Залепинос

Семинар состоялся в среду 30 сентября 2020 г, онлайн (Zoom), 19.40 - 20.45

Объявление и аннотация: <https://cs.hse.ru/geo/announcements/403089145.html>

Предметная область

Обработка данных операций (распространённые субд)

- Сложения, вычитания, умножения, деления
- Изменения разрешения
- Обрезка
- Вейвлет преобразования
- Сохранение промежуточных результатов с целью быстрого возвращения к произведению вычислений

над данными GeoTIFF в растровых СУБД.

Неформальная постановка задачи

Исследование и разработка формата и алгоритмов для наиболее распространённых операций над данными в растровых СУБД.

Пирамида – набор слоёв.

Слой – 4 квадранта, хранящие различную по значимости информацию о данных исходных данных матрицы, из которой он был построен.

Квадрант – двумерная матрица значений, в которой каждое значение имеет тип, аналогичный значениям исходной матрицы.

Матрица спектра – двумерная матрица значений одного спектра из мультиспектральной сцены GeoTIFF.

Мультиспектральная сцена – сцена в формате GeoTIFF, формат хранения данных спутников ДЗЗ.

GeoTIFF – формат хранения данных, расширения формата TIFF путём добавления специальных тэгов для добавления к хранимым данным геопривязки.

Цель работы

Исследование и разработка алгоритмов приблизительных расчётов для растровых СУБД с помощью вейвлетов

Задачи работы

1. Изучение формата хранения пространственных данных GeoTIFF (например сцен аппарата Landsat 8)
2. Изучение вейвлет-преобразований, видов вейвлет в них применяемых
3. Изучение основных операций в растровых СУБД при работе с геоданными
4. Разработать представление растровых данных (далее пирамида), позволяющее производить операции над геоданными с предустановленной погрешностью
5. Разработать алгоритм преобразования данных из GeoTIFF в пирамиду
6. Разработать алгоритмы для основных операций над пирамидой с нулевой и ненулевой погрешностью (сложение, вычитание, умножение, деление, изменение разрешения, обрезка, свёртка, сохранения и загрузки результатов)
7. Произвести экспериментальную оценку времени работы алгоритмов в режиме вычислений с нулевой и ненулевой погрешностью и сравнить с временем аналогичных операций из пакетов для обработки данных в формате GeoTIFF.
8. Сформулировать выводы по работе

Обоснование актуальности работы

Данная работа является востребованной, так как её применение в различных сферах, связанных с использованием данных формата GeoTIFF в растровых СУБД позволят существенно увеличить скорость обработки информации, что может быть очень важным в различных ситуациях, например при реагировании на чрезвычайные ситуации (необходимо быстро оценить границы участка для локализации ЧС), планировании инфраструктуры, управлении сельским хозяйством, мониторинга состояния окружающей среды (растительности, земли, воды, воздуха).

АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

На данный момент операции в растровых СУБД выполняются не приближенно

Аналогов алгоритмов для ускорения операций над данными GeoTIFF на основе вейвлет-преобразований в растровых СУБД в открытом доступе нет.



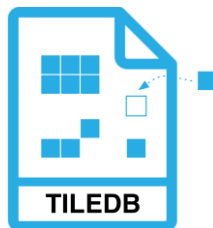
SciDB



RasDaMan



PostGIS



Intel TileDB



Google Earth Engine



ArcGIS



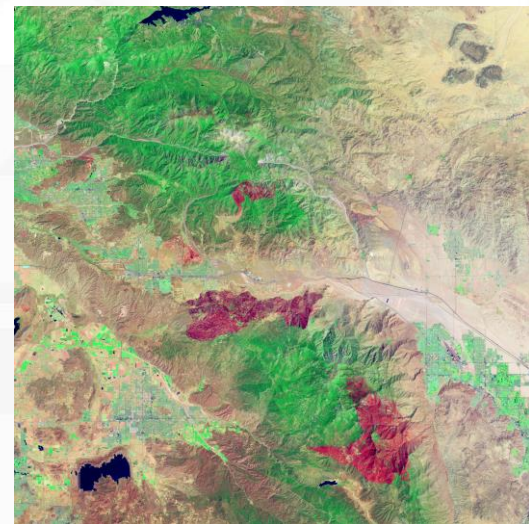
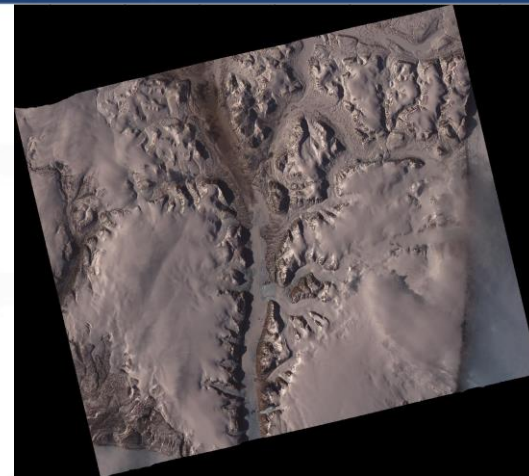
Обоснование актуальности работы

Формат GeoTIFF – это открытый формат представления растровых данных в формате TIFF совместно с данными о ГЕОпривязке, другими словами – открытый стандарт хранения картографической информации.

GeoTIFF расширяет набор тегов используемых форматом TIFF для привязки к географическим координатам, работы с картографическими проекциями и другими свойствами.

GeoTIFF предназначен и в основном используется для хранения данных с спутников ДЗЗ, так как выступает в качестве надстройки к формату TIFF, позволяет хранить мультиспектральные данные и имеет множество утилит для работы с ним.

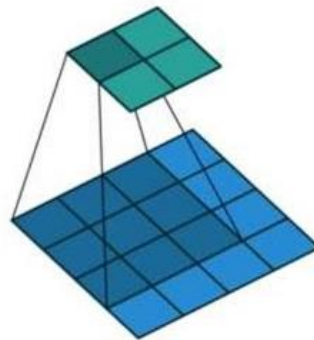
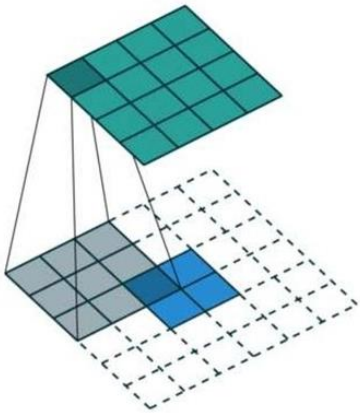
- Данный формат имеет несколько недостатков:
 - перед тем как производить какие-либо операции необходимо преобразовать его в TIFF и отдельно сохранить данные о геопривязке, чтобы после обработки была возможность их восстановить;
 - применение операции свёртки также полностью очищает какие-либо данные о геопривязке и для их восстановления необходимо произвести пересчёт, так как при применении вейвлет-преобразования происходит не только изменение значений, но и изменение размеров матрицы.



Вейвлет преобразование представляет собой наложение матрицы (вейвлета) на обрабатываемую матрицу и выполнение попарной операции с сохранением последующего результата каждой ячейки вейвлета в новую матрицу.

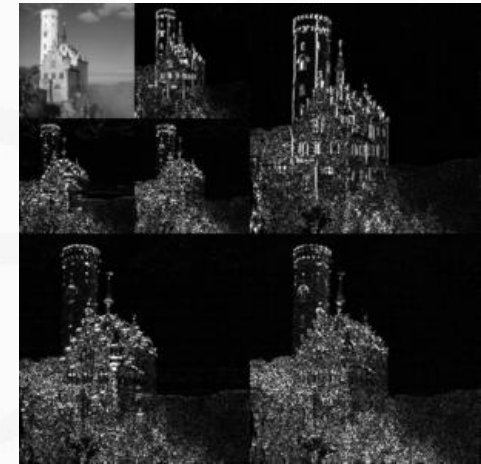
Существует множество видов вейвлет, например Добеши, Хаара и другие.

Также вейвлет-преобразования бывают без буфера и с буфером



Вейвлет-преобразование с буфером

Вейвлет-преобразование без буфера



Вейвлет Хаара



Вейвлет Добеши 9/7

Алгоритм уменьшения разрешения

Algorithm 5 Алгоритм уменьшения разрешения в 2^k раз

Input:

pyramid ▷ Пирамида, элементы – слои соответствующих уровней
 ▷ 0 – наибольший, $n - 1$ – наименьший

Output:

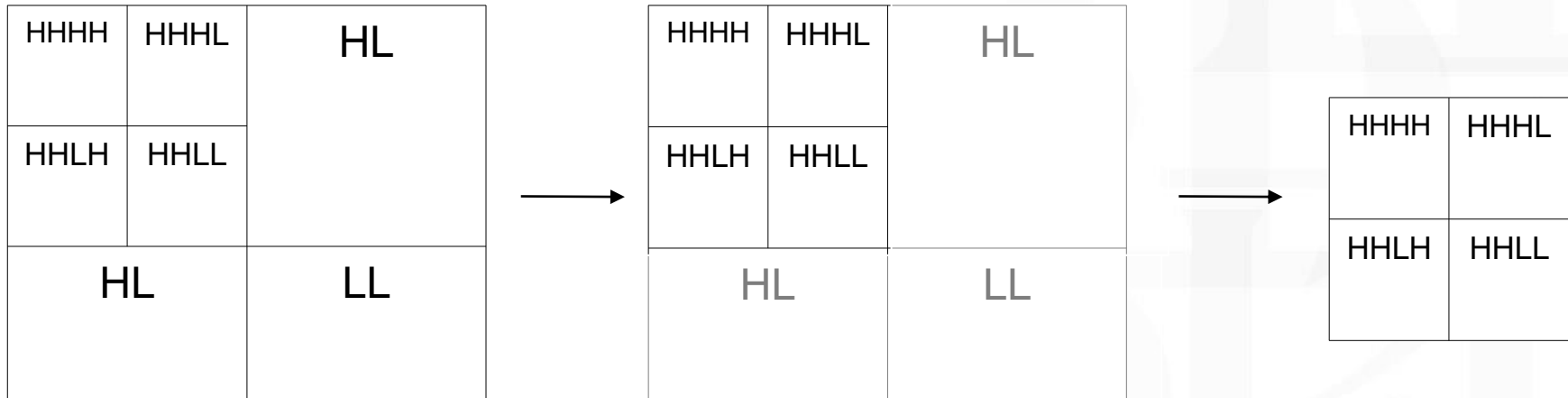
\mathbb{P} ▷ Пирамида, содержащая слои начиная с k -го уровня

function DOWNSAMPLING(*pyramid*, k)

```

P = new array[n - k]
for i ∈ [k, n) do
    P[i - k] = pyramid[i]
return P
    
```

Уменьшение разрешения производится путём отбрасывания нижних слоёв пирамиды.



Algorithm 7 Алгоритм обрезания Пирамиды

Input:

- $pyramid$ ▷ Пирамида
- x ▷ Столбец, с которого начинается интересующая область
- y ▷ Строка, с которой начинается интересующая область
- w ▷ Ширина, интересующей области
- h ▷ Высота, интересующей области

Output:

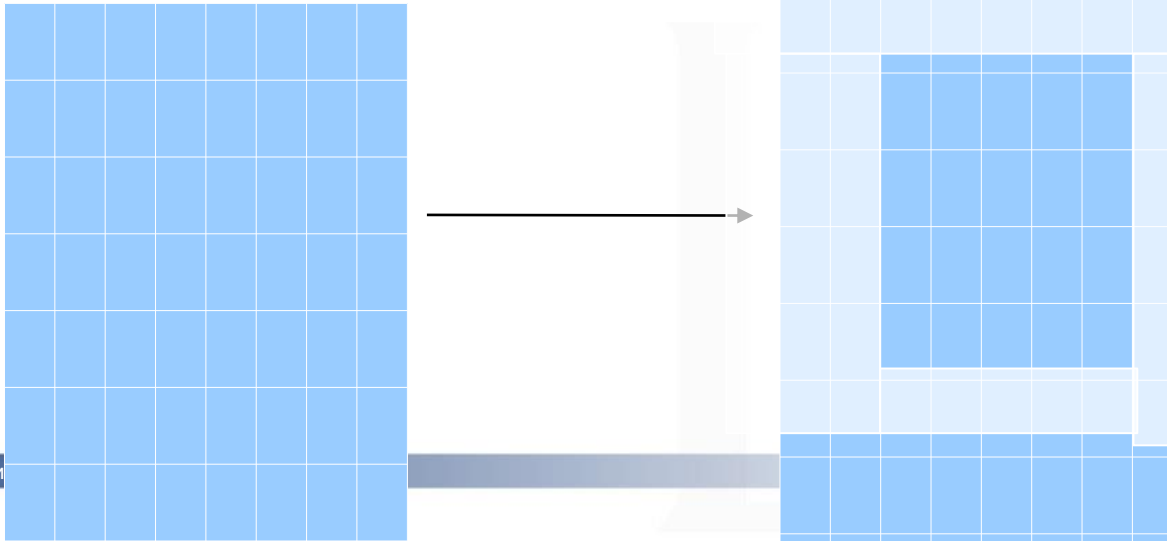
- P ▷ Пирамида, содержащая слои начиная с 0-го уровня

```

function CROP( $pyramid, x, y, w, h$ )
  for  $i \in [0, n)$  do
     $x' = x \gg i$ 
     $y' = y \gg i$ 
     $P.append(pyramid[k][x' : x' + w \gg i, y' : y' + h \gg i])$ 
  return  $P$ 

```

При обрезке происходит изменение границ, что приводит к обозначению более неиспользованных значений в качестве отброшенных и в будущем над ними операции не производятся.



Структура файла для промежуточного хранения данных

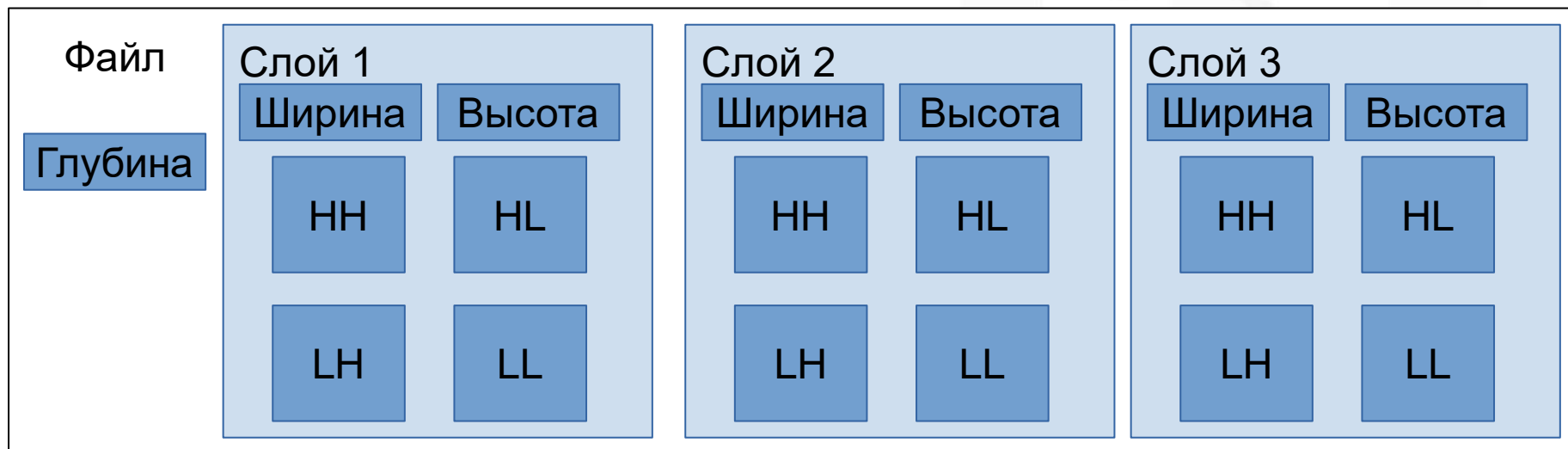
Первые 4 байта - глубина пирамиды.

Затем для каждого слоя пирамиды:

2 числа по 4 байта - ширина и высота слоя

Информация о значениях, хранимых внутри каждой ячейки для каждого квадранта данного слоя в порядке обхода квадрантов (НН НL LH LL).

При наличии незавершённой операции указывается тип операции и сериализуются параметры, необходимые для продолжения операции - параметры операции и блок, начиная с которого операция ещё не была применена.



Алгоритмы чтения и записи данных

Algorithm 9 Алгоритм Записи

Input:

$A : \{L_1, L_2, \dots, L_N\}$

▷ Пирамида закодированная N уровнями

procedure WRITEPYRAMID(A)

```

write(N)
write(state)
for  $l \in [0, N)$  do
  write( $A[l].getHeight$ )
  write( $A[l].getWidth$ )
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
      write( $A[l].HH[i][j]$ )
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
      write( $A[l].LH[i][j]$ )
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
      write( $A[l].HL[i][j]$ )
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
      write( $A[l].LL[i][j]$ )

```

Algorithm 10 Алгоритм Чтения

Output:

$A : \{L_1, L_2, \dots, L_N\}$

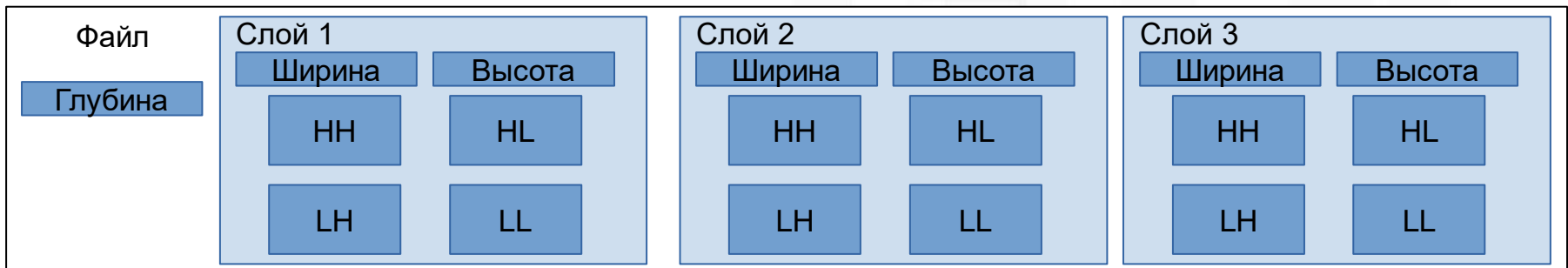
▷ Пирамида закодированная N уровнями

function READPYRAMID

```

N=reade()
for  $l \in [0, N)$  do
   $A[l].setHeight=reade()$ 
   $A[l].setWidth=reade()$ 
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
       $A[l].HH[i][j]=reade()$ 
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
       $A[l].LH[i][j]=reade()$ 
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
       $A[l].HL[i][j]=reade()$ 
  for  $i \in [0, A[l].getHeight)$  do
    for  $j \in [0, A[l].getWidth)$  do
       $A[l].LL[i][j]=reade()$ 
return  $A$ 

```



Анализ и тестирование алгоритмов

Таблица 1: Производительность 2-d операций (10^{-3} сек.)

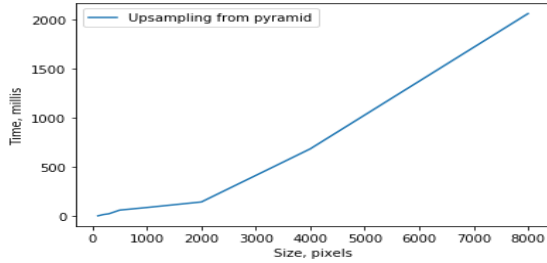
Операция	Исходные данные (пикселей × пикселей)	Без потерь (миллисекунд)	С потерями (миллисекунд)
Преобразование из GeoTIFF	256 × 256	15.6	-
Преобразование из GeoTIFF	512 × 512	68.6	-
Преобразование из GeoTIFF	1024 × 1024	99.5	-
Преобразование из GeoTIFF	7838 × 7718	2329	-
Пирамида (7 уровней)	256 × 256	7.4	-
Пирамида (7 уровней)	512 × 512	37.2	-
Пирамида (7 уровней)	1024 × 1024	49.1	-
Пирамида (7 уровней)	7838 × 7718	412	-
Суммирование	256 × 256	15	4
Суммирование	512 × 512	54	9.6
Суммирование	1024 × 1024	87.2	23.1
Суммирование	7838 × 7718	1754	481
Произведение	256 × 256	17	3.1
Произведение	512 × 512	62	8.3
Произведение	1024 × 1024	93	24.5
Произведение	7838 × 7718	1841	352
Downsampling	произвольный	0	-

Операция	Исходные данные (пикселей × пикселей)	Без потерь (миллисекунд)	С потерями (миллисекунд)
Upsampling	256 × 256	13	-
Upsampling	512 × 512	47	-
Upsampling	1024 × 1024	132	-
Crop	произвольный (вырезаем 100 × 100)	7	-
Conv 3x3	256 × 256	167	-
Conv 3x3	512 × 512	158	-
Conv 3x3	1024 × 1024	172	-
Conv 3x3 with buff 1	256 × 256	173	-
Conv 3x3 with buff 1	512 × 512	169	-
Conv 3x3 with buff 1	1024 × 1024	177	-

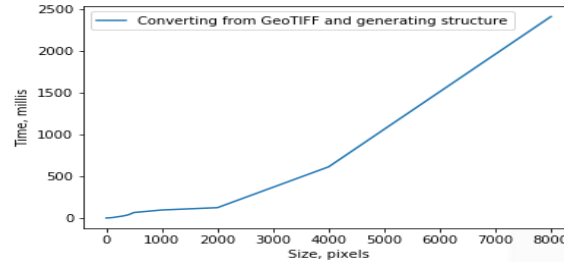


Анализ и тестирование алгоритмов

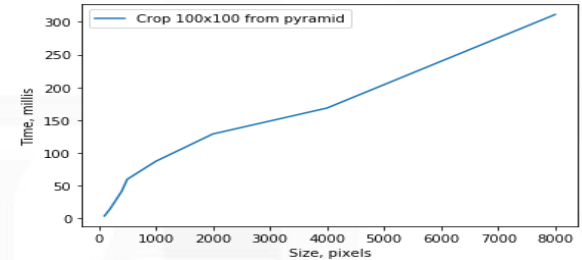
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ



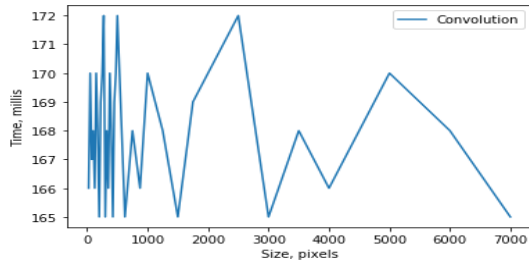
Увеличение разрешения
 $O(N^2)$



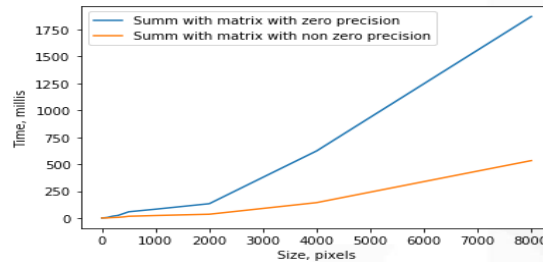
Приведение к пирамиде
 $O(N^2)$



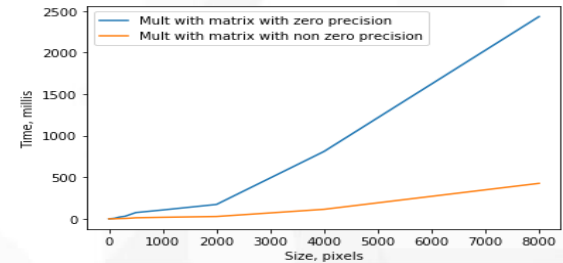
Обрезка 100x100 пикселей
 $O(N)$



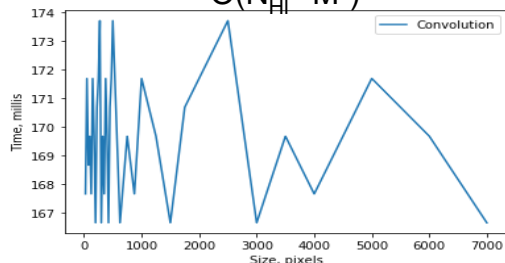
Свёртка без буфера
 $O(N_{HI}^2 * M^2)$



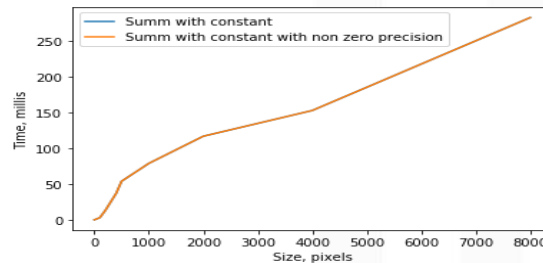
Сложение с матрицей



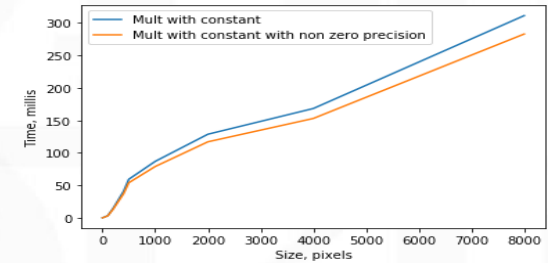
Умножение на матрицу



Свёртка с буфером
 $O(N_{HI}^2 * M^2)$



Сложение с константой
 $O(N_{HI}^2)$



Умножение на константу
 $O(N_{HI}^2)$



ВЫВОДЫ и ПУТИ ДАЛЬНЕЙШЕЙ РАБОТЫ

Практическая значимость

Данное представление, при применении в различных сферах, связанных с использованием данных формата GeoTIFF в растровых СУБД, позволят существенно увеличить скорость обработки информации

Намечено два направления развития:

- 1) Разработка новых операций для полученного представления и увеличение производительности уже разработанных операций.*
- 2) Создание алгоритмов для оптимизации переданной цепочки операций путём объединения операций и отброса заведомо неиспользуемых уровней.*

- 1) How Satellite And Imaging Technologies Are Changing The World As We Know It (Part 2) [Электронный ресурс]. Режим доступа: <https://www.forbes.com/sites/truebridge/2016/05/06/how-imaging-technologies-are-changing-the-world-part-2/> , свободный. (дата обращения: 20.12.19).
- 2) 80 Data Visualization Examples Using Location Data and Maps [Электронный ресурс]. Режим доступа: <https://carto.com/blog/eighty-data-visualizations-examples-using-location-data-maps/> , свободный. (дата обращения: 20.12.19).
- 3) Cudre-Mauroux P. [и др.]. A demonstration of SciDB: A science-oriented DBMS // Proc. of VLDB Endowment. 2009. Т. 2, No 2. с. 1534–1537.
- 4) Baumann P. [и др.]. The Array Database That Is Not a Database: File Based Array Query Answering in rasdaman // SSTD 2013.
- 5) PostGIS Raster Data Management. http://postgis.net/docs/manual-2.2/using_raster_dataman.html , Режим доступа: свободный. (дата обращения: 15.10.17).
- 6) Papadopoulos S. [и др.]. The TileDB Array Data Storage Manager // Proceedings of the VLDB Endowment. Т. 10. 2016. С. 349–360.
- 7) Rodrigues Zalipynis Ramon Antonio. ChronosDB: Distributed, File Based, Geospatial Array DBMS // PVLDB. 2018. Т. 11, No 10. с. 1247–1261. URL: <http://www.vldb.org/pvldb/vol11/p1247-zalipynis.pdf>.

- 8) Google Earth Engine: Planetary-scale geospatial analysis for everyone / Noel Gorelick, Matt Hancher, Mike Dixon [и др.] // Remote Sensing of Environment. 2017.
- 1) ArcGIS for Server | Image Extension [Электронный ресурс]. Режим доступа: <http://www.esri.com/software/arcgis/arcgisserver/extensions/image-extension> , свободный. дата обращения: 20.12.19).
- 2) Daubechies Ingrid. The wavelet transform, time-frequency localization and signal analysis // IEEE transactions on information theory. 1990. Т. 36, No 5. С. 961–1005.
- 3) Rao Raghuvver. Wavelet transforms // Encyclopedia of Imaging Science and Technology. 2002.
- 4) Image coding using wavelet transform / Marc Antonini, Michel Barlaud, Pierre Mathieu [и др.] // IEEE Transactions on image processing. 1992. Т. 1, No 2. С. 205–220.
- 5) Stanković Radomir S, Falkowski Bogdan J. The Haar wavelet transform: its status and achievements // Computers & Electrical Engineering. 2003. Т. 29, No 1. С. 25–44.
- 6) Vitter Jeffrey Scott, Wang Min, Iyer Bala. Data cube approximation and histograms via wavelets // ckm / Citeseer. Т. 98. 1998. С. 96–104.
- 7) Synopses for massive data: Samples, histograms, wavelets, sketches / Graham Cormode, Minos Garofalakis, Peter J Haas [и др.] // Foundations and Trends® in Databases. 2011. Т. 4, No 1-3. С. 1–294.
- 8) Jestes Jeffrey, Yi Ke, Li Feifei. Building wavelet histograms on large data in MapReduce // Proceedings of the VLDB Endowment. 2011. Т. 5, No 2. С. 109–120.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 17) “Information technology – digital compression and coding of continuous-tone still image – requirements and guidelines,” // CCITT, T.81, September 1992.
- 10) GeoTIFF. Режим доступа: <https://trac.osgeo.org/geotiff/> , свободный. (дата обращения: 20.12.19).
- 1) Simple Wavelet transform [Электронный ресурс]. Режим доступа: https://cs8.pikabu.ru/post_img/2017/08/02/8/1501676400179810564.jpg, свободный. (дата обращения: 20.12.19).
- 2) Landsat-8 [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/Landsat-8> , свободный. (дата обращения: 20.12.19).
- 3) Изображения в ArcGIS. Современный подход. Новый взгляд. [Электронный ресурс]. Режим доступа: <https://learn.arcgis.com/ru/arcgis-imagery-book/> , свободный (дата обращения: 20.12.19).
- 4) Исследование и разработка подходов приближённых вычислений для растровых СУБД,” / НИРС, 2020г.



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Спасибо за внимание!

Герасименко Е.Р.,
ergerasimenko@edu.hse.ru

Москва - 2019