

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ФИЗИЧЕСКИХ И ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

УДК528.063.1

**ВЫЧИСЛЕНИЕ GPS-КООРДИНАТ ОБЪЕКТОВ,  
ОБНАРУЖЕННЫХ ПО МАТЕРИАЛАМ АЭРОФОТОСЪЁМКИ**

© 2023 Г.С. Финякин<sup>1\*</sup>, В.Б. Чемоданов<sup>2</sup>, А.А. Шацкий<sup>1</sup>

<sup>1</sup>Московский государственный технический университет имени Н.Э. Баумана,  
Москва, 105005, Россия

<sup>2</sup>Московский авиационный институт (национальный исследовательский университет),  
Москва, 121552, Россия

\*e-mail: shatskiyalex@gmail.com

Поступила в редакцию: 06.02.2023

После доработки: 08.02.2023

Принята к публикации: 14.02.2023

В данной статье рассмотрен алгоритм, позволяющий рассчитать GPS-координаты объекта, обнаруженного на снимках, полученных с беспилотного летательного аппарата (квадрокоптера). Разработанный алгоритм может быть использован в задачах обнаружения различных объектов и последующего нанесения их координат на карты. Авторами статьи описан подход к решению данной задачи, обозначены основные этапы алгоритма. Представлено описание подхода к написанию программы, реализованной на языке программирования C++ с использованием библиотеки с открытым исходным кодом OpenCV (библиотеки машинного зрения). Продемонстрированы результаты работы программы. Авторам удалось достичь точности вычисления GPS-координат объектов порядка одного метра, что сравнимо с точностью спутникового позиционирования квадрокоптера, с которого осуществляется аэрофотосъёмка подстилающей поверхности.

*Ключевые слова:* беспилотный летательный аппарат, квадрокоптер, поиск по шаблону (template matching), сшивание изображений (stitching), гомография.

**DOI:** 10.26583/vestnik.2023.249

**ВВЕДЕНИЕ**

В настоящее время актуальной является задача разработки методов поиска и обнаружения различных (в том числе опасных) объектов. Перспективным представляется метод визуального обнаружения с помощью беспилотных летательных аппаратов (квадрокоптеров). Суть метода заключается в том, чтобы производить подробную съёмку подстилающей поверхности с небольшой высоты в диапазоне 5–15 м, а затем, используя искусственную нейронную сеть (нейросеть), или какие-либо алгоритмы распознавания, находить объекты на полученных изображениях, а данные об их местоположении (GPS-координаты) наносить на карты. Снимки в формате JPG с современных квадрокоптеров, помимо изображения, содержат в себе «метаданные» – зашифрованную информацию о дате (date) и времени съёмки (time), положении коптера в пространстве (высота полёта, широта и долгота с точностью до шестого знака) и его

углах ориентирования: тангаж, крен и рыскание. К сожалению, высокая погрешность определения азимутальных углов ориентации квадрокоптера не позволяет с высокой степенью точности определить GPS-координаты объекта только по одному изображению. В данной работе предложен алгоритм, с помощью которого мы можем решить задачу обнаружения объектов и установления их положения в пространстве с погрешностью порядка одного метра. Для достижения этой цели мы используем только материалы съёмки с беспилотного летательного аппарата типа квадрокоптер с зашифрованными в этих материалах метаданными. Для автоматического обнаружения объектов используются искусственные нейронные сети (нейросети), предварительно обученные на их распознавание. Современные нейросети могут не только обнаруживать объекты на текущем кадре, но и выдавать пиксельные координаты этих объектов.

В последние годы нейросети стали активно использоваться в самых разнообразных областях науки и техники (см., например, [1–7]). Однако в этой работе мы уделим внимание не нейросетевому обнаружению интересующих нас объектов, а последующему вычислению их GPS-координат.

Общая идея алгоритма заключается в нахождении матриц гомографии, связывающих известные пиксельные координаты искомого объекта на изображении и необходимые нам GPS-координаты этого же объекта. Гомография – взаимно-однозначное преобразование всех точек из одной физической плоскости (из одной системы декартовых координат) ко всем точкам в другой физической плоскости (в другой системе декартовых координат). При таком преобразовании каждой точке в первой физической плоскости будет взаимно-однозначно соответствовать другая точка из второй физической плоскости. Гомография полностью определяется матрицей 3 на 3. Для вычисления этой матрицы необходимо знать как минимум четыре пары точек соответствия для каждой из систем координат [8, 9]. Если камера квадрокоптера смотрит вертикально вниз, то вертикальная ось, проведенная через камеру, пересекает центр снимаемого кадра. Поэтому мы можем считать, что  $x_u$ -GPS-координаты коптера в момент съёмки кадра соответствуют точке центра этого кадра. Таким образом, если мы «сошьём» четыре соседних изображения в одно, то из данных о центрах каждого из исходных изображений мы сможем получить матрицу гомографии, связывающую GPS-координаты и пиксельные координаты сшитого изображения. Сшиваемые изображения далее мы будем называть «фреймами». Изображение, на котором нейросеть обнаружила интересующий нас объект, обозначим как «нулевой фрейм». Изображение, полученное в результате сшивания соседних фреймов, назовём «панно».

Введём обозначения: «малыми пикселями» ( $pix$ ) будем обозначать пиксельные координаты на отдельных исходных фреймах; «большими пикселями» ( $PIX$ ) будем обозначать пиксельные координаты на панно, которое сшито из исходных фреймов; матрицу преобразования больших пикселей в GPS-координаты обозначим как  $(PIX - GPS)$ . Для сшивания соседних фреймов к ним применяются аффинные преобразования – такие преобразования на плоскости, при которых параллельные прямые переходят в параллельные прямые, пересекающиеся – в пересекающиеся, скрещивающиеся – в скрещивающи-

еся [10]. При таких преобразованиях иногда происходит взаимное перекрытие центров фреймов, что не всегда позволяет увидеть пиксельные координаты интересующих нас точек на общем панно. По этой причине мы не можем достоверно определить положение объекта на панно, несмотря на то, что его пиксельные координаты на нулевом фрейме нам известны. Чтобы решить эту проблему, мы вычисляем еще одну матрицу гомографии, связывающую малые пиксели и большие пиксели ( $pix - PIX$ ). Для получения матрицы гомографии, связывающей нулевой фрейм и панно, мы наносим специальную разметку на каждый из четырёх исходных фреймов перед сшиванием их в общее панно. Это делается для того, чтобы, найдя затем метки на панно (например, методом поиска по шаблону), восстановить по этим меткам геометрию фреймов на панно и затем вычислить большие пиксельные координаты ( $PIX$ ) центров каждого из сшиваемых фреймов. Получив две матрицы гомографии, связывающие малые ( $pix$ ) и большие ( $PIX$ ) пиксели, а также большие пиксели ( $PIX$ ) и GPS-координаты, можем найти гомографию, связывающую малые пиксели и GPS-координаты по формуле:

$$(pix - GPS) = (pix - PIX) \cdot (PIX - GPS). \quad (1)$$

В формуле (1) стоит матричное произведение матриц:  $(pix - PIX)$  – матрица преобразования из «малых пикселей» в «большие пиксели»,  $(PIX - GPS)$  – матрица преобразования из «больших пикселей» в GPS-координаты,  $(pix - GPS)$  – матрица преобразования из «малых пикселей» в GPS-координаты.

Итак, полный алгоритм вычисления GPS-координат объекта состоит из следующих этапов

1. Обнаружение нейросетью объектов на изображениях с коптера.
2. Выбор комбинации изображений для сшивания.
3. Разметка и сшивание изображений.
4. Обнаружение элементов разметки и поиск центров сшитых изображений на панно.
5. Построение матриц гомографии, связывающих пиксельные координаты точек на нулевом фрейме с их GPS-координатами.
6. Вычисление GPS-координат найденных объектов.

В данной статье описывается создание и результаты работы программы по методике, описанной ранее. Программа написана на языке C++ с использованием библиотеки с открытым исходным кодом OpenCV (библиотеки машин-

ного зрения). Считаем, что информация о положении объекта представлена в виде малых пиксельных координат (pix) центра объекта на нулевом фрейме.

### ВЫБОР КОМБИНАЦИЙ ИЗОБРАЖЕНИЙ

Доступная память квадрокоптера может достигать одного терабайта, а значит содержать более 60000 изображений, которые требуется обработать после возвращения коптера. Число групп перестановок из  $n$  элементов по  $k$  штук в группе вычисляется по формуле:

$$C_k^n = \frac{n!}{k!(n-k)!} = \frac{60000!}{4!(60000-4)!} \approx 5,4 \cdot 10^{17}. \quad (2)$$

С учётом того, что разметка четырёх фреймов, их сшивание в панно, поиск меток на панно и вычисление матриц гомографии занимает несколько секунд, вычислительная задача для всех возможных комбинаций  $C_k^n$  является непосильной и поэтому необходимо провести предварительную фильтрацию данных (отбор среди всех только самых необходимых изображений). Например, будем сшивать не все изображения, а только те из них, для которых:

- углы рыскания равны углу рыскания нулевого фрейма с точностью до введённой нами погрешности:

$$\psi = \psi_0 + \delta\psi; \quad (3)$$

- высота полёта при съёмке фрейма (следовательно, и масштаб изображения) с точностью до введённой нами погрешности совпадает с высотой съёмки нулевого фрейма:

$$H = H_0 + \delta H; \quad (4)$$

- Расстояние между фреймами такое, чтобы с одной стороны фреймы не перекрывались полностью, а с другой стороны – чтобы фреймы имели полосу пересечения с минимальной требуемой для сшивания площадью.

- искомый объект попадает в четырёхугольник, образованный центрами сшиваемых фреймов.

Если фреймов, из которых будут формироваться панно, все ещё слишком много, то стоит дополнительно провести фильтрацию по площади четырёхугольника, образованного центрами фреймов, предварительно переведя GPS-координаты в метры. Для сшивки мы выбираем только те фреймы (удовлетворяющие всем вышеперечисленным условиям), для которых эта площадь максимальна.

### РАЗМЕТКА И СШИВАНИЕ ФРЕЙМОВ

Четыре точки на панно, соответствующие центрам сшиваемых фреймов, позволят нам построить матрицу гомографии, преобразующую большие пиксели (PIX) в GPS-координаты (PIX – GPS). Чтобы после сшивания найти центры фреймов на панно, мы до сшивания наносим метки на исходные фреймы, располагая их на диагоналях фреймов, с некоторым фиксированным отступом от углов фреймов. Каждая метка представляет из себя комбинацию из четырёх концентрических окружностей красного и зелёного цветов, где красный цвет соответствует нулю, а зелёный – единице в двоичной системе счисления (принцип двоичного кодирования показан на рис. 1). Первое число (внешняя пара окружностей) – номер фрейма в сшиваемом панно, второе число (внутренняя пара окружностей) – номер угла на текущем фрейме.

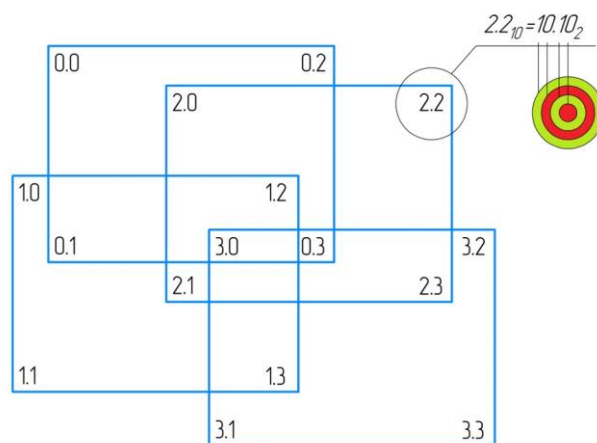


Рис. 1. Принцип кодирования меток углов фреймов

Нулевой фрейм всегда имеет номер кадра, равный нулю (00). Остальные три фрейма формируются из перестановок по три элемента из всех выбранных для сшивания фреймов.

Две внутренние окружности меток являются номером угла фрейма в двоичной системе счисления. Нулевой угол (00) – левый верхний, первый угол (01) – левый нижний, второй угол (10) – правый верхний, третий угол (11) – правый нижний.

В данной работе для сшивания используется метод «stitcher» библиотеки компьютерного зрения с открытым исходным кодом OpenCV.

Сшиватель (stitcher) может работать в двух режимах.

1. Панорама (PANORAMA) реализует модель гомографии. Эта модель применяет сферическое преобразование (SphericalWarper), так как пытается подчинить панно модели объектива «рыбий глаз».

2. SCAN – аффинная модель преобразования сшиваемых изображений.

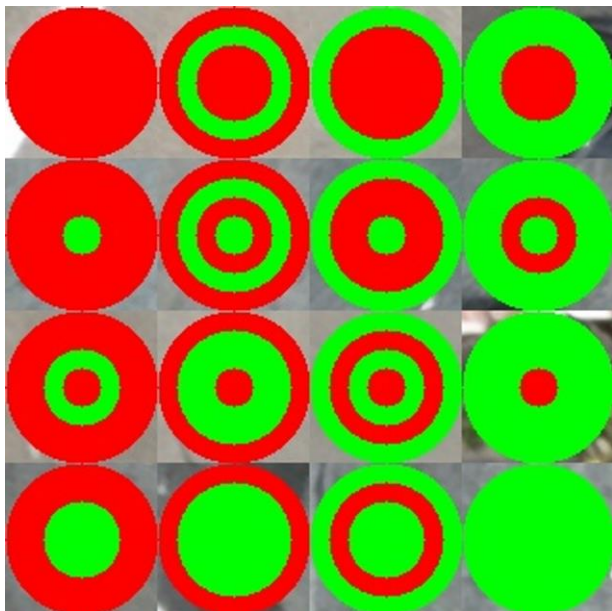


Рис. 2. Угловые метки фреймов на панно

Гомографическая модель полезна для создания фотопанорам, снятых камерой, а аффинная модель может использоваться для сшивания сканов и объектов, снятых специализированными устройствами (в том числе дронами). Исходя из этого, для сшивания будем применять метод SCAN.

Обнаружение координат меток на панно после сшивания осуществляется методом поиска по шаблону (matchTemplate в OpenCV). Шаблон поиска – изображение самой метки. Для минимизации ошибок поиска меток, а также исключения ложного срабатывания меток, шаблон метки создаётся как скопированная с фрейма квадратная область вокруг центра будущей метки (размером с метку) плюс наложенные сверху на этот квадрат цветные окружности самой метки. Полный набор шаблонов меток, которые мы ищем на панно, приведен на рис. 2.

#### ОБНАРУЖЕНИЕ ЭЛЕМЕНТОВ РАЗМЕТКИ И ПОИСК ЦЕНТРОВ СШИТЫХ ФРЕЙМОВ НА ПАННО

Теперь, когда все метки нанесены на фреймы и панно из фреймов сшито, нам необходимо выполнить ряд действий.

1. Применить инструмент TemplateMatching (поиск по шаблону) к панно. Шаблонами являются метки углов. Так как сшивание в режиме SCAN применяет аффинные преобразования, то квадраты, образованные метками, могут повернуться и изменить свой масштаб. Однако при условии, что коптер не меняет высоты при съёмке и при известном отношении сторон каждого фрейма, можем считать, что метки углов для каждого фрейма не меняют своего масштаба, их поворот не влияет на вид круглой метки, а прямоугольники, образованные метками на фреймах, останутся прямоугольниками. Тем самым поиск меток по шаблону не приведет к ложным срабатываниям. Когда найдены все видимые метки, сохранившиеся после сшивки панно, мы можем найти центры сшиваемых фреймов в больших пикселях (PIX) при условии, что найдены хотя бы по две метки из четырёх для каждого из сшиваемых фреймов.

2. Если на панно найдены хотя бы две метки нулевого фрейма, то оставшиеся не найденными метки можно восстановить по двум (или трём) найденным. В итоге мы получаем четыре пары точек в координатах, определённых и на нулевом фрейме в малых пикселях (pix), и на панно в больших пикселях (PIX). А значит, мы можем построить матрицу гомографии, преобразующую малые пиксели в большие пиксели (pix – PIX).

3. После нахождения центра нулевого фрейма аналогично находим также центры остальных сшиваемых фреймов. При удачной комбинации фреймов, при сшивании мы получим четыре центра фреймов на панно, указывающих положение коптера в момент съёмки. Тогда мы можем построить матрицу гомографии (PIX – GPS).

В результате получаем две матрицы гомографии, позволяющие связать малые пиксельные координаты (pix) объекта на нулевом фрейме с большими пиксельными координатами (PIX) объекта на панно, а большие пиксели связаны матрицей гомографии с GPS-координатами. По формуле (1) вычисляем матрицу гомографии, преобразующую малые пиксели в GPS-координаты (pix – GPS).

Если перед вычислением матрицы гомографии отнормировать на диапазон от нуля до единицы координаты всех точек в обеих системах координат, то погрешности этих преобразований значительно снизятся (а точность повысится). В данном алгоритме точность определения координат выросла на порядок после нормирования GPS-координат, по которым составлялась



матрица гомографии. Критерий точности оценки – разница между GPS-координатой искомого объекта на выходе алгоритма и GPS-координатой этого объекта по данным сервиса Яндекс-карты. При наиболее удачных комбинациях фреймов точность определения положения точки достигает десятых долей метра.

Стоит отметить, что не всегда при сшивании удаётся найти по две метки каждого из четырёх сшиваемых фреймов. В случае, когда удалось определить центры только трёх сшиваемых фреймов, мы можем создать фиктивную точку, отразив одну из точек относительно прямой, образованной двумя другими. Тоже самое следует сделать с соответствующими точкам GPS-координатами. При тестировании программы погрешность внутри полученного четырёхугольника будет порядка 1,5 м, а за пределами этого четырёхугольника погрешность может быстро возрасти до десятков метров даже при небольшом удалении от этого четырёхугольника – особенность работы гомографии.

## РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ

Продемонстрируем возможности реализованной программы.

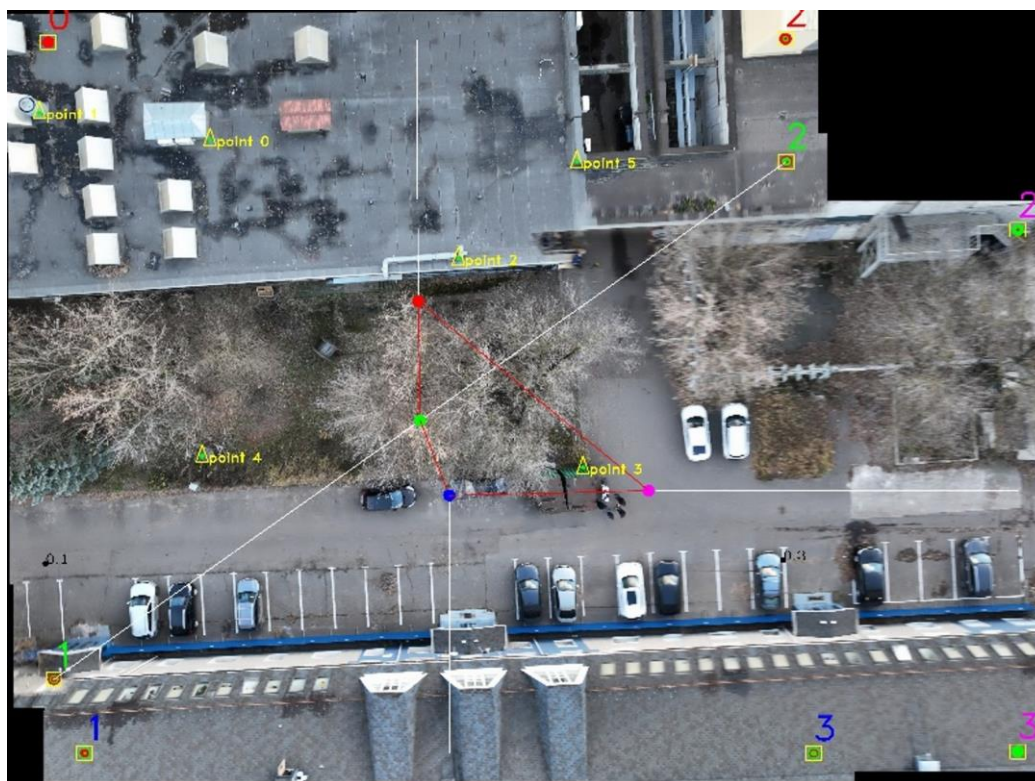
Выберем несколько характерных, узнаваемых точек объектов на нулевом фрейме (таких, чтобы мы без труда смогли найти их на Яндекс-картах). Пиксельные координаты ( $pix$ ) выбранных объектов на нулевом фрейме (рис. 3) отмечены желтыми метками. Координаты этих точек в малых пикселях ( $pix$ ): point0(1310, 870); point1(210, 680); point2(2915, 1690); point3(3725, 3120); point4(1270, 3030); point5(3675, 1030).

В результате работы программы произведена сшивка четырёх фреймов, поиск по шаблону шестнадцати нанесенных меток (рис. 4), расчёт координат центров фреймов на панно (большие точки красного, зелёного, синего и фиолетового цветов для 0, 1, 2 и 3 фреймов соответственно), зелёные точки в треугольной рамке – искомые точки после преобразования ( $pix - PIX$ ). После расчёта матриц гомографии ( $pix - PIX$ ) и ( $PIX - GPS$ ) и вычисления матрицы ( $pix - GPS$ ) мы нашли GPS-координаты всех шести искомым объектов.

GPS-координаты точек, полученные в результате применения гомографии ( $pix - GPS$ ): point0(55.811575, 37.804393); point1(55.811567, 37.804547); point2(55.811642, 37.804143); point3(55.811796, 37.803998); point4(55.811789, 37.804422); point5(55.811568, 37.804009).

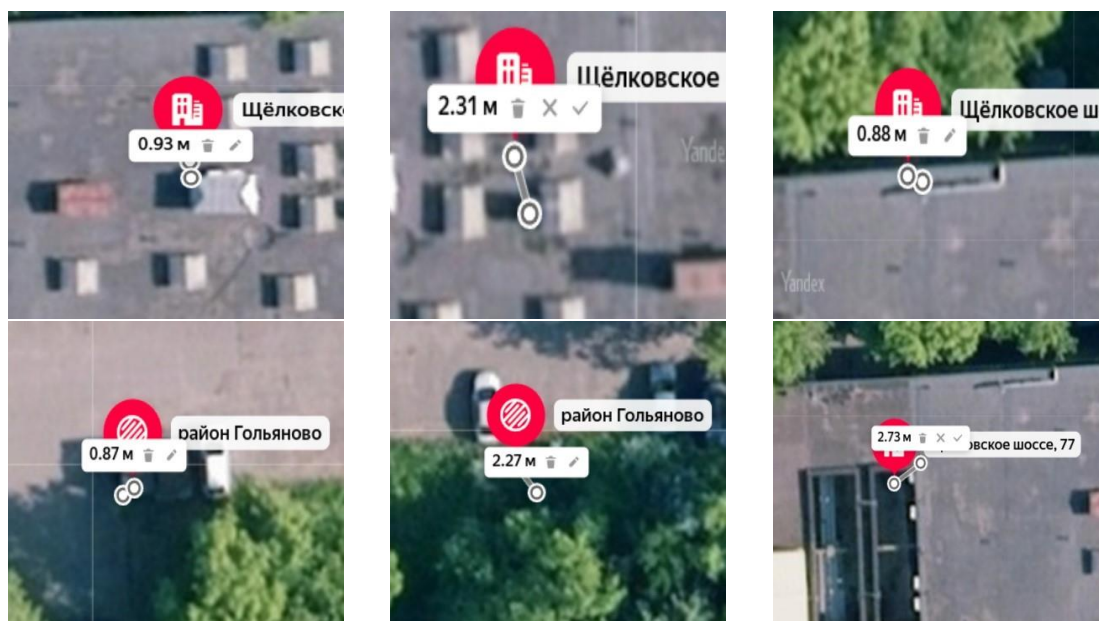


Рис. 3. Нулевой фрейм с шестью точками – объектами



**Рис. 4.** Панно: желтой квадратной рамкой обведены найденные углы разметки; цифры характеризуют номер метки, их цвет – номер фрейма (0 – красный, 1 – зелёный, 2 – синий, 3 – фиолетовый). Красной линией соединены точки центров исходных фреймов. Зелёные точки в желтых треугольниках – искомые объекты на нулевом фрейме, перенесённые на панно с помощью преобразования гомографии ( $pix - PIX$ )

Для оценки качества работы программы полученные координаты вводим в Яндекс-карты. Затем находим расстояние между полученной координатой и отмеченной на нулевом фрейме характерной точкой (рис. 5).



**Рис. 5.** Результат работы программы: точка 0 (слева сверху) – угол флигеля; точка 1 (в центре сверху) – основание трубы; точка 2 (справа сверху) – фитинг трубы; точка 3 (слева внизу) – угол контейнера; точка 4 (в центре внизу) – крышка люка; точка 5 (справа внизу) – угол крыши над внутренним двором



Заметно, что точность тем выше, чем ближе точка к четырёхугольнику, образованному точками, используемыми при построении гомографии.

## ЗАКЛЮЧЕНИЕ

Разработан и реализован алгоритм на языке C++ с использованием библиотек OpenCV, позволяющий в автоматическом режиме находить GPS-координаты объекта, обнаруженного на снимках подстилающей поверхности с квадрокоптера. Алгоритм показывает высокую точность, сравнимую с точностью спутникового позиционирования коптера.

Точность позиционирования квадрокоптера:

- в вертикальной плоскости –  $\pm 0,1$  м (визуальное позиционирование);  $\pm 0,5$  м (спутниковое позиционирование);
- в горизонтальной плоскости –  $\pm 0,3$  м (визуальное позиционирование);  $\pm 0,5$  м (высокоточная система позиционирования).

Одной из особенностей метода является требование к квалификации оператора, управляющего коптером. Сшивка изображений разного масштаба и разной ориентации в пространстве затруднительна, т.е. для стабильной работы метода коптер должен перемещаться не меняя высоты и угла поворота (рыскания) на протяжении всего полёта.

Мы экспериментально подтвердили известный факт о том, что применение гомографии даёт наиболее корректный результат при условии, что искомые точки лежат внутри четырёхугольника, вершины которого используются для построения матрицы гомографии.

## СПИСОК ЛИТЕРАТУРЫ И ИНТЕРНЕТ-ИСТОЧНИКОВ

1. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. М.: Горячая линия-Телеком, 2002. С. 382.
2. Hadsell R., Chopra S. and LeCun Y. Dimensionality reduction by learning an invariant mapping [электронный ресурс]. URL: <http://yann.lecun.com/exdb/publis/pdf/hadsellcho-pra-lecun-06.pdf> 2006 (дата обращения: 29.01.2023).
3. Воронцов К.В. Нейронные сети, видеокурс [электронный ресурс]. URL: <https://www.youtube.com/watch?v=WjwA5DqxL-c> (дата обращения: 29.01.2023).
4. Johnson J. Li F.F. and Yeung S. Convolutional neural networks [электронный ресурс]. URL: <http://cs231n.stanford.edu/syllabus.html> (дата обращения: 29.01.2023).
5. Bochkovskiy Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. Optimal speed and accuracy of object detection [электронный ресурс]. URL: <https://arxiv.org/pdf/2004.10934.pdf> (дата обращения: 29.01.2023).
6. Wang Chien-Yao, Bochkovskiy A., and Hong-Yuan Mark Liao. Scaling cross stage partial network // Proceedings of the IEEE/CVF «Conference on Computer Vision and Pattern Recognition (CVPR)». 2021. P. 13029.
7. Шацкий А.А., Евгеньев И.Ю. Нейросетевая астрономия как новый инструмент наблюдения ярких и компактных объектов // ЖЭТФ. 2019. Т. 155. № 4. С. 693.
8. Байер Р. Линейная алгебра и проективная геометрия. М.: Иностранной литературы, 1955. С. 399.
9. Обработка и анализ цифровых изображений с примерами на LabVIEW и IMAQ Vision / Ю.В. Визильтер, С.Ю. Желтов, В.А. Князь, А.Н. Ходарев. М.: ДМК Пресс, 2007. С. 464.
10. Корн Т., Корн Г. Справочник по математике. М.: Наука, 1973. С. 832.

---

Vestnik Natsional'nogo issledovatel'skogo yadernogo universiteta «MIFI», 2023, vol. 12, no. 1, pp. 20–27

---

## CALCULATION OF GPS COORDINATES OF OBJECTS DETECTED FROM AERIAL PHOTOGRAPHY

G.S. Finyakin<sup>1\*</sup>, V.B. Chemodanov<sup>2</sup>, A.A. Shatsky<sup>1</sup>

<sup>1</sup>Bauman Moscow State University, Moscow, 105005, Russia

<sup>2</sup>Moscow Aviation Institute (National Research University), Moscow, 121552 Russia

\*e-mail: [grishafinyakin@mail.ru](mailto:grishafinyakin@mail.ru)

Received February 6, 2023; revised February 8, 2023; accepted February 14, 2023

This article discusses an algorithm that allows you to calculate the GPS coordinates of an object detected in images taken from an unmanned aerial vehicle (quadcopter). The developed algorithm can be

used in the tasks of detecting various objects and then plotting their coordinates on maps. The authors of the article describe an approach to solving this problem, identify the main stages of the algorithm. An approach to writing a program implemented in the C++ programming language using the OpenCV open source library (machine vision library) is described. The results of the program are demonstrated. The authors managed to achieve the accuracy of calculating the GPS coordinates of objects of the order of one meter, which is comparable to the accuracy of satellite positioning of a quadcopter from which aerial photography of the underlying surface is carried out.

**Keywords:** unmanned aerial vehicle, quadcopter, template matching, stitching, homography.

## REFERENCES

1. Kruglov V.V., Borisov V.V. *Iskustvennie neyronnie sety. Teoriya i praktika* [Artificial neural networks. Theory and practice]. Moscow, Goryachaya liniya-Telecom Publ., 2002. 382 p.
2. Hadsell R., Chopra S., LeCun Y. Dimensionality reduction by learning an invariant mapping [electronic resource]. Available at: <http://yann.lecun.com/exdb/publis/pdf/hadsell-cho-pra-lecun-06.pdf> (accessed 29.01.2023).
3. Vorontsov K.V. *Neyronnie sety, videokurs* [neural networks, video course: electronic resource]. Available at: <https://www.youtube.com/watch?v=WjwA5DqxL-c> (accessed 29.01.2023).
4. Li F.F., Johnson J., Yeung S. Convolutional neural networks [electronic resource]. Available at: <http://cs231n.stanford.edu/syllabus.html> (accessed 29.01.2023).
5. Bochkovskiy A. and Chien-Yao Wang, Hong-Yuan Mark Liao. Optimal speed and accuracy of object detection [electronic resource]. Available at: <https://arxiv.org/pdf/2004.10934.pdf> (accessed 29.01.2023).
6. Wang, Chien-Yao and Bochkovskiy, Alexey and Liao, Hong-Yuan Mark. Scaling cross stage partial network. Proceedings of the IEEE/CVF «Conference on Computer Vision and Pattern Recognition (CVPR)», 2021. P. 13029.
7. Shatskit A.A., Evgeniev I.Yu. Neural network astronomy as a new tool for observing bright and compact objects. *Journal of Experimental and Theoretical Physics*. 2019. Vol. 128. P. 592–598.
8. Baer R. *Linear algebra and projective geometry*. Dover Publications, Publ. 2005. 336 p.
9. Hodarev. *Obrabotka i analiz tsifrovih izobrazeniy s primeramy na LabVIEW y IMAQ Vision* / Yu.Yu. Vizilter, S.Yu. Jeltov, V.A., Knyaz A.H. [Processing and analysis of digital images with examples on LabVIEW and IMAQ Vision]. M.: DMK Press, Publ., 2007. 464 p.
10. Korn G., Korn T. *Spravochnik po matematike* [handbook of mathematics]. M.: Nauka Publ., 1973. 832 p.