

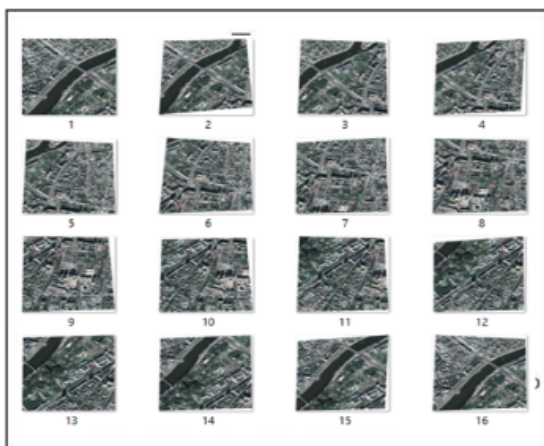
ДОДАТОК А
Слайди презентації

Дослідження методів розпізнавання за допомогою нейронної мережі щодо аналізу та побудови карт з використанням GPS треку з прив'язками до візуальних даних

Виконала:
ст.гр. ІПЗм 17-1 Черепакіна К.К.
Керівник:
доц. каф. ПІ Назаров О.С.

1

Мета роботи



Мета полягає в створенні розумної системи, здатної працювати в режимі реального часу, та виконувати не тільки безпомилкове опрацювання великої кількості зображень в одне, та й ще враховувати властивості поверхні (такі як гори та інше), вміти розпізнавати об'єкти, яким система навчена, та вміти робити це, асинхронно, тобто без чіткої послідовності.

2

Суть проблеми

Суть проблеми - створення окремих функціональних ядер, опрацювання зображень, завдяки мікроконтролеру та серверу у зв'язку з ним, а саме розподілення функції колаборацій мапи з зображень, що отримуються в "РРЧ", на стороні контролера, та послідовної або асинхронної функцій розпізнавання об'єктів (хатинок, доріг, тощо) на сервері, важливо що це, може виконуватись також в "РРЧ", а система має вміти працювати як енергоефективно, коли серверна частина опрацьовує відкладено, й у режимі усе й одразу, коли є безпосередній зв'язок з сервером, що аналізує та накладає мета теги тотчас.

3

Вимоги до контролера та сервера:

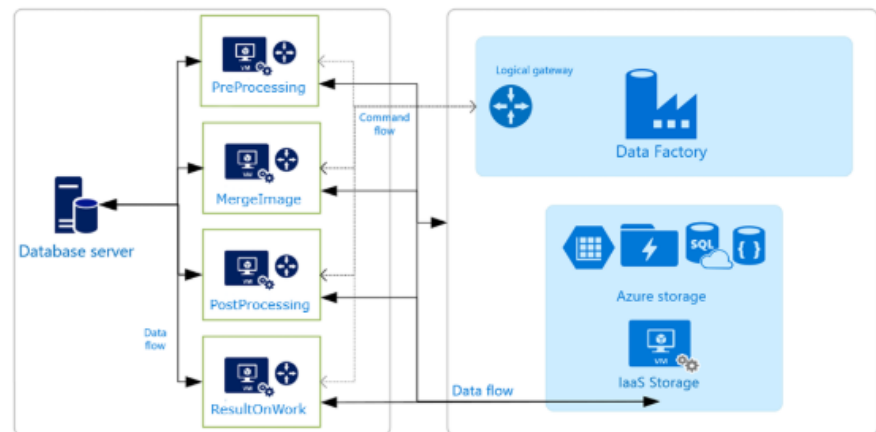
- Контроллер достаточно мощный, чтоб опрацювати декілька "стрімів" зображень;
- Контроллер має модуль зв'язку з сервером;
- Сервер має усі статуси контролерів, аби розуміти коли підключатись;
- Трансляція напів зібраних частин мапи до серверу через інтернет;
- Сервер вміє помічати мета теги на сирому зображення а не тільки на готовому.

А також дві що є зависимими вимоги:

- Надійний зв'язок між компонентами у режимі "усе й одразу";
- Відносно широко навчена нейронна мережа, щодо пошукових об'єктів .

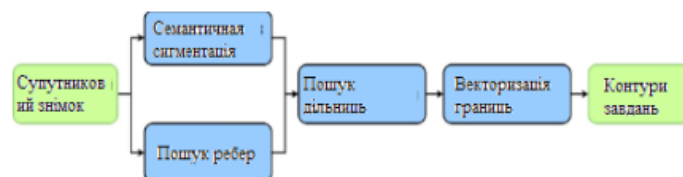
4

Схема усього “поточку” ядра з MDF



5

Первична обработка зображення задля виявлення контурів зв'язку:



6

Етапи рішення:

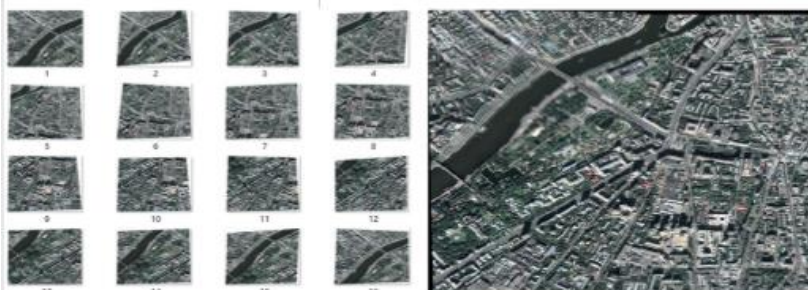
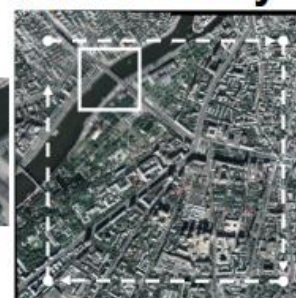
Використання математичного аналізу - хоч у системі використовується нейронна мережа, котра в теорії може бути навчання на зклейку зображень, поте рішення цієї задачі, дало не тільки покращений перфоменс етапу а й можливість бути виконаним на малопотужний частині - контроллер

Використовування нейронної мережі - етап який був вирішений завдяки другому відомому методу "навчання з вчителем", що дало можливість у подальшому додавати об'єкти котрі потрібно метатегувати (шукати та помічати) на мапі.

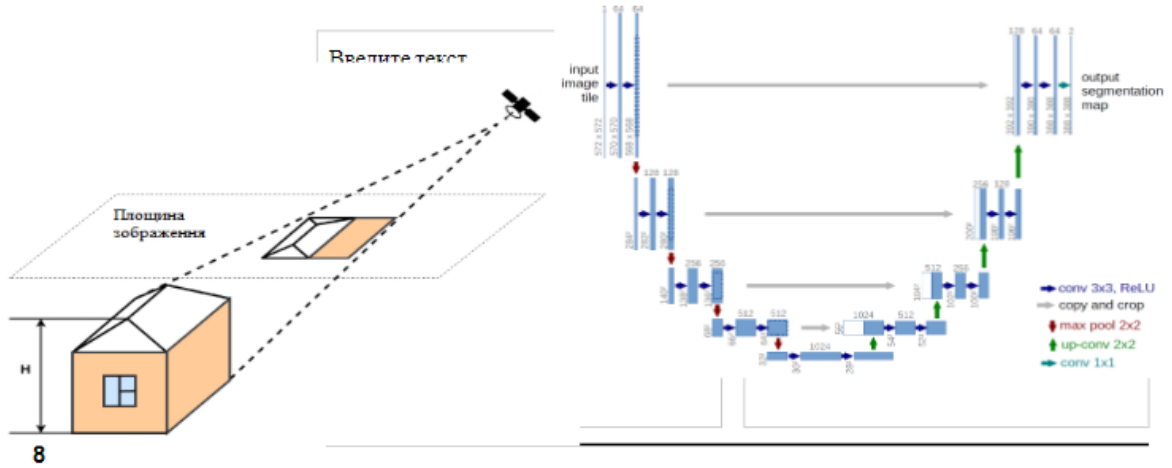
7

Тестова задача "першого" етапу

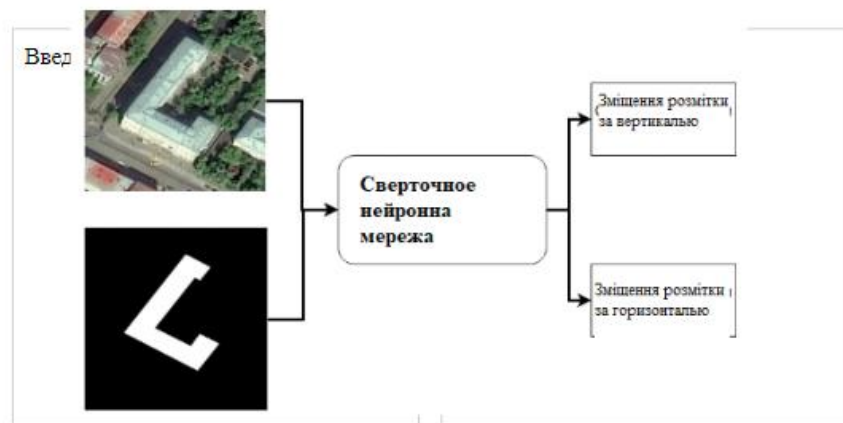
Введіть текст



Тестова задача "другого" етапу



Міжетапний результат

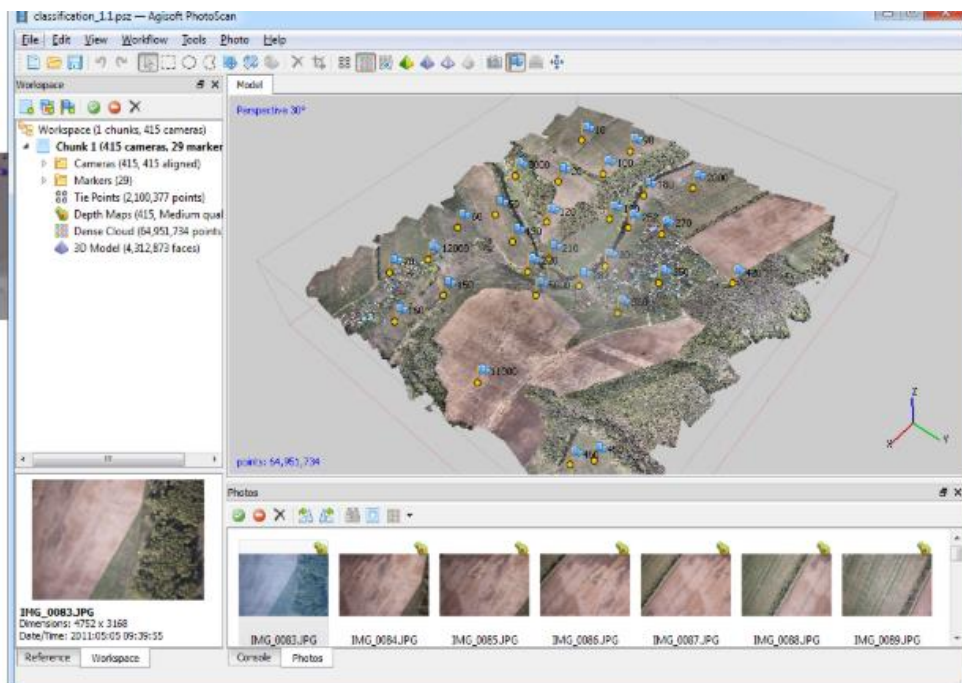


Результат
пошуку на
сирой та
готовій мапі

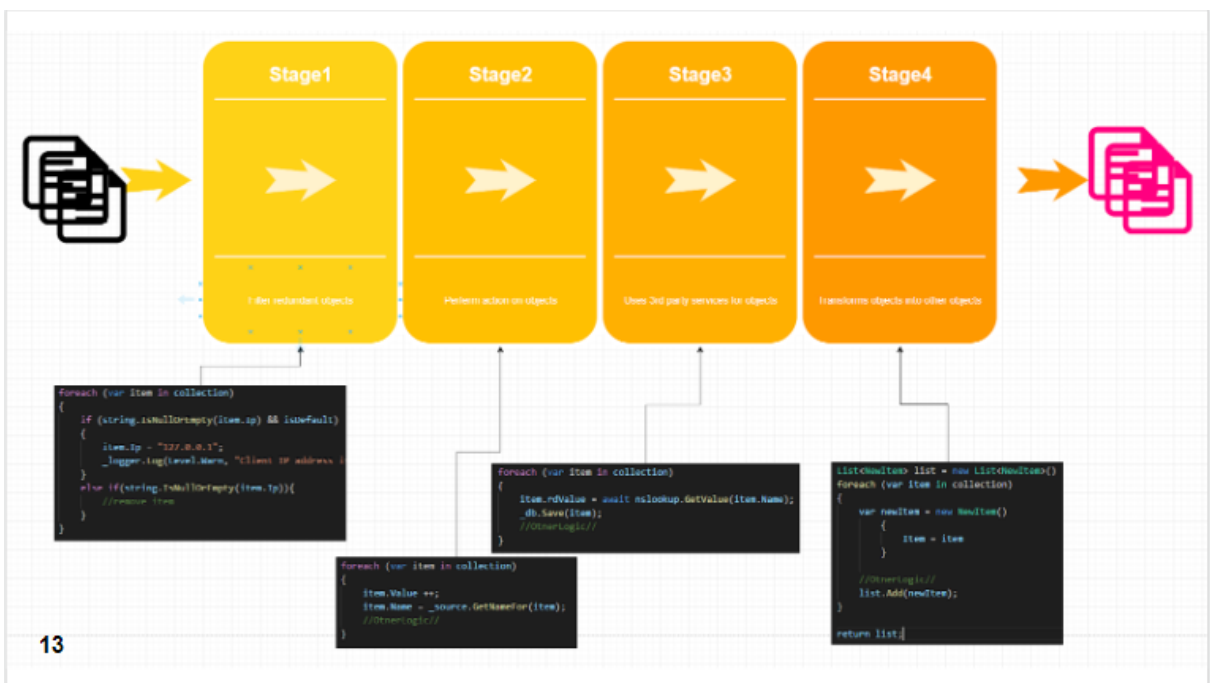
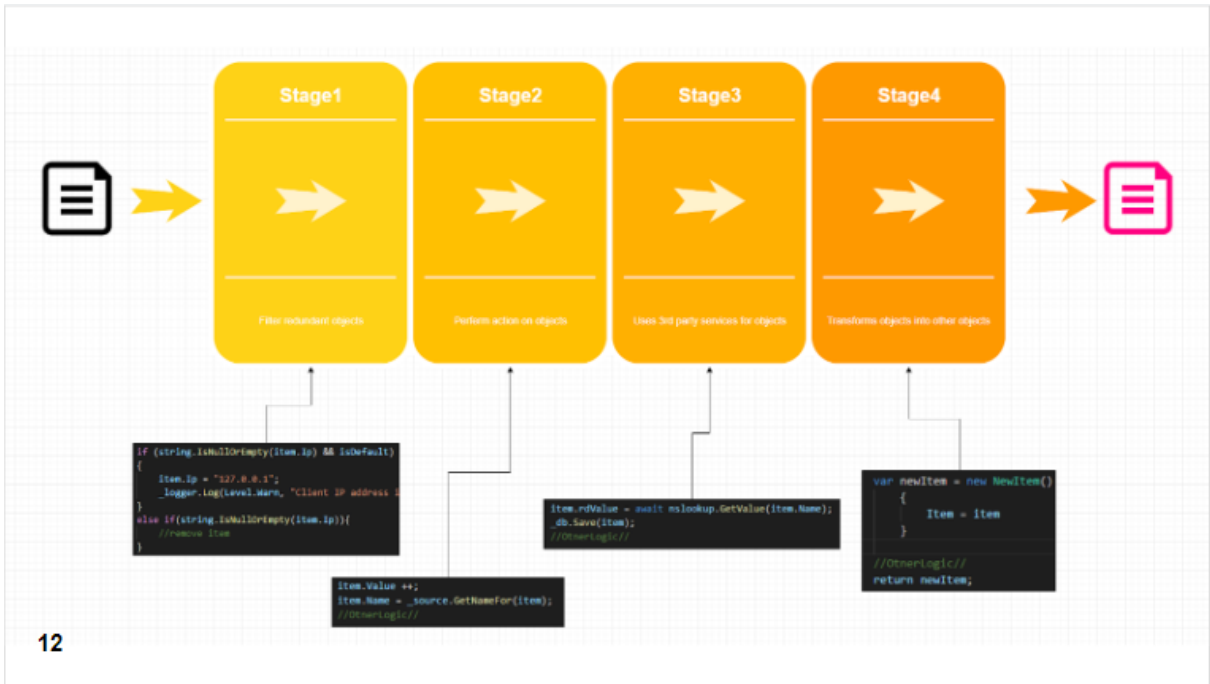
Введіть те



10



11



14



Висновки:

Підсумком дипломної роботи, була розроблена система, здатна працювати у кількох режимах, та спроможна, бути розподіленою, на логічні модулі системи, що виконуються на різних типах процесорів, та виконують основну поставлену задачу - "створення карти з зображенням, та пошук різного типу об'єктів", що є дуже складною проте вирішеною вже задачею. Система може бути інтегрована як наприклад з дронами так й використовуватись окремою системою.

16

ДОДАТОК Б**Лістинг коду програми**

```
public class Wind : Utils
{
    Wind self;
    public float speed;
    public float direction;
    public float turbulence;
    public double cross_section;
    public double turbulence_time_constant;
    public DateTime tlast;
    public double turbulence_mul;

    public Wind(string windstring, double cross_section = 0.1)
    {
        self = this;

        string[] a = windstring.Split(',');
        if (len(a) != 3)
        {
            return;
        }
        self.speed = float.Parse(a[0]);
        self.direction = float.Parse(a[1]);
        self.turbulence = float.Parse(a[2]);
        self.cross_section = cross_section;

        self.turbulence_time_constant = 5.0;

        self.tlast = DateTime.Now;

        self.turbulence_mul = 1.0;
    }

    public void current(double deltat, out double speed, out
double direction)
    {
        if (deltat == 0)
        {
            DateTime tnow = DateTime.Now;
            deltat = (tnow - self.tlast).TotalSeconds;
            self.tlast = tnow;
        }

        double w_delta = sqrt(deltat)*(1.0 - new
GaussianRandom().NextGaussian(1.0, self.turbulence + 0.01));
```

```

        w_delta -= (self.turbulence_mul -
1.0)*(deltat/self.turbulence_time_constant);
        self.turbulence_mul += w_delta;
        speed = self.speed*(float) fabs(self.turbulence_mul);

        direction = self.direction;
    }

    public Vector3 drag(Vector3 velocity, double deltat = 0) //,
testing=None)
    {

        double speed, direction;
        self.current(deltat, out speed, out direction);

        Vector3 w = toVec(speed, Utils.radians(direction));

        double obj_speed = velocity.length();

        double d = w.length()*obj_speed;
        if (d == 0)
        {
            alpha = 0;
        }
        else

            double rel_speed, beta;
            apparent_wind(speed, obj_speed, alpha, out rel_speed,
out beta);

            Vector3 relWindVec = toVec(rel_speed, beta +
Utils.atan2(velocity.y, velocity.x));

            return new Vector3(acc(relWindVec.x, drag_force(self,
relWindVec.x))
                , acc(relWindVec.y, drag_force(self, relWindVec.y))
                , 0);
        }

        public void apparent_wind(double wind_sp, double obj_speed,
double alpha, out double rel_speed, out double beta)
        {
            double delta = wind_sp*Utils.cos(alpha);
            double x = Math.Pow(wind_sp, 2) + Math.Pow(obj_speed, 2)
+ 2*obj_speed*delta;
            rel_speed = Utils.sqrt(x);
            if (rel_speed == 0)
            {
                beta = Math.PI;
            }
            else

```

```

        {
            beta = acos((delta + obj_speed)/rel_speed);
        }
    }

    public double drag_force(Wind wind, double sp)
    {
        return (Math.Pow(sp, 2.0))*0.1*wind.cross_section;
    }

    public double acc(double val, double mag)
    {
        if (val == 0)
        {
            return mag;
        }
        else
        {
            return (val/Utils.fabs(val))*(0 - mag);
        }
    }

    public Vector3 toVec(double magnitude, double angle)
    {
        Vector3 v = new Vector3(magnitude, 0, 0);
        Matrix3 m = new Matrix3();
        m.from_euler(0, 0, angle);
        return m.transposed()*v;
    }
}

public delegate void ProgressEventHandler(int progress, string
status);

public abstract class Hil
{
    internal static readonly ILog log =
LogManager.GetLogger(MethodBase.GetCurrentMethod().DeclaringType);

    [StructLayout(LayoutKind.Sequential, Pack = 1)]
    public struct sitl_fdm
    {
        public double latitude, longitude; // degrees
        public double altitude; // MSL
        public double heading; // degrees
        public double speedN, speedE; // m/s
        public double xAccel, yAccel, zAccel; // m/s/s in body
frame
        public double rollRate, pitchRate, yawRate; //
degrees/s/s in earth frame
        public double rollDeg, pitchDeg, yawDeg; // euler
angles, degrees
        public double airspeed; // m/s
        public UInt32 magic; // 0x4c56414e
    };
};

```

```

public const float ft2m = (float) (1.0/3.2808399);
public const float kts2fps = (float) 1.68780986;

internal DateTime lastgpsupdate = DateTime.Now;
internal sitl_fdm[] sitl_fdmbuffer = new sitl_fdm[5];
internal sitl_fdm oldgps = new sitl_fdm();

internal int gpsbufferindex = 0;

internal int REV_pitch = 1;
internal int REV_roll = 1;
internal int REV_rudder = 1;
internal int GPS_rate = 200;

internal int rollgain = 10000;
internal int pitchgain = 10000;
internal int ruddergain = 10000;
internal int throttlegain = 10000;

public float roll_out, pitch_out, throttle_out, rudder_out,
collective_out;

public event ProgressEventHandler Status;

public bool sitl { get; set; }
public bool heli { get; set; }
public bool quad { get; set; }
public bool xplane10 { get; set; }

public abstract void SetupSockets(int Recvport, int
SendPort, string SimIP);
public abstract void Shutdown();

public abstract void GetFromSim(ref sitl_fdm data);
public abstract void SendToSim();
public abstract void SendToAP(sitl_fdm data);
public abstract void GetFromAP();

internal void UpdateStatus(int progress, string status)
{
    if (Status != null)
        Status(progress, status);
}

internal float Constrain(float value, float min, float max)
{
    if (value > max)
    {
        value = max;
    }
    if (value < min)
    {
        value = min;
    }
}

```

```
    }  
    return value;  
}  
  
internal short Constrain(double value, double min, double  
max)  
{  
    if (value > max)  
    {  
        value = max;  
    }  
    if (value < min)  
    {  
        value = min;  
    }  
    return (short) value;  
}  
}
```

ДОДАТОК В

Наукові публікації

Accent Graphics
Publishing & Communications
Canada

Accent Graphics Communications & Publishing, Hamilton,



PREMIER Premier Publishing s.r.o.
Publishing

Центр научных исследований «Solution»

12th International conference

Science and society

7th June 2019

Hamilton, Canada
2019

The 12th International conference “Science and society” (June 7, 2019) Accent Graphics Communications & Publishing, Hamilton, Canada. 2019. 429 p.

ISBN 978-1-77192-360-6

The recommended citation for this publication is:

Busch P. (Ed.) (2019). Humanitarian approaches to the Periodic Law // Science and society. Proceedings of the 12th International conference. Accent Graphics Communications & Publishing. Hamilton, Canada. 2019. Pp. 12–17

Editor	Lucas Koenig, Austria	Morozova Natalay Ivanovna, Russia
Editorial board	Abdulkasimov Ali, Uzbekistan	Moskvin Victor Anatolevich, Russia
	Adieva Aynura Abduzhalalovna, Kyrgyzstan	Nagiyev Polad Yusif, Azerbaijan
	Arabaev Cholponkul Isaevich, Kyrgyzstan	Naletova Natalia Yurevna, Russia
	Zagir V. Atayev, Russia	Novikov Alexei, Russia
	Akhmedova Raziya Abdullayevna	Salaev Sanatbek Komiljanovich, Uzbekistan
	Balabiev Kairat Rahimovich, Kazakhstan	Shadiev Rizamat Davranovich, Uzbekistan
	Barlybaeva Saule Hatiyatovna, Kazakhstan	Shahutova Zarema Zorievna, Russia
	Bestugin Alexander Roaldovich, Russia	Soltanova Nazilya Bagir, Azerbaijan
	Boselin S.R. Prabhu, India	Spasennikov Boris Aristarkhovich, Russia
	Bondarenko Natalia Grigorievna, Russia	Spasennikov Boris Aristarkhovich, Russia
	Bogolib Tatiana Maksimovna, Ukraine	Suleymanov Suleyman Fayzullaevich, Uzbekistan
	Bulatbaeva Ayyul Abdimazhitovna, Kazakhstan	Suleymanova Rima, Russia
	Chiladze George Bidzinovich, Georgia	Tereschenko-Kaidan Liliya Vladimirovna, Ukraine
	Dalibor M. Elezovic, Serbia	Tersvadze Mzia Giglaevna, Georgia
	Gurov Valeriy Nikolaevich, Russia	Vijaykumar Muley, India
	Hajiyev Mahammad Shahbaz oglu, Azerbaijan	Yurova Kseniya Igorevna, Russia
	Ibragimova Liliya Ahmatyanovna, Russia	Zhaplova Tatiana Mikhaylovna, Russia
	Blahun Ivan Semenovich, Ukraine	Zhdanovich Alexey Igorevich, Ukraine
	Ivannikov Ivan Andreevich, Russia	
	Jansarayeva Rima, Kazakhstan	Proofreading
	Khubaev Georgy Nikolaevich	Andrey Simakov
	Khurtsidze Tamila Shalvovna, Georgia	Cover design
	Khoutyz Zaur, Russia	Andreas Vogel
	Khoutyz Irina, Russia	Contacts
	Korz Marina Vladimirovna, Russia	Premier Publishing s.r.o.
	Kocherbaeva Aynura Anatolevna, Kyrgyzstan	Praha 8 – Karlin,
	Kushaliyev Kaiser Zhalitovich, Kazakhstan	Lyčkovo nám. 508/7, PSČ 18600
	Lekerova Gulsim, Kazakhstan	1807-150 Charlton st.East,
	Melnichuk Marina Vladimirovna, Russia	Hamilton, Ontario, L8N 3×3 Canada
	Meymanov Bakyt Kattoevich, Kyrgyzstan	
	Moldabek Kulakhmet, Kazakhstan	

Material disclaimer

The opinions expressed in the conference proceedings do not necessarily reflect those of the Premier Publishing s.r.o. or Accent Graphics Communications & Publishing, the editor, the editorial board, or the organization to which the authors are affiliated.

The Premier Publishing s.r.o. or Accent Graphics Communications & Publishing is not responsible for the stylistic content of the article. The responsibility for the stylistic content lies on an author of an article.

Included to the open access repositories:

LIBRARY.RU

© Premier Publishing s.r.o.

© Accent Graphics Communications & Publishing

© Центр научных исследований «Solution»

All rights reserved; no part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the Publisher.

Typeset in Berling by Ziegler Buchdruckerei, Linz, Austria.
Printed by Premier Publishing s.r.o., Vienna, Austria on acid-free paper

14.	СРОШКІНА Т.В., МАТВІЄНКО В.П., БОРИСЕНКО С.С. ПРОБЛЕМИ БЕЗРОБІТТЯ В УКРАЇНІ ТА У СВІТІ: СОЦІАЛЬНО – ПСИХОЛОГІЧНІ УСТАНОВКИ ОСІБ, ЯКІ ВТРАТИЛИ РОБОТУ.	115
15.	ШОЛОХ М.В. АНАЛІЗ МАРКШЕЙДЕРСЬКОГО ЗБЕЗПЕЧЕННЯ РОБІТ ДЛЯ ВИЗНАЧЕННЯ СТАНУ І РУХОМОСТІ, ОБЛІКУ І НОРМУВАННЯ БАЛАНСОВО-ПРОМИСЛОВИХ ЗАПАСІВ ЗА СТУПЕНЕМ ПІДГОТОВЛЕНОСТІ ДО ВИДОБУВАННЯ НА КАР'ЄРАХ.	127
16.	ЗИМОГЛЯД Н.Ю. ФОРТЕПАННЕ ВИКОНАВСТВО 30-х р. ХХ ст. У КОНТЕКСТІ РОЗВИТКУ ПАНІСТИЧНОЇ КУЛЬТУРИ УКРАЇНИ.	140
17.	ОГАР А.В. ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ НА ПІДПРИЄМСТВАХ ГОТЕЛЬНО-РЕСТОРАННОГО ГОСПОДАРСТВА.	151
18.	МОКРОМЕНКО О.В. МІРА ПРИВАТНОЇ ІНІЦІАТИВИ, ПРОСВІТНИЦЬКОЇ ДІЯЛЬНОСТІ ЩОДО ФІНАНСУВАННЯ Й МАСОВОГО ХАРАКТЕРУ ЗАКЛАДІВ ЕЛЕМЕНТАРНОЇ ОСВІТИ У ВЕЛИКІЙ БРИТАНІЇ ХІХ СТ.	158
19.	РУМ'ЯНЦЕВА А.Ю. ОСОБЛИВОСТІ ВИКОНАВСТВА ХАРКІВСЬКИХ ПАНІСТІВ У СОЦІОКУЛЬТУРНОМУ ПРОСТОРІ ПОВОЄННОГО ПЕРІОДУ.	166
20.	КОТЛЯРОВА В.Г., ЧЕЧОТКА О.В. НАУКОВО-МЕТОДИЧНІ ПІДХОДИ ДО ПОБУДОВИ МОДЕЛІ ПРОСВІТНИЦЬКОЇ ДІЯЛЬНОСТІ ЩОДО ПОВОДЖЕННЯ З ЛІКАРСЬКИМИ КОСМЕТИЧНИМИ ЗАСОБАМИ.	178
21.	КОПЧАК І.В. ТРАГЕДІЯ МАРТИНА ІДЕНА (ЗА РОМАНОМ ДЖЕКА ЛОНДОНА "МАРТИН ІДЕН").	187
22.	ЧЕРЕПАХИНА Е.К., НАЗАРОВ А.С. АВТОМАТИЗОВАНА СИСТЕМА ДЛЯ ПОСТРОЕННЯ КАРТ С МЕТАІНФОРМАЦІЄЮ В РЕЖИМЕ РЕАЛЬНОГО ЧАСУ.	195
23.	ШЕВЦІВ З.М. ГУМАНІСТИЧНІ ЗАСАДИ ПРОФЕСІЙНОЇ ПІДГОТОВКИ МАЙБУТНІХ УЧИТЕЛІВ ПОЧАТКОВИХ КЛАСІВ ДО ІНКЛЮЗИВНОГО НАВЧАННЯ В ЗАКЛАДІ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ.	203
24.	МАННАПОВА К.Р. ТЕОРЕТИЧНИЙ АНАЛІЗ ПІДХОДІВ ДО РОЗВИТКУ ЖІТТЄСТІЙКОСТІ ПІДЛІТКІВ.	214
25.	МАЛЄЙ А.В., СТЕЦЮК В.З. МОЖЛИВІСТЬ ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ РАННЬОЇ ДІАГНОСТИКИ ДЛЯ ПРОЦЕСУ ДІАГНОСТУВАННЯ СПАДКОВИХ МЕТАБОЛІЧНИХ ЗАХВОРЮВАНЬ.	222

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ДЛЯ ПОСТРОЕНИЯ КАРТ С МЕТАИНФОРМАЦИЕЙ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ

ЧЕРЕПАХИНА Е.К.

студент кафедры Програмної інженерії

Харківській національний університет радіоелектроніки

м.Харків, Україна

НАЗАРОВ А.С.

*Доцент кафедри програмної інженерії, заступитель декана
факультета КН, член НТС, кандидат технических наук, доцент*

Харківській національний університет радіоелектроніки

м.Харків, Україна

На сегодня существует множество способов посмотреть и ознакомиться с любым уголком нашей планеты при помощи различных способов, будь то просмотр фото, снятых со спутника, либо картографическая фиксация поверхности участков земли, но сложность и стоимость создания новых карт велика. Начиная от процесса фиксации и склейки новых материалов, заканчивая обнаружением на них объектов и аномалий.



(1)

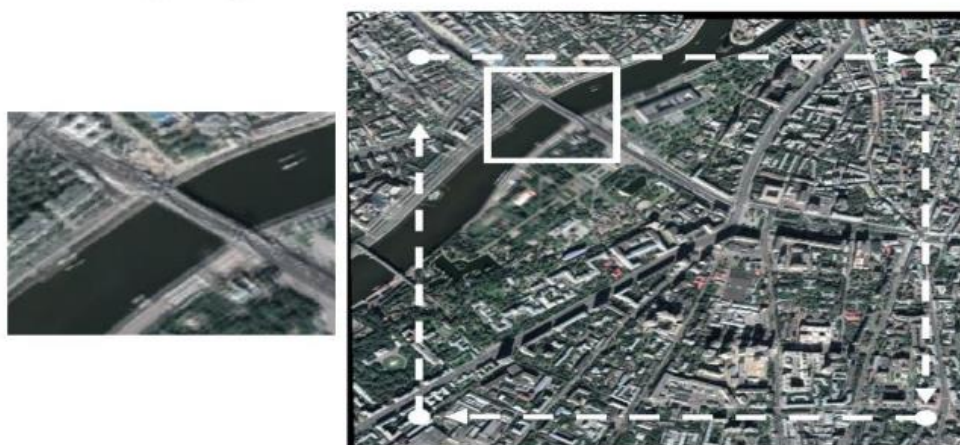
Первая проблема панорамирования заключается в построении одного составного изображения на основе набора исходных изображений (рис. 1). Она находит применение в решении таких практических задач, как:

- зондирование поверхности Земли со спутника или с дрона;
- склейка изображений, полученных с помощью микроскопа;
- склейка видео;
- получение изображения в супер-разрешении.

В общем виде алгоритм склейки панорамы можно сформулировать следующим образом. В самом начале требуется извлечь из видеопотока достаточное количество кадров. Это можно делать в режиме "онлайн", последовательно считывая все кадры и выбирая отдельные из них с необходимой частотой.

Таким образом существующий алгоритм, представляет нам первую ошибку, которая следует из основных условий алгоритма:

- ведется съемка неподвижного псевдориgidного объекта;
- ведется съемка объекта близкого к плоскому с достаточно большого расстояния;
- для всех положений камеры во время съемки выполнено требование: для всех точек изображения лучи, соединяющие эти точки с фокусом камеры, не совпадают друг с другом.



(2)

А именно склейка в процессе движения, что в задаче динамического построения карт, является неизбежным, следовательно, появляется потребность в внедрении нового и усовершенствованного типа обработки такого рода изображений, а именно внедрение алгоритма согласования графа проективных преобразований

Введем понятие единой системы координат. Под единой системой координат будем понимать такую систему координат, где одни и те же точки объекта с разных изображений будут иметь одинаковые координаты. Это требование можно выразить следующей формулой:

$$f(x)=y,$$

где f – это отображение определенное на общей части кадров и переводящее точки первого кадра в точки второго кадра, x – координаты точки в системе координат первого кадра, y – координаты точки в системе координат второго кадра.

В том случае, когда отображение f может быть корректно продолжено за область пересечения кадров, мы можем дополнить второй кадр информацией из первого. Таким образом будет получена карта, склеенная как мозаика, из двух или более кадров.

После нахождения проективных преобразований между соседними кадрами имеется первоначальная склейка, задающая однозначное расположение кадров в единой системе координат (см. рис. 4).

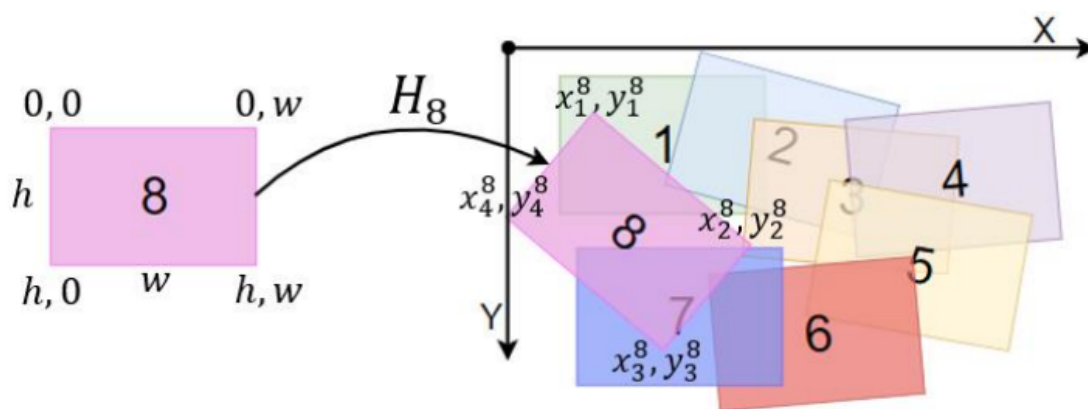
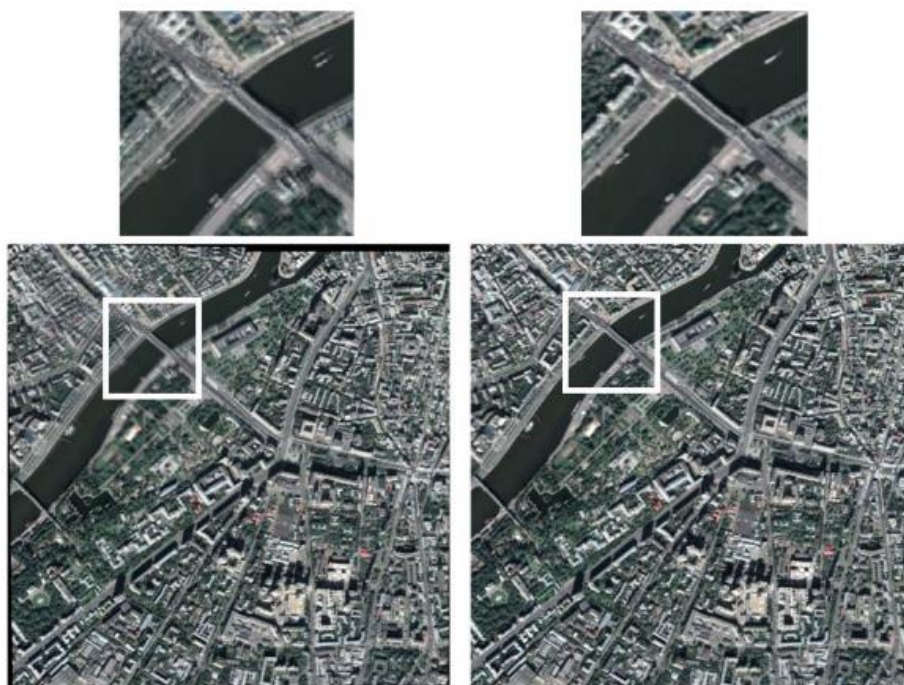


Рисунок 4 — Однозначное расположение кадра на карте

На сегодняшний день не существует универсального метода оценки качества склейки изображений. Как правило качество склейки оценивается экспертами органолептически, но для научных исследований предпочтительней иметь количественную, автоматически вычисляемую оценку качества.

В работе с ново-внедренным алгоритмом для количественной оценки качества склейки панорамы, имея изображение высокого разрешения, создать искусственное видео, кадрами которого являются проективно искаженные области исходного изображения. Проективно искажаются все кадры, за исключением первого, поскольку единая система координат задается относительно первого кадра. Далее эти кадры искусственного видео склеиваются в панораму, которая в дальнейшем сравнивается с исходным эталонным изображением. При данном подходе удастся избежать проблем разницы яркости полученной и эталонной склеек, а также искажений сцены.



Следующей не решённой проблемой является согласование карты, с результатами и материалами полученными в ходе исследования.



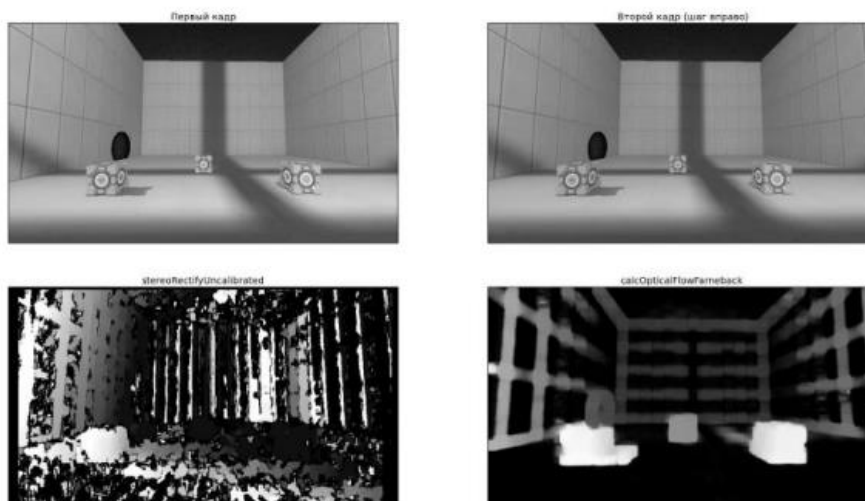
Само движение и корректировка камеры, в этой задаче не имеет значения, а вот материалы, которые в дальнейшем потребуют пост обработки на момент нахождения на ней объектов – есть очень сложная задача, которую разумным будет решить при помощи внедрения нейронной сети.

Самое сложное, что не видно, как можно решить этот вопрос механическим способом. Компьютерное зрение сталкивалось с соответствующими проблемами очень давно, с момента своего появления, и

периодически находило эффективные частные решения — так, мы можем опознать сдвинутый в сторону предмет, последовательно передвигая свой проверочный паттерн по всему изображению (чем успешно пользуются сверточные сети), можем справляться с отмасштабированными или повернутыми картинками с помощью признаков SIFT, SURF и ORB, но эффекты перспективы и поворот предмета в пространстве сцены — похоже, вещи качественно другого уровня. Здесь нам нужно знать, как предмет выглядит *со всех сторон*, получить его истинную трехмерную форму, иначе нам не с чем работать. Поэтому чтобы *распознавать* картинки, не нужно распознавать *картинки*. Они лживы, обманчивы и заведомо неполноценны. Они — не наши друзья.

Важный вопрос — как бы нам получать трехмерную модель всего, что мы видим? Еще более важный вопрос — как при этом обойтись без необходимости покупать лазерный пространственный сканер? Даже не столько по той причине, что нам жалко, а потому, что животные в процессе эволюции зрительной системы явно каким-то образом обошлись без него, одними только глазами, и было бы любопытно выяснить, как они так, хешение:

1. Взять два кадра, снятые с откалиброванной камеры.
2. Вместе с параметрами калибровки (матрицей камеры) положить их оба в функцию `stereoRectify`, которая выпрямит (ректифицирует) эти два кадра — это преобразование, которое искажает изображение так, чтобы точка и ее смещение оказывались на одной горизонтальной прямой.
3. Эти ректифицированные кадры мы кладем в функцию `stereoBM` и получаем карту смещений (`disparity map`) — такую картинку в оттенках серого, где чем пиксель ярче, тем большее смещение он выражает (по ссылке есть пример).
4. Полученную карту смещений кладем в функцию с говорящим названием `projectImageTo3D` (понадобится еще и матрица Q , которую в числе прочих мы получим на шаге 2). Получаем наш трехмерный результат.



Таким образом, реализовав два отличных решения, для основных проблем в этой области, и скооперировав их для парной и поэтапной обработки изображений, появляется возможность построить быструю и максимально точную систему, при помощи которой возможно будет строить карты включающее в себя, метатеги такие как объекты и привязки к координатам, в реальном времени для неизвестных или не детализированных участках земли.



Список литературы:

[1] Губин А.Ю., Ковин Р.В. Простой подход к задаче склейки перекрывающихся изображений в панораму // X Международная научно-

практическая конференция студентов, аспирантов и молодых ученых "Молодежь и современные информационные технологии", с. 79-81, 2012.

2 Drummond T., Rosten E. Machine Learning for high-speed corner detection // 9th European Conference on Computer Vision (ECCV), p. 430-443, 2006.

3 Lowe D.G. Distinctive Image Features from Scale-Invariant Keypoints // International Journal of Computer Vision, p. 91-110, 2004.

4 Bay H., Ess A., Yuitelaars T., Van Gool L. SURF: Speeded up robust features // Computer Vision and Image Understanding, v. 110, p. 346-359, 2008.

5 Martin A. Fischler, Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography // Comm. of the ACM, v. 24, p. 381-395, 1981.

6 Арлазаров В.Л., Булатов К.Б., Чернов Т.С. Метод нечеткого поиска изображений в больших объемах видеоданных // Системы высокой доступности, Т. 12, № 1, с. 53-58, 2016.

7 Skoryukina N. et al. Snapscreen: TV-stream frame search with projectively distorted and noisy query // 9th International Conference on Machine Vision (ICMV) — Proc. SPIE V. 10341, P. 103410Y, 2017.

8 Bouguet J.Y. Pyramidal implementation of the affine lucas kanade feature tracker: description of the algorithm // Intel corporation, V. 5, p. 1-10, 2001.

9 Newman P., Ho K. SLAM-loop closing with visually salient features // IEEE Proc. of International Conference on Robotics and Automation, p. 635-642, 2005.

10 Paalanen P., Kamarainen J.K., Kalviainen H. Image based quantitative mosaic evaluation with artificial video // Scandinavian Conference on Image Analysis, Springer (Berlin, Heidelberg), p. 470-479, 2009.